

# Compound matrices: properties, numerical issues and analytical computations

Christos Kravvaritis · Marilena Mitrouli

Received: 1 February 2008 / Accepted: 9 June 2008  
© Springer Science + Business Media, LLC 2008

**Abstract** This paper studies the possibility to calculate efficiently compounds of real matrices which have a special form or structure. The usefulness of such an effort lies in the fact that the computation of compound matrices, which is generally noneffective due to its high complexity, is encountered in several applications. A new approach for computing the Singular Value Decompositions (SVD's) of the compounds of a matrix is proposed by establishing the equality (up to a permutation) between the compounds of the SVD of a matrix and the SVD's of the compounds of the matrix. The superiority of the new idea over the standard method is demonstrated. Similar approaches with some limitations can be adopted for other matrix factorizations, too. Furthermore, formulas for the  $n - 1$  compounds of Hadamard matrices are derived, which dodge the strenuous computations of the respective numerous large determinants. Finally, a combinatorial counting technique for finding the compounds of diagonal matrices is illustrated.

**Keywords** Compound matrices · Determinants · Matrix factorizations · Special matrices

**Mathematics Subject Classifications (2000)** 15A21 · 15A57 · 15A23 · 05B20 · 65F40 · 15A75

---

C. Kravvaritis · M. Mitrouli (✉)  
Department of Mathematics, University of Athens, Panepistimiopolis 15784, Athens, Greece  
e-mail: mmitroul@math.uoa.gr

C. Kravvaritis  
e-mail: ckrav@math.uoa.gr

### 1 Introduction

Let  $A$  be a matrix in  $\mathbb{C}^{m \times n}$ . For nonempty subsets  $\alpha$  of  $\{1, \dots, m\}$  and  $\beta$  of  $\{1, \dots, n\}$  we denote by  $A(\alpha|\beta)$  the submatrix of  $A$  whose rows are indexed by  $\alpha$  and whose columns are indexed by  $\beta$  in their natural order (i.e. ordered lexicographically).

Let  $k$  be a positive integer,  $k \leq \min\{m, n\}$ . We denote by  $C_k(A)$  the  $k$ th compound of the matrix  $A$ , that is, the  $\binom{m}{k} \times \binom{n}{k}$  matrix whose elements are the minors  $\det A(\alpha|\beta)$ , for all possible  $\alpha, \beta \subseteq \{1, \dots, n\}$ , with cardinality  $|\alpha| = |\beta| = k$ . We index  $C_k(A)$  by  $\alpha \subseteq \{1, \dots, n\}$ ,  $|\alpha| = k$  (ordered lexicographically).

For example, if  $A \in \mathbb{R}^{3 \times 3}$ , then

$$C_2(A) = \begin{bmatrix} \det A(\{1, 2\}|\{1, 2\}) & \det A(\{1, 2\}|\{1, 3\}) & \det A(\{1, 2\}|\{2, 3\}) \\ \det A(\{1, 3\}|\{1, 2\}) & \det A(\{1, 3\}|\{1, 3\}) & \det A(\{1, 3\}|\{2, 3\}) \\ \det A(\{2, 3\}|\{1, 2\}) & \det A(\{2, 3\}|\{1, 3\}) & \det A(\{2, 3\}|\{2, 3\}) \end{bmatrix}$$

and if  $A \in \mathbb{R}^{4 \times 4}$ , then  $C_3(A) =$

$$\begin{bmatrix} \det A(\{1, 2, 3\}|\{1, 2, 3\}) & \det A(\{1, 2, 3\}|\{1, 2, 4\}) & \det A(\{1, 2, 3\}|\{1, 3, 4\}) & \det A(\{1, 2, 3\}|\{2, 3, 4\}) \\ \det A(\{1, 2, 4\}|\{1, 2, 3\}) & \det A(\{1, 2, 4\}|\{1, 2, 4\}) & \det A(\{1, 2, 4\}|\{1, 3, 4\}) & \det A(\{1, 2, 4\}|\{2, 3, 4\}) \\ \det A(\{1, 3, 4\}|\{1, 2, 3\}) & \det A(\{1, 3, 4\}|\{1, 2, 4\}) & \det A(\{1, 3, 4\}|\{1, 3, 4\}) & \det A(\{1, 3, 4\}|\{2, 3, 4\}) \\ \det A(\{2, 3, 4\}|\{1, 2, 3\}) & \det A(\{2, 3, 4\}|\{1, 2, 4\}) & \det A(\{2, 3, 4\}|\{1, 3, 4\}) & \det A(\{2, 3, 4\}|\{2, 3, 4\}) \end{bmatrix}.$$

More facts about compound matrices, which are sometimes called the  $p$ th exterior powers of  $A$ , can be found in [1, 7, 14].

In this paper we consider computations of compounds of specific real matrices, which can be carried out efficiently due to the special form or structure of these matrices. Generally, the evaluation of compound matrices is a noneffective process due to the demanding determinant computations involved causing a severely high complexity. For example, the computation of all compounds of a random  $15 \times 15$  matrix with a standard implementation of the definition of compounds in Matlab cannot be executed within a realistic, acceptable period of time. So, there emerges the need to study the possibility of fast evaluation of compounds for specific matrices by taking advantage of their properties.

The usefulness of such a research is justified by the fact that computations of compounds arise in several applications in many fields. Some of the most important applications, where the evaluation of the compounds of a real matrix is encountered, are:

1. *Computation of exterior products*

For  $u_i \in \mathbb{R}^n$ ,  $i = 1, \dots, k$ , the exterior product  $u_1 \wedge \dots \wedge u_k$  can be expressed in terms of the  $k$ th compound of the matrix  $U = [u_1|u_2|\dots|u_k]$ , specifically

$$\bigwedge_{i \in \{1, \dots, k\}} u_i = C_k(U).$$

More properties and relative results can be found in [5, 21].

2. *Determinant of the sum of two matrices*

For the  $n \times n$  matrices  $A, B$ , the later presented Binet-Cauchy Theorem provides the separation

$$\det(A + B) = \det \left( [A \ I_n] \begin{bmatrix} I_n \\ B \end{bmatrix} \right) = C_n([A \ I_n]) C_n \left( \begin{bmatrix} I_n \\ B \end{bmatrix} \right),$$

since the determinant of an  $n \times n$  matrix  $A$  is identical to its  $n$ th compound matrix  $C_n(A)$ . Furthermore, in [20] the above scalar product is expressed in terms of the compounds of  $A$  and  $B$ , i.e. both matrices are separated in the sense that the resulting expression consists of a sum of traces of products of their compound matrices. A special case of the main result of this paper is the formula

$$\det(X + I_n) = 1 + \det(X) + \sum_{i=1}^{n-1} \text{tr} C_i(X)$$

for  $X \in \mathbb{C}^{n \times n}$ .

3. *Evaluation of the Smith Normal Form*

In [11] a numerical algorithm for evaluating the Smith Normal Form of integral matrices via compound matrices was proposed. This technique can work theoretically for any given integral matrix. However it is not recommended in practice due to its high complexity. One reasonable compound matrix implementation for the same purpose but for polynomial matrices is the parallel method proposed in [8], which can be very effective in practice.

4. *Computation of decentralization characteristics*

In [10] the Decentralized Determinantal Assignment Problem and the Decentralization Characteristic are defined. Compound matrices are used in relevant evaluation procedures.

5. *The Determinantal Assignment Problem*

All Linear Control Theory problems of frequency assignment (pole, zero) can be reduced to a standard common problem known as the Determinantal Assignment Problem [9, 16]. This unified approach, which considers that determinantal problems are of multilinear nature, involves evaluations of compound matrices.

6. *Computation of an uncorrupted base*

A convenient technique for selecting a best uncorrupted base for the row space of a matrix can be developed in terms of compound matrices, cf. [15, 17]. By the term “uncorrupted” we mean that we want to find a base for a given set of vectors without transforming the original data and evidently introducing roundoff-error even before the method starts. The term “best” reflects the property that the specific constructive process yields mostly orthogonal vectors. This approach is useful in several computational problems arising from Control Theory and requiring the selection of linearly independent sets of vectors without transforming the initial data.

Other applications of compounds of real matrices, several properties and computational aspects are discussed in [3].

The compounds of polynomial matrices are also useful in applications, since they are used e.g. for computing the Smith Normal Form of a polynomial matrix, the Weierstrass Canonical Form of a regular matrix pencil and Plucker matrices.

For further information on existing literature and research topics about compound matrices, the reader can consult [4, 19–22] and the references therein.

This study demonstrates in Section 2 that instead of computing the Singular Value Decompositions (SVD's) of the compounds of a matrix, which is a noneffective process, one can compute the compounds of the SVD of the matrix, which constitutes a more efficient method. The possibility to adopt this idea for other known matrix factorizations is discussed. In Section 3 formulas for the  $n - 1$  compounds of Hadamard matrices are derived. Finally, Section 4 gives formulas for the compounds of specific diagonal matrices and describes how the compounds of every diagonal matrix can be obtained. The paper concludes with Section 5, which summarizes the work and highlights future possibilities.

An important preliminary result, which states some principal properties of compound matrices, is the following Theorem 1.

**Theorem 1** [1, 13, 18] *Let  $A \in \mathbb{C}^{n \times m}$ ,  $B \in \mathbb{C}^{m \times l}$ ,  $X \in \mathbb{C}^{n \times n}$  and  $1 \leq p \leq \min\{n, m, l\}$ .*

- (i) (Binet-Cauchy Theorem)  $C_p(AB) = C_p(A)C_p(B)$ .
- (ii) *If  $A$  is unitary, then  $C_p(A)$  is unitary.*
- (iii) *If  $X$  is diagonal, then  $C_p(X)$  is diagonal.*
- (iv) *If  $X$  is upper (lower) triangular, then  $C_p(X)$  is upper (lower) triangular.*
- (v) (Silvester-Franke Theorem)  $\det(C_p(X)) = \det(X)^{\binom{n-1}{p-1}}$ .
- (vi)  $C_p(A^T) = (C_p(A))^T$ .

## 2 Matrix factorizations of compound matrices

We are interested in finding a mathematical connection between the compounds of some famous matrix factorizations and the corresponding factorization of the compounds of the initial matrix. For this purpose we will demonstrate the main idea through the example of the SVD, because its diagonal form leads to better results. We will discuss the derivation of a similar result for other known triangular matrix factorizations.

The motivation of this study lies in the fact that the computation of the SVD of the compounds of a matrix  $A$  is numerically noneffective, since it requires operations of order greater than  $n^4$ . Therefore it would be sensible to investigate whether there is a more efficient manner to produce an equivalent result. A question towards this direction arising naturally is whether the initial problem is equivalent to computing the compounds of the SVD of  $A$ . The answer is affirmative, as it is shown in the next theorem, and the proposed

idea leads to an acceptable, satisfactory complexity. So, in this manner one calculates the various compounds of the SVD of  $A$ , which consists of a diagonal matrix with positive, nonincreasing entries, instead of computing expensive SVD's of the various compounds of  $A$ . The importance of the proposed method is justified by the need to calculate efficiently basic matrix decompositions of compound matrices in several applications arising in Control Theory. Besides the practical benefit of such a tool, the effort for finding an acceptable complexity to a numerically noneffective problem has its own intrinsic interest and beauty as well.

In the following, SVD stands for the Singular Value Decomposition of a matrix. QR denotes the factorization of a matrix, while  $Q \cdot R$  denotes the product of the matrices  $Q$  and  $R$ .

**Definition 1** [7, p.164] Two matrices  $A, B \in \mathbb{R}^{n \times n}$  are called equivalent (written  $B \sim A$ ) if  $B = PAQ$  for some nonsingular matrices  $P, Q \in \mathbb{R}^{n \times n}$ .

**Theorem 2** Let  $A \in \mathbb{R}^{n \times n}$  be a given matrix and  $p$  an integer,  $1 \leq p \leq n$ . Then

$$C_p(\Sigma) = Q\Sigma_1Q^T,$$

where  $\Sigma$  and  $\Sigma_1$  are the diagonal matrices produced by the SVD of  $A$  and  $C_p(A)$ , respectively, and  $Q$  is a permutation matrix.

*Proof* For simplicity we will omit in this proof the index  $p$  of  $C_p$ , so  $C(M)$  denotes the  $p$ -th compound matrix of a matrix  $M$ .

Let  $A = U\Sigma V^T$  be the SVD of  $A$ , where  $U, V \in \mathbb{R}^{n \times n}$  are orthogonal. From Binet-Cauchy Theorem we have that  $C(A) = C(U)C(\Sigma)C(V^T)$ . According to Theorem 1 (ii), since  $U, V$  are orthogonal, the matrices  $C(U), C(V^T)$  are also orthogonal and nonsingular, hence from Definition 1 we have that

$$C(A) \sim C(\Sigma). \tag{1}$$

Let  $C(A) = U_1\Sigma_1V_1^T$  be the SVD of the  $p$ -th compound matrix of  $A$ , where  $U_1, V_1 \in \mathbb{R}^{\binom{n}{p} \times \binom{n}{p}}$  are orthogonal. Then

$$C(A) \sim \Sigma_1, \tag{2}$$

since  $U_1, V_1$  are nonsingular. Equations (1) and (2) imply

$$C(\Sigma) \sim \Sigma_1. \tag{3}$$

Since the matrices  $C(\Sigma)$  and  $\Sigma_1$  are diagonal with nonnegative entries, there exists a permutation matrix  $Q$  arranging the diagonal entries of  $\Sigma_1$  in nonincreasing order so that  $C(\Sigma) = Q\Sigma_1Q^T$ , which is equivalent to the statement of the theorem.  $\square$

Theorem 2 suggests that instead of computing the SVD of the compounds of a matrix whenever needed, which is very expensive from a computational

point of view, one can find the compounds of the SVD. Hence, the following standard algorithm  $SVD(C_p(A))$  can be replaced with the more effective new algorithm  $C_p(SVD(A))$ .

---

**Algorithm**  $SVD(C_p(A))$

---

**Input:** any  $n \times n$  matrix  $A$ .

**Output:** The SVD's of the compound matrices of  $A$ .

FOR  $p = 1, \dots, n$

    evaluate  $C_p(A) = [c_{ij}]$ ,  $i, j = 1, 2, \dots, \binom{n}{p}$

    compute  $SVD(C_p(A))$

END

END {of Algorithm}

---

**Algorithm**  $C_p(SVD(A))$

---

**Input:** any  $n \times n$  matrix  $A$ .

**Output:** The compound matrices of the SVD of  $A$ , which according to Theorem 2 are equal (up to a permutation) to the SVD's of the compound matrices of  $A$ .

compute  $SVD(A)$

FOR  $p = 1, \dots, n$

    evaluate  $C_p(SVD(A)) = [c_{ij}]$ ,  $i, j = 1, 2, \dots, \binom{n}{p}$

END

END {of Algorithm}

---

The superiority of the proposed algorithm over the standard technique, which is easily justified intuitively, can be demonstrated formally as well, as is done in the sequel by considering the computational costs.

**Complexity**

1. For  $SVD(C_p(A))$ .

First, the  $p$  compound matrices of  $A$  should be computed,  $p = 1, \dots, n$ .

The construction of the  $p$ -th compound matrix of  $A$  requires, according to its nature,

$$\binom{n}{p}^2 \frac{p^3}{3}$$

flops. The internal determinantal computations required are based on using Gaussian Elimination with partial pivoting which is a stable numerical method. So, after the appropriate summation, the total cost for finding all the compounds of  $A$  is

$$\frac{1}{6}n^2(n+1) \frac{\Gamma(2n-1)}{\Gamma(n)^2}$$

flops, where  $\Gamma$  is the Gamma function defined by  $\Gamma(0) := 1$  and  $\Gamma(n+1) = n!$ .

Generally, the formation of the SVD of a matrix of order  $n$  requires  $\frac{4n^3}{3}$  flops. Since the  $p$ -th compound matrix is of order  $\binom{n}{p}$ , there will be required totally

$$\frac{4}{3} \sum_{p=1}^n \binom{n}{p}^3$$

flops, which sums to  $\frac{4}{3}n^3 {}_4F_3([1, 1 - n, 1 - n, 1 - n], [2, 2, 2], -1)$ .  ${}_pF_q(a, b, z)$  stands for the generalized hypergeometric function defined as

$$\begin{aligned} {}_pF_q(a, b, z) &= {}_pF_q([a_1, \dots, a_p], [b_1, \dots, b_q], z) \\ &= \sum_{k=0}^{\infty} \frac{(a_1)_k (a_2)_k \dots (a_p)_k}{(b_1)_k (b_2)_k \dots (b_q)_k} \frac{z^k}{k!}, \end{aligned}$$

with  $(a)_n = a(a + 1)(a + 2) \dots (a + n - 1)$ . Hence, the total cost  $t_1$  for this case is

$$t_1 = \frac{1}{6}n^2(n + 1) \frac{\Gamma(2n - 1)}{\Gamma(n)^2} + \frac{4}{3}n^3 {}_4F_3([1, 1 - n, 1 - n, 1 - n], [2, 2, 2], -1)$$

flops.

2. For  $C_p(SVD(A))$ .

At the beginning, the SVD of  $A$  is formed with  $\frac{4n^3}{3}$  flops. Then, the  $p$  compounds of the SVD of  $A$  are computed. We consider the worst case where  $rank(A) = n$ , so the SVD has  $n$  singular values on the diagonal. So,  $\binom{n}{p}$  multiplications are needed for the computation of the  $p$ -th compound. The cost for finding all the compounds will be

$$\sum_{p=1}^n \binom{n}{p} = 2^n - 1.$$

Hence, the total cost  $t_2$  for this case is

$$t_2 = \frac{4n^3}{3} + 2^n - 1 \text{ flops.}$$

The computational cost  $t_2$  of the proposed method is significantly lower than the standard cost  $t_1$  for computing the SVD of the compounds of a matrix. The comparison between  $t_1$  and  $t_2$  is illustrated in Table 1 for various values of  $n$ .

*Remark 1* The superiority of the proposed approach over the standard strategy for computing the SVD of a compound matrix is also apparent if one considers the computation only of the  $p$ -th compound matrix for  $p$  fixed, and

**Table 1** Values of  $t_1$  and  $t_2$  for SVD

$n$	$t_1$	$t_2$
3	109	43
4	727	100
5	4751	198
8	1315020	938
10	$5.9801 \times 10^7$	2356
12	$2.9493 \times 10^9$	6399
15	$1.1482 \times 10^{12}$	37267
20	$2.7832 \times 10^{16}$	$1.0592 \times 10^6$
25	$7.3083 \times 10^{20}$	$3.3575 \times 10^7$
30	$1.9999 \times 10^{25}$	$1.0738 \times 10^9$

not of all compound matrices up to  $p = n$  as it was done before. In this case the flop counts are given by

$$\binom{n}{p}^2 \frac{p^3}{3} + \frac{4}{3} \binom{n}{p}^3 \text{ and } \frac{4n^3}{3} + \binom{n}{p}$$

for the standard and proposed technique, respectively.

The result of Theorem 2 can be verified in the next example. For the sake of brevity we omit the trivially calculated first and last compounds, which are the matrix itself and its determinant, respectively.

*Example 1* Let

$$A = \begin{bmatrix} 7 & 3 & 9 & 5 \\ 7 & 6 & 5 & 9 \\ 9 & 7 & 6 & 3 \\ 8 & 1 & 2 & 6 \end{bmatrix}.$$

1. The standard method computes first the compounds of  $A$  and then the SVD of each one of them.

$$C_2(A) = \begin{bmatrix} 21 & -28 & 28 & -39 & -3 & 56 \\ 22 & -39 & -24 & -45 & -26 & -3 \\ -17 & -58 & 2 & -3 & 13 & 44 \\ -5 & -3 & -60 & 1 & -45 & -39 \\ -41 & -26 & -30 & 7 & 27 & 12 \\ -47 & -30 & 30 & 8 & 39 & 30 \end{bmatrix}$$

$$\Sigma_2 = \begin{bmatrix} 132.0004 & 0 & 0 & 0 & 0 & 0 \\ 0 & 97.1018 & 0 & 0 & 0 & 0 \\ 0 & 0 & 87.3061 & 0 & 0 & 0 \\ 0 & 0 & 0 & 22.2894 & 0 & 0 \\ 0 & 0 & 0 & 0 & 20.0408 & 0 \\ 0 & 0 & 0 & 0 & 0 & 14.7424 \end{bmatrix}$$



$$C_3(A) = \begin{bmatrix} -29 & -160 & 252 & 293 \\ -242 & 74 & 224 & -172 \\ -277 & -52 & -210 & -221 \\ 1 & -330 & -210 & 57 \end{bmatrix}$$

$$\Sigma_3 = \begin{bmatrix} 534.5032 & 0 & 0 & 0 \\ 0 & 480.5822 & 0 & 0 \\ 0 & 0 & 353.5248 & 0 \\ 0 & 0 & 0 & 81.1504 \end{bmatrix}$$

2. According to the proposed method, the SVD of  $A$  is produced first and then the compound matrices of it.

$$\Sigma = \begin{bmatrix} 23.9802 & 0 & 0 & 0 \\ 0 & 5.5046 & 0 & 0 \\ 0 & 0 & 4.0493 & 0 \\ 0 & 0 & 0 & 3.6408 \end{bmatrix}$$

$$C_2(\Sigma) = \begin{bmatrix} 132.0004 & 0 & 0 & 0 & 0 & 0 \\ 0 & 97.1018 & 0 & 0 & 0 & 0 \\ 0 & 0 & 87.3061 & 0 & 0 & 0 \\ 0 & 0 & 0 & 22.2894 & 0 & 0 \\ 0 & 0 & 0 & 0 & 20.0408 & 0 \\ 0 & 0 & 0 & 0 & 0 & 14.7424 \end{bmatrix}$$

$$C_3(\Sigma) = \begin{bmatrix} 534.5032 & 0 & 0 & 0 \\ 0 & 480.5822 & 0 & 0 \\ 0 & 0 & 353.5248 & 0 \\ 0 & 0 & 0 & 81.1504 \end{bmatrix}$$

The above described idea, which works effectively for computing the SVD of the compound matrices of a matrix  $A$ , can be also adopted for finding other decompositions of compounds of  $A$  as well, such as QR, LU and Schur Triangularization. We will demonstrate the idea only through the comprehensive example of the QR factorization, since it can be done absolutely similarly for the other decompositions.

**Theorem 3** Let  $A \in \mathbb{R}^{n \times n}$  be a given invertible matrix and  $p$  an integer,  $1 \leq p \leq n$ . Then

$$C_p(R) \sim R_1,$$

where  $R$  and  $R_1$  are the upper triangular matrices produced by the QR factorizations of  $A$  and  $C_p(A)$ , respectively. Furthermore

$$C_p(Q) \sim Q_1,$$

where  $Q$  and  $Q_1$  are the respective orthogonal matrices from the previous  $QR$  factorizations.

*Proof* For simplicity we will omit in this proof the index  $p$  of  $C_p$ , so  $C(M)$  denotes the  $p$ -th compound matrix of a matrix  $M$ .

Let  $A = Q \cdot R$  be the QR decomposition of  $A$ , where  $Q$  is orthogonal and  $R$  upper triangular. From Binet-Cauchy Theorem we have that  $C(A) = C(Q) \cdot C(R)$ . Since  $Q$  is orthogonal, then according to Theorem 1  $C(Q)$  is also orthogonal and nonsingular, hence from the definition of equivalence of matrices we have that

$$C(A) \sim C(R). \tag{4}$$

Let  $C(A) = Q_1 \cdot R_1$  be the QR decomposition of the  $p$ -th compound matrix of  $A$ . Then  $R_1 = Q_1^T \cdot C(A)$ , hence

$$R_1 \sim C(A), \tag{5}$$

since  $Q_1^T$  is nonsingular. Equations (4) and (5) imply

$$R_1 \sim C(R). \tag{6}$$

Actually,

$$R_1 = Q_1^T \cdot C(Q) \cdot C(R). \tag{7}$$

Moreover we have

$$\begin{aligned} Q_1 \cdot R_1 &= C(Q) \cdot C(R) \Rightarrow \\ Q_1 &= C(Q) \cdot C(R) \cdot R_1^{-1}. \end{aligned} \tag{8}$$

According to the Sylvester-Franke Theorem, since  $A$  is invertible, every compound matrix of  $A$  is invertible, too, which guarantees the invertibility of  $R_1$ .

In order to prove the equivalence of  $Q_1$  and  $C(Q)$ , according to relation (8) it is sufficient to show that  $\det(C(R) \cdot R_1^{-1}) \neq 0$ .

It holds

$$C(Q)^{-1} \cdot Q_1 = C(R) \cdot R_1^{-1} \Rightarrow \det(C(R) \cdot R_1^{-1}) = \det(C(Q)^{-1}) \cdot \det Q_1.$$

$Q_1$  is orthogonal, so  $\det Q_1 = \pm 1$ . Furthermore, since  $C(Q)$  is orthogonal,  $C(Q)^{-1}$  is also orthogonal, hence  $\det(C(Q)^{-1}) = \pm 1$ . Finally,  $\det(C(R) \cdot R_1^{-1}) = \pm 1$  and  $Q_1 \sim C(Q)$ , which completes the proof of the statement of the Theorem. □

*Remark 2* The application of the proposed method to decompositions, like QR, LU and Schur Triangularization, faces two drawbacks. First, there is again a complexity improvement since we compute the compounds of sparse matrices instead of finding the decompositions of the compounds. But in this case the sparse matrices are upper triangular, and not diagonal like in the SVD, so the complexity benefit is only a lowering of the cost, not reduction of the order of the complexity, which was achieved with the SVD. And second, it is possible to derive only an equivalence relation between the two required forms

of matrices. In order to determine the exact manner of equivalence, one has to compute explicitly the respective orthogonal matrices and this doesn't lead to a computational improvement. On the contrary, for the case of the SVD we could prove the equality expression given in Theorem 2, which is facilitated by the fact that the diagonal entries of the SVD are all nonnegative and can be arranged in nonincreasing order. These two remarks constitute interesting open problems, which could contribute to finding effectively decompositions of the compound matrices of a given matrix.

Theorem 3, similarly to Theorem 2, suggests that instead of computing the QR of the compounds of a matrix, one can find the compounds of the QR and the following standard algorithm  $QR(C_p(A))$  can be replaced with the more effective new algorithm  $C_p(QR(A))$ .

---

**Algorithm**  $QR(C_p(A))$

---

**Input:** any  $n \times n$  matrix  $A$ .

**Output:** The QR's of the compound matrices of  $A$ .

FOR  $p = 1, \dots, n$

evaluate  $C_p(A) = [c_{ij}]$ ,  $i, j = 1, 2, \dots, \binom{n}{p}$

compute  $QR(C_p(A))$

END

END {of Algorithm}

---



---

**Algorithm**  $C_p(QR(A))$

---

**Input:** any  $n \times n$  matrix  $A$ .

**Output:** The compound matrices of the QR of  $A$ , which according to Theorem 3 are equal to the QR's of the compound matrices of  $A$ .

compute  $QR(A)$

FOR  $p = 1, \dots, n$

evaluate  $C_p(QR(A)) = [c_{ij}]$ ,  $i, j = 1, 2, \dots, \binom{n}{p}$

END

END {of Algorithm}

---

**Complexity**

1. For  $QR(C_p(A))$ .

First, the  $p$  compound matrices of  $A$  should be computed,  $p = 1, \dots, n$ . As it is described already, the total cost for finding all the compounds of  $A$  is

$$\frac{1}{6}n^2(n+1)\frac{\Gamma(2n-1)}{\Gamma(n)^2}$$

flops, where  $\Gamma$  is the Gamma function defined by  $\Gamma(0) := 1$  and  $\Gamma(n+1) = n!$ .

Generally, the computation of the QR of a matrix of order  $n$  requires  $\frac{2n^3}{3}$  flops. Since the  $p$ -th compound matrix is of order  $\binom{n}{p}$ , there will be required totally

$$\frac{2}{3} \sum_{p=1}^n \binom{n}{p}^3$$

flops, which sums to  $\frac{2}{3}n^3 {}_4F_3([1, 1-n, 1-n, 1-n], [2, 2, 2], -1)$  and  ${}_pF_q(a, b, z)$  stands for the generalized hypergeometric function as given above. Hence, the total cost  $t_1$  for this case is

$$t_1 = \frac{1}{6}n^2(n+1) \frac{\Gamma(2n-1)}{\Gamma(n)^2} + \frac{2}{3}n^3 {}_4F_3([1, 1-n, 1-n, 1-n], [2, 2, 2], -1)$$

flops.

2. For  $C_p(QR(A))$ .

At the beginning, the QR of  $A$  is formed with  $\frac{2n^3}{3}$  flops. Then, the  $p$  compounds of the QR of  $A$  are computed. The construction of the  $p$ -th compound matrix of  $R$  requires, due to the triangular form of  $R$ ,

$$\sum_{i=1}^{\binom{n}{p}} \sum_{j=i}^{\binom{n}{p}} \frac{p^3}{3}$$

flops, where the internal determinantal computations are carried out again with Gaussian Elimination with partial pivoting. So, after the appropriate summation, the total cost  $t_2$  for finding all the compounds of  $A$  is

$$\frac{1}{12}n^2(n+1) \frac{\Gamma(2n-1)}{\Gamma(n)^2} + \frac{1}{12}2^n n + \frac{1}{8}2^n n(n-1) + \frac{1}{48}2^n n(n-1)(n-2)$$

flops.

The computational cost  $t_2$  of the proposed method is always less than the standard cost  $t_1$  for computing the QR of the compounds of a matrix, but unfortunately it does not express a reduction of the order of the complexity. The comparison between  $t_1$  and  $t_2$  concerning the QR decomposition is illustrated in Table 2 for various values of  $n$ .

**Table 2** Values of  $t_1$  and  $t_2$  for QR

$n$	$t_1$	$t_2$
3	73	45
4	497	213
5	3251	1092
8	822246	168832
10	$3.4357 \times 10^7$	$4.4852 \times 10^6$
12	$1.5847 \times 10^9$	110232864
15	$5.8615 \times 10^{11}$	$1.2038 \times 10^{10}$
20	$1.3941 \times 10^{16}$	$2.4742 \times 10^{13}$
25	$3.6546 \times 10^{20}$	$4.3669 \times 10^{16}$
30	$9.9995 \times 10^{24}$	$6.9906 \times 10^{19}$

The result of Theorem 3 is discussed in the following example.

*Example 2* Let

$$A = \begin{bmatrix} 1 & 4 & 2 & 0 \\ 1 & 3 & 4 & 1 \\ 1 & 6 & 5 & 2 \\ 4 & 3 & 2 & 1 \end{bmatrix}.$$

1. The standard method computes first the compounds of  $A$  and then the QR of each one of them.

$$C_2(A) = \begin{bmatrix} -1 & 2 & 1 & 10 & 4 & 2 \\ 2 & 3 & 2 & 8 & 8 & 4 \\ -13 & -6 & 1 & 2 & 4 & 2 \\ 3 & 1 & 1 & -9 & 0 & 3 \\ -9 & -14 & -3 & -6 & 0 & 2 \\ -21 & -18 & -7 & -3 & 0 & 1 \end{bmatrix}$$

$$R_2 = \begin{bmatrix} 26.5518 & 22.1830 & 6.2896 & 2.6364 & -1.5065 & -1.8831 \\ 0 & -8.8269 & -3.5662 & -11.6143 & -4.6923 & -0.3142 \\ 0 & 0 & -3.5670 & 0 & -4.6934 & -3.5670 \\ 0 & 0 & 0 & 12.3352 & 4.9836 & 1.2416 \\ 0 & 0 & 0 & 0 & -4.9848 & -4.1646 \\ 0 & 0 & 0 & 0 & 0 & -1.6571 \end{bmatrix}$$

$$C_3(A) = \begin{bmatrix} -7 & -4 & 1 & 12 \\ 32 & 12 & 8 & 8 \\ 27 & 28 & 15 & 4 \\ -33 & 0 & 11 & 0 \end{bmatrix}$$

$$R_3 = \begin{bmatrix} 53.7680 & 21.7229 & 5.4121 & 5.2076 \\ 0 & 21.7282 & 18.1530 & 2.1574 \\ 0 & 0 & -7.2233 & -9.5043 \\ 0 & 0 & 0 & -10.0943 \end{bmatrix}$$

2. The proposed method produces first the QR of  $A$  and then the compound matrices of it.

$$R = \begin{bmatrix} -4.3589 & -5.7354 & -4.3589 & -1.6059 \\ 0 & -6.0914 & -5.0891 & -1.4429 \\ 0 & 0 & 2.0250 & 0.8181 \\ 0 & 0 & 0 & 0.8183 \end{bmatrix}$$

$$C_2(R) = \begin{bmatrix} 26.5518 & 22.1830 & 6.2896 & 2.6364 & -1.5065 & -1.8831 \\ 0 & -8.8269 & -3.5662 & -11.6143 & -4.6923 & -0.3142 \\ 0 & 0 & -3.5670 & 0 & -4.6934 & -3.5670 \\ 0 & 0 & 0 & -12.3352 & -4.9836 & -1.2416 \\ 0 & 0 & 0 & 0 & -4.9848 & -4.1646 \\ 0 & 0 & 0 & 0 & 0 & 1.6571 \end{bmatrix}$$

$$C_3(R) = \begin{bmatrix} 53.7680 & 21.7229 & 5.4121 & 5.2076 \\ 0 & 21.7282 & 18.1530 & 2.1574 \\ 0 & 0 & -7.2233 & -9.5043 \\ 0 & 0 & 0 & -10.0943 \end{bmatrix}$$

We notice that although matrices  $R_3$  and  $C_3(R)$  are the same,  $R_2$  and  $C_2(R)$  are only equivalent.

The essence of Theorem 3, which is also valid for the LU factorization, can be demonstrated by considering the upper triangular matrices produced by the LU decomposition of the previous  $A$ .

1. The standard method computes the LU factorizations of the previously computed compounds of  $A$ .

$$U_2 = \begin{bmatrix} -21.0000 & -18.0000 & -7.0000 & -3.0000 & 0 & 1.0000 \\ 0 & -6.2857 & 0 & -4.7143 & 0 & 1.5714 \\ 0 & 0 & 5.3333 & 0.0000 & 4.0000 & 2.6667 \\ 0 & 0 & 0 & -8.2500 & 0 & 2.7500 \\ 0 & 0 & 0 & 0 & 7.0000 & 6.0000 \\ 0 & 0 & 0 & 0 & 0 & 2.0952 \end{bmatrix}$$

$$U_3 = \begin{bmatrix} -33.0000 & 0 & 11.0000 & 0 \\ 0 & 28.0000 & 24.0000 & 4.0000 \\ 0 & 0 & 8.3810 & 6.2857 \\ 0 & 0 & 0 & 11.0000 \end{bmatrix}$$

2. The new method produces first the LU of  $A$  and then the compounds of it.

$$U = \begin{bmatrix} 4.0000 & 3.0000 & 2.0000 & 1.0000 \\ 0 & 5.2500 & 4.5000 & 1.7500 \\ 0 & 0 & 1.5714 & 0 \\ 0 & 0 & 0 & -1.3333 \end{bmatrix}$$

$$C_2(U) = \begin{bmatrix} 21.0000 & 18.0000 & 7.0000 & 3.0000 & 0 & -1.0000 \\ 0 & 6.2857 & 0 & 4.7143 & 0 & -1.5714 \\ 0 & 0 & -5.3333 & 0 & -4.0000 & -2.6667 \\ 0 & 0 & 0 & 8.2500 & 0 & -2.7500 \\ 0 & 0 & 0 & 0 & -7.0000 & -6.0000 \\ 0 & 0 & 0 & 0 & 0 & -2.0952 \end{bmatrix}$$

$$C_3(U) = \begin{bmatrix} 33.0000 & 0 & -11.0000 & 0 \\ 0 & -28.0000 & -24.0000 & -4.0000 \\ 0 & 0 & -8.3810 & -6.2857 \\ 0 & 0 & 0 & -11.0000 \end{bmatrix}$$

$U_2$  and  $U_3$  are equivalent with  $C_2(U)$  and  $C_3(U)$ , respectively.

### 3 Compounds of Hadamard matrices

In this section we focus on calculating  $n - 1$  compound matrices of Hadamard matrices. In general, it is always interesting to find a formula for calculating the compounds of a specially structured matrix, if possible, because the computation of compound matrices requires a high complexity cost and is totally non-effective. For example, the standard algorithm fails to be executed in a sensible period of time even for a Hadamard matrix of order 12. After calculating the compounds of Hadamard matrices of orders 4 and 8, we observed that there seems to be a connection between some compounds and the initial Hadamard matrix. We illustrate here this connection theoretically, which actually suggests a formula for calculating  $n - 1$  compounds of Hadamard matrices.

**Definition 2** A Hadamard matrix  $H$  of order  $n \times n$  is a matrix with elements  $\pm 1$  satisfying the orthogonality relation  $HH^T = H^T H = nI_n$ .

From this definition it follows obviously that every two distinct rows or columns of a Hadamard matrix are orthogonal, i.e. their inner product is zero. It can be proved [6] that if  $H$  is a Hadamard matrix of order  $n$  then  $n = 1, 2$  or  $n \equiv 0 \pmod{4}$ . However it is still an open conjecture whether Hadamard matrices exist for every  $n$  being a multiple of 4.

*Remark 3* We assume, without loss of generality, that the first entry of a row and a column of a Hadamard matrix is always  $+1$  (normalized form of a Hadamard matrix), because this can be always achieved by multiplying by  $-1$  and preserves the properties of the matrix.

#### 3.1 Preliminary results

**Lemma 1** [2, Chapter 2.13] Let  $B = \begin{bmatrix} B_1 & B_2 \\ B_3 & B_4 \end{bmatrix}$ ,  $B_1$  or  $B_4$  nonsingular. Then

$$\det B = \det B_1 \cdot \det (B_4 - B_3 B_1^{-1} B_2) \tag{9}$$

$$= \det B_4 \cdot \det (B_1 - B_2 B_4^{-1} B_3). \tag{10}$$

**Lemma 2** If  $B$  is the lower right  $(n - 1) \times (n - 1)$  submatrix of a normalized Hadamard matrix of order  $n$ , then the sum of the elements of every row and column of  $B^{-1}$  is equal to  $-1$ .

*Proof* Every row of  $B = [b_{ij}]$  contains  $n/2 - 1$  1's and  $n/2 - 1$ 's, due to the assumed property of normalized matrix. This observation follows easily from the fact that every row is orthogonal to the first one, which contains only 1's, hence each row except for the first contains exactly  $n/2$  1's and  $n/2 - 1$ 's. Since  $BB^{-1} = I_{n-1}$  must hold, the only possibility for  $B^{-1} = [x_{ij}]$  is:

$$x_{ij} = \begin{cases} -2/n, & \text{if } b_{ij} = -1 \\ 0, & \text{if } b_{ij} = 1 \end{cases} .$$

Since every row has  $n/2 - 1$ 's, the result follows obviously. The same argument can be used for the columns, too. □

### 3.2 Main result

**Theorem 4** *Let  $H = (h_{ij}), 1 \leq i, j \leq n$ , be a Hadamard matrix of order  $n$  and  $C = C_{n-1}(H) = (c_{ij}), 1 \leq i, j \leq n$ , the  $(n - 1)$ -th compound matrix of  $H$ . Then*

$$c_{ij} = \begin{cases} (-1)^{i+j}n^{\frac{n}{2}-1} & \text{for } h_{n-i+1,n-j+1} = 1 , \\ (-1)^{i+j+1}n^{\frac{n}{2}-1} & \text{for } h_{n-i+1,n-j+1} = -1 . \end{cases}$$

*Proof* In order to carry out the proof it is essential to understand that the calculation of the  $(i, j)$  entry  $c_{ij}, 1 \leq i, j \leq n$ , of the  $(n - 1)$ -th compound matrix  $C$  is done as follows, strictly according to the definition of compound matrices:  $c_{ij}$  is the determinant of the  $(n - 1) \times (n - 1)$  matrix occurring by deleting the  $(n - i + 1)$ -th row and the  $(n - j + 1)$ -th column of  $H$ . So, for calculating  $c_{ij}$ , we write  $H$  in the following form

$$H = \left[ \begin{array}{ccc|c|ccc} 1 & 1 & \dots & 1 & \dots & 1 \\ 1 & * & & * & & * \\ \vdots & & & & & \\ \hline 1 & * & & y & & * \\ \vdots & & & & & \\ \hline 1 & * & & & & * \end{array} \right] ,$$

where  $y \equiv h_{n-i+1,n-j+1} = \pm 1$  and the entries  $*$  denote any element  $\pm 1$  of  $H$ . For calculating the  $(n - 1) \times (n - 1)$  minor occurring by deleting the  $(n - i + 1)$ -th row and the  $(n - j + 1)$ -th column of  $H$  we perform the appropriate row and column interchanges that bring  $y$  in position  $(1, 1)$ . We obtain

$$H' = \left[ \begin{array}{c|cccc} y & * & * & \dots & * \\ * & & & & \\ \vdots & & & & \\ * & & & & \end{array} \right] .$$



The lower right  $(n - 1) \times (n - 1)$  minor of  $H'$  is  $c_{ij}$ , in which we are interested. Since we performed  $n - i$  row and  $n - j$  column interchanges, it holds

$$\det H' = (-1)^{2n-i-j} \det H = (-1)^{i+j} \det H,$$

and since  $H$  is a Hadamard matrix, we have  $\det H = n^{n/2}$  and

$$\det H' = (-1)^{i+j} n^{n/2}. \tag{11}$$

We discriminate two cases for the value of  $y$ .

*First case,  $y = 1$*

In this case the first row of  $H'$  contains  $n/2 - 1$  times the element 1 and  $n/2$  times the element  $-1$  among the entries between columns 2 and  $n$ . The same holds for the first column. By multiplying the  $n/2$  columns and rows starting with  $-1$  through with  $-1$  the determinant remains unaffected and we obtain

$$H'' = \left[ \begin{array}{c|cccc} 1 & 1 & 1 & \dots & 1 \\ \hline 1 & & & & \\ \vdots & & & B & \\ 1 & & & & \end{array} \right] = \begin{bmatrix} 1 & \mathbf{a}^T \\ \mathbf{a} & B \end{bmatrix},$$

where  $\mathbf{a}$  is the all 1's vector of order  $(n - 1) \times 1$ ,  $\det B = c_{ij}$  and  $\det H' = \det H''$ . The matrix  $B$  is always nonsingular because it is proved in [12] that the  $(n - 1) \times (n - 1)$  minors of a Hadamard matrix are of magnitude  $n^{\frac{n}{2}-1}$ , hence not equal to zero. From Lemma 1 we have

$$\begin{aligned} \det H'' &= \det B \cdot \det (1 - \mathbf{a}^T B^{-1} \mathbf{a}) \\ &= c_{ij} (1 - \mathbf{a}^T B^{-1} \mathbf{a}). \end{aligned} \tag{12}$$

From Lemma 2, since the sum of the elements of a column of  $B^{-1}$  is  $-1$ , after carrying out carefully the calculations we have

$$\mathbf{a}^T B^{-1} = -[1 \ 1 \ \dots \ 1] = -\mathbf{a}^T$$

and

$$\mathbf{a}^T B^{-1} \mathbf{a} = -\mathbf{a}^T \mathbf{a} = -(n - 1).$$

Equation (12) yields

$$\det H'' = nc_{ij}. \tag{13}$$

Equations (11) and (13), taking into consideration  $\det H' = \det H''$ , imply

$$nc_{ij} = (-1)^{i+j} n^{\frac{n}{2}} \Rightarrow c_{ij} = (-1)^{i+j} n^{\frac{n}{2}-1},$$

which verifies the formula of the enunciation for  $h_{n-i+1, n-j+1} = y = 1$ .

Second case,  $y = -1$

Working absolutely similarly to the first case leads to

$$c_{ij} = (-1)^{i+j} n^{\frac{n}{2}-1} (-1),$$

so the formula of the enunciation holds for all values of  $h_{n-i+1, n-j+1}$ , which are  $\pm 1$ . □

*Remark 4* The standard technique for evaluating the  $n - 1$  compound of a (Hadamard) matrix requires  $n^2$  evaluations of determinants of order  $n - 1$ . If we consider again Gaussian Elimination for evaluating a determinant requiring  $(n - 1)^3/3$  flops for each of these matrices, then it becomes obvious that we are led to a forbiddingly non effective algorithm. The proposed scheme requires only about  $n^2$  comparisons and negligible calculations, since every entry of the  $n - 1$  compound matrix is formed according to the respective entry in the original matrix taking into account the relation in the enunciation of Theorem 4.

The following corollary illustrates the equivalence relation between a Hadamard matrix and its  $(n - 1)$ -th compound, as it is actually described by the previous theorem.

**Corollary 1** *The  $(n - 1)$ -th compound matrix  $C$  of a Hadamard matrix  $H$  of order  $n$  satisfies*

$$PHQ = n^{1-\frac{n}{2}}C,$$

where

$$p_{ij} = \begin{cases} -1, & j = n - i + 1, i \text{ odd} \\ 1, & j = n - i + 1, i \text{ even} \\ 0, & j \neq n - i + 1 \end{cases}$$

and

$$q_{ij} = \begin{cases} 1, & j = n - i + 1, i \text{ odd} \\ -1, & j = n - i + 1, i \text{ even} \\ 0, & j \neq n - i + 1 \end{cases},$$

i.e.

$$P = \begin{bmatrix} 0 & 0 & \dots & 0 & -1 \\ 0 & 0 & \dots & 1 & 0 \\ \vdots & \vdots & & \vdots & \vdots \\ 0 & -1 & \dots & 0 & 0 \\ 1 & 0 & \dots & 0 & 0 \end{bmatrix} \quad \text{and} \quad Q = P^T = \begin{bmatrix} 0 & 0 & \dots & 0 & 1 \\ 0 & 0 & \dots & -1 & 0 \\ \vdots & \vdots & & \vdots & \vdots \\ 0 & 1 & \dots & 0 & 0 \\ -1 & 0 & \dots & 0 & 0 \end{bmatrix}.$$

*Proof*  $HH^T = nI_n$  implies  $H^{-1} = \frac{1}{n}H^T$ . Since  $H^{-1} = \frac{1}{\det H} \text{adj } H$  and  $\det H = n^{n/2}$  it follows that

$$H^T = n^{1-\frac{n}{2}} \text{adj } H, \tag{14}$$

where  $\text{adj } H$  denotes the adjugate (called also classical adjoint) of  $H$ .

Recalling the definition of the adjugate in terms of compound matrices [1, p.99] we have

$$\text{adj } H = QC_{n-1}(H^T)Q^{-1},$$

where  $Q$  is as given in the enunciation and  $Q^{-1} = P$ . Substitution in Eq. (14) yields

$$H^T = n^{1-\frac{n}{2}} QC_{n-1}(H^T)P.$$

The result follows by taking the transpose of both sides of the above equation and considering Theorem 1 (vi),  $P^T = Q$  and  $P^{-1} = Q$ .  $\square$

*Remark 5* The equivalence relation of Corollary 1 can be also derived straightforwardly by considering carefully the result of Theorem 4 and taking into account that the left multiplication with a unitary matrix performs the appropriate row interchanges (and multiplications by  $-1$ , if necessary), while the right multiplication has impact on the columns.

*Example 3* The usefulness of Theorem 4 can be easily seen by the computation of the 15-th compound of a Hadamard matrix  $H$  of order 16.

$$H = \begin{bmatrix} 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & -1 & 1 & -1 & 1 & -1 & 1 & -1 & 1 & -1 & 1 & -1 & 1 & -1 & 1 & -1 \\ 1 & 1 & -1 & -1 & 1 & 1 & -1 & -1 & 1 & 1 & -1 & -1 & 1 & 1 & -1 & -1 \\ 1 & -1 & -1 & 1 & 1 & -1 & -1 & 1 & 1 & -1 & -1 & 1 & 1 & -1 & -1 & 1 \\ 1 & 1 & 1 & 1 & -1 & -1 & -1 & -1 & 1 & 1 & 1 & 1 & -1 & -1 & -1 & -1 \\ 1 & -1 & 1 & -1 & -1 & 1 & -1 & 1 & 1 & -1 & 1 & -1 & -1 & 1 & -1 & 1 \\ 1 & 1 & -1 & -1 & -1 & -1 & 1 & 1 & 1 & 1 & -1 & -1 & -1 & -1 & 1 & 1 \\ 1 & -1 & -1 & 1 & -1 & 1 & 1 & -1 & 1 & -1 & -1 & 1 & -1 & 1 & 1 & -1 \\ 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 \\ 1 & -1 & 1 & -1 & 1 & -1 & 1 & -1 & -1 & 1 & -1 & 1 & -1 & 1 & -1 & 1 \\ 1 & 1 & -1 & -1 & 1 & 1 & -1 & -1 & -1 & -1 & 1 & 1 & -1 & -1 & 1 & 1 \\ 1 & -1 & -1 & 1 & 1 & -1 & -1 & 1 & -1 & 1 & 1 & -1 & -1 & 1 & 1 & -1 \\ 1 & 1 & 1 & 1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & 1 & 1 & 1 & 1 \\ 1 & -1 & 1 & -1 & -1 & 1 & -1 & 1 & -1 & 1 & -1 & 1 & 1 & -1 & 1 & -1 \\ 1 & 1 & -1 & -1 & -1 & -1 & 1 & 1 & -1 & -1 & 1 & 1 & 1 & 1 & -1 & -1 \\ 1 & -1 & -1 & 1 & -1 & 1 & 1 & -1 & -1 & 1 & 1 & -1 & 1 & -1 & -1 & 1 \end{bmatrix}$$

According to Theorem 4 no determinant computations are needed at all in order to find that

$$C_{15}(H) = 16^7 \begin{bmatrix} 1 & 1 & -1 & -1 & -1 & -1 & 1 & 1 & -1 & -1 & 1 & 1 & 1 & 1 & -1 & -1 \\ 1 & -1 & -1 & 1 & -1 & 1 & 1 & -1 & -1 & 1 & 1 & -1 & 1 & -1 & -1 & 1 \\ -1 & -1 & -1 & -1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & -1 & -1 & -1 & -1 \\ -1 & 1 & -1 & 1 & 1 & -1 & 1 & -1 & 1 & -1 & 1 & -1 & -1 & 1 & -1 & 1 \\ -1 & -1 & 1 & 1 & -1 & -1 & 1 & 1 & 1 & 1 & -1 & -1 & 1 & 1 & -1 & -1 \\ -1 & 1 & 1 & -1 & -1 & 1 & 1 & -1 & 1 & -1 & -1 & 1 & 1 & -1 & -1 & 1 \\ 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 \\ 1 & -1 & 1 & -1 & 1 & -1 & 1 & -1 & -1 & 1 & -1 & 1 & -1 & 1 & -1 & 1 \\ -1 & -1 & 1 & 1 & 1 & 1 & -1 & -1 & -1 & -1 & 1 & 1 & 1 & 1 & -1 & -1 \\ -1 & 1 & 1 & -1 & 1 & -1 & -1 & 1 & -1 & 1 & 1 & -1 & 1 & -1 & -1 & 1 \\ 1 & 1 & 1 & 1 & -1 & -1 & -1 & -1 & 1 & 1 & 1 & 1 & -1 & -1 & -1 & -1 \\ 1 & -1 & 1 & -1 & -1 & 1 & -1 & 1 & 1 & -1 & 1 & -1 & -1 & 1 & -1 & 1 \\ 1 & 1 & -1 & -1 & 1 & 1 & -1 & -1 & 1 & 1 & -1 & -1 & 1 & 1 & -1 & -1 \\ 1 & -1 & -1 & 1 & 1 & -1 & -1 & 1 & 1 & -1 & -1 & 1 & 1 & -1 & -1 & 1 \\ -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 \\ -1 & 1 & -1 & 1 & -1 & 1 & -1 & 1 & -1 & 1 & -1 & 1 & -1 & 1 & -1 & 1 \end{bmatrix}.$$

### 4 Compounds of diagonal matrices

It is often useful to find the compound matrices of diagonal matrices with specific form. These matrices can arise, e.g. as the Smith Normal Form of a given matrix. First we give a generalization of a result that was proved in [18] only for  $k = 2, 3$ . In the following propositions the symbol  $a_{i_1} \dots a_{i_j}(q)$  denotes that the factor  $a_{i_1} \dots a_{i_j}$  appears  $q$  times.

**Proposition 1** *Let*

$$M = \begin{bmatrix} 1 & 0 & 0 & \dots & 0 & 0 \\ 0 & a_2 & 0 & \dots & 0 & 0 \\ 0 & 0 & a_2 & \dots & 0 & 0 \\ \vdots & & & \ddots & & \\ 0 & 0 & 0 & \dots & a_2 & 0 \\ 0 & 0 & 0 & \dots & 0 & a_3 \end{bmatrix} = \text{diag}\{1, a_2(n-2), a_3\} \in \mathbb{R}^{n \times n}.$$

Then  $C_k(M) = \text{diag}\left\{a_2^{k-1} \binom{n-2}{k-1}, a_2^k \binom{n-2}{k}, a_2^{k-2} a_3 \binom{n-2}{k-2}, a_2^{k-1} a_3 \binom{n-2}{k-1}\right\}.$

*Proof* The result is derived by considering the definition of the  $k$ -th compound of  $M$ , i.e. that  $C_k(M)$  is constructed by forming all possible  $k \times k$  minors of  $M$ . But since  $M$  is diagonal,  $C_k(M)$  is also diagonal according to Theorem 1 (iii), and it is sufficient to form all possible combinations of products of its diagonal elements that contain  $k$  entries.

If we consider all possible  $k$ -combinations of the first  $n - 1$  diagonal elements, with the first entry 1 always included in the counting, we see that there are  $\binom{n-2}{k-1}$  possibilities to choose  $k - 1$  entries  $a_2$  among all  $n - 2$  entries  $a_2$ . Since the entry 1 is always included, the factor  $a_2$  should appear  $k - 1$  times in the product. Hence, the first factor  $a_2^{k-1} \binom{n-2}{k-1}$  is derived.

Similarly, if we consider the possibilities for choosing  $k$  entries  $a_2$ , or  $k - 2$  entries  $a_2$  and the entries 1 and  $a_3$  are always included, or  $k - 1$  entries  $a_2$  and  $a_3$  is always included, then the other three factors of the enunciation can be derived, too. □

By using the same logic it is possible to derive more general results, like in the next proposition. The pre-multiplication with the factor 1 denotes that the first entry 1 was included in the specific counting.

**Proposition 2** *Let  $M = \text{diag}\{1, a_2(n_2), a_3(n_3), a_4\} \in \mathbb{R}^{n \times n}$ . Then*

$$\begin{aligned}
 C_k(M) = \text{diag} & \left\{ 1 \cdot a_2^{k-1} \binom{n_2}{k-1}, a_2^k \binom{n_2}{k}, 1 \cdot a_3^{k-1} \binom{n_3}{k-1}, a_3^k \binom{n_3}{k}, \right. \\
 & a_2^i a_3^{k-i} \binom{n_2}{i} \binom{n_3}{k-i}, 1 \cdot a_2^i a_3^{k-i-1} \binom{n_2}{i} \binom{n_3}{k-i-1}, \\
 & a_2^{k-1} a_4 \binom{n_2}{k-1}, a_3^{k-1} a_4 \binom{n_3}{k-1}, 1 \cdot a_2^{k-2} a_4 \binom{n_2}{k-2}, \\
 & 1 \cdot a_3^{k-2} a_4 \binom{n_3}{k-2}, a_2^i a_3^{k-i-1} a_4 \binom{n_2}{i} \binom{n_3}{k-i-1}, \\
 & \left. 1 \cdot a_2^i a_3^{k-i-2} \binom{n_2}{i} \binom{n_3}{k-i-2} \right\},
 \end{aligned}$$

with  $i = 1, \dots, k - 1$ .

In the formulation of Proposition 2  $\binom{a}{b}$  with  $a < b$  is replaced by zero. Proposition 2 is a special case of finding the compounds for

$$M = \text{diag}\{1, a_2(n_2), a_3(n_3), \dots, a_{n-1}(n_{n-1}), a_n\} \in R^{n \times n},$$

which was mentioned as an open problem in [18]. The technique used in the proof of Propositions 1 and 2 can be used to provide results for this more general form of the problem as well. However, as it can be verified from the enunciation of Proposition 2, it is better not to give here the most general result for the sake of brevity and better presentation.

*Example 4* Let  $A = \text{diag}\{1, 2, 2, 2, 3, 3, 5\} \in \mathbb{R}^{7 \times 7}$ . The result of Proposition 2 for  $k = 2, 5, 6$ , which can be easily verified with the standard algorithm for finding compounds, is:

$$C_2(A) = \text{diag}\{2(3), 3(2), 4(3), 5(1), 6(6), 9(1), 10(3), 15(2)\},$$

$$C_5(A) = \text{diag}\{24(2), 40(1), 36(3), 60(6), 72(1), 90(3), 120(2), 180(3)\},$$

$$C_6(A) = \text{diag}\{72(1), 120(2), 180(3), 360(1)\}.$$

## 5 Conclusion

We have shown that the evaluation of compounds of specific matrices, which is generally a non effective computational procedure, can be carried out in an efficient way if the respective properties are taken appropriately into account. Precisely, if the SVD of the compounds of a matrix is required, then one can compute equivalently with significantly less operations the compounds of the SVD instead. A similar idea can be applied to QR, LU and Schur Triangularization as well. Moreover, analytical formulas for the  $n - 1$  compounds of Hadamard matrices have been derived, which avoid any complicated computations. Finally, the form of the compounds of diagonal matrices is presented. The efficient evaluation of compounds of matrix factorizations, apart from SVD, is an issue under consideration. The possibility to exploit further the structure of Hadamard and other related orthogonal matrices in order to derive analytical formulas for their compounds is currently also an open problem.

**Acknowledgements** We thank the referee for the valuable comments and suggestions that contributed to an improvement of the presentation of this paper.

This research was financially supported by ΠΕΝΕΔ 03ΕΔ 740 of the Greek General Secretariat for Research and Technology.

## References

1. Aitken, A.C.: Determinants and Matrices. Oliver & Boyd, Edinburgh (1967)
2. Bernstein, D.S.: Matrix Mathematics: Theory, Facts, and Formulas with Application to Linear Systems Theory. Princeton University Press, Princeton (2005)
3. Boutin, D.L., Gleeson R.F., Williams, R.M.: Wedge Theory / Compound Matrices: Properties and Applications. Office of Naval Research, Arlington, Report number NAWCADPAX-96-220-TR. <http://handle.dtic.mil/100.2/ADA320264> (1996)
4. Elsner, L., Hershkowitz, D., Schneider, D.: Bounds on norms of compound matrices and on products of eigenvalues. Bull. Lond. Math. Soc. **32**, 15–24 (2000)
5. Fiedler, M.: Special Matrices and Their Applications in Numerical Mathematics. Martinus Nijhoff, Dordrecht (1986)
6. Hadamard, J.: Résolution d'une question relative aux déterminants. Bull. Sci. Math. **17**, 240–246 (1893)
7. Horn, R.A., Johnson, C.R.: Matrix Analysis. Cambridge University Press, Cambridge (1985)
8. Kaltofen, E., Krishnamoorthy, M.S., Saunders, B.D: Fast parallel computation of Hermite and Smith forms of polynomial matrices. SIAM J. Algebr. Discrete Methods **8**, 683–690 (1987)

9. Karcantias, N., Giannakopoulos, C.: Grassmann invariants, almost zeros and the determinantal zero pole assignment problems of linear systems. *Internat. J. Control* **40**, 673–698 (1984)
10. Karcantias, N., Laios, B., Giannakopoulos, C.: Decentralized determinantal assignment problem: fixed and almost fixed modes and zeros. *Internat. J. Control* **48**, 129–147 (1988)
11. Koukouvinos, C., Mitrouli, M., Seberry, J.: Numerical algorithms for the computation of the Smith normal form of integral matrices. *Congr. Numer.* **133**, 127–162 (1998)
12. Kravvaritis, C., Mitrouli, M.: An algorithm to find values of minors of Hadamard matrices. *Bull. Greek Math. Soc.* **54**, 221–238 (2007)
13. Marcus, M.: *Finite Dimensional Multilinear Algebra, Two Volumes*. Marcel Dekker, New York (1973–1975)
14. Marcus, M., Minc, H.: *A Survey of Matrix Theory and Matrix Inequalities*. Allyn and Bacon, Boston (1964)
15. Mitrouli, M., Karcantias, N.: Computation of the GCD of polynomials using Gaussian transformation and shifting. *Internat. J. Control* **58**, 211–228 (1993)
16. Mitrouli, M., Karcantias, N., Giannakopoulos, C.: The computational framework of the determinantal assignment problem. In: *Proceedings ECC '91 European Control Conference*, vol. 1, pp. 98–103. ECC, Grenoble (1991)
17. Mitrouli, M., Karcantias, N., Koukouvinos, C.: Numerical aspects for nongeneric computations in control problems and related applications. *Congr. Numer.* **126**, 5–19 (1997)
18. Mitrouli, M., Koukouvinos, C.: On the computation of the Smith normal form of compound matrices. *Numer. Algorithms* **16**, 95–105 (1997)
19. Nambiar, K.K., Sreevalsan, S.: Compound matrices and three celebrated theorems. *Math. Comput. Model.* **34**, 251–255 (2001)
20. Prells, U., Friswell, M.I., Garvey, S.D.: Use of geometric algebra: compound matrices and the determinant of the sum of two matrices. *Proc. R. Soc. Lond. A* **459**, 273–285 (2003)
21. Tsatsomeros, M., Maybee, J.S., Olesky, D.D., Driessche, P.V.D.: Nullspaces of matrices and their compounds. *Linear Multilinear Algebra* **34**, 291–300 (1993)
22. Zhang, F.: A majorization conjecture for Hadamard products and compound matrices. *Linear Multilinear Algebra* **33**, 301–303 (1993)