International Journal of Mathematics and Computer Research
ISSN: 2320-7167
Volume 13 Issue 02 February 2025, Page no. – 4808-4817
Index Copernicus ICV: 57.55, Impact Factor: 8.615
DOI: 10.47191/ijmcr/v13i2.02

On the heptadiagonal matrix CL factorization

B. TALIBI[1], A.AIAT HADJ[2], D.SARSRI[1]

[1]Department of Industrial Engineering and Logistics, National School of Applied Sciences of TANGIER; Abdelmalek Essaâdi University, Morocco.

[2]Regional Center of the Trades of Education and Training (CRMEF)-Tangier, Avenue My Abdelaziz, Souani, BP: 3117, Tangier. Morocco.

1 Abstract

The main contribution of this paper is a new CL type decomposition of heptadiagonal matrices for fast inversion. We further provide two algorithms and study their execution times to measure the benefit of the proposed decomposition. Therefore, we are able to obtain a number of interesting results.

Keywords: *Heptadiagonal matrices, decomposition, CL factorization, matrix inverse.*

2 Introduction

Heptadiagonal matrices, which have seven nonzero diagonals, frequently arise in many scientific and engineering applications, including finite difference methods for partial differential equations, signal processing, andnumerical modeling of physical phenomena. In this regard, the efficient manipulation and inversion of such matrices is also key tocomputational efficiency in these domains.

Matrix factorization, is a wonderful tool in numerical linear algebra that provides knowledge about the latent structural properties of matrices and enables efficient computations for matrix inversion, determinant calculation, eigenvalue approximation, etc. While several factorization approaches exist, the CL factorization, which allows writing a matrix as the product of a companion-like and a lower triangular matrix, is especially interesting due tothe sparseness and special form of the matrices.

We will examine the CL factorization of heptadiagonal matrices in this study and identify ways in which the configuration can be simplified as much as possible in order to eliminate unnecessary invert operations. This approach takes advantage of the natural sparsity of heptadiagonal matrix in ATX-A, and offers a general method to further reduce the computational burden. This becomes especially useful in problems where thesize becomes large enough that traditional techniques become intractable or prohibitively time consuming.

In order to measure the actual usefulness of the proposed decomposition, we provide two different algorithms for performing the CL factorization, and we study their running times and costs.

By using comparative analysis, we can also showwhich algorithm performs the best and that our approach works well with heptadiagonal matrices.

The results from our experiments validate the effectiveness of our proposed CL factorization and show the potential for wide applications of the same in scientificcomputing. This work takes a step towards improving numerical methods for structured matrices and also provides a starting point for future workin this channel.

This work aims to explore the decomposition approach for heptadiagonal matricesin an unconventional manner. This method utilizes the CL factorization,the factorization is expressed such that the heptadiagonal matrix A can be written as $A = CL$, where:

$$A = \begin{pmatrix} d & c & b & a & 0 & 0 & \cdots & 0 \\ \gamma & d & c & b & a & 0 & \cdots & 0 \\ \beta & \gamma & d & c & b & a & \cdots & 0 \\ \alpha & \beta & \gamma & d & c & b & \cdots & 0 \\ 0 & \alpha & \beta & \gamma & d & c & \cdots & 0 \\ \vdots & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots & \vdots \\ 0 & \cdots & 0 & \alpha & \beta & \gamma & d & c \\ 0 & \cdots & 0 & 0 & \alpha & \beta & \gamma & d \end{pmatrix} = CL$$

Where C is a companion matrix and L is lower triangularmatrix. This gives us a new relationto define the inverse of the general heptadiagonal matrix A.

3 Factorization of Heptadiagonal matrix:

In this section, we present a simple way to factorize heptadiagonal matrix A and a systematic way to compute the matrices C andL, where the factorized form of T with the theoretical background will be introduced in the next theorem.

Theorem 1 . *Then every heptadiagonal matrix A canbe written as $A = CL$, where:*

$$C = \begin{pmatrix} 0 & 1 & 0 & \cdots & \cdots & 0 \\ \vdots & \ddots & \ddots & \ddots & & \vdots \\ \vdots & & \ddots & \ddots & \ddots & \vdots \\ \vdots & & & \ddots & 1 & 0 \\ 0 & \cdots & \cdots & \cdots & 0 & 1 \\ x_1 & \frac{x_2}{a} & \cdots & \cdots & \frac{x_{n-1}}{a} & \frac{x_n}{a} \end{pmatrix},$$

And

$$L = \begin{pmatrix} 1 & 0 & & & \cdots & & & 0 \\ d & c & b & a & 0 & \cdots & & \vdots \\ \gamma & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots & \\ \beta & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots & \vdots \\ \alpha & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots & 0 \\ 0 & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots & a \\ \vdots & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots & b \\ 0 & \cdots & 0 & \ddots & \ddots & \ddots & \ddots & c \\ 0 & \cdots & & 0 & \alpha & \beta & \gamma & d \end{pmatrix},$$

With:

$$x_n = d,\ x_r = (-1)^{n-r}(\prod_{k=r}^{n-1} a_k)\Delta_r.$$

And:

$$\Delta_r = \begin{pmatrix} d_r & c_r & b_r & a_r & 0 & \cdots & & \\ \gamma_r & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots & \\ \beta_r & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots & \vdots \\ \alpha_r & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots & 0 \\ 0 & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots & a_{n-3} \\ \vdots & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots & b_{n-2} \\ 0 & \cdots & 0 & \ddots & \ddots & \ddots & \ddots & c_{n-1} \\ 0 & \cdots & & 0 & \alpha_{n-3} & \beta_{n-2} & \gamma_{n-1} & d_n \end{pmatrix}$$

4 Inverse of Heptadiagonal matrix: Algorithm 1

Here we present a novel formula for inverting the generalizedheptadiagonal matrix in terms of the inverse of the efficient factorized matrix.

lemma: Let C a companion matrix, then $det(C) = (-1)^{n-1}x_1$, and:

$$C^{-1} = \begin{pmatrix} -\frac{x_2}{x_1 a} & -\frac{x_3}{x_1 a} & \cdots & \cdots & -\frac{x_{n-1}}{x_1 a} & -\frac{x_n}{x_1 a} \\ 1 & 0 & \cdots & & & 0 \\ 0 & \ddots & \ddots & \ddots & \ddots & \vdots \\ \vdots & & & \ddots & & \\ & \cdots & \cdots & \cdots & 0 & \\ 0 & & \cdots & 0 & 1 & 0 \end{pmatrix}$$

Proof: It is trivial to check that $C^{-1}C = I_n$ where I_n is the $n \times n$ identity matrix.

Theorem 2 *If A is a heptadiagonal matrix and $A = CL$, then $A^{-1} = L^{-1}C^{-1}$.*
With:

$$L^{-1} = \sum_{k=0}^{n-1} U_k J^k,\ U_k = -\frac{1}{a}(U_{k-1}b + U_{k-2}c + U_{k-3}d + U_{k-4}\gamma + U_{k-5}\beta + U_{k-6}\alpha)\ \ k \geq 6.$$

Where: $J = \begin{pmatrix} 0 & & & \cdots & 0 \\ 1 & \ddots & & & \vdots \\ 0 & \ddots & \ddots & & \vdots \\ \vdots & \ddots & \ddots & \ddots & \\ 0 & \cdots & 0 & 1 & 0 \end{pmatrix}$

5 Inverse of Heptadiagonal matrix: Algorithm 2

Next, we performa comparative study between our method and another existing method. This has been done to ensure the performance and efficacy state of our algorithm, providing further insight in terms of the strengths and improvements our technique has to offer.

Definition: The Hessenberg (or lower Hessenberg) matrices are the matrices $A = [a_{ij}]$ such that this means $a_{ij} = 0$ for $j - i > 1$. More generally, the matrix is $K - Hessenberg$ except when $A_{ij} = 0$ for $j - i > K$.

theorem [17] : We consider a strict $K - Hessenberg$ matrix and denote the block decomposition of A by:

$$\mathbf{A} = \begin{pmatrix} B & E \\ D & C \end{pmatrix}\ ,\ \ E \in \mathbf{M}_{n-k}(F)\ ,\ B \in \mathbf{M}_{(n-k)\times k}(F)\ ,\ C \in \mathbf{M}_{k\times(n-k)}(F)\ ,\ D \in \mathbf{M}_k.$$

A is invertible exactly when $CE^{-1}B - D$ is invertible and if A is invertible we have:

$$A^{-1} = \begin{pmatrix} 0 & 0 \\ E^{-1} & 0 \end{pmatrix} - \begin{pmatrix} I_k \\ -E^{-1}B \end{pmatrix} (CE^{-1}B - D)^{-1} \begin{pmatrix} -CE^{-1} & I_k \end{pmatrix}$$

Toeplitz-Hessenberg matricesare the subject of research by $RoksanaSlowik$ [16].

We apply ourwork to a Toeplitz heptadiagonal matrix A, and we then get:

$$\mathbf{A}^{-1} = \begin{pmatrix} 0 & 0 \\ E^{-1} & 0 \end{pmatrix} - \begin{pmatrix} I_3 \\ -E^{-1}B \end{pmatrix} (CE^{-1}B - D)^{-1} \begin{pmatrix} -CE^{-1} & I_3 \end{pmatrix} \tag{1}$$

The blocks B, C,D and E have the expressions:

$$B = \begin{pmatrix} d & c & b \\ \gamma & d & c \\ \beta & \gamma & d \\ \alpha & \beta & \gamma \\ 0 & \alpha & \beta \\ 0 & 0 & \alpha \\ 0 & 0 & 0 \\ \vdots & \vdots & \vdots \\ 0 & 0 & 0 \end{pmatrix}$$

$$C = \begin{pmatrix} 0 & \cdots & 0 & \alpha & \beta & \gamma & d & c & b \\ 0 & \cdots & 0 & 0 & \alpha & \beta & \gamma & d & c \\ 0 & \cdots & 0 & 0 & 0 & \alpha & \beta & \gamma & d \end{pmatrix}$$

$$D = \begin{pmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix}$$

And:

$$E = \begin{pmatrix} d & 0 & 0 & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & 0 \\ c & d & 0 & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & 0 \\ b & c & d & 0 & \cdots & \cdots & \cdots & \cdots & \cdots & 0 \\ a & b & c & d & 0 & \cdots & \cdots & \cdots & \cdots & 0 \\ \alpha & a & b & c & d & 0 & \cdots & \cdots & \cdots & \vdots \\ \beta & \alpha & a & b & c & d & 0 & \cdots & \cdots & \vdots \\ \gamma & \beta & \alpha & a & b & c & d & 0 & \ddots & \vdots \\ 0 & \gamma & \beta & \alpha & a & b & c & d & 0 & \vdots \\ \vdots & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots & 0 \\ 0 & \cdots & 0 & \gamma & \beta & \alpha & a & b & c & d \end{pmatrix}$$

Then the inverse of the matrix E will be:

$$E^{-1} = \frac{1}{a}\sum_{k=0}^{n-2}\sum_{r=1}^{n-1-k} l_k E_{r+k,r} \tag{2}$$

Where:

$$l_0 = 1;\quad l_1 = \frac{-b}{a};\quad l_2 = (\frac{-b}{a})^2 - \frac{c}{a} = -\frac{1}{a}(bl_1 + cl_0);\quad l_3 = (\frac{-b}{a})^3 + \frac{2cb}{a^2} - \frac{d}{a}$$

$$And\quad l_k = -\frac{1}{a}(bl_{k-1} + cl_{k-2} + dl_{k-3} + \gamma l_{k-4} + \beta l_{k-5} + \alpha l_{k-6})$$

Proof 1 *Since A is a triangular Toeplitz matrix, so is A^{-1}. We prove (11) inductively on n. We have:*

$$\begin{bmatrix} 1 & & & & & \\ 0 & 1 & & & & \\ 0 & -\frac{b}{a} & 1 & & & \\ 0 & (-\frac{b}{a})^2 & -\frac{2b}{a} & 1 & & \\ 0 & (-\frac{b}{a})^3 & -\frac{c}{a}+\frac{3c^2}{d^2} & -\frac{2c}{d} & 1 & \\ 0 & (-\frac{b}{a})^4 & (-\frac{b}{a})^3+\frac{3cb}{a^2} & \frac{3b^2}{a^2}-\frac{2c}{a} & -\frac{2b}{a} & 1 \end{bmatrix} \begin{bmatrix} 1 & & & & & \\ -\frac{b}{a} & 1 & & & & \\ -\frac{c}{a} & 0 & 1 & & & \\ -\frac{d}{a} & -\frac{c}{a} & \frac{b}{a} & 1 & & \\ -\frac{\gamma}{a} & -\frac{d}{a} & 0 & \frac{b}{a} & 1 & \\ -\frac{\beta}{a} & -(\frac{c}{a})^2-\frac{\gamma}{a} & \frac{cb}{a^2}-\frac{d}{a} & \frac{b}{a} & 1 & \end{bmatrix} = \begin{bmatrix} 1 & & & & & \\ l_1 & 1 & & & & \\ l_2 & l_1 & 1 & & & \\ l_3 & l_2 & l_1 & 1 & & \\ l_4 & l_3 & l_2 & l_1 & 1 & \\ l_5 & l_4 & l_3 & l_2 & l_1 & 1 \end{bmatrix} \tag{3}$$

With:

$$l_0 = 1;\quad l_1 = \frac{-b}{a};\quad l_2 = (\frac{-b}{a})^2 - \frac{c}{a} = -\frac{1}{a}(bl_1 + cl_0);\quad l_3 = (\frac{-b}{a})^3 + \frac{2cb}{a^2} - \frac{d}{a};\quad l_4 = (\frac{-b}{a})^4 - \frac{3bc^2}{a^3} + \frac{2db+c^2}{a^2}$$

$$And\ l_5 = (\frac{-b}{a})^5 + \frac{4bc^3}{a^4} - \frac{3c^2b+3db^2}{a^3} + \frac{2dc}{a^2} + \frac{\gamma b}{a^2} - \frac{\beta}{a}$$

The first step of induction holds.
Consider now $n > 4$. We have:

$$\begin{bmatrix} 1 & & & & & & & & & \\ 0 & 1 & & & & & & & & \\ 0 & l_1 & 1 & & & & & & & \\ 0 & l_2 & l_1 & 1 & & & & & & \\ 0 & l_3 & l_2 & l_1 & 1 & & & & & \\ 0 & l_4 & l_3 & l_2 & l_1 & 1 & & & & \\ 0 & l_5 & l_4 & l_3 & l_2 & l_1 & 1 & & & \\ 0 & l_6 & l_5 & l_4 & l_3 & l_2 & l_1 & 1 & & \\ \vdots & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots & \\ 0 & l_{n-3} & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & l_1 & 1 \end{bmatrix} \begin{bmatrix} 1 & & & & & & & & & \\ -\frac{b}{a} & 1 & & & & & & & & \\ -\frac{c}{a} & & & & & & & & & \\ -\frac{d}{a} & & & & & & & & & \\ & & & & \ddots & & & & & \\ -\frac{\gamma}{a} & & & & & & & & & \\ -\frac{\beta}{a} & & & & & & & & & \\ -\frac{\alpha}{a} & & & & & & & & & \\ 0 & & & & & & & & & \\ \vdots & & & & & & & & \ddots & \\ 0 & & & & & & & & & 1 \end{bmatrix} =$$

$$\begin{bmatrix} 1 & & & & & & & & \\ l_1 & 1 & & & & & & & \\ l_2 & l_1 & 1 & & & & & & \\ l_3 & l_2 & l_1 & 1 & & & & & \\ l_4 & l_3 & l_2 & l_1 & 1 & & & & \\ l_5 & l_4 & l_3 & l_2 & l_1 & 1 & & & \\ l_6 & l_5 & l_4 & l_3 & l_2 & l_1 & 1 & & \\ l_7 & l_6 & l_5 & l_4 & l_3 & l_2 & l_1 & 1 & \\ \vdots & \vdots & \vdots & & & & & \ddots & \\ l_{n-2} & l_{n-3} & \cdots & & \cdots & & \cdots & l_1 & 1 \end{bmatrix}$$

Then:

$$l_{n-2} = -\tfrac{1}{a}(bl_{n-3} + cl_{n-4} + dl_{n-5} + \gamma l_{n-6} + \beta l_{n-6} + \alpha l_{n-7})$$

4. Examples

Now, we present a numerical example to show how our algorithm works in practice. MATLAB R2024b is used to test our algorithm.
Example of a9-by-9 pentadiagonal matrix

$$A = \begin{pmatrix} 1 & 2 & 3i & 7+i & 0 & 0 & 0 & 0 & 0 \\ -2i & 1 & 2 & 3i & 7+i & 0 & 0 & 0 & 0 \\ 0.5 & -2 & 1 & 2 & 3i & 7 & 0 & 0 & 0 \\ 4+i & 0.5 & -2i & 1 & 2 & 3i & 7+i & 0 & 0 \\ 0 & 4+i & 0.5 & -2i & 1 & 2 & 3i & 7+i & 0 \\ 0 & 0 & 4+i & 0.5 & -2i & 1 & 2 & 3i & 7+i \\ 0 & 0 & 0 & 4+i & 0.5 & -2i & 1 & 2 & 3i \\ 0 & 0 & 0 & 0 & 4+i & 0.5 & -2i & 1 & 2 \\ 0 & 0 & 0 & 0 & 0 & 4+i & 0.5 & -2i & 1 \end{pmatrix}$$

The columns of the inverse $\mathbf{A}^{-1}$ are:

$$C_1 = \begin{pmatrix} -0.1724 + 0.0749i \\ -0.1215 + 0.0190i \\ -0.0963 - 0.2815i \\ 0.0835 + 0.0132i \\ 0.0276 - 0.0112i \\ -0.0100 - 0.0140i \\ 0.1633 - 0.0648i \\ 0.0436 - 0.0202i \\ -0.0153 + 0.1856i \end{pmatrix}, \; C_2 = \begin{pmatrix} 0.0389 + 0.2279i \\ 0.0769 - 0.0197i \\ 0.0358 + 0.1314i \\ 0.0223 - 0.0455i \\ 0.0317 - 0.0377i \\ -0.0262 - 0.0099i \\ -0.0616 - 0.0868i \\ -0.0647 + 0.0411i \\ 0.0436 - 0.0202i \end{pmatrix}$$

$$C_3 = \begin{pmatrix} 0.2490 + 0.1148i \\ 0.0894 + 0.3139i \\ -0.1841 + 0.2316i \\ 0.0336 - 0.0320i \\ -0.0141 - 0.0522i \\ 0.0290 - 0.0047i \\ -0.2209 - 0.1376i \\ -0.0616 - 0.0868i \\ 0.1633 - 0.0648i \end{pmatrix}, \; C_4 = \begin{pmatrix} 0.1361 - 0.0476i \\ 0.0260 - 0.0120i \\ -0.0035 + 0.0602i \\ 0.0006 + 0.0117i \\ 0.0188 + 0.0205i \\ 0.0015 - 0.0094i \\ 0.0290 - 0.0047i \\ -0.0262 - 0.0099i \\ -0.0100 - 0.0140i \end{pmatrix}$$

$$C_5 = \begin{pmatrix} 0.0104 + 0.0334i \\ 0.1334 + 0.0083i \\ -0.0697 + 0.0258i \\ -0.0248 + 0.0263i \\ 0.0032 + 0.0046i \\ 0.0188 + 0.0205i \\ -0.0141 - 0.0522i \\ 0.0317 - 0.0377i \\ 0.0276 - 0.0112i \end{pmatrix}, \; C_6 = \begin{pmatrix} 0.0090 + 0.0483i \\ -0.0457 + 0.0459i \\ 0.0297 - 0.0637i \\ -0.0198 - 0.0299i \\ -0.0248 + 0.0263i \\ 0.0006 + 0.0117i \\ 0.0336 - 0.0320i \\ 0.0223 - 0.0455i \\ 0.0835 + 0.0132i \end{pmatrix}$$

$$C_7 = \begin{pmatrix} 0.2579 - 0.2279i \\ 0.1079 - 0.1806i \\ 0.3352 + 0.2484i \\ 0.0297 - 0.0637i \\ -0.0697 + 0.0258i \\ -0.0035 + 0.0602i \\ -0.1841 + 0.2316i \\ 0.0358 + 0.1314i \\ -0.0963 - 0.2815i \end{pmatrix}, \; C_8 = \begin{pmatrix} -0.1778 - 0.5023i \\ 0.0010 - 0.0485i \\ 0.1079 - 0.1806i \\ -0.0457 + 0.0459i \\ 0.1334 + 0.0083i \\ 0.0260 - 0.0120i \\ 0.0894 + 0.3139i \\ 0.0769 - 0.0197i \\ -0.1215 + 0.0190i \end{pmatrix}, \; C_9 = \begin{pmatrix} -0.3432 - 0.1164i \\ -0.1778 - 0.5023i \\ 0.2579 - 0.2279i \\ 0.0090 + 0.0483i \\ 0.0104 + 0.0334i \\ 0.1361 - 0.0476i \\ 0.2490 + 0.1148i \\ 0.0389 + 0.2279i \\ -0.1724 + 0.0749i \end{pmatrix}$$

The table compares 'Toeplitz-Hessenberg' andour algorithm (implemented in MATLAB R2024b) execution time. Execution time (in seconds) for the two considered proposed algorithms evaluated in MATLAB R2024b.

Table 1: The running time

Size of the matrix (n)	Algorithm 1	Algorithm 2	LU method
100	0.036954	0.111019	0.918161
200	0.061992	0.195042	2.547606
300	0.090051	0.385720	6.816085
500	0.149696	1.189282	24.165349
1000	0.314484	3.835987	149.750575

6 Conclusion

We develop new numerical and some symbolicalgorithms to compute the inverse of any non-singular heptadiagonal matrix. They arecreated to tackle both the computation speed and the precision in estimating matrix inversion problems. In order to rigorously assess the performance and effectiveness of our fast algorithm (Algorithm 1), we present a comprehensive comparison with two well-known algorithms: the Toeplitz-Hessenberg algorithm (Algorithm 2) and the LU decompositionmethod. The quality from the mention ofcomputational time to accuracy and to the other factors that directly or indirectly concern the computational time and accuracy on heptadiagonal matrix is here to be evaluated.

References

[1] Sogabe T (2008) A fast numerical algorithm for the determinant of a pentadiagonal matrix. Appl Math Comput 196(2):835–841

[2] Losonczi L (2021) Determinants of some pentadiagonal matrices. Glasnik Matematicki 56(2):271–286

[3] general pentadiagonal Toeplitz matrices. Comput Math Appl 71(4):1036–1044

[4] S.M. El-Sayed, D. Kaya, An application of the ADM to seven order Sawada Kotara equations, Appl. Math. Comput. 157 (2004) 93-101.

[5] Jie Shen, Tao Tang, Spectral and High-Order Methods with Applications, Science Press, Beijing, 2006

[6] J. Monterde, H. Ugail, A general 4th-order PDE method to generate bezier surfaces from the boundary, Comput. Aided Geom. Design 23 (2006) 208-225.

[7] P.G. Patil, Y.S. Swamy, An eficient model for vibration control by piezoelectric smart structure using finite element method, Eur. J. Comput. Sci. Netw.Secu. 8 (2008) 258-264.

[8] J.S. Respondek, Numerical recipes for the high efficient inverse of the confluent Vandermonde matrices, Appl. Math. Comput. 218 (2011) 2044-2054.

[9] J.S. Respondek, Recursive numerical recipes for the high efficient inversion of the confluent Vandermonde matrices, Appl. Math. Comput. 225 (2013)718-730.

[10] H. Li, D. Zhao, An extension of the Golden-Thompson theorem, J. Inequal.Appl. (2014).

[11] H.Y. Li, Z.G. Gong, D. Zhao, Least squares solutions of the matrix equation AXB+CYD=E with the least norm for symmetric arrowhead matrices, Appl.Math. Comput. 226 (2014) 719-724.

[12] Burden RL, Faires JD, Burden AM (2015) Numerical analysis. Cengage Learning, Boston

[13] A. Hadj, M. Elouafi, A fast numerical algorithm for the inverse of a tridiagonal and pentadiagonal matrix, Appl. Math. Comput. 202 (2008) 441-445.

[14] D. Aiat Hadj, M. Elouafi the characteristic polynomial, eigenvectors and determinant of a pentadiagonal matrix, Appl. Math.Comput. 198 (2) (2008) 634-642

[15] L. Verde-Star, Elementary triangular matrices and inverses of k-Hessenberg and triangular matrices, Spec. Matrices 3 (2015), 250256.

[16] R. Slowik, Inverses and Determinants of Toeplitz-Hessenberg Matrices, TAIWANESE JOURNAL OF MATHEMATICS. Vol. 22, No. 4, pp. 901–908, August 2018.

[17] J. Abderram´an Marrero and V. Tomeo, On the closed representation for the inverses of Hessenberg matrices, J. Comput. Appl. Math. 236 (2012), no. 12, 2962-2970.

[18] J. Abderram´an Marrero, V. Tomeo and E. Torrano, On inverses of infinite Hessenberg matrices, J. Comput. Appl. Math. 275 (2015), 356-365.

[19] B. Bukhberger and G. A. Emel'yanenko, Methods of inverting tridiagonal matrices, USSR Comput. Math. and Math. Phys. 13 (1973), no. 3, 10-20.

[20] B. A. Turlach, W. N. Venables, and S. J. Wright. Simultaneous variable selection. Technometrics, 47(3):349–363, 2005.

[21] Y. Ikebe, On inverses of Hessenberg matrices, Linear Algebra Appl. 24 (1979), 93-97.

[22] J. Maroulas, Factorization of Hessenberg matrices, Linear Algebra Appl. 506 (2016), 226-243.

[23] M. Merca, A note on the determinant of a Toeplitz-Hessenberg matrix, Spec. Matrices 2 (2014), 10-16.

[24] D. M. Witten, R. Tibshirani, and T. Hastie. A penalized matrix decomposition, with applications to sparse principal components and canonical correlation analysis. Biostatistics, 10(3):515–534, 2009.

[25] T. T. Wu and K. Lange. Coordinate descent algorithms for Lasso penalized regression. Annals of Applied Statistics, 2(1):224–244, 2008.