# EXACT LOW-ORDER POLYNOMIAL EXPRESSIONS TO COMPUTE THE KOLMOGOROFF-NAGUMO MEAN IN THE AFFINE SYMPLECTIC GROUP OF OPTICAL TRANSFERENCE MATRICES

SIMONE FIORI*‡ AND STILIAN PRIFTI†

**Abstract.** The current contribution presents exact third-order polynomial expressions of matrix functions that arise in the computation of the Kolmogoroff-Nagumo mean of a set of optical transference matrices, that belong to the affine symplectic group ASp(4).

**Key words.** Affine symplectic group of matrices; Exact low-order polynomial representation; Kolmogoroff-Nagumo mean; Linear optical transference.

## 1. Introduction.

The group of real symplectic matrices $\mathrm{Sp}(2n)$, with $n \in \mathbb{N}$ and $n \geq 1$, is an instance of quadratic matrix group [14]. Symplectic matrices find wide applications in sciences and engineering. Noteworthy examples of applications are found in vibration analysis [3], in optimal control [2, 15], in electromagnetism [16] and in computational optics [7].

The present contribution is motivated by an application in linear optics: When a ray of light passes through a lens (such as the lens in a telescope) the ray is refracted and the change in direction may be described by a symplectic matrix.

The recent contribution [8] deals with the computation of the Kolmogoroff-Nagumo mean of a set of *centered* linear optical systems. Each of such systems is described by a $4 \times 4$ real symplectic matrix, namely, by an element of the Lie group $\mathrm{Sp}(4)$, termed *astigmatic* matrix. Given as set of $N$ astigmatic matrices $S_n \in \mathrm{Sp}(4)$, its Kolmogoroff-Nagumo mean is denoted by:

$$\bar{S}^{\varphi}_{\mathrm{KN}} := \varphi\left(\frac{1}{N}\sum_{n=1}^{N}\varphi^{-1}(S_n)\right), \tag{1.1}$$

where $\varphi^{-1} : \mathrm{Sp}(4) \to \mathfrak{sp}(4)$, with $\mathfrak{sp}(4)$ denoting the Lie algebra associated with the Lie group $\mathrm{Sp}(4)$ formed by all the $4 \times 4$ Hamiltonian matrices, and $\varphi : \mathfrak{sp}(4) \to \mathrm{Sp}(4)$ denotes its inverse. (In general, the function $\varphi^{-1}$ is defined only locally in an open set $U_\varphi \subset \mathrm{Sp}(4)$.) The rationale of the Kolmogoroff-Nagumo mean is that while the symplectic group $\mathrm{Sp}(4)$ is a curved manifold, hence it is impossible to take an arithmetic mean over such a space directly, its Lie algebra is a linear space, where an arithmetic mean is well-defined. Therefore, the symplectic matrices $S_n$ are first mapped to the Lie algebra through a suitable map $\varphi^{-1}$ and the result of arithmetic averaging is brought back to the Lie group through its inverse $\varphi$. In fact, the actual calculations are taken in the Lie algebra. This is the approach usually adopted, for instance, in the numerical integration of ODEs on Lie groups (see, for instance, [4, 5]).

In the contribution [8], the following pairs of maps were considered:

- The map $\varphi^{-1} = \log$, namely, the matrix logarithm. Its inverse is $\varphi = \exp$, namely, the

*Dipartimento di Ingegneria dell'Informazione, Università Politecnica delle Marche, Via Brecce Bianche, I-60131 Ancona (Italy)

†School of Mechanical Engineering, Università Politecnica delle Marche, Via Brecce Bianche, I-60131 Ancona (Italy)

‡Corresponding author. eMail: s.fiori@univpm.it. Cite this paper as: **S. Fiori and S. Prifti, "Exact low-order polynomial expressions to compute the Kolmogoroff-Nagumo mean in the affine symplectic group of optical transference matrices", *Linear and Multilinear Algebra*, Vol. 65, No. 4, pp. 840 — 856, 2017**

matrix exponential. Such maps are defined as matrix-to-matrix series, namely:

$$\exp(X) := I + \sum_{k=1}^{\infty} \frac{X^k}{k!}, \tag{1.2}$$

$$\log(X) := -\sum_{k=1}^{\infty} \frac{(I-X)^k}{k}. \tag{1.3}$$

The series for the logarithmic map converges as long as $\|X - I\| < 1$. Given a symplectic matrix $S \in \mathrm{Sp}(4)$, its logarithm $\log(S)$ is defined by the matrix-power series (1.3) and is hence cumbersome to evaluate; likewise, given an Hamiltonian matrix $H \in \mathfrak{sp}(4)$, its exponential $\exp(H)$ is cumbersome to compute by the series (1.2).

- The map $\varphi^{-1} = \mathrm{cay}$, namely, the matrix Cayley transform, defined as

$$\mathrm{cay}(X) := (I + X)(I - X)^{-1}, \tag{1.4}$$

  well-defined if $X - I \in \mathrm{GL}(n) := \{Y \in \mathbb{R}^{n\times n} \mid \det(Y) \neq 0\}$. Since the Cayley transform is self-inverse, the inverse map is $\varphi = \mathrm{cay}$.

In particular, the contribution [8] recalled how an analytic function $\varphi$ of a $4 \times 4$ real-valued matrix may be calculated *exactly* as a third-order polynomial, namely:

$$\varphi(H) = \varphi_3 H^3 + \varphi_2 H^2 + \varphi_1 H + \varphi_0 I, \tag{1.5}$$

$$\varphi^{-1}(S) = \varphi_3^{(-1)} S^3 + \varphi_2^{(-1)} S^2 + \varphi_1^{(-1)} S + \varphi_0^{(-1)} I, \tag{1.6}$$

and gave explicit formulas for the coefficients $\varphi_0, \varphi_1, \varphi_2, \varphi_3$ and $\varphi_0^{(-1)}, \varphi_1^{(-1)}, \varphi_2^{(-1)}, \varphi_3^{(-1)}$ as functions of the eigenvalues of the matrices $H$ and $S$, respectively, both for the exp/log maps-pair and for the cay/cay maps-pair. Since the eigenvalue structure of an Hamiltonian matrix and of a symplectic matrix presents several possible cases, several cases of interest were classified and solved for, separately.

The Kolmogoroff-Nagumo mean may be calculated for every matrix Lie group and it coincides with the output of the first step, with a specific starting point (namely, the group identity), of a general averaging algorithm developed in [10]. In order to compare the Kolmogoroff-Nagumo mean with the general algorithm presented in [10], it is also necessary to identify the pair $(\varphi, \varphi^{-1})$ with a pseudo-retraction/pseudo-lifting maps pair. A further generalization on this line is that the pair of conjugate functions $(\varphi, \varphi^{-1})$ may be replaced with a pair $(\varphi, \psi)$, such that the composition $\varphi \circ \psi$ behaves as an *approximate identity* [9], with the aim of lightening the computational burden associated with the Kolmogoroff-Nagumo mean. It is worth underlying that not every Kolmogoroff-Nagumo-like averaging rule is an instance of the the Lie-group averaging scheme (1.1). This is the case, for example, of the *resolvent average* $S^{\mathrm{res}} := \varphi\left(\sum_{n=1}^{N} \varphi^{-1}(S_n)/N\right)$ with $\varphi(H) := H^{-1} - I$ introduced in [1]. Since $\varphi$ does not map an $\mathfrak{sp}(4)$-matrix into an $\mathrm{Sp}(4)$-matrix, such an averaging rule does not result to be an instance of the Lie-group-type Kolmogoroff-Nagumo averaging rule (1.1). The Kolmogoroff-Nagumo mean as well as taking averages over the real symplectic group are special cases of a more general problem, namely, averaging over curved manifolds (see, for instance, the study [6]).

The above setting is of limited scope in the sense that it was developed for centered optical systems only. In the presence of *decentered* optical systems, the above setting is no longer suitable. In fact, a general optical system cannot be represented only in terms of an astigmatic matrix, but it is necessary to call for an augmented optical transference matrix of the form:

$$T = \begin{bmatrix} S & \delta \\ 0 & 1 \end{bmatrix}, \tag{1.7}$$

where $S \in \mathrm{Sp}(4)$ is the astigmatic submatrix and $\delta \in \mathbb{R}^4$ represents a shift of the optical system with respect to its center. The resulting $5 \times 5$ matrix belongs to the *affine symplectic group* $\mathrm{ASp}(4)$, that is a Lie group under standard matrix multiplication/inverse, whose Lie algebra is denoted as $\mathfrak{asp}(4)$. For a review of the (affine) symplectic group, see, e.g., [11]. A generic element $L$ of the algebra $\mathfrak{asp}(4)$ presents the following structure:

$$L = \begin{bmatrix} H & v \\ 0 & 0 \end{bmatrix}, \tag{1.8}$$

with $H \in \mathfrak{sp}(4)$ and $v \in \mathbb{R}^4$. The exponential-logarithm-based Kolmogoroff-Nagumo mean of a set of $N$ optical system transference matrices $T_n$ reads:

$$\bar{T}_{\mathrm{KN}}^{\exp} := \exp\left(\frac{1}{N}\sum_{n=1}^{N}\log(T_n)\right), \tag{1.9}$$

while the Cayley-Cayley-based Kolmogoroff-Nagumo mean of a set of $N$ optical system transference matrices $T_n$ reads

$$\bar{T}_{\mathrm{KN}}^{\mathrm{cay}} := -\mathrm{cay}\left(\frac{1}{N}\sum_{n=1}^{N}\mathrm{cay}(-T_n)\right). \tag{1.10}$$

The reason for the change of sign in the definition (1.10) is that, from the general structure of an optical transference matrix (1.7), it follows immediately that the matrix $I - T$ is not invertible, while the matrix $I + T$ is surely invertible.

The maps $\varphi$ and $\varphi^{-1}$ considered in the previous definition may be evaluated on the basis of exponential/logarithmic/Cayley maps applied to symplectic and Hamiltonian matrices. In fact, it holds that:

$$\exp(L) = \begin{bmatrix} \exp(H) & M_1(H)v \\ 0 & 1 \end{bmatrix},\ \log(T) = \begin{bmatrix} \log(S) & M_2(S)\delta \\ 0 & 0 \end{bmatrix}, \tag{1.11}$$

$$\mathrm{cay}(L) = \begin{bmatrix} \mathrm{cay}(H) & M_3(H)v \\ 0 & 1 \end{bmatrix},\ \mathrm{cay}(-T) = \begin{bmatrix} \mathrm{cay}(-S) & (\mathrm{cay}(-S)+I)\delta/2 \\ 0 & 0 \end{bmatrix}, \tag{1.12}$$

where the matrix functions $M_1$, $M_2$ and $M_3$ are defined as:

$$M_1 : \mathfrak{sp}(4) \to \mathrm{Sp}(4), \quad M_1(H) := (\exp(H) - I)H^{-1}, \tag{1.13}$$

$$M_2 : \mathrm{Sp}(4) \to \mathfrak{sp}(4), \quad M_2(S) := \log(S)(S - I)^{-1}, \tag{1.14}$$

$$M_3 : \mathfrak{sp}(4) \to \mathrm{Sp}(4), \quad M_3(H) := (\mathrm{cay}(H) - I)H^{-1} = 2H(I - H)^{-1}H^{-1}. \tag{1.15}$$

The function $M_1$ is well-defined only if $H \in \mathrm{GL}(4)$, the matrix function $M_3$ is well-defined only if $H \in \mathrm{GL}(4)$ and $H - I \in \mathrm{GL}(4)$, while the function $M_2$ is well-defined only if $S - I \in \mathrm{GL}(4)$. Since the matrix-functions $H^{-1}$ and $(I-H)^{-1}$ are analytic within the existence field of the function $M_3$, and since analytic matrix-functions of the same matrix-variable commute, it holds that $2H(I - H)^{-1}H^{-1} = 2HH^{-1}(I - H)^{-1} = 2(I - H)^{-1}$, hence:

$$M_3(H) = 2(I - H)^{-1}. \tag{1.16}$$

The functions $M_1$, $M_2$ and $M_3$ differ from the functions exp, log and cay, hence, in the present work, we propose to compute them explicitly as third-order polynomials. For the evaluation of a term like $\mathrm{cay}(-T)$ there is no special function needed, because the evaluation of the quantity $\mathrm{cay}(-S)$ was already covered in the previous contribution [8].

**2. Third-order polynomials to compute $4\times 4$-matrix functions.** The method presented in the current paper to compute a non-linear matrix function by means of a low-order polynomial is a special case of a general method based on the Lagrange generalized polynomials recalled, for example, in the Appendix A of the book [12].

An analytic function of a $4 \times 4$ matrix may be evaluated through an exact third-order polynomial formula whose coefficients depend on the eigenvalues of the matrix. Let $X$ denote a $4\times 4$ real-valued matrix. The characteristic polynomial associated with the matrix $X$ is $p_X(z) := \det(zI - X)$, where $z \in \mathbb{C}$ denotes a scalar complex-valued variable, which is a monic polynomial of degree 4:

$$p_X(z) = z^4 + p_3 z^3 + p_2 z^2 z + p_1 z + p_0, \tag{2.1}$$

where $p_3, p_2, p_1, p_0 \in \mathbb{R}$ are the scalar coefficients of the characteristic polynomial (that coincide, up to a change of sign, with the principal invariants associated to the matrix $X$ and may be computed through Newton's identities [17]) and the roots of the polynomial (2.1) are the eigenvalues of the matrix $X$. Namely, denoting with $\lambda_X$ an eigenvalue of the matrix $X$, it holds that

$$p_X(\lambda_X) = 0. \tag{2.2}$$

The *Cayley-Hamilton theorem* states that each matrix satisfies its own characteristic equation. In the present context, the Cayley-Hamilton theorem casts as:

$$P_X(X) := X^4 + p_3 X^3 + p_2 X^2 + p_1 X + p_0 I = 0. \tag{2.3}$$

By definition, any analytic function may be expanded as a polynomial. Moreover, to any matrix-to-matrix function $F$ may be associated a scalar-to-scalar function by replacing the matrix argument with a scalar argument. Hence, the matrix-to-matrix analytic function $F(X)$ may be thought of as a polynomial $f(z)$ in the variable $z \in \mathbb{C}$. The polynomial $f(z)$ may be written as:

$$f(z) = q(z) p_X(z) + r(z), \tag{2.4}$$

where $q(z)$ is the quotient polynomial and $r(z)$ is the remainder polynomial of degree (at most) 3, namely:

$$r(z) = f_3 z^3 + f_2 z^2 + f_1 z + f_0, \tag{2.5}$$

where $f_3, f_2, f_1, f_0 \in \mathbb{R}$ are the scalar coefficients of the polynomial $r(z)$. By setting $z = \lambda_X$ in the equation (2.4), thanks to the identity (2.2), it is readily seen that:

$$f(\lambda_X) = r(\lambda_X). \tag{2.6}$$

The corresponding relationship in the matrix variable $X$ reads:

$$F(X) = Q(X) P_X(X) + R(X). \tag{2.7}$$

Since, by the Cayley-Hamilton theorem (2.3), it holds that $P_X(X) = 0$, from the equation (2.7) it follows that $F(X) = R(X)$. Therefore, the matrix function $F(X)$ may be evaluated, *exactly*, as a third-order matrix polynomial, namely, as:

$$F(X) = f_3 X^3 + f_2 X^2 + f_1 X + f_0 I, \tag{2.8}$$

where the scalar coefficients of the polynomial expression of the function $F(X)$ are to be sought.

The key point in the computation of the coefficients $f_3, f_2, f_1, f_0$ in the polynomial expression (2.8) is that they are also the coefficients of the polynomial expression of the associated scalar-to-scalar function $f(z)$. Therefore, one may exploit the identity (2.6) to calculate such coefficients.

For symmetry reasons, it is known that the eigenvalues of an Hamiltonian and of a symplectic $4 \times 4$ matrix may appear with multiplicity 1, 2 or 4.

Given a matrix-to-matrix analytic function $F$ (and its associated scalar counterpart $f$), given the four eigenvalues of the matrix $X$, namely $\lambda_X^a, \lambda_X^b, \lambda_X^c, \lambda_X^d$ and supposing that they are *all distinct*, the coefficients $f_3, f_2, f_1, f_0$ are found by solving the following linear system:

$$\left\{ \begin{array}{rcl} (\lambda_X^a)^3 f_3 + (\lambda_X^a)^2 f_2 + \lambda_X^a f_1 + f_0 &=& f(\lambda_X^a), \\ (\lambda_X^b)^3 f_3 + (\lambda_X^b)^2 f_2 + \lambda_X^b f_1 + f_0 &=& f(\lambda_X^b), \\ (\lambda_X^c)^3 f_3 + (\lambda_X^c)^2 f_2 + \lambda_X^c f_1 + f_0 &=& f(\lambda_X^c), \\ (\lambda_X^d)^3 f_3 + (\lambda_X^d)^2 f_2 + \lambda_X^d f_1 + f_0 &=& f(\lambda_X^d). \end{array} \right. \tag{2.9}$$

In matrix format, the above linear system casts as

$$\underbrace{\begin{bmatrix} 1 & \lambda_X^a & (\lambda_X^a)^2 & (\lambda_X^a)^3 \\ 1 & \lambda_X^b & (\lambda_X^b)^2 & (\lambda_X^b)^3 \\ 1 & \lambda_X^c & (\lambda_X^c)^2 & (\lambda_X^c)^3 \\ 1 & \lambda_X^d & (\lambda_X^d)^2 & (\lambda_X^d)^3 \end{bmatrix}}_{V} \begin{bmatrix} f_0 \\ f_1 \\ f_2 \\ f_3 \end{bmatrix} = \underbrace{\begin{bmatrix} f(\lambda_X^a) \\ f(\lambda_X^b) \\ f(\lambda_X^c) \\ f(\lambda_X^d) \end{bmatrix}}_{g}. \tag{2.10}$$

The matrix $V$ of the coefficients is square Vandermonde. The coefficients of the polynomial (2.8) are found by inverting the matrix $V$ and then by computing $V^{-1}g$. The inversion of a square Vandermonde matrix is facilitated by its LU decomposition, that is given explicitly for the $4 \times 4$ case, e.g., in the contribution [13]. The explicit solution of the above Vandermonde-type system is:

$$\left\{ \begin{array}{rcl} f_0 &=& \frac{f(\lambda_X^b)\,\lambda_X^a\,\lambda_X^c\,\lambda_X^d}{(\lambda_X^a-\lambda_X^b)\,(\lambda_X^b-\lambda_X^c)\,(\lambda_X^b-\lambda_X^d)} - \frac{f(\lambda_X^a)\,\lambda_X^b\,\lambda_X^c\,\lambda_X^d}{(\lambda_X^a-\lambda_X^b)\,(\lambda_X^a-\lambda_X^c)\,(\lambda_X^a-\lambda_X^d)} \\ && - \frac{f(\lambda_X^c)\,\lambda_X^a\,\lambda_X^b\,\lambda_X^d}{(\lambda_X^a-\lambda_X^c)\,(\lambda_X^b-\lambda_X^c)\,(\lambda_X^c-\lambda_X^d)} + \frac{f(\lambda_X^d)\,\lambda_X^a\,\lambda_X^b\,\lambda_X^c}{(\lambda_X^a-\lambda_X^d)\,(\lambda_X^b-\lambda_X^d)\,(\lambda_X^c-\lambda_X^d)} \\ f_1 &=& \frac{f(\lambda_X^a)\,(\lambda_X^b\,\lambda_X^c+\lambda_X^b\,\lambda_X^d+\lambda_X^c\,\lambda_X^d)}{(\lambda_X^a-\lambda_X^b)\,(\lambda_X^a-\lambda_X^c)\,(\lambda_X^a-\lambda_X^d)} - \frac{f(\lambda_X^b)\,(\lambda_X^a\,\lambda_X^c+\lambda_X^a\,\lambda_X^d+\lambda_X^c\,\lambda_X^d)}{(\lambda_X^a-\lambda_X^b)\,(\lambda_X^b-\lambda_X^c)\,(\lambda_X^b-\lambda_X^d)} \\ && + \frac{f(\lambda_X^c)\,(\lambda_X^a\,\lambda_X^b+\lambda_X^a\,\lambda_X^d+\lambda_X^b\,\lambda_X^d)}{(\lambda_X^a-\lambda_X^c)\,(\lambda_X^b-\lambda_X^c)\,(\lambda_X^c-\lambda_X^d)} - \frac{f(\lambda_X^d)\,(\lambda_X^a\,\lambda_X^b+\lambda_X^a\,\lambda_X^c+\lambda_X^b\,\lambda_X^c)}{(\lambda_X^a-\lambda_X^d)\,(\lambda_X^b-\lambda_X^d)\,(\lambda_X^c-\lambda_X^d)}, \\ f_2 &=& \frac{f(\lambda_X^b)\,(\lambda_X^a+\lambda_X^c+\lambda_X^d)}{(\lambda_X^a-\lambda_X^b)\,(\lambda_X^b-\lambda_X^c)\,(\lambda_X^b-\lambda_X^d)} - \frac{f(\lambda_X^a)\,(\lambda_X^b+\lambda_X^c+\lambda_X^d)}{(\lambda_X^a-\lambda_X^b)\,(\lambda_X^a-\lambda_X^c)\,(\lambda_X^a-\lambda_X^d)} \\ && - \frac{f(\lambda_X^c)\,(\lambda_X^a+\lambda_X^b+\lambda_X^d)}{(\lambda_X^a-\lambda_X^c)\,(\lambda_X^b-\lambda_X^c)\,(\lambda_X^c-\lambda_X^d)} + \frac{f(\lambda_X^d)\,(\lambda_X^a+\lambda_X^b+\lambda_X^c)}{(\lambda_X^a-\lambda_X^d)\,(\lambda_X^b-\lambda_X^d)\,(\lambda_X^c-\lambda_X^d)}, \\ f_3 &=& \frac{f(\lambda_X^a)}{(\lambda_X^a-\lambda_X^b)\,(\lambda_X^a-\lambda_X^c)\,(\lambda_X^a-\lambda_X^d)} - \frac{f(\lambda_X^b)}{(\lambda_X^a-\lambda_X^b)\,(\lambda_X^b-\lambda_X^c)\,(\lambda_X^b-\lambda_X^d)} \\ && + \frac{f(\lambda_X^c)}{(\lambda_X^a-\lambda_X^c)\,(\lambda_X^b-\lambda_X^c)\,(\lambda_X^c-\lambda_X^d)} - \frac{f(\lambda_X^d)}{(\lambda_X^a-\lambda_X^d)\,(\lambda_X^b-\lambda_X^d)\,(\lambda_X^c-\lambda_X^d)}. \end{array} \right. \tag{2.11}$$

Whenever the eigenvalues of the matrix $X$ are not distinct, repeating the equations (2.5) and (2.6) leads to an unsolvable (under-determined) linear system, of little use. In this case, it is necessary to evaluate the polynomial expression of the derivatives of the function $f$. For example, starting over from equation (2.4) and computing the analytic derivative with respect to the variable $z$ of both sides, gives:

$$f'(z) = q'(z)p_X(z) + q(z)p'_X(z) + r'(z). \tag{2.12}$$

Upon evaluating such expression in an eigenvalue $\lambda_X$, supposed of algebraic multiplicity equal to 2, the first term on the right-hand side vanishes because $p_X(\lambda_X) = 0$ and the second term on the right-hand side vanishes because $p'_X(\lambda_X) = 0$ as well. Therefore, for an eigenvalue of multiplicity 2, it holds that

$$f'(\lambda_X) = r'(\lambda_X) = 3(\lambda_X)^2 f_3 + 2\lambda_X f_2 + f_1. \tag{2.13}$$

Hence, for a matrix $X$ with two distinct eigenvalues $\lambda_X^a, \lambda_X^b$, both of multiplicity 2, the resolving linear system reads:

$$\begin{cases} (\lambda_X^a)^3 f_3 + (\lambda_X^a)^2 f_2 + \lambda_X^a f_1 + f_0 &= f(\lambda_X^a), \\ 3(\lambda_X^a)^2 f_3 + 2\lambda_X^a f_2 + f_1 &= f'(\lambda_X^a), \\ (\lambda_X^b)^3 f_3 + (\lambda_X^b)^2 f_2 + \lambda_X^b f_1 + f_0 &= f(\lambda_X^b), \\ 3(\lambda_X^b)^2 f_3 + 2\lambda_X^b f_2 + f_1 &= f'(\lambda_X^b). \end{cases} \tag{2.14}$$

In matrix format, the above linear system casts as

$$\begin{bmatrix} 1 & \lambda_X^a & (\lambda_X^a)^2 & (\lambda_X^a)^3 \\ 0 & 1 & 2\lambda_X^a & 3(\lambda_X^a)^2 \\ 1 & \lambda_X^b & (\lambda_X^b)^2 & (\lambda_X^b)^3 \\ 0 & 1 & 2\lambda_X^b & 3(\lambda_X^b)^2 \end{bmatrix} \begin{bmatrix} f_0 \\ f_1 \\ f_2 \\ f_3 \end{bmatrix} = \begin{bmatrix} f(\lambda_X^a) \\ f'(\lambda_X^a) \\ f(\lambda_X^b) \\ f'(\lambda_X^b) \end{bmatrix} \tag{2.15}$$

In the mixed case that there are two distinct eigenvalues $\lambda_X^a$ and $\lambda_X^b$ with multiplicity 1 and an eigenvalue $\lambda_X^c$ with multiplicity 2, the resolving linear system reads:

$$\begin{cases} (\lambda_X^a)^3 f_3 + (\lambda_X^a)^2 f_2 + \lambda_X^a f_1 + f_0 &= f(\lambda_X^a), \\ (\lambda_X^b)^3 f_3 + (\lambda_X^b)^2 f_2 + \lambda_X^b f_1 + f_0 &= f(\lambda_X^b), \\ (\lambda_X^c)^3 f_3 + (\lambda_X^c)^2 f_2 + \lambda_X^c f_1 + f_0 &= f(\lambda_X^c), \\ 3(\lambda_X^c)^2 f_3 + 2\lambda_X^c f_2 + f_1 &= f'(\lambda_X^c). \end{cases} \tag{2.16}$$

In matrix format, the above linear system casts as

$$\begin{bmatrix} 1 & \lambda_X^a & (\lambda_X^a)^2 & (\lambda_X^a)^3 \\ 1 & \lambda_X^b & (\lambda_X^b)^2 & (\lambda_X^b)^3 \\ 1 & \lambda_X^c & (\lambda_X^c)^2 & (\lambda_X^c)^3 \\ 0 & 1 & 2\lambda_X^c & 3(\lambda_X^c)^2 \end{bmatrix} \begin{bmatrix} f_0 \\ f_1 \\ f_2 \\ f_3 \end{bmatrix} = \begin{bmatrix} f(\lambda_X^a) \\ f(\lambda_X^b) \\ f(\lambda_X^c) \\ f'(\lambda_X^c) \end{bmatrix} \tag{2.17}$$

Likewise, for an eigenvalue $\lambda_X$ of algebraic multiplicity 4, it holds that $f''(\lambda_X) = r''(\lambda_X)$, namely:

$$f''(\lambda_X) = 6\lambda_X f_3 + 2f_2, \tag{2.18}$$

and that $f'''(\lambda_X) = r'''(\lambda_X)$, namely:

$$f'''(\lambda_X) = 6f_3. \tag{2.19}$$

Hence, for a matrix $X$ with an eigenvalues $\lambda_X$ of multiplicity 4, the resolving linear system reads:

$$\begin{cases} (\lambda_X)^3 f_3 + (\lambda_X)^2 f_2 + \lambda_X f_1 + f_0 &= f(\lambda_X), \\ 3(\lambda_X)^2 f_3 + 2\lambda_X f_2 + f_1 &= f'(\lambda_X), \\ 6\lambda_X f_3 + 2f_2 &= f''(\lambda_X), \\ 6f_3 &= f'''(\lambda_X), \end{cases} \tag{2.20}$$

whose explicit solution is:

$$\begin{cases} f_0 = f(\lambda_X) - \lambda_X f'(\lambda_X) + \frac{1}{2}\lambda_X^2 f''(\lambda_X) - \frac{1}{6}\lambda_X^3 f'''(\lambda), \\ f_1 = f'(\lambda_X) - \lambda_X f''(\lambda_X) + \frac{1}{2}\lambda_X^2 f'''(\lambda_X), \\ f_2 = \frac{1}{2} f''(\lambda_X) - \frac{1}{2}\lambda_X f'''(\lambda_X), \\ f_3 = \frac{1}{6} f'''(\lambda_X). \end{cases} \tag{2.21}$$

The polynomial expressions to compute the special functions $M_1(H)$, $M_2(S)$ and $M_3(H)$, along with the associated scalar generating functions to compute the coefficients of the polynomials, are:

$$M_1(H) = a_3 H^3 + a_2 H^2 + a_1 H + a_0 I,\ m_1(z) := \frac{\exp(z) - 1}{z}, \tag{2.22}$$

$$M_2(S) = b_3 S^3 + b_2 S^2 + b_1 S + b_0 I,\ m_2(z) := \frac{\log(z)}{z - 1}, \tag{2.23}$$

$$M_3(H) = c_3 H^3 + c_2 H^2 + c_1 H + c_0 I,\ m_3(z) := \frac{2}{1 - z}. \tag{2.24}$$

While the functions $m_1$ and $m_2$ may be prolonged by continuity in their singularity, in fact:

$$\lim_{z\to 0} \frac{\exp(z) - 1}{z} = 1, \quad \lim_{z \to 1} \frac{\log(z)}{z - 1} = 1,$$

the function $m_3$ has an essential singularity in $z = 1$.

The computation of the three sets of four coefficients is carried out and discussed in the following Sections for some cases of interest. The Appendix A provides some details about the MATLAB© implementation of the evaluation of the polynomial expressions for the three matrix functions $M_1(H)$, $M_2(S)$ and $M_3(H)$ as well as for the functions $\log(T)$, $\exp(L)$ and $\mathrm{cay}(L)$.

**3. Computation of the coefficients of the polynomial expression of the functions $M_1(H)$ and $M_3(H)$.** The matrix functions $M_1$ and $M_3$, as defined in (1.13) and (1.14), apply to a $4 \times 4$ Hamiltonian matrix. In the Section 2, it was pointed out that the coefficients of the polynomial expression of the quantities $M_1(H)$ and of $M_3(H)$ depend on the eigenvalues of the matrix $H$ in argument. The four eigenvalues of a $4 \times 4$ Hamiltonian matrix come in complex-valued antipodal pairs, namely $\lambda^a, -\lambda^a, \lambda^b, -\lambda^b \in \mathbb{C}$ and are computed by the expression [8]:

$$\pm \frac{1}{2}\sqrt{\mathrm{tr}(H^2) \pm \sqrt{\mathrm{tr}^2(H^2) - 16 \det(H)}}. \tag{3.1}$$

In order to determine the coefficients of the polynomial expressions, it is necessary to distinguish between the cases that the complex-valued numbers $\lambda^a, \lambda^b$ are distinct or coincident. In the following sections, we cover three cases, namely:

- Four distinct eigenvalues of algebraic multiplicity 1;
- Two distinct eigenvalues, case $(\lambda_H, -\lambda_H, \lambda_H, -\lambda_H)$, with $\lambda_H \in \mathbb{C} - \{0\}$;
- Three distinct eigenvalues, case $(\lambda_H, -\lambda_H, 0, 0)$, with $\lambda_H \in \mathbb{C} - \{0\}$.

**3.1. Coefficients of the polynomial expression of the matrix function $M_1(H)$.** The presents subsection covers the computation of the coefficients $a_3$, $a_2$, $a_1$ and $a_0$ pertaining to the polynomial expression of the function $M_1(H)$ (2.22). It pays to recall the analytic derivatives of the associated scalar function $m_1(z)$, namely:

$$m_1'(z) = \frac{e^z}{z} - \frac{e^z - 1}{z^2}, \tag{3.2}$$

$$m_1''(z) = \frac{2\,e^z + z^2\,e^z - 2\,z\,e^z - 2}{z^3}, \tag{3.3}$$

$$m_1'''(z) = \frac{z^3\,e^z - 3\,z^2\,e^z - 6\,e^z + 6\,z\,e^z + 6}{z^4}. \tag{3.4}$$

**3.1.1. Case I: Distinct eigenvalues of algebraic multiplicity 1.** The four distinct eigenvalues are denote by $\lambda_H^a, -\lambda_H^a, \lambda_H^b, -\lambda_H^b \in \mathbb{C}$. In order for such eigenvalues to be distinct from each other, it must hold that $\lambda_H^a \neq 0$, $\lambda_H^b \neq 0$ and $(\lambda_H^a)^2 \neq (\lambda_H^b)^2$. In the current case, the resolving system is square Vandermonde and the explicit solutions (2.11) may be used to compute the coefficients $a_3$, $a_2$, $a_1$ and $a_0$. The explicit solution for the coefficients reads:

$$\begin{cases} a_0 = \frac{(\lambda_H^a)^3 \sinh\left(\lambda_H^b\right) - (\lambda_H^b)^3 \sinh(\lambda_H^a)}{\lambda_H^a \lambda_H^b \left((\lambda_H^a)^2 - (\lambda_H^b)^2\right)}, \\ a_1 = \left(\frac{\lambda_H^a}{\lambda_H^b}\right)^2 \frac{\cosh\left(\lambda_H^b\right) - 1}{(\lambda_H^a)^2 - (\lambda_H^b)^2} - \left(\frac{\lambda_H^b}{\lambda_H^a}\right)^2 \frac{\cosh(\lambda_H^a) - 1}{(\lambda_H^a)^2 - (\lambda_H^b)^2}, \\ a_2 = -\frac{\lambda_H^a \sinh\left(\lambda_H^b\right) - \lambda_H^b \sinh(\lambda_H^a)}{\lambda_H^a \lambda_H^b \left((\lambda_H^a)^2 - (\lambda_H^b)^2\right)}, \\ a_3 = \frac{\cosh(\lambda_H^a) - 1}{(\lambda_H^a)^2 \left((\lambda_H^a)^2 - (\lambda_H^b)^2\right)} - \frac{\cosh\left(\lambda_H^b\right) - 1}{(\lambda_H^b)^2 \left((\lambda_H^a)^2 - (\lambda_H^b)^2\right)}. \end{cases} \tag{3.5}$$

**3.1.2. Case II: Two distinct eigenvalues, case $(\lambda_H, -\lambda_H, \lambda_H, -\lambda_H)$, with $\lambda_H \in \mathbb{C} - \{0\}$.** The explicit solution for the coefficients of the third-order polynomial reads:

$$\begin{cases} a_0 = \frac{3}{2} \frac{\sinh(\lambda_H)}{\lambda_H} - \frac{1}{2} \cosh(\lambda_H), \\ a_1 = \frac{2(\cosh(\lambda_H) - 1)}{\lambda_H^2} - \frac{\sinh(\lambda_H)}{2\lambda_H}, \\ a_2 = \frac{\cosh(\lambda_H)}{2\lambda_H^2} - \frac{\sinh(\lambda_H)}{2\lambda_H^3}, \\ a_3 = \frac{1 - \cosh(\lambda_H)}{\lambda_H^4} + \frac{\sinh(\lambda_H)}{2\lambda_H^3}. \end{cases} \tag{3.6}$$

**3.1.3. Case III: Three distinct eigenvalues, case $(\lambda_H, -\lambda_H, 0, 0)$, with $\lambda_H \in \mathbb{C} - \{0\}$.** Whenever the matrix $H$ possesses the eigenvalue structure $(\lambda_H, -\lambda_H, 0, 0)$, with $\lambda_H \in \mathbb{C} - \{0\}$, the matrix function $M_1(H)$ is not well-defined (in fact, the function $m_1'(z)$ cannot be evaluated in $z = 0$).

**3.2. Coefficients of the polynomial expression of the matrix function $M_3(H)$.** The presents subsection covers the computation of the coefficients $c_3$, $c_2$, $c_1$ and $c_0$ pertaining to the polynomial expression of the function $M_3(H)$ (2.24). In order to complete the necessary calculations, it pays to recall the expressions of the derivatives of the auxiliary function $m_3(z)$, namely:

$$m_3'(z) = \frac{2}{(z-1)^2}, \tag{3.7}$$

$$m_3''(z) = -\frac{4}{(z-1)^3}, \tag{3.8}$$

$$m_3'''(z) = \frac{12}{(z-1)^4}. \tag{3.9}$$

In the present case, the functions $m_3$, $m_3'$, $m_3''$, $m_3'''$ are rational, therefore, according to the expressions (2.11), the coefficients $c_3$, $c_2$, $c_1$ and $c_0$ are rational functions of the eigenvalues.

**3.2.1. Case I: Distinct eigenvalues of algebraic multiplicity 1.** The four distinct eigenvalues are denoted again by $\lambda_H^a, -\lambda_H^a, \lambda_H^b, -\lambda_H^b \in \mathbb{C}$. In order for the eigenvalues to be distinct from each other, it must hold that $\lambda_H^a \neq 1$, $\lambda_H^b \neq 1$ and $(\lambda_H^a)^2 \neq (\lambda_H^b)^2$. In this case, the explicit solutions (2.11) may be used to compute the coefficients $c_3$, $c_2$, $c_1$ and $c_0$. The coefficients of the third-order polynomial read:

$$\begin{cases} c_0 = c_1 = \frac{2\,(\lambda_H^b)^2}{\left((\lambda_H^a)^2-1\right)\left((\lambda_H^a)^2-(\lambda_H^b)^2\right)} - \frac{2\,(\lambda_H^a)^2}{\left((\lambda_H^b)^2-1\right)\left((\lambda_H^a)^2-(\lambda_H^b)^2\right)}, \\ c_2 = c_3 = \frac{2}{\left((\lambda_H^b)^2-1\right)\left((\lambda_H^a)^2-(\lambda_H^b)^2\right)} - \frac{2}{\left((\lambda_H^a)^2-1\right)\left((\lambda_H^a)^2-(\lambda_H^b)^2\right)}. \end{cases} \tag{3.10}$$

In the present case, the polynomial expression of the matrix function $M_3(H)$ simplifies in:

$$M_3(H) = (c_2 H^2 + c_0 I)(H + I),$$

because the coefficients $c_3$, $c_2$, $c_1$, $c_0$ result to be pairwise identical.

**3.2.2. Case II: Two distinct eigenvalues, case $(\lambda_H, -\lambda_H, \lambda_H, -\lambda_H)$, with $\lambda_H \in \mathbb{C} - \{0\}$.** In order for the eigenvalues to result of multiplicity 2, it must hold that $\lambda_H \neq \pm 1$. The explicit solution for the coefficients of the third-order polynomial was computed as:

$$\begin{cases} c_0 = -\frac{\lambda_H+2}{2\,(\lambda_H+1)^2} - \frac{3\,\lambda_H-2}{2\,(\lambda_H-1)^2}, \\ c_1 = \frac{2\,\lambda_H+3}{2\,\lambda_H\,(\lambda_H+1)^2} - \frac{4\,\lambda_H-3}{2\,\lambda_H\,(\lambda_H-1)^2}, \\ c_2 = \frac{1}{2\,\lambda_H\,(\lambda_H-1)^2} - \frac{1}{2\,\lambda_H\,(\lambda_H+1)^2}, \\ c_3 = \frac{2\,\lambda_H-1}{2\,\lambda_H^3\,(\lambda_H-1)^2} - \frac{1}{2\,\lambda_H^3\,(\lambda_H+1)^2}. \end{cases} \tag{3.11}$$

**3.2.3. Case III: Three distinct eigenvalues, case $(\lambda_H, -\lambda_H, 0, 0)$, with $\lambda_H \in \mathbb{C} - \{0\}$.** It must hold that $\lambda_H \neq \pm 1$. The explicit solution for the coefficients of the third-order polynomial was found to be:

$$\begin{cases} c_0 = c_1 = 2, \\ c_2 = -\frac{2\left(\lambda_H^2+\lambda_H-1\right)}{\lambda_H^2\left(\lambda_H^2-1\right)}, \\ c_3 = -\frac{2\left(\lambda_H^3-\lambda_H+1\right)}{\lambda_H^3\left(\lambda_H^2-1\right)}. \end{cases} \tag{3.12}$$

It is worth underlying that the corresponding case, whenever using the function $M_1$, is undefined. Therefore, the Cayley transform based averaging can be applied to a wider variety of cases.

**4. Computation of the coefficients of the polynomial expression of the function $M_2(S)$.** Given a $4 \times 4$ real-valued symplectic matrix $S$, the coefficients of its characteristic polynomial $p_S(z)$ (2.1) may be computed through the Newton's identities:

$$\begin{cases} p_3 = p_1 = -\mathrm{tr}(S), \\ p_2 = \frac{1}{2}(\mathrm{tr}^2(S) - \mathrm{tr}(S^2)), \\ p_0 = 1, \end{cases} \tag{4.1}$$

where the noticeable property $\det(S) = 1$ was used in the expression of $p_0$. Hence, the eigenvalues of a $4 \times 4$ symplectic matrix are given by the formula:

$$\frac{\zeta \pm \sqrt{\zeta^2 - 4}}{2}, \text{ where } \zeta = \frac{-p_1 \pm \sqrt{p_1^2 - 4p_2 + 8}}{2}. \tag{4.2}$$

A consequence of the above formulas is that if $\lambda$ denotes an eigenvalue of a symplectic matrix, then $\bar{\lambda}$ and $\lambda^{-1}$ must be eigenvalues as well (the over-bar denotes complex conjugation).

When all eigenvalues are of multiplicity 1, the following cases may occur:

1. Case of complex-valued quartet: There are four distinct complex-valued roots $\lambda \in \mathbb{C}$, $\bar{\lambda}$, $\lambda^{-1}$, $\bar{\lambda}^{-1}$.
2. Case of a real-valued duet and a unimodular duet: There are two distinct real-valued roots $\lambda_1 \in \mathbb{R}$, $\lambda_1^{-1}$, and two distinct complex-valued, unimodular roots $\lambda_2 \in \mathbb{C}$, $\bar{\lambda}_2$, with $|\lambda_2| = 1$ (the symbol $|\cdot|$ denotes the modulus of a complex-valued number).
3. Case of two real-valued duets: There are four distinct real-valued roots $\lambda_1 \in \mathbb{R}$, $\lambda_1^{-1}$, $\lambda_2 \in \mathbb{R}$, $\lambda_2^{-1}$.
4. Case of two unimodular duets: There are four distinct complex-valued roots $\lambda_1 \in \mathbb{C}$, $\bar{\lambda}_1$, with $|\lambda_1| = 1$, and $\lambda_2 \in \mathbb{C}$, $\bar{\lambda}_2$, with $|\lambda_2| = 1$.

The eigenvalues may come with multiplicity 2 and with multiplicity 4 as well. There are six possible cases that cover the occurrence of eigenvalues with multiplicity 2:

1. Case of eigenvalues $(x, x^{-1}, x, x^{-1})$ with $x \in \mathbb{R} - \{0, \pm 1\}$.
2. Case of eigenvalues $(x, x^{-1}, 1, 1)$ with $x \in \mathbb{R} - \{0, \pm 1\}$.
3. Case of eigenvalues $(x, x^{-1}, -1, -1)$ with $x \in \mathbb{R} - \{0, \pm 1\}$.
4. Case of eigenvalues $(e^{\iota\theta}, e^{-\iota\theta}, 1, 1)$ with $\theta \in \mathbb{R} - \pi\mathbb{Z}$ (where the symbol $\iota$ denotes the imaginary unit, namely, $\iota^2 = -1$).
5. Case of eigenvalues $(e^{\iota\theta}, e^{-\iota\theta}, -1, -1)$ with $\theta \in \mathbb{R} - \pi\mathbb{Z}$.
6. Case of eigenvalues $(1, 1, -1, -1)$.

There are two possible cases that cover the occurrence of eigenvalues with multiplicity 4:

1. Case of eigenvalues $(1, 1, 1, 1)$.
2. Case of eigenvalues $(-1, -1, -1, -1)$.

The presents Section covers the computation of the coefficients $b_3$, $b_2$, $b_1$ and $b_0$ pertaining to the polynomial expression of the function $M_2(S)$ (2.23). For the convenience of the reader, the expressions of the derivatives of the function $m_2(z)$, are given as follows:

$$m_2'(z) = -\frac{z\,(\log(z) - 1) + 1}{z\,(z-1)^2}, \tag{4.3}$$

$$m_2''(z) = \frac{4\,z + z^2\,(2\,\log(z) - 3) - 1}{z^2\,(z-1)^3}, \tag{4.4}$$

$$m_2'''(z) = -\frac{18\,z^2 - 9\,z + z^3\,(6\,\log(z) - 11) + 2}{z^3\,(z-1)^4}. \tag{4.5}$$

In particular, the following cases are given full consideration:

- Case of complex-valued quartet $\lambda$, $\bar{\lambda}$, $\lambda^{-1}$, $\bar{\lambda}^{-1}$;
- Case of real-valued quartet $(x, x^{-1}, y, y^{-1})$ with $x, y \in \mathbb{R} - \{0, \pm 1\}$;
- Case of purely imaginary quartet $(e^{\iota\theta}, e^{-\iota\theta}, e^{\iota\psi}, e^{-\iota\psi},)$ with $\theta, \psi \in (0, 2\pi)$, $\theta \neq \psi$, $\theta + \psi \neq 2\pi$.

**4.1. Case I: Distinct eigenvalues of algebraic multiplicity 1, complex-valued quartet.** The four distinct eigenvalues of a symplectic matrix $S$ are denoted as $\lambda_S, \bar{\lambda}_S, \lambda_S^{-1}, \bar{\lambda}_S^{-1}$. The explicit solutions (2.11) may be used to compute the coefficients $b_3$, $b_2$, $b_1$ and $b_0$. The explicit solution for the coefficients of the third-order polynomial was computed as:

$$\begin{cases} b_0 = \Im\left\{\frac{\log(\lambda_S)\bar{\lambda}_S(\lambda_S^5-1)}{(\lambda_S-1)^2(\lambda_S+1)\Im\{\bar{\lambda}_S(\lambda_S^2+1)\}}\right\}, \\ b_1 = \Im\left\{\frac{\log(\lambda_S)((1-\lambda_S^5)(\bar{\lambda}_S^2+1)-|\lambda_S|^2(\lambda_S^3-1))}{(\lambda_S-1)^2(\lambda_S+1)\Im\{\bar{\lambda}_S(\lambda_S^2+1)\}}\right\}, \\ b_2 = \Im\left\{\frac{\log(\bar{\lambda}_S)(\lambda_S(1-\bar{\lambda}_S^5)+\bar{\lambda}_S(1-\bar{\lambda}_S^3)(\lambda_S^2+1))}{(\bar{\lambda}_S-1)^2(\bar{\lambda}_S+1)\Im\{\bar{\lambda}_S(\lambda_S^2+1)\}}\right\}, \\ b_3 = \Im\left\{\frac{|\lambda_S|^2\log(\lambda_S)(1-\lambda_S^3)}{(\lambda_S-1)^2(\lambda_S+1)\Im\{\bar{\lambda}_S(\lambda_S^2+1)\}}\right\}, \end{cases} \tag{4.6}$$

where the symbol $\Im\{\cdot\}$ denotes the imaginary part of a complex number.

**4.2. Case II: Distinct eigenvalues of algebraic multiplicity 1, purely imaginary duets $(e^{\iota\theta}, e^{-\iota\theta}, e^{\iota\psi}, e^{-\iota\psi})$ with $\theta, \psi \in (0, 2\pi)$, $\theta \neq \psi$, $\theta + \psi \neq 2\pi$.** With the considered configuration of the four eigenvalues, the explicit solution for the coefficients of the third-order polynomial becomes:

$$\begin{cases} b_0 = \frac{\psi\sin\left(\frac{5\psi}{2}\right)}{2\sin\left(\frac{\psi}{2}\right)(\sin(2\psi)-2\cos(\theta)\sin(\psi))} + \frac{\theta\sin\left(\frac{5\theta}{2}\right)}{2\sin\left(\frac{\theta}{2}\right)(\sin(2\theta)-2\cos(\psi)\sin(\theta))}, \\ b_1 = -\frac{\theta\left(\sin\left(\frac{3\theta}{2}\right)+2\sin\left(\frac{5\theta}{2}\right)\cos(\psi)\right)}{2\sin\left(\frac{\theta}{2}\right)(\sin(2\theta)-2\cos(\psi)\sin(\theta))} - \frac{\psi\left(\sin\left(\frac{3\psi}{2}\right)+2\sin\left(\frac{5\psi}{2}\right)\cos(\theta)\right)}{2\sin\left(\frac{\psi}{2}\right)(\sin(2\psi)-2\cos(\theta)\sin(\psi))}, \\ b_2 = -\frac{\theta\left(\sin\left(\frac{5\theta}{2}\right)+2\sin\left(\frac{3\theta}{2}\right)\cos(\psi)\right)}{2\sin\left(\frac{\theta}{2}\right)(\sin(2\theta)-2\cos(\psi)\sin(\theta))} - \frac{\psi\left(\sin\left(\frac{5\psi}{2}\right)+2\sin\left(\frac{3\psi}{2}\right)\cos(\theta)\right)}{2\sin\left(\frac{\psi}{2}\right)(\sin(2\psi)-2\cos(\theta)\sin(\psi))}, \\ b_3 = -\frac{\psi\sin\left(\frac{3\psi}{2}\right)}{2\sin\left(\frac{\psi}{2}\right)(\sin(2\psi)-2\cos(\theta)\sin(\psi))} - \frac{\theta\sin\left(\frac{3\theta}{2}\right)}{2\sin\left(\frac{\theta}{2}\right)(\sin(2\theta)-2\cos(\psi)\sin(\theta))}. \end{cases} \tag{4.7}$$

Note that the condition $\theta, \psi \in (0, 2\pi)$ ensures that $\sin\left(\frac{\theta}{2}\right) \neq 0$ and $\sin\left(\frac{\psi}{2}\right) \neq 0$, and that the conditions $\theta \neq \psi$ and $\theta+\psi \neq 2\pi$ ensure that $\sin(2\theta)-2\cos(\psi)\sin(\theta) \neq 0$ and $\sin(2\psi)-2\cos(\theta)\sin(\psi) \neq 0$.

**4.3. Case III: Distinct eigenvalues of algebraic multiplicity 1, purely real-valued duets $(x, x^{-1}, y, y^{-1})$ with $x, y \in \mathbb{R}-\{0, \pm 1\}$, $x \neq y$, $x \neq y^{-1}$.** With the considered configuration of the four eigenvalues, the explicit solution for the coefficients of the third-order polynomial becomes:

$$\begin{cases} b_0 = -\frac{y\log(x)(x^4+1)}{y(x^4-1)+(x-x^3)(y^2+1)} - \frac{x\log(y)(y^4+1)}{x(y^4-1)+(y-y^3)(x^2+1)}, \\ b_1 = \frac{\log(x)(y^2-1)(x^3(xy^2+y+x)+xy+y^2+1)}{D} - \frac{\log(y)(x^2-1)(y^3(yx^2+x+y)+xy+x^2+1)}{D}, \\ b_2 = -\frac{\log(x)(y(x^4+1)+(y^2+1)(x^3+x))}{y(x^4-1)+(x-x^3)(y^2+1)} - \frac{\log(y)(x(y^4+1)+(x^2+1)(y^3+y))}{x(y^4-1)+(y-y^3)(x^2+1)}, \\ b_3 = \frac{xy\log(x)(x^2+1)(y^2-1)}{D} - \frac{xy\log(y)(x^2-1)(y^2+1)}{D}, \end{cases} \tag{4.8}$$

with $D := (1-x^2)(y^2-1)(x(y^2+1)-y(x^2+1))$.

**5. Conclusions.** The present contribution completes the calculations carried out in the previous paper [8]. Specifically, the present paper covered the calculations necessary to compute some special matrix functions arising in the evaluation of the Kolmogoroff-Nagumo mean of a set of full $5 \times 5$ optical transference matrices.

An alternative method to evaluate such special matrix functions, with special reference to the function $M_2(S)$, to be explored in the future, concerns a type of singular-value decomposition for symplectic matrices explained in [18].

The Kolmogoroff-Nagumo mean is of interest in a number of applications involving different kinds of matrix groups, such as the space of special Euclidean matrices [6]. It will be interesting to apply the powerful method for matrix function computation based on generalized Lagrange polynomials [12] in future research efforts.

REFERENCES


[1] H.H. Bauschke, S.M. Moffat and X. Wang, "The resolvent average for positive semidefinite matrices", *Linear Algebra and its Applications*, Vol. 432, No. 7, pp. 1757 – 1771, 2010

[2] M. Chu, N. Del Buono, F. Diele, T. Politi and S. Ragni, "On the semigroup of standard symplectic matrices and its applications", *Linear Algebra and its Applications*, Vol. 389, pp. 215 – 225, September 2004

[3] E.K.-W. Chua, T.-M. Hwang, W.-W. Lin and C.-T. Wu, "Vibration of fast trains, palindromic eigenvalue problems and structure-preserving doubling algorithms", *Journal of Computational and Applied Mathematics*, Vol. 219, No. 237 – 252, 2008

[4] N. Del Buono, L. Lopez and T. Politi, "Computation of functions of Hamiltonian and skew-symmetric matrices", *Mathematics and Computers in Simulation*, Vol. 79, No. 4, pp. 1284 – 1297, December 2008

[5] F. Diele, L. Lopez and T. Politi, "One step semi-explicit methods based on the Cayley transform for solving isospectral flows", *Journal of Computational and Applied Mathematics*, Vol. 89, No. 2, pp. 219 – 223, March 1998

[6] X. Duan, H. Sun and L. Peng, "Riemannian means on special Euclidean group and unipotent matrices group", *Scientific World Journal*, 2013: 292787, October 2013 (doi:10.1155/2013/292787)

[7] S. Fiori, "Averaging over the Lie group of optical systems transference matrices", *Frontiers of Electrical and Electronic Engineering in China*, Vol. 6, No. 1, pp. 137 – 145, March 2011

[8] S. Fiori, "Kolmogoroff-Nagumo mean over the affine symplectic group of matrices", *Applied Mathematics and Computation*, Vol. 266, pp. 820 – 837, September 2015

[9] S. Fiori, T. Kaneko and T. Tanaka, "Mixed maps for learning a Kolmogoroff-Nagumo-type average element on the compact Stiefel manifold", in *Proceedings of the 2014 International Conference on Acoustics, Speech and Signal Processing* (ICASSP 2014, Florence (Italy), May 4-9, 2014), pp. 4518 – 4522, 2014

[10] S. Fiori and T. Tanaka, "An algorithm to compute averages on matrix Lie groups", *IEEE Transactions on Signal Processing*, Vol. 57, No. 12, pp. 4734 – 4743, December 2009

[11] M.A. de Gosson, *Symplectic Methods in Harmonic Analysis and in Mathematical Physics*, Chapter 2 ("The Symplectic Group"), Pseudo-Differential Operators 7, Springer Basel AG, 2011

[12] V. Lakshmikantham and D. Trigiante, *Theory of Difference Equations – Numerical Methods and Applications*, Second Edition, CRC Press – Taylor & Francis group (ISBN: 9780824708030), 2002

[13] H. Oruç and G.M. Phillips, "Explicit factorization of the Vandermonde matrix", *Linear Algebra and its Applications*, Vol. 315, pp. 113 – 123, 2000

[14] L. Lopez, C. Mastroserio and T. Politi, "Newton-type methods for solving nonlinear equations on quadratic matrix groups", *Journal of Computational and Applied Mathematics*, Vol. 115, No.s 1-2, pp. 357 – 368, March 2000

[15] C. Mehl, V. Mehrmann, A.C.M. Ran and L. Rodman, "Perturbation analysis of Lagrangian invariant subspaces of symplectic matrices", *Linear and Multilinear Algebra*, Vol. 57, No. 2, pp. 141 – 184, 2009

[16] W. Sha, X.L. Wu, Z.X. Huang and M.S. Chen, "Maxwell's equation, symplectic matrix and grid", *Progress in Electromagnetics Research B*, Vol. 8, 115 – 127, 2008

[17] A.J.M. Spencer, *Continuum Mechanics*, Longman Mathematical Texts Series, Addison-Wesley Longman, 1980

[18] H. Xu, "An SVD-like matrix decomposition and its applications", *Linear Algebra and its Applications*, Vol. 368, pp. 1 – 24, July 2003


**Appendix A. MATLAB implementation.** In the present Appendix, we show the MATLAB© implementation of three cases of interest. Specifically, the Section A.1 details the functions that afford the evaluation of the functions $M_1(H)$, $M_2(S)$ and $M_3(H)$ in the hypothesis that *all the eigenvalues of the matrix in argument are distinct.* The Section A.2 suggests optimized codes to evaluate the functions $\log(T)$, $\exp(L)$ and $\mathrm{cay}(L)$.

**A.1. Evaluation of the matrix-functions $M_1$, $M_2$ and $M_3$.** The MATLAB© function that implements the numerical evaluation of the matrix-function $M_1(H)$ is shown below.

```
% Evaluation of the function M_1(H) corresponding to the case that all the
% eigenvalues of the matrix H are distinct
function P = FunctionM1(H)
% Calculation of lambda_a (z1) and lambda_b (z2)
z1 = sqrt( trace(H^2) + sqrt( trace(H^2)^2 - 16*det(H) ) )/2;
z2 = sqrt( trace(H^2) - sqrt( trace(H^2)^2 - 16*det(H) ) )/2;
% Calculation of the coefficients of the polynomial expression
a0 = (sinh(z2)*z1^3 - sinh(z1)*z2^3)/(z1*z2*(z1^2-z2^2));
a1 = z2^2*(cosh(z1)-1)/(z1^2*(z2^2-z1^2)) - z1^2*(cosh(z2)-1)/(z2^2*(z2^2-z1^2));
a2 = (z2*sinh(z1) - z1*sinh(z2))/(z1*z2*(z1^2-z2^2));
a3 = (cosh(z1)-1)/(z1^2*(z1^2-z2^2)) - (cosh(z2)-1)/(z2^2*(z1^2-z2^2));
% Calculation of the polynomial
P = a3*H^3 + a2*H^2 + a1*H + a0*eye(4);
```

The MATLAB© function that implements the numerical evaluation of the matrix-function $M_2(S)$ is shown below.

```
% Evaluation of the function M_2(S) corresponding to the case that all the
% eigenvalues of the matrix S are distinct of the type (lambda, 1/lambda,
% lambda*, 1/lambda*)
function P = FunctionM2(S)
% Calculation of lambda (L)
p2 = ( trace(S)^2 - trace(S^2) )/2;
p1 = -trace(S);
x = ( -p1 + sqrt( p1^2 - 4*p2 + 8) )/2;
z = ( x + sqrt( x^2 - 4 ) )/2;
% Calculation of the coefficients of the polynomial expression
zc = conj(z);
b0 = imag(log(z)*zc*(z^5-1)/((z-1)^2*(z+1)*imag(zc*(z^2+1))));
b1 = imag(log(z)*((1-z^5)*(zc^2+1)-z*zc*(z^3-1))/((z-1)^2*(z+1)*imag(zc*(z^2+1))));
b2 = imag(log(zc)*(z*(1-zc^5)+zc*(1-zc^3)*(z^2+1))/((zc-1)^2*(zc+1)*imag(zc*(z^2+1))));
b3 = imag(z*zc*log(z)*(1-z^3)/((z-1)^2*(z+1)*imag(zc*(z^2+1))));
% Calculation of the polynomial
P = b3*S^3 + b2*S^2 + b1*S + b0*eye(4);
```

The MATLAB© function that implements the numerical evaluation of the matrix-function $M_3(H)$ is shown below.

```
% Evaluation of the function M_3(H) corresponding to the case that all the
% eigenvalues of the matrix H are distinct
function P = FunctionM3(H)
% Calculation of lambda_a (z1) and lambda_b (z2)
z1 = sqrt( trace(H^2) + sqrt( trace(H^2)^2 - 16*det(H) ) )/2;
z2 = sqrt( trace(H^2) - sqrt( trace(H^2)^2 - 16*det(H) ) )/2;
% Calculation of the coefficients of the polynomial expression
c0 = (2*z2^2/((z1^2-z2^2)*(z1^2-1)) - 2*z1^2/((z1^2-z2^2)*(z2^2-1)));
c1 = (2*z2^2/((z1^2-z2^2)*(z1^2-1)) - 2*z1^2/((z1^2-z2^2)*(z2^2-1)));
c2 = (2/((z1^2-z2^2)*(z2^2-1)) - 2/((z1^2-z2^2)*(z1^2-1)));
c3 = (2/((z1^2-z2^2)*(z2^2-1)) - 2/((z1^2-z2^2)*(z1^2-1)));
% Calculation of the polynomial
P = c3*H^3 + c2*H^2 + c1*H + c0*eye(4);
```

**A.2. Evaluation of the complete functions** $\exp(L)$**,** $\mathrm{cay}(L)$**,** $\log(T)$**.** The numerical evaluation of the quantities $\exp(L)$ and $\log(T)$ in (1.11) and of $\mathrm{cay}(L)$ in (1.12) may be facilitated by noting that the matrix functions $M_1$, $M_2$ and $M_3$ are always multiplied by a vector $\delta$. Since matrix-to-vector multiplication is computationally lighter than matrix-to-matrix multiplication, a simple but effective optimization to lighten the computational cost would be to evaluate, for example, the product $M_1(H)\delta = (a_3H^3 + a_2H^2 + a_1H + a_0I)\delta$ by evaluating the products $H\delta$, $H(H\delta)$ and $H(H(H\delta)))$, which are all matrix-to-vector products.

The function $\exp(L)$ may be evaluated through the following ⟨⟨`PolyExp`⟩⟩ MATLAB© function (where the evaluation of the sub-matrix $\exp(H)$ is based on the results presented in [8]):

```
% Evaluation of the function exp(L) corresponding to the case that all the
% eigenvalues of the matrix H are distinct
function T = PolyExp(L)
% Extraction of the submatrices
H = L(1:4,1:4); v = L(1:4,5);
% Calculation of lambda_a (z1) and lambda_b (z2)
z1 = sqrt( trace(H^2) + sqrt( trace(H^2)^2 - 16*det(H) ) )/2;
z2 = sqrt( trace(H^2) - sqrt( trace(H^2)^2 - 16*det(H) ) )/2;
% Calculation of the coefficients of the polynomial expression of M_1(H)
a0 = (sinh(z2)*z1^3 - sinh(z1)*z2^3)/(z1*z2*(z1^2-z2^2));
a1 = z2^2*(cosh(z1)-1)/(z1^2*(z2^2-z1^2)) - z1^2*(cosh(z2)-1)/(z2^2*(z2^2-z1^2));
a2 = (z2*sinh(z1) - z1*sinh(z2))/(z1*z2*(z1^2-z2^2));
a3 = (cosh(z1)-1)/(z1^2*(z1^2-z2^2)) - (cosh(z2)-1)/(z2^2*(z1^2-z2^2));
% Calculation of the polynomial for delta
T1 = H*v; T2 = H*T1; T3 = H*T2;
delta = a3*T3 + a2*T2 + a1*T1 + a0*v;
% Calculation of the coefficients of the polynomial expression of exp(H)
k0 = (z2^2*cosh(z1)-z1^2*cosh(z2))/(z2^2-z1^2);
k1 = (z2^2*sinh(z1)/z1-z1^2*sinh(z2)/z2)/(z2^2-z1^2);
k2 = (-cosh(z1)+cosh(z2))/(z2^2-z1^2);
k3 = (-sinh(z1)/z1+sinh(z2)/z2)/(z2^2-z1^2);
% Calculation of the polynomial for S
H2 = H^2; H3 = H*H2;
S = k3*H3 + k2*H2 + k1*H + k0*eye(4);
% Packing the matrix T
T = [  S         delta;
     zeros(1,4)    1];
```

The function $\mathrm{cay}(L)$ may be evaluated through the following ⟨⟨`PolyCay`⟩⟩ MATLAB© function (where the evaluation of the sub-matrix $\mathrm{cay}(H)$ is based on the results presented in [8]):

```
% Evaluation of the function cay(L) corresponding to the case that all the
% eigenvalues of the matrix H are distinct
function T = PolyCay(L)
% Extraction of the submatrices
H = L(1:4,1:4); v = L(1:4,5);
% Calculation of lambda_a (z1) and lambda_b (z2)
z1 = sqrt( trace(H^2) + sqrt( trace(H^2)^2 - 16*det(H) ) )/2;
z2 = sqrt( trace(H^2) - sqrt( trace(H^2)^2 - 16*det(H) ) )/2;
% Calculation of the coefficients of the polynomial expression of M_3(H)
c0 = (2*z2^2/((z1^2-z2^2)*(z1^2-1)) - 2*z1^2/((z1^2-z2^2)*(z2^2-1)));
c1 = (2*z2^2/((z1^2-z2^2)*(z1^2-1)) - 2*z1^2/((z1^2-z2^2)*(z2^2-1)));
c2 = (2/((z1^2-z2^2)*(z2^2-1)) - 2/((z1^2-z2^2)*(z1^2-1)));
c3 = (2/((z1^2-z2^2)*(z2^2-1)) - 2/((z1^2-z2^2)*(z1^2-1)));
% Calculation of the polynomial for delta
T1 = H*v; T2 = H*T1; T3 = H*T2;
delta = c3*T3 + c2*T2 + c1*T1 + c0*v;
% Calculation of the coefficients of the polynomial expression of cay(H)
k0 =  - (z1^2 + 1)/(z1^2 - 1) - (2*z1^2)/((z1^2 - 1)*(z2^2 - 1));
k1 =  -(2*(z1^2 + z2^2 - 1))/((z1^2 - 1)*(z2^2 - 1));
k2 =  2/((z1^2 - 1)*(z2^2 - 1));
% Calculation of the polynomial for S
H2 = H^2; H3 = H*H2;
S = k2*(H3 + H2) + k1*H + k0*eye(4);
% Packing the matrix T
T = [  S         delta;
     zeros(1,4)    1];
```

The function $\log(T)$ may be evaluated through the following ⟨⟨`PolyLog`⟩⟩ MATLAB© function (where the evaluation of the sub-matrix $\log(S)$ is based on the results presented in [8]):

```
% Evaluation of the function log(T) corresponding to the case that all the
% eigenvalues of the matrix S are distinct of the type (lambda, 1/lambda,
% lambda*, 1/lambda*)
function L = PolyLog(T)
% Extraction of the submatrices
S = T(1:4,1:4); delta = T(1:4,5);
% Calculation of lambda (z)
p2 = ( trace(S)^2 - trace(S^2) )/2; p1 = -trace(S);
zi = ( -p1 + sqrt( p1^2 - 4*p2 + 8) )/2;
z = ( zi + sqrt( zi^2 - 4 ) )/2;
% Calculation of the coefficients of the polynomial expression of M_2(S)
zc=conj(z);
b0 = imag(log(z)*zc*(z^5-1)/((z-1)^2*(z+1)*imag(zc*(z^2+1))));
b1 = imag(log(z)*((1-z^5)*(zc^2+1)-z*zc*(z^3-1))/((z-1)^2*(z+1)*imag(zc*(z^2+1))));
b2 = imag(log(zc)*(z*(1-zc^5)+zc*(1-zc^3)*(z^2+1))/((zc-1)^2*(zc+1)*imag(zc*(z^2+1))));
b3 = imag(z*zc*log(z)*(1-z^3)/((z-1)^2*(z+1)*imag(zc*(z^2+1))));
% Calculation of the polynomial for v
T1 = S*delta; T2 = S*T1; T3 = S*T2;
v = b3*T3 + b2*T2 + b1*T1 + b0*delta;
% Calculation of the coefficients of the polynomial expression of log(S)
k0 = 2*real(zc*log(z)*(z^4+1)/((z^2-1)*(z-zc)*(1-z*zc)));
k1 = 2*real(log(zc)*((zc^4+1)*(z^2+1)+z*zc*(zc^2+1))/((zc^2-1)*(z-zc)*(1-z*zc)));
k2 = 2*real(log(z)*((z^3+zc)*(z*zc+1)+(z*(z^2*zc^2+1)))/((z^2-1)*(z-zc)*(1-z*zc)));
k3 = z*zc*2*real(log(zc)*(zc^2+1)/((zc^2-1)*(z-zc)*(1-z*zc)));
% Calculation of the polynomial for H
S2 = S^2; S3 = S*S2;
H = k3*S3 + k2*S2 + k1*S + k0*eye(4);
% Packing the matrix T
L = [  H        v ;
      zeros(1,5) ];
```