NUMERICAL LINEAR ALGEBRA WITH APPLICATIONS
Numer. Linear Algebra Appl. 2010; **00**:1–23
Published online in Wiley InterScience (www.interscience.wiley.com). DOI: 10.1002/nla

The geometric mean of two matrices from a computational viewpoint

Bruno Iannazzo*

Dipartimento di Matematica e Informatica, Università di Perugia, Via Vanvitelli 1, 06123, Perugia, Italy.
bruno.iannazzo@dmi.unipg.it

SUMMARY

The geometric mean of two matrices is considered from a computational viewpoint. Several numerical algorithms based on different properties and representations of the geometric mean are discussed and analyzed. It is shown that most of the algorithms can be classified in terms of the rational approximations of the inverse square root function. A review of relevant applications is given. Copyright © 2010 John Wiley & Sons, Ltd.

Received ...

KEY WORDS: matrix geometric mean, polar decomposition, matrix function, matrix iteration, Gaussian quadrature, Padé approximation, rational minimax approximation, Cyclic Reduction.

1. INTRODUCTION

A typical wish in mathematics is to generalize concepts as much as possible. It is then understood why researchers have tried to generalize the concept of geometric mean to the matrix generalizations of positive numbers, namely Hermitian positive definite matrices. We denote by $\mathcal{P}_n$ the set of $n \times n$ Hermitian positive definite matrices, which we will call just positive matrices. The geometric mean of two matrices need to be a function $\varphi : \mathcal{P}_n \times \mathcal{P}_n \to \mathcal{P}_n$.

The generalization is not trivial, since the formula $\sqrt{ab}$, for two positive numbers a and b, when applied to matrices leads to $(AB)^{1/2}$, which is unsatisfactory since it may be non-Hermitian for certain $A, B \in \mathcal{P}_n$. A different, more fruitful, approach to get a fair generalization is axiomatic, that is derive the definition of geometric mean from the properties it ought to satisfy.

A natural property is the following: given a diagonal matrix $D = \operatorname{diag}(d_1, \dots, d_n)$, with $d_i > 0$, and the identity matrix I, the geometric mean is $\varphi(D, I) := \operatorname{diag}(\sqrt{d_1}, \dots, \sqrt{d_n})$. The aforementioned property is referred to as *consistency with scalars*.

The consistency with scalars is not sufficient to uniquely define a geometric mean. We need another property, namely the *invariance under congruence*: let $A, B \in \mathcal{P}_n$ and S belonging to the set $GL(n)$ of invertible matrices of size n, then $\varphi(S^*AS, S^*BS) = S^*\varphi(A, B)S$. The invariance under congruence is mathematically appealing since it states that the geometric mean interplay well with the action of $GL(n)$ over $\mathcal{P}_n$, that is the congruence.

We recall a minor variation of a result that can be found in [10, Sec. 4.1].

Theorem 1
Let $\varphi : \mathcal{P}_n \times \mathcal{P}_n \to \mathcal{P}_n$ be a function which verifies both consistency with scalars and invariance

*Correspondence to: Dipartimento di Matematica e Informatica, Università di Perugia, Via Vanvitelli 1, 06123, Perugia, Italy. bruno.iannazzo@dmi.unipg.it
†Please ensure that you use the most up to date class file, available from the NLA Home Page at
http://www3.interscience.wiley.com/journal/5957/home

under congruence, then

$$\varphi(A,B) = A^{1/2}(A^{-1/2}BA^{-1/2})^{1/2}A^{1/2} =: A\#B. \tag{1}$$

The symbol $A^{1/2}$ stands for the principal square root of the matrix A, which is a matrix satisfying the equation $X^2 = A$, and whose eigenvalues have positive real part. Such a matrix exists and is unique if A has no nonpositive real eigenvalues, in particular if A is positive then $A^{1/2}$ is positive. Moreover, for any invertible matrix M, we have $M^{-1}A^{1/2}M = (M^{-1}AM)^{1/2}$.

Using the properties of the principal square root one can derive

$$A\#B = A(A^{-1}B)^{1/2} = (BA^{-1})^{1/2}A = B(B^{-1}A)^{1/2} = (AB^{-1})^{1/2}B. \tag{2}$$

It can be proved, moreover, that $A\#B$ verifies several other properties required by a geometric mean, for instance: $A\#B = B\#A$; $A^{-1}\#B^{-1} = (A\#B)^{-1}$; for any $\alpha > 0$, we have $(\alpha A)\#B = \sqrt{\alpha}(A\#B)$; if A and B commute, then $A\#B = (AB)^{1/2}$; we have $\det(A\#B) = \sqrt{\det(A)\det(B)}$.

Another list of good properties is obtained using the Loewner partial order on Hermitian matrices, for which $A \succeq B$, for Hermitian A and B, when $A - B$ is positive semidefinite. For instance, if A and B are positive definite then

$$A \succeq A\#B \succeq B \quad \text{when} \quad A \succeq B \tag{3}$$

and thus the geometric mean lies "between" as the term "mean" suggests. Moreover, the function $A\#B$ turns out to be monotone with respect to its matrix arguments, namely if $\widehat{A} \succeq A$ and $\widehat{B} \succeq B$, then $\widehat{A}\#\widehat{B} \succeq A\#B$.

For a comprehensive list of properties with proofs and an historical account, we refer the reader to Chapter 4 of the book [10, Sec. 4] by Bhatia.

Another hint, if needed, on the goodness on the generalization of the geometric mean, based on the definition (1), is given by the fact that it generalizes the following property: let a and b be two positive real numbers, their geometric mean $\sqrt{ab}$ can be obtained as the limit of the sequences $a_{k+1} = (a_k + b_k)/2$, $b_{k+1} = 2a_kb_k/(a_k + b_k)$ with $a_0 = a$ and $b_0 = b$. The updated values a_{k+1} and b_{k+1} are the arithmetic and the harmonic mean, respectively, of a_k and b_k.

This "averaging technique" can be applied also to matrices leading to the coupled iterations

$$\begin{cases} A_0 = A, \quad B_0 = B, \\ A_{k+1} = (A_k + B_k)/2, \\ B_{k+1} = 2A_k(A_k + B_k)^{-1}B_k = 2(A_k^{-1} + B_k^{-1})^{-1}, \end{cases} \qquad k = 0, 1, 2, \ldots, \tag{4}$$

where A_k and B_k, for $k = 1, 2, \ldots$, both converge to $A\#B$. Observe that A_{k+1} is the arithmetic mean of A_k and B_k, while B_{k+1} is the harmonic mean of A_k and B_k, both of which are positive definite matrices when A_k and B_k are. This property has been noted by Anderson and Trapp [3] and by Kubo and Ando [33].

An interesting, recent characterization of the geometric mean is given in terms of a special Riemannian geometry of $\mathcal{P}_n$. The geometry is obtained by the scalar product $\langle X, Y\rangle_A = \mathrm{trace}(A^{-1}XA^{-1}Y)$ on the tangent space $T_A\mathcal{P}_n$ at a positive matrix A (which can be identified with the set of Hermitian matrices). In the resulting Riemannian manifold there exists only one geodesic, $\gamma : [0, 1] \to \mathcal{P}_n$, joining any two positive definite matrices A and B and whose natural parametrization is known to be [10, Thm. 6.1.6] (see also [34])

$$A\#_tB := \gamma(t) = A(A^{-1}B)^t = A^{1/2}(A^{-1/2}BA^{-1/2})^tA^{1/2}. \tag{5}$$

It is now apparent that $A\#B = A\#_{1/2}B$ is the mid-point of the geodesic joining A and B. The value $A\#_tB$, for $t \in (0, 1)$ is sometimes referred to as the weighted geometric mean.

The geometric mean of two matrices has several relevant applications, some of which are reviewed in Section 6.

It could be argued that requiring invariance under congruence is too much and, in order to give a physical meaning to the geometric mean, it might be sufficient to require invariance under

DOI: 10.1002/nla

unitary congruence. Relaxing this assumption, different definitions of geometric mean arise, such as $\exp(\frac{1}{2}(\log(A)+\log(B)))$ or the spectral mean of Fiedler and Pták [19]. Both definitions enjoy some of the aforementioned properties of $A\#B$; nevertheless, they do not verify others, for instance (3) and thus, in our opinion, they cannot be considered good generalizations of the scalar geometric mean to matrices.

The contributions of the paper are of different kind. First of all, we derive a simple formula for $A\#B$ in terms of the polar decomposition. Then, we survey on algorithms for the geometric mean and related quantities such as $A\#_tB$, discovering unexpected connections.

As far as we know, the methods considered in the literature are the averaging technique of Anderson and Trapp [3] (see also [33]), a method based on the matrix sign function of Higham et al. [24], the palindromic cyclic reduction of Iannazzo and Meini [29] and a method based on a continued fraction expansion of Raïssouli and Leazizi [43]. We show that the sign method and the palindromic cyclic reduction are two variants of the averaging technique.

We present some further algorithms for computing the matrix geometric mean, based on suitable modifications of known techniques: the first one is based on the Cholesky factorization and the Schur (spectral) decomposition; the second is based on the expression of $A\#B$ in terms of the polar decomposition of certain matrices; the third is a Gaussian quadrature applied to an integral representation of the geometric mean; while the fourth is based on the rational minimax approximation to the inverse square root which is obtained as a direct application the algorithm of Higham, Hale and Trefethen [21].

A perhaps surprising property is that the polar decomposition algorithm and the Gaussian quadrature, in their basic definition, produce one of the sequences of the averaging technique. Moreover, they can be described in terms of certain Padé approximations of the inverse square root as the argument approaches to 1.

The organization of the paper is as follows. In the next section we give a formula for the the geometric mean which will be useful later. In Section 3 we discuss the Cholesky-Schur algorithm which can be applied also to compute $A\#_tB$. In Section 4 we discuss the algorithms related to the Padé approximation of $z^{-1/2}$ while in Section 5 we discuss the one related to its rational minimax approximation. In Section 6 we review some applications where a matrix geometric mean is required. In Section 7 we perform some numerical tests, while in Section 8 we draw the conclusions.

Now, we recall some concepts and facts that will be used in the paper. Any nonsingular matrix M can be written as HU where H is Hermitian and U is unitary; the latter is called the polar factor of M, which we denote by $\mathrm{polar}(M)$, and has explicit expression $U = M(M^*M)^{-1/2}$. Finally, lef $f(A)$ be a matrix function, then for any invertible matrix M, we have

$$f(MAM^{-1}) = Mf(A)M^{-1}; \tag{6}$$

we call this property *similarity invariance* of matrix functions. Beside similarity invariance, we use several other properties of general and specific matrix functions, for this topic we address the reader to the book [22] by Higham.

2. A PROPERTY OF THE GEOMETRIC MEAN

Any positive matrix A can be written as $A = C^*C$ for an invertible C. Two noticeable examples are $A = A^{1/2}A^{1/2}$ and the Cholesky factorization $A = R^*R$, where R is upper triangular with positive diagonal entries.

Given two positive matrices A and B, with factorizations $A = C^*C$ and $B = D^*D$, the matrix geometric mean of A and B can be characterized using the following result which generalizes Proposition 4.1.8 of [10] (compare [19, Thm. 2.1]).

Proposition 2
Let $A = C^*C$ and $B = D^*D$ with $C, D \in \mathbb{C}^{n\times n}$ nonsingular. Then

$$A\#B = C^* \,\mathrm{polar}(CD^{-1})D, \tag{7}$$

DOI: 10.1002/nla

where $\mathrm{polar}(CD^{-1})$ is the unitary polar factor of CD^{-1}. Moreover, let U be a unitary matrix such that $C^*UD > 0$, then $C^*UD = A\#B$ and $U = \mathrm{polar}(CD^{-1})$.

Proof
Using the formula $\mathrm{polar}(M) = M(M^{-1}M^{-*})^{1/2}$ and the similarity invariance of the square root (6) we get

$$C^* \,\mathrm{polar}(CD^{-1})D = C^*CD^{-1}(DC^{-1}C^{-*}D^*)^{1/2}D \\ = A(D^{-1}DA^{-1}D^*D)^{1/2} = A(A^{-1}B)^{1/2} = A\#B.$$

The second statement can be obtained suitably modifying the proof of Proposition 4.1.8 of [10]. □

The novelty of Proposition 2 with respect to Proposition 4.1.8 of [10] is the identification of the matrix U as a polar factor, namely formula (7). This will be crucial to design an efficient numerical algorithm for computing $A\#B$.

3. AN ALGORITHM BASED ON THE SCHUR DECOMPOSITION

We explain how to efficiently compute a point of the geodesic $A\#_tB$ (or, that is the same, computing the weighted geometric mean of A and B) using the Schur decomposition and the Cholesky factorization. The resulting algorithm can be used to compute the matrix geometric mean for $t = 1/2$. This kind of procedure is quite common in numerical linear algebra (compare [22, Problem 2.7]).

Consider the Cholesky factorizations $A = R_A^*R_A$ and $B = R_B^*R_B$. Using the similarity invariance of matrix functions we get

$$A\#_tB = A(A^{-1}B)^t = R_A^*R_A(R_A^{-1}R_A^{-*}BR_A^{-1}R_A)^t = R_A^*(R_A^{-*}BR_A^{-1})^tR_A, \tag{8}$$

and thus, the evaluation of $A\#_tB$ can be obtained by forming the Cholesky decomposition of A, inverting the Cholesky factor R_A (whose condition number is the square root of the one of A) and computing the t-th power of the positive definite matrix $V = R_A^{-*}BR_A^{-1}$. This is done by computing the Schur form $V = UDU^*$ (that is, the spectral decomposition of V) and getting

$$A\#_tB = R_A^*UD^tU^*R_A, \qquad R_A^{-*}BR_A^{-1} = UDU^*, \tag{9}$$

The power of D is obtained computing the scalar power of its diagonal elements.

If the condition number of A is greater than the one of B, it may be convenient to interchange A and B in order to get a possibly more accurate result. Using the simple equality $A\#_tB = B\#_{1-t}A$, the formula to be used is

$$A\#_tB = B\#_{1-t}A = R_B^*UD^{1-t}U^*R_B, \qquad R_B^{-*}AR_B^{-1} = UDU^*. \tag{10}$$

We synthesize the procedure.

Algorithm 3.1 (Cholesky-Schur method) Given A and B positive definite matrices, $t \in (0, 1)$, compute $A\#_tB$.

1. if the condition number of A is greater than the condition number of B then interchange A and B, and then compute $A\#_{1-t}B$;
2. compute the Cholesky factorizations $A = R_A^*R_A$, $B = R_B^*R_B$ and form $V = R_A^{-*}BR_A^{-1} = X^*X$ where X is the upper triangular matrix solving $XR_B = R_A$;
3. compute the Schur decomposition $UDU^* = V$;
4. compute $A\#_tB = R_B^*UD^tU^*R_B$.

DOI: 10.1002/nla

The computational cost of the procedure is given by the Cholesky factorizations ($\frac{2}{3}n^3$ arithmetic operations (ops)), the computation of V (n^3 ops), the Schur decomposition (about $9n^3$ ops), the computation of $R_B^* U D^t U^* R_B$ ($3n^3$ ops), for a total cost of about $(14 + \frac{2}{3})n^3$ ops.

Notice that all steps of Algorithm 3.1 can be performed in a stable way. Observe, moreover, that the algorithm works for any $t \in \mathbb{R}$.

4. ALGORITHMS BASED ON THE PADÉ APPROXIMATION OF $z^{-1/2}$

We give three methods (with variants) for computing the matrix geometric mean, based on matrix iterations or a quadrature formula. The algorithms are derived using different properties of the matrix geometric mean, however, perhaps surprisingly, they give essentially the same sequences which can be also derived using certain Padé approximation of $z^{-1/2}$ in the formula $A(B^{-1}A)^{-1/2}$.

The first method is well-known and it is based on the simple property, recalled in Section 1, that iterating two means one obtains a new mean: the geometric mean is obtained as the limit of an iterated arithmetic-harmonic mean. The second method is based on the polar decomposition, which benefits from the possibility to perform the computation in a numerically stable way [32, 41]. The latter is based on an integral representation of the matrix geometric mean computed with a Gauss-Chebyshev quadrature, that method could be useful if one is interested in the computation of $(A\#B)v$, for A and B large and sparse.

4.1. Scaled averaging iteration

The averaging iteration

$$\begin{cases} A_0 = A, \quad B_0 = B, \\ A_{k+1} = (A_k + B_k)/2, \\ B_{k+1} = 2A_k(A_k + B_k)^{-1}B_k = 2(A_k^{-1} + B_k^{-1})^{-1}, \end{cases} \qquad k = 0, 1, 2, \ldots, \tag{11}$$

is, as far as we know, the first algorithm to efficiently compute $A\#B$ [3].

The sequences A_k and B_k are related by the simple formulae $A_k = AB_k^{-1}B = BB_k^{-1}A$ (or equivalently $B_k = AA_k^{-1}B = BA_k^{-1}A$), which are trivial for $k = 0$ and, assuming them true for k, then, the equality $B_{k+1}^{-1} = (A_k^{-1} + B_k^{-1})/2$ yields

$$\begin{aligned} A_{k+1} = \frac{1}{2}(A_k + B_k) &= \frac{1}{2}(AB_k^{-1}B + AA_k^{-1}B) = AB_{k+1}^{-1}B, \\ &= \frac{1}{2}(BB_k^{-1}A + BA_k^{-1}A) = BB_{k+1}^{-1}A, \end{aligned} \tag{12}$$

hence, the formulae are proved by an induction argument. Notice that $A_k\#B_k = A\#B$.

Using the previous relationships, iteration (11) can be uncoupled obtaining the single iterations

$$A_0 = A \text{ (or } B), \quad A_{k+1} = \frac{1}{2}(A_k + AA_k^{-1}B), \quad k = 0, 1, 2, \ldots, \tag{13}$$

and

$$B_0 = A \text{ (or } B), \quad B_{k+1} = 2(B_k^{-1} + B^{-1}B_kA^{-1})^{-1}, \quad k = 0, 1, 2, \ldots, \tag{14}$$

each of which converges to $A\#B$.

Iteration (13) and (14) have the same computational cost as (11), and seem to be more attractive from a computational point of view since they require less storage. However, iterations (13) and (14) are prone to numerical instability.

Yet another elegant way to write the averaging iteration is obtained observing that

$$B_{k+1} = 2A_k(A_k + B_k)^{-1}(A_k + B_k - A_k) = 2A_k - 2A_k(A_k + B_k)^{-1}A_k = 2A_k - A_kA_{k+1}^{-1}A_k,$$

DOI: 10.1002/nla

which yields the three-terms recurrence

$$\begin{cases} A_0 = A, \quad A_1 = \dfrac{1}{2}(A+B), \\ A_{k+2} = \dfrac{1}{2}(A_{k+1} + 2A_k - A_k A_{k+1}^{-1} A_k), \end{cases} \qquad k = 0, 1, 2, \dots \tag{15}$$

Essentially, the same algorithm is obtained applying Newton's method for computing the sign of a matrix in the following equality proved by Higham et al. [24]:

$$\operatorname{sign}(C) = \begin{bmatrix} 0 & A\#B \\ (A\#B)^{-1} & 0 \end{bmatrix}, \quad C := \begin{bmatrix} 0 & B \\ A^{-1} & 0 \end{bmatrix}. \tag{16}$$

The sign of a matrix M having nonimaginary eigenvalues can be defined as the limit of the iteration $M_{k+1} = (M_k + M_k^{-1})/2$ with $M_0 = M$. Applying the latter iteration to the matrix C of (16) yields a sequence $C_k = \left[\begin{smallmatrix} 0 & X_k \\ Y_k & 0 \end{smallmatrix}\right]$ and the coupled iterations

$$\begin{cases} X_0 = A, \quad Y_0 = B^{-1}, \\ X_{k+1} = \dfrac{1}{2}(X_k + Y_k^{-1}), \\ Y_{k+1} = \dfrac{1}{2}(Y_k + X_k^{-1}), \end{cases} \qquad k = 1, 2, \dots \tag{17}$$

where X_k converges to $A\#B$ and Y_k converges to $(A\#B)^{-1}$.

We prove by induction that the sequences (11) and (17) are such that $X_k = A_k$, $Y_k = B_k^{-1}$, for $k = 0, 1, 2, \dots$ In fact $X_0 = A = A_0$, $Y_0 = B^{-1} = B_0^{-1}$, while $X_{k+1} = (X_k + Y_k^{-1})/2 = (A_k + B_k)/2 = A_{k+1}$ and $Y_{k+1} = (Y_k + X_k^{-1})/2 = (A_k^{-1} + B_k^{-1})/2 = B_{k+1}^{-1}$.

Iteration (11), based on averaging can be implemented at the cost per step of three inversion of positive matrices, that is $3n^3$ ops, while iteration (17) based on the sign function can be implemented at a cost of $2n^3$ ops. Moreover, the scaling technique for the sign function allows one to accelerate the convergence. Let M be a matrix such that the sign is well defined, from $\operatorname{sign}(M) = \operatorname{sign}(\gamma M)$ for each $\gamma > 0$, one obtains the scaled sign iteration which is $M_0 = \gamma_0 M$, $M_{k+1} = (\gamma_k M_k + (\gamma_k M_k)^{-1})/2$, where γ_k is a suitable positive number which possibly reduces the number of steps needed for the required accuracy. A common choice is the determinantal scaling $\gamma_k = |\det(M_k)|^{-1/n}$ [15], a quantity that can be computed in an inexpensive way during the inversion of M_k. Another possibility is to use the spectral scaling $\gamma_k = \sqrt{\rho(M_k^{-1})/\rho(M_k)}$ [31], which is interesting in our case since the eigenvalues of $C = \left[\begin{smallmatrix} 0 & B \\ A^{-1} & 0 \end{smallmatrix}\right]$ are all real and simple (in fact $C^2 = \left[\begin{smallmatrix} BA^{-1} & 0 \\ 0 & A^{-1}B \end{smallmatrix}\right]$ has only real positive simple eigenvalues) and in this case a theorem of Barraud [8, 22] guarantees the convergence to the exact value of the sign in a number of steps equal to the number of distinct eigenvalues of the matrix.

To get the proper values of the scaling parameters it is enough to observe that $|\det(C_k)| = |\det(X_k)\det(Y_k)|$ and thus for the determinantal scaling $\gamma_k = |\det(X_k)\det(Y_k)|^{-1/(2n)}$, while $\rho(C_k) = \sqrt{\rho(X_k Y_k)}$ and thus for the spectral scaling $\gamma_k = \sqrt[4]{\rho((X_k Y_k)^{-1})/\rho(X_k Y_k)}$.

A scaled sign iteration is thus obtained.

Algorithm 4.1a (Scaled averaging iteration: sign based) Given A and B positive definite matrices. The matrix $A\#B$ is the limit of the matrix iteration

$$\begin{cases} X_0 = A, \quad Y_0 = B^{-1}, \\ \gamma_k = |\det(X_k)\det(Y_k)|^{-1/(2n)} \quad \left(\text{or } \gamma_k = \sqrt[4]{\rho((X_k Y_k)^{-1})/\rho(X_k Y_k)}\right) \\ X_{k+1} = \dfrac{1}{2}(\gamma_k X_k + (\gamma_k Y_k)^{-1}), \\ Y_{k+1} = \dfrac{1}{2}(\gamma_k Y_k + (\gamma_k X_k)^{-1}), \end{cases} \qquad k = 0, 1, 2, \dots \tag{18}$$

DOI: 10.1002/nla

Using the aforementioned connections between the sign iterates and the averaging algorithm the scaling can be applied to the latter obtaining the following three-terms scaled algorithm.

Algorithm 4.1b (Scaled averaging iteration: three-terms) Given A and B positive definite matrices. The matrix $A\#B$ is the limit of the matrix iteration

$$\begin{cases} \gamma_k = \left|\dfrac{\det(A_k)^2}{\det(A)\det(B)}\right|^{-1/(2n)}, \\ A_0 = A, \quad A_1 = \dfrac{1}{2}(\gamma_0 A + B/\gamma_0), \\ A_{k+2} = \dfrac{1}{2}(\gamma_{k+1}A_{k+1} + 2\gamma_k A_k/\gamma_{k+1} - \gamma_k^2/\gamma_{k+1}A_k A_{k+1}^{-1}A_k), \end{cases} \qquad k = 0, 1, 2 \ldots \tag{19}$$

The same sequence is obtained considering the Palindromic Cyclic Reduction (PCR)

$$\begin{cases} P_0 = \frac{1}{4}(A - B), \quad Q_0 = \frac{1}{2}(A + B), \\ P_{k+1} = -P_k Q_k^{-1} P_k, \\ Q_{k+1} = Q_k - 2P_k Q_k^{-1} P_k, \end{cases} \qquad k = 0, 1, 2, \ldots \tag{20}$$

whose limits are $\lim_k Q_k = A\#B$ and $\lim_k P_k = 0$. This convergence result is based on the fact that the matrix Laurent polynomial

$$\mathcal{L}(z) = \frac{1}{4}(A^{-1} - B^{-1})z^{-1} + \frac{1}{2}(A^{-1} + B^{-1}) + \frac{1}{4}(A^{-1} - B^{-1})z,$$

is invertible in an annulus containing the unit circle and the sequence Q_k of the PCR converges to the central coefficient of its inverse, namely $A\#B$ [30]. Since the PCR verifies the same three-terms recurrence (19) as the averaging iteration (see [29]), one obtains that $Q_k = A_{k+1}$ and thus $P_k = (A_k - B_k)/4$.

The connection with PCR allows one to describe more precisely the quadratic convergence of the averaging technique, using the results on the convergence of the PCR.

Corollary 3
Let A and B be positive definite matrices. The sequences A_k and B_k converge to $A\#B$ with $\|A_k - A\#B\| = O(\gamma^{2^k})$ and $\|B_k - A\#B\| = O(\gamma^{2^k})$, for any $\rho < \gamma < 1$ with

$$\rho = \max_{\lambda \in \sigma(BA^{-1})} \left|\frac{\sqrt{\lambda} - 1}{\sqrt{\lambda} + 1}\right|. \tag{21}$$

Proof
From Theorem 5.9 of [13], we have $\limsup_{k\to\infty} \|Q_k - A\#B\| = \rho(X_*)^2$, where X_* is the minimal solution of the matrix equation

$$\frac{1}{4}(A^{-1} - B^{-1})X^2 + \frac{1}{2}(A^{-1} + B^{-1})X + \frac{1}{4}(A^{-1} - B^{-1}) = 0,$$

which implies that $\|Q_k - A\#B\| = O(\xi^{2^k})$ for any $\rho^2 < \xi < 1$ with $\rho := \rho(X_*)$.

On the other hand, we have the explicit formula $X_* = -N(I + (I - N^2)^{1/2})^{-1}$ [30, Thm. 5], with $N = (A^{-1} + B^{-1})^{-1}(A^{-1} - B^{-1}) = (BA^{-1} + I)^{-1}(BA^{-1} - I)$, thus

$$\rho(X_*) = \frac{\sigma}{1 + \sqrt{1 - \sigma^2}}, \qquad \sigma = \max_{\lambda \in \sigma(BA^{-1})} \left|\frac{\lambda - 1}{\lambda + 1}\right|, \tag{22}$$

and by a simple manipulation we get $\rho := \rho(X_*) = \max_{\lambda\in\sigma(BA^{-1})} |\frac{\sqrt{\lambda}-1}{\sqrt{\lambda}+1}|$. The corollary is proved observing that $A_k = Q_{k-1}$, for $k \geqslant 1$. □

DOI: 10.1002/nla

4.2. Padé approximants to $z^{-1/2}$

We give another interpretation of the sequences obtained by the averaging technique in terms of the Padé appoximants of the function $z^{-1/2}$. To this end, we manipulate the sequence A_k of (11) showing its connection with Newton's method for the matrix square root and with the matrix sign iteration.

Let $S = A^{-1}B$ and consider the (simplified) Newton method for the square root of S, namely

$$\widehat{A}_0 = I, \quad \widehat{A}_{k+1} = \frac{1}{2}(\widehat{A}_k + \widehat{A}_k^{-1}S). \tag{23}$$

The sequence $\widehat{A}_k$ converges to $S^{1/2}$ for any A and B, since the eigenvalues of S are real and positive [22, Thm. 6.9]. We claim that $\widehat{A}_k = A^{-1}A_k$, where A_k is one of the two sequences obtained by the averaging iteration. To prove this fact, a simple induction is sufficient, in fact assuming that $A_k = A\widehat{A}_k$, we have

$$A\widehat{A}_{k+1} = \frac{1}{2}(A\widehat{A}_k + A\widehat{A}_k^{-1}S) = \frac{1}{2}(A_k + AA_k^{-1}AA^{-1}B) = A_{k+1},$$

in virtue of (13).

It is well known that Newton's method for the square root of the matrix S (23) is related to the matrix sign iteration

$$Z_{k+1} = \frac{1}{2}(Z_k + Z_k^{-1}), \quad Z_0 = S^{-1/2},$$

through the equality $Z_k = S^{-1/2}\widehat{A}_k$ [22], and thus we have that

$$A_k = A\widehat{A}_k = AS^{1/2}Z_k = (A\#B)Z_k. \tag{24}$$

The latter relation allows one to relate the averaging iteration to the Padé approximants to the function $t^{-1/2}$ in a neighborhood of 1. We use the reciprocal Padé iteration functions defined in [20] as

$$\varphi_{2m,2n+1}(z) = \frac{Q_{n,m}(1-z^2)}{zP_{n,m}(1-z^2)},$$

where $P_{n,m}(\xi)/Q_{n,m}(\xi)$ is the (n,m) Padé approximant to $(1-\xi)^{-1/2}$ at the point 0, that is

$$\frac{P_{n,m}(\xi)}{Q_{n,m}(\xi)} - (1-\xi)^{-1/2} = O(\xi^{m+n+1}),$$

as ξ tends to 0 and $P_{n,m}$ and $Q_{n,m}$ are polynomials of degree n and m, respectively.

We define the *principal reciprocal Padé iteration* for $m = n+1$ and $m = n$ as $\widetilde{g}_r(z) := \widetilde{g}_{m+n+1}(z) = \varphi_{2m,2n+1}(z)$, for which we prove the following composition property.

Lemma 4
Let r, s be positive integers. If r is even then $\widetilde{g}_{rs}(z) = \widetilde{g}_r(\widetilde{g}_s(z))$, if r is odd then $\widetilde{g}_{rs}(z) = \widetilde{g}_r\left(\frac{1}{\widetilde{g}_s(z)}\right)$.

Proof
The principal reciprocal Padé iterations are the reciprocal of the well-known principal Padé iterations, namely

$$\widetilde{g}_k(z) = \frac{1}{g_k(z)} = \frac{(1+z)^k + (1-z)^k}{(1+z)^k - (1-z)^k} \tag{25}$$

where the latter equality follows from the explicit expression of $g_k(z)$ given in [22, Thm. 5.9]. Notice that if r is even, then $g_r(1/z) = g_r(z)$, moreover, $g_{rs}(z) = g_r(g_s(z))$, in fact it is easy to see that the principal Padé iterations are conjugated to the powers through the Cayley transform $\mathcal{C}(z) = (1-z)/(1+z)$, that is

$$g_r(z) = \mathcal{C}(\mathcal{C}(z)^r), \tag{26}$$

DOI: 10.1002/nla

and thus

$$\widetilde{g}_r(\widetilde{g}_s(z)) = \frac{1}{g_r(\frac{1}{g_s(z)})} = \frac{1}{g_r(g_s(z))} = \frac{1}{g_{rs}(z)} = \widetilde{g}_{rs}(z),$$

while if r is odd, then $g_r(1/z) = 1/g_r(z)$ and we get $\widetilde{g}_{rs}(z) = \widetilde{g}_r\left(\frac{1}{\widetilde{g}_s(z))}\right)$. □

We are ready to state the main result of the section where we use $\widetilde{g}_2(z) = \frac{1+z^2}{2z}$.

Theorem 5
Let $P_k(z)/Q_k(z)$ be the $[2^{k-1}, 2^{k-1}-1]$ Padé approximant at 0 to the function $(1-z)^{-1/2}$, with $k > 0$, then $A_k = AQ_k(I - A^{-1}B)P_k(I - A^{-1}B)^{-1}$.

Proof
Let $Z_0 = (A^{-1}B)^{-1/2}$. We prove that $Z_k = \widetilde{g}_{2^k}(Z_0) = \varphi_{2^k,2^k-1}(Z_0)$, this is true for $k = 1$, in fact $Z_1 = \widetilde{g}_2(Z_0)$, while to prove the inductive step we use Lemma 4 so that $\widetilde{g}_{2^{k+1}}(Z_0) = \widetilde{g}_2(\widetilde{g}_{2^k}(Z_0)) = \widetilde{g}_2(Z_k) = Z_{k+1}$.

Equation (25) gives $\widetilde{g}_{2^k}(z) = \widetilde{g}_{2^k}(1/z)$ and then $\widetilde{g}_{2^k}(Z_0) = \widetilde{g}_{2^k}(Z_0^{-1}) = \varphi_{2^k,2^k-1}(Z_0^{-1})$. Thus, in view of equation (24) and recalling that $A\#B = AZ_0^{-1}$, we have

$$\begin{aligned} A_k = AZ_0^{-1}Z_k = \\ = AZ_0^{-1}Z_0Q_{2^{k-1},2^{k-1}-1}(I - Z_0^{-2})P_{2^{k-1},2^{k-1}-1}(I - Z_0^{-2})^{-1} \\ = AQ_k(I - A^{-1}B)P_k(I - A^{-1}B)^{-1}. \end{aligned}$$

□

As a byproduct of the previous analysis we get that the Newton method for the scalar square root is related to the Padé approximation of the square root function.

Corollary 6
Let $z \in \mathbb{C} \setminus (-\infty, 0]$, and let

$$z_{k+1} = \frac{1}{2}(z_k + zz_k^{-1}), \quad z_0 = z,$$

be the Newton iteration for the square root of z, then $z_k = \frac{p(z)}{q(z)}$, where $p(z)/q(z)$ is the $[2^{k-1}, 2^{k-1} - 1]$ Padé approximant at 1 of the square root function $z^{1/2}$.

Remark 7
Raïssouli and Leazizi propose in [43] an algorithm for the matrix geometric mean based on a matrix version of the continuous fraction expansion for scalars $a, b > 0$,

$$\sqrt{ab} = \left[\frac{a+b}{2}; \frac{-(\frac{a-b}{2})^2}{a+b}\right]_{k=1}^{\infty}.$$

The partial convergent $t_N = \left[\frac{a+b}{2}; \frac{-(\frac{a-b}{2})^2}{a+b}\right]_{k=1}^{N}$ is proved to be

$$t_N = \sqrt{ab}\frac{(1+\sqrt{ab})^{2N+2} + (1-\sqrt{ab})^{2N+2}}{(1+\sqrt{ab})^{2N+2} - (1-\sqrt{ab})^{2N+2}},$$

thus from the expression for the Padé approximation in (25), and the characterization of the averaging iteration in terms of the Padé approximation we get that $A_k = t_{2^{k-2}-1}$, for $k \geqslant 2$, where A_k is one of the sequences obtained by the averaging iteration with $A_0 = a$ and $B_0 = b$.

The same equivalence holds in the matrix case, so we get that the sequence t_N converges linearly to the matrix geometric mean with a cost similar to the averaging iteration which indeed converges quadratically and moreover can be scaled. Thus, in our opinion the sequence t_N is of little computational interest.

DOI: 10.1002/nla

4.3. Algorithm based on the polar decomposition

Let $A = R_A^* R_A$ and $B = R_B^* R_B$ be the Cholesky factorizations of A and B, respectively. Using these factorization in formula (7) we obtain the following representations for the matrix geometric mean

$$\begin{aligned} A\#B &= R_A^* \operatorname{polar}(R_A R_B^{-1}) R_B = R_A^* \operatorname{polar}(R_B R_A^{-1})^* R_B \\ &= R_B^* \operatorname{polar}(R_A R_B^{-1})^* R_A = R_B^* \operatorname{polar}(R_B R_A^{-1}) R_A, \end{aligned} \tag{27}$$

where we have used the symmetry of the matrix $A\#B$ and the commutativity of the matrix geometric mean function.

From (27) we derive an algorithm for computing the matrix geometric mean, which can be also seen as a variation of [26, Alg. 4.9] (see also [22, Alg. 6.22]).

Algorithm 4.2 (Polar decomposition) Given A and B positive definite matrices.

1. Compute the Cholesky factorizations $A = R_A^* R_A$ and $B = R_B^* R_B$;
2. Compute the unitary polar factor U of $R_B R_A^{-1}$;
3. Compute $A\#B = R_B^* U R_A$.

The polar factor of a matrix M can be computed forming its singular value decomposition, say $M = Q_1^* \Sigma Q_2$; from which we get the polar factor of M as $Q_1^* Q_2$ [22, Ch. 8]. This procedure is suitable for an accurate computation due to the good numerical property of the SVD algorithm, but it is expensive with respect to a method based on matrix iterations.

A more viable way to compute the unitary polar factor of M is to use the scaled Newton method

$$Z_{k+1} = \frac{1}{2}(\gamma_k Z_k + (\gamma_k Z_k)^{-*}), \quad Z_0 = M, \tag{28}$$

where $\gamma_k > 0$ can be chosen in order to reduce the number of steps needed for convergence.

A nice property of the scaled Newton method for the unitary polar factor of a matrix is that the number of steps can be predicted in advance for a certain machine precision and the algorithm is numerically stable if the inversion is performed in a mixed backward/forward way (see [32, 41]). An alternative is to compute the polar decomposition using a scaled Halley iteration as in [41].

The better choice for the scaling factor in Newton's iteration is the optimal scaling $\gamma_k = (\sigma_1(X_k)\sigma_n(X_k))^{1/2}$, where $\sigma_1(X_k)$ and $\sigma_n(X_k)$ are the extreme singular values of X_k. In practice, cheaper approximations of the optimal scaling are available [22, Sec. 8.6], such as

$$\gamma_k = \left(\frac{\|Z_k^{-1}\|_1 \|Z_k^{-1}\|_\infty}{\|Z_k\|_1 \|Z_k\|_\infty} \right)^{1/4}. \tag{29}$$

If $\gamma_k = 1$ for each k, then the sequence Z_k obtained by iteration (28) with $Z_0 = R_B R_A^{-1}$ is strictly related to the sequence obtained by the averaging technique, in fact $Z_k = R_B^{-*} A_k R_A^{-1}$, where A_k is defined in (13) with $A_0 = B$. This equality can be proved by an induction argument in fact $R_B^{-*} A_0 R_A^{-1} = R_B^{-*} R_B^* R_B R_A^{-1} = Z_0$ and if the equality is true for k, then

$$\begin{aligned} Z_{k+1} = \frac{1}{2}(Z_k + Z_k^{-*}) &= R_B^{-*} \left(\frac{A_k}{2} + R_B^* \frac{R_B A_k^{-1} R_A^*}{2} R_A \right) R_A^{-1} \\ &= R_B^{-*} \left(\frac{A_k + B A_k^{-1} A}{2} \right) R_A^{-1} = R_B^{-*} \left(\frac{A_k + A A_k^{-1} B}{2} \right) R_A^{-1} = R_B^{-*} A_{k+1} R_A^{-1}. \end{aligned}$$

Remark 8

Notice that for $A = I$, Algorithm 4.2 reduces to the algorithm of Higham [22, Alg. 6.21] for the square root of a positive matrix B, and can be seen as a generalization of it. A side-result of the previous discussion is that Higham's algorithm can be seen as yet another variant of the Newton method for the matrix square root.

DOI: 10.1002/nla

4.4. *Gaussian quadrature*

The definition $A\#B = A(B^{-1}A)^{-1/2}$ in terms of an inverse square root yields a rather large number of integral representations [30] among which we note the following:

$$A\#B = \frac{1}{\pi}\int_0^1 \frac{(tB^{-1}+(1-t)A^{-1})^{-1}}{\sqrt{t(1-t)}}\,dt, \tag{30}$$

obtained by Ando, Li and Mathias [4] using an Euler integral. The same representation can be obtained from the Cauchy integral formula for the function $z^{-1/2}$ [30].

An algorithm for the geometric mean of two matrices is obtained through the integral representation (30). The change of variable $z = \frac{t+1}{2}$ yields

$$A\#B = \frac{2}{\pi}\int_{-1}^1 \frac{((1+z)B^{-1}+(1-z)A^{-1})^{-1}}{\sqrt{1-z^2}}\,dz, \tag{31}$$

which is well suited for Gaussian quadrature with respect to the weight function $\omega(z) = (1-z^2)^{-1/2}$, referred to as Gauss-Chebyshev quadrature since the orthogonal polynomials with respect to the weight $\omega(z)$ are the Chebyshev polynomials (see [16] for more details). For an integral of the form

$$\int_{-1}^1 \frac{f(z)}{\sqrt{1-z^2}}\,dz,$$

where f is a suitable function, the formula with N nodes is

$$\Sigma_{N+1} = \frac{\pi}{N+1}\sum_{k=0}^{N} f(x_k), \quad x_k = \cos\left(\frac{(2k+1)\pi}{2(N+1)}\right), \quad k = 0,\dots,N.$$

Applying the Gauss-Chebyshev quadrature formula to (31) we obtain the following approximation of $A\#B$

$$\begin{aligned} T_{N+1}(A,B) &= \frac{2}{N+1}\sum_{k=0}^{N}((1+x_k)B^{-1}+(1-x_k)A^{-1})^{-1} \\ &= B\left(\frac{2}{N+1}\sum_{k=0}^{N}((1+x_k)A+(1-x_k)B)^{-1}\right)A. \end{aligned} \tag{32}$$

Algorithm 4.3 (Gauss-Chebyshev quadrature) Given A and B positive definite matrices. Choose N and set

$$A\#B \approx T_N(A,B).$$

where $T_N(A,B)$ is defined in (32).

The computation cost is the inversion of a positive matrix, that is n^3 ops, for each node of the quadrature and two matrix multiplication at the end. The number of nodes required to get a fixed accuracy depends on the regularity of the function $\psi(z) = ((1+z)A+(1-z)B)^{-1}$. The function $\psi(z)$ is rational and thus analytic in the complex plane except the values of z such that $\psi(z)$ is singular, which are the reciprocal of the nonzero eigenvalues of the matrix $(B-A)(B+A)^{-1} = (BA^{-1}-I)(BA^{-1}+I)^{-1}$.

We claim that all the poles are real and lie outside the interval $[-1,1]$, which is equivalent to require that the eigenvalues of $(BA^{-1}-I)(BA^{-1}+I)^{-1}$ lie in the interval $(-1,1)$. Define $\mathcal{C}(z) = (z-1)/(z+1)$, then the image under $\mathcal{C}(z)$ of the positive real numbers is the interval $(-1,1)$, then the eigenvalues of $\mathcal{C}(BA^{-1})$ lie in the interval $(-1,1)$ since BA^{-1} has positive eigenvalues $\lambda_1,\dots,\lambda_n$ and the eigenvalues of $\mathcal{C}(BA^{-1})$ are $\mathcal{C}(\lambda_1),\dots,\mathcal{C}(\lambda_n)$ (compare [22, Thm. 1.13]).

DOI: 10.1002/nla

Standard results on the convergence of the Gauss-Chebyshev quadrature (see [16, Thm. 3]) imply that the sequence $T_N(A,B)$ converges to $A\#B$ linearly, in particular for each $\rho^2 < \xi < 1$, we have $\|T_N(A,B) - A\#B\| = O(\xi^N)$, where $1/\rho$ is the sum of the semiaxes of an ellipse with foci in 1 and -1 and whose internal part is fully contained in the region of analiticity of $\psi(z)$.

Since the poles of $\psi(z)$ are real and lie outside the interval $(-1,1)$, then the largest ellipse is obtained for $\rho = 1/(\frac{1}{\sigma} + \sqrt{\frac{1}{\sigma^2} - 1}) = \sigma/(1+\sqrt{1-\sigma^2})$, where $\sigma = \max_i\{|\mathcal{C}(\lambda_i)|\}$ (notice that $1/\sigma$ is the pole of $\psi(z)$ nearest to $[-1,1]$). A straightforward manipulation gives $\rho = \max_i\{|\mathcal{C}(\sqrt{\lambda_i})|\}$.

If m and M are the smallest and largest, respectively, eigenvalues of BA^{-1}, then the convergence of $T_N(A,B)$ is slow if m is small or M is large. By a suitable scaling of A, it is possible to have $mM = 1$, which gives a faster convergence, however, when M/m tends to infinity the parameter of linear convergence tends to 1, in this case a simple analysis shows that $\rho = 1 + O\left(\sqrt[4]{\frac{m}{M}}\right)$ and thus the parameter of linear convergence of $T_N(A,B)$ depends on m/M as a square root. In Section 5 another quadrature formula whose dependence on M/m is just logarithmic is considered.

A comparison of the parameters of linear convergence for the Gauss-Chebyshev formula $T_N(A,B)$ and the parameters of quadratic convergence for the averaging iteration in Corollary 3 reveals that they are essentially the same. This is not a mere coincidence, in view of the following result.

Theorem 9
Let T_k be the quadrature formula of (32) and B_k be the sequence obtained by the averaging technique (11) then $B_k = T_{2^{k-1}}$, for $k = 1, 2, \ldots$

Proof
The equivalence follows from the formula for the term Q_k of the PCR

$$Q_k^{-1} = \frac{1}{2^k}\sum_{j=0}^{2^k-1}\left(Q_0 + 2P_0\cos\frac{(2j+1)\pi}{2^{k+1}}\right)^{-1},$$

provided in [29, Eq. 28], and from the equivalence $Q_k^{-1} = A_{k+1}^{-1} = B^{-1}B_{k+1}A^{-1}$ (see Eq. (12)) for $Q_0 = \frac{1}{2}(A+B)$ and $P_0 = \frac{1}{4}(A-B)$.

□

4.5. *Generalization to the weighted geometric mean*

In Section 3 we have shown that the Cholesky-Schur method is able to compute the weighted geometric mean $A\#_t B$ for $t \in (0,1)$. One might wonder how the methods presented in this section can be generalized to the weighted case.

Unfortunately, the weighted case appears to be more complicated, and most of the algorithms of this section, in the weighted case, become uncorrelated.

First, we consider the weighted version of the averaging iteration

$$\begin{cases} A_0 = A, \quad B_0 = B, \\ A_{k+1} = tA_k + (1-t)B_k, & k = 0, 1, 2, \ldots, \\ B_{k+1} = (tA_k^{-1} + (1-t)B_k^{-1})^{-1}, \end{cases}$$

for $t \in (0,1)$. It can be proved that these two sequences yield a common limit, but it can be different from $A\#_t B$. For instance, if $A = 2$ and $B = 1$, then the sequence A_k with $t = 1/3$ is easily shown to be decreasing and we have $A_2 = \frac{56}{45} < 2^{1/3} = A\#_t B$ so that $\lim_k A_k \neq A\#_t B$.

Second, it is possible to derive a generalization of (31) for a generic $t \in (0,1)$, namely

$$A\#_t B = \frac{2\sin(\pi t)}{\pi}\int_{-1}^{1}\frac{\left((1+z)B^{-1} + (1-z)A^{-1}\right)^{-1}}{(1-z)^t(1+z)^{1-t}}dz.$$

Using this integral representation, one could approximate the weighted mean using a Gaussian quadrature with weight function $\omega(z) = (1-z)^{-t}(1-z)^{t-1}$, whose poles are the zeros of certain

DOI: 10.1002/nla

Jacobi polynomials. The resulting method converges quite slowly and its practical usage is restricted to large-scale problems, not considered here.

Third, the Newton method for the pth root can be considered when $t = 1/p$, using the approximations of the pth root of $A^{-1}B$ to get a sequence of approximations to the weighted mean $A\#_{1/p}B = A(A^{-1}B)^{1/p}$. For stability reasons, the sequence generated by Newton's method for the pth root of a matrix $C \in \mathbb{C}^{n\times n}$ is obtained through the simplified Newton method in the form [27]

$$\begin{cases} X_0 = I, \quad N_0 = C, \\ X_{k+1} = X_k\left(\frac{(p-1)I+N_k}{p}\right), \\ N_{k+1} = \left(\frac{(p-1)I+N_k}{p}\right)^{-p} N_k. \end{cases} \tag{33}$$

We propose a variant of the simplified Newton method specifically tailored for the weighted geometric mean, namely

$$\begin{cases} Y_0 = A, \quad M_0 = A^{-1}B, \\ Y_{k+1} = Y_k\left(\frac{(p-1)I+M_k}{p}\right), \\ M_{k+1} = \left(\frac{(p-1)I+M_k}{p}\right)^{-p} M_k. \end{cases} \tag{34}$$

We summarize the properties of Newton's method for the weighted geometric mean in the following.

Theorem 10
Let A and B be positive definite, then the sequence $\{Y_k\}_k$ generated by the simplified Newton method (34) converges to $A\#_{1/p}B$.

Proof
The sequences $\{X_k\}_k$ and $\{N_k\}$ of (33), with $C = A^{-1}B$ converge to $(A^{-1}B)^{1/p}$ and I, respectively. This follows from the fact that, in the scalar case, the same algorithm, with $C = \lambda$, where λ is any eigenvalue of $A^{-1}B$, yields two sequences converging to $\lambda^{1/p}$ and 1, respectively (see [27] for further details).

To prove the convergence of the sequence $\{Y_k\}_k$ of (34) to $A\#_tB$ it is enough to observe that $Y_k = AX_k$ and $M_k = N_k$ for each k. This can be proved by an easy induction argument. □

The simplified Newton method is an interesting alternative to the Cholesky-Schur method, for $t = 1/p$, since the sequence Y_k is made of positive definite matrices. Unfortunately, the matrices M_k are not positive definite and thus the method does not fully exploit the structure. Moreover, it is easy to see that the sequence obtained by Newton's method is not related to the aforementioned Gaussian quadrature method for $p > 2$.

Finally, an alternative to compute $A\#_tB$ is to use directly one of the formulae

$$\begin{aligned} A\#_tB &= A(A^{-1}B)^t = B(B^{-1}A)^{1-t} \\ &= A\exp(t\log(A^{-1}B)) = A\exp(-t\log(B^{-1}A)). \end{aligned} \tag{35}$$

The expressions in the first row of (35) can be evaluated either by using the spectral decomposition of the matrix $A^{-1}B$ (or $B^{-1}A$) which is nonnormal in the generic case or using an algorithm for fractional powers of matrices [23, 28]. Alternatively, one could use the expressions in the second row of (35) where the exponential and the logarithm can be computed as explained in [22]. Unfortunately, none of these alternatives is of interest for this problem since they are more expensive than the Cholesky-Schur algorithm and do not exploit the positive definite structure of A, B and $A\#_tB$.

In summary, the best suited method for the weighted geometric mean of dense matrices seems to be the Cholesky-Schur method, while for the case of large and sparse matrices, the quadrature methods could have some interest. For $t = 1/p$, it would be interesting to get a variant of the Newton method which works only with positive definite matrices, together with a scaling technique, which, in this case, could become competitive with the Cholesky-Schur method.

DOI: 10.1002/nla

5. ALGORITHMS BASED ON THE RATIONAL MINIMAX APPROXIMATION OF $z^{-1/2}$

In Section 4 we have found that many algorithms for computing the matrix geometric mean are variations of the one obtained by using certain Padé approximations of $z^{-1/2}$ in the formula $A\#B = A(B^{-1}A)^{-1/2}$. To get something really different, one should change the rational approximation. The natural direction is towards the (relative) rational minimax approximation.

Let $R_{k-1,k}$ be the set of rational functions whose numerator and denominator have degree $k-1$ and k, respectively. The function $\widetilde{r}_{k,\gamma}(z)$ is said to be the rational relative minimax approximation to $z^{-1/2}$ in the interval $[1,\gamma]$ if it minimizes over $R_{k-1,k}$ the quantity

$$\max_{z\in[1,\gamma]} \left| \frac{r(z) - z^{-1/2}}{z^{-1/2}} \right|.$$

An explicit expression for $\widetilde{r}_{k,\gamma}(z)$, in terms of elliptic function is known since the work of Zolotarev in 1877 (see [18]).

The same approximation is obtained by Hale, Higham and Trefethen [21] by a trapezoidal quadrature following a clever sequence of substitutions applied to the Cauchy integral formula for $A^{-1/2}$, namely,

$$A^{-1/2} = \frac{1}{2\pi \mathbf{i}} \oint_{\Gamma} z^{-1/2}(zI - A)^{-1} dz,$$

where Γ is a contour enclosing the spectrum of A.

Since $A\#B = A(A^{-1}B)^{1/2} = B(A^{-1}B)^{-1/2}$, using the results of [21], we get immediately the following approximation (obtained by a quadrature formula on N nodes on a suitable integral representation of $A\#B$)

$$S_N(A,B) = B\left(\frac{-2K'\sqrt{m}}{\pi N} \sum_{j=1}^{N} (\omega(t_j)^2 A - B)^{-1} \mathrm{cn}(t_j|\gamma)\mathrm{dn}(t_j|\gamma) \right) A \tag{36}$$

which is proved to coincide with $A\, r_{N,\gamma}(B^{-1}A)$ for $\gamma = M/m$, where M and m are the largest and the smallest eigenvalues of $A^{-1}B$, respectively.

The notation of (36) has the following meaning:

$$t_j = \big(j - 1/2\big)\frac{K'}{N}\mathbf{i}, \quad 1 \leqslant j \leqslant N,$$

$\omega(t_j) = \sqrt{m}\,\mathrm{sn}(t_j|\gamma)$, where $\mathrm{sn}(t_j|\gamma)$, $\mathrm{cn}(t_j|\gamma)$ and $\mathrm{dn}(t_j|\gamma)$ are the Jacobi elliptic functions, while K' is the complete elliptic integral of the second kind associated with $\sqrt{\gamma}$ (see [1] for an introduction to elliptic functions and integrals).

The convergence of $S_N(A,B)$ to $A\#B$ can be deduced from Theorem 4.1 of [21]. In particular,

$$\|A\#B - S_N(A,B)\| = O(e^{-2\pi^2 N/(\log(M/m)+3)}).$$

Thus, the convergence of the sequence $S_N(A,B)$ to $A\#B$ is dominated by a sequence whose convergence is linear with a rate which tends to 1 as M/m tends to ∞, but whose dependence on M/m is just logarithmic. On the contrary, the rate of linear convergence of the Gauss-Chebyshev sequence $T_N(A,B)$ of (32) depends linearly on M/m, and thus we expect that the formula $S_N(A,B)$ requires less nodes than $T_N(A,B)$ to get the same accuracy on the approximation of $A\#B$ at least for large values of M/m.

We describe the synthetic algorithm.

Algorithm 5.1 (Rational minimax) Given A and B positive definite matrices. Choose N and set

$$A\#B \approx S_N,$$

where S_N is defined in (36).

DOI: 10.1002/nla

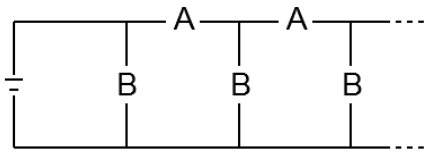

Figure 1. An electric circuit whose joint resistance is related to the matrix geometric mean

6. APPLICATIONS

We review some of the applications in which the geometric mean of two matrices is required, they range from electrical network analysis [3] to medical imaging [6], from image deblurring [17], to the computation of the geometric mean of several matrices [4, 14], with indirect applications to radar [7] and elasticity [39].

6.1. Electrical networks

Fundamental elements of a circuit are the resistances which can be modeled by positive real numbers. It is a customary high school argument that two consecutive resistances r_1 and r_2 in the same line can be modeled by a unique joint resistance whose value is the sum $r_1 + r_2$, while if the two resistances lie in two parallel lines their joint resistance is the "parallel sum" $(r_1^{-1} + r_2^{-1})^{-1}$.

More sophisticated devices based on resistances are n-port networks, which are "objects" with $2n$ ports at which current and voltage can be measured, without knowing what happens inside. The usual way to model n-port networks is through positive definite matrices. In this way two consecutive n-ports A and B can be modeled as the joint n-port $A + B$, while two parallel n-ports give the joint n-port $(A^{-1} + B^{-1})^{-1}$.

Complicated circuits, made of several n-ports can be reduced to a joint n-port using these sums and parallel sums. Consider the circuit in Figure 1: it is an infinite network (which models a large finite network).

Let Z_k be the joint resistance of the subcircuit obtained selecting the first k loops, then it can be shown that $Z_1 = B$ and

$$Z_{k+1} = (B^{-1} + (A + Z_k)^{-1})^{-1},$$

and the sequence has limit $\lim_{k\to\infty} Z_k = \frac{1}{2}(-A + (A\#(A + 4B)))$. This limit is the joint resistance of the infinite circuit. For further details see [2], from which the example is taken.

It is worth pointing out that the definition of geometric mean of two matrices first appeared in connection with these kind of applications [42].

6.2. Diffusion tensor imaging

The technique of Nuclear Magnetic Resonance (NMR) in medicine produces images of some internal parts of the body which are used by medics to give a diagnose of important pathologies or to decide how to perform a surgery.

One of the quantities measured by the NMR is the diffusion tensor which is a 3×3 positive matrix describing the diffusion of the water in tissues like the white matter of the brain or the prostate. The technique is called Diffusion Tensor Imaging (DTI) [35].

The diffusion tensor is measured for any of the points of an ideal grid into the tissue, thus one has a certain number of positive matrices indexed by their positions.

A problem in DTI is the "interpolation" of tensors, that is, given two tensors, find one or more tensors in the line joining them, the more adherent to the real data as possible. This is useful for instance to increase the resolution of an image or to reconstruct some corrupted parts.

Many models have been given for the interpolation of tensors in DTI, the most obvious of which is the linear interpolation, where k points between A and B are $P_j = \frac{j}{k+1}A + \frac{k+1-j}{k+1}B$, for

DOI: 10.1002/nla

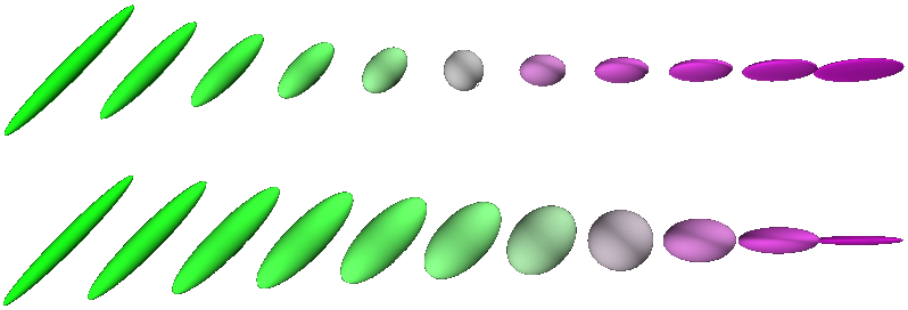

Figure 2. Interpolation of tensors, displayed as ellipsoids, using the Riemannian geometry (upper line) and the Euclidean geometry (lower line). One can observe the swelling effect in the second line, due to the large determinant of the central tensors. The picture has been obtained using the program TenVis by R. Parcus (`https://github.com/quiuquio/TenVis`).

$j = 1, \dots, k$. The linear interpolation finds point equally spaced on the line joining A and B in the space of the matrices, that is, uses the Euclidean geometry of $\mathbb{C}^{n\times n}$.

One of the problems related to the use of the Euclidean geometry arises when A and B have determinants of the same magnitude, while $(A+B)/2$ (or another point on the line joining A and B) has a much larger determinant. In terms of diffusion tensors, this means that the anisotropy at the mid-point is much larger than the anisotropy at two near points. This is called the *swelling effect* and it is removed using the geometry given in Section 1 [9], for which we get the interpolation points

$$P_k^a = A\#_{j/(k+1)}B = A(A^{-1}B)^{j/(k+1)}, \quad j = 1, \dots, k.$$

A visualization of the swelling effect is depicted in Figure 2.

Using the *log Euclidean* geometry defined in [6], we get the interpolation points

$$P_k^b = \exp\left(\frac{j}{k+1}\log(A) + \frac{k+1-j}{k+1}\log(B)\right), \quad j = 1, \dots, k.$$

The log Euclidean geometry has been introduced as an approximation to the Riemannian geometry where quantities are easier to be computed. However, in the interpolation problem described here, using the Cholesky-Schur algorithm of Section 3 to compute P_k^a (reusing the Schur factorization of $R_A^{-*}BR_A^{-1}$ for each j) is much less expensive than the computation of P_k^b using the customary algorithms for the logarithm and the exponential of a Hermitian matrix.

6.3. *Computing means of more than two matrices*

The generalization of the geometric mean to more than two positive matrices is usually identified with their Karcher mean in the geometry given in Section 1 (see [11] for a precise definition).

The Karcher mean of $A_1, \dots, A_m$ can been obtained as the limit of the sequence $S_k = S_{k-1}\#_{1/k}A_{(k \bmod m)+1}$, with $S_1 = A_1$ as proved by Holbrook [25] and extended by Lim and Pálfia [37]. The resulting sequence is very slow and cannot be used to design an efficient algorithm for the computation of the Karcher mean, however it may be useful to construct an initial value for some other iterative methods such as the Richardson-like iteration by Bini and Iannazzo [11]. Another way to approximate the Karcher mean is through a sequence of power means, which are in turn computed by a fixed-point iteration based on the weighed geometric mean of two matrices [36].

Other geometric-like means of more than two matrices are based on recursive definitions like the mean proposed by Ando, Li and Mathias [4], which for three matrices A_0, B_0 and C_0 is defined as the common limit of the sequences

$$A_{k+1} = B_k\#C_k, \quad B_{k+1} = C_k\#A_k, \quad C_{k+1} = A_k\#B_k, \quad k = 0, 1, 2, \dots$$

DOI: 10.1002/nla

These sequences converge linearly to their limit. Another similar definition which gives cubic convergence (to a different limit) has been proposed in [14, 40], where we find the iteration

$$A_{k+1} = A_k \#_{2/3} (B_k \# C_k), \quad B_{k+1} = B_k \#_{2/3} (C_k \# A_k), \quad C_{k+1} = C_k \#_{2/3} (A_k \# B_k).$$

As one can see, the efficient computation of $A\#_t B$ is a basic step to implement these kind of iterations.

6.4. *Image deblurring*

A classical problem in image processing is the image deblurring which consists in finding a clear image from a blurred one. In the classical models, the true and the blurred images are vectors and the blurring is seen as a linear operator A, thus the problem is reduced to the linear system $Af = g$ which in practice is very large and ill-conditioned. A computationally easy case is the one in which A is a band Toeplitz matrix, which corresponds to the so-called shift-invariant blurred operators.

Even if A is not shift-invariant, it can be possible, in certain cases, that a change of coordinates M makes it shift-invariant, i.e. $M^T AM$ is band Toeplitz. If such a M exists and is known, then the linear system $Af = g$ has the same nice computational properties as a band Toeplitz system.

When A and T are positive definite, the matrix $M = A^{-1}\#T$ is an explicit change of coordinates. For further details see [17].

6.5. *Norms of finite interpolation spaces*

The material of this section is taken from [5] to which we address the reader for a full detailed description with references.

Let $\Omega \subseteq \mathbb{R}^n$ be open, bounded and with smooth boundary, and let $H_0^1(\Omega)$ be the Sobolev space of differentiable functions on $L^2(\Omega)$ with zero trace, while $H_0^0(\Omega)$ be the set of functions on $L^2(\Omega)$ with zero trace. Let $\{\varphi_1, \ldots, \varphi_n\}$ be a set of linearly independent piecewise linear polynomials on a suitable subdivision of Ω (arising, for instance, from a finite elements method), then the span in H_0^1 (resp. H_0^0) of $\{\varphi_i\}_{i=1,\ldots,n}$, is an Hilbert subspace X_h (resp. Y_h).

Define the matrices L_0 and L_1 such that

$$(L_0)_{ij} = \langle \varphi_i, \varphi_j \rangle_{L^2(\Omega)}, \qquad (L_1)_{ij} = \langle \nabla\varphi_i, \nabla\varphi_j \rangle_{L^2(\Omega)}.$$

The matrices L_0 and L_1 are positive definite since they are Grammians with respect to a scalar product, in particular L_0 is a discrete identity and L_1 is a discrete Dirichlet Laplacian. A norm for the interpolation space $[X_h, Y_h]_\vartheta$ is given by the energy norm of the matrix

$$L_0 (L_0^{-1} L_1)^{1-\vartheta} = L_0 \#_{1-\vartheta} L_1.$$

The most interesting case is $\vartheta = 1/2$, where the norm is given by the geometric mean of L_0 and L_1.

This property holds in more generality. In fact, let X and Y be two n-dimensional Hilbert spaces obtained endowing $\mathbb{R}^n$ with the scalar products $\langle \cdot, \cdot \rangle_X$ and $\langle \cdot, \cdot \rangle_Y$, respectively. Choosing a basis or $\mathbb{R}^n$, there exist A and B positive definite such that $\langle u, v \rangle_X = u^T A v$ and $\langle u, v \rangle_Y = u^T B v$, where u and v are expressed in the given basis. Arioli and Loghin [5] have observed that the scalar product $u^T (A\#_t B) v$ induces an interpolation norm on the space $[X, Y]_t$ (see also [38]).

The same construction can be used to generate norm of interpolation spaces between finite dimensional subspaces of generic Sobolev spaces with applications to preconditioners of the Poincaré–Steklov operator or boundary preconditioners for the biharmonic operator.

The computational problems related to this applications are reduced to the computation of $(A\#_t B)v$, where A and B are usually of large dimension and sparse and v is a vector.

7. NUMERICAL EXPERIMENTS

We present some numerical tests to illustrate the behaviour of the algorithms presented in the paper in finite precision arithmetic. The tests have been performed using MATLAB on a computer with

DOI: 10.1002/nla

Pentium Dual-Core T4300 processor. The scripts of the tests are available at the author's personal web page. Regarding Algorithm 5.1, based on the rational minimax approximation, we have used the code of [21]. The implementation of the best algorithms for the matrix geometric mean has been added to the Matrix Means Toolbox [12].

As a measure of accuracy we consider the relative error $\|\widetilde{G} - G\|/\|G\|$, where $\widetilde{G}$ is the computed value of the geometric mean, while G is the *exact* solution obtained using the variable precision arithmetic and rounded to double precision.

The matrices used in the tests are obtained either by the `random` function of the Matrix Means Toolbox [12], which allows one to get positive definite matrices of any size, having norm one and with prescribed condition number; or using the `gallery` function of MATLAB, from which we have selected ten matrices which are positive definite.

Test 1

We compare the behaviour of the algorithms showing how the convergence of the iterative algorithms and quadrature formulae depends on the eigenvalues of BA^{-1}. In fact, for the averaging iteration, the convergence depends on the value ρ of (21), while for the quadrature the linear parameter of convergence depends on M/m, where M and m are the largest and the smallest eigenvalues of BA^{-1} (after a proper scaling such that $Mm = 1$).

We consider for $n = 10$, the matrices $A = \alpha I + E \in \mathbb{C}^{n\times n}$, where E is the matrix made of ones and $\alpha > 0$, and $B = (b_{ij}) \in \mathbb{C}^{n\times n}$ such that $b_{ij} = \sigma^{|i-j|}$, which is positive definite for $0 < |\sigma| < 1$. These two matrices, in the MATLAB gallery, are called Pei and KMS (Kac-Murdock-Szëgo), respectively.

For $\alpha = 0.56$ and $\sigma = 1 - 10^{-3}$, we have $\rho \approx 0.94$, while for $\alpha = 0.56$ and $\sigma = 1 - 10^{-6}$ we have $\rho \approx 0.998$. For these values of α and σ, we compute an approximation of $A\#B$ in double precision using the different algorithms and monitor the relative error at each step for the iterations and for an increasing number of nodes for quadrature rules. The results for the algorithms based on iterations are drawn in Figure 3, where the algorithms considered are: the averaging algorithm (`avera`), namely iteration (11); the sign iteration (`sign`), namely Algorithm 4.1a, the three-term recurrence (`tterm`), namely Algorithm 4.1b, and the polar decomposition algorithm (`polar`), namely Algorithm 4.2. For both sign iteration and three-term recurrence, the determinantal scaling has been used, while the polar factor is computed by Newton's method with the scaling parameter (29).

As ρ becomes large, the number of steps needed by the averaging iteration without scaling increases and so the scaling technique is recommended. The polar decomposition algorithm, in this example shows a faster convergence. Nevertheless, using a spectral scaling for the sign method would make it as fast as the polar decomposition, but with the further requirement of the approximation of two spectral radii.

Test 2

We repeat Test 1 with $\alpha = 0.56$ and $\sigma = 1 - 10^{-2}$ for which $M/m \approx 149$ and with $\alpha = 0.56$ and $\sigma = 1 - 10^{-3}$ for which $M/m \approx 10^3$; where now we consider the quadrature methods. The results are drawn in Figure 4, where we consider the Gauss-Chebyshev quadrature (`gauss`), namely Algorithm 4.3, the same algorithm after a scaling on A and B such that $Mm = 1$ (`gauss scaled`) and the rational minimax approximation algorithm (`minimax`), namely Algorithm 5.1.

All methods show linear convergence, but the one based on rational minimax is much more effective, as soon as M/m increases. Unfortunately, while for the Gaussian quadrature it is possible to double the number of nodes, re-using the computed values and deriving an automatic quadrature algorithm; for the quadrature based on the rational minimax approximation it is not simple to predict the exact number of nodes needed, moreover some complex arithmetic is needed.

Even if these algorithms are slower than the algorithms considered in Test 1, they are the only algorithms presented in the paper which are suited for the problem $(A\#B)v$, where A and B are of large size and sparse and v is a vector. In fact, both give an expression of $A\#B$ of the type

$$A\#B = \gamma B \left(\sum_{k=0}^{N} (\alpha_k A + \beta_k B)^{-1} \right) A.$$

DOI: 10.1002/nla

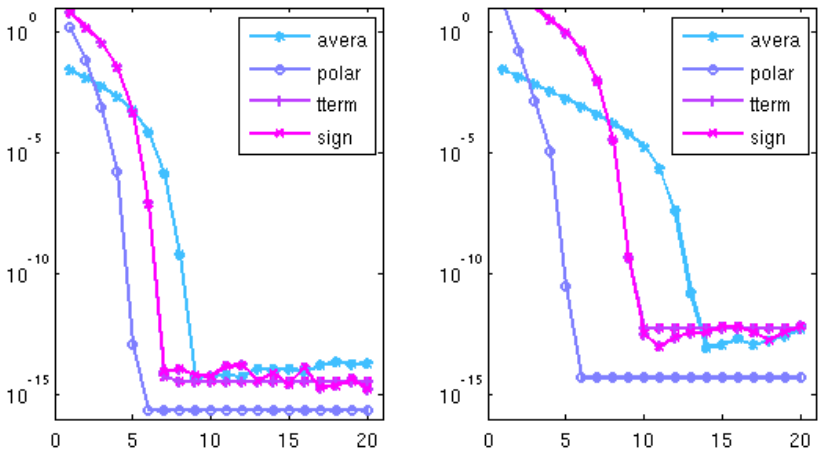

Figure 3. Convergence behaviour of algorithms based on iterations for $A = 0.56I + E$, where E is the matrix whose entries are all one, and $B = (\sigma^{|i-j|})_{i,j}$ with $\sigma = 1 - 10^{-3}$ (left figure) or $\sigma = 1 - 10^{-6}$ (right figure).

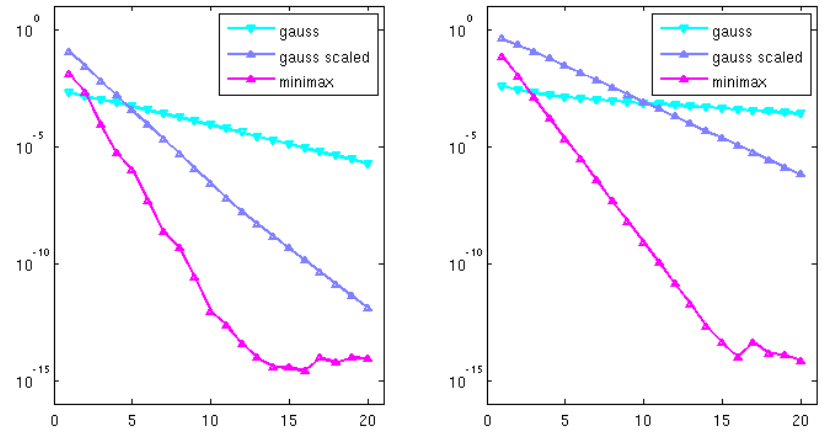

Figure 4. Convergence behaviour of algorithms based on quadrature formulae for $A = 0.56I + E$, where E is the matrix whose entries are all one, and $B = (\sigma^{|i-j|})_{i,j}$ with $\sigma = 1 - 10^{-2}$ (left figure) or $\sigma = 1 - 10^{-3}$ (right figure).

with $\alpha_k, \beta_k > 0$. The computation of $(A\#B)v$ is then reduced to matrix-vector multiplications and solutions of linear systems of the type $(\alpha_k A + \beta_k B)x = b$, where the matrix coefficient is large, sparse and positive definite.

As N increases, the approximation error in computing $(A\#B)v$ follows the same trend as the approximation error in computing $A\#B$ in Figure 4.

Test 3

We test the accuracy of the algorithms considered in the paper. We compare the Cholesky-Schur (`sharp`) algorithm 4.1 of Section 3; the polar decomposition algorithm 5.2 (`polar`) and the three iteration obtained by the vanilla averaging iteration in (11) (`avera`) and its scaled variants, namely the sign algorithm 5.1a (`sign`) and the three-terms recurrence algorithm 5.1b (`tterm`). We stop

DOI: 10.1002/nla

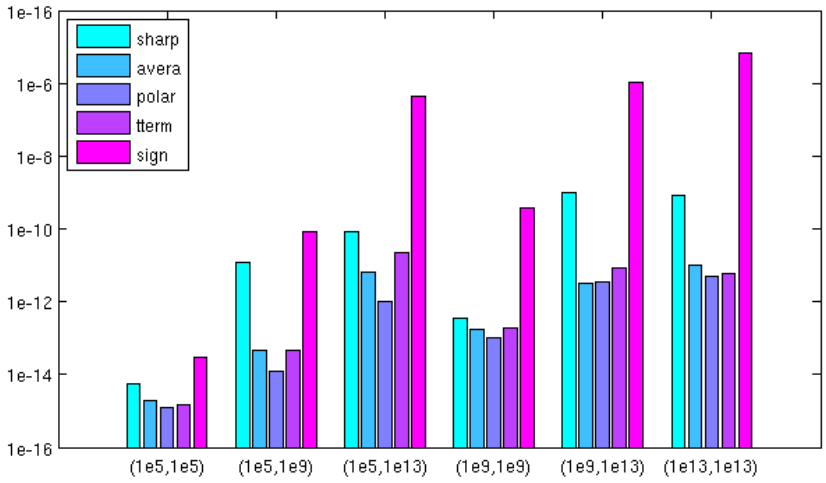

Figure 5. Accuracy obtained using different matrix mean algorithms with six random couples of 8×8 matrices with condition number between 10^5 and 10^{13}.

the iterations one step after the relative difference of two consecutive iterates becomes smaller than 10^{-10} or after a fixed number of steps.

We do not consider the algorithms based on quadrature since they require sometimes too many steps and this spoils convergence.

We generate positive definite matrices with condition numbers between 10^5 and 10^{13} using the `random` command in the Matrix Means Toolbox and compute the geometric mean of the couples. The accuracy is measured by the relative error $\|\widetilde{G} - G\|/\|G\|$, where $\widetilde{G}$ is the value obtained by the algorithm, while G is the geometric mean correct up to the machine precision, obtained using variable precision arithmetic. In Figure 5, which is obtained with $n = 8$, we compare the accuracy of the algorithms. The quality of the approximation is comparable for the polar, the unscaled averaging iteration and the three-terms iterations, with slightly better results for the polar decomposition algorithm. Similar results are obtained repeating the computation with different random matrices.

As a second test, we consider the previous six couples together with all 55 different couples that can be formed with ten 7×7 matrices chosen from the MATLAB gallery. For each couple we compute the relative errors obtained with the considered algorithms. Then for any algorithm we sort these errors by nondecreasing order obtaining a nondecreasing function from $\{1, \dots, 61\}$ to $\mathbb{R}$. In Figure 6, we plot these functions yielding a kind of measure of "cumulated accuracy" of the methods. As one can see, the methods are comparable, but the algorithm based on the polar decomposition is apparently the best, followed by the Cholesky-Schur algorithm. The figure is split in two subplots to get a more clear visualization.

8. CONCLUSIONS

We have studied the computational issues related to the matrix geometric mean of two positive definite matrices A and B, surveying the numerical algorithms for computing $A\#B$. We have analysed many algorithms, some of which had not yet been considered in the literature for this specific problem. The algorithms are either based on the Schur decomposition or are iterations or quadrature formulae converging to the geometric mean. A very nice fact is that all iterations and all quadrature formulae we were able to find are related to the two important rational approximation of $z^{-1/2}$, namely, the Padé approximation and the rational relative minimax approximation.

DOI: 10.1002/nla

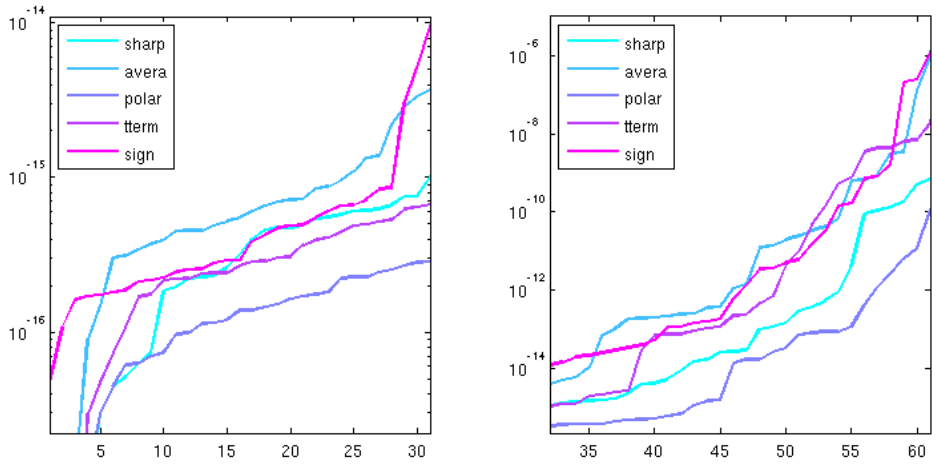

Figure 6. Cumulated accuracy obtained using different matrix mean algorithm on 61 examples.

We have observed that the Padé approximation requires a much higher degree than the rational relative minimax approximation to get the same accuracy on typical problems. On the other hand, the advantage of the Padé approximation is that there exists a recurrence relation between the $[2^k, 2^k - 1]$ Padé approximants to $z^{-1/2}$ and this recurrence leads to a quadratically convergent algorithm which outperforms the one based on rational minimax approximation. The quadratically convergent iterations can be scaled to get very efficient algorithms, as the one based on the polar decomposition of a suitable matrix.

Our preferred algorithm for computing the matrix geometric mean is the one based on the scaled polar decomposition, but excellent results are obtained also by the one based on the Schur decomposition, namely the Cholesky-Schur algorithm, and the one based on the averaging iteration. Notice that, among the three, the polar decomposition algorithm is the one requiring the smallest computational effort. Nevertheless, for very large and sparse matrices it may be necessary to use a quadrature formula as the rational minimax approximation, even if it is not clear how to efficiently implement an algorithm based on these techniques, in comparison with techniques based on Krylov subspace methods. A better understanding of the problem of computing $(A\#B)v$ with A and B large and sparse matrices and v a vector is needed and is the topic of a future work.

We wonder if some kind of approximate recurrence could be found for the rational relative minimax approximation. Moreover, the algorithms based on the Padé approximation benefit considerably by the scaling technique. One might wonder what is the interpretation of the scaling in terms of the approximation and if it is possible to get a "scaled rational minimax" approximation in order to accelerate the convergence.

Another issue is related to the equivalence of methods. For this problem we have found the equivalence between a Newton method, a Padé approximation, the Cyclic Reduction and a Gaussian quadrature. We wonder if this intimate connection is true in more general settings. For instance, it would be nice to see the Cyclic Reduction algorithm as a function approximation algorithm.

ACKNOWLEDGMENTS

The author wish to thank the GNCS of the Istituto Nazionale di Alta Matematica for supporting his research and the referees for their accurate revisions and valuable suggestions. He wish to thank George Trapp who kindly sent him some classical papers about the matrix geometric mean and Elena Addis, a student who defended a thesis on these topics, who gave the remarkable quote

(freely translated from Italian) about the interpretation of the geometric mean as the mid-point of a geodesic:

> It fills of geometric meaning what of geometric had just the name.

REFERENCES

1. M. Abramowitz and I. A. Stegun. *Handbook of mathematical functions*. Dover, 2007.
2. W. N. Anderson, T. D. Morley and G. Trapp. Ladder networks, fixpoints, and the geometric mean. *Circuit, Syst. Sig. Proc.*, 2:259–268, 1983.
3. W. N. Anderson and G. E. Trapp. Operator means and electrical networks. Proc. 1980 IEEE International Symposium on Circuits and Systems.
4. T. Ando, C.-K. Li and R. Mathias. Geometric means. *Linear Algebra Appl.*, 385:305–334, 2004.
5. M. Arioli and D. Loghin. Discrete interpolation norms with applications. *SIAM J. Numer. Anal.*, 47(4):2924–2951, 2009.
6. V. Arsigny, P. Fillard, X. Pennec and N. Ayache. Geometric means in a novel vector space structure on symmetric positive-definite matrices. *SIAM J. Matrix Anal. Appl.*, 29(1):328–347, 2006/07.
7. F. Barbaresco. New Foundation of Radar Doppler Signal Processing based on Advanced Differential Geometry of Symmetric Spaces: Doppler Matrix CFAR & Radar Application. In *International Radar Conference 2009*, Bordeaux, France, October 2009.
8. A. Y. Barraud. Produit étoile et fonction signe de matrice. Application à l'équation de Riccati dans le cas discret. *RAIRO Automat.*, 14(1):55–85, 1980. With comments by P. Bernhard.
9. P. G. Batchelor, M. Moakher, D. Atkinson, F. Calamante and A. Connelly. A rigorous framework for diffusion tensor calculus. *Magn. Reson. Med.*, 53:221–225, 2005.
10. R. Bhatia. *Positive definite matrices*. Princeton Series in Applied Mathematics. Princeton University Press, Princeton, NJ, 2007.
11. D. A. Bini and B. Iannazzo. Computing the Karcher mean of symmetric positive definite matrices. *Linear Algebra Appl.*, 438(4):1700–1710, 2013.
12. D. A. Bini and B. Iannazzo. The Matrix Means Toolbox. http://bezout.dm.unipi.it/mmtoolbox.
13. D. A. Bini, B. Meini and B. Iannazzo. *Numerical Solution of Algebraic Riccati Equations.* Society for Industrial and Applied Mathematics (SIAM), Philadelphia, PA, 2012.
14. D. A. Bini, B. Meini and F. Poloni. An effective matrix geometric mean satisfying the Ando-Li-Mathias properties. *Math. Comp.*, 79(269):437–452, 2010.
15. R. Byers. Solving the algebraic Riccati equation with the matrix sign function. *Linear Algebra Appl.*, 85:267–279, 1987.
16. M. M. Chawla and M. K. Jain. Error estimates for Gauss quadrature formulas for analytic functions. *Math. Comp.*, 22:82–90, 1968.
17. F. Di Benedetto and C. Estatico. Shift-invariant approximations of structured shift-variant blurring matrices. *Numer. Algorithms*, 62(4):615–635, 2013.
18. J. van den Eshof, A. Frommer, T. Lippert, K. Schilling and H. A. van der Vorst. Numerical methods for the QCD overlap operator. I. Sign-function and error bounds. *Computer Physics Commun.*, 146(2):203–224, 2002.
19. M. Fiedler and V. Pták, A new positive definite geometric mean of two positive definite matrices. *Linear Algebra Appl.*, 251:1–20, 1997.
20. F. Greco, B. Iannazzo and F. Poloni. The Padé iterations for the matrix sign function and their reciprocals are optimal. *Linear Algebra Appl.*, 436(3):472–477, 2012.
21. N. Hale, N. J. Higham and L. N. Trefethen. Computing $\mathbf{A}^\alpha$, $\log(\mathbf{A})$, and related matrix functions by contour integrals. *SIAM J. Numer. Anal.*, 46(5):2505–2523, 2008.
22. N. J. Higham. *Functions of Matrices:Theory and computation*. Society for Industrial and Applied Mathematics (SIAM), Philadelphia, PA, 2008.
23. N. J. Higham and L. Lin. A Schur-Padé algorithm for fractional powers of a matrix. *SIAM J. Matrix Anal. Appl.*, 32(3):1056–1078, 2011.
24. N. J. Higham, D. S. Mackey, N. Mackey and F. Tisseur. Functions preserving matrix groups and iterations for the matrix square root. *SIAM J. Matrix Anal. Appl.*, 26(3):849–877, 2005.
25. J. Holbrook. No dice: a deterministic approach to the Cartan centroid. *J. Ramanujan Math. Soc.*, 27(4):509–521, 2012.
26. B. Iannazzo. *Numerical Solution of Certain Nonlinear Matrix Equations*. PhD Thesis, Università degli studi di Pisa, Pisa, Italy, 2007.
27. B. Iannazzo. *On the Newton method for the matrix pth root*. *SIAM J. Matrix Anal. Appl.*, 28(2):503–523, 2006.
28. B. Iannazzo and C. Manasse. A Schur logarithmic algorithm for fractional powers of matrices. *SIAM J. Matrix Anal. Appl.*, 34(2):794–813, 2013.
29. B. Iannazzo and B. Meini. The palindromic cyclic reduction and related algorithms. Technical report, september 2013. To appear in Calcolo.
30. B. Iannazzo and B. Meini. Palindromic matrix polyomials, matrix functions and integral representations. *Linear Algebra Appl.*, 434(1):174–184, 2011.
31. C. Kenney and A. J. Laub. On scaling Newton's method for polar decomposition and the matrix sign function. *SIAM J. Matrix Anal. Appl.*, 13(3):698–706, 1992.
32. A. Kiełbasiński and K. Ziętak. Numerical behaviour of Higham's scaled method for polar decomposition. *Numer. Algorithms*, 32(2-4):105–140, 2003.
33. F. Kubo and T. Ando. Means of positive linear operators. *Math. Ann.*, 246(3):205–224, 1979/80.

DOI: 10.1002/nla

34. J. D. Lawson and Y. Lim. The geometric mean, matrices, metrics, and more. *Amer. Math. Monthly*, 108(9):797–812, 2001.
35. D. Le Bihan, J.-F. Mangin, C. Poupon, C. A. Clark, S. Pappata, N. Molko and H. Chabriat. Diffusion tensor imaging: Concepts and applications. *J. Magn. Reson. Imaging*, 13:534–546, 2001.
36. Y. Lim and M. Pálfia. Matrix power means and the Karcher mean. *J. Funct. Anal.*, 262(4):1498–1514, 2012.
37. Y. Lim and M. Pálfia. Weighted deterministic walks for the squares mean on Hadamard spaces. *Bull. Lond. Math. Soc.*, 46(3):561–570, 2014.
38. J.-L. Lions and E. Magenes. *Problèmes aux limites non homogènes et applications. Vol. 1. (French)*. Travaux et Recherches Mathématiques, No. 17, Dunod, Paris, 1968.
39. M. Moakher. On the averaging of symmetric positive-definite tensors. *J. Elasticity*, 82(3):273–296, 2006.
40. N. Nakamura. Geometric means of positive operators. *Kyungpook Math. J.*, 49(1):167–181, 2009.
41. Y. Nakatsukasa and N. J. Higham. Backward stability of iterations for computing the polar decomposition. *SIAM J. Matrix Anal. Appl.*, 33(2):460–479, 2012.
42. G. Pusz and S. L. Woronowicz. Functional calculus for sesquilinear forms and the purification map. *Rep. Math. Phys.*, 8:159–170, 1975.
43. M. Raïssouli and F. Leazizi. Continued fraction expansion of the geometric matrix mean and applications. *Linear Algebra Appl.*, 359:37–57, 2003.

DOI: 10.1002/nla