axioms

MDPI

Article

A Heuristic Method for Solving Polynomial Matrix Equations

Juan Luis González-Santander *,† and Fernando Sánchez Lasheras *,†

Department of Mathematics, Universidad de Oviedo, 33007 Oviedo, Asturias, Spain

* Correspondence: gonzalezmarjuan@uniovi.es (J.L.G.-S.); sanchezfernando@uniovi.es (F.S.L.)

† These authors contributed equally to this work.

Abstract: We propose a heuristic method to solve polynomial matrix equations of the type $\sum_{k=1}^{m} a_k X^k = B$, where a_k are scalar coefficients and X and B are square matrices of order n. The method is based on the decomposition of the B matrix as a linear combination of the identity matrix and an idempotent, involutive, or nilpotent matrix. We prove that this decomposition is always possible when $n = 2$. Moreover, in some cases we can compute solutions when we have an infinite number of them (singular solutions). This method has been coded in MATLAB and has been compared to other methods found in the existing literature, such as the diagonalization and the interpolation methods. It turns out that the proposed method is considerably faster than the latter methods. Furthermore, the proposed method can calculate solutions when diagonalization and interpolation methods fail or calculate singular solutions when these methods are not capable of doing so.

Keywords: polynomial matrix equations; idempotent matrix; involutive matrix; nilpotent matrix

MSC: 15A24

check for updates

Citation: González-Santander, J.L.; Sánchez Lasheras, F. A Heuristic Method for Solving Polynomial Matrix Equations. *Axioms* **2024**, *13*, 239. https://doi.org/10.3390/axioms13040239

Academic Editor: Jorge Delgado Gracia

Received: 26 February 2024
Revised: 30 March 2024
Accepted: 2 April 2024
Published: 4 April 2024

1. Introduction

A polynomial matrix equation of degree m is an equation of the type

$$\sum_{k=0}^{m} A_k X^k = \mathbf{0}, \tag{1}$$

where $\mathbf{0}$ denotes the null square matrix of order n, and A_k, X are square matrices of order n, i.e., $A_k, X, \mathbf{0} \in \mathcal{M}_n(\mathbb{C})$, where $A_m \neq \mathbf{0}$. The matrices A_k are given matrices, and X is the unknown matrix to be solved. In a similar way to the theory of polynomial equations, polynomial matrix equations have a noncommutative analog of the Vieta theorem [1].

The solution of (1) is fundamental in the analysis of queuing problems modeled by Markov chains [2], as well as in the interplay between Toeplitz matrices and polynomial computations (see [3] and the references therein).

A particular problem of (1) that has been examined by many researchers is the calculation of the m-th roots of a given matrix $B \in \mathcal{M}_n(\mathbb{C})$ [4–7]:

$$X^m = B.$$

Another particular problem arising from (1) is to consider X a scalar matrix, i.e., $X = \lambda I$, thus (1) is reduced to so-called lambda matrices

$$L(\lambda) = \sum_{k=1}^{m} A_k \lambda^k = B.$$

When $m = 2$, the quadratic matrix polynomial $L(\lambda)$ arises in a natural way in the study of damped vibrating systems [8]. Matrix polynomials of arbitrary degree m are studied in [9].

Newton's method [10] has been proposed to numerically solve (1). However, the solution obtained depends on the initial iteration. Also, the solution has to a be a simple root in order to obtain quadratic convergence. In addition, this method has a high computational cost per iteration. Therefore, this method will not be considered further here.

When X is a diagonalizable matrix, the solutions of (1) can be obtained for arbitrary m [11]. The maximum finite number of diagonalizable solutions are $\binom{m\,n}{m}$. Next, we describe this method. Let $\vec{v}_i$ be an eigenvector of matrix X, and λ_i the corresponding eigenvalues, such that

$$X\,\vec{v}_i = \lambda_i\,\vec{v}_i. \tag{2}$$

According to (1) and (2), we have

$$\vec{0} = \mathbf{0}\,\vec{v}_i = \left(\sum_{k=0}^{m} A_k X^k\right)\vec{v}_i = \underbrace{\left(\sum_{k=0}^{m} A_k \lambda_i^k\right)}_{L(\lambda_i)}\vec{v}_i.$$

Therefore, the eigenvalues λ_i of X satisfy the polynomial equation:

$$\det L(\lambda_i) = 0, \tag{3}$$

and the eigenvectors $\vec{v}_i$ satisfy

$$\vec{v}_i \in \ker L(\lambda_i). \tag{4}$$

Let P be the following matrix:

$$P = (\vec{v}_1, \dots, \vec{v}_\ell),$$

where the eigenvectors $\vec{v}_i$ are set in columns in matrix P, and $\ell \leq n$. If X is a diagonalizable matrix, then

$$X = P\begin{pmatrix} \lambda_1 & & \\ & \ddots & \\ & & \lambda_\ell \end{pmatrix}P^{-1}. \tag{5}$$

The above method has been coded in MATLAB, and it is available at https://shorturl.at/oHN15 (accessed on 19 March 2024). However, this method has two main drawbacks:

- It only works when X is a diagonalizable matrix;
- When X is diagonalizable, it is computationally expensive. Note that we have to solve in (3) a polynomial equation of at most degree $m \times n$. Also, for each eigenvalue λ_i, we have to solve in (4) the corresponding eigenvectors $\vec{v}_i$, but many of these eigenvectors are redundant. Finally, there are many possibilities to construct the factorization in (5) in order to obtain all the different solutions for X. Again, there are many redundant possibilities.

The aim of this paper is to propose a very simple heuristic method to obtain solutions of (1) when the matrices A_k in (1) are scalar matrices. This heuristic method always works when the order of the matrices are $n = 2$, and it is not difficult to program. In the recent literature, we have found another approach to solve this kind of polynomial matrix equation with scalar coefficients [12], which is based on an interpolation method. Despite the fact that this interpolation method is much more complicated, we have coded it in MATLAB in order to compare it to our heuristic method. In fact, all the examples presented in this paper have been computed using a MATLAB code available at https://shorturl.at/oHN15.

This paper is organized as follows. Section 2 describes the heuristic method for square matrices of arbitrary order n when matrix B can be expressed as a linear combination of the identity matrix and a nilpotent, involutive or idempotent matrix. Whenever this decomposition is possible, an algorithm to find it is described. Section 3 deals with the particular case $n = 2$, since the aforementioned decomposition for matrix B is always possible in this case. When B is a scalar matrix, we sometimes have an infinite number of solutions by virtue of the Cayley–Hamilton theorem. We explicitly calculate this kind of solution for $n = 2$. As an application, some examples are given from graph theory. Finally, we present our conclusions in Section 4.

2. Polynomial Matrix Equations of Arbitrary Order

Consider the polynomial matrix equation in X of degree m:

$$\sum_{k=1}^{m} a_k\, X^k = B, \tag{6}$$

where $a_k \in \mathbb{C}$, and X, B are square matrices of order n, i.e., $X, B \in \mathcal{M}_n(\mathbb{C})$. Assume that B admits the following decomposition:

$$B = p\,N + q\,I, \tag{7}$$

where $p, q \in \mathbb{C}$; $I, N \in \mathcal{M}_n(\mathbb{C})$, where N is an idempotent, involutive, or nilpotent matrix with index 2; and I is the identity matrix, i.e.,

$$\begin{aligned} N \text{ idempotent} &\quad\Leftrightarrow\quad N^2 = N, \\ N \text{ involutive} &\quad\Leftrightarrow\quad N^2 = I, \\ N \text{ nilpotent with index 2} &\quad\Leftrightarrow\quad N^2 = \mathbf{0}. \end{aligned}$$

Note that if N is a nilpotent matrix, $H = p\,N$ is also a nilpotent matrix, thus in this case, we will consider the decomposition

$$B = H + q\,I. \tag{8}$$

We look for solutions of the form:

$$X = \lambda\,N + \mu\,I \tag{9}$$

or

$$X = \lambda\,H + \mu\,I. \tag{10}$$

2.1. The Heuristic Decomposition

We want to know when the heuristic decomposition given in (7) is possible. Notice that the idempotent decomposition is always possible when B is a scalar matrix, when $p = 0$ and N is arbitrary (say $N = \mathbf{0}$). When B is not a scalar matrix, from (7), we have

$$(B - qI)^2 = p^2 N^2,$$

and

$$N = \frac{1}{p}(B - qI), \tag{11}$$

thus, defining

$$M_{\alpha,\beta}(B) = B^2 - \alpha\,B + \beta\,I, \tag{12}$$

we have that

$$M_{\alpha,\beta}(B) = \mathbf{0}, \tag{13}$$

where

$$N \text{ idempotent} \quad\Rightarrow\quad \left\{ \begin{array}{l} \alpha = 2q + p, \\ \beta = q(p + q), \end{array} \right. \tag{14}$$

$$N \text{ involutive} \quad\Rightarrow\quad \left\{ \begin{array}{l} \alpha = 2q, \\ \beta = q^2 - p^2. \end{array} \right. \tag{15}$$

According to (11), if $p = 0$, the heuristic decomposition does not exist. In this case, we can look for a nilpotent decomposition $B = H + q\,I$. Therefore, if (13) is satisfied for some calculated parameters α and β, then

$$H \text{ nilpotent index } 2 \Rightarrow \begin{cases} \alpha = 2q, \\ \beta = q^2. \end{cases}$$

In any case, we have to calculate α and β from (13). We will see in Section 3 that this is always possible when $B \in \mathcal{M}_2(\mathbb{C})$. When $B \in \mathcal{M}_n(\mathbb{C})$ with $n \geq 3$, this is not always possible. However, when $n \geq 3$, we can look for heuristic solutions calculating α and β in (13) for some matrix elements of $M_{\alpha,\beta}(B)$, and then checking if $M_{\alpha,\beta}(B) = \mathbf{0}$ for these calculated parameters α and β. If this is so, the heuristic decomposition is possible. This method has been coded in MATLAB, and it is available at https://shorturl.at/oHN15 (accessed on 19 March 2024).

2.2. Idempotent Case

Theorem 1. *Consider the polynomial matrix Equation (6), i.e.,*

$$\sum_{k=1}^{m} a_k\, X^k = B.$$

If B admits an idempotent decomposition (7), i.e.,

$$B = p\,N + q\,I,$$

where N is an idempotent matrix, then the solution is

$$X = \lambda\,N + \mu\,I,$$

where μ satisfies the polynomial equation:

$$\sum_{k=1}^{m} a_k\,\mu^k = q,$$

and for each solution of μ, λ satisfies the polynomial equation:

$$p = \sum_{\ell=1}^{m} \lambda^\ell \sum_{k=\ell}^{m} a_k \binom{k}{\ell} \mu^{k-\ell}.$$

Proof. Insert (7) and (9) into (6), and consider that N is idempotent, to obtain

$$\begin{aligned} p\,N + q\,I &= \sum_{k=1}^{m} a_k\,(\lambda\,N + \mu\,I)^k \\ &= \sum_{k=1}^{m} a_k \sum_{\ell=0}^{k} \binom{k}{\ell} \lambda^\ell \mu^{k-\ell} N^\ell \\ &= \sum_{k=1}^{m} a_k \left[\mu^k\,I + N \sum_{\ell=1}^{k} \binom{k}{\ell} \lambda^\ell \mu^{k-\ell} \right] \\ &= I \sum_{k=1}^{m} a_k\,\mu^k + N \sum_{k=1}^{m} a_k \sum_{\ell=1}^{k} \binom{k}{\ell} \lambda^\ell \mu^{k-\ell}. \end{aligned}$$

Therefore,

$$\sum_{k=1}^{m} a_k\, \mu^k = q, \tag{16}$$

$$\sum_{k=1}^{m} a_k \sum_{\ell=1}^{k} \binom{k}{\ell} \lambda^\ell \mu^{k-\ell} = p. \tag{17}$$

Taking into account the discrete Heaviside function, defined as

$$\theta(n) := \begin{cases} 1, & n \geq 0, \\ 0, & n < 0, \end{cases}$$

rewrite (17) as follows:

$$\begin{aligned} p &= \sum_{k=1}^{\infty} a_k\, \theta(m-k) \sum_{\ell=1}^{\infty} \binom{k}{\ell} \lambda^\ell \mu^{k-\ell}\, \theta(k-\ell) \\ &= \sum_{\ell=1}^{\infty} \lambda^\ell \sum_{k=1}^{\infty} a_k \binom{k}{\ell} \mu^{k-\ell}\, \theta(m-k)\, \theta(k-\ell) \\ &= \sum_{\ell=1}^{m} \lambda^\ell \sum_{k=\ell}^{m} a_k \binom{k}{\ell} \mu^{k-\ell}. \end{aligned} \tag{18}$$

□

Remark 1. *Notice that from (16) we have a maximum of m different solutions for μ. According to (18), for each solution of μ, we have a maximum of m solutions for λ. Therefore, we have a maximum of m^2 different pairs (λ, μ), and, according to (9), a maximum of m^2 solutions for X.*

Example 1. *Consider the matrix*

$$B = \begin{pmatrix} 2 & 2 & 2 \\ 2 & 2 & 2 \\ 2 & 2 & 2 \end{pmatrix}.$$

If each matrix element of B denotes the number of paths in one or two steps in a graph, calculate the adjacency matrix R of the graph.

According to Chapter 2 in [13], the adjacency matrix R satisfies the quadratic matrix equation

$$R^2 + R = B.$$

The MATLAB 2023b code developed to find heuristic decompositions yields $p = -1$, $q = 1$, and

$$N = \frac{1}{3} \begin{pmatrix} 2 & -1 & -1 \\ -1 & 2 & -1 \\ -1 & -1 & 2 \end{pmatrix},$$

so that $B = p\,N + q\,I$, and N is an idempotent matrix. Applying the method described in Theorem 1, we obtain four different solutions:

$$\begin{aligned} R_1 &= (-1)\begin{pmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{pmatrix}, \quad R_2 = \frac{2}{3}\begin{pmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{pmatrix}, \\ R_3 &= \begin{pmatrix} 0 & 1 & 1 \\ 1 & 0 & 1 \\ 1 & 1 & 0 \end{pmatrix}, \quad R_4 = -\frac{1}{3}\begin{pmatrix} 5 & 2 & 2 \\ 2 & 5 & 2 \\ 2 & 2 & 5 \end{pmatrix}. \end{aligned}$$

Since the adjacency matrix can only contain 0 or 1 as matrix elements, the solution is $R = R_3$. It is worth noting that we obtain the same set of solutions $R_1, \dots, R_4$ by using both the diagonalization and interpolation methods. However, when performing 100 tests, the diagonalization method is ≈3500 times slower on average than the idempotent method, and the interpolation method is ≈1000 times slower on average than the idempotent method.

Example 2. *Calculate the square root matrix of the following matrix:*

$$B = \begin{pmatrix} 4 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix}.$$

The MATLAB code developed to find heuristic decompositions yields $p = -4$, $q = 4$, and

$$N = \begin{pmatrix} 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix},$$

so that $B = pN + qI$, and N is an idempotent matrix. Applying the method described in Theorem 1, we obtain two different solutions:

$$X = \begin{pmatrix} \pm 2 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix}. \tag{19}$$

It is worth noting that we also obtain the above solution (19) using the diagonalization method. However, if we apply the interpolation method to this problem, we will obtain a "division by zero" error. It is worth noting that none of the methods are able to obtain the following infinite sets of solutions:

$$X \in \left\{ \begin{pmatrix} \pm 2 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & a & 0 \end{pmatrix}, \begin{pmatrix} \pm 2 & 0 & 0 \\ 0 & 0 & a \\ 0 & 0 & 0 \end{pmatrix} : \quad a \in \mathbb{C} \right\}.$$

2.3. Involutive Case

Theorem 2. *Consider the polynomial matrix Equation (6), i.e.,*

$$\sum_{k=1}^{m} a_k \, X^k = B.$$

If B admits an involutive decomposition (7), i.e.,

$$B = p\, N + q\, I,$$

where N is an involutive matrix, then the solution is

$$X = \lambda_{ij} \, N + \mu_{ij} \, I,$$

where

$$\lambda_{ij} = \frac{r_i^+ - r_j^-}{2}, \quad \mu_{ij} = \frac{r_i^+ + r_j^-}{2},$$

being $r_j^\pm$, $j = 1, \dots, m$ the m roots of the polynomial:

$$h_\pm(x) = \sum_{k=1}^{m} a_k \, x^k - q \mp p.$$

Proof. Insert (7) and (9) into (6), and consider that N is involutive, to obtain

$$\begin{aligned} & p\,N + q\,I \\ = & \sum_{k=1}^{m} a_k\,(\lambda\,N + \mu\,I)^k \\ = & \sum_{k=1}^{m} a_k \sum_{\ell=0}^{k} \binom{k}{\ell} \lambda^{\ell} \mu^{k-\ell} N^{\ell} \\ = & \sum_{k=1}^{m} a_k \left[I \sum_{\ell=0}^{\lfloor k/2 \rfloor} \binom{k}{2\ell} \lambda^{2\ell} \mu^{k-2\ell} + N \sum_{\ell=0}^{\lfloor (k-1)/2 \rfloor} \binom{k}{2\ell+1} \lambda^{2\ell+1} \mu^{k-2\ell-1} \right]. \end{aligned}$$

Therefore,

$$q = \sum_{k=1}^{m} a_k \sum_{\ell=0}^{\lfloor k/2 \rfloor} \binom{k}{2\ell} \lambda^{2\ell} \mu^{k-2\ell}, \tag{20}$$

$$p = \sum_{k=1}^{m} a_k \sum_{\ell=0}^{\lfloor (k-1)/2 \rfloor} \binom{k}{2\ell+1} \lambda^{2\ell+1} \mu^{k-2\ell-1}. \tag{21}$$

Sum up (20) and (21) to obtain

$$\begin{aligned} q + p &= \sum_{k=1}^{m} a_k \sum_{\ell=0}^{k} \binom{k}{\ell} \lambda^{\ell} \mu^{k-\ell} \\ &= \sum_{k=1}^{m} a_k (\mu + \lambda)^k. \end{aligned} \tag{22}$$

Also, subtracting (21) from (20), we have

$$\begin{aligned} q - p &= \sum_{k=1}^{m} a_k \sum_{\ell=0}^{k} \binom{k}{\ell} (-\lambda)^{\ell} \mu^{k-\ell} \\ &= \sum_{k=1}^{m} a_k (\mu - \lambda)^k. \end{aligned} \tag{23}$$

Now, denote as $r_j^{\pm}$, $j = 1, \ldots, m$ the m roots of the polynomial

$$h_{\pm}(x) = \sum_{k=1}^{m} a_k\,x^k - q \mp p. \tag{24}$$

According to (22) and (23), we have

$$\lambda_{ij} = \frac{r_i^{+} - r_j^{-}}{2}, \quad \mu_{ij} = \frac{r_i^{+} + r_j^{-}}{2}, \quad i, j = 1, \ldots, m.$$

Hence we have a maximum of m^2 different solutions. □

Example 3. *Consider the matrix*

$$B = \begin{pmatrix} 1 & 0 & 1 \\ 0 & 2 & 0 \\ 1 & 0 & 1 \end{pmatrix}.$$

If each matrix element of B denotes the number of paths in one or two steps in a graph, calculate the adjacency matrix R of the graph.

The adjacency matrix R satisfies the quadratic matrix equation

$$R^2 + R = B.$$

The MATLAB code developed to find heuristic decompositions yields $p = 1$, $q = 1$, and

$$N = \begin{pmatrix} 0 & 0 & 1 \\ 0 & 1 & 0 \\ 1 & 0 & 0 \end{pmatrix},$$

so that $B = pN + qI$, and N is an involutive matrix. Applying the method described in Theorem 2, we obtain four different solutions:

$$\begin{aligned} R_1 &= \begin{pmatrix} -1 & 0 & -1 \\ 0 & -2 & 0 \\ -1 & 0 & -1 \end{pmatrix}, \quad R_2 = -\frac{1}{2}\begin{pmatrix} 3 & 0 & 1 \\ 0 & 4 & 0 \\ 1 & 0 & 3 \end{pmatrix}, \\ R_3 &= \frac{1}{2}\begin{pmatrix} 1 & 0 & 1 \\ 0 & 2 & 0 \\ 1 & 0 & 1 \end{pmatrix}, \quad R_4 = \begin{pmatrix} 0 & 0 & 1 \\ 0 & 1 & 0 \\ 1 & 0 & 0 \end{pmatrix}. \end{aligned}$$

However, the adjacency matrix can only contain 0 or 1 as matrix elements, therefore the solution is $R = R_4$. It is worth noting that we obtain the same set of solutions $R_1, \ldots, R_4$ by using both the diagonalization and interpolation methods. However, performing 100 tests, the diagonalization method is ≈3900 times slower on average than the involutive method, and the interpolation method is ≈1100 times slower on average than the involutive method.

2.4. Nilpotent Case

Theorem 3. *Consider the polynomial matrix Equation (6), i.e.,*

$$\sum_{k=1}^{m} a_k\, X^k = B.$$

If B admits a nilpotent decomposition (8), i.e.,

$$B = H + q\, I,$$

where H is an nilpotent matrix with index 2*, then the solution is*

$$X = \lambda\, H + \mu\, I,$$

where μ satisfies the polynomial equation:

$$\sum_{k=1}^{m} a_k\, \mu^k = q,$$

and for each solution of μ, λ is calculated as:

$$\lambda = \frac{1}{\sum_{k=1}^{m} a_k k\, \mu^{k-1}}.$$

Proof. Insert (8) and (10) into (6), and consider that H is nilpotent with index 2, to obtain

$$\begin{aligned} H + q\,I &= \sum_{k=1}^{m} a_k\,(\lambda\,H + \mu\,I)^k \\ &= \sum_{k=1}^{m} a_k \sum_{\ell=0}^{k} \binom{k}{\ell} \lambda^{\ell} \mu^{k-\ell} H^{\ell} \\ &= \sum_{k=1}^{m} a_k \Big(\mu^k\, I + \lambda k\,\mu^{k-1} H \Big). \end{aligned}$$

Therefore,

$$\sum_{k=1}^{m} a_k\,\mu^k = q, \tag{25}$$

$$\lambda = \frac{1}{\sum_{k=1}^{m} a_k k\,\mu^{k-1}}. \tag{26}$$

□

Remark 2. *Notice that we have* 1 *solution for* λ *for each solution of* μ*, thus according to (9), we have a maximum of m different solutions for X. However, if*

$$\sum_{k=1}^{m} a_k k\,\mu^{k-1} = 0,$$

then, according to (26), λ *does not exist. Therefore, we have to eliminate these cases as solutions of X.*

Example 4. *Solve the quadratic matrix equation:*

$$X^2 - X = B,$$

where

$$B = \begin{pmatrix} 1 & -1 & -1 \\ 1 & 3 & 1 \\ 0 & 0 & 2 \end{pmatrix}.$$

The MATLAB code developed to find heuristic decompositions yields $q = 2$, and

$$H = \begin{pmatrix} -1 & -1 & -1 \\ 1 & 1 & 1 \\ 0 & 0 & 0 \end{pmatrix},$$

so that $B = H + qI$, and H is an nilpotent matrix with index 2. Applying the method described in Theorem 3, we obtain two different solutions:

$$X \in \left\{ \frac{1}{3}\begin{pmatrix} -5 & 1 & 1 \\ -1 & -7 & -1 \\ 0 & 0 & -6 \end{pmatrix}, \frac{1}{3}\begin{pmatrix} 2 & -1 & -1 \\ 1 & 4 & 1 \\ 0 & 0 & 3 \end{pmatrix} \right\}.$$

It is worth noting that the interpolation method provides the same set of solutions for X, but since X is not a diagonalizable matrix, the diagonalization method does not provide a solution. Despite this fact, performing 100 tests, the diagonalization method is ≈500 times slower on average than the nilpotent heuristic method. Also, the interpolation method is ≈1500 times slower on average than the nilpotent method.

Example 5. *Compute a matrix A that satisfies the following equation:*

$$A^3 - 6A^2 + 11A = 6I. \tag{27}$$

Here, matrix $B = 6I$ is a scalar matrix. Since $I^2 = I$ (i.e., the identity matrix is idempotent as well as involutive), B admits infinite ways of idempotent or involutive decompositions, i.e.,

$$B = p\,I + (6-p)I, \quad \forall p \in \mathbb{C}.$$

Also, B admits the following nilpotent decomposition:

$$B = \mathbf{0} + 6I.$$

All the heuristic methods, as well as the diagonalization and interpolation methods, yield the same set of solutions:

$$A \in \{I, 2I, 3I\}. \tag{28}$$

The computational performance of all heuristic methods is very similar. Nevertheless, when performing 100 tests, the diagonalization method is ≈1200 times slower than the heuristic method on average, and the interpolation method is ≈9000 times slower than the heuristic method on average. Note that the solutions given in (28) are "trivial", since they are based on the following factorization of (27):

$$(I - A)(2I - A)(3I - A) = \mathbf{0}.$$

3. Polynomial Matrix Equation of Order 2

The main drawback of the examples presented above is that matrix B has to admit an idempotent, involutive, or nilpotent decomposition. In general, this is not always possible, but in the particular case of $n = 2$, i.e., $X, B \in \mathcal{M}_2(\mathbb{C})$, the decomposition given in (7) or (8) for B is always possible. Next, we consider how to calculate the corresponding decomposition.

3.1. Idempotent Decomposition

For $n = 2$, (13) reads as

$$\underbrace{\begin{pmatrix} c_{11} & c_{12} \\ c_{21} & c_{22} \end{pmatrix}}_{C=B^2} - \alpha \underbrace{\begin{pmatrix} b_{11} & b_{12} \\ b_{21} & b_{22} \end{pmatrix}}_{B} = -\beta \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}. \tag{29}$$

From the first and last matrix elements of (29), we have

$$c_{11} - \alpha\, b_{11} = -\beta = c_{22} - \alpha\, b_{22}. \tag{30}$$

If $b_{11} \neq b_{22}$, we calculate

$$\alpha = \frac{c_{11} - c_{22}}{b_{11} - b_{22}}. \tag{31}$$

However, if $b_{11} = b_{22}$, we can solve for α considering the second matrix element in (29). Therefore, if $b_{11} = b_{22}$ and $b_{12} \neq 0$,

$$c_{12} - \alpha\, b_{12} = 0 \Rightarrow \alpha = \frac{c_{12}}{b_{12}}. \tag{32}$$

Nevertheless, if $b_{11} = b_{22}$ and $b_{12} = 0$, we can solve for α considering the third matrix element in (29). Therefore, if $b_{11} = b_{22}$, $b_{12} = 0$, and $b_{21} \neq 0$,

$$c_{21} - \alpha\, b_{21} = 0 \Rightarrow \alpha = \frac{c_{21}}{b_{21}}. \tag{33}$$

Nonetheless, if $b_{11} = b_{22} \neq 0$, $b_{12} = 0$, and $b_{21} = 0$, matrix B is a scalar matrix, thus the decomposition in (7) is trivial, for instance,

$$B = b_{11} I \Rightarrow \begin{cases} p = 0, \\ q = b_{11}, \\ N = \begin{pmatrix} 0 & 0 \\ 0 & 0 \end{pmatrix}. \end{cases} \tag{34}$$

Finally, if $b_{11} = b_{22} = b_{12} = b_{21} = 0$, the B matrix is the null matrix, thus the decomposition is also trivial,

$$B = \begin{pmatrix} 0 & 0 \\ 0 & 0 \end{pmatrix} \Rightarrow \begin{cases} p = 1, \\ q = 0, \\ N = \begin{pmatrix} 0 & 0 \\ 0 & 0 \end{pmatrix}. \end{cases} \tag{35}$$

Once α is calculated, we have from (30)

$$\beta = \alpha\, b_{11} - c_{11}. \tag{36}$$

Now, according to (14), we have

$$\begin{aligned} \alpha &= 2q + p, \\ \beta &= q(p+q), \end{aligned}$$

that can be solved as

$$\begin{aligned} q &= \frac{\alpha \pm \sqrt{\alpha^2 - 4\beta}}{2}, \\ p &= \alpha - 2q. \end{aligned} \tag{37}$$

Therefore, from (7), the idempotent matrix is

$$N = \frac{1}{p}(B - q\, I). \tag{38}$$

Note that from (37), we may obtain two different decompositions. It will be remembered that if $p = 0$, the decomposition is not possible, since N does not exist, according to (38). In summary, we have proved the following result. If B admits an idempotent decomposition, (i.e., $B = p\, N + q\, I$, where N is an idempotent matrix), we calculate the parameters p and q of such a decomposition from the matrix elements of B. However, if the calculation of p yields $p = 0$, and B is not a scalar nor a null matrix, such a decomposition is not possible.

3.2. Involutive Decomposition

For the involutive case, (13), taking into account (15), reads as

$$\underbrace{\begin{pmatrix} c_{11} & c_{12} \\ c_{21} & c_{22} \end{pmatrix}}_{B^2 = C} - \underbrace{2q}_{\alpha} \underbrace{\begin{pmatrix} b_{11} & b_{12} \\ b_{21} & b_{22} \end{pmatrix}}_{B} = -\underbrace{\left(q^2 - p^2\right)}_{\beta} \underbrace{\begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}}_{I}, \tag{39}$$

thus we can apply the same result as before for α and β (recalling that B is different from a scalar matrix or a null matrix). Thereby,

$$\alpha = \begin{cases} \dfrac{c_{11} - c_{22}}{b_{11} - b_{22}}, & b_{11} \neq b_{22}, \\ \dfrac{c_{12}}{b_{12}}, & b_{11} = b_{22},\, b_{12} \neq 0, \\ \dfrac{c_{21}}{b_{21}}, & b_{11} = b_{22},\, b_{12} = 0,\, b_{21} \neq 0, \end{cases} \tag{40}$$

and

$$\beta = \alpha\, b_{11} - c_{11}. \tag{41}$$

When B is a scalar matrix, the decomposition is trivial, thereby for $t \neq 0$,

$$B = t\begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} \Rightarrow \begin{cases} p = 0, \\ q = t, \\ N = \begin{pmatrix} 0 & 0 \\ 0 & 0 \end{pmatrix}. \end{cases} \tag{42}$$

Also, when B is the null matrix, the decomposition is trivial as well,

$$B = \begin{pmatrix} 0 & 0 \\ 0 & 0 \end{pmatrix} \Rightarrow \begin{cases} p = 1, \\ q = -1, \\ N = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}. \end{cases} \tag{43}$$

According to (15), we have

$$\begin{aligned} q &= \frac{\alpha}{2}, \\ p &= \pm\sqrt{q^2 - \beta}, \end{aligned} \tag{44}$$

and the involutive matrix is given by

$$N = \frac{1}{p}(B - q\,I). \tag{45}$$

Note that from (44) we may obtain two different decompositions. It will be remembered that if $p = 0$, the decomposition is not possible, since N does not exist, according to (45).

Theorem 4. *If a matrix $B \in \mathcal{M}_2(\mathbb{C})$ cannot be decomposed as $B = p\,N + q\,I$, where $p, q \in \mathbb{C}$; $I, N \in \mathcal{M}_2(\mathbb{C})$, being N an idempotent or involutive matrix, and I the identity matrix, then B is of the form*

$$B = a\,I + H, \quad a \in \mathbb{C}, \tag{46}$$

where $H \in \mathcal{M}_2(\mathbb{C})$ is a non-null nilpotent matrix with index 2.

Proof. As previously mentioned, the idempotent or involutive decomposition is not possible when $p = 0$ in (38) or (45), respectively. According to (37) and (44), the latter occurs when $\alpha^2 - 4\beta = 0 \rightarrow \alpha = \pm 2\sqrt{\beta}$. Consequently, according to (29) or (39), B has to satisfy

$$B^2 \mp 2\sqrt{\beta}\,B = -\beta\,I \rightarrow \left(B \mp \sqrt{\beta}\,I\right)^2 = \mathbf{0}. \tag{47}$$

Therefore, $B \mp \sqrt{\beta}\,I$ is a nilpotent matrix with index 2, so that we have the following cases:

- $B \mp \sqrt{\beta}\,I = \mathbf{0}$, thus $B = \pm\sqrt{\beta}I$ is a scalar matrix. However, according to (34) and (42), this case is an exception in both decompositions, where $p = 0$, but the decomposition is possible.
- $B \mp \sqrt{\beta}\,I = H$, where H is a non-null nilpotent matrix with index 2. This is the case given by (46), as we wanted to prove.

□

3.3. Nilpotent Decomposition

According to Theorem 4, if B does not admit an idempotent nor involutive decomposition, then B admits a nilpotent decomposition. In this case, from (8) we have

$$\begin{aligned} & B - q\,I = H \\ \Rightarrow\quad & \underbrace{B^2}_{C} - \underbrace{2q}_{\alpha}\,B = -\underbrace{q^2}_{\beta}\,I. \end{aligned} \tag{48}$$

Note that if H is the null matrix (thus nilpotent with index 2), then $B = q\,I$, i.e., a scalar matrix, so that the decomposition can be obtained by using the idempotent or the involutive decomposition. Consequently, we will consider that H is a non-null nilpotent matrix with index 2. Also, note that H cannot be a scalar matrix since scalar matrices are not nilpotent. In summary, since $B = H + qI$, we determine that B is not a scalar nor a null matrix. Therefore, α and β can be calculated with (40) and (41), and, according to (48), we have

$$q = \frac{\alpha}{2},$$

where

$$\alpha^2 - 4\beta = 0.$$

Remark 3. *Note that this method generalizes the methods given in the existing literature to solve the square root of a matrix [14], i.e., the equation $X^2 = B$. Unlike the square root of a scalar, the square root of a matrix may not exist. For example, we may verify with the proposed heuristic algorithm that the nilpotent matrix with index 2:*

$$B = \begin{pmatrix} 0 & 1 \\ 0 & 0 \end{pmatrix},$$

has no square root, in agreement with [15]. It is worth noting that the diagonalization method does not yield any solutions either. However, the diagonalization method does not clarify if X exists or not. Also, the interpolation method yields a "division by zero" error.

3.4. Singular Case: Infinite Number of Solutions

As was mentioned in Example 5, if B is a scalar matrix, then B admits infinite ways of idempotent or involutive decompositions. However, following the method described above, all these decompositions yield "trivial" solutions. Nevertheless, according to the Cayley–Hamilton theorem, if the order of the matrices equals the degree of the polynomial matrix equation, i.e., $n = m$, there are infinite solutions. Next, we calculate all these solutions for the case $n = m = 2$. For this purpose, consider the quadratic polynomial equation (thus $a_2 \neq 0$):

$$\begin{aligned} & a_1 X + a_2 X^2 = p\,I \\ \Rightarrow\quad & X^2 + \frac{a_1}{a_2} X - \frac{p}{a_2} = \mathbf{0}. \end{aligned} \tag{49}$$

The characteristic polynomial of matrix X is

$$p(\lambda) = |X - \lambda I| = \lambda^2 - \operatorname{tr}(X)\lambda + |X|.$$

According to the Cayley–Hamilton theorem

$$X^2 - \operatorname{tr}(X)X + |X|I = \mathbf{0}. \tag{50}$$

Consider

$$X = \begin{pmatrix} a & b \\ c & d \end{pmatrix},$$

and compare (49) with (50), to obtain

$$\begin{aligned} a+d &= -\frac{a_1}{a_2} := \alpha, & (51)\\ ad-bc &= -\frac{p}{a_2} := \beta. & (52) \end{aligned}$$

Multiply (51) by $-d$ and add the result to (52) to arrive at

$$\begin{aligned} & d^2 - \alpha d + bc + \beta = 0 \\ \Rightarrow \quad & d = \frac{\alpha \pm \sqrt{\alpha^2 - 4(bc+\beta)}}{2}. \end{aligned} \tag{53}$$

According to (51) and (53), we obtain

$$a = d - \alpha = \frac{\alpha \mp \sqrt{\alpha^2 - 4(bc+\beta)}}{2}.$$

Finally, we obtain the following infinite set of solutions:

$$X = \begin{pmatrix} \frac{1}{2}(\alpha \mp \Delta) & b \\ c & \frac{1}{2}(\alpha \pm \Delta) \end{pmatrix}, \quad b, c \in \mathbb{C}, \tag{54}$$

where

$$\begin{aligned} \alpha &= -\frac{a_1}{a_2}, \quad \beta = -\frac{p}{a_2}, & (55)\\ \Delta &= \sqrt{\alpha^2 - 4(bc+\beta)}. & (56) \end{aligned}$$

We can generalize the above result for singular polynomial equations of the form

$$\left(a_0 I + a_1 X + a_2 X^2\right)^k = q\, I, \tag{57}$$

where $k = 1, 2, \ldots$ Notice that (57) can be reduced to (49) as follows

$$a_1 X + a_2 X^2 = \underbrace{\left(q^{1/k} - a_0\right)}_{p} I,$$

where now we have k different solutions for $p = q^{1/k} - a_0$.

Example 6. *Calculate the square root of the identity matrix of order* $n = 2$.

We have to solve the polynomial matrix equation

$$X^2 = B = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}.$$

According to Theorem 1, we obtain two "trivial" solutions:

$$X = \pm \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}. \tag{58}$$

However, according to the method described above, we obtain

$$X = \begin{pmatrix} \pm\sqrt{1-bc} & b \\ c & \mp\sqrt{1-bc} \end{pmatrix}, \quad \forall b, c \in \mathbb{C}. \tag{59}$$

Note that the regular solutions obtained in (58) are not contained in (59). When performing 100 tests, the diagonalization and interpolation methods compute the solution in similar times on average. However, it is worth noting that the diagonalization and interpolation methods do not calculate the singular solutions (59).

Example 7. *Calculate the solutions of the polynomial matrix equation of order $n = 2$*

$$2X^2 + X^4 = 3I. \tag{60}$$

Note that (60) can be rewritten as

$$\left(I + X^2\right)^2 = I + 2X^2 + X^4 = 4I, \tag{61}$$

thus

$$I + X^2 = \pm 2I \ \rightarrow \ X^2 = \begin{cases} I, \\ -3I. \end{cases} \tag{62}$$

Therefore, according to the method described above, the singular solutions of (62) are $\forall b, c \in \mathbb{C}$

$$X = \begin{cases} \begin{pmatrix} \pm\sqrt{1-bc} & b \\ c & \mp\sqrt{1-bc} \end{pmatrix}, \\ \begin{pmatrix} \pm\sqrt{-3-bc} & b \\ c & \mp\sqrt{-3-bc} \end{pmatrix}, \end{cases}$$

and the regular ones are

$$X \in \left\{ \pm\begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}, \pm i\begin{pmatrix} \sqrt{3} & 0 \\ 0 & \sqrt{3} \end{pmatrix} \right\}. \tag{63}$$

Again, the diagonalization and the interpolation methods calculate the regular solutions, but not the singular ones. Moreover, the interpolation method provides a slight computational error in the two last regular solutions of (63). After 10 tests, the computational performance between the diagonalization and the heuristic methods is quite similar on average. Nonetheless, the interpolation method is ≈12 times slower on average than the heuristic method. This is quite significant since the interpolation method does not provide any of the singular solutions, as mentioned before.

3.5. General Case

According to the above Sections, we propose the following procedure to solve a polynomial matrix equation

$$a_1 X + a_2 X^2 + \cdots + a_m X^m = B \tag{64}$$

of order $n = 2$:

- Attempt the decomposition $B = p\,N + q\,I$, where N is an idempotent or an involutive matrix, according to Section 3.1. If the decomposition is successful, apply Theorem 1 (if N is idempotent), or Theorem 2 (if N is involutive), in order to solve (64).
- If the idempotent or the involutive decomposition is not successful, then perform the decomposition $B = H + q\,I$ (being H a nilpotent matrix with index 2) according to Section 3.3. Apply Theorem 3 to solve (64).
- Check if there are singular cases, i.e., a polynomial equation of the form

$$\left(a_0 I + a_1 X + a_2 X^2\right)^k = q\,I.$$

If this is so, we have an infinite number of extra-solutions. This algorithm has been coded in MATLAB, and it is available at https://shorturl.at/oHN15 (accessed on 19 March 2024).

Despite the fact that this algorithm has been developed for polynomial matrix equations with scalar coefficients, i.e., (6), we can use it when this is not the case. To illustrate this, consider the following example.

Example 8. *Solve the following polynomial matrix equation*

$$X^3 - \begin{pmatrix} 2 & -1 \\ 0 & 2 \end{pmatrix} X + \begin{pmatrix} 1 & -1 \\ 0 & 1 \end{pmatrix} = \mathbf{0}. \tag{65}$$

The diagonalization method yields only the following diagonal solutions:

$$X = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}, \begin{pmatrix} \frac{\sqrt{5}-1}{2} & 0 \\ 0 & 1 \end{pmatrix}, \begin{pmatrix} -\frac{\sqrt{5}+1}{2} & 0 \\ 0 & 1 \end{pmatrix}.$$

However, we can obtain other solutions with the heuristic method. Indeed, define

$$B = \begin{pmatrix} 1 & -1 \\ 0 & 1 \end{pmatrix},$$

to rewrite (65) as

$$X^3 - (B + I)X + B = \mathbf{0}, \tag{66}$$

Now, recast (66) as follows

$$\begin{aligned} & X^3 - (B + I)X + B + I = I \\ \Rightarrow \quad & X^3 + (B + I)(I - X) = I \\ \Rightarrow \quad & B + I = \left(I - X^3\right)(I - X)^{-1}. \end{aligned} \tag{67}$$

Now, apply the partial sum of a matrix geometrical series, which is given by

$$S = I + X + X^2 + \cdots + X^n = \left(I - X^{n+1}\right)(I - X)^{-1},$$

thus (67) reduces to

$$B = X + X^2. \tag{68}$$

Applying the heuristic method, we obtain two new non-diagonalizable solutions:

$$X \in \left\{ \begin{pmatrix} -\frac{1+\sqrt{5}}{2} & \frac{\sqrt{5}}{5} \\ 0 & -\frac{1+\sqrt{5}}{2} \end{pmatrix}, \begin{pmatrix} \frac{\sqrt{5}-1}{2} & -\frac{\sqrt{5}}{5} \\ 0 & \frac{\sqrt{5}-1}{2} \end{pmatrix} \right\}. \tag{69}$$

Solving (68) with the interpolation method, we obtain the same set of solutions given in (69).

4. Conclusions

We have derived a heuristic method to solve polynomial matrix equations, i.e., $a_1 X + a_2 X^2 + \cdots + a_m X^m = B$, where the coefficients a_k are scalars and X, B are square matrices of order n, as long as B admits an idempotent, involutive, or nilpotent decomposition. Whenever this decomposition is possible, we have described an algorithm to find it. Moreover, we have proved that this decomposition is always possible when $B \in \mathcal{M}_2(\mathbb{C})$. Also, for square matrices of order $n = 2$, we have described an algorithm that calculates solutions. Further, the algorithm has the capacity to determine the nonexistence of the solution. In addition, when B is a scalar matrix, and $n = 2$, the algorithm computes singular solutions (i.e., infinite sets of solutions), if any exist.

We have compared the proposed heuristic method with other methods found in the existing literature, such as the diagonalization and interpolation methods. It turns out that the heuristic method is usually considerably faster than the diagonalization or the interpolation methods (see Examples 1, 3, and 4). Also, when the diagonalization method fails (see Example 4 and Remark 3), we do not know if the solution does not exist, or the solution is not diagonalizable. Further, the interpolation method sometimes fails with a "division by zero" error (see Example 2 and Remark 3), and we do not know whether the solution exists or not. Consequently, we have shown some examples for which the proposed heuristic method is able to compute solutions, even though the diagonalization or the interpolation methods fail. Moreover, the diagonalization and interpolation methods are not able to compute singular solutions, as the heuristic method does for $n = 2$ (see Examples 6 and 7). The best strength of the diagonalization method is its ability to find solutions of (1) when A_k are not scalar matrices, unlike the interpolation or the heuristic methods. However, Example 8 shows how to compute non-diagonalizable solutions with the proposed heuristic method when some of the coefficients in the polynomial matrix equation are not scalar matrices.

In a future study, the intention is to prove if the proposed algorithm provides all the possible solutions for $n = 2$. Finally, it is worth noting that all the examples have been computed by using a MATLAB code available at https://shorturl.at/oHN15 (accessed on 19 March 2024).

Author Contributions: Conceptualization, J.L.G.-S. and F.S.L.; Methodology, J.L.G.-S. and F.S.L.; Writing—original draft, J.L.G.-S. and F.S.L.; Writing—review and editing, J.L.G.-S. and F.S.L. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Data Availability Statement: Data are contained within the article.

Conflicts of Interest: The authors declare no conflicts of interest.

References

1. Connes, A.; Schwarz, A. Matrix Vieta theorem revisited. *Lett. Math. Phys.* **1997**, *39*, 349–353. [CrossRef]
2. Bini, D.A.; Latouche, G.; Meini, B. Solving matrix polynomial equations arising in queueing problems. *Linear Algebra Its Appl.* **2002**, *340*, 225–244. [CrossRef]
3. Bini, D.A.; Gemignani, L.; Meini, B. Computations with infinite Toeplitz matrices and polynomials. *Linear Algebra Its Appl.* **2002**, *343*, 21–61. [CrossRef]
4. Iannazzo, B. On the Newton method for the matrix p-th root. *SIAM J. Matrix Anal. Appl.* **2006**, *28*, 503–523. [CrossRef]
5. Choudhry, A. Extraction of nth roots of 2 × 2 matrices. *Linear Algebra Its Appl.* **2004**, *387*, 183–192. [CrossRef]
6. Psarrakos, P. On the m-th roots of a complex matrix. *Electron. J. Linear Algebra* **2002**, *9*, 32–41. [CrossRef]
7. Lakić, S. On the computation of the matrix k-th root. *ZAMM-J. Appl. Math. Mech.* **1998**, *78*, 167–172. [CrossRef]
8. Gohberg, I.; Lancaster, P.; Rodman, L. Quadratic matrix polynomials with a parameter. *Adv. Appl. Math.* **1986**, *7*, 253–281. [CrossRef]
9. Gohberg, I.; Lancaster, P.; Rodman, L. *Matrix Polynomials*; Springer: Berlin/Heidelberg, Germany, 2005.
10. Kratz, W.; Stickel, E. Numerical solution of matrix polynomial equations by Newton's method. *IMA J. Numer. Anal.* **1987**, *7*, 355–369. [CrossRef]
11. Fuchs, D.; Schwarz, A. matrix Vieta theorem. *arXiv* **1994**, arXiv:math/9410207.
12. Petraki, D.; Samaras, N. Solving the n-th degree polynomial matrix equation. *J. Interdiscip. Math.* **2021**, *24*, 1079–1092. [CrossRef]
13. Biggs, N. *Algebraic Graph Theory*; Cambridge University Press: Cambridge, UK, 1993; p. 67.
14. Deadman, E.; Higham, N.J.; Ralha, R. Blocked Schur algorithms for computing the matrix square root. In Proceedings of the International Workshop on Applied Parallel Computing, New Orleans, LA, USA, 26 February 2012; Springer: Berlin/Heidelberg, Germany, 2012; pp. 171–182.
15. Björck, Å.; Hammarling, S. A Schur method for the square root of a matrix. *Linear Algebra Its Appl.* **1983**, *52*, 127–140. [CrossRef]

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.