University of Groningen

Linear-Time Complete Positivity Detection and Decomposition of Sparse Matrices

Dickinson, Peter J. C.; Duer, Mirjam

Published in:
SIAM Journal on Matrix Analysis and Applications

DOI:
10.1137/110848177

IMPORTANT NOTE: You are advised to consult the publisher's version (publisher's PDF) if you wish to cite from it. Please check the document version below.

Document Version
Publisher's PDF, also known as Version of record

Publication date:
2012

Link to publication in University of Groningen/UMCG research database

Citation for published version (APA):
Dickinson, P. J. C., & Duer, M. (2012). Linear-Time Complete Positivity Detection and Decomposition of Sparse Matrices. *SIAM Journal on Matrix Analysis and Applications*, *33*(3), 701-720. https://doi.org/10.1137/110848177

Copyright
Other than for strictly personal use, it is not permitted to download or to forward/distribute the text or part of it without the consent of the author(s) and/or copyright holder(s), unless the work is under an open content license (like Creative Commons).

The publication may also be distributed here under the terms of Article 25fa of the Dutch Copyright Act, indicated by the "Taverne" license. More information can be found on the University of Groningen website: https://www.rug.nl/library/open-access/self-archiving-pure/taverne-amendment.

Take-down policy
If you believe that this document breaches copyright please contact us providing details, and we will remove access to the work immediately and investigate your claim.

Downloaded from the University of Groningen/UMCG research database (Pure): http://www.rug.nl/research/portal. For technical reasons the number of authors shown on this cover page is limited to 10 maximum.

Download date: 25-08-2022

SIAM J. MATRIX ANAL. APPL.
Vol. 33, No. 3, pp. 701–720
© 2012 Society for Industrial and Applied Mathematics

Downloaded 10/23/12 to 129.125.63.113. Redistribution subject to SIAM license or copyright; see http://www.siam.org/journals/ojsa.php

LINEAR-TIME COMPLETE POSITIVITY DETECTION AND DECOMPOSITION OF SPARSE MATRICES*

PETER J. C. DICKINSON† AND MIRJAM DÜR‡

Abstract. A matrix X is called completely positive if it allows a factorization $X = \sum_{b\in\mathcal{B}} bb^{\mathsf{T}}$ with nonnegative vectors b. These matrices are of interest in optimization, as it has been found that several combinatorial and quadratic problems can be formulated over the cone of completely positive matrices. The difficulty is that checking complete positivity is $\mathcal{NP}$-hard. Finding a factorization of a general completely positive matrix is also hard. In this paper we study complete positivity of matrices whose underlying graph possesses a specific sparsity pattern, for example, being acyclic or circular, where the underlying graph of a symmetric matrix of order n is defined to be a graph with n vertices and an edge between two vertices if the corresponding entry in the matrix is nonzero. The types of matrices that we analyze include tridiagonal matrices as an example. We show that in these cases checking complete positivity can be done in linear-time. A factorization of such a completely positive matrix can be found in linear-time as well. As a by-product, our method provides insight on the number of different minimal rank-one decompositions of the matrix.

Key words. completely positive, sparse matrix, linear-time

AMS subject classifications. 15A23, 15B48, 65F30, 65F50

DOI. 10.1137/110848177

1. Introduction.

The idea of completely positive matrices is of interest due to its uses in optimization [26, 27]. It has been found that some $\mathcal{NP}$-hard problems can be reformulated as linear optimization problems over the cone of completely positive matrices [7, 8, 13, 15]. From these important applications we are motivated to study properties of completely positive matrices, in order to build up intuition on them.

We say that a matrix X is *completely positive* if there exists a finite set $\mathcal{B}$ contained in the nonnegative orthant such that

$$X = \sum_{b\in\mathcal{B}} bb^{\mathsf{T}}.$$

In this case we say that $\mathcal{B}$ is a *rank-one decomposition set* of X.

We can immediately see that if a matrix is completely positive, then it must be positive semidefinite and nonnegative.

One property that can be considered with regard to complete positivity is the *cp-rank*. The cp-rank of a completely positive matrix X is defined as

$$\text{cp-rank}(X) := \min\{|\mathcal{B}| \mid \mathcal{B} \text{ is a rank-one decomposition set of } X\}.$$

If X is a completely positive matrix, then from Carathéodory's theorem we have that cp-rank$(X) \leq \frac{1}{2}n(n+1)$, where n is the order of the matrix. This bound can be

*Received by the editors September 16, 2011; accepted for publication (in revised form) by P. Absil March 7, 2012; published electronically July 3, 2012.
http://www.siam.org/journals/simax/33-3/84817.html
†Johann Bernoulli Institute of Mathematics and Computer Science, University of Groningen, 9700 AK Groningen, The Netherlands (P.J.C.Dickinson@rug.nl).
‡Department of Mathematics, University of Trier, 54286 Trier, Germany (duer@uni-trier.de). This author was supported by the Netherlands Organisation for Scientific Research (NWO) through Vici grant 639.033.907.

Copyright © by SIAM. Unauthorized reproduction of this article is prohibited.

Downloaded 10/23/12 to 129.125.63.113. Redistribution subject to SIAM license or copyright; see http://www.siam.org/journals/ojsa.php

improved to cp-rank$(X) \leq \frac{1}{2}k(k+1)-1$, where $2 \leq k = \text{rank}(X) \leq n$, as was shown in [1, 20]. It has also been conjectured in [12] that cp-rank$(X) \leq \max\{n, \lfloor \frac{n^2}{4} \rfloor\}$, which is proved true for $n \leq 4$. If a matrix is not completely positive, then its cp-rank is defined to be infinite [4].

We define a minimal rank-one decomposition set of a completely positive matrix X to be a rank-one decomposition set $\mathcal{B}$ such that $|\mathcal{B}| = \text{cp-rank}(X)$. In general this minimal rank-one decomposition set is not unique, as we shall see in section 8. Properties of a rank-one decomposition of a completely positive matrix have been studied previously in [9, 14, 21], and in this paper we will be investigating minimal rank-one decompositions for sparse matrices.

The set of completely positive matrices is a proper cone, and the dual of this cone is the cone of copositive matrices, where a symmetric matrix A is copositive if and only if $v^\mathsf{T} A v \geq 0$ for all nonnegative vectors v. Surveys of both of these cones and their applications are provided in [5, 13, 15].

Checking whether a matrix is completely positive has been shown to be an $\mathcal{NP}$-hard problem, while checking whether a matrix is copositive has been shown to be a co-$\mathcal{NP}$-complete problem [10, 24]. In spite of the complexity of checking whether a matrix is copositive, for special cases there are efficient algorithms, even ones that run in linear-time. For example, in [6] a method was discussed for checking whether a tridiagonal matrix is copositive in linear-time, while in [16] this was extended to acyclic matrices.

In this paper we will similarly consider special cases when we are able to check whether a matrix is completely positive in linear-time and, if so, find a minimal rank-one decomposition set for it.

Related work. While the problem of finding a factorization of a general completely positive matrix is still unsolved, the problem of factorizing matrices with special structure has been studied before. Kaykobad [18] proved that if a matrix is positive semidefinite and nonnegative and diagonally dominant, then it is completely positive. He also gives an easy procedure for constructing a factorization. Berman and coauthors [2, 3] considered matrices whose underlying graph has a special structure. In [3] they characterize completely positive matrices whose underlying graph is acyclic. They do not, however, use this characterization for an algorithmic factorization procedure. In [2] they study matrices with bipartite graphs and state a simple algorithmic procedure to factorize.

The complete positivity of circular matrices has previously been studied in [29, 30]. In [29] Xu and Li characterize completely positive circular matrices of order greater than 3, but it seems unclear how this characterization can actually be used to check algorithmically whether a circular graph is completely positive. In [30] Zhang and Li give conditions for complete positivity of a circular matrix in terms of its comparison matrix. The proof of their result includes a method for finding a minimal rank-one decomposition set; however, this was a relatively complicated method and not subjected to much analysis.

Li, Kummert, and Frommer [20] show how—starting from an arbitrary factorization of an $n \times n$ matrix X—one can obtain a smaller factorization $X = \sum_{b \in \mathcal{B}} bb^\mathsf{T}$ with $|\mathcal{B}| = \frac{1}{2}n(n+1)-1$.

Shaked-Monderer [28] considers matrices which are positive semidefinite, nonnegative with rank r and have an $r \times r$ principal submatrix that is diagonal. This corresponds to the graph of the matrix having a maximal stable set of size r. Such a matrix is shown to be completely positive, and a factorization is immediate from the

Copyright © by SIAM. Unauthorized reproduction of this article is prohibited.

Downloaded 10/23/12 to 129.125.63.113. Redistribution subject to SIAM license or copyright; see http://www.siam.org/journals/ojsa.php

proof. Kalofolias and Gallopoulos [17] extend this result and construct a factorization of completely positive rank-two matrices.

Finally, Dong, Lin, and Chu [11] provide a heuristic method for the so-called (nonsymmetric) nonnegative rank factorization, i.e., finding a decomposition $X = UV$ of X with U, V nonnegative but not necessarily $U = V^T$ (which would correspond to our setting). Their procedure can be applied to completely positive matrices and would be able to heuristically check whether cp-rank(X) = rank(X) and, if affirmative, compute a factorization of X.

Our paper will provide a unified approach to these ideas and extend the domain of cases where a factorization can be found. We will present an algorithmic method for this and pay special attention to the run-time of this algorithm. One of our results will be that—as in the copositive case studied in [6, 16]—for tridiagonal and acyclic matrices complete positivity can be checked in linear-time. Our method could also be used for preprocessing a matrix which we wish to test for complete positivity in order to reduce the problem.

Notation. We will be using the following notation for sets of vectors:

the set of real n-vectors $= \mathbb{R}^n$,
the set of nonnegative n-vectors $= \mathbb{R}^n_+$,
the set of strictly positive n-vectors $= \mathbb{R}^n_{++}$,
the set of integer n-vectors $= \mathbb{Z}^n$,

where we shall suppress the "n" if the dimension is equal to one.

We will also be using the following notation for sets of matrices:

the cone of $n \times n$ symmetric matrices $= \mathcal{S}^n$,
the cone of $n \times n$ symmetric positive semidefinite matrices $= \mathcal{S}^n_+$,
the cone of $n \times n$ symmetric nonnegative matrices $= \mathcal{N}^n$,
the cone of $n \times n$ completely positive matrices $= \mathcal{C}^{*n}$,

where we shall suppress the "n" if the dimension is obvious from the context.

Additionally for a matrix $A \in \mathcal{S}^n$ we define $G(A)$ to be the *underlying graph* of A such that we have $G(A) = (V, E)$ with $V = \{1, \ldots, n\}$ and $E = \{(ij) \mid i < j,\ (A)_{ij} \neq 0\}$. When we talk of an index i of A having a certain degree, we are referring to the degree of the vertex i in the graph $G(A)$ having this degree. Similarly when we refer to graph properties of a matrix A, for example, being acyclic, circular, or connected, we are referring to the properties of the graph $G(A)$. Recall that a circular graph is a graph consisting of a single cycle. We will use the phrase *component submatrix* for a principal submatrix whose graph is a connected component in the graph of the full matrix. Finally a weighted-graph of A refers to $G(A)$ with weights on the vertices and edges equal to the corresponding values in A. We use this in order to be able to consider certain structures in a matrix with more ease.

2. Rank-one decomposition.

We will now look at some basic properties of (minimal) rank-one decomposition sets of sparse completely positive matrices.

First we note, from the definition, that $\mathcal{C}^* \subseteq \mathcal{N} \cap \mathcal{S}_+$. From this we see that a completely positive matrix must be nonnegative, and if an on-diagonal element of a completely positive matrix is equal to zero, then all the off-diagonal elements on this row and column must also be equal to zero. We can in fact check whether these necessary conditions hold in linear-time.

Copyright © by SIAM. Unauthorized reproduction of this article is prohibited.

Downloaded 10/23/12 to 129.125.63.113. Redistribution subject to SIAM license or copyright; see http://www.siam.org/journals/ojsa.php

We now look at how the graph of a completely positive matrix corresponds to the support of the vectors in a rank-one decomposition of the matrix. We consider a completely positive matrix $X \neq 0$ with a rank-one decomposition set $\mathcal{B}$. For a vector $b \in \mathcal{B}$ we must have that the set $\{i \mid (b)_i > 0\}$ is a clique of $G(X)$. Correspondingly, if a set of vertices $\mathcal{J} \subseteq \{1, \ldots, n\}$ is not a clique of $G(X)$, then there cannot be a vector $b \in \mathcal{B}$ such that $\{i \mid (b)_i > 0\} = \mathcal{J}$. Therefore we need only consider each component submatrix of a matrix separately, and it should be noted that using, for example, a breadth-first search we can split a graph into its component submatrices in linear-time.

From now on, without loss of generality, we shall assume that the matrices we wish to analyze are nonnegative and connected and have all on-diagonal elements strictly positive.

We finish this section by looking at a special property which always holds for at least one minimal rank-one decomposition of a completely positive matrix.

THEOREM 2.1. *For any completely positive matrix A there exists a minimal rank-one decomposition of it such that no two vectors in the decomposition have the same support.*

Proof. Consider two vectors $a, b \in \mathbb{R}^n_{++}$. We define the following:

$$\begin{aligned} \mu &= \max\{\lambda \in \mathbb{R} \mid b - \lambda a \geq 0\}, \\ c &= \frac{1}{\sqrt{1+\mu^2}}(b - \mu a), \\ d &= \frac{1}{\sqrt{1+\mu^2}}(a + \mu b). \end{aligned}$$

Then we have that

$$\begin{aligned} &c \in \mathbb{R}^n_{+} \setminus \mathbb{R}^n_{++}, \\ &d \in \mathbb{R}^n_{++}, \\ &aa^{\mathsf{T}} + bb^{\mathsf{T}} = cc^{\mathsf{T}} + dd^{\mathsf{T}}. \end{aligned}$$

This can easily be extended to any two matrices with the same support. We can now take any minimal rank-one decomposition of a completely positive matrix and use this method to get the desired property. ∎

3. Indices of degree zero or one. In this section we look at how we can reduce the problem of checking whether a matrix is completely positive by considering indices of the matrix with degree zero or one. Recall that we have defined the degree of an index to be the degree of the corresponding vertex in the graph of the matrix.

Degree-zero indices are themselves component submatrices and so can be considered separately. As they are size 1×1 matrices, checking them for complete positivity and, if this is found, providing a minimal rank-one decomposition set are a trivial task.

In order to see how to deal with indices of a higher degree we first consider the following theorem.

THEOREM 3.1. *We define the matrices $X, Y_\theta, Z_\theta \in \mathcal{S}^n$ as*

$$X = \begin{pmatrix} A_1 & a_1 & 0 \\ a_1^{\mathsf{T}} & \alpha & a_2^{\mathsf{T}} \\ 0 & a_2 & A_2 \end{pmatrix}, \qquad Y_\theta = \begin{pmatrix} A_1 & a_1 & 0 \\ a_1^{\mathsf{T}} & \theta & 0 \\ 0 & 0 & 0 \end{pmatrix}, \qquad Z_\theta = \begin{pmatrix} 0 & 0 & 0 \\ 0 & \alpha - \theta & a_2^{\mathsf{T}} \\ 0 & a_2 & A_2 \end{pmatrix},$$

Copyright © by SIAM. Unauthorized reproduction of this article is prohibited.

Downloaded 10/23/12 to 129.125.63.113. Redistribution subject to SIAM license or copyright; see http://www.siam.org/journals/ojsa.php

where $\alpha, \theta \in \mathbb{R}$, $A_1 \in \mathcal{S}^p$, $A_2 \in \mathcal{S}^q$, $a_1 \in \mathbb{R}^p$, $a_2 \in \mathbb{R}^q$, $p, q, n \in \mathbb{Z}$, $p, q > 0$, *and* $n = p + q + 1$. *Then the following three statements are equivalent:*

(a) X *is completely positive.*

(b) *There exists* θ *such that* Y_θ *and* Z_θ *are completely positive.*

(c) Z_φ *is completely positive, where* $\varphi := \min\{\theta \mid Y_\theta \in \mathcal{C}^*\}$.

Proof. We first note that the value of the minimization in (c) is either infinity, and so Z_φ cannot be completely positive, or it is attained, and so both Y_φ and Z_φ are completely positive. It can now be immediately seen that (c) $\Rightarrow$ (b) $\Rightarrow$ (a). From considering the cliques of $G(X)$ we see that (a) $\Rightarrow$ (b). It is also a simple task to show that (b) $\Rightarrow$ (c), by noting that if $Z_\theta \in \mathcal{C}^*$ for some θ, then $Z_\phi \in \mathcal{C}^*$ for all $\phi \leq \theta$, which completes the proof. ∎

From this theorem we see that in some special cases, when it is relatively easy to find $\varphi = \min\{\theta \mid Y_\theta \in \mathcal{C}^*\}$, we can reduce the problem of checking whether X is completely positive to checking whether the smaller nonzero principal submatrix of Z_φ is completely positive. An example of when it is relatively easy to find φ is when the underlying graph given by Y_θ is a completely positive graph. A completely positive graph is defined to be a graph such that for all Y with this underlying graph we have that $Y \in \mathcal{C}^*$ if and only if $Y \in \mathcal{N} \cap \mathcal{S}_+$. A characterization of these graphs is that they have no odd cycles of length greater than or equal to five [19]. This means that in such a case we have $\varphi = \min\{\theta \mid Y_\theta \in \mathcal{N} \cap \mathcal{S}_+\}$, and this optimization problem can be solved in polynomial time up to any required accuracy [25].

In the following theorem we now look at a very simple but very useful special case, which was first considered by Berman and Hershkowitz [3]. They took a different approach to this problem and proved part (a) and a similar result to part (d), except that they gave an inequality relation, whereas we will give an equality.

THEOREM 3.2. *Let* $X \in \mathcal{N}^n$, $Y \in \mathcal{S}^n$ *be given as*

$$X = \begin{pmatrix} \alpha & \beta & 0 \\ \beta & \gamma & a^\mathsf{T} \\ 0 & a & A \end{pmatrix}, \qquad Y = \begin{pmatrix} 0 & 0 & 0 \\ 0 & \gamma - \frac{1}{\alpha}\beta^2 & a^\mathsf{T} \\ 0 & a & A \end{pmatrix},$$

where $\alpha, \beta, \gamma \in \mathbb{R}_+$, $\alpha \neq 0$, $a \in \mathbb{R}_+^{n-2}$, *and* $A \in \mathcal{N}^{n-2}$. *Then we have the following:*

(a) $X \in \mathcal{C}^* \;\Leftrightarrow\; Y \in \mathcal{C}^*$.

(b) *For* $X \in \mathcal{C}^*$, *if we let* $\mathcal{B}_Y \subset \mathbb{R}_+^n$ *be a rank-one decomposition set of* Y, *then the following set is a rank-one decomposition set of* X:

$$\mathcal{B}_X = \mathcal{B}_Y \cup \left\{ \begin{pmatrix} \sqrt{\alpha} \\ \beta/\sqrt{\alpha} \\ 0 \end{pmatrix} \right\}.$$

(c) *If in* (b) $\mathcal{B}_Y$ *is a minimal rank-one decomposition set of* Y, *then* $\mathcal{B}_X$ *is a minimal rank-one decomposition set of* X.

(d) *We have that*

$$\text{cp-rank}(X) = \begin{cases} \text{cp-rank}(Y) + 1 & \textit{if } Y \in \mathcal{C}^*, \\ \infty & \textit{otherwise.} \end{cases}$$

Proof. From [22] we have that $\mathcal{C}^{*2} = \mathcal{N}^2 \cap \mathcal{S}_+^2$. Therefore

$$\begin{pmatrix} \alpha & \beta \\ \beta & \theta \end{pmatrix} \in \mathcal{C}^* \Leftrightarrow \theta \geq \beta^2/\alpha,$$

Copyright © by SIAM. Unauthorized reproduction of this article is prohibited.

Downloaded 10/23/12 to 129.125.63.113. Redistribution subject to SIAM license or copyright; see http://www.siam.org/journals/ojsa.php

and using this, Theorem 3.1 gives us a proof for (a). Part (b) is trivial to prove, and part (d) comes directly from part (c). We will now prove part (c). Another way of expressing part (c) is that there exists a minimal rank-one decomposition of X given by $\mathcal{B}_X$ such that

$$\{v \in \mathcal{B}_X \mid (v)_1 > 0\} = \left\{ \begin{pmatrix} \sqrt{\alpha} \\ \beta/\sqrt{\alpha} \\ 0 \end{pmatrix} \right\}. \tag{3.1}$$

Due to Theorem 2.1 there exists a minimal rank-one decomposition of X given by $\widehat{\mathcal{B}}_X$ such that no two vectors in the decomposition have the same support. If property (3.1) holds, then we are done. If not, then by considering the cliques of $G(X)$, we see that there must exist $\varphi \in \mathbb{R}$ such that $0 < \varphi < \sqrt{\alpha}$ and

$$\left\{ v \in \widehat{\mathcal{B}}_X \mid (v)_1 > 0 \right\} = \left\{ \begin{pmatrix} \varphi \\ \beta/\varphi \\ 0 \end{pmatrix}, \begin{pmatrix} \sqrt{\alpha - \varphi^2} \\ 0 \\ 0 \end{pmatrix} \right\}.$$

It is trivial to see that

$$\begin{aligned} &\begin{pmatrix} \varphi \\ \beta/\varphi \\ 0 \end{pmatrix} \begin{pmatrix} \varphi \\ \beta/\varphi \\ 0 \end{pmatrix}^{\mathsf{T}} + \begin{pmatrix} \sqrt{\alpha - \varphi^2} \\ 0 \\ 0 \end{pmatrix} \begin{pmatrix} \sqrt{\alpha - \varphi^2} \\ 0 \\ 0 \end{pmatrix}^{\mathsf{T}} \\ &\quad = \begin{pmatrix} \sqrt{\alpha} \\ \beta/\sqrt{\alpha} \\ 0 \end{pmatrix} \begin{pmatrix} \sqrt{\alpha} \\ \beta/\sqrt{\alpha} \\ 0 \end{pmatrix}^{\mathsf{T}} + \begin{pmatrix} 0 \\ \beta\sqrt{(\alpha - \varphi^2)/(\alpha\varphi^2)} \\ 0 \end{pmatrix} \begin{pmatrix} 0 \\ \beta\sqrt{(\alpha - \varphi^2)/(\alpha\varphi^2)} \\ 0 \end{pmatrix}^{\mathsf{T}}. \end{aligned}$$

We can use this fact to obtain an alternative minimal rank-one decomposition of X such that property (3.1) does hold. ∎

From this theorem and the result on degree-zero indices we can now construct Algorithm 1 for reducing the problem of checking whether a matrix is completely positive and, if so, finding a (minimal) rank-one decomposition set. Although Berman and Hershkowitz [3] considered a similar method of going through a matrix, this was only to prove that an acyclic matrix is completely positive if and only if it is both nonnegative and positive semidefinite. They did not consider how it could also be used to produce a rank-one decomposition set or its computation time. We now give the following results for this algorithm.

THEOREM 3.3. *Algorithm* 1 *gives the required output, and, if it does not produce the message "$X \notin \mathcal{C}^*$," then the X' and $\mathcal{B}$ produced will have the following properties:*

(a) $X = X' + \sum_{b \in \mathcal{B}} bb^{\mathsf{T}}$.
(b) $\text{cp-rank}(X) = \text{cp-rank}(X') + |\mathcal{B}|$.
(c) *If an index i of X' has degree zero, then $(X')_{ii} = 0$.*
(d) *X' has no indices of degree one.*

Also, provided that our inputs and outputs of the matrices and vectors were "efficient" in Algorithm 1*, then this algorithm runs in linear-time.*

Proof. From going through the algorithm and using Theorem 3.2 this is trivial to prove. It should be noted that we required the matrices and vectors to be inputted/outputted efficiently. First, this means dealing with the square roots. Second, this is because the inputting/outputting of a full vector or matrix would involve, respectively, n and n^2 entries which would limit the algorithm to working in quadratic time,

Copyright © by SIAM. Unauthorized reproduction of this article is prohibited.

Downloaded 10/23/12 to 129.125.63.113. Redistribution subject to SIAM license or copyright; see http://www.siam.org/journals/ojsa.php

ALGORITHM 1. Reducing the problem of checking for complete positivity.

Input: A matrix $X \in \mathcal{N}^n$ such that $(X)_{ii} > 0$ for all $i = 1, \ldots, n$.

Output: Either "$X \notin \mathcal{C}^*$" or a matrix $X' \in \mathcal{S}^n$ and a finite set $\mathcal{B} \subset \mathbb{R}^n_+$ (see Theorem 3.3).

initiate a set $\mathcal{B} = \emptyset$.
initiate a set $\mathcal{R} = \{1, \ldots, n\}$ to keep track of indices remaining.
analyze X producing
  (a) a set $\mathcal{J} \subseteq \mathcal{R}$ of indices with degree zero or one,
  (b) a vector $d \in \mathbb{Z}^n$ such that $(d)_i$ is defined to be the degree of index i,
  (c) a set $\{\mathfrak{N}_1, \ldots, \mathfrak{N}_n\}$ such that $\mathfrak{N}_i := \{j \mid j \text{ is a neighbor of } i \text{ in } G(X)\}$.
while $\mathcal{J} \neq \emptyset$ **do**
  pick an $i \in \mathcal{J}$ to analyze.
  update $\mathcal{R} \leftarrow \mathcal{R} \setminus \{i\}$, $\quad \mathcal{J} \leftarrow \mathcal{J} \setminus \{i\}$
  if $(d)_i = 0$ **then**
    update $\mathcal{B} \leftarrow \mathcal{B} \cup \{x\}$ such that $x \in \mathbb{R}^n_+$ and

$$(x)_k := \begin{cases} \sqrt{(X)_{ii}} & \text{if } k = i, \\ 0 & \text{otherwise.} \end{cases}$$

    update $(X)_{ii} \leftarrow 0$
  else
    find $j \in \mathfrak{N}_i \cap \mathcal{R}$
    update $(X)_{jj} \leftarrow (X)_{jj} - (X)^2_{ij}/(X)_{ii}$
    if $(X)_{jj} < 0$ **then**
      output "$X \notin \mathcal{C}^*$"
      exit
    else if $(X)_{jj} = 0$ and $d_j \geq 2$ **then**
      output "$X \notin \mathcal{C}^*$"
      exit
    end if
    output $\mathcal{B} \leftarrow \mathcal{B} \cup \{x\}$ such that $x \in \mathbb{R}^n_+$ and

$$(x)_k := \begin{cases} \sqrt{(X)_{ii}} & \text{if } k = i, \\ (X)_{ij}/\sqrt{(X)_{ii}} & \text{if } k = j, \\ 0 & \text{otherwise.} \end{cases}$$

    update $(X)_{ij} \leftarrow 0$
    update $(X)_{ii} \leftarrow 0$
    update $(d)_j \leftarrow (d)_j - 1$
    if $(d)_j = 1$ **then**
      update $\mathcal{J} \leftarrow \mathcal{J} \cup \{j\}$
    else if $(d)_j = 0$ and $(X)_{jj} = 0$ **then**
      update $\mathcal{J} \leftarrow \mathcal{J} \setminus \{j\}$
      update $\mathcal{R} \leftarrow \mathcal{R} \setminus \{j\}$
    end if
  end if
end while
output $X' \leftarrow X$
output $\mathcal{B}$

Copyright © by SIAM. Unauthorized reproduction of this article is prohibited.

Downloaded 10/23/12 to 129.125.63.113. Redistribution subject to SIAM license or copyright; see http://www.siam.org/journals/ojsa.php

but a more efficient way of specifying a sparse vector or matrix is to give only its nonzero entries, and we are required to do this in order for the algorithm to work in linear-time. ∎

For Algorithm 1 we can see that if X'' is the maximal principal submatrix of X' such that no row/column is equal to zero, then $G(X'')$ is the maximal induced subgraph of $G(X)$ such that $G(X'')$ has no vertices of degree one or zero. This means that if X was acyclic, for example, tridiagonal, then in linear-time either the algorithm would output $X \notin \mathcal{C}^*$ or we would have $X' = 0$ and therefore a certificate of complete positivity in the form of a minimal rank-one decomposition set $\mathcal{B}$. We can also see that such a minimal rank-one decomposition set would be of cardinality $n-1$ or n. It was found in [5, Theorem 3.7] that this number is actually equal to the rank of the matrix.

It should also be noted that the choice of the next vertex to consider in step 5 of Algorithm 1 affects the way in which the algorithm goes through the vertices and can lead to a different set $\mathcal{B}$ at the end of the algorithm. If we simply go through the vertices in $\mathcal{J}$ in numerical order, then given a permutation matrix P the algorithm will not necessarily return the same solution (up to permutation) when X and $P^{\mathsf{T}}XP$ are inputted.

4. Chains. We define a chain of a graph to be a simple path in it such that all vertices within the path (excluding the two end vertices) have degree equal to two. This is equivalent to this part of the matrix being tridiagonal. The form of chain that we shall consider is shown in Figure 4.1, where we let the internal vertices and the edges of the chain have fixed weightings (given by the α's and β's, respectively), we let y be a variable, and we let $z(y)$ be the minimum allowable value such that the chain is completely positive. We then consider how Algorithm 1 would run through the chain starting at y, moving through the vertices in turn and being used to give us the value of $z(y)$ for each y. We show that rather than having to recompute the algorithm for different values of y, we can instead find a simple formula linking y and $z(y)$. We consider this, not only to help improve our understanding of Algorithm 1, but also due to the useful applications that this method will provide in sections 5 and 6.

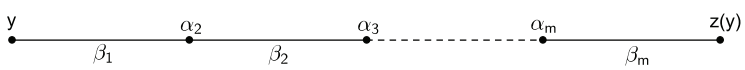

FIG. 4.1. *A chain we consider the algorithm working through with $y > 0$ and $\alpha_i, \beta_i > 0$ for all i.*

For $i \geq 2$ the algorithm would take each vertex from α_i to $f_i(y)$ and then to zero, where

$$f_i(y) := \alpha_i - \frac{\beta_{i-1}^2}{f_{i-1}(y)} \qquad \text{for } i = 2, \ldots, m,$$
$$f_1(y) := y.$$

We have the requirement that $f_i(y) > 0$ for all i, and we also have that $z(y) = \beta_m^2 / f_m(y)$.

We now present the following lemmas, which will be used to remove the recursion from y, thus meaning that we do not need to consider the recursion separately for each different value of y.

Copyright © by SIAM. Unauthorized reproduction of this article is prohibited.

Downloaded 10/23/12 to 129.125.63.113. Redistribution subject to SIAM license or copyright; see http://www.siam.org/journals/ojsa.php

LEMMA 4.1. *We have that*

$$f_i(y) = \frac{\lambda_i y - \mu_i}{\lambda_{i-1} y - \mu_{i-1}} \qquad \text{for } i = 1, \ldots, m,$$

where $\lambda_0 = 0$, $\mu_0 = -1$, $\lambda_1 = 1$, $\mu_1 = 0$,

$$\lambda_i = \alpha_i \lambda_{i-1} - \beta_{i-1}^2 \lambda_{i-2} \qquad \text{for } i = 2, \ldots, m,$$
$$\mu_i = \alpha_i \mu_{i-1} - \beta_{i-1}^2 \mu_{i-2} \qquad \text{for } i = 2, \ldots, m.$$

Also we have that the requirement "$f_i(y) > 0$ *for all* i*" is equivalent to*

$$\lambda_i y > \mu_i \qquad \text{for all } i. \tag{4.1}$$

Proof. This is trivial to prove by induction. ∎

LEMMA 4.2. *We shall always get that* $\mu_i \lambda_{i+1} < \mu_{i+1} \lambda_i$ *for all* $i = 0, \ldots, m-1$.

Proof. We have that $\mu_0 \lambda_1 = -1 < 0 = \mu_1 \lambda_0$, and for $i \geq 1$,

$$\begin{aligned} \mu_{i+1}\lambda_i - \mu_i \lambda_{i+1} &= (\alpha_{i+1}\mu_i - \beta_i^2 \mu_{i-1})\lambda_i - \mu_i(\alpha_{i+1}\lambda_i - \beta_i^2 \lambda_{i-1}) \\ &= \beta_i^2 (\mu_i \lambda_{i-1} - \mu_{i-1}\lambda_i). \end{aligned}$$

We can now use proof by induction. ∎

LEMMA 4.3. *Requirement* (4.1) *implies that* $\lambda_i, \mu_i > 0$ *for all* $i \geq 2$.

Proof. We shall again use proof by induction. For the case when $i = 2$ we have that

$$\lambda_2 = \alpha_2 > 0, \qquad \mu_2 = \beta_1^2 > 0.$$

Therefore the statement is true for $i = 2$. Now for the sake of induction suppose that it is true for $i = k - 1$. From Lemma 4.2 and requirement (4.1) we get that

$$\frac{\mu_{k-1}}{\lambda_{k-1}} < y \qquad \text{and} \qquad \frac{\mu_{k-1}\lambda_k}{\lambda_{k-1}} < \mu_k < \lambda_k y.$$

From this we see that we must have that $\lambda_k, \mu_k > 0$. ∎

LEMMA 4.4. *If* $\lambda_i > 0$ *for all* $i \geq 1$, *then*

$$\max\left\{ \frac{\mu_i}{\lambda_i} \mid i = 1, \ldots, m \right\} = \frac{\mu_m}{\lambda_m}.$$

Proof. This is simple to prove using Lemma 4.2. ∎

THEOREM 4.5. *Requirement* (4.1) *is equivalent to*

$$y > \frac{\mu_m}{\lambda_m} \qquad \text{and} \qquad \lambda_i > 0 \quad \text{for all } i \geq 2.$$

Proof. The proof comes trivially from Lemmas 4.3 and 4.4. ∎

METHOD 4.6. *Therefore the problem of going through the chain can be split into the following three parts:*

(a) *Compute the following:*

$$\begin{aligned} &\lambda_0 = 0,\ \mu_0 = -1,\ \lambda_1 = 1,\ \mu_1 = 0, \\ &\lambda_i = \alpha_i \lambda_{i-1} - \beta_{i-1}^2 \lambda_{i-2} \qquad \text{for } i = 2, \ldots, m, \\ &\mu_i = \alpha_i \mu_{i-1} - \beta_{i-1}^2 \mu_{i-2} \qquad \text{for } i = 2, \ldots, m. \end{aligned}$$

Copyright © by SIAM. Unauthorized reproduction of this article is prohibited.

Downloaded 10/23/12 to 129.125.63.113. Redistribution subject to SIAM license or copyright; see http://www.siam.org/journals/ojsa.php

(b) *Check that $\lambda_i > 0$ for all $i \geq 2$; otherwise it cannot be part of a completely positive matrix.*
(c) *Require that $y > \mu_m/\lambda_m$ and*

$$z(y) = \frac{\beta_m^2(\lambda_{m-1}y - \mu_{m-1})}{\lambda_m y - \mu_m}.$$

In the following two sections we look at two alternative ways in which this result can be used.

5. Matrices with circular graphs. If Algorithm 1 did not determine whether the original matrix was completely positive or not, then the degree of the indices in the remaining matrix must be strictly greater than one, and so the simplest form that it can take is being a circular matrix, where we recall that a circular matrix is one with an underlying circular graph and a circular graph is a graph consisting of a single cycle. This is also sometimes referred to as a cycle graph.

The complete positivity of circular matrices has previously been studied in [29, 30]. In [29] Xu and Li found a necessary and sufficient condition for a circular matrix (of order greater than 3) to be completely positive; however, it is unclear how this result can actually be used to check whether a circular graph is completely positive. In [30] Zhang and Li showed that a circular matrix (of order greater than 3) is completely positive if and only if the determinant of its comparison matrix is nonnegative. The paper also included a method for finding a minimal rank-one decomposition set of a circular matrix (of order greater than 3); however, this was included only for the purpose of providing a proof to a theorem related to the number of minimal rank-one decompositions that a circular matrix has. As a result this method was not subjected to much analysis and is relatively complicated.

In this section we will use the results from section 4 to develop an alternative algorithm for checking whether a circular matrix is completely positive and, if so, providing a minimal rank-one decomposition set of the matrix. It will also be seen that this method runs in linear-time.

We will begin by considering the following two theorems.

THEOREM 5.1 (see [5, Remark 3.3]). *If A is a triangle-free completely positive matrix which is not acyclic, then* $\text{cp-rank}(A) = |E|$, *where E is the set of edges in the graph $G(A)$.*

THEOREM 5.2 (see [5, Theorem 3.2]). *Let A be an $n \times n$ completely positive matrix. If $n \leq 3$, then*

$$\text{cp-rank}(A) = \text{rank}(A).$$

As we see from these theorems we should consider the cases of $n = 3$ and $n > 3$ separately. We will first extend our method from section 4 for the case when $n > 3$. We let $A \in \mathcal{N}^n$ be a circular matrix and without loss of generality suppose that

$$A = \begin{pmatrix} \alpha_1 & \beta_1 & 0 & \cdots & 0 & \beta_n \\ \beta_1 & \alpha_2 & \beta_2 & \cdots & 0 & 0 \\ 0 & \beta_2 & \alpha_3 & \cdots & 0 & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & 0 & \cdots & \alpha_{n-1} & \beta_{n-1} \\ \beta_n & 0 & 0 & \cdots & \beta_{n-1} & \alpha_n \end{pmatrix}. \tag{5.1}$$

Copyright © by SIAM. Unauthorized reproduction of this article is prohibited.

Downloaded 10/23/12 to 129.125.63.113. Redistribution subject to SIAM license or copyright; see http://www.siam.org/journals/ojsa.php

If A is completely positive, then due to its cp-rank being equal to n (Theorem 5.1) and by considering its cliques, we see that its minimal decompositions must be of the form

$$\mathcal{B} = \left\{ \begin{pmatrix} \upsilon_1 \\ \omega_1 \\ 0 \\ \vdots \\ 0 \\ 0 \end{pmatrix}, \begin{pmatrix} 0 \\ \upsilon_2 \\ \omega_2 \\ \vdots \\ 0 \\ 0 \end{pmatrix}, \ldots, \begin{pmatrix} 0 \\ 0 \\ 0 \\ \vdots \\ \upsilon_{n-1} \\ \omega_{n-1} \end{pmatrix}, \begin{pmatrix} \omega_n \\ 0 \\ 0 \\ \vdots \\ 0 \\ \upsilon_n \end{pmatrix} \right\}.$$

From this it can be seen that A is completely positive if and only if there exists a $y = \upsilon_1^2$ such that the chain in Figure 4.1 is completely positive, with $z(y) = \alpha_1 - y$, and it can easily be seen how minimal rank-one decompositions of the original matrix A and the chain are related. From section 4 we now get the following linear-time method for analyzing the matrix.

METHOD 5.3. *The problem of determining whether A given in* (5.1) *is completely positive is equivalent to computing*

$$\begin{aligned} &\lambda_0 = 0,\ \mu_0 = -1,\ \lambda_1 = 1,\ \mu_1 = 0, \\ &\lambda_i = \alpha_i \lambda_{i-1} - \beta_{i-1}^2 \lambda_{i-2} \qquad \textit{for } i = 2, \ldots, n, \\ &\mu_i = \alpha_i \mu_{i-1} - \beta_{i-1}^2 \mu_{i-2} \qquad \textit{for } i = 2, \ldots, n, \end{aligned}$$

checking that $\lambda_i > 0$ for all $i \geq 2$, and solving

$$\begin{aligned} &\textit{find} && y \\ &\textit{subject to} && y > \mu_n/\lambda_n, \\ & && 0 = \lambda_n y^2 + (\beta_n^2 \lambda_{n-1} - \alpha_1 \lambda_n - \mu_n) y + (\alpha_1 \mu_n - \beta_n^2 \mu_{n-1}). \end{aligned}$$

We also have that if y is a feasible solution, then the corresponding minimal rank-one decomposition is

$$\mathcal{B} = \left\{ \begin{pmatrix} \upsilon_1 \\ \omega_1 \\ 0 \\ \vdots \\ 0 \\ 0 \end{pmatrix}, \begin{pmatrix} 0 \\ \upsilon_2 \\ \omega_2 \\ \vdots \\ 0 \\ 0 \end{pmatrix}, \ldots, \begin{pmatrix} 0 \\ 0 \\ 0 \\ \vdots \\ \upsilon_{n-1} \\ \omega_{n-1} \end{pmatrix}, \begin{pmatrix} \omega_n \\ 0 \\ 0 \\ \vdots \\ 0 \\ \upsilon_n \end{pmatrix} \right\},$$

$$\begin{aligned} \textit{where } \upsilon_i &= \sqrt{\frac{\lambda_i y - \mu_i}{\lambda_{i-1} y - \mu_{i-1}}}, \\ \omega_i &= \beta_i / \upsilon_i \end{aligned} \qquad \textit{for all } i.$$

We note that this method even checks the complete positivity in linear-time of circular matrices such that their order is odd and greater than or equal to 5, even though for these types of matrices, their underlying graph is not a completely positive graph.

Copyright © by SIAM. Unauthorized reproduction of this article is prohibited.

Downloaded 10/23/12 to 129.125.63.113. Redistribution subject to SIAM license or copyright; see http://www.siam.org/journals/ojsa.php

For completeness we will now also consider how to test whether a strictly positive matrix $X \in \mathcal{S}^3$ is completely positive and, if so, find a minimal rank-one decomposition set for it. In order to do this we need the following lemmas, in which we will consider the matrix

$$X = \begin{pmatrix} \alpha_1 & \beta_1 & \beta_3 \\ \beta_1 & \alpha_2 & \beta_2 \\ \beta_3 & \beta_2 & \alpha_3 \end{pmatrix}, \qquad \text{where } \alpha, \beta \in \mathbb{R}^3_{++} \text{ and } \alpha_3\beta_1^2 \leq \alpha_1\beta_2^2. \tag{5.2}$$

It should be noted that we can always permute a 3×3 strictly positive symmetric matrix so that the required inequalities hold.

LEMMA 5.4. *For X given in* (5.2) *we have that $X \in \mathcal{C}^*$ if and only if $\beta_1^2 \leq \alpha_1\alpha_2$, $\beta_2^2 \leq \alpha_2\alpha_3$, $\beta_3^2 \leq \alpha_3\alpha_1$, and* $\det(X) \geq 0$.

Proof. From [22] we have that $\mathcal{C}^{*3} = \mathcal{N}^3 \cap \mathcal{S}^3_+$. From the conditions in (5.2) we have that $X \in \mathcal{N}^3$. It is known that a matrix is positive semidefinite if and only if all its principal minors are nonnegative [23, p. 40]. This combined with the fact that the diagonal entries of X are nonnegative gives us the required result. ∎

LEMMA 5.5. *For X given in* (5.2) *such that $X \in \mathcal{C}^*$ we have that*

$$\text{cp-rank}(X) = 1 \Leftrightarrow \beta_1^2 = \alpha_1\alpha_2.$$

Proof. We have that cp-rank$(X) = 1$ if and only if there exists a $b \in \mathbb{R}^3_+$ such that $X = bb^\mathsf{T}$. It is obvious that such a b must be given by

$$b = \frac{1}{\sqrt{\alpha_1}} \begin{pmatrix} \alpha_1 \\ \beta_1 \\ \beta_3 \end{pmatrix}.$$

From this we get that the cp-rank$(X) = 1$ if and only if

$$X = \begin{pmatrix} \alpha_1 & \beta_1 & \beta_3 \\ \beta_1 & \beta_1^2/\alpha_1 & \beta_1\beta_3/\alpha_1 \\ \beta_3 & \beta_1\beta_3/\alpha_1 & \beta_3^2/\alpha_1 \end{pmatrix}.$$

The forward implication is seen by comparing the required form of X to the original form of X.

For the reverse implication we note that if $\beta_1^2 = \alpha_1\alpha_2$, then from the requirements for complete positivity and the restrictions on X we have that $\alpha_2\alpha_3 \geq \beta_2^2 \geq \alpha_3\beta_1^2/\alpha_1 = \alpha_2\alpha_3$, implying that $\beta_2^2 = \alpha_2\alpha_3$. We also have that

$$\begin{aligned} 0 \leq \det(X) &= \alpha_1\alpha_2\alpha_3 + 2\beta_1\beta_2\beta_3 - \beta_1^2\alpha_3 - \beta_2^2\alpha_1 - \beta_3^2\alpha_2 \\ &= \alpha_1\alpha_2\alpha_3 + 2(\alpha_2\sqrt{\alpha_1\alpha_3})\beta_3 - \alpha_1\alpha_2\alpha_3 - \alpha_1\alpha_2\alpha_3 - \beta_3^2\alpha_2 \\ &= -\alpha_2(\beta_3 - \sqrt{\alpha_1\alpha_3})^2. \end{aligned}$$

This implies that X must be in the required form. ∎

LEMMA 5.6. *For X given in* (5.2) *such that $X \in \mathcal{C}^*$ we have that $\alpha_1\beta_2 - \beta_1\beta_3 \geq 0$.*

Proof. From Lemma 5.4 and the restrictions on X we have that $\beta_3^2 \leq \alpha_3\alpha_1$ and $0 \leq \alpha_3\beta_1^2 \leq \alpha_1\beta_2^2$. This implies that $\alpha_3\beta_1^2\beta_3^2 \leq \alpha_3\alpha_1^2\beta_2^2$, which gives the required result, due to X being strictly positive. ∎

LEMMA 5.7. *For $X \in \mathcal{C}^*$ as given in* (5.2) *such that* cp-rank$(X) \neq 1$ *we have that*

$$\text{cp-rank}(X) = 2 \Leftrightarrow \det(X) = 0.$$

Copyright © by SIAM. Unauthorized reproduction of this article is prohibited.

Downloaded 10/23/12 to 129.125.63.113. Redistribution subject to SIAM license or copyright; see http://www.siam.org/journals/ojsa.php

Proof. From [5, Theorem 3.2] we have that cp-rank(X) = rank(X). From the restrictions on X we have that $X \neq 0$, which, combined with the requirement that cp-rank$(X) \neq 1$, implies that rank$(X) \geq 2$. From this we have

$$\begin{aligned} \text{cp-rank}(X) = 2 &\Leftrightarrow \text{rank}(X) = 2 \\ &\Leftrightarrow \text{rank}(X) \neq 3 \\ &\Leftrightarrow \det(X) = 0, \end{aligned}$$

which completes the proof. ∎

From these lemmas we now present Algorithm 2 for testing whether a matrix X of the form given in (5.2) is completely positive and, if so, finding a minimal rank-one decomposition set for it.

ALGORITHM 2. For testing whether a matrix $X \in \mathcal{S}^3$ of the form given in (5.2) is completely positive and, if so, finding a minimal rank-one decomposition for it.

Input: A matrix X of the form given in (5.2).

Output: Either "$X \notin \mathcal{C}^*$" or a set $\mathcal{B} \subseteq \mathbb{R}^3_+$ such that $|\mathcal{B}|$ = cp-rank(X) and $X = \sum_{b \in \mathcal{B}} bb^\mathsf{T}$.

if $\beta_1^2 > \alpha_1\alpha_2$ **or** $\beta_2^2 > \alpha_2\alpha_3$ **or** $\beta_3^2 > \alpha_3\alpha_1$ **or** $\det(X) < 0$ **then**
  output "$X \notin \mathcal{C}^*$"
else
  initiate

$$\mathcal{B} = \left\{ \frac{1}{\sqrt{\alpha_1}} \begin{pmatrix} \alpha_1 \\ \beta_1 \\ \beta_3 \end{pmatrix} \right\}$$

  if $\beta_1^2 \neq \alpha_1\alpha_2$ **then**
    update

$$\mathcal{B} \leftarrow \mathcal{B} \cup \left\{ \sqrt{\frac{1}{\alpha_1(\alpha_1\alpha_2 - \beta_1^2)}} \begin{pmatrix} 0 \\ \alpha_1\alpha_2 - \beta_1^2 \\ \alpha_1\beta_2 - \beta_1\beta_3 \end{pmatrix} \right\}$$

    if $\det(X) \neq 0$ **then**
      update $\mathcal{B} \leftarrow \mathcal{B} \cup \left\{ \sqrt{\frac{\det(X)}{\alpha_1\alpha_2 - \beta_1^2}} \begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix} \right\}$.
    end if
  end if
  output $\mathcal{B}$
end if

This is a linear-time method for checking whether a circular matrix $A \in \mathcal{N}^3$ is completely positive and, if so, giving a minimal rank-one decomposition set for it.

6. Reducing chain lengths. Suppose that the matrix we wish to check for being completely positive gives the weighted-graph in Figure 6.1. In this section we will see how we can reduce the length of the chain to give a smaller matrix while maintaining the property of whether the matrix is completely positive or not. For simplicity we shall view the matrices using their weighted-graph forms.

Copyright © by SIAM. Unauthorized reproduction of this article is prohibited.

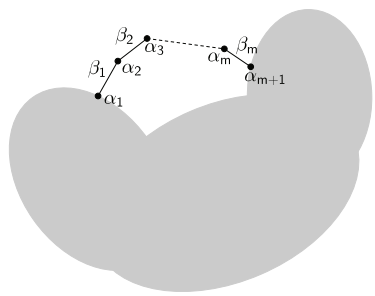

FIG. 6.1. *We wish to analyze the matrix giving this weighted-graph to check for complete positivity, where the end points of the chain are distinct, the gray area represents an arbitrary structure in the graph, and* $m > 3$.

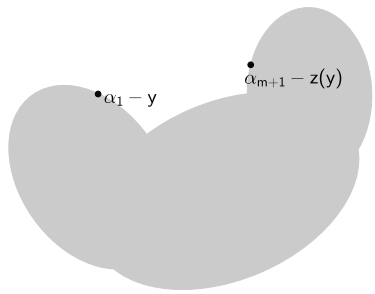

FIG. 6.2. *The weighted-graph in Figure* 6.1 *gives a completely positive matrix if and only if there exists a* y *such that the chain in Figure* 4.1 *and the weighted-graph above give completely positive matrices, where the gray area represents an arbitrary structure in the graph.*

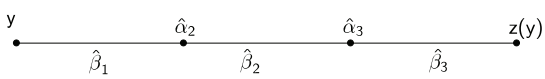

FIG. 6.3. *A chain we consider the algorithm working through with* $n > 3$, $y > 0$, *and the values for* $\widehat{\alpha}_2$, $\widehat{\alpha}_3$, $\widehat{\beta}_1$, $\widehat{\beta}_2$, *and* $\widehat{\beta}_3$ *given in* (6.1).

From considering the form of the rank-one decompositions when this graph is completely positive we see that we have that the graph gives a completely positive matrix if and only if there exists a y such that the chain in Figure 4.1 and the weighted-graph in Figure 6.2 give completely positive matrices.

We now consider the chain in Figure 4.1 in which the values of y and $z(y)$ are not fixed. We consider Method 4.6 on this chain. If the second step (checking $\lambda_i > 0$) finds that the chain cannot be part of a graph giving a completely positive matrix, then we are done. Otherwise we compare this chain to the chain in Figure 6.3.

In the chain in Figure 6.3 we set the values $\widehat{\alpha}_2$, $\widehat{\alpha}_3$, $\widehat{\beta}_1$, $\widehat{\beta}_2$, $\widehat{\beta}_3$, from the results of

Copyright © by SIAM. Unauthorized reproduction of this article is prohibited.

Downloaded 10/23/12 to 129.125.63.113. Redistribution subject to SIAM license or copyright; see http://www.siam.org/journals/ojsa.php

Downloaded 10/23/12 to 129.125.63.113. Redistribution subject to SIAM license or copyright; see http://www.siam.org/journals/ojsa.php

running the method for the original chain and where we pick arbitrary $\gamma_1, \gamma_2 > 0$:

$$\begin{aligned}
\widehat{\alpha}_2 &= \gamma_1 \lambda_{m-1}, \\
\widehat{\alpha}_3 &= \gamma_2 \mu_m, \\
\widehat{\beta}_1 &= \sqrt{\gamma_1 \mu_{m-1}}, \\
\widehat{\beta}_2 &= \sqrt{\gamma_1 \gamma_2 (\lambda_{m-1}\mu_m - \lambda_m \mu_{m-1})}, \\
\widehat{\beta}_3 &= \beta_m \sqrt{\gamma_2 \mu_{m-1}}.
\end{aligned} \tag{6.1}$$

We are free to pick whatever strictly positive values of γ_1 and γ_2 that we like without changing the theory. This freedom may, however, be able to be put to some advantages in reducing numerical difficulties in an algorithm, and it is recommended to pick values such that the order of magnitude on these vertices and edges is approximately that of the original weighted-graph.

From the results in section 4 we can immediately see that all the vertices and edges in the chain have strictly positive values. We now consider Method 4.6 running through this chain.

(a) We compute the values for $\widehat{\lambda}_i$, $\widehat{\mu}_i$, displayed in the following table:

i	$\widehat{\lambda}_i$	$\widehat{\mu}_i$
0	0	-1
1	1	0
2	$\gamma_1 \lambda_{m-1}$	$\gamma_1 \mu_{m-1}$
3	$\gamma_1 \gamma_2 \lambda_m \mu_{m-1}$	$\gamma_1 \gamma_2 \mu_m \mu_{m-1}$

(b) We can easily see that the values of $\widehat{\lambda}_i$ for $i \geq 2$ are strictly positive.

(c) We now have the following requirement on y and corresponding value for $z(y)$:

$$\begin{aligned}
y &> \frac{\widehat{\mu}_3}{\widehat{\lambda}_3} = \frac{\mu_m}{\lambda_m}, \\
z(y) &= \frac{\widehat{\beta}_3^2 (\widehat{\lambda}_2 y - \widehat{\mu}_2)}{\widehat{\lambda}_3 y - \widehat{\mu}_3} = \frac{\beta_m^2 (\lambda_{m-1} y - \mu_{m-1})}{\lambda_m y - \mu_m}.
\end{aligned}$$

Therefore, viewed from the end points, the chains in Figures 4.1 and 6.3 are equivalent. Therefore, the graph in Figure 6.1 is completely positive if and only if the following two conditions hold:

(a) When computing the first two steps of Method 4.6 on the chain we do not find that the chain cannot be part of a completely positive graph.

(b) The graph in Figure 6.4 is completely positive, using the values given in (6.1).

We let X be the original matrix and Y be the matrix produced from our method. If we had a (minimal) rank-one decomposition set of Y, then it would be a trivial task to convert this into a (minimal) rank-one decomposition set of X. We note that we have that

$$\text{cp-rank}(X) = \text{cp-rank}(Y) + m - 3.$$

Copyright © by SIAM. Unauthorized reproduction of this article is prohibited.

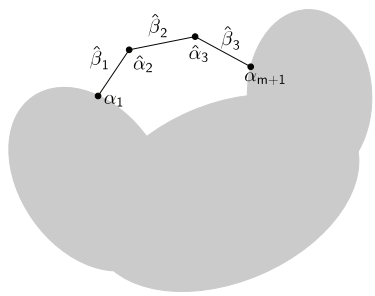

FIG. 6.4. *Smaller graph for checking for complete positivity, which is equivalent to that in Figure* 6.1 *using the results in section* 6.

We finish this section by discussing the computational time of such a process. One simple method for applying this process is as follows, where we assume that no component submatrix is circular:

(a) Find $\mathcal{W} = \{v \in \{1, \dots, n\} \mid \text{The degree of } v \text{ in } G(X) \text{ is equal to } 2\}$.
(b) Find connected components of the subgraph of $G(X)$ induced by the vertices $\mathcal{W}$. Let these be denoted by the following with consecutive vertices being connected: $\{\{v_1^1, \dots, v_{k_1}^1\}, \dots, \{v_1^l, \dots, v_{k_l}^l\}\}$.
(c) For all $i \in \{1, \dots, l\}$ such that $k_i \geq 3$, do the following:
  (i) Find $u, w \in \{1, \dots, n\} \setminus \mathcal{W}$ such that $u, v_1^i, \dots, v_{k_i}^i, w$ is a chain in $G(X)$.
  (ii) If $u \neq w$, then apply the method for reducing chain lengths to this chain.
  (iii) If $u = w$ and $k_i \geq 4$, then apply the method for reducing chain lengths to the chain $\{u, v_1^i, \dots, v_{k_i}^i\}$.

This method takes a linear number of calculations. In general we cannot compute the square roots exactly, but if we are computing to a set level of accuracy, then this method could be carried out in linear-time.

7. Preprocessing. For a matrix $X \in \mathcal{S}^n$ we can now reduce the problem of checking whether it is completely positive and finding a (minimal) rank-one decomposition using the following linear-time method:

(a) Check whether the matrix is nonnegative.
(b) Check that whenever one of the matrix's on-diagonal entries is equal to zero, all of the off-diagonal entries in this row and column are also equal to zero.
(c) Reduce the problem to considering the maximal principal submatrix with strictly positive on-diagonal elements.
(d) Use Algorithm 1 to reduce the problem.
(e) Split a matrix into its component submatrices (for example, with a breadth-first search).
(f) Use section 2 to connect results from these submatrices to those for the original matrix.
(g) For each of these submatrices, do the following:
  (i) If the resultant matrix is in $\mathcal{S}^3$, then use Algorithm 2 to process it.
  (ii) Otherwise, if the resultant matrix is circular, then use Method 5.3 to process it.
  (iii) Otherwise use the method from section 6 to reduce the chain lengths.

This method fully processes all component submatrices which have a maximum of one cycle. If all the component submatrices have a maximum of one cycle, then

Copyright © by SIAM. Unauthorized reproduction of this article is prohibited.

Downloaded 10/23/12 to 129.125.63.113. Redistribution subject to SIAM license or copyright; see http://www.siam.org/journals/ojsa.php

Downloaded 10/23/12 to 129.125.63.113. Redistribution subject to SIAM license or copyright; see http://www.siam.org/journals/ojsa.php

this method determines whether the matrix is completely positive in linear-time and, if so, also outputs a minimal rank-one decomposition of it. Otherwise the method reduces the problem. As the method runs in linear-time and all known algorithms for computing the cp-rank in the general case run in exponential time [4], this is a very efficient preprocessor.

8. Number of minimal decompositions. Our method finds a single minimal rank-one decomposition set for a completely positive matrix such that every component submatrix has a maximum of one cycle. In this section we briefly look at how many minimal rank-one decomposition sets these matrices actually have. For simplicity we assume that the matrices are completely positive and connected and all the on-diagonal elements are strictly positive. We could then use section 3 to extend these results to matrices where the assumptions do not hold.

If the cp-rank of a matrix $X \in \mathcal{S}^n$ is equal to one, then it is trivial to see that it must have exactly one minimal rank-one decomposition set. Next we consider when the cp-rank of X is equal to two.

THEOREM 8.1. *Let $X \in \mathcal{C}^{*n}$ be a connected matrix such that all the on-diagonal elements are strictly positive and* $\text{cp-rank}(X) = 2$. *Then the following hold:*

(a) *If there exists $i, j \in \{1, \ldots, n\}$ such that $(X)_{ij} = 0$, then there is exactly one minimal rank-one decomposition set.*

(b) *If there does not exist $i, j \in \{1, \ldots, n\}$ such that $(X)_{ij} = 0$, then there are infinitely many minimal rank-one decomposition sets.*

Proof. This proof comes from considering the proof in [5, Theorem 2.1].

A minimal rank-one decomposition set must be of the form

$$\left\{ \begin{pmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{pmatrix}, \begin{pmatrix} y_1 \\ y_2 \\ \vdots \\ y_n \end{pmatrix} \right\}.$$

We now consider the ordered set of vectors

$$\mathcal{V} = \{v_1, \ldots, v_n\} \subset \mathbb{R}^2_+ \qquad \text{such that } v_i = \begin{pmatrix} x_i \\ y_i \end{pmatrix} \text{ for all } i.$$

X is then the gram matrix of these vectors; i.e., $(X)_{ij} = \langle v_i, v_j \rangle$ for all i, j. The minimal rank-one decomposition set is unique if and only if the ordered set $\mathcal{V}$ is unique up to a swapping of the coordinates of the v_i's. For $i, j = 1, \ldots, n$ such that $i < j$, let θ_{ij} be the angle between the vectors $v_i, v_j \in \mathcal{V}$, such that $0 \leq \theta_{ij} \leq \pi$.

$$\frac{(X)_{ij}}{\sqrt{(X)_{ii}(X)_{jj}}} = \frac{\langle v_i, v_j \rangle}{\|v_i\| \|v_j\|} = \cos \theta_{ij}.$$

We have that

$$X_{ij} > 0 \Leftrightarrow \theta_{ij} < \pi/2 \qquad \text{and} \qquad X_{ij} = 0 \Leftrightarrow \theta_{ij} = \pi/2.$$

Now let v_k, v_l be the pair of vectors from $\mathcal{V}$ with maximal angle θ_{kl}. As $\mathcal{V} \subset \mathbb{R}^2_+$, we see that once the vectors v_k, v_l are set, all the other vectors are uniquely defined and must lie between them.

Copyright © by SIAM. Unauthorized reproduction of this article is prohibited.

Downloaded 10/23/12 to 129.125.63.113. Redistribution subject to SIAM license or copyright; see http://www.siam.org/journals/ojsa.php

TABLE 8.1

Properties of minimal rank-one decompositions for some matrices which are completely positive and connected and have all on-diagonal entries strictly positive. Using the results of section 3 we can extend these results to all the matrices which can be checked and decomposed by our method.

Type	cp-rank	Number of minimal rank-one decomposition sets	Sketch of proof on number of minimal rank-one decomposition sets
Acyclic	$n-1$	1	From the form that a minimal rank-one decomposition set must take.
	n	∞	Consider stopping Algorithm 1 when exactly two (consecutive) indices are remaining. This effectively leaves us with a strictly positive matrix in $\mathcal{C}^{*2}$ to decompose with cp-rank equal to 2. Now use Theorem 8.1.
Circular, $n=3$	1	1	As the matrix has cp-rank equal to 1.
	2	∞	Theorem 8.1.
	3	∞	Consider stopping Algorithm 2 after step 4, effectively leaving a strictly positive matrix in $\mathcal{C}^{*2}$ to decompose with cp-rank equal to 2. Now use Theorem 8.1.
Circular, $n>3$	n	1 or 2	Method 5.3 results in either one or two solutions for y. Considering acyclic matrices with cp-rank equal to $n-1$, we see that each value of y gives exactly one minimal rank-one decomposition set. This was also found in [30], where the authors found that the number of minimal rank-one decomposition sets of a completely positive circular matrix was dependent on the determinant of its comparison matrix.
$G(X)$ has exactly one cycle, which is of length equal to 3	$n-2$	1	From the form that a minimal rank-one decomposition set must take.
	$n-1$ or n	∞	Algorithm 1 followed by Algorithm 2 and considering circular matrices with $n=3$ and cp-rank greater than or equal to 2.
$G(X)$ has exactly one cycle, which is of length greater than 3	n	1 or 2	Algorithm 1 followed by Method 5.3 gives the form that a minimal rank-one decomposition set must take, and then we consider circular matrices with $n>3$.

Copyright © by SIAM. Unauthorized reproduction of this article is prohibited.

Downloaded 10/23/12 to 129.125.63.113. Redistribution subject to SIAM license or copyright; see http://www.siam.org/journals/ojsa.php

We now look at the two cases given in the theorem.

(a) We must have that $\theta_{kl} = \pi/2$, and so v_k and v_l must lie on perpendicular axes. This implies that $\mathcal{V}$ is unique up to a swapping of coordinates and therefore there is exactly one minimal rank-one decomposition set.

(b) We must have that $\theta_{kl} < \pi/2$. This gives us the freedom to rotate $\mathcal{V}$ while keeping it within $\mathbb{R}^2_+$; therefore there must be infinitely many minimal rank-one decomposition sets.

This completes the proof of the theorem. ∎

In Table 8.1 we now use this result to consider different types of matrices with the conditions given at the start of this section, i.e., completely positive and connected and all the diagonal elements strictly positive. The matrices we look at can, in fact, be easily extended to all the types of matrices that our method can check and decompose. For finding the cp-rank we simply consider how our method would work through this type of matrix.

Acknowledgments. This work was done while Mirjam Dür was working at the Johann Bernoulli Institute of Mathematics and Computer Science, University of Groningen, The Netherlands. The authors would like to thank the anonymous referees for their helpful comments.

REFERENCES

[1] F. Barioli and A. Berman, *The maximal cp-rank of rank k completely positive matrices*, Linear Algebra Appl., 363 (2003), pp. 17–33.

[2] A. Berman and R. Grone, *Bipartite completely positive matrices*, Math. Proc. Cambridge Philos. Soc., 103 (1988), pp. 269–276.

[3] A. Berman and D. Hershkowitz, *Combinatorial results on completely positive matrices*, Linear Algebra Appl., 95 (1987), pp. 111–125.

[4] A. Berman and U. G. Rothblum, *A note on the computation of the CP-rank*, Linear Algebra Appl., 419 (2006), pp. 1–7.

[5] A. Berman and N. Shaked-Monderer, *Completely Positive Matrices*, World Scientific, River Edge, NJ, 2003.

[6] I. M. Bomze, *Linear-time copositivity detection for tridiagonal matrices and extension to block-tridiagonality*, SIAM J. Matrix Anal. Appl., 21 (2000), pp. 840–848.

[7] I. M. Bomze, M. Dür, E. de Klerk, C. Roos, A. J. Quist, and T. Terlaky, *On copositive programming and standard quadratic optimization problems*, J. Global Optim., 18 (2000), pp. 301–320.

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

[9] P. J. C. Dickinson, *An improved characterisation of the interior of the completely positive cone*, Electron. J. Linear Algebra, 20 (2010), pp. 723–729.

[10] P. J. C. Dickinson and L. Gijben, *On the computational complexity of membership problems for the completely positive cone and its dual*, Math. Program., submitted.

[11] B. Dong, M. M. Lin, and M. T. Chu, *Nonnegative Rank Factorization via Rank Reduction*, preprint, 2009; available online at http://www4.ncsu.edu/~mtchu/Research/Papers/NRF04d.pdf.

[12] J. H. Drew, C. R. Johnson, and R. Loewy, *Completely positive matrices associated with M-matrices*, Linear Multilinear Algebra, 37 (1994), pp. 303–310.

[13] M. Dür, *Copositive programming—A survey*, in Recent Advances in Optimization and Its Applications in Engineering, M. Diehl, F. Glineur, E. Jarlebring, and W. Michiels, eds., Springer, Berlin, Heidelberg, 2010, pp. 3–20.

[14] L. J. Gray and D. G. Wilson, *Nonnegative factorization of positive semidefinite nonnegative matrices*, Linear Algebra Appl., 31 (1980), pp. 119–127.

[15] J.-B. Hiriart-Urruty and A. Seeger, *A variational approach to copositive matrices*, SIAM Rev., 52 (2010), pp. 593–629.

[16] K. D. Ikramov, *Linear-time algorithm for verifying the copositivity of an acyclic matrix*, Comput. Math. Math. Phys., 42 (2002), pp. 1701–1703.

Copyright © by SIAM. Unauthorized reproduction of this article is prohibited.

[17] V. Kalofolias and E. Gallopoulos, *Computing symmetric nonnegative rank factorizations*, Linear Algebra Appl., 436 (2012), pp. 421–435.

[18] M. Kaykobad, *On nonnegative factorization of matrices*, Linear Algebra Appl., 96 (1987), pp. 27–33.

[19] N. Kogan and A. Berman, *Characterization of completely positive graphs*, Discrete Math., 114 (1992), pp. 297–304.

[20] Y. Li, A. Kummert, and A. Frommer, *A linear programming based analysis of the CP-rank of completely positive matrices*, Int. J. Appl. Math. Comput. Sci., 14 (2004), pp. 25–31.

[21] T. L. Markham, *Factorizations of completely positive matrices*, Math. Proc. Cambridge Philos. Soc., 69 (1971), pp. 53–58.

[22] J. E. Maxfield and H. Minc, *On the matrix equation $X'X = A$*, Proc. Edinb. Math. Soc. (2), 13 (1962), pp. 125–129.

[23] K. Murota, *Discrete Convex Analysis*, SIAM, Philadelphia, 2003.

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

[25] Y. Nesterov and A. Nemirovskii, *Interior-Point Polynomial Algorithms in Convex Programming*, SIAM, Philadelphia, 1994.

[26] J. C. Preisig, *Copositivity and the minimization of quadratic functions with nonnegativity and quadratic equality constraints*, SIAM J. Control Optim., 34 (1996), pp. 1135–1150.

[27] A. J. Quist, E. de Klerk, C. Roos, and T. Terlaky, *Copositive relaxation for general quadratic programming*, Optimization Methods Softw., 9 (1998), pp. 185–208.

[28] N. Shaked-Monderer, *A note on upper bounds of the cp-rank*, Linear Algebra Appl., 431 (2009), pp. 2407–2413.

[29] C. Xu and J.-S. Li, *A note on completely positive graphs*, Systems Sci. Math. Sci., 13 (2000), pp. 121–125.

[30] X.-D. Zhang and J.-S. Li, *Completely positive matrices having cyclic graphs*, J. Math. Res. Exposition, 20 (2000), pp. 27–31.

Downloaded 10/23/12 to 129.125.63.113. Redistribution subject to SIAM license or copyright; see http://www.siam.org/journals/ojsa.php

Copyright © by SIAM. Unauthorized reproduction of this article is prohibited.