Int. J. Appl. Math. Comput. Sci., 2004, Vol. 14, No. 1, 25–31

A LINEAR PROGRAMMING BASED ANALYSIS OF THE CP-RANK OF COMPLETELY POSITIVE MATRICES

YINGBO LI*, ANTON KUMMERT*
ANDREAS FROMMER**

* Department of Electrical and Information Engineering
University of Wuppertal, Rainer-Gruenter-Street 21
42119 Wuppertal, Germany
e-mail: {yingbo, kummert}@uni-wuppertal.de

** Department of Mathematics
University of Wuppertal, Gauß Street 20
42119 Wuppertal, Germany
e-mail: frommer@math.uni-wuppertal.de

A real matrix $\boldsymbol{A}$ is said to be *completely positive* (CP) if it can be decomposed as $\boldsymbol{A} = \boldsymbol{B}\boldsymbol{B}^T$, where the real matrix $\boldsymbol{B}$ has exclusively non-negative entries. Let k be the rank of $\boldsymbol{A}$ and Φ_k the least possible number of columns of the matrix $\boldsymbol{B}$, the so-called completely positive rank (*cp-rank*) of $\boldsymbol{A}$. The present work is devoted to a study of a general upper bound for the cp-rank of an arbitrary completely positive matrix $\boldsymbol{A}$ and its dependence on the ordinary $\operatorname{rank} k$. This general upper bound of the cp-rank has been proved to be at most $k(k+1)/2$. In a recent pioneering work of Barioli and Berman it was slightly reduced by one, which means that $\Phi_k \le k(k+1)/2 - 1$ holds for $k \ge 2$. An alternative constructive proof of the same result is given in the present paper based on the properties of the simplex algorithm known from linear programming. Our proof illuminates complete positivity from a different point of view. Discussions concerning dual cones are not needed here. In addition to that, the proof is of constructive nature, i.e. starting from an arbitrary decomposition $\boldsymbol{A} = \boldsymbol{B}_1\boldsymbol{B}_1^T$ ($\boldsymbol{B}_1 \ge 0$) a new decomposition $\boldsymbol{A} = \boldsymbol{B}_2\boldsymbol{B}_2^T$ ($\boldsymbol{B}_2 \ge 0$) can be generated in a constructive manner, where the number of column vectors of $\boldsymbol{B}_2$ does not exceed $k(k+1)/2 - 1$. This algorithm is based mainly on the well-known techniques stemming from linear programming, where the pivot step of the simplex algorithm plays a key role.

Keywords: completely positive matrices, cp-rank, linear programming, simplex algorithm, basic feasible solution, pivot process

1. Introduction

An $n \times n$ real symmetric matrix $\boldsymbol{A}$ is called *completely positive* (belonging to the set CP of completely positive matrices) if an entry-wise non-negative $n \times m$ matrix $\boldsymbol{B}$ exists with $\boldsymbol{A} = \boldsymbol{B}\boldsymbol{B}^T$. The product $\boldsymbol{A} = \boldsymbol{B}\boldsymbol{B}^T$ can alternatively be written in its rank 1 representation as $\boldsymbol{A} = \sum_{i=1}^{m} \boldsymbol{b}_i\boldsymbol{b}_i^T$, in which each $\boldsymbol{b}_i \in \mathbb{R}_+^n$ denotes the i-th non-negative column of the matrix $\boldsymbol{B}$. The minimum number of columns m, for which such a factorization $\boldsymbol{A} = \boldsymbol{B}\boldsymbol{B}^T$ exists, is called the cp-rank of $\boldsymbol{A}$, and denoted by Φ_k in relation to the ordinary $\operatorname{rank} k$ of $\boldsymbol{A}$. It is to be pointed out that all vectors throughout this paper are column vectors, unless explicitly defined to be row vectors.

The non-negative decomposition of completely positive matrices is of interest in various applications including, e.g., the study of block designs arising in combinatorial analysis (Hall, 1967), problems in statistics, the theory of inequalities, energy conservation (Gray and Wilson, 1980) and models of DNA evolution (Kelly, 1994).

No definitive test is known yet to determine whether a given real symmetric matrix $\boldsymbol{A}$ is completely positive. Many studies on this problem have been performed (Berman and Plemmons, 1979; Berman and Shaked-Monderer, 2003). One sufficient condition for a non-negative symmetric matrix to be completely positive given in (Drew *et al.*, 1994) says that the comparison matrix of $\boldsymbol{A}$ is an M-matrix. A special case of this sufficient condition is that the matrix $\boldsymbol{A}$ is diagonally dominant (Kaykobad, 1988). A qualitative characterization of completely positive matrices using graphs is that the graph of a cp matrix contains no odd cycle of length 5 or more (Berman, 1993; Kogan and Berman, 1993).

The question of a general upper bound for the cp-rank which works for any completely positive matrix of dimension $n \times n$ was asked by Hall and Newman (1963), with the conclusion that $\Phi_k < 2^n$. The definition of the cp-rank as above implies that Φ_k is obviously greater than or equal to the ordinary $\operatorname{rank} k$ of $\boldsymbol{A}$. Hannah and Laffey (1983) derived the inequality $\Phi_k \leq n(n+1)/2$ (which is independent of k, with $k \leq n$). Recently, the sharpened upper bound $k(k+1)/2$ for Φ_k was reduced by Barioli and Berman (2003) to $k(k+1)/2-1$ if $k \geq 2$. In (Drew *et al.*, 1994) it was conjectured that $\Phi_k < n^2/4$. However, this conjecture was proved only for matrices with special graphs, which are bipartite with the two parts as balanced as possible, or for graphs which contain no odd cycle of length 5 or more (Drew and Johnson, 1996), or for all graphs on 5 vertices which are not a complete graph (Berman, 1993). In the present paper we give a constructive proof alternative to Barioli and Berman's result that

$$\Phi_k \leq \frac{k(k+1)}{2} - 1, \quad k \geq 2.$$

The technique of the proof presented here involves the properties of the simplex algorithm known from linear programming and illuminates the problem from a different point of view. Above all, discussions concerning dual cones are not required. Linear programming is the process of minimizing (or maximizing) a linear real-valued objective function subject to a finite number of linear equality and inequality constraints. Linear programming problems are encountered in many branches of technology, science, and economics. In this work, the problem of determining an upper bound for the cp-rank of completely positive matrices is reduced to the problem of determining the maximal number of non-zero elements in the basic feasible solutions of a linear programming problem. By means of appropriate matrix operations and based on the properties of the simplex algorithm, the desired result will be achieved. Furthermore, according to the described constructive procedure, such a decomposition which satisfies the claimed bound constraint can be generated step by step from a known decomposition of an arbitrary size.

2. First Reduction Step

We start with a known *rank 1 representation* $\boldsymbol{A} = \sum_{i=1}^{m} \boldsymbol{b}_i \boldsymbol{b}_i^T$ of a given $n \times n$ completely positive matrix, where m is an arbitrary large, however finite, integer. From now on, we assume that no column $\boldsymbol{b}_i$ of $\boldsymbol{B}$ is zero, since otherwise such a column could have been removed from $\boldsymbol{B}$. After the application of the vec-operator (stacking the columns of a matrix to form a column vector) to both the sides, the following expression can be obtained:

$$\operatorname{vec}(\boldsymbol{A}) = \operatorname{vec}\Big(\sum_{i=1}^{m} \boldsymbol{b}_i \boldsymbol{b}_i^T\Big) = \sum_{i=1}^{m} \operatorname{vec}(\boldsymbol{b}_i \boldsymbol{b}_i^T)$$
$$= \sum_{i=1}^{m} \boldsymbol{b}_i \otimes \boldsymbol{b}_i, \tag{1}$$

where '$\otimes$' denotes the Kronecker product. Consequently, we may define an $n^2 \times m$ matrix

$$C = \Big[\boldsymbol{b}_1 \otimes \boldsymbol{b}_1, \quad \boldsymbol{b}_2 \otimes \boldsymbol{b}_2, \quad \ldots, \quad \boldsymbol{b}_m \otimes \boldsymbol{b}_m \Big] \tag{2}$$

with $\boldsymbol{c}_i = \boldsymbol{b}_i \otimes \boldsymbol{b}_i$, $i = 1, \ldots, m$, so that (1) can be represented as

$$\operatorname{vec}(\boldsymbol{A}) = \boldsymbol{C}\Big[1, \quad 1, \quad \ldots, \quad 1\Big]^T. \tag{3}$$

It is evident that the non-negative linear system of equations $\operatorname{vec}(\boldsymbol{A}) = \boldsymbol{C}\boldsymbol{\lambda}$ with $\lambda_i \geq 0$ has at least one valid solution, namely the column vector $\boldsymbol{\lambda}_0 = [1,\ 1,\ \ldots,\ 1\,]^T$. In fact, from now on we may look upon $\boldsymbol{C}\boldsymbol{\lambda} = \operatorname{vec}(\boldsymbol{A})$ with $\lambda_i \geq 0$ as the linear constraints for an arbitrary linear objective function, which is a linear programming problem in a standard form with all variables of $\boldsymbol{\lambda}$ non-negatively constrained.

First of all, it is important to indicate that the feasible polyhedron of this linear programming problem is on the one hand non-empty due to the existence of the feasible solution $\boldsymbol{\lambda}_0$ and, on the other hand, bounded due to the non-negativity of $\boldsymbol{C}$, $\operatorname{vec}(\boldsymbol{A})$ and $\boldsymbol{\lambda}$, and the fact that no column in $\boldsymbol{B}$, and thus no column in $\boldsymbol{C}$, is zero. Thus, the feasible region represents a polytope. Consequently, the available feasible solutions which satisfy these linear constraints can be expressed as convex combinations of the so-called basic feasible solutions, i.e. the extreme points (or vertices) of the polytope.

According to the properties of linear programming problems, in the basic feasible solutions of $\boldsymbol{C}\boldsymbol{\lambda} = \operatorname{vec}(\boldsymbol{A})$ with $\lambda_i \geq 0$, only the elements λ_i which correspond to a maximal set of linear independent columns of $\boldsymbol{C}$ may be non-zero. Therefore, all of the basic feasible solutions possess at most $\operatorname{rank}(\boldsymbol{C})$ strictly positive elements. Because of the existence of the solution $\boldsymbol{\lambda}_0 = [1,\ 1,\ \ldots,\ 1\,]^T$, it can be concluded that at least one basic feasible solution with at most $\operatorname{rank}(\boldsymbol{C})$ strictly positive elements for $\boldsymbol{C}\boldsymbol{\lambda} = \operatorname{vec}(\boldsymbol{A})$ with $\lambda_i \geq 0$ exists. Writing $\operatorname{rank}(\boldsymbol{C}) = r$, and, if necessary, permuting the columns of $\boldsymbol{B}$, we may assume that there is a basic feasible solution of the form $\boldsymbol{\lambda}_B = [\lambda_1,\ \lambda_2,\ \ldots,\ \lambda_r,\ 0,\ \ldots,\ 0\,]^T$ with $\lambda_i \geq 0$ for $1 \leq i \leq r$. It is obvious that

$$\operatorname{vec}(\boldsymbol{A}) = \boldsymbol{C}\boldsymbol{\lambda}_B = \sum_{i=1}^{r} \boldsymbol{c}_i \lambda_i = \sum_{i=1}^{r} \Big(\sqrt{\lambda_i}\boldsymbol{b}_i) \otimes (\sqrt{\lambda_i}\boldsymbol{b}_i\Big), \tag{4}$$

which is equivalent to

$$\boldsymbol{A} = \sum_{i=1}^{r} \left(\sqrt{\lambda_i}\boldsymbol{b}_i\right)\left(\sqrt{\lambda_i}\boldsymbol{b}_i\right)^T. \tag{5}$$

Defining $\widetilde{\boldsymbol{B}} = [\sqrt{\lambda_1}\boldsymbol{b}_1,\ \sqrt{\lambda_2}\boldsymbol{b}_2,\ \dots,\ \sqrt{\lambda_r}\boldsymbol{b}_r\ ]$, we have established an alternative non-negative factorization of $\boldsymbol{A}$, namely

$$\boldsymbol{A} = \widetilde{\boldsymbol{B}}\widetilde{\boldsymbol{B}}^T = \left[\sqrt{\lambda_1}\boldsymbol{b}_1, \quad \dots, \quad \sqrt{\lambda_r}\boldsymbol{b}_r\ \right] \cdot \left[\sqrt{\lambda_1}\boldsymbol{b}_1, \quad \dots, \quad \sqrt{\lambda_r}\boldsymbol{b}_r\ \right]^T. \tag{6}$$

It should be noted that a basic feasible solution can always be generated constructively by using linear programming techniques (Glashoff and Gustafson, 1983; Karloff, 1991).

3. Upper Bounds on the CP-Rank

In this section we first clarify that the cp-rank of $\boldsymbol{A}$, Φ_k, is bounded by the rank of the matrix $\boldsymbol{C}$, which is in turn limited by $k(k+1)/2$ again, where $k = \operatorname{rank}(\boldsymbol{A})$. The proof of the sharpened upper bound of Φ_k, namely $k(k+1)/2-1$, will be provided afterwards.

Lemma 1. *The cp-rank Φ_k of $\boldsymbol{A} \in \mathrm{CP}$ is bounded by the rank of the matrix $\boldsymbol{C}$, i.e. we have*

$$\Phi_k \leq r = \operatorname{rank}(\boldsymbol{C}). \tag{7}$$

Proof. Based on the non-negative factorization of $\boldsymbol{A}$ achieved in (6), which can always be realized constructively from an arbitrary non-negative decomposition of $\boldsymbol{A}$, and the definition of Φ_k as the least number of non-zero columns in any decomposition, it is obvious that $\Phi_k \leq r = \operatorname{rank}(\boldsymbol{C})$. ∎

Our final goal is to prove that Φ_k is bounded by $k(k+1)/2-1$. In order to keep the proof clear and progressive, we will first prove that $\operatorname{rank}(\boldsymbol{C}) \leq k(k+1)/2$.

Theorem 1. *The ordinary rank of matrix $\boldsymbol{C}$ as defined in (2) is bounded by*

$$\operatorname{rank}(\boldsymbol{C}) \leq \frac{k(k+1)}{2}. \tag{8}$$

Proof. Since $\boldsymbol{A} = \boldsymbol{B}\boldsymbol{B}^T$ with $\operatorname{rank}(\boldsymbol{A}) = k$, it follows immediately that $\operatorname{rank}(\boldsymbol{B}) = k$. With respect to a geometric interpretation, this means that the dimension of the vector space generated by the $\boldsymbol{b}_i$'s is k, i.e. $\dim(\operatorname{span}(\boldsymbol{b}_i \mid 1 \leq i \leq m)) = k$. As a consequence, there must be an isometry available which maps $\boldsymbol{b}_i \in \mathbb{R}^n_+$ onto $\mathbb{R}^k$, $i = 1,\dots,m$. In matrix terms, this means that we may find an orthonormal matrix $\boldsymbol{V}$ ($\boldsymbol{V}\boldsymbol{V}^T = \boldsymbol{I}$, where $\boldsymbol{I}$ is the identity matrix), so that $\boldsymbol{V}\boldsymbol{B} = \binom{\boldsymbol{F}}{\boldsymbol{0}}$, the dimension of $\boldsymbol{F}$ being $k \times m$. Computationally, $\boldsymbol{V}$ and $\boldsymbol{F}$ can be obtained from the standard QR-decomposition of $\boldsymbol{B}$, see (Golub and Van Loan, 1989). The following equality results from this isometry:

$$\begin{aligned}(\boldsymbol{V}\otimes\boldsymbol{V})\boldsymbol{C} &= (\boldsymbol{V}\otimes\boldsymbol{V})\left[\boldsymbol{b}_1\otimes\boldsymbol{b}_1, \quad \dots, \quad \boldsymbol{b}_m\otimes\boldsymbol{b}_m\ \right] \\ &= \left[(\boldsymbol{V}\boldsymbol{b}_1)\otimes(\boldsymbol{V}\boldsymbol{b}_1),\dots,(\boldsymbol{V}\boldsymbol{b}_m)\otimes(\boldsymbol{V}\boldsymbol{b}_m)\right] \\ &= \left[\begin{pmatrix}\boldsymbol{f}_1\\ \boldsymbol{0}\end{pmatrix}\otimes\begin{pmatrix}\boldsymbol{f}_1\\ \boldsymbol{0}\end{pmatrix},\dots,\begin{pmatrix}\boldsymbol{f}_m\\ \boldsymbol{0}\end{pmatrix}\otimes\begin{pmatrix}\boldsymbol{f}_m\\ \boldsymbol{0}\end{pmatrix}\right],\end{aligned} \tag{9}$$

where $\boldsymbol{f}_i = \boldsymbol{V}\boldsymbol{b}_i$ is the i-th column vector of matrix $\boldsymbol{F}$. Since there are at most $k(k+1)/2$ distinct non-zero row vectors in the matrix $(\boldsymbol{V}\otimes\boldsymbol{V})\boldsymbol{C}$ and $\boldsymbol{V}\otimes\boldsymbol{V}$ is orthonormal and thus non-singular, we have

$$r = \operatorname{rank}(\boldsymbol{C}) = \operatorname{rank}((\boldsymbol{V}\otimes\boldsymbol{V})\boldsymbol{C}) \leq \frac{k(k+1)}{2}.$$

∎

Corollary 1. *In view of (7) and Theorem 1, we can conclude that Φ_k is at least bounded by*

$$\Phi_k \leq \frac{k(k+1)}{2}. \tag{10}$$

Let us note that the above proofs are constructive: Given $\boldsymbol{A} = \boldsymbol{B}\boldsymbol{B}^T$, we have shown how to obtain $\widetilde{\boldsymbol{B}}$ in $\boldsymbol{A} = \widetilde{\boldsymbol{B}}\widetilde{\boldsymbol{B}}^T$ by computing a basic feasible solution of a certain polytope.

The next step we want to take is proving the sharpened bound $\Phi_k \leq k(k+1)/2-1$ for $k \geq 2$. The corresponding proof can be divided into distinct steps. If for a given matrix $\boldsymbol{A}$ the associated matrix $\boldsymbol{C}$ has a rank less than $k(k+1)/2$, we can stop, otherwise we have $\operatorname{rank}(\boldsymbol{C}) = k(k+1)/2$ and the reduction procedure must be continued. The outline of this algorithm is as follows: After extension of $\boldsymbol{C}$ by a column vector, which is the Kronecker product of an adequately selected column of $\boldsymbol{A}$ with itself, we will show that the rank of the extended matrix $\boldsymbol{C}$ is identical to its original rank. Then a pivot step known from simplex techniques will be used, so that a new non-negative decomposition $\boldsymbol{A} = \boldsymbol{G}\boldsymbol{G}^T$ could be created, whereby the column number of $\boldsymbol{G}$ does not exceed that of $\widetilde{\boldsymbol{B}}$. Relying on some requirements regarding $\boldsymbol{G}$, the column number of G can be further reduced by 1 after some straightforward computations. By this last step the proof will be completed.

Theorem 2. *For $k \geq 2$ the cp-rank of any completely positive matrix $\boldsymbol{A}$ with its ordinary rank equal to k is* *bounded by*

$$\text{cp-rank}\,(\boldsymbol{A}) = \Phi_k \leq \frac{k(k+1)}{2} - 1. \tag{11}$$

Proof. From Theorem 1 and Corollary 1 we know that $\Phi_k \leq \text{rank}(\boldsymbol{C}) \leq k(k+1)/2$. If $\text{rank}(\boldsymbol{C}) < k(k+1)/2$, nothing has to be proven, so that from now on we assume that $r = \text{rank}(\boldsymbol{C}) = k(k+1)/2$.

Before the actual proof, we would like to point out a useful property of the matrix $\widetilde{\boldsymbol{B}}$ (cf. (6)) at first. From the decomposition $\boldsymbol{A} = \widetilde{\boldsymbol{B}}\widetilde{\boldsymbol{B}}^T$ in (6) with $\widetilde{\boldsymbol{B}} = [\sqrt{\lambda_1}\boldsymbol{b}_1,\ \sqrt{\lambda_2}\boldsymbol{b}_2,\ \ldots,\ \sqrt{\lambda_r}\boldsymbol{b}_r\ ]$, the fact that $2 \leq \text{rank}(\widetilde{\boldsymbol{B}}) = k \leq n$ and the assumption $r = k(k+1)/2$, it can be concluded that at least one of the rows of the non-negative matrix $\widetilde{\boldsymbol{B}}$ has more than one non-zero entry. Otherwise, if each row of the non-negative matrix $\widetilde{\boldsymbol{B}}$ had at most one non-zero element, all columns of $\widetilde{\boldsymbol{B}}$ (which does not have zero columns) would be linearly independent, so $k = r$ and thus $\Phi_k = r = k \leq k(k+1)/2 - 1$ for $k \geq 2$ and we have a contradiction to the assumption $r = k(k+1)/2$. So, for the rest of the proof we may therefore assume that at least one row of $\widetilde{\boldsymbol{B}}$ contains at least two non-zero entries.

Now we extend $\text{vec}(\boldsymbol{A}) = \boldsymbol{C}\boldsymbol{\lambda}_B$ in (4), which is associated with $\boldsymbol{A} = [\boldsymbol{a}_1,\ \boldsymbol{a}_2,\ \ldots,\ \boldsymbol{a}_n\ ]$ and $\text{rank}(\boldsymbol{C}) = r = k(k+1)/2$, to

$$\text{vec}(\boldsymbol{A}) = \begin{bmatrix}\boldsymbol{C}, & \boldsymbol{a}_j \otimes \boldsymbol{a}_j\end{bmatrix}\boldsymbol{\lambda}_{\text{E}}, \tag{12}$$

with $\boldsymbol{\lambda}_{\text{E}} = [\begin{smallmatrix}\boldsymbol{\lambda}_B\\ \mathbf{0}\end{smallmatrix}]$, where, in view of the above discussion, the integer j, $1 \leq j \leq n$, can be chosen in such a way that the j-th row of non-negative $\widetilde{\boldsymbol{B}}$ has at least two non-zero entries. The rank of the extended matrix $[\boldsymbol{C},\ \boldsymbol{a}_j \otimes \boldsymbol{a}_j\ ]$ in (12) is unmodified compared with that of the matrix $\boldsymbol{C}$, i.e.

$$\text{rank}\left(\begin{bmatrix}\boldsymbol{C}, & \boldsymbol{a}_j \otimes \boldsymbol{a}_j\end{bmatrix}\right) = \text{rank}(\boldsymbol{C}) = r = \frac{k(k+1)}{2}, \tag{13}$$

as we show now. The orthonormal matrix $\boldsymbol{V}$ from the proof of Theorem 1 will be reused. From $\boldsymbol{A} = \boldsymbol{B}\boldsymbol{B}^T$ and $\boldsymbol{V}\boldsymbol{B} = (\begin{smallmatrix}\boldsymbol{F}\\ \mathbf{0}\end{smallmatrix})$ it follows that $\boldsymbol{V}\boldsymbol{A} = \boldsymbol{V}\boldsymbol{B}\boldsymbol{B}^T = (\begin{smallmatrix}\boldsymbol{F}\\ \mathbf{0}\end{smallmatrix})\boldsymbol{B}^T = (\begin{smallmatrix}\boldsymbol{E}\\ \mathbf{0}\end{smallmatrix})$ and, accordingly, $\boldsymbol{V}\boldsymbol{a}_j = (\begin{smallmatrix}\boldsymbol{e}_j\\ \mathbf{0}\end{smallmatrix})$ with $\boldsymbol{e}_j$ as the j-th column of $\boldsymbol{E}$, whereby the dimension of $\boldsymbol{E}$ is $k \times n$, so that

$$\begin{aligned}
&(\boldsymbol{V} \otimes \boldsymbol{V})\begin{bmatrix}\boldsymbol{C}, & \boldsymbol{a}_j \otimes \boldsymbol{a}_j\end{bmatrix}\\
&= (\boldsymbol{V} \otimes \boldsymbol{V})\begin{bmatrix}\boldsymbol{b}_1 \otimes \boldsymbol{b}_1, & \ldots, & \boldsymbol{b}_m \otimes \boldsymbol{b}_m & ,\boldsymbol{a}_j \otimes \boldsymbol{a}_j\end{bmatrix}\\
&= \Big[(\boldsymbol{V}\boldsymbol{b}_1) \otimes (\boldsymbol{V}\boldsymbol{b}_1), \quad \ldots, \quad (\boldsymbol{V}\boldsymbol{b}_m) \otimes (\boldsymbol{V}\boldsymbol{b}_m),\\
&\qquad (\boldsymbol{V}\boldsymbol{a}_j) \otimes (\boldsymbol{V}\boldsymbol{a}_j)\ \Big]\\
&= \left[\begin{pmatrix}\boldsymbol{f}_1\\ \mathbf{0}\end{pmatrix} \otimes \begin{pmatrix}\boldsymbol{f}_1\\ \mathbf{0}\end{pmatrix}, \ldots, \begin{pmatrix}\boldsymbol{f}_m\\ \mathbf{0}\end{pmatrix} \otimes \begin{pmatrix}\boldsymbol{f}_m\\ \mathbf{0}\end{pmatrix},\right.\\
&\qquad \left.\begin{pmatrix}\boldsymbol{e}_j\\ \mathbf{0}\end{pmatrix} \otimes \begin{pmatrix}\boldsymbol{e}_j\\ \mathbf{0}\end{pmatrix}\right].
\end{aligned} \tag{14}$$

As in the proof of Theorem 1, we have $\text{rank}([\boldsymbol{C},\ \boldsymbol{a}_j \otimes \boldsymbol{a}_j\ ]) \leq k(k+1)/2$. Since $\text{rank}(\boldsymbol{C})$ was assumed to be equal to $k(k+1)/2$ and for the reason that $\text{rank}(\boldsymbol{C}) \leq \text{rank}([\boldsymbol{C},\ \boldsymbol{a}_j \otimes \boldsymbol{a}_j\ ])$, we get $\text{rank}([\boldsymbol{C},\ \boldsymbol{a}_j \otimes \boldsymbol{a}_j\ ]) = \text{rank}(\boldsymbol{C}) = k(k+1)/2$.

Because all entries of the extended linear constraints system (12) are also non-negative, and there exists a feasible solution $\boldsymbol{\lambda}_{\text{E}} = [\begin{smallmatrix}\boldsymbol{\lambda}_B\\ \mathbf{0}\end{smallmatrix}]$, the feasible region is bounded and non-empty, i.e. it is again a polytope. Consequently, basic feasible solutions of the linear programming problem (12) exist and have at most $r = k(k+1)/2$ non-negative variables.

Furthermore, $\boldsymbol{\lambda}_{\text{E}}$ is a basic solution where the basis is represented exactly by the variables $\lambda_1, \ldots, \lambda_r$. To be more specific, since $\text{rank}([\boldsymbol{C},\ \boldsymbol{a}_j \otimes \boldsymbol{a}_j\ ]) = \text{rank}(\boldsymbol{C})$, any basis of the old linear system is also a basis of the extended system. (This would not be true if the rank of the extended system were increased by one when compared with that of the old system.) Suppose that the simplex tableau associated with the extended linear system is given for the basis $\lambda_1, \ldots, \lambda_r$, i.e. λ_{m+1}, the variable associated with column $\boldsymbol{a}_j \otimes \boldsymbol{a}_j$ of the extended system is a non-basic variable yet.

Next, the *simplex algorithm* will be considered, which has been the method of choice used to solve linear programming problems for decades. Due to the fact that the basic feasible solutions are nothing else than extreme points of the polytope, and the optimal point of the objective function subject to the polytope is always among these vertices, the simplex algorithm can be regarded as an ordered way of scanning through such vertices. The simplex algorithm starts from an arbitrary vertex of the feasible polytope and tries to find a cheaper adjacent vertex, till ideally no neighbouring vertex with a cheaper cost is found.

The process of changing the basis to move to an adjacent vertex is called the *pivot step*. Basically, a pivot step takes one basic variable out of the basis and, at the same time, it shifts another, originally non-basic, variable into the basis. Hence, during a pivot step carried out on the simplex tableau defined above, the last variable of $\boldsymbol{\lambda}_{\text{E}}$, namely λ_{m+1}, can be deliberately forced to enter the basis, for which one of the formerly basic variables, say λ_s, leaves the basis.

It is worth noticing that through the exchange of the basic variable and the corresponding modification of the basis, the maximal number of non-zero elements in the new basic feasible solution, $r = k(k+1)/2$, is unchanged, and all of the elements are still non-negative. Denoting by $d_1, \ldots, d_r$ the new non-negative basic variable values, we have

$$\begin{aligned}\operatorname{vec}(\boldsymbol{A}) &= \Big[\boldsymbol{b}_1 \otimes \boldsymbol{b}_1, \ldots, \boldsymbol{b}_{s-1} \otimes \boldsymbol{b}_{s-1}, \boldsymbol{a}_j \otimes \boldsymbol{a}_j, \\ &\quad \boldsymbol{b}_{s+1} \otimes \boldsymbol{b}_{s+1}, \ldots, \boldsymbol{b}_r \otimes \boldsymbol{b}_r\Big] \\ &\quad \cdot [d_1, \ldots, d_s, \ldots, d_r]^T. \end{aligned} \tag{15}$$

Following the same procedure as above, a new non-negative factorization of $\boldsymbol{A}$ of the form

$$\begin{aligned} A = &\Big[\sqrt{d_1}\boldsymbol{b}_1, \ldots, \sqrt{d_{s-1}}\boldsymbol{b}_{s-1}, \sqrt{d_s}\boldsymbol{a}_j, \\ &\sqrt{d_{s+1}}\boldsymbol{b}_{s+1}, \ldots, \sqrt{d_r}\boldsymbol{b}_r\Big] \\ &\cdot \Big[\sqrt{d_1}\boldsymbol{b}_1, \ldots, \sqrt{d_{s-1}}\boldsymbol{b}_{s-1}, \sqrt{d_s}\boldsymbol{a}_j, \\ &\sqrt{d_{s+1}}\boldsymbol{b}_{s+1}, \ldots, \sqrt{d_r}\boldsymbol{b}_r\Big]^T \end{aligned} \tag{16}$$

is obtained. If at least one of the non-negative numbers d_l, $l = 1, \ldots, r$, is zero, then $\text{cp-rank}(\boldsymbol{A}) \le r - 1 = k(k+1)/2 - 1$, so that the desired result is obtained. Otherwise we proceed as follows: After a reordering of the columns of the matrix $[\sqrt{d_1}\boldsymbol{b}_1, \ldots, \sqrt{d_{s-1}}\boldsymbol{b}_{s-1}, \sqrt{d_s}\boldsymbol{a}_j, \sqrt{d_{s+1}}\boldsymbol{b}_{s+1}, \ldots, \sqrt{d_r}\boldsymbol{b}_r]$ in such a way that the column $\sqrt{d_s}\boldsymbol{a}_j$ enters the last position, we have the decomposition

$$\begin{aligned}\boldsymbol{A} = \boldsymbol{G}\boldsymbol{G}^T = &\Big[\boldsymbol{g}_1, \ldots, \boldsymbol{g}_{r-1}, \sqrt{d_s}\boldsymbol{a}_j\Big] \\ &\cdot \Big[\boldsymbol{g}_1, \ldots, \boldsymbol{g}_{r-1}, \sqrt{d_s}\boldsymbol{a}_j\Big]^T \end{aligned} \tag{17}$$

with $\boldsymbol{g}_l = \Big[g_{1l}, \ldots, g_{nl}\Big]^T \ge 0$, $l = 1, \ldots, r-1$, where the $\boldsymbol{g}_l$'s denote the newly arranged non-negative columns. Based on these results, we finally can construct the non-negative decomposition

$$\boldsymbol{A} = \widetilde{\boldsymbol{G}}\widetilde{\boldsymbol{G}}^T, \tag{18}$$

with

$$\begin{cases} \widetilde{\boldsymbol{G}} = \Big[\boldsymbol{g}_1, \ldots, \boldsymbol{g}_{r-1}\Big] + \alpha \boldsymbol{a}_j \left[g_{j1}, \ldots, g_{j,r-1}\right], \\ \alpha = \sqrt{\left(\dfrac{(1-d_s a_{jj})}{\sum_{l=1}^{r-1} g_{jl}^2}\right)^2 + \dfrac{d_s}{\sum_{l=1}^{r-1} g_{jl}^2}} - \dfrac{(1-d_s a_{jj})}{\sum_{l=1}^{r-1} g_{jl}^2}. \end{cases} \tag{19}$$

Since $d_s > 0$, the scalar α is also strictly positive, i.e. $\alpha > 0$, which shows that $\widetilde{\boldsymbol{G}}$ is non-negative. Furthermore, α always exists since $\sum_{l=1}^{r-1} g_{jl}^2 \neq 0$, which is true since $\sum_{l=1}^{r-1} g_{jl}^2 = \sum_{l=1, l\neq s}^{r} d_l b_{jl}^2$, and we have chosen j in such a manner that at least two values from among b_{jl}, $l = 1, \ldots, r$ are non-zero. The validity of the decomposition in (18) can be simply checked by inserting (19) into (18) and using

$$\Big[\boldsymbol{g}_1, \ldots, \boldsymbol{g}_{r-1}\Big] \left[g_{j1}, \ldots, g_{j,r-1}\right]^T = (1 - d_s a_{jj})\boldsymbol{a}_j, \tag{20}$$

which follows from

$$\begin{aligned}\boldsymbol{a}_j = &\Big[\boldsymbol{g}_1, \ldots, \boldsymbol{g}_{r-1}, \sqrt{d_s}\boldsymbol{a}_j\Big] \\ &\cdot \Big[g_{j1}, \ldots, g_{j,r-1}, \sqrt{d_s}a_{jj}\Big]^T, \end{aligned} \tag{21}$$

see (17). In view of (18) and (19), the number of columns in $\widetilde{\boldsymbol{G}}$ equals $k(k+1)/2 - 1$. Hence we have proved that

$$\text{cp-rank}\,(\boldsymbol{A}) = \Phi_k \le r - 1 \le \frac{k(k+1)}{2} - 1. \tag{22}$$

■

In order to have all reduction steps in a condensed form without being interrupted by proofs, we summarize them in the form of the sequential algorithm, shown in Fig. 1.

4. Illustrative Example

In this section the constructive algorithm is demonstrated by applying it to an illustrative example. Consider a 3×3 completely positive matrix

$$\boldsymbol{A} = \begin{bmatrix} 4 & 3 & 4 \\ 3 & 7 & 7 \\ 4 & 7 & 12 \end{bmatrix},$$

whose rank also equals three. There exists a non-negative decomposition $\boldsymbol{A} = \boldsymbol{B}\boldsymbol{B}^T$ with

$$\boldsymbol{B} = \begin{bmatrix} 1 & 0 & 0 & 1 & 0 & 1 & 1 \\ 0 & 1 & 0 & 1 & 1 & 0 & 2 \\ 0 & 0 & 1 & 0 & 1 & 1 & 3 \end{bmatrix}.$$

start, given a cp-matrix $\boldsymbol{A}(n\times n)$ *with* $\operatorname{rank}(\boldsymbol{A})=k$, *let* $\boldsymbol{B}(n\times m)$ *be any known solution to* $\boldsymbol{A}=\boldsymbol{B}\boldsymbol{B}^T,\ \boldsymbol{B}\geq\boldsymbol{0}$
if column number of $\boldsymbol{B}$, m, *is less than* $k(k+1)/2$
  we have done, exit
otherwise
  generate the linear system $\operatorname{vec}(\boldsymbol{A})=\boldsymbol{C}\boldsymbol{\lambda}$,
  $\boldsymbol{\lambda}\geq 0$ *with* $\boldsymbol{C}$ *defined by*
  $\boldsymbol{C}=\Big[\boldsymbol{b}_1\otimes\boldsymbol{b}_1,\ \boldsymbol{b}_2\otimes\boldsymbol{b}_2,\ \ldots,\ \boldsymbol{b}_m\otimes\boldsymbol{b}_m\Big]$
compute a basic feasible solution $\boldsymbol{\lambda}_B$ *by a known linear programming method*
generate $\widetilde{\boldsymbol{B}}$ *by means of the non-zero elements of* $\boldsymbol{\lambda}_B$
if column number of $\widetilde{\boldsymbol{B}}$, r, *is less than* $k(k+1)/2$
  $\boldsymbol{A}=\widetilde{\boldsymbol{B}}\widetilde{\boldsymbol{B}}^T$ *is already the desired factorisation, done and exit*
otherwise
  $\boldsymbol{\lambda}_{\mathrm{E}}=\Big[\boldsymbol{\lambda}_{\mathrm{B}}^T\ \ 0\Big]^T$
extend $\boldsymbol{C}$ *by* $\boldsymbol{a}_j\otimes\boldsymbol{a}_j$, *with* $1\leq j\leq n$
generate the new linear system
$\operatorname{vec}(\boldsymbol{A})=\Big[\boldsymbol{C},\ \boldsymbol{a}_j\otimes\boldsymbol{a}_j\Big]\cdot\boldsymbol{\lambda}_{\mathrm{E}}$
apply one pivot step to bring non-basic variable λ_{m+1} *into the basis, which leads to a new basic feasible solution* $\boldsymbol{d}$
generate $\boldsymbol{G}$ *by means of the non-zero elements of* $\boldsymbol{d}$
if column number of $\boldsymbol{G}$ *is less than* $k(k+1)/2$
  $\boldsymbol{A}=\boldsymbol{G}\boldsymbol{G}^T$ *is already the desired factorisation, done and exit*
otherwise
  compute $\widetilde{\boldsymbol{G}}$ *according to (19), herewith we have*
  $\boldsymbol{A}=\widetilde{\boldsymbol{G}}\widetilde{\boldsymbol{G}}^T$ *and the column number of*
  $\widetilde{\boldsymbol{G}}$ *is at most* $k(k+1)/2-1$
end

Fig. 1. Summary of all algorithmic steps.

The linear system $\operatorname{vec}(\boldsymbol{A})=\boldsymbol{C}\boldsymbol{\lambda}$ is given by

$$\begin{bmatrix}4\\3\\4\\3\\7\\7\\4\\7\\12\end{bmatrix}=\begin{bmatrix}1&0&0&1&0&1&1\\0&0&0&1&0&0&2\\0&0&0&0&0&1&3\\0&0&0&1&0&0&2\\0&1&0&1&1&0&4\\0&0&0&0&1&0&6\\0&0&0&0&0&1&3\\0&0&0&0&1&0&6\\0&0&1&0&1&1&9\end{bmatrix}\begin{bmatrix}\lambda_1\\\lambda_2\\\lambda_3\\\lambda_4\\\lambda_5\\\lambda_6\\\lambda_7\end{bmatrix}.$$

By means of standard techniques, the basic solution

$$\boldsymbol{\lambda}_B=[5/3\ \ 5/3\ \ 1\ \ 2/3\ \ 0\ \ 1/2\ \ 7/6\,]^T$$

can be generated . From (6) we have a new non-negative factorization, namely $\boldsymbol{A}=\widetilde{\boldsymbol{B}}\widetilde{\boldsymbol{B}}^T$, where $\widetilde{\boldsymbol{B}}$ has only six columns,

$$\widetilde{\boldsymbol{B}}=\begin{bmatrix}\sqrt{5/3}&0&0&\sqrt{2/3}&\sqrt{1/2}&\sqrt{7/6}\\0&\sqrt{5/3}&0&\sqrt{2/3}&0&\sqrt{14/3}\\0&0&1&0&\sqrt{1/2}&\sqrt{21/2}\end{bmatrix}.$$

Our next step is to construct a new linear system by extending the matrix $\boldsymbol{C}$ by $\boldsymbol{a}_3\otimes\boldsymbol{a}_3$, which leads to

$$\begin{bmatrix}4\\3\\4\\3\\7\\7\\4\\7\\12\end{bmatrix}=\begin{bmatrix}1&0&0&1&0&1&1&16\\0&0&0&1&0&0&2&28\\0&0&0&0&0&1&3&48\\0&0&0&1&0&0&2&28\\0&1&0&1&1&0&4&49\\0&0&0&0&1&0&6&84\\0&0&0&0&0&1&3&48\\0&0&0&0&1&0&6&84\\0&0&1&0&1&1&9&144\end{bmatrix}\begin{bmatrix}\lambda_1\\\lambda_2\\\lambda_3\\\lambda_4\\\lambda_5\\\lambda_6\\\lambda_7\\\lambda_8\end{bmatrix}.$$

The Kronecker product $\boldsymbol{a}_3\otimes\boldsymbol{a}_3$ is used here, which fulfils the requirement that the third row of $\widetilde{\boldsymbol{B}}$ has at least two strictly positive entries. It can be verified that the rank of the new extended system matrix is the same as that of $\boldsymbol{C}$, namely $r=6$. Starting from the already available basic solution $\boldsymbol{\lambda}_E=[5/3\ \ 5/3\ \ 1\ \ 2/3\ \ 0\ \ 1/2\ \ 7/6\ \ \ 0\,]^T$, we can include λ_8 into the basis by applying only one pivot step, which leads to the new basic solution $\boldsymbol{\lambda}_E=[0\ \ 0.15\ \ 0.4\ \ 1.6\ \ 2.8\ \ 1.6\ \ 0\ \ 0.05\,]^T$. From this a newly arranged non-negative factorization $\boldsymbol{A}=\boldsymbol{G}\boldsymbol{G}^T$ results in

$$\boldsymbol{G}=\begin{bmatrix}0&0&\sqrt{8/5}&0&\sqrt{8/5}&\sqrt{4/5}\\\sqrt{3/20}&0&\sqrt{8/5}&\sqrt{14/5}&\sqrt{8/5}&\sqrt{49/20}\\0&\sqrt{2/5}&0&\sqrt{14/5}&0&\sqrt{36/5}\end{bmatrix}.$$

Since the matrix $\boldsymbol{G}$ has still six columns, we have to apply (19) in order to reduce the number of columns. In our example the scalar α has a positive value of $\sqrt{5/288}-1/12$, which leads to the final decomposition $\boldsymbol{A}=\widetilde{\boldsymbol{G}}\widetilde{\boldsymbol{G}}^T$ with

$$\widetilde{\boldsymbol{G}}=\begin{bmatrix}0&0.1225&1.2649&0.3241&1.5099\\0.3873&0.2144&1.2649&2.2406&0.4288\\0&1&0&2.6458&2\end{bmatrix}.$$

As has been expected, the non-negative factorisation $\boldsymbol{A} = \widetilde{\boldsymbol{G}}\widetilde{\boldsymbol{G}}^T$ has at most $k(k+1)/2 - 1 = 5$ columns. By equating $\widetilde{\boldsymbol{G}}\widetilde{\boldsymbol{G}}^T$ to $\boldsymbol{A}$ the correctness of this factorisation can be easily verified. It has thus been demonstrated that given any known decomposition of a cp-matrix $\boldsymbol{A}$ a decomposition with at most $k(k+1)/2 - 1$ columns can be generated in a constructive manner. Obviously, .to