Journal of Interdisciplinary Mathematics
ISSN: 0972-0502 (Print), ISSN: 2169-012X (Online)
Vol. 29 (2026), No. 4, pp. 809–823
DOI : 10.47974/JIM-2229

On a numerical approach to compute the powers of doubly Leslie matrices using Fibonacci-Horner decomposition and determinantal form

Amal Aloui *
Department of Mathematics
Faculty of Sciences
University of Ibn Tofail
Kénitra
Morocco

Mustapha Rachidi †
Institute of Mathematics
INMA, Federal University of Mato Grosso do Sul
UFMS, Campo Grande
MS 79070-900
Brazil

Abstract

In this paper we explore a numerical approach of the powers of doubly Leslie matrices, through the Fibonacci-Horner method and also using a determinantal form. Two algorithms for computing the entries of the powers of doubly Leslie matrix are built. In addition, Python programs are designed for both of these approaches. For illustrative purpose, some examples and applications are provided.

Subject Classification: *15A15, 39-08, 11B37, 65Q10.*

Keywords: *Fibonacci-Horner, Doubly Leslie matrix, Hessenberg matrix, Determinant, Weighted generalized Fibonacci sequences.*

* *ORCID-ID:* `0000-0002-7535-1652`
† *ORCID-ID:* `0000-0002-8210-7383`

* *E-mail:* `amal.alaoui@uit.ac.ma (Corresponding Author)`
† *E-mail:* `mu.rachidi@gmail.com`

1. Introduction

Powers of square matrices are significant in various mathematical fields and applied science disciplines, including physics, biology, and economics. Therefore, several theoretical approach and numerical methods have been provided in the literature for computing the powers of square matrices. Generally, for computing the powers of a square matrix its eigenvalues play a fundamental role (see, for instance, [8, 9, 11]). Especially, the theoretical and numerical computation of the powers of special matrices has taken on great importance, given their applications in several topics of pure and applied mathematics, as well as in applied sciences. In particular, in the last decades, several research studies have been devoted to the theoretical and numerical calculations of a large family of special matrices, such as tridiagonal matrices, companion matrices, Hessenberg matrices, Leslie matrices and Lefkovitch matrices. (see, for example, [2, 4, 6, 7, 13], and references therein). The present study concerns a new computational process of $\mathbb{L}^n$ the powers of the doubly Leslie $r \times r$ matrix $\mathbb{L}$, defined in [17] under the form

$$\mathbb{L} = \begin{bmatrix} a_0 & a_1 & \cdots & \cdots & a_{r-2} & a_{r-1} \\ s_0 & 0 & \cdots & \cdots & 0 & b_{r-2} \\ 0 & s_1 & \ddots & \cdots & \vdots & b_{r-3} \\ \vdots & \ddots & \ddots & \ddots & \vdots & \vdots \\ \vdots & & \ddots & \ddots & \vdots & \vdots \\ 0 & \cdots & \cdots & 0 & s_{r-2} & b_0 \end{bmatrix}, \tag{1.1}$$

with $0 < s_i \leq 1$, for $0 \leq i \leq r-2$, and $a_i, b_j \geq 0$ for $(0 \leq i \leq r-1, 0 \leq j \leq r-2)$, where not at all zeros. The doubly Leslie matrix (1.1) was also considered in [19] with $a_j, b_k \in \mathbb{R}$ for $(0 \leq j \leq r-1,\ 0 \leq k \leq r-2)$. Furthermore, if $b_k = 0$ for every $0 \leq k \leq r-2$, then $\mathbb{L}$ is a Leslie matrix (see [6]). In [3] the authors embarked on their exploration of the matrix powers for the matrix (1.1) in their prior study, which laid the foundation for the computational and numerical approaches presented in [3]. Their study aimed to furnish concise expressions of the entries for the powers of such kind of matrices, leveraging properties from linear recursive difference equations of constant coefficients.

The goal of this paper is to explore a new numerical approach to establish compact expressions for doubly Leslie matrix powers, through two methods of calculating the inputs of these matrix powers. The first method consists in considering the Fibonacci-Horner decomposition for computing the entries for the powers of the matrix (1.1). The second method for calculating the entries of the powers of the matrix (1.1) is based on the approach of the determinantal form. Moreover, two algorithms were furnished for the computation of the entries of the powers $\mathbb{L}^n$ of the matrix (1.1). In addition, Python programs are designed for both of the algorithms of these two approaches. For illustrative purpose, some applications and examples are exhibited.

The plan of our paper is as follows. Section 2 concerns a compact expression of the power elements of doubly Leslie matrices, with the aid of the Fibonacci-Horner approach. In section 3, we give an explicit expression for the entries of the powers $\mathbb{L}^n$ of the matrix (1.1) under a determinantal form. In Section 4 we elaborate two algorithms for computing the entries of the matrix powers of (1.1). In conclusion, final remarks and perspectives are provided.

2. Fibonacci-Horner decomposition and computation of the n −th powers of the matrix (1.1)

In this section, we will recall the Fibonacci-Horner decomposition method for computing the powers of a square matrix introduced by Ben Taher-Mouline-Rachidi in (see [5, 16]). Let A be in $\mathbb{C}^{r\times r}$ the algebra of square $r\times r$ matrices ($r\geq 2$), whose characteristic polynomial is given by $P(z)=z^r-a_0z^{r-1}-\cdots-a_{r-1}$. Thus, the Theorem of Cayley-Hamilton implies that we have $P(A)=\Theta_r$ (the zero $r\times r$ matrix). Therefore, the matrix sequence $\{A^n\}_{n\geq 0}$ represents a generalized Fibonacci sequence (of order r) in the algebra $\mathbb{C}^{r\times r}$, whose coefficient are $a_0,\cdots,a_{r-1}$, and initial conditions $A^0=I_r$ (the identity matrix), $A,\cdots,A^{r-1}$. Results of [5, 16] allow us to get

$$A^n=u_nA_0+u_{n-1}A_1+\cdots+u_{n-r+1}A_{r-1},\ \text{for all}\ n\geq r, \tag{2.2}$$

where

$$A_0=I_r;A_i=A^i-a_0A^{i-1}-\cdots-a_{i-1}I_r,\ \text{for}\ i=1,\cdots,r-1, \tag{2.3}$$

and the sequence $\{u_n\}_{n\geq -r+1}$ satisfies the following linear recurrence relation

$$\begin{cases} u_{n+1}=a_0u_n+a_1u_{n-1}+\cdots+a_{r-1}u_{n-r+1},\ \text{for all}\ n\geq 0, \\ \text{with}\ u_0=1\ \text{and}\ u_i=0;\ \text{for}\ i=-r+1,\dots,-1. \end{cases} \tag{2.4}$$

The set of matrices $\{A_0,A_1,\cdots,A_{r-1}\}$ is labelled the Fibonacci-Hörner system associated to the matrix A. In order to compute the $n-th$ power of the doubly Leslie matrix (1.1) using the Fibonacci-Horner decomposition [5, 16], we need to compute the explicit formula of the characteristic polynomial of the matrix (1.1). Result of Wanicharpichat shows that this characteristic polynomial of the matrix (1.1) is given as follows.

Lemma 2.1: [19] (Wanicharpichat Lemma.). *The characteristic polynomial of* $\mathbb{L}$ *the matrix of the form (1.1) is given by*

$$P_{\mathbb{L}}(z)=z^r-f_0z^{r-1}-f_1z^{r-2}-\cdots-f_{r-2}z-f_{r-1}, \tag{2.5}$$

where

$$\begin{cases} f_0=c_0+d_0 \\ f_k=-\sum_{i+j=k-1}c_id_j+c_k+d_k\ \text{for every}\ k=1,\cdots,r-2 \\ f_{r-1}=-\sum_{i+j=r-2}c_id_j+c_{r-1} \end{cases} \tag{2.6}$$

with

$$\begin{cases} c_0=a_0,c_i=a_i\prod_{k=0}^{i-1}s_k,\ \text{for}\ 1\leq i\leq r-1, \\ d_0=b_0,d_j=b_j\prod_{k=r-1-j}^{r-2}s_k,\ \text{for}\ 1\leq j\leq r-2. \end{cases} \tag{2.7}$$

Combining Lemma 2.1, namely, Expressions (2.6)-(2.7), with Expressions (2.2)-(2.4), we arrive to the following result.

Theorem 2.2: *Let* $\mathbb{L}$ *be a doubly Leslie matrix (1.1) and* $P_{\mathbb{L}}(z) = z^r - f_0 z^{r-1} - f_1 z^{r-2} - \cdots - f_{r-2}z - f_{r-1}$ *its characteristic polynomial with* f_i *(*$0 \leq i \leq r-1$*) are as in (2.6)-(2.7). Then, the powers* L^n *of the doubly Leslie matrix expressed in the Fibonacci-Horner system* $\{L_0, L_1, \cdots, L_{r-1}\}$ *is as follows,*

$$\mathbb{L}^n = u_n L_0 + u_{n-1} L_1 + \cdots + u_{n-r+1} L_{r-1}, \quad for \quad n \geq r-1, \tag{2.8}$$

with $L_0 = I_r$ *,* $L_1 = \mathbb{L} - f_0 I_r$ *,* $L_2 = \mathbb{L}^2 - f_0\mathbb{L} - f_1 I_r$ *,* $\cdots$ *,* $L_{r-1} = \mathbb{L}^{r-1} - f_0\mathbb{L}^{r-2} - \cdots - f_{r-1} I_r$ *, where* $\{u_n\}_{n \geq -r+1}$ *verifies the following recursive expression of Fibonacci type*

$$u_{n+1} = f_0 u_n + f_1 u_{n-1} + \cdots + f_{r-2} u_{n-r+2} + f_{r-1} u_{n-r+1}, \; n \geq r-1, \tag{2.9}$$

where $u_0 = 1$ *and* $u_i = 0$ *for* $-r+1 \leq i \leq -1$*, and* f_i *(*$0 \leq i \leq r-1$*) are as in (2.6)-(2.7).*

Proof: As mentioned below, the characteristic polynomial (2.5) of the doubly Leslie matrix defined by (1.1) is explicitly given by the Expressions (2.6)-(2.7). Then, using Fibonacci-Horner's decomposition (2.2)-(2.4), we get the formula (2.8) of the powers $\mathbb{L}^n$ $(n \geq 0)$.

Result of Theorem 2.2 can be illustrated by considering the following numerical example.

Example 2.3: Let $\mathbb{L}$ be the doubly Leslie matrix given by

$$\mathbb{L} = \begin{bmatrix} 0 & 12 & 24 \\ 0.25 & 0 & 0.4 \\ 0 & 0.75 & 0.6 \end{bmatrix},$$

such that $P(z) = z^3 - 0.6z^2 - 3.3z - 2.7$. Then, the related linear recursive sequence sequence of Fibonacci type $\{u_n\}_{n \geq -2}$ is given by

$$\begin{cases} u_{n+1} = 0.6u_n + 3.3u_{n-1} + 2.7u_{n-2}, \text{ for } n \geq 0, \\ u_0 = 1 \text{ and } u_{-2} = u_{-1} = 0. \end{cases}$$

Suppose we need to compute $\mathbb{L}^4$. Then, using Expressions (2.8)-(2.9), we get

$$\mathbb{L}^4 = u_4 L_0 + u_3 L_1 + u_2 L_2,$$

with

$$L_0 = I_3, L_1 = \begin{bmatrix} -0.6 & 12 & 24 \\ 0.25 & -0.6 & 0.4 \\ 0 & 0.75 & 0 \end{bmatrix}, L_2 = \begin{bmatrix} -0.3 & 10.8 & 4.8 \\ -0.15 & 0 & 6 \\ 0.1875 & 0 & -3 \end{bmatrix}$$

and $u_2 = 3.66$, $u_3 = 6.876$, $u_4 = 17.8236$. Therefore, we have

$$\mathbb{L}^4 = \begin{bmatrix} 12.6 & 122.04 & 182.592 \\ 1.17 & 13.698 & 24.7104 \\ 0.68625 & 5.157 & 6.8436 \end{bmatrix}.$$

It was shown in [14] that $\{u_n\}_{n \geq -r+1}$ given by (2.4) can be provided under its combinatorial form. Indeed, we can state the following lemma.

Lemma 2.4: *The combinatorial expression of the sequence* $\{u_n\}_{n\geq -r+1}$ *is given under the form*

$$u_n = \sum_{k_0+2k_1+\cdots+rk_{r-1}=n} \frac{(k_0+k_1+\cdots+k_{r-1})!}{k_0!k_1!\cdots k_{r-1}!} a_0^{k_0} a_1^{k_1} \cdots a_{r-2}^{k_{r-2}} a_{r-1}^{k_{r-1}}, \qquad (2.10)$$

for every $n \geq 1$, *where* $u_0 = 1$ *and* $u_i = 0$ *when* $-r+1 \leq i \leq -1$.

For reason of convenience we use the following notation for u_n given as in (2.10)

$$u_n = \rho(n+r,r) = \sum_{k_0+2k_1+\cdots+rk_{r-1}=n} \frac{(k_0+k_1+\cdots+k_{r-1})!}{k_0!k_1!\cdots k_{r-1}!} a_0^{k_0} a_1^{k_1} \cdots a_{r-2}^{k_{r-2}} a_{r-1}^{k_{r-1}}.$$

Application of formula (2.10) to the sequence (2.9) related to the doubly Leslie matrix (1.1), allows us to formulate the following result.

Theorem 2.5: *Under the notation and hypothesis of Theorem 2.2, the powers* $\mathbb{L}^n$ *of the matrix (1.1), expressed as in (2.8), are given under the combinatorial form as follows*

$$\mathbb{L}^n = \rho(n+r,r)L_0 + \rho(n+r-1,r)L_1 + \cdots + \rho(n+1,r)L_{r-1}, \qquad (2.11)$$

for every $n \geq r-1$, *where*

$$\rho(n+r,r) = \sum_{k_0+2k_1+\cdots+rk_{r-1}=n} \frac{(k_0+k_1+\cdots+k_{r-1})!}{k_0!k_1!\cdots k_{r-1}!} f_0^{k_0} f_1^{k_1} \cdots f_{r-2}^{k_{r-2}} f_{r-1}^{k_{r-1}}, \qquad (2.12)$$

for every $n \geq 0$, *with* $\rho(r,r) = 1$ *and* $\rho(i+r,r) = 0$, *when* $-r+1 \leq i \leq -1$, *and the scalars* f_i *(*$0 \leq i \leq r-1$*) are as in* (2.6)-(2.7).

3. Powers of the matrix (1.1) under a determinantal form

In the section we provide the approach for computing the powers of matrix (1.1), using a method based on the Hessenberg matrix and its determinant. Recall that for a given sequence $\{a_n\}_{n\geq 0}$ the related the upper Hessenberg matrix of order $n \times n$ is stated as follows

$$H_n = \begin{bmatrix} a_1 & a_2 & \cdots & \cdots & a_{n-1} & a_n \\ a_0 & a_1 & \ddots & \cdots & a_{n-2} & a_{n-1} \\ 0 & a_0 & \ddots & \ddots & a_{n-3} & a_{n-2} \\ \vdots & \ddots & \ddots & \ddots & \vdots & \vdots \\ \vdots & \cdots & \ddots & \ddots & a_1 & a_2 \\ 0 & \cdots & \cdots & 0 & a_0 & a_1 \end{bmatrix}. \qquad (3.13)$$

The well known usual formula for the determinant of matrix (3.13) is exhibited under the following combinatorial formula

$$det(H_n) = \sum_{s_1+2s_2+\cdots+ns_n=n} \frac{(-a_0)^{n-|s|}}{|s|} \left(\sum_{i=1}^{n} s_i\right) m_n(s) a_1^{s_1} a_2^{s_2} \cdots a_n^{s_n}, \quad n \geq 1, \qquad (3.14)$$

where $m_n(s) = \frac{(s_1+\cdots+s_n)!}{s_1!\cdots s_n!}$ and $|s| = s_1 + \cdots + s_n$ for an $\mathrm{n}-$ tuple $s = (s_1, \cdots, s_n)$ of non−negative integers (see, for example, [10, 15]). In addition, it is shown in [10, 15] that $det(H_n)$ can be also expressed recursively, by repeatedly expanding along the last column, under the form

$$det(H_n) = (-a_0)^{n-1}a_n + \sum_{i=1}^{n-1} (-a_0)^{i-1}a_i det(H_{n-i}). \qquad (3.15)$$

When the sequence $\{a_n\}_{n\geq 0}$ is finite, namely $(a_0, a_1, \cdots, a_{r-1})$, the Hessenberg matrix of order n is given by

$$H_n = \begin{bmatrix} a_1 & a_2 & \cdots & a_{r-1} & 0 & \cdots & 0 \\ a_0 & a_1 & \cdots & a_{r-2} & a_{r-1} & \ddots & 0 \\ 0 & a_0 & a_1 & \cdots & a_{r-2} & a_{r-1} & 0 \\ \vdots & \cdots & \ddots & \ddots & \ddots & \vdots & \vdots \\ \vdots & \cdots & \ddots & \ddots & \ddots & \vdots & \vdots \\ \vdots & \cdots & \cdots & \ddots & \ddots & a_1 & a_2 \\ 0 & \cdots & \cdots & \cdots & 0 & a_0 & a_1 \end{bmatrix} \qquad (3.16)$$

and its determinant is

$$\det(H_n) = \sum_{s_1+2s_2+\cdots+(r-1)s_{r-1}=n} (-a_0)^{n-|s|}m(s)a_1^{s_1}a_2^{s_2}\cdots a_{r-1}^{s_{r-1}}, \quad n \geq r-1, \qquad (3.17)$$

where $m(s) = \frac{(s_1+\cdots+s_{r-1})!}{s_1!\cdots s_{r-1}!}$. In the sequel, we furnish another expression of (2.8) with the aid of the determinantal form. Indeed, we can also get the expression of the general term of (2.4) under a determinantal expression. More precisely, such determinantal expression of u_m, for $m \geq 1$, is obtained using a matrix of order $m \times m$ (see, for instance, [12, 18]), is formulated under the following form

$$u_m = \begin{vmatrix} a_0 & a_1 & & a_{r-1} & 0 & & 0 \\ -1 & a_0 & & a_{r-2} & a_{r-1} & & 0 \\ \vdots & \vdots & & & & & \vdots \\ \vdots & \vdots & & \ddots & & & \vdots \\ 0 & 0 & & & & a_0 & a_1 \\ 0 & 0 & & & & -1 & a_0 \end{vmatrix}. \qquad (3.18)$$

When $r+1 \leq m$, Expression (3.18) is obtained from notion of the determinant of a special Toeplitz $(r+1)$-band matrix. Following ([10, 15]), we can deduce that (3.18) can be expressed as follows

$$u_m = \sum_{s_0+2s_1+\cdots+rs_{r-1}=m} m(s)a_0^{s_0}a_1^{s_1}\cdots a_{r-1}^{s_{r-1}}, \qquad (3.19)$$

where $m(s) = \frac{(s_0+\cdots+s_{r-1})!}{s_0!\cdots s_{r-1}!}$. The formula (3.19) is, in essence, identical to formula (2.10).

Despite the fact that the following result will not be taken into account in the sequel, inspired by the formulation (3.18) of the term u_m, we will give a new expression of the determinant for the matrix powers A^n $(n \geq r)$ of the matrix A, in the power basis.

Proposition 3.1: Taking into account the preceding data, we have

$$A^n = \sum_{k=0}^{r-1} u_{n-r+1}^{(k)}A^{r-1-k} = \sum_{k=0}^{r-1} u_{n-r+1}^{(r-1-k)}A^k, \qquad (3.20)$$

for all $n \geq r$, where $u_{n-r+1}^{(k)}$ represents the determinant of a matrix of order $(n-r+1) \times (n-r+1)$, which is identical to that of u_{n-r+1} given by (3.18), where the first row is simply shifted forward with k positions and $u_{n-r+1}^{(0)} = u_{n-r+1}$ (see for instance, [1, 12]).

Proof: We will proceed by induction. Indeed, since we have $P(A) = \Theta_r$, we deduce that the result is verified because $A^r = a_0 A^{r-1} + a_1 A^{r-2} + \cdots + a_{r-1} I_r$. Let us suppose that A^{n-1} is written in the following form $A^{n-1} = \sum_{k=0}^{r-1} u_{n-r}^{(k)} A^{r-1-k}$. Therefore, using the induction we get

$$A^n = \sum_{k=1}^{r-1} u_{n-r}^{(k)} A^{r-k} + u_{n-r}^{(0)} A^r = \sum_{k=0}^{r-2} u_{n-r}^{(k+1)} A^{r-1-k} + u_{n-r}^{(0)} \sum_{k=0}^{r-1} a_k A^{r-1-k}.$$

Hence, we have $A^n = \sum_{k=0}^{r-2} \left(a_k u_{n-r}^{(0)} + u_{n-r}^{(k+1)} \right) A^{r-1-k} + a_{r-1} u_{n-r}^{(0)} A^0$. Recall that we have $a_{r-1} u_{n-r}^{(0)} = u_{n-r+1}^{(r-1)}$. Furthermore, taking into account the linear recurrence formulas satisfied by these determinants, $a_k u_{n-r}^{(0)} + u_{n-r}^{(k+1)} = a_k u_{n-r}^{(0)} + a_{k+1} u_{n-r-1}^{(0)} + \cdots + a_{r-1} u_{n-r-(r-k-1)}^{(0)} = u_{n-r+1}^{(k)}$. Therefore, the formula of A^n in the basis of matrix powers $\{A^j;\ 0 \le j \le r-1\}$ yields $A^n = \sum_{k=0}^{r-1} u_{n-r+1}^{(k)} A^{r-1-k}$.

Let us apply Expression (3.20) to the sequence $\{u_n\}_{n \ge -r+1}$ defined by (2.9), which is related to the powers $\mathbb{L}^n$ of the matrix (1.1). That is, for the sequence (2.9) the determinantal representation of the general term u_m, for $m \ge 1$, using an upper Hessenberg matrix $m \times m$ (see [12, 18]), is as follows

$$u_m = \begin{vmatrix} f_0 & f_1 & \cdots & f_{r-1} & 0 & \cdots & 0 \\ -1 & f_0 & \cdots & f_{r-2} & f_{r-1} & \ddots & 0 \\ \vdots & \vdots & \ddots & \cdots & \cdots & \cdots & \vdots \\ \vdots & \vdots & \cdots & \ddots & \cdots & \cdots & \vdots \\ \vdots & \vdots & \cdots & \cdots & \ddots & \cdots & \vdots \\ 0 & 0 & \cdots & \cdots & \cdots & f_0 & f_1 \\ 0 & 0 & \cdots & \cdots & \cdots & -1 & f_0 \end{vmatrix} \tag{3.21}$$

For $r+1 \le m$, Expression (3.21) results from the determinant form of a $(r+1)$-banded Toeplitz matrix. We introduce a novel determinantal formulation for the powers $\mathbb{L}^n$ $(n \ge r)$ in the power basis.

Theorem 3.2: *Under the preceding data, we have*

$$\mathbb{L}^n = \sum_{k=0}^{r-1} u_{n-r+1}^{(k)} \mathbb{L}^{r-1-k} = \sum_{k=0}^{r-1} u_{n-r+1}^{(r-1-k)} \mathbb{L}^k, \tag{3.22}$$

for all $n \ge r$, *where* $u_{n-r+1}^{(k)}$ *is the determinant of a matrix of order* $(n-r+1) \times (n-r+1)$, *which is identical to that of* u_{n-r+1} *given by* (3.21), *where the first row is simply shifted forward with* k *positions and* $u_{n-r+1}^{(0)} = u_{n-r+1}$.

The following two examples allows us to show the efficiency of the determinental method, for computing the powers $\mathbb{L}^n$ $(n \ge r)$ of the matrix (1.1).

Example 3.3: Let consider the same matrix as in Example 2.3. Suppose that the calculation of the power $\mathbb{L}^4$ is requested. Then, with the aid of formula (3.22) of Theorem 3.2, we obtain

$$\mathbb{L}^4 = u_2^{(2)} I_3 + u_2^{(1)} \mathbb{L} + u_2^{(0)} \mathbb{L}^2,$$

where $u_2^{(0)} = \begin{vmatrix} 0.6 & 3.3 \\ -1 & 0.6 \end{vmatrix} = 3.66$, $u_2^{(1)} = \begin{vmatrix} 3.3 & 2.7 \\ -1 & 0.6 \end{vmatrix} = 4.68$ and $u_2^{(2)} = \begin{vmatrix} 2.7 & 0 \\ -1 & 0.6 \end{vmatrix} = 1.62$. Therefore, we obtain the same result as in Example 2.3.

Example 3.4: Let consider the following doubly Leslie matrix $\mathbb{L}$,

$$\mathbb{L} = \begin{bmatrix} 3 & 10 & 25 & 20 & 32 \\ 0.5 & 0 & 0 & 0 & 0.7 \\ 0 & 0.25 & 0 & 0 & 0.1 \\ 0 & 0 & 0.1 & 0 & 0.3 \\ 0 & 0 & 0 & 0.75 & 0.2 \end{bmatrix},$$

such that $P(z) = z^5 - 3.2z^4 - 4.625z^3 - 1.4575z^2 + 1.509375z + 0.53$. Suppose that we have to calculate $\mathbb{L}^8$. Then, with the aid of formula (3.22) of Theorem 3.2, we obtain

$$\mathbb{L}^8 = u_4^{(4)} I_5 + u_4^{(3)} \mathbb{L} + u_4^{(2)} \mathbb{L}^2 + u_4^{(1)} \mathbb{L}^3 + u_4^{(0)} \mathbb{L}^4,$$

where

$$u_4^{(0)} = \begin{vmatrix} 3.2 & 4.625 & 1.4575 & -1.509375 \\ -1 & 3.2 & 4.625 & 1.4575 \\ 0 & -1 & 3.2 & 1.4625 \\ 0 & 0 & -1 & 3.2 \end{vmatrix} = 276.14685,$$

$$u_4^{(1)} = \begin{vmatrix} 4.625 & 1.4575 & -1.509375 & -0.53 \\ -1 & 3.2 & 4.625 & 1.4575 \\ 0 & -1 & 3.2 & 1.4625 \\ 0 & 0 & -1 & 3.2 \end{vmatrix} = 311.498675,$$

$$u_4^{(2)} = \begin{vmatrix} 1.4575 & -1.509375 & -0.53 & 0 \\ -1 & 3.2 & 4.625 & 1.4575 \\ 0 & -1 & 3.2 & 1.4625 \\ 0 & 0 & -1 & 3.2 \end{vmatrix} = 68.89280,$$

$$u_4^{(3)} = \begin{vmatrix} -1.509375 & -0.53 & 0 & 0 \\ -1 & 3.2 & 4.625 & 1.4575 \\ 0 & -1 & 3.2 & 1.4625 \\ 0 & 0 & -1 & 3.2 \end{vmatrix} = -104.21506$$

and

$$= \begin{vmatrix} -0.53 & 0 & 0 & 0 \\ -1 & 3.2 & 4.625 & 1.4575 \\ 0 & -1 & 3.2 & 1.4625 \\ 0 & 0 & -1 & 3.2 \end{vmatrix} = -33.827515.$$

Therefore,

$$\mathbb{L}^8 = \begin{bmatrix} 91145.0654 & 241837.429 & 539597.669 & 560271.523 & 801609.202 \\ 10533.3871 & 27948.3674 & 62358.6278 & 64748.8690 & 92638.8973 \\ 608.680591 & 1615.12777 & 3603.89416 & 3741.74953 & 5353.59832 \\ 14.2492403 & 37.7940025 & 84.3328754 & 87.5676631 & 125.293437 \\ 2.58887672 & 6.87615084 & 15.3278593 & 15.9331041 & 22.7797784 \end{bmatrix}.$$

4. Algorithms

4.1 *Algorithm 1*

As a summary of Section 2, we show an algorithm which is designed to compute $\mathbb{L}^n$ with $n \geq r - 1$ using Fibonacci-Horner decomposition. We show the algorithm in Algorithm 1.

Algorithm 1: Computing $\mathbb{L}^n$ using Fibonacci-Horner approach

Input: $\mathbf{L}$,n.

Let r be the order of the doubly Leslie matrix $\mathbb{L}$ (1.1)

Step 1 : *Construct doubly Leslie matrix*

Step 2 : *Compute coefficients c_i, $i = 0, \cdots, r-1$ and $d_j, j = 0, \cdots, r-2$*

Step 3 : *Compute $f_i (0 \leq i \leq r-1)$ by the use of* (2.6) – (2.7)

Step 4 : *Compute Fibonacci – Horner basis* $\{L_0, L_1, \cdots, L_{r-1}\}$

Step 5 : *Compute u_n for every $-r+1 \leq n$ using Expression* (2.9)

Step 6 : *Retrieve the terms computed in Step 5 for $n \geq 0$*

Step 6 : *return $\mathbb{L}^n$ using* (2.8)

Output: $\mathbf{L}^n$.

4.2 *Algorithm 2*

To summarize Section 3, we present an algorithm that computes $\mathbb{L}^n$ for $n \geq r$ using the determinantal approach. This algorithm is detailed in Algorithm 2.

Algorithm 2: Computing $\mathbb{L}^n$ using determinantal approach

Input: $\mathbf{L}$,n.

Let r be the order of the doubly Leslie matrix $\mathbb{L}$ (1.1)

Step 1 : *Construct doubly Leslie matrix*

Step 2 : *Compute coefficients $c_i, i = 0, \cdots, r-1$ and $d_j, j = 0, \cdots, r-2$*

Step 3 : *Compute $f_i (0 \leq i \leq r-1)$ by the use of* (2.6) – (2.7)

Step 4 : *Compute u^k_{n-r+1} for every $0 \leq k \leq r-1$*

Step 5 : *return $\mathbb{L}^n$ using* (3.22)

Output: $\mathbf{L}^n$.

5. Conclusion and perspectives

In this paper we investigated the powers of doubly Leslie matrices using both the Fibonacci-Horner method and a determinantal method, leading to the development of two algorithms for the calculation of the entries of these matrices. We illustrated our methods through examples, demonstrating the practicality and efficiency of the proposed algorithms in handling such computations. Future research could explore extending these techniques to other classes of matrices and further applications in various fields, enhancing the understanding and utility of matrix power computations.

6. Appendices

A. *Python program using Fibonacci-Horner decomposition for compute the powers of (1.1)*

```
import numpy as np

# Function to construct the doubly Leslie matrix
def doubly_leslie_matrix(a, b, s):
r = len(a)
L = np.zeros((r, r))
L[0, :] = a
for i in range(1, r):
L[i, i-1] = s[i-1]
for j in range(len(b)):
L[r-1-j, r-1] = b[j]
return L

# Function to compute the coefficients c_i and d_j
def calculate_c_d(a, b, s):
r = len(a)
c = [a[0]]
d = [b[0]]

for i in range(1, r):
c.append(a[i] * np.prod(s[:i]))

for j in range(1, r-1):
d.append(b[j] * np.prod(s[r-j-1:r-1]))

return c, d

# Function to compute f_i
def calculate_f(c, d):
r = len(c)
f = [c[0] + d[0]]

for k in range(1, r-1):
sum_terms = -sum(c[i] * d[k-1-i] for i in range(k))
f.append(sum_terms + c[k] + d[k])
sum_terms = -sum(c[i] * d[r-2-i] for i in range(r-1))
f.append(sum_terms + c[-1])

return f

# Function to compute L_i
def calculate_L_i(L, f):
r = L.shape[0]
L_i = [np.eye(r)]
```

```
L_i.append(L - f[0] * np.eye(r))

for i in range(2, r):
Li = np.linalg.matrix_power(L, i)
for j in range(i):
Li -= f[j] * np.linalg.matrix_power(L, i-1-j)
L_i.append(Li)

return L_i

# Function to compute the terms of the sequence u_n
def calculate_u_n(f, n, r):

u = [0] * (r-1) + [1] # u_0 = 1 et u_i = 0 pour i < 0

for i in range(n):
u_next = sum(f[j] * u[-1-j] for j in range(r) if -1-j >= -len(u))
u.append(u_next)

return u

def remove_leading_zeros(u):
for i in range(len(u)):
if u[i] == 1:
return u[i:]
return u #

# Function to compute L^n
def power_of_doubly_leslie_matrix(a, b, s, n):
L = doubly_leslie_matrix(a, b, s)
c, d = calculate_c_d(a, b, s)
f = calculate_f(c, d)
L_i = calculate_L_i(L, f)
u = calculate_u_n(f, n, len(a))
u=remove_leading_zeros(u)

Ln = sum(u[n-i] * L_i[i] for i in range(len(a)))

return Ln

a = [0, 12, 24]
b = [0.6, 0.4]
s = [0.25, 0.75]
n = 800 # Par exemple

L=doubly_leslie_matrix(a, b, s)
print(L)
c,d=calculate_c_d(a, b, s)
print(c,d)
f=calculate_f(c, d)
print(f)
u=calculate_u_n(f, n, 3)
print(u)
u=remove_leading_zeros(u)
print(u)
L_i=calculate_L_i(L, f)
print(L_i)
print(np.linalg.matrix_power(L,800))
```

```
Ln = power_of_doubly_leslie_matrix(a, b, s, n)
print("L^n =", Ln)
```

B. *Python program using determinantal form*

```
import numpy as np

# Function to construct the doubly Leslie matrix
def doubly_leslie_matrix(a, b, s):
    r = len(a)
    L = np.zeros((r, r))
    L[0, :] = a
    for i in range(1, r):
        L[i, i-1] = s[i-1]
    for j in range(len(b)):
        L[r-1-j, r-1] = b[j]
    return L

# Function to calculate coefficients c_i and d_j
def calculate_c_d(a, b, s):
    r = len(a)
    c = [a[0]]
    d = [b[0]]

    for i in range(1, r):
        c.append(a[i] * np.prod(s[:i]))

    for j in range(1, r-1):
        d.append(b[j] * np.prod(s[r-j-1:r-1]))

    return c, d

# Function to calculate f_i
def calculate_f(c, d):
    r = len(c)
    f = [c[0] + d[0]]

    for k in range(1, r-1):
        sum_terms = -sum(c[i] * d[k-1-i] for i in range(k))
        f.append(sum_terms + c[k] + d[k])

    sum_terms = -sum(c[i] * d[r-2-i] for i in range(r-1))
    f.append(sum_terms + c[-1])

    return f

# Function to calculate the determinant of the shifted upper Hessenberg
↪ matrix
def calculate_u_m_shifted(f, m, r, k):
    T = np.zeros((m, m))

    # Fill the first row
    if k < r:
        for j in range(m):
            if j + k < r:
                T[0, j] = f[j + k]

    # Fill the remaining rows
```

```
    for i in range(1, m):
        T[i, i-1] = -1
        for j in range(i, m):
            if j - i < r:
                T[i, j] = f[j - i]

    return np.linalg.det(T)

# Main function to calculate L^n using the determinant approach def
power_of_doubly_leslie_matrix_determinantal(a, b, s, n):
    L = doubly_leslie_matrix(a, b, s)
    c, d = calculate_c_d(a, b, s)
    f = calculate_f(c, d)

    r = len(a)
    m = n - r + 1
    if m < 1:
        raise ValueError("The value of n is too small to calculate the
        ↪ matrix power using this method.")

    u_shifted = []
    for k in range(r):
        u_m_k = calculate_u_m_shifted(f, m, r, k)
        u_shifted.append(u_m_k)
    Ln = sum(u_shifted[r-1-k] * np.linalg.matrix_power(L, k) for k in
    ↪range(r))

    return Ln

# Example usage
a = [0, 12, 24]    # Replace with your values
b = [0.6, 0.4]     # Replace with your values
s = [0.25, 0.75]   # Replace with your values
n = 4              # Example power
Ln = power_of_doubly_leslie_matrix_determinantal(a, b, s, n)
print("L^n =", Ln)
L=doubly_leslie_matrix(a, b, s)
print(np.linalg.matrix_power(L,4))
```

References

[1] J. A. Marrero, R. Bentaher, and M. Rachidi, "On explicit formulas for the principal matrix logarithm," *Appl. Math. Comput.*, vol. 220, pp. 142–148 (2013).

[2] D. A. Hadj Ahmed, A. Bentaleb, M. Rachidi, and F. Zitan, "Powers of matrices by density and divided differences," *Int. J. Algebra*, vol. 3, no. 9, pp. 407–422 (2009).

[3] A. Aloui, M. Rachidi, and B. El Wahbi, "On a numerical approach for the powers of the doubly leslie and doubly companion matrices with applications," *Comput. Sci.*, vol. 16, no. 2, pp. 613–638 (2021).

[4] A. Aloui and M. Rachidi, "On the computational and numerical approaches for the powers of the doubly Lefkovitch matrix by linear difference equations," *J. Math. Comput. Sci.*, vol. 31, no. 3, pp. 287–304 (2023).

[5] R. Ben Taher, M. Mouline, and M. Rachidi, "Fibonacci-Horner decomposition of the matrix exponential and the fundamental system of solutions," *Electron. J. Linear Algebra*, vol. 15, pp. 178–190 (2006).

[6] R. Ben Taher, N. Naassi, and M. Rachidi, "On the Leslie matrices, Fibonacci sequences and population dynamics," *J. Discrete Math. Sci. Cryptogr.*, vol. 20, no. 2, pp. 565–594 (2017).

[7] B. N. Datta and K. Datta, "An algorithm for computing powers of a Hessenberg matrix and its applications," *Linear Algebra Appl.*, vol. 14, no. 3, pp. 273–284 (1976).

[8] F. R. Gantmacher, *The Theory of Matrices*, vol. 1. New York, NY, USA: Chelsea (1959).

[9] F. R. Gantmacher, *Applications of the Theory of Matrices*, vol. 2. New York, NY, USA: Interscience (1959).

[10] T. Goy and M. Shattuck, "Determinant formulas of some Hessenberg matrices with Jacobsthal entries," *Appl. Appl. Math. Int. J.*, vol. 16, no. 1, Art. no. 10 (2021).

[11] R. A. Horn and C. R. Johnson, *Matrix Analysis*. Cambridge, U.K.: Cambridge Univ. Press (1997).

[12] R. K. Kittappa, "A representation of the solution of the nth order linear difference equation with variable coefficients," *Linear Algebra Appl.*, vol. 193, pp. 211–222 (1993).

[13] H. Kıyak, F. Yılmaz, and D. Bozkurt, "A formula for computing integer powers for one type of tridiagonal matrix," *Hacet. J. Math. Stat.*, vol. 39, no. 3, pp. 351–363 (2010).

[14] M. Mouline and M. Rachidi, "Application of Markov chains properties to r-generalized Fibonacci sequences," *Fibonacci Quart.*, vol. 37, no. 1, pp. 34–38 (1999).

[15] M. Mouline and M. Rachidi, "Suites de Fibonacci généralisées, Théorème de Cayley–Hamilton et Chaines de Markov," *Rend. Semin. Mat. Messina, Ser. II*, vol. 4, no. 19, pp. 107–115 (1996/97).

[16] T. Muir, *The Theory of Determinants in the Historical Order of Development*, vol. 3. New York, NY, USA: Dover (1960).

[17] H. Pickmann-Soto, S. Arela-Perez, H. Nina, and E. Valero, "Inverse maximal eigenvalues problems for Leslie and doubly Leslie matrices," *Linear Algebra Appl.*, vol. 592, pp. 93–112 (2020).

[18] L. Verde-Star, "Functions of matrices," *Linear Algebra Appl.*, vol. 406, pp. 285–300 (2005).

[19] W. Wanicharpichat, "Explicit minimum polynomial, eigenvector and inverse formula of doubly Leslie matrix," *J. Appl. Math. Inform.*, vol. 33, no. 3-4, pp. 247–260 (2015).

Received November, 2024