AIMS Mathematics

https://www.aimspress.com/journal/Math

AIMS Mathematics, 11(8): 26397–26408.
DOI: 10.3934/math.20261059
Received: 22 April 2026
Revised: 13 August 2026
Accepted: 18 August 2026
Published: 24 August 2026

Research article

Accurate computations with generalized Frank matrices and total positivity

Héctor Orera* and J. M. Peña

Departamento de Matemática Aplicada/IUMA, Facultad de Ciencias, Universidad de Zaragoza, Calle de Pedro Cerbuna, 12, Zaragoza, 50009, Spain

* **Correspondence:** Email: hectororera@unizar.es; Tel: +34976762023.

Abstract: Total positivity of a class of generalized Frank matrices is proved. A bidiagonal decomposition of these matrices is obtained with high relative accuracy (HRA). It can be applied to compute their singular values, eigenvalues, inverses, or solutions of linear systems with HRA. Numerical tests confirming the theoretical results are included. They show that the proposed methods outperform standard numerical algorithms.

Keywords: total positivity; bidiagonal decompositions; frank matrix; accurate computations
Mathematics Subject Classification: 65G50, 15A23, 65F05, 65F15

1. Introduction

Accuracy is a very desirable goal in computational mathematics and, in particular, in numerical linear algebra. However, high relative accuracy (HRA) for the computations with ill-conditioned matrices has been only achieved with a few classes of structured matrices. Let us recall that a solution $\hat{x}$ of a problem is calculated with HRA if we compute $\hat{x}$ such that $\frac{|\hat{x}-x|}{|x|} \leq C\varepsilon$, where ε is the machine precision and C is a small constant. For the purpose of obtaining HRA computations for ill-conditioned matrices, a previous task of finding an adequate parametrization of the matrix has been necessary. For instance, for diagonally dominant M-matrices, the parametrization has been provided by the off-diagonal entries and the row sums (see [5, 20]). For nonsingular totally positive (TP) matrices, which is another important class of structured matrices, the parametrization is given by their bidiagonal decomposition (see [15]), which will be commented on later. A matrix is TP when all its minors are nonnegative. TP matrices, also known as totally nonnegative matrices in the literature (cf. [7]), arise in many interesting applications of many different fields. Among these fields, we can mention approximation theory (cf. [21]), mechanics (cf. [9]), computer aided geometric design (cf. [10]), combinatorics (cf. [1, 3]), statistics (cf. [14]), finances (cf. [14]), biology (cf. [10]), differential equations (cf. [1, 10]), or computing (cf. [16]).

In [8], Frank introduced the following matrix $\hat{F} = (\hat{f}_{ij})_{1\le i,j\le n}$ as an example of an ill-conditioned matrix:

$$\hat{f}_{ij} = \begin{cases} n-j+2, & \text{if } i \le j, \\ n-j+1, & \text{if } i = j+1, \\ 0, & \text{otherwise.} \end{cases} \tag{1.1}$$

This matrix has been a seminal source of many interesting matrices studied in the literature that generalize it. As some recent references see, for instance, [12, 22, 23] and references in there. Let us mention that these generalizations of the Frank matrix include r-Frank and geometric r-Frank matrices, and their structural and spectral properties have been analyzed.

The ill-conditioning of the family of matrices with entries defined by (1.1) made them also a classical example for numerical experimentation. Interested in the reason behind this ill-conditioning, Varah [24] proposed the following generalization of the Frank matrices, $F = (f_{ij})_{1\le i,j\le n}$:

$$f_{ij} = \begin{cases} a + b_j^2, & \text{if } i \le j < n, \\ b_j^2, & \text{if } i = j+1, \\ a, & \text{if } j = n, \\ 0, & \text{otherwise.} \end{cases} \tag{1.2}$$

Moreover, in his work he showed how to compute their eigensystem accurately.

In [13], Higham studied the total positivity of the Frank matrix (1.1). In our work, we show that the generalization given by Varah (1.2) also defines a family of TP matrices. Moreover, we study a simple decomposition that can be used to achieve accurate computations for the solution of many usual linear algebra problems with this class. In order to assure the accurate computations, we rely on the sufficient condition described in [6]. An algorithm that only uses multiplications, divisions, sums of real numbers of the same sign, and subtractions of initial data achieves results with HRA. In particular, an algorithm that avoids all subtractions also satisfies this sufficient condition and assures HRA. In fact, we provide a bidiagonal decomposition of a generalized Frank matrix A with HRA. The bidiagonal decomposition $\mathcal{BD}(A)$ arises naturally in the process of Neville elimination (see [11]). This process is an elimination procedure alternative to Gaussian elimination, which, roughly speaking, makes zeros in a column by adding to each row an adequate multiple of the previous one. It is known (cf. [15, 19]) that if we can compute $\mathcal{BD}(A)$ with HRA for a nonsingular TP matrix A, then we can perform other linear algebra calculations with HRA, including the eigenvalues, singular values, inverses, or the solution of linear systems $Ax = b$ for b with alternating signs. Some examples of subclasses of nonsingular TP matrices for which HRA computations have been achieved can be seen in [4, 17, 18]. As we have already mentioned, in this paper we obtain the bidiagonal decomposition of a generalized Frank matrix with HRA, so that we can apply to them the algorithms presented in [15].

We now summarize the main original contributions of this paper relative to the existing literature. In [13], the total positivity of classical Frank matrices (1.1) was shown, and their bidiagonal decompositions with HRA were obtained. These decompositions can be used to implement the linear algebra algorithms of [15] with HRA. In [24], Varah presented the generalized Frank matrices (1.2) and obtained an accurate method to compute their eigensystem. However, in [24] there were no accurate methods to compute singular values, inverses, or linear systems of generalized Frank matrices. Here, we obtain the values of the parameters of generalized Frank matrices for which these matrices are nonsingular TP and, in these cases, we also obtain their bidiagonal decomposition with HRA, and this decomposition is used to implement the linear algebra algorithms of [15] with HRA. As a consequence, we can compute with HRA all eigenvalues, singular values, inverses, and some associated linear systems of these generalized Frank matrices.

The manuscript is organized as follows. In Section 2, we characterize TP generalized Frank matrices and nonsingular TP generalized Frank matrices. In this last case, we present the mentioned decomposition from the parameters of the generalized Frank matrix. Section 3 includes the corresponding algorithm and numerical tests showing the high accuracy of the presented HRA method. Finally, Section 4 summarizes the main results and conclusions of the paper.

2. Bidiagonal decomposition and total positivity

In this section, we denote $E_k(\alpha)$ as the $n \times n$ lower elementary bidiagonal matrix with ones on the main diagonal and whose only nonzero off-diagonal entry is at the position $(k, k-1)$ and whose value is α. The following result provides the bidiagonal decomposition of a generalized Frank matrix given by (1.2) and, when the matrix is nonsingular TP, this decomposition (and so, other linear algebra calculations) can be computed with HRA.

Theorem 1. *Given the n parameters $a, b_1, b_2, \ldots, b_{n-1}$ defining the generalized Frank matrix $F = (f_{ij})_{1\le i,j\le n}$, we have that F is nonsingular TP if and only if $a > 0$ and, in this case, we can compute the following bidiagonal factorization of F with HRA:*

$$F = E_2(m_{2,1}) \ldots E_{n-1}(m_{n-1,n-2}) E_n(m_{n,n-1}) D E_2^T(\tilde{m}_{1,2}) E_3^T(\tilde{m}_{1,3}) \ldots E_n^T(\tilde{m}_{1,n}), \tag{2.1}$$

with $m_{21} = \frac{b_1^2}{a+b_1^2}$, $m_{k,k-1} = \frac{b_{k-1}^2}{a+b_{k-1}^2} \cdot \frac{a+b_{k-2}^2}{a}$ *for* $k = 3, \ldots, n$, $\tilde{m}_{1,n} = \frac{a}{a+b_{n-1}^2}$ *and* $\tilde{m}_{1,k} = \frac{a+b_k^2}{a+b_{k-1}^2}$ *for* $k = 2, \ldots, n-1$, *and*

$$D = diag\left(a + b_1^2, \frac{a(a+b_2^2)}{a+b_1^2}, \ldots, \frac{a(a+b_{n-1}^2)}{a+b_{n-2}^2}, \frac{a^2}{a+b_{n-1}^2}\right). \tag{2.2}$$

Moreover, all its singular values, all its eigenvalues, F^{-1} and the solution of linear systems $Fx = b$ for b with alternating signs can be computed to HRA.

Proof. Let us first observe that if $a = 0$, then F is singular, and if $a < 0$, then F is not TP. Hence, we consider the case $a > 0$. Let us now check that (2.1) holds by diagonalizing F through elementary matrix operations. First, let us apply Gaussian elimination (G-E) to F so that we obtain the upper triangular matrix U:

$$F = F^{(1)} \to F^{(2)} \to \ldots F^{(n)} = U.$$

By the zero structure of F, on the k-th step we just need to multiply by $E_{k+1}(-m_{k+1,k})$. For the first step of G-E, we have that $m_{21} = \frac{b_1^2}{a+b_1^2}$ and that $f_{ij}^{(2)} = f_{ij}$ whenever $i \neq 2$. For $i = 2$, $f_{21}^{(2)} = 0$, $f_{2j}^{(2)} = (a+b_j^2)\frac{a}{a+b_1^2}$ for $j = 2, \ldots, n-1$ and $f_{2n}^{(2)} = \frac{a^2}{a+b_1^2}$.

In general, for the k-th step we have that $m_{k+1,k} = \frac{b_k^2}{a+b_k^2}\frac{a+b_{k-1}^2}{a}$, $f_{ij}^{(k+1)} = f_{ij}^{(k)}$ for $i \neq k+1$, $f_{k+1,j}^{(k+1)} = (a+b_j^2)\frac{a}{a+b_k^2}$ for $j = k+1, \ldots, n-1$, and $f_{k+1,n}^{(k+1)} = \frac{a^2}{a+b_k^2}$. After finishing G-E, we obtain the upper triangular matrix U whose i-th row coincides with the i-th row of $F^{(k)}$. Let us now diagonalize the matrix U by applying elementary column operations, building the sequence

$$U = U^{(1)} \to U^{(2)} \to \ldots U^{(n)} = D.$$

First, we compute the product $U^{(2)} = UE_n^T(-\tilde{m}_{1,n})$, with $\tilde{m}_{1,n} = \frac{a}{a+b_{n-1}^2}$. This product produces zeros on every off-diagonal entry of the n-th column of U. For the k-th step, with $k = 2, \ldots, n-1$, we perform the product $U^{(k+1)} = U^{(k)}E_{n-k+1}^T(-\tilde{m}_{1,n-k+1})$, where $\tilde{m}_{1,n-k+1} = \frac{a+b_{n-k+1}^2}{a+b_{n-k}^2}$. At the last step, the diagonal matrix that we obtain is D (2.2). Hence, if we reorder the elementary matrices used in this process, we deduce (2.1): the bidiagonal factorization $\mathcal{BD}(F)$ of F.

Finally, since $a > 0$, we have that all the multipliers $m_{ij}, \tilde{m}_{ij}$ are nonnegative and that F admits a factorization as the product of a diagonal matrix with positive diagonal entries and of elementary matrix operations $E_k(\alpha)$ with $\alpha \geq 0$. Since each matrix of the right-hand side of (2.1) is TP and the product of TP matrices is TP (cf. Theorem 3.1 of [2]), F is TP. Also, since $a > 0$, the computation of this decomposition is subtraction-free, and it can be obtained with HRA. So, this also holds for the remaining mentioned algebraic operations. □

The following consequence of the previous theorem characterizes TP generalized Frank matrices.

Corollary 2. *The generalized Frank matrix $F = (f_{ij})_{1\leq i,j\leq n}$ given by* (1.2) *is TP if and only if $a \geq 0$.*

Proof. If $a < 0$, the matrix F has negative entries, and so it cannot be TP. Let us now suppose that $a \geq 0$. If $a > 0$, then it is TP by Theorem 1. Finally, take into account that the determinant is a continuous function on the matrix entries. In fact, matrices with a given nonnegative minor form the closed set given by $f^{-1}[0, \infty)$, where f is the determinant corresponding to that minor, and so, TP matrices are given by the intersection of closed sets, and they form a closed set. If $a = 0$, let us observe that $F = \lim_{\varepsilon\to 0} F_\varepsilon$, where F_ε has the same parameters $b_1, \ldots, b_{n-1}$ as F but with $a = \varepsilon$ instead of $a = 0$. By Theorem 1, matrices F_ε are TP matrices, and so F is also a TP matrix. □

The next section provides the algorithm for computing the bidiagonal decomposition of a generalized Frank matrix with HRA and presents numerical experiments with linear algebra computations.

3. The algorithm and numerical experiments

In Algorithm 1, we present the pseudocode for the computation of the bidiagonal decomposition of the generalized Frank matrix.

Given the bidiagonal decomposition of a nonsingular TP matrix with enough accuracy, Koev [15] developed routines to compute its eigenvalues and singular values with HRA. Besides, Marco and Martínez presented in [19] an algorithm to compute the inverse of a nonsingular TP matrix with HRA. These routines are implemented and available in the library `TNTool` to be used in MATLAB. This library is available for download in `https://sites.google.com/sjsu.edu/plamenkoev/home/software/tntool`. If we know the bidiagonal decomposition of a nonsingular TP matrix accurately, then we can use the following functions that assure results with HRA: `TNEigenValues` to compute the eigenvalues, `TNSingularValues` to compute the singular values, and `TNInverseExpand` to obtain the inverse. The function `TNSolve` obtains the solution of linear systems of equations $Ax = b$ and guarantees HRA whenever the independent term b has an alternating sign pattern. The functions `TNInverseExpand` and `TNSolve` have a computational cost of $O(n^2)$ elementary operations, and the computational cost of both `TNEigenValues` and `TNSingularValues` is of $O(n^3)$, while Algorithm 1 has a computational cost of $O(n)$ elementary operations.

Algorithm 1 Computation of $\mathcal{BD}(F)$.

Require: $a > 0$ and $b_1, b_2, \ldots, b_{n-1}$.
Ensure: $B = \mathcal{BD}(F)$ bidiagonal decomposition of F

$B = zeros(n)$
$c = (b_1^2, b_2^2, \ldots, b_{n-1}^2)$
$B(1,1) = a + c(1)$
$B(2,1) = c(1)/(a + c(1))$
$B(1,n) = a/(a + c(n-1))$
for i=2:n-1 **do**
  $B(1,i) = (a + c(i))/(a + c(i-1))$;
  $B(i,i) = a * (a + c(i))/(a + c(i-1))$;
  $B(i+1,i) = c(i) * (a + c(i-1))/(a + c(i))/a$
end for
$B(n,n) = a^2/(a + c(n-1))$

3.1. Numerical experiments with the classical Frank matrix

In order to test the accuracy of this approach, we have considered the classical Frank matrix, with parameters $a = 1$ and $b_k = \sqrt{n-k}$ for $k = 1, \ldots, n-1$ for sizes $n = 6, 8, 10, \ldots, 30$. We have tested the accuracy of the functions of `TNTool` taking our decomposition of the Frank matrix as input, and we have compared the results with those obtained with the usual MATLAB routines that take the original matrix as input. In Table 1, we show the condition number of the classical Frank matrices to illustrate the reason behind the difference in accuracy of both approaches.

Table 1. Condition number of the classical Frank matrix F depending on their size.

n	$\kappa_\infty(F)$
6	6447
8	4.2577e+05
10	4.5002e+07
12	6.8569e+09
14	1.4175e+12
16	3.8124e+14
18	1.2928e+17
20	5.3939e+19
22	2.7148e+22
24	1.6218e+25
26	1.1343e+28
28	9.1824e+30
30	8.5169e+33

First, we have computed the eigenvalues of A, $\lambda_1 > \lambda_2 > \ldots > \lambda_n > 0$ both with `TNEigenValues` and with the usual MATLAB routine `eig`. We have also computed the singular values $\sigma_1 > \sigma_2 > \ldots > \sigma_n > 0$ with `TNSingularValues` and with the usual MATLAB routine `svd`. In Figure 1, we present the relative errors for the smallest eigenvalue and singular value, λ_n and σ_n, which have been estimated using as exact the eigenvalues and singular values obtained with the symbolic MATLAB toolbox. Let us notice that the broken line for the HRA method means that the estimated relative error for the computation of the singular value is zero for those values of n.

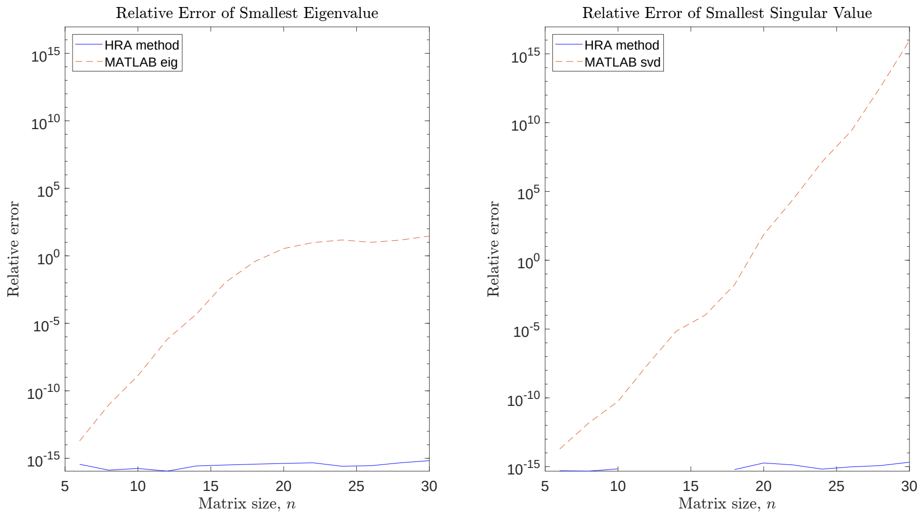

Figure 1. Relative errors for the smallest eigenvalue and singular value of the classical Frank matrix.

Then, we have computed the inverse of the classical Frank matrix F both with the HRA function `TNInverseExpand` as well as with the MATLAB routine `inv` for sizes $n = 6, 8, 10, \ldots, 30$. In order to check the accuracy of both methods, we have also computed the exact inverse using the symbolic routines in MATLAB. In Table 2, we present the maximum relative error and the mean relative error of the nonzero entries of the inverse obtained with both methods.

Table 2. Mean and maximum relative errors for the inverses of the classical Frank matrix.

n	maxHRA	meanHRA	max_inv	mean_inv
6	4.737e-16	1.0753e-16	3.1086e-15	6.9355e-16
8	4.0602e-16	7.9175e-17	7.5273e-13	3.7397e-13
10	8.0202e-16	1.497e-16	4.0051e-11	1.7716e-11
12	9.3326e-16	1.8936e-16	4.6562e-09	1.9958e-09
14	9.3326e-16	1.6663e-16	8.2073e-07	3.0937e-07
16	9.3326e-16	1.5801e-16	0.00015344	7.183e-05
18	9.3326e-16	1.505e-16	0.0029796	0.00047061
20	9.3326e-16	1.4642e-16	0.36093	0.15972
22	9.3326e-16	1.4161e-16	1.2831	0.56142
24	9.3326e-16	1.3916e-16	1.8809	0.56202
26	9.3326e-16	1.3209e-16	1.5154	0.55748
28	9.3326e-16	1.2757e-16	1.25	0.55077
30	9.3326e-16	1.2728e-16	2.5	0.5536

Finally, we have solved the linear systems of equations of the form $F_n x = y$, where F_n is the classical Frank matrix of size $n \times n$ and y is a vector whose entries are integers with an alternating sign pattern and whose absolute value was randomly chosen in the interval $[0, 1000]$. These vectors were generated with the MATLAB function `randi`. We have solved these systems both with the HRA routines that take the bidiagonal decomposition as input as well as with the usual MATLAB backslash routine. Table 3 shows the results of these tests.

Table 3. Mean and maximum relative errors for the solution of linear systems with the classical Frank matrix.

n	maxHRA	meanHRA	max_system	mean_system
6	4.8506e-16	3.0244e-16	1.0914e-15	8.7922e-16
8	1.6425e-16	9.2517e-17	6.937e-13	6.2801e-13
10	7.5812e-16	4.3324e-16	2.9181e-11	2.8408e-11
12	7.447e-16	4.5201e-16	3.3721e-09	3.2813e-09
14	5.3983e-16	1.918e-16	5.1313e-07	5.1081e-07
16	6.1592e-16	2.3856e-16	0.00012381	0.0001234
18	5.2955e-16	1.8862e-16	0.0012115	0.00082562
20	6.1005e-16	2.1623e-16	0.29223	0.28091
22	5.1752e-16	1.3908e-16	0.99646	0.99639
24	4.8042e-16	2.7537e-16	1.0001	1.0001
26	6.5801e-16	2.3116e-16	1	1
28	6.5516e-16	2.1162e-16	1	1
30	4.1614e-16	1.5271e-16	1	1

3.2. Numerical experiments with generalized Frank matrices

We have now considered the family of generalized Frank matrices for sizes $n = 6, 8, 10, \ldots, 30$, with parameters $a = 1/n^2$ and $b_k = 1/\sqrt{k}$ for $k = 1, \ldots, n-1$. We have tested again the accuracy of the functions of `TNTool` that take our decomposition of the generalized Frank matrix as input, and we have compared the results with those obtained with the usual MATLAB routines that take the original matrix as input. In Table 4, we include the condition number of these generalized Frank matrices.

We have started by computing the eigenvalues of A, $\lambda_1 > \lambda_2 > \ldots > \lambda_n > 0$ with `TNEigenValues` and with the usual MATLAB routine `eig`. We have also computed the singular values $\sigma_1 > \sigma_2 > \ldots > \sigma_n > 0$ with `TNSingularValues` and with the MATLAB routine `svd`. In Figure 2, we show the relative errors for the smallest eigenvalue and singular value, λ_n and σ_n. These relative errors have been estimated using as exact the eigenvalues and singular values obtained with the symbolic MATLAB toolbox.

Table 4. Condition number of the generalized Frank matrix F_n with parameters $a = 1/n^2$ and $b_k = 1/\sqrt{k}$ for $k = 1, \ldots, n-1$, depending on their size n.

n	$\kappa_\infty(F_n)$
6	9.1504e+07
8	3.0847e+11
10	1.6308e+15
12	1.2448e+19
14	1.2962e+23
16	1.7664e+27
18	3.0520e+31
20	6.5201e+35
22	1.6877e+40
24	5.2055e+44
26	1.8863e+49
28	7.9347e+53
30	3.8350e+58

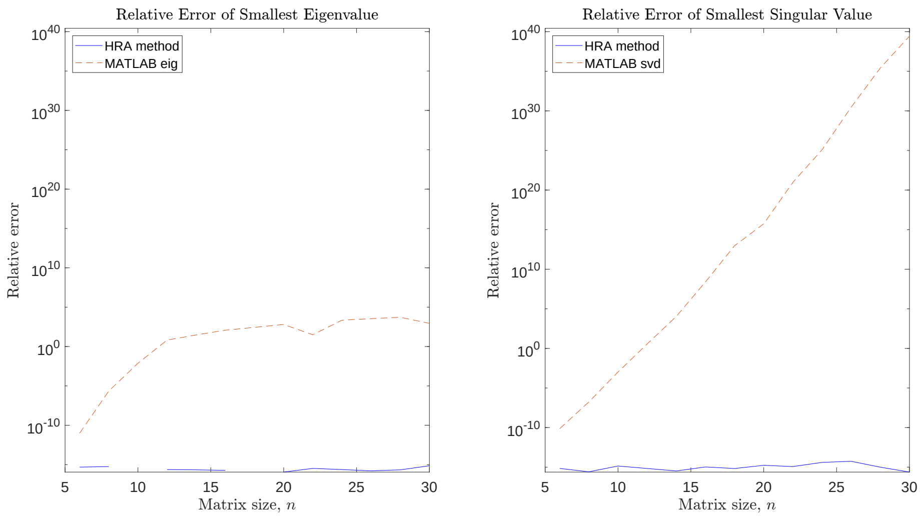

Figure 2. Relative errors for the smallest eigenvalue and singular value of the generalized Frank matrices with parameters $a = 1/n^2$ and $b_k = 1/\sqrt{k}$, $k = 1, \ldots, n-1$.

We have also computed the inverse of the generalized Frank matrix both with the HRA function `TNInverseExpand` as well as with the MATLAB routine `inv` for sizes $n = 6, 8, 10, \ldots, 30$. In order to check the accuracy of both methods, we have also computed the exact inverse using the symbolic routines in MATLAB. In Table 5, we present the maximum relative error and the mean relative error of the nonzero entries of the inverses obtained with both methods.

Table 5. Mean and maximum relative errors for the inverses of the generalized Frank matrices with parameters $a = 1/n^2$ and $b_k = 1/\sqrt{k}$, $k = 1, \ldots, n-1$.

n	maxHRA	meanHRA	max_inv	mean_inv
6	6.4779e-16	1.4901e-16	2.1101e-11	1.0308e-11
8	6.3631e-16	1.2957e-16	5.1318e-09	3.0166e-09
10	7.616e-16	1.4107e-16	8.8563e-05	5.2123e-05
12	6.7896e-16	1.0123e-16	0.45999	0.25607
14	1.7097e-15	3.0189e-16	1.2465	0.60661
16	6.6346e-16	1.2327e-16	1.0543	0.59124
18	1.4622e-15	2.5177e-16	1.1744	0.57731
20	8.073e-16	1.3089e-16	1.8936	0.57814
22	1.4265e-15	1.9709e-16	1.0718	0.56558
24	1.2764e-15	2.3765e-16	1.5261	0.56324
26	1.1528e-15	1.889e-16	1.0355	0.55686
28	2.8691e-15	4.1867e-16	1.0332	0.55282
30	9.6342e-16	1.2861e-16	1.0291	0.54697

Finally, we have solved linear systems of equations of the form $F_n x = y$, where F_n is the generalized Frank matrix of size $n \times n$ using the parameters $a = 1/n^2$ and $b_k = 1/\sqrt{k}$, for $k = 1, \ldots, n-1$ and y is a vector with integer entries. The absolute value of each entry was randomly chosen in the interval $[0, 1000]$, and the vector was constructed with an alternating sign pattern. These vectors were generated with the MATLAB function `randi`. We have solved these systems both with the HRA method that takes the bidiagonal decomposition as input as well as with the usual MATLAB backslash routine. In Table 6, we include the results of these experiments.

Table 6. Mean and maximum relative errors for the solution of linear systems with the generalized Frank matrices using parameters $a = 1/n^2$ and $b_k = 1/\sqrt{k}$, $k = 1, \ldots, n-1$.

n	maxHRA	meanHRA	max_system	mean_system
6	3.9531e-16	2.3418e-16	1.3837e-11	1.3614e-11
8	6.4681e-16	2.3063e-16	4.4793e-09	4.4773e-09
10	3.8235e-16	1.7527e-16	8.1038e-05	8.0987e-05
12	4.1067e-16	1.447e-16	0.41186	0.41158
14	1.4559e-15	4.6472e-16	1.0003	1.0003
16	4.6575e-16	1.609e-16	1	1
18	1.3318e-15	5.3394e-16	1	1
20	5.1427e-16	1.9933e-16	1	1
22	1.3213e-15	7.9158e-16	1	1
24	1.0419e-15	5.2067e-16	1	1
26	8.5311e-16	2.6443e-16	1	1
28	2.2364e-15	8.9192e-16	1	1
30	7.7925e-16	3.683e-16	1	1

4. Conclusions

Varah's generalization of Frank matrices is analyzed. TP generalized Frank matrices are characterized by means of the parameters a. A bidiagonal decomposition of nonsingular TP generalized Frank matrices is constructed with HRA, and the corresponding algorithm is presented. This algorithm is used in turn to derive HRA computations of all eigenvalues and singular values of these matrices, as well as of their inverses and of the solutions of some linear systems. Let us remark that our methods are restricted to generalized Frank matrices that are TP. Numerical experiments show that our methods outperform standard methods and achieve results with a relative error of the order of the unit roundoff of the double precision arithmetic. In future research, we shall consider HRA computations for other generalized Frank matrices.

Author contributions

All authors have contributed equally to this work. All authors have read and approved the final version of the manuscript for publication.

Use of Generative-AI tools declaration

The authors declare they have not used Artificial Intelligence (AI) tools in the creation of this article.

Acknowledgments

This research was partially supported through the Spanish research grant PID2022-138569NB-I00 (MCIU/AEI) and by Gobierno de Aragón (E41_23R).

Conflict of interest

Prof. Juan Manuel Peña is the Guest Editor of special issue "Advances in Computational Mathematics" for AIMS Mathematics. Prof. Juan Manuel Peña was not involved in the editorial review and the decision to publish this article.

The authors declare no conflicts of interest.

References

1. P. Alonso, J. Delgado, R. Gallego, J. M. Peña, Conditioning and accurate computations with Pascal matrices, *J. Comput. Appl. Math.*, **252** (2013), 21–26. https://doi.org/10.1016/j.cam.2011.12.007
2. T. Ando, Totally positive matrices, *Linear Algebra Appl.*, **90** (1987), 165–219. https://doi.org/10.1016/0024-3795(87)90313-2
3. J. Delgado, H. Orera, J. M. Peña, Accurate bidiagonal decomposition and computations with generalized Pascal matrices, *J. Comput. Appl. Math.*, **391** (2021), 113443. https://doi.org/10.1016/j.cam.2021.113443

4. J. Delgado, J. M. Peña, High-relative-accuracy computations with Kac-Murdock-Szegö matrices and their generalizations, *Comput. Appl. Math.*, **44** (2025), 188. https://doi.org/10.1007/s40314-025-03137-7

5. J. Demmel, P. Koev, Accurate SVDs of weakly diagonally dominant M-matrices, *Numer. Math.*, **98** (2004), 99–104. https://doi.org/10.1007/s00211-004-0527-8

6. J. Demmel, I. Dumitriu, O. Holtz, P. Koev, Accurate and efficient expression evaluation and linear algebra, *Acta Numer.*, **17** (2008), 87–145. https://doi.org/10.1017/s0962492906350015

7. S. M. Fallat, C. R. Johnson, *Totally nonnegative matrices*, Princeton and Oxford: Princeton University Press, 2011. https://doi.org/10.23943/princeton/9780691121574.001.0001

8. W. L. Frank, Computing eigenvalues of complex matrices by determinant evaluation and by methods of Danilewski and Wielandt, *J. Soc. Indust. Appl. Math.*, **6** (1958), 378–392. https://doi.org/10.1137/0106026

9. F. P. Gantmacher, M. G. Krein, *Oscillation matrices and kernels and small vibrations of mechanical systems: revised edition*, Providence: AMS Chelsea, 2002. https://doi.org/10.1090/chel/345

10. M. Gasca, C. A. Micchelli, *Total positivity and its applications*, Mathematics and its Applications, Vol. 359, Dordrecht: Kluwer Academic Publishers, 1996. https://doi.org/10.1007/978-94-015-8674-0

11. M. Gasca, J. M. Peña, Total positivity and Neville elimination, *Linear Algebra Appl.*, **165** (1992), 25–44. https://doi.org/10.1016/0024-3795(92)90226-z

12. H. Gökbaş, Some properties of the generalized max Frank matrices, *AIMS Math.*, **9** (2024), 26826–26835. https://doi.org/10.3934/math.20241305

13. N. J. Higham, The power of bidiagonal matrices, *Electron. J. Linear Algebra*, **40** (2024), 453–474.

14. S. Karlin, *Total positivity*, Vol. I, Stanford: Stanford University Press, 1968.

15. P. Koev, Accurate computations with totally nonnegative matrices, *SIAM J. Matrix Anal. Appl.*, **29** (2007), 731–751. https://doi.org/10.1137/04061903x

16. E. Mainar, J. M. Peña, B. Rubio, On the total positivity of q-Bernstein mass matrices and their accurate computations, *J. Comput. Appl. Math.*, **451** (2024), 116034. https://doi.org/10.1016/j.cam.2024.116034

17. A. Marco, J. J. Martínez, A fast and accurate algorithm for solving Bernstein-Vandermonde linear systems, *Linear Algebra Appl.*, **422** (2007), 616–628. https://doi.org/10.1016/j.laa.2006.11.020

18. A. Marco, J. J. Martínez, Accurate computations with Said-Ball-Vandermonde matrices, *Linear Algebra Appl.*, **432** (2010), 2894–2908. https://doi.org/10.1016/j.laa.2009.12.034

19. A. Marco, J. J. Martínez, Accurate computation of the Moore–Penrose inverse of strictly totally positive matrices, *J. Comput. Appl. Math.*, **350** (2019), 299–308. https://doi.org/10.1016/j.cam.2018.10.009

20. J. M. Peña, LDU decompositions with L and U well conditioned, *Electron. Trans. Numer. Anal.*, **18** (2004), 198–208.

21. A. Pinkus, *Totally positive matrices*, Cambridge Tracts in Mathematics, Cambridge: Cambridge University Press, 2010. https://doi.org/10.1017/cbo9780511691713

22. B. Shi, C. Kızılateş, A new generalization of the Frank matrix and its some properties, *Comput. Appl. Math.*, **43** (2024), 19. https://doi.org/10.1007/s40314-023-02524-2

23. B. Shi, C. Kızılateş, Geometric r-Frank matrix: some properties and applications, *Comput. Appl. Math.*, **44** (2025), 388. https://doi.org/10.1007/s40314-025-03345-1

24. J. M. Varah, A generalization of the Frank matrix, *SIAM J. Sci. Statist. Comput.*, **7** (1986), 835–839. https://doi.org/10.1137/0907056