

COMPUTING THE WEIGHTED GEOMETRIC MEAN OF TWO LARGE-SCALE MATRICES AND ITS INVERSE TIMES A VECTOR

MASSIMILIANO FASI* AND BRUNO IANNAZZO†

Abstract. We investigate different approaches for computing the action of the weighted geometric mean of two large-scale positive definite matrices on a vector. We derive and analyze several algorithms, based on numerical quadrature and on the Krylov subspace, and compare them in terms of convergence speed and execution time. By exploiting an algebraic relation between the weighted geometric mean and its inverse, we show how these methods can be used to efficiently solve large linear systems whose coefficient matrix is a weighted geometric mean. According to our experiments, some of the algorithms proposed in both families are suitable choices for black-box implementations.

1. Introduction. The weighted geometric mean of parameter t of two positive numbers, say a and b , is defined as $a^{1-t}b^t$ for any $t \in [0, 1]$. This definition covers as a special case the standard geometric mean \sqrt{ab} , arising for $t = 1/2$. The extension of this concept to positive definite matrices is not trivial, but there is large agreement that the *right* generalization, for $A, B \in \mathbb{C}^{n \times n}$ (Hermitian) positive definite and $t \in [0, 1]$, is

$$(1) \quad A\#_t B = A(A^{-1}B)^t = A(B^{-1}A)^{-t},$$

which turns out to be positive definite and is called the *matrix weighted geometric mean* of A and B . The reasons behind this choice and the properties of the matrix weighted geometric are discussed by Bhatia [11, Ch. 4] and Lawson and Lim [41]. Relevant applications of the weighted geometric mean of two dense matrices of moderate size, along with algorithms for its computations, can be found in the survey [35].

Here we are interested in the approximation of $(A\#_t B)v$ and $(A\#_t B)^{-1}v$, where $v \in \mathbb{C}^n$ and A, B are large and sparse. These problems arise in a preconditioning technique for some domain decomposition methods and in methods for the biharmonic equation [4, 5, 6], and in the clustering of signed complex networks [44]. The geometric mean of large-scale matrices appears also in image processing [22].

In particular, we want to avoid the explicit computation of the matrix function $A\#_t B$, which may be unduly slow or even practically infeasible, for A and B large enough. We explore two classes of methods to achieve this goal, namely numerical quadrature of certain integral representations of the matrix function Z^{-t} for $t \in (0, 1)$, and Krylov subspace methods for computing the product of a matrix function and a vector.

It is well known that the geometric mean $A\#B := A\#_{1/2}B$ [2, 3, 12, 46] (the weighted geometric mean with weight $t = 1/2$) has several nice integral representations (see [37] and the references therein). In particular, the formula

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

*School of Mathematics, The University of Manchester, Oxford Road, Manchester M13 9PL, UK (massimiliano.fasi@manchester.ac.uk).

†Dipartimento di Matematica e Informatica, Università di Perugia, Via Vanvitelli 1, 06123 Perugia, Italy (bruno.iannazzo@dmi.unipg.it). The work of this author was supported by the Istituto Nazionale di Alta Matematica, INdAM-GNCS Project 2015.

is well suited for Gaussian quadrature with respect to the weight function $(1-z^2)^{-1/2}$, and is considered in comparison with other algorithms for $A\#B$ by Iannazzo [35]. We generalize this approach to the matrix weighted geometric mean.

Quadrature formulae are particularly attractive in the large-scale case, since they produce an approximation of the form

$$(2) \quad (A\#_t B)v \approx \sum_{i=0}^N w_i A(r_i A + s_i B)^{-1} Bv,$$

where the w_i s are the weights of the quadrature and the r_i s and the s_i s are parameters obtained from the nodes of the quadrature. By exploiting the identity $(A\#_t B)^{-1} = B^{-1}(B\#_t A)A^{-1}$, a similar approximation for the inverse of the geometric mean, namely

$$(3) \quad (A\#_t B)^{-1}v \approx \sum_{i=0}^N w_i (r_i B + s_i A)^{-1}v,$$

can be easily derived. The problem is thus reduced to the solution of linear systems and the evaluation of matrix-vector products. Moreover, if r_i and s_i are positive for all i , then the matrix coefficients of these linear systems are positive definite, being convex combinations of the positive definite matrices A and B , and we say that the quadrature formula *preserves the positivity structure* of the problem.

We consider and analyze three quadrature formulae for $A\#_t B$. The first two are obtained from integral representations of the inverse of real powers [14, 23], by exploiting the fact that $A\#_t B = A(B^{-1}A)^{-t}$. The third is based on a clever conformal mapping [30], which achieves fast convergence speed but does not preserve the positivity structure of the problem for $t \neq 1/2$.

Regarding Krylov subspace methods, we adapt to our problem standard techniques for the approximation of $f(Z^{-1}Y)v$, where Z and Y are large-scale matrices. In this case, the usual way to proceed is to consider a projection of the matrix onto a small Krylov subspace and thereby reduce the original problem to a small sized one. Since $(A\#_t B)v = A(B^{-1}A)^{-t}v$, the computation of $(A\#_t B)v$ reduces to that of $(B^{-1}A)^{-t}v$, which is well suited for the aforementioned techniques. For instance, when approximating $(B^{-1}A)^{-t}v$ by means of the Arnoldi method, we get the generalized Lanczos method [45, Ch. 15], which has been considered for $(A\#_t B)v$ in previous work [5, 4]. We revise the generalized Lanczos method and then investigate some more powerful Krylov subspace techniques such as the extended Krylov subspace method [21] and the rational Krylov subspace methods [48, 49, 50], with poles chosen according to the adaptive strategy by Güttel and Knizhnerman [29] or the rational Krylov fitting by Berljafa and Güttel [8]. We show that these methods, in most cases, outperform the generalized Lanczos algorithm. Prior to our work, rational Krylov methods have been considered for the computation of $(A\#B)v$, where the implementations are meant for and tested on sparse matrices of moderate size [15].

For the sake of generality, in describing the Krylov subspace techniques, we work with the more general problem $Af(A^{-1}B)v$, where A is positive definite, B is Hermitian and f is the matrix extension of a real positive function. Our implementations, tailored for the function $f(z) = z^{-t}$, are well suited to the computation of $(A\#_t B)^{-1}v$, and could, in principle, be used for any choice of the function f .

The paper is organized as follows. In the next section we give some notation and preliminary results. Quadrature methods for the weighted geometric mean are

discussed in Section 3, while Section 4 is devoted to Krylov subspace methods. The application of these techniques to the solution of the linear system $(A\#_t B)y = v$ is discussed in Section 5, and an experimental comparison is provided in Section 6. In the final section, we draw the conclusions.

2. Notation and preliminaries. Throughout the paper we denote by I_n the identity matrix of size n , omitting the size when there is no ambiguity. The set \mathbb{R}^+ will denote the positive real numbers, while $\overline{\mathbb{R}} = \mathbb{R} \cup \{\pm\infty\}$. We will denote by $\sigma(A)$ the spectrum of the square matrix A . Throughout the paper, we consider the spectral norm $\|A\| = \max_{\|x\|_2=1} \|Ax\|_2$. For $x_1, \dots, x_n \in \mathbb{C}$, we denote by $\text{diag}(x_1, \dots, x_n)$ the $n \times n$ diagonal matrix with x_1, \dots, x_n on the main diagonal. Let $\mathcal{V} \subset \mathbb{C}^n$ be a subspace, and $A \in \mathbb{C}^{n \times n}$, by $A\mathcal{V}$ we denote the subspace $\{Av : v \in \mathcal{V}\}$.

Let $A \in \mathbb{C}^{n \times n}$ be diagonalizable with eigenvalues in $\Omega \subset \mathbb{C}$ and let $f : \Omega \rightarrow \mathbb{C}$. If $M^{-1}AM = \text{diag}(\lambda_1, \dots, \lambda_n)$, then $f(A) := M \text{diag}(f(\lambda_1), \dots, f(\lambda_n))M^{-1}$. Note that if A is Hermitian, then $f(A)$ is Hermitian as well. This definition can be extended to nondiagonalizable matrices [33, Def. 1.2], and is independent of the choice of M .

We have the *similarity invariance of matrix functions*, that is, if $f(A)$ is well defined, then $f(KAK^{-1}) = Kf(A)K^{-1}$, for any invertible K . We give now a well-known property regarding an expression commonly encountered when dealing with functions of Hermitian matrices.

LEMMA 2.1. *Let $f : \mathcal{U} \rightarrow \mathbb{R}^+$, with \mathcal{U} subset of \mathbb{R} . For any $A \in \mathbb{C}^{n \times n}$ positive definite and $B \in \mathbb{C}^{n \times n}$ Hermitian, such that $\sigma(A^{-1}B) \subset \mathcal{U}$, the matrix $Af(A^{-1}B)$ is Hermitian positive definite.*

Proof. Note that $f(A^{-1}B)$ is well defined, since $A^{-1}B$ is diagonalizable with spectrum in \mathcal{U} . Because of the similarity invariance of matrix functions, we have that $Af(A^{-1}B) = A^{1/2}f(A^{-1/2}BA^{-1/2})A^{1/2}$. The matrix $A^{-1/2}BA^{-1/2}$ is Hermitian and diagonalizable with real eigenvalues in \mathcal{U} , thus $T = f(A^{-1/2}BA^{-1/2})$ is Hermitian with positive eigenvalues and the same holds for $Af(A^{-1}B)$, which is obtained from T through a congruence. \square

If A and B are positive definite, then $\sigma(A^{-1}B) \subset \mathbb{R}^+$. Thus, the previous lemma, applied to $f(z) = z^t$, with $\mathcal{U} = \mathbb{R}^+$, shows that $A\#_t B = A(A^{-1}B)^t$ is positive definite. Using other properties of matrix functions one obtains the following equivalent expressions:

$$(4) \quad \begin{aligned} A\#_t B &= A(A^{-1}B)^t = A(B^{-1}A)^{-t} = B(A^{-1}B)^{t-1} = B(B^{-1}A)^{1-t}, \\ &= (BA^{-1})^t A = (AB^{-1})^{-t} A = (BA^{-1})^{t-1} B = (AB^{-1})^{1-t} B. \end{aligned}$$

Another useful property of the weighted geometric mean is

$$(5) \quad (A\#_t B)^{-1} = B^{-1}(B\#_t A)A^{-1},$$

which follows from an algebraic manipulation of the formulae in (4)

$$(A\#_t B)^{-1} = ((BA^{-1})^{t-1} B)^{-1} = B^{-1}(BA^{-1})^{1-t} AA^{-1} = B^{-1}(B\#_t A)A^{-1}.$$

3. Quadrature methods. In this section, we exploit the formula $A\#_t B = A(B^{-1}A)^{-t}$ to obtain three quadrature formulae for $A\#_t B$ from the corresponding quadrature formulae for the inverse real power function z^{-t} .

In the next subsection we describe and analyze two integral representations for z^{-t} and in Sections 3.2 and 3.3 we discuss their application to the matrix weighted geometric mean. Finally, in Section 3.4 we adapt an algorithm based on a conformal map transformation to the matrix weighted geometric mean.

3.1. Integral representations for z^{-t} . Since $A\#_t B = A(B^{-1}A)^{-t}$, useful integral representations of the matrix weighted geometric mean can be obtained from the representations of the fractional inverse power function. The function $\mathbb{C} \setminus [-\infty, 0] \ni z \rightarrow z^{-t}$ for $t \in (0, 1)$ is a Markov function [10, p. 116], which can be written as

$$(6) \quad z^{-t} = \frac{\sin(\pi t)}{\pi} \int_0^\infty \frac{dx}{x^t(x+z)}, \quad 0 < t < 1.$$

To rewrite this integral in a more practical form, we exploit the Cayley transform $\mathcal{C}(x) = \frac{1-x}{1+x}$, which sends the positive real numbers to the interval $(-1, 1)$.

The variable transformation $s = \mathcal{C}(x)$ gives

$$(7) \quad z^{-t} = \frac{2\sin(\pi t)}{\pi} \int_{-1}^1 (1-s)^{-t}(1+s)^{t-1} \frac{ds}{(1-s) + (1+s)z}.$$

On the other hand, by applying the transformation $s = -\mathcal{C}(x^{1-t})$ to the integral in (6), we obtain

$$(8) \quad z^{-t} = \frac{2\sin(\pi(1-t))}{\pi(1-t)} \int_{-1}^1 (1-s)^{\frac{2t-1}{1-t}} \frac{ds}{(1+s)^{\frac{1}{1-t}} + (1-s)^{\frac{1}{1-t}}z},$$

which has been considered in a similar form in order to compute the p th root [14].

Both (7) and (8) are integrals of the form

$$\int_{-1}^1 (1-s)^\alpha (1+s)^\beta f(s) ds,$$

with $(\alpha, \beta) = (-t, t-1)$ and $(\alpha, \beta) = (\frac{2t-1}{1-t}, 0)$, respectively. These integrals, for $\alpha, \beta > -1$, can be approximated by using Gaussian quadrature with respect to the weight

$$(9) \quad \omega_{\alpha, \beta}(s) = (1-s)^\alpha (1+s)^\beta, \quad s \in [-1, 1].$$

These formulae are known as the Gauss–Jacobi quadrature formulae [47, Sec. 4.8].

A nice feature of the Gauss–Jacobi quadrature applied to the integral (7) is that the function to be integrated with respect to the weighted measure, namely

$$(10) \quad f_{1,z}(s) = \frac{1}{1-s + (1+s)z},$$

is analytic on $[-1, 1]$, for any $z \in \mathbb{C} \setminus (-\infty, 0)$, and thus the convergence of the quadrature formulae is exponential.

In particular, given a function f analytic on the interval $[-1, 1]$, for the error of the Gaussian quadrature with nodes s_i and weights w_i for $i = 0, \dots, N-1$, we have the estimate [25, 53]

$$(11) \quad |R_N(f)| = \left| \int_{-1}^1 f(x)\omega(x)dx - \sum_{i=0}^{N-1} w_i f(s_i) \right| \leq 4\mu_0 \frac{1}{\rho^{2N}} \left(\frac{\rho^2}{\rho^2 - 1} \right) \max_{x \in \Gamma} |f(x)|,$$

where $\mu_0 = \int_{-1}^1 \omega(x)dx$ and the curve Γ is an ellipse with foci -1 and 1 and sum of the semimajor and semiminor axes ρ , entirely enclosed (with its interior part) in the domain of analyticity of f .

When f is analytic on $[-1, 1]$, we may assume that $\rho > 1$. Hence, for any ellipse contained in the region of analyticity corresponding to ρ , the convergence of the quadrature formula is exponential with rate γ such that $1/\rho^2 < \gamma < 1$. On the other hand, for the integral (8), the integrand is

$$(12) \quad f_{2,z}(s) = \frac{1}{(1+s)^{\frac{1}{1-t}} + (1-s)^{\frac{1}{1-t}} z},$$

which is analytic on $[-1, 1]$ for any $z \in \mathbb{C} \setminus (-\infty, 0)$ only if t is of the form $(p-1)/p$, with $p \in \mathbb{N}$. When $1/(1-t)$ is not an integer, the integrand (12) has two branch points at -1 and 1 , which makes the use of this second quadrature method less attractive for our purposes. Nevertheless, in some cases the Gauss–Jacobi quadrature applied to (8) converges faster than the same method applied to (7).

We analyze the convergence just for $z \in \mathbb{R}^+$, because we want to apply the formulae to diagonalizable matrices having positive real eigenvalues and, in this case, the convergence of the quadrature formulae for the matrix follows from that of the same formulae for its eigenvalues.

Convergence for the integrand $f_{1,z}(s)$. Let us start by considering the quadrature formula for $f_{1,z}(s)$, which has only one pole at $\zeta = 1/\mathcal{C}(z)$. The function $1/\mathcal{C}(z)$ maps the half line $(0, \infty)$ to $\overline{\mathbb{R}} \setminus [-1, 1]$, thus we are guaranteed that the pole lies outside the interval $[-1, 1]$ for any $z > 0$ and that the convergence result for analytic functions applies.

If $z \in (0, \infty)$, then it is easy to identify the smallest ellipse not contained in the domain of analyticity of $f_{1,z}(s)$ as the one passing through ζ . The real semiaxis of such an ellipse has length $|\zeta|$ and its imaginary semiaxis has length $\sqrt{\zeta^2 - 1}$, thus, the sum of its semiaxes is

$$(13) \quad \begin{aligned} \rho^{(1)}(z) &= |\zeta| + \sqrt{\zeta^2 - 1} = \frac{1}{|\mathcal{C}(z)|} + \sqrt{\frac{1}{\mathcal{C}(z)^2} - 1} \\ &= \frac{|1+z| + 2\sqrt{z}}{|1-z|} = \frac{1+\sqrt{z}}{|1-\sqrt{z}|} = \frac{1}{|\mathcal{C}(\sqrt{z})|}, \end{aligned}$$

and hence a lower bound for the rate of convergence is $|\mathcal{C}(\sqrt{z})|^2$.

Convergence for the integrand $f_{2,z}(s)$. The convergence analysis for $f_{2,z}(s)$ is more problematic, since the function lacks analyticity at 1 and -1 when $1/(1-t) \notin \mathbb{N}$. For $t = (p-1)/p$, with $p \in \mathbb{N}$, the function $f_{2,z}(s)$ is rational and its poles are given by the solutions of the equation

$$(1 + \zeta)^p + (1 - \zeta)^p z = 0,$$

which are the p distinct points

$$(14) \quad \zeta_\ell = -\mathcal{C}\left(z^{1/p} e^{\frac{1}{p}i\pi(2\ell+1)}\right), \quad \ell = 0, \dots, p-1.$$

Since none of them lies on the interval $[-1, 1]$, the integrand is analytic there.

In order to get the rate of convergence of the quadrature formula, we consider the sum of the semiaxes of the smallest ellipse not contained in the domain of analyticity of $f_{2,z}(s)$.

PROPOSITION 3.1. *For any positive integer p , the smallest ellipse not contained in the domain of analyticity of $f_{2,z}(s)$ (defined in (12)), with $t = (p-1)/p$, passes*

through ζ_0 (defined in (14)) and the sum of its semiaxes is

$$(15) \quad \rho^{(2)}(z) = \frac{1 + z^{1/p} + \sqrt{2z^{1/p}(1 - \cos(\pi/p))}}{\sqrt{1 + z^{2/p} + 2z^{1/p} \cos(\pi/p)}}.$$

Proof. We know that the poles of $f_{2,s}(z)$ are $\zeta_\ell = -\mathcal{C}(\xi_\ell)$ with $\xi_\ell = z^{\frac{1}{p}} e^{\frac{2\ell+1}{p}i\pi}$, for $\ell = 0, \dots, p-1$.

We want to find the smallest sum of the semiaxes of an ellipse not including the points $\{\zeta_\ell\}$ in its interior part, and with foci 1 and -1 . If we denote by x the length of the major semiaxis of such an ellipse, then the sum of the length of the semiaxes is $\rho = x + \sqrt{x^2 - 1}$.

We know that the sum of the distances between a point of the ellipse and the foci is twice the major semiaxis. To find the major semiaxis of the ellipse passing through ζ_ℓ we can use the fact that

$$|\zeta_\ell - 1| + |\zeta_\ell + 1| = 2x_\ell,$$

which readily gives x_ℓ and thus ρ_ℓ .

Since $\zeta_\ell = -\mathcal{C}(\xi_\ell)$, we have

$$\zeta_\ell + 1 = \frac{2\xi_\ell}{\xi_\ell + 1}, \quad \zeta_\ell - 1 = \frac{-2}{\xi_\ell + 1}, \quad x_\ell = \frac{1}{2}(|\zeta_\ell + 1| + |\zeta_\ell - 1|) = \frac{|\xi_\ell| + 1}{|\xi_\ell + 1|},$$

from which, by using $|\xi_\ell| = z^{1/p}$ and $(|\xi| + 1)^2 - |\xi + 1|^2 = 2|\xi| - 2\operatorname{Re}\xi$, we get

$$\rho_\ell = x_\ell + \sqrt{x_\ell^2 - 1} = \frac{|\xi_\ell| + 1 + \sqrt{2|\xi_\ell| - 2\operatorname{Re}\xi_\ell}}{|\xi_\ell + 1|} = \frac{1 + z^{1/p} + \sqrt{2z^{1/p}(1 - \cos(\vartheta_\ell))}}{\sqrt{1 + z^{2/p} + 2z^{1/p} \cos(\vartheta_\ell)}},$$

where $\vartheta_\ell = \frac{2\ell+1}{p}\pi$. Now observe that ρ_ℓ decreases as $\cos(\vartheta_\ell)$ grows, and thus that the nearer ϑ_ℓ is to a multiple of 2π , the smaller is the value of ρ_ℓ . Noting that ϑ_0 is the nearest such value concludes the proof. \square

Hence, for $t = (p-1)/p$, we have a lower bound for the rate of convergence, namely $(1/\rho^{(2)}(z))^2$. For $t \neq (p-1)/p$, by lack of analyticity of the integrand, we cannot use these asymptotic results to study the convergence of the quadrature formula involving $f_{2,z}(s)$. Nonetheless, it appears that the formula converges also for values of t not of the type $(p-1)/p$.

Comparison. We can compare the bounds for the rates of convergence of the two quadrature formulae, namely $(1/\rho^{(1)}(z))^2$, with $\rho^{(1)}(z)$ defined as in (13); and $(1/\rho^{(2)}(z))^2$, with $\rho^{(2)}(z)$ given by (15), just for $t = (p-1)/p$. Since $\rho^{(1)}(1/z) = \rho^{(1)}(z)$ and $\rho^{(2)}(1/z) = \rho^{(2)}(z)$, we can restrict our attention to $z \geq 1$.

In a neighborhood of 1, the quadrature formula using $f_{1,z}(s)$ works better since $1/\rho^{(1)}(1) = 0$, while $1/\rho^{(2)}(1) > 0$.

On the other hand, as $z \rightarrow \infty$, we have

$$(16) \quad 1 - \left(\frac{1}{\rho^{(1)}(z)}\right)^2 \approx 4z^{-\frac{1}{2}}, \quad 1 - \left(\frac{1}{\rho^{(2)}(z)}\right)^2 \approx 2\sqrt{2(1 - \cos(\pi/p))}z^{-\frac{1}{2p}}.$$

and thus the second formula works better for large values of z .

Gauss–Jacobi quadrature and Padé approximation. Quadrature on Markov functions is related to Padé approximation. In particular, applying the Gauss–Jacobi quadrature to the integral in (7) yields the $[N - 1/N]$ Padé approximant to z^{-t} as $z \rightarrow 1$. We give a short proof of this property (see also the one given by Frommer, Güttel and Schweitzer [23]).

THEOREM 3.2. *The Gauss–Jacobi quadrature of (7) with N nodes coincides with the $[N - 1, N]$ Padé approximant to z^{-t} as $z \rightarrow 1$.*

Proof. The Gaussian quadrature formula with N nodes, say $\mathcal{J}_N(z)$, is a rational function of z whose numerator and denominator have degree at most $N - 1$ and exactly N , respectively.

We have that $f_{1,z}^{(k)}(s) = (-1)^k k! (z - 1)^k f_{1,z}^{k+1}(s)$ for $k \geq 0$. From the latter and using standard results on the remainder of Gaussian quadrature we have that there exists $\xi = \xi(z) \in (-1, 1)$ such that

$$z^{-t} - \mathcal{J}_N(z) = \frac{2 \sin(\pi t)}{\pi} \frac{f_{1,z}^{(2N)}(\xi)}{(2N)!} \langle P_N^{(-t, 1-t)}, P_N^{(-t, 1-t)} \rangle = c_n \frac{(z - 1)^{2N}}{(z - 1)\xi + (z + 1)},$$

where $P_N^{(\alpha, \beta)}$ is the N th Jacobi polynomial, $\langle \cdot, \cdot \rangle$ is the scalar product with respect to the weight (9) and c_n is a constant independent of z .

As $z \rightarrow 1$ we get that $z^{-t} - \mathcal{J}_N(z) = O((z - 1)^{2N})$ and thus $\mathcal{J}_N(z)$ is the $[N - 1, N]$ Padé approximant to z^{-t} . \square

3.2. Integral representations of $A\#_t B$. The integral representations in Section 3.1 for z^{-t} readily yield analogous representations for the matrix weighted geometric mean (through $A\#_t B = A(B^{-1}A)^{-t}$).

From the formula (7) we obtain

$$\begin{aligned} (17) \quad A\#_t B &= c_1 A \int_{-1}^1 (1 - s)^{-t} (1 + s)^{t-1} ((1 - s)I + (1 + s)B^{-1}A)^{-1} ds \\ &= c_1 A^{1/2} \int_{-1}^1 (1 - s)^{-t} (1 + s)^{t-1} ((1 - s)I + (1 + s)A^{1/2}B^{-1}A^{1/2})^{-1} ds \cdot A^{1/2}, \\ &= c_1 A \int_{-1}^1 (1 - s)^{-t} (1 + s)^{t-1} ((1 - s)B + (1 + s)A)^{-1} B ds, \end{aligned}$$

with $c_1 = \frac{2 \sin(\pi t)}{\pi}$, and the corresponding quadrature formula on $N + 1$ nodes gives

$$(18) \quad A\#_t B \approx S_{N+1}^{(1)} := \frac{2 \sin(\pi t)}{\pi} \sum_{i=0}^N w_i A ((1 - s_i)B + (1 + s_i)A)^{-1} B,$$

where the w_i s are the weights of the Gauss–Jacobi quadrature formula with $N + 1$ nodes and s_i s are the nodes, which belong to the interval $[-1, 1]$. Therefore, for $i = 0, \dots, N$, the matrix $(1 - s_i)B + (1 + s_i)A$ is positive definite.

On the other hand, from (8) we have

(19)

$$\begin{aligned}
A\#_t B &= c_2 A \int_{-1}^1 (1-s)^{\frac{2t-1}{1-t}} \left((1+s)^{\frac{1}{1-t}} I + (1-s)^{\frac{1}{1-t}} B^{-1} A \right)^{-1} ds \\
&= c_2 A^{1/2} \int_{-1}^1 (1-s)^{\frac{2t-1}{1-t}} \left((1+s)^{\frac{1}{1-t}} I + (1-s)^{\frac{1}{1-t}} A^{1/2} B^{-1} A^{1/2} \right)^{-1} ds \cdot A^{1/2}, \\
&= c_2 A \int_{-1}^1 (1-s)^{\frac{2t-1}{1-t}} \left((1+s)^{\frac{1}{1-t}} B + (1-s)^{\frac{1}{1-t}} A \right)^{-1} B ds,
\end{aligned}$$

with $c_2 = \frac{2 \sin(\pi(1-t))}{\pi(1-t)}$, and the corresponding quadrature formula with $N+1$ nodes gives

$$(20) \quad A\#_t B \approx S_{N+1}^{(2)} := \frac{2 \sin(\pi(1-t))}{\pi(1-t)} \sum_{i=0}^N w_i A \left((1+s_i)^{\frac{1}{1-t}} B + (1-s_i)^{\frac{1}{1-t}} A \right)^{-1} B.$$

Even in this case the matrices *to be inverted*, for $i = 0, \dots, N$, are positive definite.

3.3. Matrix convergence. In order to analyze the convergence of the quadrature formulae for the matrix weighted geometric mean, we consider the convergence of the quadrature formulae for (7) and (8) when applied to a Hermitian positive definite matrix C . In this case, the functions to be integrated are

$$f_{1,C}(s) = ((1-s)I + (1+s)C)^{-1} \quad \text{and} \quad f_{2,C}(s) = ((1+s)^{\frac{1}{1-t}} I + (1-s)^{\frac{1}{1-t}} C)^{-1},$$

whose domain of analyticity is the intersection of the domain of analyticity of the corresponding function applied to all the eigenvalues of C .

If $Q^* C Q = \text{diag}(\lambda_1, \dots, \lambda_n)$, with Q unitary, and the function to be integrated is analytic on $[-1, 1]$, then the error in the quadrature formulae with N nodes (defined in (11)), in the spectral norm, is

$$\|R_N(f_{k,C}(s))\| = \|\text{diag}(R_N(f_{k,\lambda_i}(s)))\| = \max_{i=1, \dots, n} \{|R_N(f_{k,\lambda_i}(s))|\}, \quad k = 1, 2,$$

and is ruled by the eigenvalue whose corresponding pole gives the smallest ellipse with foci 1 and -1 , enclosed in the domain of analyticity.

Convergence for the integrand $f_{1,C}(s)$. Let the eigenvalues of C be ordered so that $0 < \lambda_m = \lambda_1 \leq \lambda_2 \leq \dots \leq \lambda_{n-1} \leq \lambda_n = \lambda_M$. The infimum of the acceptable values for ρ (the ellipse parameter) is now obtained by minimizing the function $|\zeta| + \sqrt{\zeta^2 - 1}$ for $\zeta \in \sigma(C)$, where $\sigma(C)$ denotes the spectrum of C , so that the bound on the rate of convergence, in view of (13), is

$$\tau^{(1)}(C) = \max_{\lambda \in \sigma(C)} \frac{1}{(\rho^{(1)}(\lambda))^2} = \max_{\lambda \in \sigma(C)} |\mathcal{C}(\sqrt{\lambda})|^2 = \max\{|\mathcal{C}(\sqrt{\lambda_m})|^2, |\mathcal{C}(\sqrt{\lambda_M})|^2\},$$

since the function $|\mathcal{C}(\sqrt{\lambda})|$ is monotonically decreasing in $(0, 1)$ and monotonically increasing in $(1, \infty)$.

Since C is positive definite, its condition number in the 2-norm, denoted by $\kappa := \mu_2(C)$, is λ_M/λ_m . If we further assume that $\lambda_M \lambda_m = 1$, then $\kappa = \lambda_M^2 = 1/\lambda_m^2$ and since $|\mathcal{C}(\sqrt{\lambda_m})| = |\mathcal{C}(\sqrt{\lambda_M})|$, we have

$$\tau^{(1)}(C) = |\mathcal{C}(\sqrt{\lambda_M})|^2 = \mathcal{C}(\sqrt[4]{\kappa})^2.$$

Expanding $\tau^{(1)}$ as $\kappa \rightarrow \infty$, we get

$$(21) \quad \tau^{(1)}(C) = \left(\frac{\sqrt[4]{\kappa} - 1}{\sqrt[4]{\kappa} + 1} \right)^2 = \left(1 - \frac{2}{\sqrt[4]{\kappa} + 1} \right)^2 \approx 1 - \frac{4}{\sqrt[4]{\kappa}} \approx \exp(-4/\sqrt[4]{\kappa}).$$

Note that the condition $\lambda_M \lambda_m = 1$ is not restrictive, since any positive definite matrix verifies it up to scaling, but can significantly accelerate the convergence of these quadrature algorithms for matrices such that $\lambda_M \lambda_m$ is far from 1.

Convergence for the integrand $f_{2,C}(s)$ and comparison. As before, for a positive definite matrix C , a bound for the rate of convergence of the matrix quadrature formula is given by the largest bound on the rate of convergence of the scalar formula applied to the eigenvalues of C .

Since the scalar convergence is complicated by the branch points at 1 and -1 and by the presence of a possibly large number of poles in certain cases, also the matrix convergence is hardly predictable.

Nevertheless, if $\lambda_M \lambda_m = 1$, then for $t = 1/2$ we can get an asymptotic estimate as $\kappa \rightarrow \infty$, which is

$$(22) \quad \tau^{(2)}(C) = \max_{\lambda \in \sigma(C)} \frac{1}{(\rho^{(2)}(\lambda))^2} = \left(\frac{\sqrt{\sqrt{\kappa} + 1}}{\sqrt[4]{\kappa} + 1 + \sqrt{2} \sqrt[8]{\kappa}} \right)^2 \approx 1 - \frac{2\sqrt{2}}{\sqrt[8]{\kappa}}.$$

For $t = 1/2$, it can be shown, moreover, that the Gauss–Jacobi quadrature of (8) is better than that of (7) for

$$|z| \in \mathbb{R} \setminus \left[\frac{1}{\xi}, \xi \right], \quad \xi = 2 + \sqrt{5} + 2\sqrt{2 + \sqrt{5}} \approx 8.35,$$

and this is confirmed by the results of Test 1 in Section 6. Thus, for a positive definite matrix and for $t = 1/2$, unless the matrix is very well conditioned/preconditioned ($\kappa_2(C) \lesssim 70$), the method based on (19) is preferable.

Application to the weighted geometric mean. In the case of the weighted geometric mean, in view of equations (17) and (19), the functions to be integrated are $f_{1,C}(s)$ and $f_{2,C}(s)$, with $C = A^{1/2} B^{-1} A^{1/2}$, so that the previous analysis for a positive definite matrix C can be applied.

Let λ_M and λ_m be the largest and smallest eigenvalues of $A^{1/2} B^{-1} A^{1/2}$ (or of the pencil $A - \lambda B$), respectively. A *scaling* of A and/or B would change the weighted geometric mean in a simple, predictable way, since [41]

$$(\alpha A) \#_t (\beta B) = \alpha^{1-t} \beta^t (A \#_t B).$$

Thus, we may assume that $\lambda_M \lambda_m = 1$ and replace the pair (A, B) with $(\widehat{A}, \widehat{B})$, where $\widehat{A} = A/\sqrt{\lambda_M \lambda_m}$ and $\widehat{B} = B$.

The quadrature formulae $S_N^{(1)}$ of (18) converges at least linearly to $\widehat{A} \#_t \widehat{B}$, and we get the following estimate

$$(23) \quad \|\widehat{A} \#_t \widehat{B} - S_N^{(1)}\| = O(e^{-4N/\sqrt[4]{\kappa}});$$

while we have that $S_N^{(2)}$ of (20), for $t = 1/2$, converges at least linearly to $\widehat{A} \#_{1/2} \widehat{B}$, and we get the estimate

$$(24) \quad \|\widehat{A} \#_{1/2} \widehat{B} - S_N^{(2)}\| = O(e^{-2\sqrt{2}N/\sqrt[8]{\kappa}}).$$

3.4. An alternative quadrature formula. Another powerful quadrature formula for real matrix powers has been obtained in [30] by applying a few variable substitutions on the Cauchy formula for z^{-t} .

Without giving any further details, we report the results of interest from the original paper [30], referring the reader to it for a complete explanation. Let the function $f : \mathbb{C} \setminus (-\infty, 0] \rightarrow \mathbb{C}$ be analytic and let us assume that $(-\infty, 0)$ is a branch cut for f and that 0 is the only singularity, if any. Under these assumptions, the approximation of $f(Z)$, where Z is a real square matrix with positive eigenvalues, using a quadrature formula with N nodes is given by

$$(25) \quad \frac{-8K(k)Z\sqrt[4]{\lambda_m\lambda_M}}{\pi Nk} \operatorname{Im} \left(\sum_{j=1}^N \frac{f(w(t_j)^2) \operatorname{cn}(t_j) \operatorname{dn}(t_j)}{w(t_j)(k^{-1} - \operatorname{sn}(t_j))^2} (w(t_j)^2 I - Z)^{-1} \right),$$

where λ_m and λ_M are the minimum and maximum of the spectrum, respectively, $k = -\mathcal{C}(\sqrt[4]{\lambda_M/\lambda_m})$, $K(\ell)$ is the complete elliptic integral associated with ℓ [30],

$$w(t) = \sqrt[4]{\lambda_m\lambda_M} \frac{k^{-1} + \operatorname{sn}(t)}{k^{-1} - \operatorname{sn}(t)}, \quad t_j = -K(k^2) + \frac{i}{2}K(1-k^2) + \frac{2j-1}{N}K(k^2),$$

for $1 \leq j \leq N$ and $\operatorname{cn}(\cdot)$, $\operatorname{dn}(\cdot)$ and $\operatorname{sn}(\cdot)$ are Jacobi elliptic functions in standard notation (see [1]). The theoretical aspects of these functions can be found in the book by Driscoll and Trefethen [20].

This method can be easily adapted for computing $Af(A^{-1}B)v$, when $A^{-1}B$ is real with positive eigenvalues, without forming explicitly A^{-1} , providing

$$(26) \quad \frac{-8K(k^2)\sqrt[4]{\lambda_m\lambda_M}}{\pi Nk} B \operatorname{Im} \left(\sum_{j=1}^N \frac{f(w(t_j)^2) \operatorname{cn}(t_j) \operatorname{dn}(t_j)}{w(t_j)(k^{-1} - \operatorname{sn}(t_j))^2} (w(t_j)^2 A - B)^{-1} A \right) v,$$

which does not require any matrix product or inversion if evaluated from right to left.

Using the identity $A\#_t B = A(A^{-1}B)^t$, for the matrix geometric mean of real positive definite matrices, one gets the approximation $A\#_t B \approx S_N^{(3)}$ with

$$(27) \quad S_N^{(3)} := \frac{-8K(k^2)\sqrt[4]{\lambda_m\lambda_M}}{\pi Nk} A \operatorname{Im} \left(\sum_{j=1}^N \frac{w(t_j)^{2t-1} \operatorname{cn}(t_j) \operatorname{dn}(t_j)}{(k^{-1} - \operatorname{sn}(t_j))^2} (w(t_j)^2 A - B)^{-1} \right) B.$$

which is of the form (2) with $r_i = w(t_i)^2$ and $s_i = -1$. Unfortunately, for $t \neq 1/2$, the matrices $r_i A + s_i B$ can be complex and not positive definite, for some values of i .

The quadrature formula $S_N^{(3)}$ of (27) converges linearly to $A\#_t B$, in particular the following estimate can be deduced from [30, Thm. 3.1]

$$\|A\#_t B - S_N^{(3)}\| = O(e^{-2\pi^2 N/(\log(\kappa)+6)}),$$

where $\kappa = \lambda_M/\lambda_m$, with λ_m and λ_M the smallest and largest eigenvalues of $A^{-1}B$, respectively. A comparison with the analogous results for the two quadrature formulae of Section 3.2, namely (21) and (22), suggests that this formula can converge much faster when λ_M/λ_m becomes very large and this is confirmed by the experiments in Section 6.

4. Krylov subspace methods. In this section we address the problem of approximating $(A\#_t B)v = A(A^{-1}B)^t v$, using methods based on Krylov subspaces. The approach is similar to the one used in the well-developed problem of approximating $f(C)v$, where C is a large and sparse matrix (see, for instance, [24, Sec. 3] or [32, Ch. 13]). However, the fact that $C = A^{-1}B$, with A and B positive definite, requires certain additional subtleties, such as the convenience of orthogonalizing with respect to a non-Euclidean scalar product. We will refer to the resulting methods as generalized Krylov methods.

We will describe first the generalized Lanczos method in Section 4.1, then the generalized Extended Krylov method in Section 4.2 and finally the generalized rational Krylov methods in Section 4.3. Some convergence issues are addressed in Section 4.4.

The algorithms are presented for the more general problem $Af(A^{-1}B)v$, where $f : \mathcal{U} \rightarrow \mathbb{R}^+$, with \mathcal{U} an open subset of \mathbb{R} , the matrix A is positive definite and B is Hermitian, with $\sigma(A^{-1}B) \subset \mathcal{U}$.

4.1. Generalized Arnoldi and Lanczos methods. Let $A, M \in \mathbb{C}^{n \times n}$ be positive definite and let $B \in \mathbb{C}^{n \times n}$ be Hermitian. The generalized Arnoldi method generates a sequence of M -orthonormal vectors $\{v_k\}_{k=1}^n$ and a sequence of upper Hessenberg matrices $\{H_k\}_{k=1}^n$ with $H_k \in \mathbb{C}^{k \times k}$, such that the columns of $V_k := [v_1 | \dots | v_k] \in \mathbb{C}^{n \times k}$ span an M -orthonormal basis of the Krylov subspace

$$(28) \quad \mathcal{K}_k(A^{-1}B, v) = \text{span}\{v, (A^{-1}B)v, \dots, (A^{-1}B)^{k-1}v\},$$

where $v_1 = v/\|v\|_M$ and the elements of H_k , defined by $h_{ij} = v_i^* M A^{-1} B v_j$, turn out to be the coefficients of the Gram–Schmidt orthogonalization process [27, Sect. 9.4.1], with respect to the scalar product defined by M . The algorithm has a breakdown when, for some $j \leq n$, we have $v_j \in \text{span}\{v_1, \dots, v_{j-1}\}$.

If no breakdown occurs, the matrices produced by the algorithm satisfy $V_n^* M V_n = I_n$, $BV_n = AV_n H_n$ and, for $k < n$,

$$(29) \quad BV_k = AV_k H_k + h_{k+1,k} A v_{k+1} e_k^*,$$

where e_k is the last column of $I_k \in \mathbb{C}^{k \times k}$.

It is well known [33, Chap. 13] that equation (29) can be readily exploited to compute an approximation of $f(A^{-1}B)v$. If $QV_k = V_k U$, where $Q, U \in \mathbb{C}^{n \times n}$ and $V \in \mathbb{C}^{n \times k}$, then, it can be proved that $f(Q)V_k = V_k f(U)$. Thus, by imposing $BV_k \approx AV_k H_k$, we can write that

$$f(A^{-1}B)V_k \approx V_k f(H_k),$$

and by observing that $v = v_1 \|v\|_M = V_k e_1 \|v\|_M$, we obtain that

$$(30) \quad Af(A^{-1}B)v = Af(A^{-1}B)V_k e_1 \|v\|_M \approx AV_k f(H_k) e_1 \|v\|_M,$$

a relation that is useful, in practice, only when the approximation is good for k much smaller than n .

We discuss now the options for the matrix defining the inner product used in the Arnoldi process. Following the recommendation of Parlett [45, Ch. 15], Arioli and Loghin [5] develop an algorithm to approximate $(A\#_t B)v$ using $M = A$. It is immediate to see that, in this case, H_k is tridiagonal, in being both upper Hessenberg and Hermitian, since $H_k = V_k^* B V_k$. Thus, the generalized Arnoldi process becomes

a generalized Lanczos algorithm, which is superior for two main reasons. On the one hand, the computation of each v_k requires a fixed number of arithmetic operations, which considerably decreases the overall execution time of the algorithm, on the other hand, the evaluation of $f(H_k)$ becomes easier and can be accurately performed by diagonalization, since H_k is normal.

If B is positive definite, then the generalized method for (A, B) admits a minor variation: we can use the Arnoldi process to construct a basis of $\mathcal{K}_k(A^{-1}B, v)$ of (28) which is B -orthonormal. In this case, we get $BV_n = AV_nH$ with $V_n^*BV_n = I_n$ and the matrices $H_k = V_k^*BA^{-1}BV_k$ turn out to be tridiagonal.

In principle, any scalar product associated to a positive definite matrix M could be used in the Arnoldi process to construct a basis of $\mathcal{K}_k(A^{-1}B, v)$, and the sequence of upper Hessenberg matrices H_k . However, if we want H_k to be tridiagonal, we must restrict the choice for M as in the following.

PROPOSITION 4.1. *Let $A, M \in \mathbb{C}^{n \times n}$ be positive definite and $B \in \mathbb{C}^{n \times n}$ be Hermitian, and assume that the Arnoldi process applied to $A^{-1}B$ with starting vector v and orthogonalization with respect to the scalar product induced by M can be applied with no breakdown. Then for $k = 1, \dots, n$, the Hessenberg matrix H_k is Hermitian (and thus tridiagonal) if and only if $MA^{-1}B = BA^{-1}M$.*

Proof. From $H_k = V_k^*MA^{-1}BV_k$, we get that $H_k = H_k^*$ for each k , if and only if $MA^{-1}B = BA^{-1}M$. \square

The previous result shows that, for the problem $Af(A^{-1}B)v$, the customary orthogonalization procedure, that corresponds to the choice $M = I$, can cause loss of structure since H_k is nonsymmetric if A and B do not commute.

4.2. Generalized Extended Krylov subspace method. The standard extended Krylov methods [21, 51] can be easily generalized to build an M -orthonormal basis of the extended Krylov subspace

$$\mathcal{E}_k(A^{-1}B, v) = \text{span}\{v, A^{-1}Bv, B^{-1}Av, (A^{-1}B)^2v, \dots, (B^{-1}A)^{\frac{k}{2}-1}v, (A^{-1}B)^{\frac{k}{2}}v\},$$

if k is even and

$$\mathcal{E}_k(A^{-1}B, v) = \text{span}\{v, A^{-1}Bv, B^{-1}Av, (A^{-1}B)^2v, \dots, (A^{-1}B)^{\frac{k-1}{2}}v, (B^{-1}A)^{\frac{k-1}{2}}v\},$$

if k is odd.

As it is the case for the standard Arnoldi algorithm, the extended Krylov algorithm generates a sequence of M -orthonormal vectors $\{v_k\}_{k=1}^n$ and a sequence of Hessenberg matrices with an additional subdiagonal $\{H_k\}_{k=1}^n$ with $H_k \in \mathbb{C}^{k \times k}$. We stress that, in this case, H_k does not contain the orthogonalization coefficients of the Gram–Schmidt process applied to the set $\{v_1, \dots, v_k\}$. The interplay between orthogonalization coefficients and H_k , for the extended Krylov subspace methods, are discussed by Simoncini [51] and Jagels and Reichel [39, 38].

If we define $V_k = [v_1 | \dots | v_k]$ as the M -orthonormal basis of $\mathcal{E}_k(A^{-1}B, v)$, then the matrices produced by the algorithm, if no breakdown occurs, verify $BV_n = AV_nH_n$ and $V_n^*MV_n = I_n$, while for k even and $k < n$

$$(31) \quad BV_k = AV_kH_k + A[v_{k+1}|v_{k+2}]\tilde{H}E_k,$$

where $H_k \in \mathbb{C}^{k \times k}$, $\tilde{H} = [v_{k+1}|v_{k+2}]^*MA^{-1}B[v_{k-1}|v_k] \in \mathbb{C}^{2 \times 2}$, $E_k \in \mathbb{C}^{2 \times k}$ contains the last two rows of the identity matrix I_k and $V_k \in \mathbb{C}^{n \times k}$ is the M -orthonormal basis of the extended Krylov subspace at step k .

As in the previous section, we can conclude that $H_k = V_k^* M A^{-1} B V_k$ and thus that Proposition 4.1 remains valid for the extended method. The choice $M = A$, is again the most natural. Moreover, for any $k \leq n$ the function $Af(A^{-1}B)v$ can be approximated by means of

$$(32) \quad Af(A^{-1}B)v \approx AV_k f(H_k) e_1 \|v\|_M,$$

where V_k and H_k are the matrices produced by the extended algorithm.

We wish to point out that the Arnoldi decomposition (31) is specific to the basis computation approach that adds two vectors at each step [51]. Using the approach of [39, 38], one would have to add to $AV_k H_k$ only one non-zero column rather than two.

4.3. Generalized rational Krylov subspace methods. The rational Arnoldi algorithm [48, 50] can be adapted to our problem. Starting with a vector v , a positive definite matrix M , and poles $\xi_1, \dots, \xi_k \in \mathbb{C} \cup \{\infty\}$ such that $\xi_i \notin \sigma(A^{-1}B) \cup \{0\}$, we can construct a basis of the rational Krylov subspaces (we set $1/\infty = 0$)

$$\mathcal{Q}_k(A^{-1}B, v) := \prod_{j=1}^{k-1} \left(I_n - \frac{1}{\xi_j} A^{-1} B \right)^{-1} \text{span}\{v, A^{-1}Bv, \dots, (A^{-1}B)^{k-1}v\},$$

by considering $v_1 = v/\|v\|_M$ and then M -orthogonalizing the vector

$$w_j = (A - B/\xi_j)^{-1} B v_j,$$

with respect to v_1, \dots, v_j , obtaining

$$h_{ij} = w_j^* M v_i, \quad \widetilde{w}_j = w_j - \sum_{i=1}^j h_{ij} v_i, \quad h_{j+1,j} = \|\widetilde{w}_j\|_M, \quad v_{j+1} = \widetilde{w}_j / h_{j+1,j}.$$

Notice that a breakdown can occur if $\widetilde{w}_j = 0$, that is, $w_j \in \text{span}\{v_1, \dots, v_j\}$.

In this way, if no breakdown occurs, we get the rational Arnoldi decomposition

$$(33) \quad BV_k(I_k + H_k D_k) + \frac{h_{k+1,k}}{\xi_k} B v_{k+1} e_k^* = AV_k H_k + h_{k+1,k} A v_{k+1} e_k^*,$$

where $D_k = \text{diag}(1/\xi_1, \dots, 1/\xi_k)$, H_k is the matrix containing the entries h_{ij} and $V_k = [v_1 | \dots | v_k]$ is an M -orthogonal basis of $\mathcal{Q}_k(A^{-1}B, v)$. Note that we do not allow 0 to be a pole just for ease of exposition; it is possible to build a rational Arnoldi decomposition with a pole at 0, by using a slightly different definition [8, Sect. 3].

If the last pole is at infinity, then (33) simplifies to

$$BV_k(I_k + H_k D_k) \approx AV_k H_k$$

and we get the approximation

$$(34) \quad Af(A^{-1}B)v \approx AV_k f(H_k(I_k + H_k D_k)^{-1}) e_1 \|v\|_M.$$

Notice that in this case $H_k(I_k + H_k D_k)^{-1} = V_k^* M A^{-1} B V_k$, which is Hermitian if M commutes with $A^{-1}B$. Thus, the argument of f is normal and the evaluation can be done by diagonalization.

The Krylov subspaces described in Section 4.1 and Section 4.2 are in fact rational Krylov subspaces where the poles are chosen to be ∞ or 0 and ∞ , respectively. In order to achieve a convergence rate faster than that of the previous two algorithms, the choice of poles is crucial, but there is no general recipe. In Section 6 we use two black-box heuristics which are well-suited to the problem $f(A)b$.

4.4. Convergence of Krylov methods. Despite not being very practical from a computational perspective, the identity $Af(A^{-1}B)v = A^{1/2}f(A^{-1/2}BA^{-1/2})A^{1/2}v$ turns out to be very useful in the analysis of the convergence of the Krylov methods, as we will see.

By exploiting the generalized Arnoldi method, we get an approximation of the form (compare (30))

$$(35) \quad Af(A^{-1}B)v \approx AV_k f(H_k) e_1 \|v\|_M =: f_k,$$

where V_k is an M -orthogonal basis of the Krylov subspace $\mathcal{K}_k(A^{-1}B, v)$ defined in (28) and $H_k = V_k^* M A^{-1} B V_k$.

On the other hand, we can pick a positive definite matrix \widetilde{M} and apply the generalized Lanczos method to compute a matrix $W_k \in \mathbb{C}^{n \times k}$, with \widetilde{M} -orthogonal columns and span the Krylov subspace $\mathcal{K}_k(A^{-1/2}BA^{-1/2}, A^{1/2}v)$, obtaining the approximation

$$(36) \quad Af(A^{-1}B)v = A^{1/2}f(A^{-1/2}BA^{-1/2})A^{1/2}v \approx A^{1/2}W_k f(\widetilde{H}_k) e_1 \|A^{1/2}v\|_{\widetilde{M}} =: g_k,$$

with $\widetilde{H}_k = W_k^* \widetilde{M} A^{-1/2} B A^{-1/2} W_k$.

We will prove that these two approximations are equal for a suitable choice of W_k and \widetilde{M} .

PROPOSITION 4.2. *Let $A, B, M \in \mathbb{C}^{n \times n}$ be positive definite and let $v \in \mathbb{C}^n$. If the columns of $V_k \in \mathbb{C}^{n \times k}$ span an M -orthogonal basis of $\mathcal{K}_k(A^{-1}B, v)$, then the columns of $W_k := A^{1/2}V_k$ span an \widetilde{M} -orthogonal basis of $\mathcal{K}_k(A^{-1/2}BA^{-1/2}, A^{1/2}v)$, with $\widetilde{M} = A^{-1/2}MA^{-1/2}$ and f_k and g_k , defined in (35) and (36), respectively, are such that $f_k = g_k$.*

Proof. First, we observe that the columns of W_k are \widetilde{M} -orthogonal, since

$$W_k^* \widetilde{M} W_k = V_k^* A^{1/2} \widetilde{M} A^{1/2} V_k = V_k^* M V_k = I.$$

and that it is a basis of $\mathcal{K}_k(A^{-1/2}BA^{-1/2}, A^{1/2}v)$, since for $\ell = 0, \dots, k-1$ we have that $A^{1/2}(A^{-1}B)^\ell v = (A^{-1/2}BA^{-1/2})^\ell (A^{1/2}v)$.

By direct inspection, we note that

$$\widetilde{H}_k = W_k^* \widetilde{M} A^{-1/2} B A^{-1/2} W_k = V_k^* A^{1/2} \widetilde{M} A^{1/2} A^{-1} B V_k = V_k^* M A^{-1} B V_k = H_k,$$

and that

$$\|A^{1/2}v\|_{\widetilde{M}}^2 = v^* A^{1/2} \widetilde{M} A^{1/2} v = v^* M v = \|v\|_M^2,$$

from which we obtain

$$g_k = A^{1/2}W_k f(\widetilde{H}_k) e_1 \|A^{1/2}v\|_{\widetilde{M}} = AV_k f(H_k) e_1 \|v\|_M = f_k.$$

□

Observe that for $M = A$, we have $\widetilde{M} = I$, which gives yet another reason for making this choice.

The previous equivalence is true also for rational Krylov subspaces (and in particular, for extended Krylov subspaces), because approximating $(A\#_t B)v$ in the space $\mathcal{Q}_k(A^{-1}B, v)$ is equivalent to constructing a sequence of approximations to the same quantity in $\mathcal{Q}_k(A^{-1/2}BA^{-1/2}, A^{1/2}v)$.

The equivalence of approximations allows one to estimate the convergence of Krylov methods using the convergence results for functions of positive definite matrices, which are simpler than those for general matrices.

For instance, if \tilde{f}_k is the approximation of $(A\#_t B)v$ in the extended Krylov subspace $\mathcal{E}_k(A^{-1}B, v)$, using the error bound from [40], we obtain

$$\|(A\#_t B)v - \tilde{f}_k\| = O(e^{-2k/\sqrt[4]{\kappa}}),$$

where κ is the condition number of $A^{-1/2}BA^{-1/2}$.

5. Computing $(A\#_t B)^{-1}v$. The methods for computing the product of the weighted geometric mean times a vector, described in the previous sections, can be easily adapted for reducing the linear system

$$(A\#_t B)^{-1}v,$$

to the solution of a certain number of simpler linear systems.

Since $(A\#_t B)^{-1} = B^{-1}(B\#_t A)A^{-1}$, the quadrature formulae of Section 3 can still be applied. From (18) we get the approximation

$$(A\#_t B)^{-1} \approx \frac{2 \sin(\pi t)}{\pi} \sum_{i=0}^N w_i ((1 - s_i)B + (1 + s_i)A)^{-1},$$

from (20) the approximation

$$(A\#_t B)^{-1} \approx \frac{2 \sin(\pi t)}{\pi} \sum_{i=0}^N w_i ((1 - s_i)^{\frac{1}{1-t}} A + (1 + s_i)^{\frac{1}{1-t}} B)^{-1},$$

and from (27) the approximation

$$(A\#_t B)^{-1} \approx \frac{-8K(k^2)\sqrt[4]{\lambda_m \lambda_M}}{\pi N k} \operatorname{Im} \left(\sum_{j=1}^N \frac{w(t_j)^{2t-1} \operatorname{cn}(t_j) \operatorname{dn}(t_j)}{(k^{-1} - \operatorname{sn}(t_j))^2} (w(t_j)^2 B - A)^{-1} \right),$$

when both A and B are real. The three quadrature formulae have exactly the same convergence properties as the respective formulae for $A\#_t B$.

Regarding the Krylov methods of Section 4, we can exploit the identity

$$(A\#_t B)^{-1} = (A(A^{-1}B)^t)^{-1} = (A^{-1}B)^{-t} A^{-1},$$

reducing the computation of $(A\#_t B)^{-1}v$ to that of $(A^{-1}B)^{-t}A^{-1}v$, which can be performed by first computing $w = A^{-1}v$ and then approximating $(A^{-1}B)^{-t}w$ with any of the Krylov subspace methods described in Section 4.

6. Numerical tests. By means of numerical experiments, we illustrate the behavior of the methods presented in the paper for the computation of $(A\#_t B)v$ and $(A\#_t B)^{-1}v$, where A and B are medium- to large-scale matrices.

The tests were performed using MATLAB R2017a (9.2) on a machine equipped with an Intel i5-3570 Processor running at 3.40GHz and 8GB of dedicated RAM.

We compare the following methods:

1. The generalized Arnoldi algorithm [45, Sect. 15.11] (**PolY**);
2. The extended Krylov subspace method [21] (**Extended**);

3. A rational Krylov subspace method, with poles chosen according to the adaptive strategy of Güttel and Knizhnermann [28] (`RatAdapt`);
4. A rational Krylov subspace method, where the choice of the poles is based on the solution of the best rational approximation of an auxiliary problem [8] (`RatFit`);
5. The quadrature formula (18) (`Quad1`);
6. The quadrature formula (20) (`Quad2`);
7. The quadrature formula (27) (`Elliptic`).

Krylov subspace methods. Our implementations of the Krylov subspace methods are based on the modified Gram–Schmidt procedure with reorthogonalization [26]. When approximating $Af(A^{-1}B)v$, we can decide to use either the projection of $A^{-1}B$ onto the Krylov subspace or the matrix containing the orthonormalization coefficients used in the Gram–Schmidt process. When the Krylov subspace is enlarged, the projection does not have to be computed from scratch, but can be updated cheaply by exploiting an opportune recurrence. This choice still leads to a larger computational cost, due to one or more additional matrix-vector products and/or linear system solves per step, but guarantees that the projected matrix is symmetric positive definite. The matrix obtained by the orthogonalization procedure, on the other hand, is numerically not Hermitian, and it is not Hermitian when rational Arnoldi is used as described in Section 4.3.

In our implementations of `Poly` and `Extended`, we trade off maintaining the structure of the problem for efficiency, and use the orthonormalization coefficients to build the reduced matrix. In this case the fractional power of a nonnormal matrix can be computed by spectral decomposition or by using algorithms for the real power of dense matrices [36, 34] (all these algorithms require $O(\ell^3)$ ops for a matrix of size ℓ). We stress that, in our tests, this choice did not reduce the accuracy of the final result, and only marginally affected the computational cost.

Rational Krylov methods, however, produce a pair of matrices from the orthonormalization coefficients, and it is not obvious how to combine them in order to obtain an approximation of $Af(A^{-1}B)v$. For that reason we resort to the slightly more expensive projections in `RatAdapt` and `RatFit`.

For the rational Krylov methods, the poles are chosen according to either the adaptive strategy by Güttel and Knizhnerman [28] or the function `rkfit` from the `rktoolbox` [7], based on an algorithm by Berljafa and Güttel [8, 9]. In our implementation, we get the poles by running `rkfit` on a surrogate problem of size 800 whose setup requires a rough estimate of the extrema of the spectrum of $A^{-1}B$.

As a stopping criterion for the Krylov subspace methods, we use the estimate [40]

$$\frac{\|u - u_m\|}{\|u_m\|} \approx \frac{\delta_{m+j}}{1 - \delta_{m+j}},$$

where $\|\cdot\|$ is the spectral norm, $u = (A^{-1}B)^{-t}v$, u_m is the approximation at step m and δ_{m+j} is the norm of the relative difference between the approximation at the step m and $m+j$, i.e. $\|u_m - u_{m+j}\|/\|u_m\|$ where j is usually small and is set to 4 in our experiments.

Quadrature methods. For quadrature methods related to the Gauss–Jacobi quadrature, namely (18) and (20), the nodes and the weights are generated using the function `jacpts` of `Chebfun` [19], based on an algorithm by Hale and Townsend [31], which requires $O(N)$ operations to compute N nodes and weights of the quadrature.

Table 1: Comparison of the methods used in the numerical experiments in terms of knowledge of the spectrum of $A^{-1}B$ or $B^{-1}A$ (spectrum), type of linear systems to be solved (shifted systems, positive definite or not, or systems with the same left hand side), and possibility to increase the number of nodes/enlarge the Krylov subspace (update) exploiting the previous computation without starting from scratch.

Method	Spectrum	Systems	Update
Poly	no	same lhs	yes
Extended	no	same lhs	yes
RatAdapt	no	shifted pd	yes
RatFit	yes	shifted pd	yes
Quad1	yes	shifted pd	no
Quad2	yes	shifted pd	no
Elliptic ($t = 1/2$)	yes	shifted pd	no
Elliptic ($t \neq 1/2$)	yes	shifted	no

The scaling technique described at the end of Section 3.3 is used to accelerate the convergence.

For **Quad2** we use the quadrature formula (20) when $t > 1/2$, and if $t \leq 1/2$ we exploit the identity $A\#_t B = B\#_{1-t} A$ to reduce to the former case.

In view of the remark at the end of Section 3.3, the convergence in the matrix case is exactly predicted by the scalar convergence on the extreme eigenvalues. Thus, the number of nodes needed by **Quad1** and **Quad2** to get the required approximation is estimated by applying its scalar counterpart, with a variable number of nodes and weights, to the extreme eigenvalues of the matrix $B^{-1}A$. These scalar problems are much easier and marginally affect the total computational cost of the algorithms, when dealing with large matrices.

Regarding the method described in Section 3.4, we adapt the implementation given by Hale, Higham and Trefethen [30], which uses the routines `ellipkjc` and `ellipkcp` from Driscoll’s *Schwarz–Christoffel Toolbox* [17, 18]. In this case, the number of nodes, is estimated by applying the same method to a 2×2 matrix whose eigenvalues are the extreme eigenvalues of $A^{-1}B$. Since in all our tests we consider only real matrices, the method of Section 3.4, which is designed for real problems only, can always be applied.

Linear systems and extreme eigenvalues. In both Krylov subspace methods and quadrature methods, the problem is reduced to the solution of linear systems which are solved by the MATLAB sparse linear solver, exploiting the band and the positive definite structure. The linear systems to be solved by the method **Elliptic** are not guaranteed to be positive definite for $t \neq 1/2$ and this may considerably increase the overall time required by the algorithm.

Finally, the extreme eigenvalues of $A^{-1}B$ (or $B^{-1}A$), when needed, are approximated with two significant digits by calling the function `eigs` of MATLAB, with the pair (B, A) (or (A, B)) as argument. In Table 1 we give a synoptic comparison of the key features of the methods.

TEST 1. In Section 3, we considered two Gauss–Jacobi quadrature formulae for z^{-t} , one based on (7), implemented by **Quad1** and one based on (8), implemented by **Quad2**. We derived a bound on the rate of convergence of both formulae: $|\mathcal{C}(\sqrt{z})|^2$

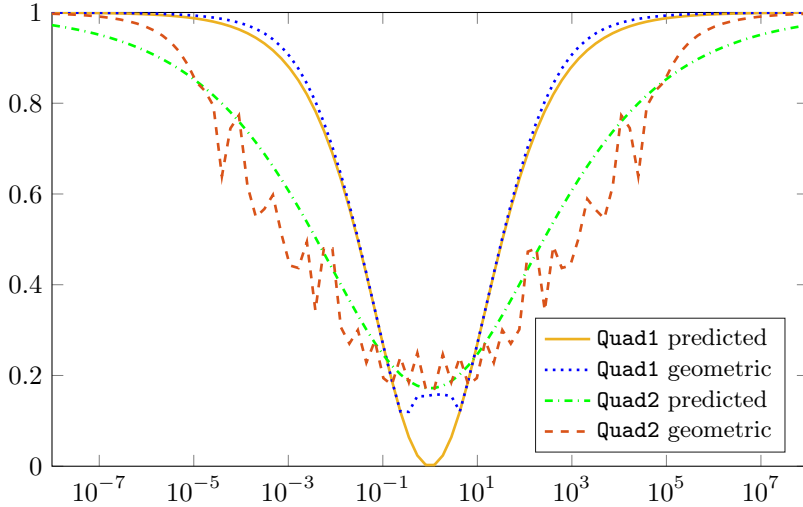


Fig. 1: Comparison of the parameters of convergence (on the y -axis) of the two Gaussian quadrature formulae for $z^{-1/2}$ (on the semilogarithmic x -axis).

with $\mathcal{C}(x) = \frac{1-x}{1+x}$ for Quad1, and $(1/\rho^{(2)}(z))^2$ with $\rho^{(2)}$ as in (15) for Quad2. The latter is valid just for $t = 1/2$.

We compare the experimental rate of convergence, which is the median of the error reduction over a certain number of steps, with the predicted rate of convergence. The results, for $t = 1/2$, are drawn in Figure 1. As one can see, the first quadrature formula is more accurate for values of $|z|$ close, in magnitude, to 1, while the second gives better results for values of $|z|$ far from 1.

If we consider a positive definite matrix A scaled so that $\lambda_M \lambda_m = 1$ (where λ_M and λ_m are the extreme eigenvalues of A), then the first formula seems to be more convenient for well conditioned matrices, say with $\lambda_M/\lambda_m \lesssim 70$.

For $t \neq 1/2$ the bound for Quad1 is still valid, as confirmed by numerical experiments not reported here, while the bound for Quad2 is less predictive, and does not give any information for $t \neq 1/2$. Nevertheless, the asymptotic expansion (16) suggests a better convergence for Quad2 for $t = (p-1)/p$ and the quadrature formula shows an acceptable convergence rate even for values of t such that the integrand is not analytic, provided that $t \geq 1/2$. By using the formula $A\#_t B = B\#_{1-t} A$ we can achieve similar convergence properties also for $t < 1/2$.

TEST 2. Since the convergence of most of the methods depends on the conditioning of the matrix $A^{1/2} B^{-1} A^{1/2}$ (that is λ_M/λ_m , where λ_M and λ_m are the largest and the smallest, respectively, eigenvalues of the matrix), we generate two matrices A and B such that $A^{-1} B$ (and thus $A^{1/2} B^{-1} A^{1/2}$) has prescribed eigenvalues. The eigenvalues belong to a fixed interval and are clustered near the boundaries of the spectrum, to get a fair comparison between quadrature and Krylov subspace methods.

We consider matrices of size 1000, so that a reference value for $w = (A\#_t B)v$ can be computed by means of a reliable algorithm for the dense case, namely the Cholesky–Schur algorithm described in [35, Sec. 3], which is implemented by the `sharp` function of the `Matrix Means Toolbox` [13].

For each method, the relative forward error of the computed value \tilde{w} with respect

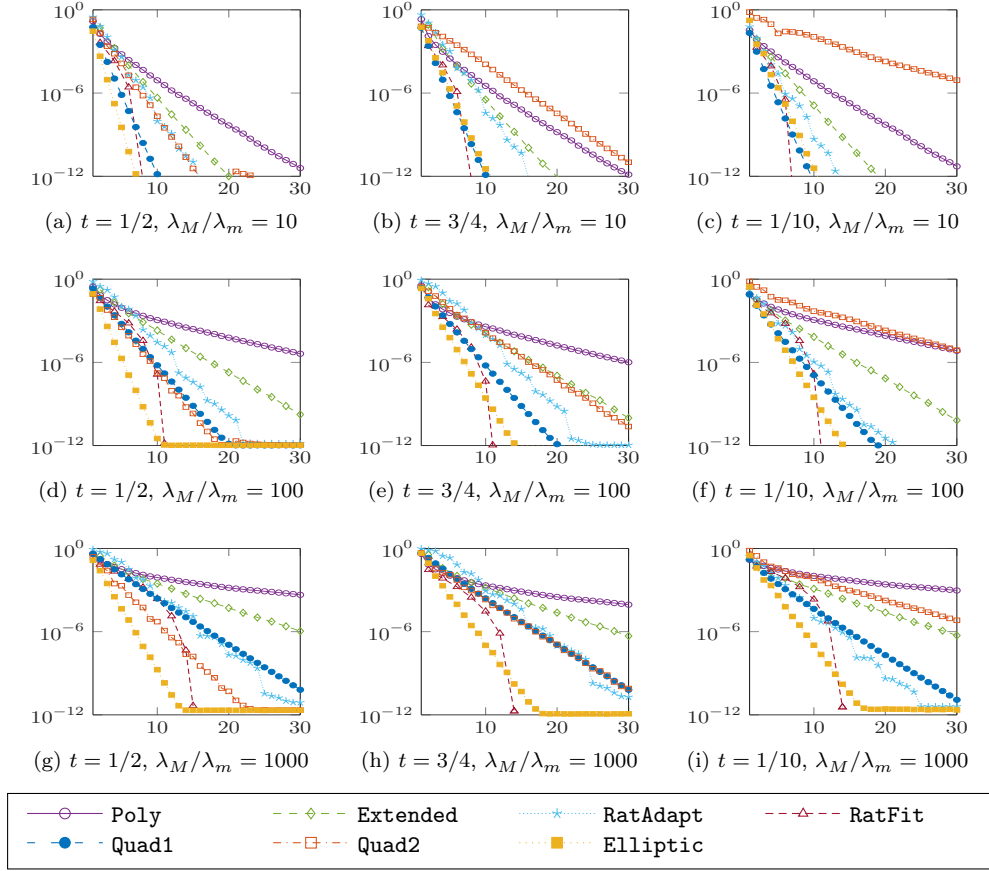


Fig. 2: Convergence of the methods in Table 1 for computing $(A\#_t B)v$ for $t \in \{1/2, 3/4, 1/10\}$ and $\lambda_M/\lambda_m \in \{10, 100, 1000\}$, where λ_M and λ_m are the extreme eigenvalues of $A^{1/2}B^{-1}A^{1/2}$. We consider on the x -axis the number of nodes for quadrature methods and the dimension of the subspace for Krylov methods; and on the y -axis the relative error with respect to a reference solution.

to the reference value, namely





$$\varepsilon = \frac{\|\tilde{w} - w\|}{\|w\|},$$

is measured in the spectral norm for a variable number of nodes of the quadrature methods and for a variable size of the Krylov subspace.

The results are drawn in Figure 2. The tests confirm the predicted dependence of the convergence on the conditioning of $A^{1/2}B^{-1}A^{1/2}$. The final accuracy of all methods is comparable, while we observe a different convergence behavior for $t = 1/2$ and for $t \neq 1/2$, for the methods **Quad2** and **Elliptic**.

For $t = 1/2$, **Elliptic** generates the best rational relative minimax approximation of the function $z^{-1/2}$ on the interval $[\lambda_m, \lambda_M]$ [30]. This is the reason why it converges faster than the other methods, which produce different rational approximations to

Table 2: ID in the University of Florida Sparse Matrix Collection, size and sparsity pattern of the matrices used in the experiments on large-scale matrices. In dataset 3, the asterisk means that a small multiple of the identity has been added to the two matrices.

Dataset	λ_M/λ_m	IDs in UFsmc	Size	Pattern
1	71.1	1312 & 1314	40 000	
2	7.5	1275 & 1276	90 449	
3	299.5	2257* & 2258*	102 158	
4	1.2	942 & 946	504 855	

$z^{-1/2}$. We note that **RatFit** converges in a similar number of steps and that **Quad2** converges much faster than **Quad1** as λ_M/λ_m grows, as predicted in (23). Regarding the Krylov subspace methods, we observe linear convergence which is very slow for the Arnoldi method and it is quite fast when the adaptive strategy is used in the rational Krylov method.

For $t \neq 1/2$, Krylov methods and **Quad1** have the same behavior they have for $t = 1/2$. The **Elliptic** method does not produce the best rational approximation anymore, and although A and B are real, it may require the solution of complex linear systems. However, despite in this case it need not be the fastest method, it still shows a remarkably fast convergence. The behavior of **Quad2** degrades fast as t gets far from $t = 1/2$, an partial explanation for this is given in Section 3. The fastest convergence for $t \neq 1/2$ is usually achieved by **RatFit**.

TEST 3. In order to illustrate the behavior of the methods when dealing with large-scale matrices, we consider four pairs of conformable symmetric positive matrices from the *University of Florida Sparse Matrix Collection* [16].

The four choices considered in our experiments are described in Table 2. In the case of dataset 3, due to the extreme ill-conditioning of one of the two matrices (whose 1-norm condition number is approximately $3 \cdot 10^{19}$) and the large rate $\lambda_M/\lambda_m \approx 10^{18}$ (where λ_M/λ_m is the conditioning of the matrix $A^{1/2}B^{-1}A^{1/2}$), we were not able to get any result. Since this dataset is interesting being the only one with non-banded matrices, we tamed the conditioning of the data, without affecting the nonzero structure, by adding the matrix $10^{-3}I$ to both matrices.

In order to test the methods in Table 1, we compare the CPU time required, for $t = 1/2$, $t = 3/4$ and $t = 1/10$, to fulfill the stopping criterion. We do not report the CPU time if the corresponding algorithm does not achieve the accuracy threshold after building a Krylov space of dimension 1000 or using 1000 quadrature nodes.

The results, given in Table 3, show that the convergence speed is dictated by the ratio λ_M/λ_m , as predicted, while the CPU time is not necessarily related to the number of linear system solves (between parentheses). Indeed, some methods require spectral information (see Table 1) and this task turns out to be costly, when the methods for computing the extreme eigenvalues converge very slowly (datasets 1, 2 and 4), while it does not influence dramatically the computational cost when it converges quickly (dataset 3). In particular, in dataset 3, if we denote by $\lambda_1 \geq \dots \geq \lambda_n$ the eigenvalues of $A^{-1}B$, where n is the size of A , then the parameters that deter-

Table 3: Comparison of the algorithms presented in the paper, when applied to large-scale matrices, in terms of CPU time (in seconds) and number of linear systems to be solved (between parentheses). We do not report any data for methods that require more than 1000 system solves to achieve the required accuracy.

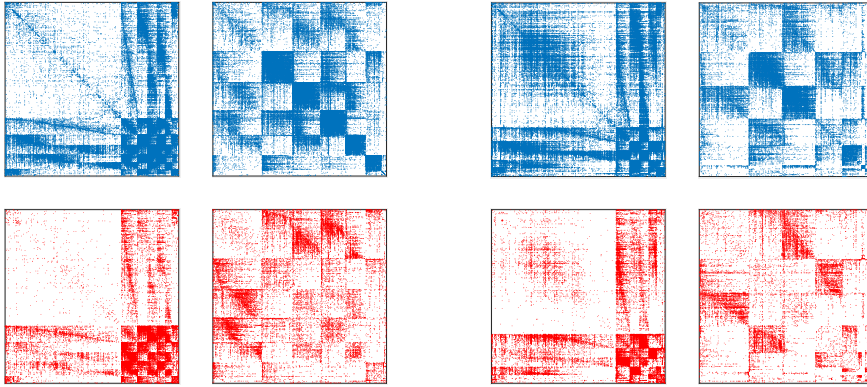
	t	Poly	Extended	RatAdapt	RatFit	Quad1	Quad2	Elliptic
1	0.50	1.6 (50)	1.0 (32)	1.6 (21)	1.7 (11)	2.1 (20)	2.1 (20)	1.6 (11)
	0.75	1.3 (45)	1.0 (32)	1.6 (21)	1.7 (11)	2.1 (20)	3.0 (35)	3.8 (13)
	0.10	1.6 (54)	1.0 (32)	1.4 (18)	1.6 (10)	2.0 (19)	6.1 (82)	3.8 (13)
2	0.50	7.7 (13)	6.8 (16)	10.0 (11)	18.8 (07)	21.1 (11)	26.0 (17)	18.6 (07)
	0.75	7.2 (12)	6.8 (16)	10.0 (11)	18.8 (07)	20.3 (10)	40.6 (35)	70.5 (11)
	0.10	7.2 (12)	6.8 (16)	8.1 (09)	18.8 (07)	20.2 (10)	80.6 (83)	65.2 (10)
3	0.50	–	17.8 (106)	15.0 (40)	10.0 (18)	–	16.5 (44)	8.9 (19)
	0.75	–	21.0 (118)	17.6 (46)	9.5 (17)	–	14.0 (36)	20.3 (19)
	0.10	–	10.2 (74)	8.9 (25)	10.4 (19)	–	22.4 (63)	22.2 (21)
4	0.50	18.9 (07)	25.4 (12)	28.3 (07)	69.6 (03)	72.3 (04)	115.9 (16)	75.1 (04)
	0.75	19.0 (07)	23.1 (12)	28.3 (07)	69.5 (03)	72.2 (04)	185.4 (35)	192.9 (06)
	0.10	17.1 (06)	19.3 (10)	24.1 (06)	69.5 (03)	72.3 (04)	364.8 (83)	192.5 (06)

mine the convergence of the power and inverse power methods, say $\gamma_1 = \lambda_2/\lambda_1$ and $\gamma_2 = \lambda_n/\lambda_{n-1}$, are bounded by 0.981, so that the extreme eigenvalues are computed very efficiently and the methods requiring the spectrum perform relatively well.

The Arnoldi and the extended Krylov subspace methods require no spectral information and the solution of linear systems with the same left hand side. In our code, we exploit this fact and begin by finding the Cholesky factorization of A and B and use it to solve efficiently all subsequent linear systems. To cope with sparse non-banded matrices and avoid excessive fill-in, we reorder the rows and columns of the matrix by applying an approximate symmetric minimum degree permutation, which we compute by means of the MATLAB `symamd` function. Notice that `Poly` gives good results for the dataset 4, where λ_M/λ_m is exceptionally small; when λ_M/λ_m grows, the fastest convergence of other Krylov methods makes them preferable. The `Extended` and `RatAdapt` are good alternative if nothing is known about the problem, but when λ_M/λ_m is very large (and an approximation of the spectrum can be reasonably computed) as in dataset 3, they may be overtaken by `RatFit` or by the quadrature methods.

On the other hand, the methods based on quadrature do not seem to be competitive for $t \neq 1/2$. While `Quad1` converges too slowly, and this results in a large computational cost, the convergence of `Quad2` is fast for $t = 1/2$, but its performance degrades rapidly as t approaches 0 or 1. Finally, the method based on the conformal transformation (`Elliptic`) requires a very small number of linear system to be solved, but these systems, for $t \neq 1/2$, are not positive definite and this results in a generally larger computational cost.

Finally, we wish to point out that in the dataset 4, the reason for the overhead of `RatFit`, among Krylov methods is related to the cost of the approximation of the spectrum of $A^{-1}B$. The big difference between the overall cost of `Quad1` and `Quad2`, with respect to the number of linear system solves, depends on the fact that, in our implementation, `Quad1` spends most of the time trying to compute the extreme eigenvalues of $A^{-1}B$. On the contrary, in the dataset 3, the conditioning is high, so that the convergence of the methods is slow, but the convergence of the power



(a) Clustering for $t = 0.35$.

(b) Clustering for $t = 1/2$.

Fig. 3: The two figures report positive (blue, top left) and negative (red, bottom left) adjacency matrices of the Wikipedia RfA signed network. The rows and columns are reordered according to a clustering of the eigenvectors corresponding to the smallest 30 eigenvalues of $W^+ \#_{0.35} W^-$ (Figure 3a) and $W^+ \#_{1/2} W^-$ (Figure 3b). The right columns shows a detail of the last rows and columns of the corresponding matrix on the left.

and inverse power methods is fast, so that the extreme eigenvalues are computed very efficiently and the fastest methods are among those requiring the spectrum (i.e., RatFit and Elliptic).

It is worth stressing that our results are just indicative and do not represent exactly what would happen if high performance implementations were used.

TEST 4. The weighted geometric mean (with $t = 1/2$) is considered by Mercado, Tudisco and Hein [44] as a tool for clustering *signed networks*, that is, networks that model both attractive and repulsive relationships by means of positive and negative (weighted) edges, respectively. It is customary to assign to these networks two distinct adjacency matrices, A^+ , for positive edges, and A^- , for negative ones.

The clustering process consists of several steps. After preprocessing the data by discarding all the rows and columns that do not belong to the largest connected component of the undirected graph of the network, the algorithm constructs W^+ , the normalized signed Laplacian of A^+ , and W^- , the normalized signless Laplacian [43] of A^- . The rows (and columns) of the matrix are divided into k communities by performing a k-means clustering of the eigenvectors of $W^+ \# W^-$ corresponding to the k smallest (in magnitude) eigenvalues. In [44], the eigenpairs of $W^+ \# W^-$ are computed by means of the inverse power method [52, Lect. 27], where each linear system of the form $(W^+ \# W^-)^{-1}v$ is solved by constructing an Extended Krylov subspace.

We test the methods discussed here on the Wikipedia Request for Adminship signed network [54], which is available as part of the Stanford Large Network Dataset Collection (SNAP) [42]. The matrices in this dataset have size 8297, and W^+ and W^- have density (number of nonzero entries divided by the total number of elements)

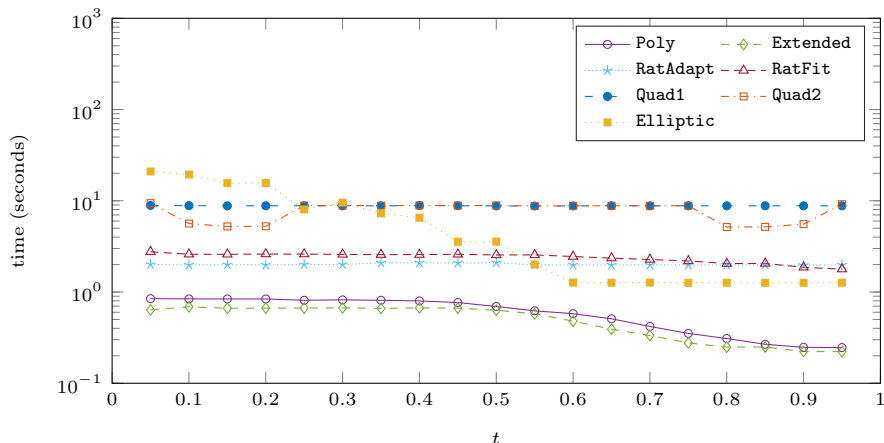


Fig. 4: CPU time needed by the various methods for computing $(A\#_t B)^{-1}v$ with respect to t .

4.10×10^{-3} and 9.98×10^{-4} , respectively. After the preprocessing stage, the size reduces to 6186 and the density to 1.20×10^{-3} and 4.30×10^{-3} , which makes them large and sparse enough to benefit from sparse matrix techniques.

First, we observe that with a similar computational effort, one can obtain a clustering using a weighted geometric mean with $t \neq 1/2$. Figure 3, compares the reordering obtained using $k = 30$ eigenvectors for $t = 0.35$ and $t = 1/2$. A quantitative comparison of the two results is not possible, since there is no widely accepted metric for measuring the quality of the clustering of a signed network. We point out, however, that the reordering for $t = 0.35$ shows a k -balanced behavior: after the reordering, the nonzeros of A^+ tend to appear in blocks along the diagonal, whereas those of A^- are localized in non-diagonal blocks.

Suitable clusterings are provided by using different values of t . An interesting open problem could be to identify the value of t providing the clustering more adherent to the model problem.

In Figure 4 we show how the parameter t influences the CPU time needed by the methods to solve the linear system $(W^+ \#_t W^-)^{-1}v$. Most methods perform better for values of t larger than $1/2$, the execution time of **Quad1** does not seem to depend on t and that of **Quad2** is symmetric with respect to $1/2$. For this network, the best methods, with a comparable CPU time, are **Poly** and **Extended**.

7. Conclusions. We consider several numerical algorithms for the approximation of $(A\#_t B)v$ and $(A\#_t B)^{-1}v$ for $t \in (0, 1)$. These methods exploit rational approximation of the function z^{-t} by either performing numerical quadrature or building a Krylov subspace. In both cases the problem is reduced to the solution of a certain number of linear systems, and thus assessing the performance of any of the algorithms discussed throughout the paper amounts to estimating the number and nature of linear systems to be solved.

The number of linear systems depends on the degree of the quadrature formula, for quadrature methods, and on the dimension of the constructed subspace, for Krylov methods. Note that this number can be efficiently estimated *a priori* in the former

case, by applying the method to either a scalar or a 2×2 case, but cannot be predicted so easily in the latter.

On the other hand, the performance is influenced by the kind of linear system to be solved. For instance, when $t \neq 1/2$ the method `Elliptic` is quasi-optimal with respect to the convergence, being not far from the rational minimax approximation, but it requires the solution of complex linear systems with non-positive definite coefficient, which results in a sensible increase in terms of computational cost. Another example is represented by the extended Krylov subspace method (`Extended`), which despite requiring more linear systems than the other two rational Krylov methods considered in the paper (`RatAdapt` and `RatFit`), is faster when the subspace need to be large. The reason behind this is that since `Extended` solves linear systems all having the same coefficient matrices, it is usually worth computing a factorization, at the price of a usually negligible overhead, in order to make the solution of the successive linear systems faster. The larger the space is, the more this approach pays off.

According to the experimental results in Section 6, the choice of the method should be dictated by the spread of the eigenvalues of the matrix $A^{-1}B$ and the structure of A and B . In extremely well-conditioned cases, we expect all the methods to converge in very few iterations, and it is enough to build a polynomial Krylov space to approximate the solution. For mildly ill-conditioned matrices, `Extended` generates a Krylov subspace which is not too large, and the overhead introduced by the factorization is balanced by the reduction in execution time of the single iterations.

For severely ill-conditioned matrices a general recipe cannot be given, but, in this case, the quadrature methods become competitive. In particular, when $t = 1/2$ or close to 0 and 1, `Elliptic` seems to be the best choice, whereas for intermediate values of t `Quad2` is very effective. The convergence of `Quad1` is considerably slowed down and this method is totally impractical in this case. Krylov methods loose their supremacy because of the growth of the space, which implies a massive overhead due to the Gram–Schmidt orthogonalization of the basis. In principle, this problem could be alleviated by making use of opportune restarting techniques during the construction of the Krylov space. This optimization is currently under investigation and will be the subject of future work.

Acknowledgements. The authors are grateful to Valeria Simoncini, who advised the first author during his MSc thesis, and to Mario Berljafa, for fruitful discussions about the rational Arnoldi methods. The authors wish to thank Nicholas J. Higham for providing useful comments which improved the presentation of the paper, and Daniel Loghin, Pedro Mercado and Francesco Tudisco, who provided some details regarding the large-scale matrices arising from applications.

REFERENCES

- [1] MILTON ABRAMOWITZ AND IRENE STEGUN, *Handbook of Mathematical Functions*, Dover Publications, New York, NY, USA, 10th ed. ed., 1972.
- [2] TSUYOSHI ANDO, *Concavity of certain maps on positive definite matrices and applications to Hadamard products*, Linear Algebra Appl., 26 (1979), pp. 203–241.
- [3] TSUYOSHI ANDO, CHI-KWONG LI, AND ROY MATHIAS, *Geometric means*, Linear Algebra Appl., 385 (2004), pp. 305–334. Special Issue in honor of Peter Lancaster.
- [4] MARIO ARIOLI, DROSOS KOUROUNIS, AND DANIEL LOGHIN, *Discrete fractional Sobolev norms for domain decomposition preconditioning*, IMA J. Numer. Anal., 33 (2011), pp. 318–342.
- [5] MARIO ARIOLI AND DANIEL LOGHIN, *Discrete interpolation norms with applications*, SIAM J. Numer. Anal., 47 (2009), pp. 2924–2951.
- [6] ———, *Spectral analysis of the anisotropic Steklov–Poincaré matrix*, Linear Algebra Appl., 488 (2016), pp. 168–183.
- [7] MARIO BERLJAJA AND STEFAN GÜTTEL, *A Rational Krylov Toolbox for MATLAB*, MIMS EPrint 2014.56, Manchester Institute for Mathematical Sciences, The University of Manchester, UK, 2014. Available for download at <http://guettel.com/rktoolbox/>.
- [8] ———, *Generalized rational Krylov decompositions with an application to rational approximation*, SIAM J. Matrix Anal. Appl., 36 (2015), pp. 894–916.
- [9] MARIO BERLJAJA AND STEFAN GÜTTEL, *The RKFIT algorithm for nonlinear rational approximation*, MIMS EPrint 2015.38, Manchester Institute for Mathematical Sciences, The University of Manchester, UK, 2015.
- [10] RAJENDRA BHATIA, *Matrix Analysis*, vol. 169 of Graduate Texts in Mathematics, Springer-Verlag, New York, NY, USA, 1997.
- [11] ———, *Positive Definite Matrices*, Princeton Series in Applied Mathematics, Princeton University Press, Princeton, NJ, USA, 2007.
- [12] ———, *The Riemannian Mean of Positive Matrices*, Springer Berlin Heidelberg, Berlin, Heidelberg, 2013, pp. 35–51.
- [13] DARIO BINI AND BRUNO IANNAZZO, *The Matrix Means Toolbox*, Last accessed: 17–07–2017.
- [14] JOÃO R. CARDOSO, *Computation of the matrix p -th root and its Fréchet derivative by integrals*, Electron. Trans. Numer. Anal., 39 (2012), pp. 414–436.
- [15] JACOPO CASTELLINI, *Krylov iterative methods for the geometric mean of two matrices times a vector*, Numerical Algorithms, 74 (2017), pp. 561–571.
- [16] TIMOTHY A. DAVIS AND YIFAN HU, *The University of Florida sparse matrix collection*, ACM Trans. Math. Softw., 38 (2011), pp. 1:1–1:25.
- [17] TOBIN A. DRISCOLL, *Algorithm 756: A MATLAB toolbox for Schwarz–Christoffel mapping*, ACM Trans. Math. Softw., 22 (1996), pp. 168–186.
- [18] ———, *Algorithm 843: Improvements to the Schwarz–Christoffel toolbox for MATLAB*, ACM Trans. Math. Softw., 31 (2005), pp. 239–251.
- [19] TOBIN A. DRISCOLL, NICHOLAS HALE, AND LLOYD N. TREFETHEN, *Chebfun Guide*, Pafnuty Publications, 2014.
- [20] TOBIN A. DRISCOLL AND LLOYD N. TREFETHEN, *Schwarz–Christoffel Mapping*, Cambridge Monographs on Applied and Computational Mathematics, Cambridge University Press, Cambridge, UK, 2002.
- [21] VLADIMIR DRUSKIN AND LEONID KNIZHNERMAN, *Extended Krylov subspaces: Approximation of the matrix square root and related functions*, SIAM J. Matrix Anal. Appl., 19 (1998), pp. 755–771.
- [22] CLAUDIO ESTATICO AND FABIO DI BENEDETTO, *Shift-invariant approximations of structured shift-variant blurring matrices*, Numer. Algorithms, 62 (2013), pp. 615–635.
- [23] ANDREAS FROMMER, STEFAN GÜTTEL, AND MARCEL SCHWEITZER, *Efficient and stable Arnoldi restarts for matrix functions based on quadrature*, SIAM J. Matrix Anal. Appl., 35 (2014), pp. 661–683.
- [24] ANDREAS FROMMER AND VALERIA SIMONCINI, *Matrix functions*, in *Model order reduction: theory, research aspects and applications*, vol. 13 of Math. Ind., Springer, Berlin, 2008, pp. 275–303.
- [25] WALTER GAUTSCHI, *A survey of Gauss–Christoffel quadrature formulae*, in E. B. Christoffel (Aachen/Monschau, 1979), Birkhäuser, Basel, Switzerland, 1981, pp. 72–147.
- [26] L. GIRAUD AND J. LANGOU, *When modified Gram–Schmidt generates a well-conditioned set of vectors*, IMA J. Numer. Anal., 22 (2002), pp. 521–528.
- [27] GENE H. GOLUB AND CHARLES F. VAN LOAN, *Matrix Computations*, Johns Hopkins University Press, Baltimore, MD, USA, 4rd ed. ed., 2013.
- [28] STEFAN GÜTTEL AND LEONID KNIZHNERMAN, *Automated parameter selection for rational*

- Arnoldi approximation of Markov functions*, PAMM, 11 (2011), pp. 15–18.
- [29] ———, *A black-box rational Arnoldi variant for Cauchy–Stieltjes matrix functions*, BIT, 53 (2013), pp. 595–616.
- [30] NICHOLAS HALE, NICHOLAS J. HIGHAM, AND LLOYD N. TREFETHEN, *Computing a^α , $\log(a)$ and related matrix functions by contour integrals*, SIAM J. Numer. Anal., 46 (2008), pp. 2505–2523.
- [31] NICHOLAS HALE AND ALEX TOWNSEND, *Fast and accurate computation of Gauss–Legendre and Gauss–Jacobi quadrature nodes and weights*, SIAM J. Sci. Comput., 35 (2013), pp. A652–A674.
- [32] NICHOLAS J. HIGHAM, *Accuracy and Stability of Numerical Algorithms*, Society for Industrial and Applied Mathematics, 2nd ed. ed., 2002.
- [33] ———, *Functions of Matrices: Theory and Computation*, Society for Industrial and Applied Mathematics, Philadelphia, PA, USA, 2008.
- [34] NICHOLAS J. HIGHAM AND LIJING LIN, *An improved Schur–Padé algorithm for fractional powers of a matrix and their Fréchet derivatives*, SIAM J. Matrix Anal. Appl., 34 (2013), pp. 1341–1360.
- [35] BRUNO IANNAZZO, *The geometric mean of two matrices from a computational viewpoint*, Numer. Linear Algebra Appl., 23 (2015), pp. 208–229.
- [36] BRUNO IANNAZZO AND CARLO MANASSE, *A Schur logarithmic algorithm for fractional powers of matrices*, SIAM J. Matrix Anal. Appl., 34 (2013), pp. 794–813.
- [37] BRUNO IANNAZZO AND BEATRICE MEINI, *Palindromic matrix polynomials, matrix functions and integral representations*, Linear Algebra Appl., 434 (2011), pp. 174–184.
- [38] CARL JAGELS AND LOTHAR REICHEL, *The extended Krylov subspace method and orthogonal Laurent polynomials*, Linear Algebra Appl., 431 (2009), pp. 441–458.
- [39] ———, *Recursion Relations for the Extended Krylov Subspace Method*, Linear Algebra Appl., 431 (2009), pp. 441–458.
- [40] LEONID KNIZHNERMAN AND VALERIA SIMONCINI, *A new investigation of the extended Krylov subspace method for matrix function evaluations*, Numer. Linear Algebra Appl., 17 (2010), pp. 615–638.
- [41] JIMMIE LAWSON AND YONGDO LIM, *Weighted means and Karcher equations of positive operators*, Proc. Natl. Acad. Sci. U.S.A., 110 (2013), pp. 15626–15632.
- [42] JURE LESKOVEC AND ANDREJ KREVL, *SNAP Datasets: Stanford large network dataset collection*. <http://snap.stanford.edu/data>, June 2014.
- [43] SHIPING LIU, *Multi-way dual cheeger constants and spectral bounds of graphs*, Advances in Mathematics, 268 (2015), pp. 306 – 338.
- [44] PEDRO MERCADO, FRANCESCO TUDISCO, AND MATTHIAS HEIN, *Clustering signed networks with the geometric mean of Laplacians*, in NIPS 2016. To appear.
- [45] BERESFORD N. PARLETT, *The Symmetric Eigenvalue Problem*, Society for Industrial and Applied Mathematics, Philadelphia, PA, USA, 1998. Unabridged, amended version of book first published by Prentice–Hall in 1980.
- [46] WIESLAW PUSZ AND STANISLAW. L. WRONOWICZ, *Functional calculus for sesquilinear forms and the purification map*, Rep. Math. Phys., 8 (1975), pp. 159–170.
- [47] ANTHONY RALSTON AND PHILIP RABINOWITZ, *A First Course in Numerical Analysis*, Dover Publications, New York, NY, USA, 2nd ed. ed., 1978.
- [48] AXEL RUHE, *Rational Krylov sequence methods for eigenvalue computation*, Linear Algebra Appl., 58 (1984), pp. 391–405.
- [49] ———, *Rational Krylov algorithms for nonsymmetric eigenvalue problems. II. matrix pairs*, Linear Algebra Appl., 197–198 (1994), pp. 283–295.
- [50] ———, *Rational Krylov: A practical algorithm for large sparse nonsymmetric matrix pencils*, SIAM J. Matrix Anal. Appl., 19 (1998), pp. 1535–1551.
- [51] VALERIA SIMONCINI, *A new iterative method for solving large-scale Lyapunov matrix equations*, SIAM J. Sci. Comput., 29 (2007), pp. 1268–1288.
- [52] L.N. TREFETHEN AND D. BAU, *Numerical Linear Algebra*, Society for Industrial and Applied Mathematics, Philadelphia, PA, USA, 1997.
- [53] BJÖRN VON SYDOW, *Error estimates for Gaussian quadrature formulae*, Numer. Math., 29 (1977/78), pp. 59–64.
- [54] ROBERT WEST, HRISTO PASKOV, JURE LESKOVEC, AND CHRISTOPHER POTTS, *Exploiting social network structure for person-to-person sentiment analysis*, Transactions of the Association for Computational Linguistics, 2 (2014), pp. 297–310.