Computing the weighted geometric mean of two large-scale matrices and its inverse times a vector

Fasi, Massimiliano and Iannazzo, Bruno

2016

MIMS EPrint: **2016.29**

Manchester Institute for Mathematical Sciences
School of Mathematics

The University of Manchester

Reports available from: http://eprints.maths.manchester.ac.uk/
And by contacting: The MIMS Secretary
School of Mathematics
The University of Manchester
Manchester, M13 9PL, UK

ISSN 1749-9097

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 the computation of the action of the weighted geometric mean of two large-scale positive definite matrices on a vector. We derive several algorithms, based on numerical quadrature and 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 for the solution of large linear system whose coefficient matrix is a weighted geometric mean. We derive two novel algorithms, based on Gauss–Jacobi quadrature, and tailor an existing technique based on contour integration. On the other hand, we adapt several existing Krylov subspace techniques to the computation of the weighted geometric mean. According to our experiments, both classes of algorithms perform well on some problems but there is no clear winner, while some problem-dependent recommendations are provided.

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 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

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

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 [9, Ch. 4] and Lawson and Lim [36]. 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 [30].

Here we are mostly 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 [2, 3, 4]. The geometric mean of large-scale matrices appears also in image processing [19] and network analysis [43].

In particular, here we want to avoid the explicit computation of the matrix function $A\#_t B$, which may be unduly slow or even practically unfeasible, 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 the computation of a function of a matrix times a vector.

It is well known that the geometric mean $A\#B := A\#_{1/2}B$ (the weighted geometric mean with weight $t = 1/2$) has several nice integral representations [32]. In particular, the formula

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

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 [30]. In particular, we generalise 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

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

*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.

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

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

can be easily derived. The problem is thus reduced to the solution of linear systems and the evaluation of matrix-vector multiplications. Moreover, if $r_i, s_i > 0$ for each 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 analyse three quadrature formulae for $A\#_t B$. The first two are obtained from integral representations of the inverse of real powers [11, 20], by exploiting the fact that $A\#_t B = A(B^{-1}A)^{-t}$, we obtain a formula of the form (2). The third is based on a clever conformal mapping [26], 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 standard techniques for the approximation of $f(Z^{-1}Y)v$, where Z and Y are a large-scale matrices, to our problem. 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 [37, Ch. 15], which has been considered for $(A\#_t B)v$ in previous work [3, 2], but with unsatisfactory results [3, Sect. 6.1.3]. We revise the generalized Lanczos method and then investigate some more powerful Krylov subspace techniques such as the extended Krylov subspace method [18] and the rational Krylov subspace methods [39, 40, 41], with poles chosen according to the adaptive strategy by Güettel and Knizhnerman [25] or the rational Krylov fitting by Berljafa and Güettel [6]. 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 case $t = 1/2$ for sparse matrices of moderate size [12].

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. We denote by $\mathbb{R}^+$ the set of positive real numbers and by $\sigma(A)$ the spectrum of the square matrix A.

Let $A \in \mathbb{C}^{n\times n}$ be diagonalizable with eigenvalues in $\Omega \subset \mathbb{C}$ and let $f : \Omega \to \mathbb{C}$. If $M^{-1}AM = \operatorname{diag}(\lambda_1, \ldots, \lambda_n)$, then $f(A) := M \operatorname{diag}(f(\lambda_1), \ldots, f(\lambda_n))M^{-1}$. Note that if A is Hermitian, then $f(A)$ is Hermitian as well. This definition can be extended to nondiagonalizable matrices [28, 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 $f : \mathbb{R} \to \mathbb{R}^+$. For any $A \in \mathbb{C}^{n\times n}$ positive definite and $B \in \mathbb{C}^{n\times n}$ Hermitian, the matrix $Af(A^{-1}B)$ is Hermitian positive definite.*

Proof. Because of the similarity invariance of matrix functions, we have that $Af(A^{-1}B) = A^{1/2}f(A^{-1/2}BA^{-1/2})A^{1/2}$. Now, $A^{-1/2}BA^{-1/2}$ is Hermitian and diagonalizable with real eigenvalues, 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. ∎

The previous lemma shows that $A\#_t B = A(A^{-1}B)^t$ is positive definite and using other properties of matrix functions one obtains the following equivalent expressions:

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

Another useful property of the weighted geometric mean is

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

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

$$(A\#_t B)^{-1} = \big((AB^{-1})^{-t}A\big)^{-1} = A^{-1}(AB^{-1})^t BB^{-1} = A^{-1}B\#_t AB^{-1}.$$

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 briefly analyse 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 $z \to z^{-t}$ for $t \in (0,1)$ is a Stieltjes function [8, p. 116], which can be written as

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

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

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

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

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

which has been considered in a similar form in order to compute the pth roots [11].

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

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

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

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

These formulae are known as the Gauss–Jacobi quadrature formulae [38, 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

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

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 [21, 44]

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

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 semiaxes 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 function is

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

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 function (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 case the Gauss–Jacobi quadrature applied to (8) converges faster than the same method applied to (7).

We analyse the convergence just for $z \in \mathbb{R}^+$, because we want to apply the formulae to matrices having positive real eigenvalues and the convergence of the quadrature formulae for matrices follows from the convergence of the same formulae for their eigenvalues.

Convergence for the integrand function $f_{1,z}(s)$. Let us start by considering the quadrature formula for $f_{1,z}(s)$, which has only one pole at $\zeta = 1/\mathcal{C}(z)$ and is thus simpler to work with. 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 not difficult 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 is long $|\zeta|$ and its imaginary semiaxis is long $\sqrt{\zeta^2 - 1}$, thus, the sums of its semiaxes is

$$\rho^{(1)}(z) = |\zeta| + \sqrt{\zeta^2 - 1} = \frac{1}{|\mathcal{C}(z)|} + \sqrt{\frac{1}{\mathcal{C}(z)^2} - 1} = \frac{|1+z| + 2\sqrt{z}}{|1-z|} = \frac{1+\sqrt{z}}{|1-\sqrt{z}|} = \frac{1}{|\mathcal{C}(\sqrt{z})|}, \tag{13}$$

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

Convergence for the integrand function $f_{2,z}(s)$. 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

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

Since none of them lies on the interval $[-1,1]$, the integrand function 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

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

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

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

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

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

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. □

So for $t = (p-1)/p$, we have a lower bound for the rate of convergence, namely $\big(1/\rho^{(2)}(z)\big)^2$. For $t \neq (p-1)/p$, by lack of analyticity of the integrand function, 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 which are not of the type $(p-1)/p$.

Comparison. We can compare the bounds for the rates of convergence of the two quadrature formulae, namely $\big(1/\rho^{(1)}(z)\big)^2$, with $\rho^{(1)}(z)$ defined as in (13); and $\big(1/\rho^{(2)}(z)\big)^2$, 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 $z \geqslant 1$.

In a neighbourhood 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 $z \to \infty$, we have

$$1 - \Big(\frac{1}{\rho^{(1)}(z)}\Big)^2 \approx 4z^{-\frac{1}{2}}, \qquad 1 - \Big(\frac{1}{\rho^{(2)}(z)}\Big)^2 \approx 2\sqrt{2\big(1 - \cos(\pi/p)\big)}\, z^{-\frac{1}{2p}}. \tag{16}$$

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 approximation of z^{-t} as $z \to 1$. We give a short proof of this property (see also the one given by Frommer, Güttel and Schweitzer [20]).

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

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.

Let $f_z(s) = \frac{1}{(1-s)+(1+s)z}$, we have that $f_z^{(k)}(s) = (-1)^k k!(z-1)^k f_z^{k+1}(s)$ for $k \geqslant 0$. 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

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

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 $z \to 1$ 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}. □

3.2. Integral representations of $A\#_t B$. The representations in the previous section 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

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

and the corresponding quadrature formula on $N+1$ nodes gives

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

where w_i are the weights of the Gauss–Jacobi quadrature formula with $N+1$ nodes and s_i are the nodes, which belong to the interval $[-1,1]$. Thus, 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

$$
\begin{aligned}
A\#_t B &= \frac{2\sin(\pi(1-t))}{\pi(1-t)} A \int_{-1}^{1} (1-s)^{\frac{2t-1}{1-t}}\big((1+s)^{\frac{1}{1-t}} I + (1-s)^{\frac{1}{1-t}} B^{-1}A\big)^{-1} ds \\
&= \frac{2\sin(\pi(1-t))}{\pi(1-t)} A^{1/2} \int_{-1}^{1} (1-s)^{\frac{2t-1}{1-t}}\big((1+s)^{\frac{1}{1-t}} I + (1-s)^{\frac{1}{1-t}} A^{1/2}B^{-1}A^{1/2}\big)^{-1} ds \cdot A^{1/2}, \\
&= \frac{2\sin(\pi(1-t))}{\pi(1-t)} A \int_{-1}^{1} (1-s)^{\frac{2t-1}{1-t}}\big((1+s)^{\frac{1}{1-t}} B + (1-s)^{\frac{1}{1-t}} A\big)^{-1} B ds,
\end{aligned} \tag{19}
$$

and the corresponding quadrature formula with $N+1$ nodes gives

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

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

3.3. Matrix convergence. In order to analyse 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 matrix C with real and positive eigenvalues. In this case, the functions to be integrated are

$$
f_{1,C}(s) = \big((1-s) + C\,(1+s)\big)^{-1} \text{ and } f_{2,C}(s) = \big((1+s)^{\frac{1}{1-t}} + C\,(1-s)^{\frac{1}{1-t}}\big)^{-1}, \tag{21}
$$

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

When the function to be integrated is analytic, if $K^{-1}CK = \operatorname{diag}(\lambda_1, \dots, \lambda_n)$, then the error in the quadrature formulae with N nodes (defined in (11)), in the spectral norm, can be bounded as

$$
\|R_N(f;C)\| \leqslant \mu_2(K) \|\operatorname{diag}(R_N(f;\lambda_i))\| \leqslant \mu_2(K) \max_{i=1,\dots,n} \{|R_N(f;\lambda_i)|\},
$$

where $\mu_2(K)$ is the condition number of K in the spectral norm. If, moreover, C is Hermitian, we may assume that $\mu_2(K) = 1$. Asymptotically, this bound 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 function $f_{1,C}(s)$. Let the eigenvalues of C be $m = \lambda_1 \leqslant \lambda_2 \leqslant \ldots \leqslant \lambda_{n-1} \leqslant \lambda_n = M$. The infimum of the acceptable values of ρ (the ellipse parameter) is now obtained by minimizing the function $|\zeta| + \sqrt{\zeta^2 - 1}$ for $\zeta \in \sigma(C)$, so that the bound for the rate of convergence, in view of (13), is

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

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

If the matrix C is Hermitian then its condition number in the 2-norm, denoted by $\kappa := \mu_2(C)$, is M/m. If we further assume that $Mm = 1$, then $\kappa = M^2 = 1/m^2$ and since $|C(\sqrt{m})| = |\mathcal{C}(\sqrt{M})|$, we have

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

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

$$\tau^{(1)}(C) = \Big(\frac{\sqrt[4]{\kappa} - 1}{\sqrt[4]{\kappa} + 1}\Big)^2 = \Big(1 - \frac{2}{\sqrt[4]{\kappa} + 1}\Big)^2 \approx 1 - \frac{4}{\sqrt[4]{\kappa}} \approx \exp(-4/\sqrt[4]{\kappa}) = \exp(-4\sqrt[4]{m/M}). \tag{22}$$

Note that the condition $Mm = 1$ 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 Mm is far from 1.

Convergence for the integrand function $f_{2,C}(s)$ and comparison. As before, for a matrix C with real and positive eigenvalues, a bound for the rate of convergence is given by the largest bound for 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 the matrix is Hermitian and $Mm = 1$, then for $t = 1/2$ we can get an asymptotic estimate as $\kappa \to \infty$, which is

$$\tau^{(2)}(C) = \max_{\lambda \in \sigma(C)} \frac{1}{\rho^{(2)}(\lambda)^2} = \Big(\frac{\sqrt{\sqrt{\kappa} + 1}}{\sqrt[4]{\kappa} + 1 + \sqrt{2}\sqrt[8]{k}}\Big)^2 \approx 1 - \frac{2\sqrt{2}}{\sqrt[8]{\kappa}} = \exp(-2\sqrt{2}\sqrt[8]{m/M}). \tag{23}$$

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

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

and this is confirmed by the results of Test 1 in Section 6. So, for a Hermitian matrix and for $t = 1/2$, unless the matrix is very well conditioned/preconditioned ($\kappa_2(C) \leqslant 70$), 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 Hermitian matrix C can be applied.

Let M and 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 [36]

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

Thus, we may assume that $Mm = 1$ replacing the couple (A, B) with $(\widehat{A}, \widehat{B})$, where $\widehat{A} = A/\sqrt{Mm}$ and $\widehat{B} = B$.

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

$$\|\widehat{A}\#_t\widehat{B} - S_N^{(1)}\| = O\big(e^{-4N\sqrt[4]{m/M}}\big), \tag{24}$$

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

$$\|\widehat{A}\#_{1/2}\widehat{B} - S_N^{(2)}\| = O\big(e^{-2\sqrt{2}N\sqrt[8]{m/M}}\big). \tag{25}$$

3.4. An alternative quadrature formula. Another powerful quadrature formula for real matrix powers has been obtained in [26] 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 [26], referring the reader to it for a complete explanation. Let $f : \mathbb{C} \to \mathbb{C}$ be analytic in $\mathbb{C} \setminus (-\infty, 0]$ 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 $f(Z)$ using a quadrature formula with N nodes is given by

$$\frac{-8KZ\sqrt[4]{mM}}{\pi Nk}\,\mathrm{Im}\left(\sum_{j=1}^{N}\frac{f\big(w(t_j)^2\big)\,\mathrm{cn}(t_j)\,\mathrm{dn}(t_j)}{w(t_j)\big(k^{-1}-\mathrm{sn}(t_j)\big)^2}\big(w(t_j)^2I-Z\big)^{-1}\right), \tag{26}$$

where m and M are the minimum and maximum of the spectrum, respectively, $k = -\mathcal{C}(\sqrt[4]{M/m})$, K is the complete elliptic integrals associated with k [26], $w(t) = \sqrt[4]{mM}(k^{-1}+\mathrm{sn}(t))/(k^{-1}-\mathrm{sn}(t))$, $t_j = -K + iK'/2 + 2(j-2^{-1}K)/N$ for $1 \leqslant j \leqslant N$ and $\mathrm{cn}(\cdot), \mathrm{dn}(\cdot)$ and $\mathrm{sn}(\cdot)$ 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 [17].

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

$$\frac{-8K\sqrt[4]{mM}}{\pi Nk}B\,\mathrm{Im}\left(\sum_{j=1}^{N}\frac{f\big(w(t_j)^2\big)\,\mathrm{cn}(t_j)\,\mathrm{dn}(t_j)}{w(t_j)\big(k^{-1}-\mathrm{sn}(t_j)\big)^2}\big(w(t_j)^2A-B\big)^{-1}Av\right), \tag{27}$$

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

Using the identity $A\#_tB = A(A^{-1}B)^t$, for the matrix geometric mean one gets the approximation

$$A\#_tB \approx S_N^{(3)} := \frac{8K\sqrt[4]{mM}}{\pi Nk}A\,\mathrm{Im}\left(\sum_{j=1}^{N}\frac{w(t_j)^{2t}\,\mathrm{cn}(t_j)\,\mathrm{dn}(t_j)}{w(t_j)\big(k^{-1}-\mathrm{sn}\,(t_j)\big)^2}\big(w(t_j)^2A-B\big)^{-1}Bv\right). \tag{28}$$

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 $r_iA + s_iB$ are not positive definite, for some values of i. Moreover, they can be complex matrices, so that we lose the positive definite structure.

The quadrature formula $S_N^{(3)}$ of (28) converges linearly to $A\#_tB$, in particular the following estimate can be deduced from [26, Thm. 3.1]

$$\|A\#_tB - S_N^{(3)}\| = O\big(e^{-2\pi^2N/(\log(M/m)+6)}\big),$$

where m and M are the smallest and largest eigenvalues of $A^{-1}B$, respectively. A comparison with the analogous formula for the two quadrature formulae of Section 3.2, namely (22) and (23), suggests that this formula can converge much faster when M/m becomes very large and this is confirmed by the numerical experiments in Section 6.

4. Krylov subspace methods. We consider some Krylov subspace methods for approximating $(A\#_tB)v = A(A^{-1}B)^tv$. The algorithms are presented for the more general problem $Af(A^{-1}B)v$, with $f : \mathbb{R}^+ \to \mathbb{R}^+$, and A positive definite, with B Hermitian, for which all methods work similarly as they do for the matrix weighted geometric mean.

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 $\{H_k\}_{k=1,\dots,n}$ 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

$$\mathcal{K}_k(A^{-1}B, v) = \mathrm{span}\{v, (A^{-1}B)v, \dots, (A^{-1}B)^{k-1}v\}, \tag{29}$$

where $v_1 = v/\|v\|_M$ and the elements of H_k, which are by definition $h_{i,j} = v_i^*Mv_j$, turn out to be the coefficients of the Gram–Schmidt orthogonalization process [23, Sect. 9.4.1], with respect to the scalar product defined by M. The algorithm has a breakdown when, for some $j \leqslant n$, we have $v_j \in \mathrm{span}\{v_1, \dots, v_{j-1}\}$.

The matrices produced by the algorithm verify $V_n^* M V_n = I_n$, $BV_n = AV_n H_n$ and, for $k < n$,

$$BV_k = AV_k H_k + h_{k+1,k} A v_{k+1} e_k^*, \tag{30}$$

where e_k is the last column of $I_k \in \mathbb{C}^{k\times k}$. Let us note that by left multiplying both sides of equation (30) by $V_k^* M A^{-1}$, we get the explicit expression for the reduced matrix $H_k = V_k^* M A^{-1} B V_k$.

It is well known [28, Chap. 13] that equation (30) 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 $v = v_1 \|b\|_M = V_k e_1 \|v\|_M$, we obtain the approximation

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

which is useful, in practice, when it is a good approximation for values of 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 [37, Ch. 15], Arioli and Loghin [3] 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 generalised Lanczos algorithm, and this has two principal merits. On the one hand, the computation of the each v_k requires a fixed number of arithmetic operations, which considerably decreases the execution time of the algorithm, while 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: in fact, we can use the Arnoldi process to construct a basis of $\mathcal{K}_k(A^{-1}B, v)$ of (29) which is B-orthonormal. In this case, we get $BV_n = AV_n H$ with $V_n^* B V_n = I_n$ and the matrices $H_k = V_k^* B A^{-1} B V_k$ turn out to be tridiagonal.

In principle, a scalar product associated to an arbitrary positive definite matrix M might be used in the Arnoldi process to construct a basis of $\mathcal{K}_k(A^{-1}B, v)$, and a 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. The Hessenberg matrices H_k are Hermitian (and thus tridiagonal) if and only if $MA^{-1}B = BA^{-1}M$.*

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

4.2. Generalized Extended Krylov subspace method. The standard extended Krylov methods [18, 42] can be easily generalized to build an M-orthonormal basis of the extended Krylov subspace

$$\mathcal{E}_k(A^{-1}B, v) = \operatorname{span}\left\{v, A^{-1}Bv, B^{-1}Av, \left(A^{-1}B\right)^2 v, \ldots, \left(B^{-1}A\right)^{\frac{k}{2}-1} v, \left(A^{-1}B\right)^{\frac{k}{2}} v\right\},$$

if k is even and

$$\mathcal{E}_k(A^{-1}B, v) = \operatorname{span}\left\{v, A^{-1}Bv, B^{-1}Av, \left(A^{-1}B\right)^2 v, \ldots, \left(A^{-1}B\right)^{\frac{k-1}{2}} v, \left(B^{-1}A\right)^{\frac{k-1}{2}} v\right\},$$

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 $\{v_k\}_{k=1,\ldots,n}$ and a sequence of Hessenberg matrices with an additional subdiagonal $\{T_k\}_{k=1,\ldots,n}$ with $T_k \in \mathbb{C}^{k\times k}$. We choose to use the letter T here, because we want to stress that T_k does not contain the orthogonalization coefficients of the Gram–Schmidt process applied to the set $\{v_1, \ldots, v_k\}$. The interplay between orthogonalization coefficients and T_k, for the extended Krylov subspace methods, are discussed by Simoncini [42] and Jagels and Reichel [34, 33].

If we define $V_k = [v_1|\cdots|v_k]$ as the M-orthonormal basis of $\mathcal{E}_k(A^{-1}B, v)$, then the matrices produced by the algorithm verify $BV_k = AV_k T_k$ and $V_n^* M V_n = I_n$, while for k even and $k < n$

$$BV_k = AV_k T_k + \widetilde{T} A \left[v_{k+1}|v_{k+2}\right] E_k, \tag{31}$$

where $T_k \in \mathbb{C}^{k\times k}$, $\widetilde{T} = [v_{k+1}|v_{k+2}]^* MA^{-1}B\,[v_{k+1}|v_{k+2}] \in \mathbb{C}^{2\times 2}$, $E_k \in \mathbb{C}^{2\times k}$ 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 space at step k.

Considerations similar to the previous case let us conclude that $T_k = V_k^* MA^{-1}BV_k$ and thus that Proposition 4.1 is valid for the extended method as well. The choice $M = A$, seems to be again the most natural. Moreover, for any k the function $Af(A^{-1}B)v$ can be approximated by means of equation

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

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

4.3. Generalized rational Krylov subspace methods. The rational Arnoldi algorithm [39, 41] 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 $\xi_i \notin \sigma\left(A^{-1}B\right) \cup \{0\}$, we can construct a basis of the rational Krylov subspaces (we set $1/\infty = 0$)

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

by considering $v_1 = v/\|v\|_M$ and constructing

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

which is M-orthogonalized with respect to $v_1, \dots, v_j$, obtaining

$$h_{ij} = w_j^* Mv_i, \qquad \widetilde{w_j} = w_j - \sum_{i=1}^{j} h_{ij}v_j, \qquad h_{j+1,j} = \|\widetilde{w_j}\|_M, \qquad v_{j+1} = \widetilde{w_j}/h_{j+1,j}.$$

In this way, we obtain the rational Arnoldi decomposition

$$BV_k(I_n + H_kD_k) + \frac{h_{k+1,k}}{\xi_k}Bv_{k+1}e_k^* = AV_kH_k + h_{k+1,k}Av_{k+1}e_k^*$$

where $D_k = \operatorname{diag}(1/\xi_1, \dots, 1/\xi_k)$, H_k is the matrix containing the entries h_{ij} and $V_k = [v_1|\cdots|v_k]$ 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 [6, Sect. 3].

If the last pole is at infinity, then the approximation

$$BV_k(I_n + H_kD_k) \approx AV_kH_k$$

might be considered and thus we get

$$Af(A^{-1}B)v \approx AV_k f(H_k(I_n + H_kD_k)^{-1})e_1\|v\|_M. \tag{32}$$

Notice that in this case $H_k(I_n + H_kD_k)^{-1} = V_k^* MA^{-1}BV_k$, which is Hermitian if M commutes with $A^{-1}B$. Thus the argument of the function f to be evaluated is a normal matrix and the evaluation can be done by diagonalization.

Note that the Krylov methods described in Section 4.1 and Section 4.2 are in fact rational Krylov methods 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 apparently well-suited to the problem $f(A)b$.

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

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

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 (28) the approximation

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

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 $(A^{-1}B)^{-t}A^{-1}v$, which can be obtained by computing first $w = A^{-1}v$ and then approximating $(A^{-1}B)^{-t}w$ with any of the Krylov suspace methods described in Section 4.

6. Numerical tests. By means of numerical experiments, we illustrate the behaviour 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 have been made using MATLAB R2015a (8.5.0) on a machine equipped with an Intel i5-3570 Processor running at 3.40GHz and 8GB of dedicated RAM memory.

We compare the following methods:

1. The generalized Arnoldi algorithm [37] (`Poly`);
2. The extended Krylov subspace method [18] (`Extended`);
3. A rational Krylov subspace method, with poles chosen according to the adaptive strategy of Güttel and Knizhnermann [24] (`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 [6] (`RatFit`);
5. The quadrature formula (18) (`Quad1`);
6. The quadrature formula (20) (`Quad2`);
7. The quadrature formula (28) (`Elliptic`).

Krylov subspace methods. To make the computed basis orthonormal, in our implementations of the Krylov subspace methods we use the modified Gram–Schmidt method with reorthogonalization [22], to achieve better numerical stability. When computing the approximation of $Af(A^{-1}B)v$, we have the choice between using the projection of $A^{-1}B$ onto the Krylov subspace or the orthonormalization coefficients. 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 leads to a larger computational cost, due to one or more additional matrix-vector products and/or linear system solve per step. However, while the matrix obtained by the orthogonalization procedure is numerically not Hermitian, and it is not Hermitian when rational Arnoldi is used as described in Section 4.3, the

projected matrix is guaranteed to be positive definite, which preserves the structure of the problem and makes the evaluation of f much cheaper.

Even though it would be more natural to use projections, in our experiments 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 using algorithms for the real power of dense matrices [31, 29]. We stress that, in our tests, this choice did not reduce the accuracy of the final result, and only marginally affected the convergence speed.

For the rational Krylov methods, the poles are chosen according to either the adaptive strategy by Güttel and Knizhnerman [24] or the function `rkfit` from the `rktoolbox` [5], based on an algorithm by Berljafa and Güttel [6, 7]. In our implementation, when the matrices are larger than 1000×1000, we get the poles by running `rkfit` on a surrogate problem of size 1000×1000 whose setup requires a rough estimate of the extrema of the spectrum of $A^{-1}B$. In the case of rational Krylov methods, in order to obtain an approximation of $Af(A^{-1}B)v$, we use the estimate (32), even when the last pole is not at infinity, as done by Güttel and Knizhnermann [24], with good results.

As a stopping criterion for the Krylov subspace methods, we use the estimate [35]

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

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 5 in our experiments.

Quadrature methods. For quadrature methods related to the Gauss–Jacobi quadrature, namely (18) and (20), the nodes and the weights have been generated using the function `jacpts` of `Chebfun` [16], which is based on an algorithm by Hale and Townsend [27]. The scaling technique described at the end of Section 3.3 has been used to accelerate the convergence.

For `Quad2` we use the quadrature formula (20) when $t \geqslant 1/2$, and if $t < 1/2$ 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 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 for the matrices we consider.

Regarding the method described in Section 3.4 we adapt the implementation given by Hale, Higham and Trefethen [26], which exploits the elliptic functions `ellipkjc` and `ellipkkp` from Driscoll's *Schwarz–Christoffel Toolbox* [14, 15]. 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 $A^{-1}B$.

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 to the couple (B, A) (or (A, B)). In Table 1 we give a synoptic comparison of the key features of the methods.

Test 1. In Section 3, we have considered two Gauss–Jacobi quadrature formulae for z^{-t}, based on different integral representations, where we have denoted by `Quad1` the one based on (7) and by `Quad2` the one based on (8). We have derived a bound for the rate of convergence of the quadrature formulae, which for `Quad1` is $|\mathcal{C}(\sqrt{z})|^2$, with $\mathcal{C}(x) = \frac{1-x}{1+x}$, while for `Quad2` is $(1/\rho^{(2)}(z))^2$, where $\rho^{(2)}$ is defined in (15). 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, whose largest and smallest eigenvalues in modulus are m and M, respectively, and scaled so that $Mm = 1$, then the first formula seems to be more convenient for well conditioned matrices, say with $M/m < 70$.

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.

Legend	Method	Spectrum	Systems	Update
	`Poly`	no	same lhs	yes
	`Extended`	no	same lhs	yes
	`RatAdapt`	yes	shifted pd	yes
	`RatFit`	yes	shifted pd	yes
	`Quad1`	yes	shifted pd	no
	`Quad2`	yes	shifted pd	no
	`Elliptic`	yes	shifted	no

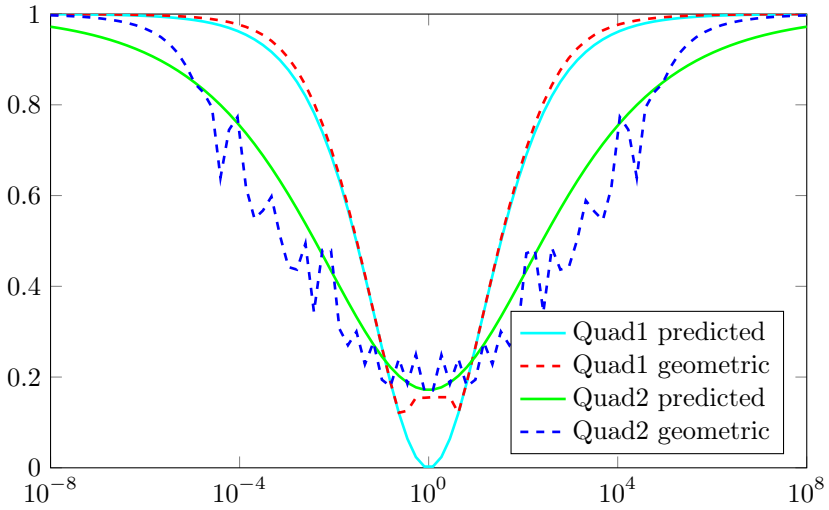


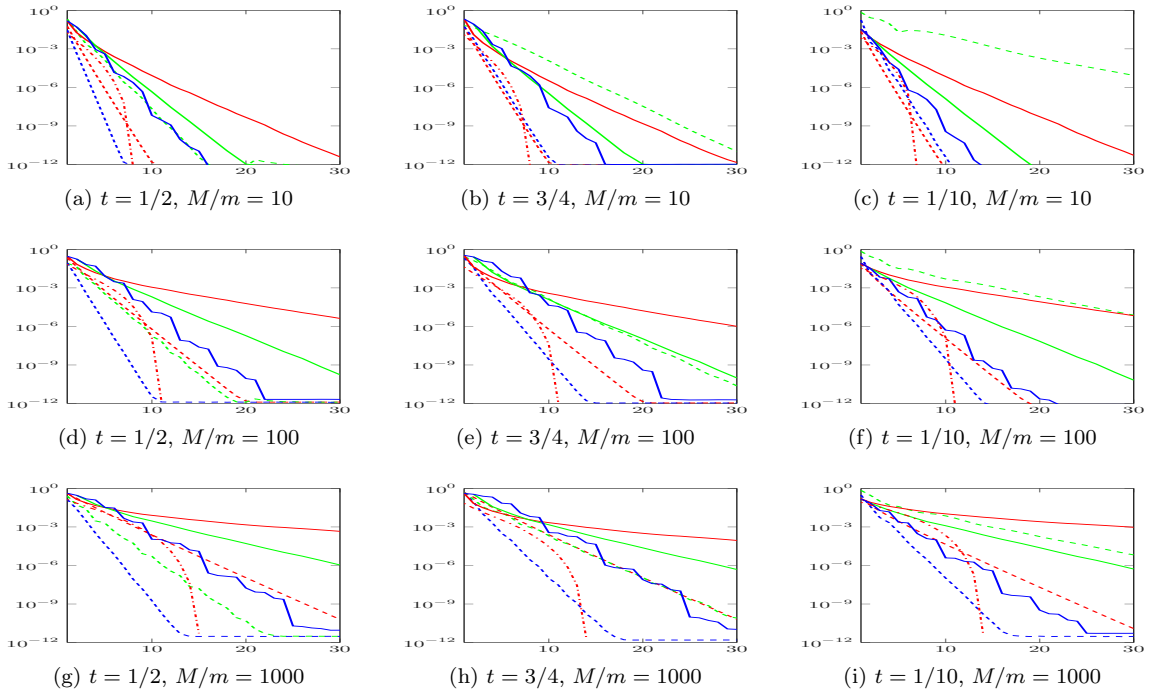
Fig. 1: Comparison of the parameters of convergence (on the y-axis) of the two Gaussian quadrature formulae for $z^{-1/2}$ (on the semilogarithmic x-axis).

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 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 so that the integrand function is not analytic, provided that $t \geqslant 1/2$. 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 the quadrature methods considered so far depends on the conditioning of the matrix $A^{1/2}B^{-1}A^{1/2}$ 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 1000×1000, so that the 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 [30, Sec. 3], which is implemented by the `sharp` function of the `Matrix Means Toolbox` [10].

For each method, the relative forward error of the computed value $\widetilde{w}$ with respect to the reference



(a) $t = 1/2$, $M/m = 10$ (b) $t = 3/4$, $M/m = 10$ (c) $t = 1/10$, $M/m = 10$

(d) $t = 1/2$, $M/m = 100$ (e) $t = 3/4$, $M/m = 100$ (f) $t = 1/10$, $M/m = 100$

(g) $t = 1/2$, $M/m = 1000$ (h) $t = 3/4$, $M/m = 1000$ (i) $t = 1/10$, $M/m = 1000$

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 $M/m \in \{10, 100, 1000\}$, where M and m are the extreme eigenvalues of $A^{-1}B$. 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.

value, namely

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

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 dependence of the convergence on the conditioning of $A^{1/2}B^{-1}A^{1/2}$ also for the Krylov subspace methods.

The final accuracy of all methods is comparable, while we observe a different convergence behaviour for $t = 1/2$ and for $t \neq 1/2$.

For $t = 1/2$, `Elliptic` generates the best rational relative minimax approximation of the function $z^{-1/2}$ on the interval $[m, M]$, where m and M are the extreme eigenvalues of $A^{-1}B$ [6, 26]. This is the reason why it converges faster than the other methods, which produce a different rational approximation to $z^{-1/2}$. We note that `RatFit` converges in a similar number of steps for $t = 1/2$ and is usually the one which achieves the fastest convergence in all the other cases.

We observe that `Quad2` converges much faster than `Quad1` as M/m grows, as predicted in (24). 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 behaviour they have for $t = 1/2$. The `Elliptic` method does not produce the best rational approximation anymore, and thus is not the fastest method, but nevertheless shows a remarkably fast convergence. Finally, the behaviour of `Quad2` degrades fasts as t gets far from $t = 1/2$, this has been partly explained in Section 3.

Test 3. To illustrate the behaviour of the methods when dealing with large-scale matrices, we consider pairs of positive definite matrices from *the University of Florida Sparse Matrix Collection* [13].

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 A and B, we were not able to get any result. Since this dataset

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 the identity has been added to the two matrices.

Dataset	M/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	

is interesting being the only with non-banded matrices, we have tamed the conditioning of the dataset, adding the identity to both matrices.

First, we study the convergence of the methods as in Test 2. In Figure 3 we show the history of convergence of the methods for $t = 1/2$, $t = 3/4$ and $t = 1/10$ for the four datasets. As a reference for the forward error, we consider the solution computed by using `RatFit` to build a larger space (of size between 25 and 30 in this example).

Then, we compare the CPU time required to fulfill the stopping criterion or to reach the largest number of nodes of quadrature/dimension of the Krylov subspace.

The results, given in Table 3, show that even if the rational Krylov methods need fewer steps, they require the solution of a sequence of shifted linear systems and an estimate of the extreme eigenvalues of $A - \lambda B$, while 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 factorisation 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 matrix by applying an approximate symmetric minimum degree permutation, which we compute by means of MATLAB `symamd` function. For this reason, in our implementation, among Krylov methods, `Poly` is the fastest choice when the problem is extremely well-conditioned, and `Extended` is the fastest in all the other cases.

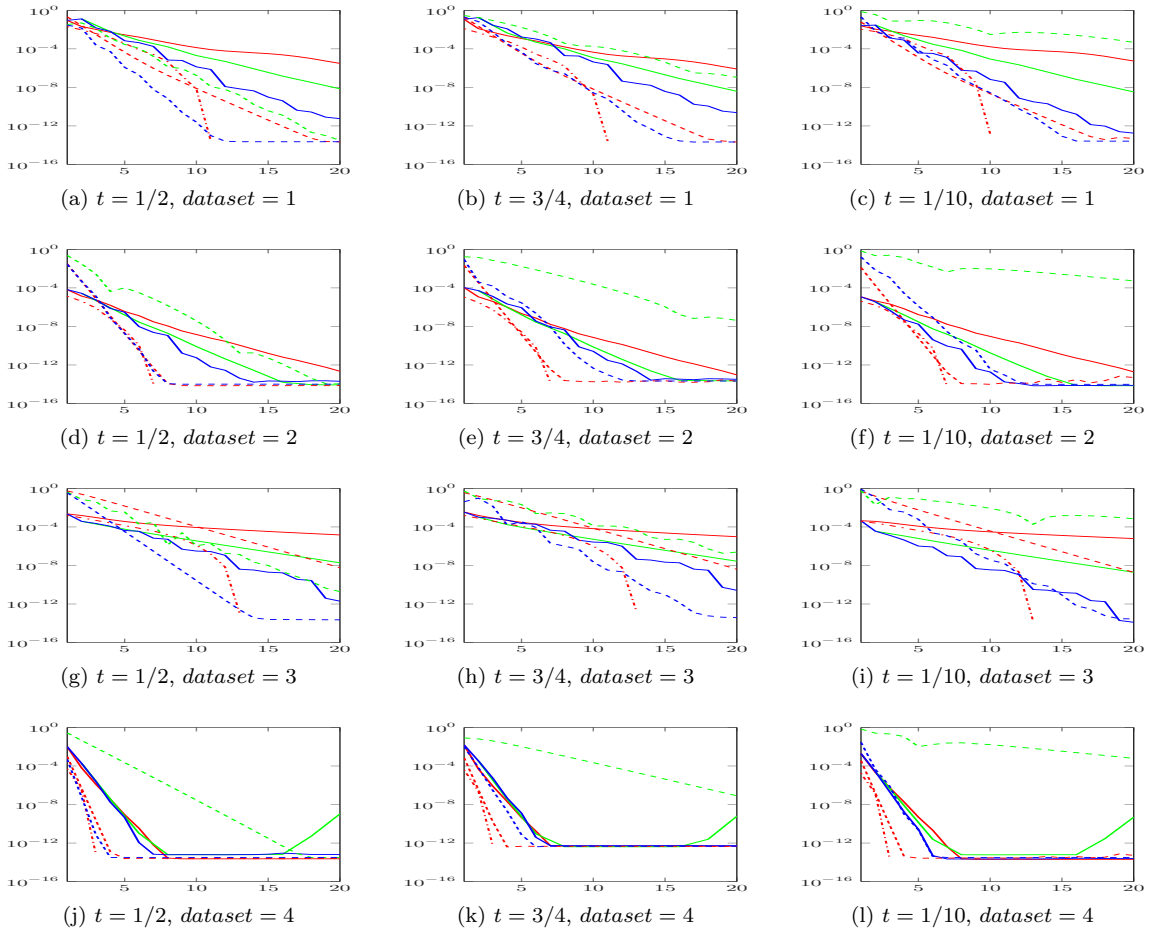
It is worth pointing out that our results are just indicative and do not represent exactly what would happen if high performance implementations were used. In particular, orthogonalisation is, in our implementation, unduly costly, since the control flow is managed at MATLAB level, and this in particular penalises `RatFit`, which heavily relies on orthogonalisation to solve the surrogate problem and find the poles.

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.

Test 4. As a final test we use the techniques of Section 5 to solve a linear system whose coefficient matrix is a weighted geometric mean. This problem arises in a domain decomposition method for PDEs [4].

In [4], the authors consider as a preconditioner the matrix

$$H_\alpha = h_x M_n \left(\frac{1}{h_x^2} M_n^{-1} T_n \right)^{1-\alpha} = \frac{1}{h_x} (h_x^2 M_n) \#_{1-\alpha} T_n = h_x^{2\alpha-1} (T_n \#_\alpha M_n),$$



(a) $t = 1/2$, $dataset = 1$ (b) $t = 3/4$, $dataset = 1$ (c) $t = 1/10$, $dataset = 1$

(d) $t = 1/2$, $dataset = 2$ (e) $t = 3/4$, $dataset = 2$ (f) $t = 1/10$, $dataset = 2$

(g) $t = 1/2$, $dataset = 3$ (h) $t = 3/4$, $dataset = 3$ (i) $t = 1/10$, $dataset = 3$

(j) $t = 1/2$, $dataset = 4$ (k) $t = 3/4$, $dataset = 4$ (l) $t = 1/10$, $dataset = 4$

Fig. 3: Convergence of the methods in Table 1 for computing $(A\#_t B)v$ for the datasets of Table 2 and for $t \in \{1/2, 3/4, 1/10\}$. 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.

where $\alpha \in [0,1]$, $h_x > 0$ is a discretization parameter and the matrices $M_n, T_n \in \mathbb{R}^{n^2 \times n^2}$ are

$$M_n = \frac{1}{6}\begin{bmatrix} 4 & 1 & & & \\ 1 & 4 & 1 & & \\ & \ddots & \ddots & \ddots & \\ & & 1 & 4 & 1 \\ & & & 1 & 4 \end{bmatrix}, \qquad T_n = \begin{bmatrix} 2 & -1 & & & \\ -1 & 2 & -1 & & \\ & \ddots & \ddots & \ddots & \\ & & -1 & 2 & -1 \\ & & & -1 & 2 \end{bmatrix}.$$

In Table 5 we report the performance of the algorithms on the test problem considered in [4, Sect. 5.1], which we instantiate with the same data as in the original paper, corresponding to four levels of refinement of the mesh, and thus the matrices described in Table 4. We keep the notation of the original paper, so that the matrices are of size $n^2 \times n^2$ rather $n \times n$.

We compute the solution of the linear system

$$H_{\widehat{\alpha}} y = v, \tag{33}$$

with a random right hand side $v \in \mathbb{R}^{n^2}$. For each dataset, we consider three different values of $\widehat{\alpha}$, defined by

$$\widehat{\alpha} := \widehat{\alpha}(\vartheta) = \frac{1+\vartheta}{2}, \qquad \vartheta = \frac{a}{\log h_x},$$

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).

	t	Poly	Ext	RatAdapt	RatFit	Quad1	Quad2	Elliptic
1	0.50	2.4 (45)	**1.1 (24)**	1.8 (16)	2.8 (10)	2.6 (20)	2.4 (20)	1.8 (11)
1	0.75	2.1 (39)	**1.1 (24)**	1.8 (17)	2.5 (10)	2.4 (20)	3.3 (33)	4.8 (15)
1	0.10	2.7 (48)	**1.1 (24)**	1.8 (16)	2.6 (10)	2.3 (19)	7.3 (82)	4.3 (13)
2	0.50	7.9 (08)	**7.8 (08)**	12.3 (08)	20.9 (07)	22.5 (11)	28.7 (17)	19.6 (07)
2	0.75	7.7 (08)	**7.5 (08)**	14.7 (10)	20.8 (07)	21.6 (10)	42.9 (34)	68.9 (10)
2	0.10	7.9 (08)	**7.5 (08)**	9.4 (06)	20.6 (07)	21.6 (10)	88.9 (83)	62.5 (09)
3	0.50	32.9 (89)	**4.8 (28)**	9.2 (18)	10.5 (13)	13.2 (29)	11.9 (23)	7.8 (12)
3	0.75	28.7 (83)	**4.9 (30)**	10.0 (19)	9.8 (12)	13.3 (29)	15.2 (33)	18.4 (15)
3	0.10	29.9 (86)	**4.0 (24)**	6.4 (13)	9.0 (12)	13.5 (28)	34.9 (82)	17.6 (14)
4	0.50	**26.3 (03)**	37.4 (04)	28.6 (04)	77.8 (03)	78.8 (04)	127.7 (16)	81.4 (04)
4	0.75	**20.6 (03)**	36.7 (04)	33.9 (05)	75.3 (03)	77.5 (04)	173.5 (29)	206.0 (06)
4	0.10	**16.9 (02)**	28.5 (02)	23.3 (03)	74.9 (03)	77.2 (04)	384.4 (83)	205.9 (06)

Table 4: The datasets for Test 4.

	n	n^2	M/m	h_x
1	129	16641	3.36×10^8	1.54×10^2
2	257	66049	5.30×10^9	7.75×10^3
3	513	263169	8.41×10^{10}	3.89×10^3
4	1025	1050625	1.34×10^{12}	1.95×10^3

corresponding to $a = 0, 2, 4$.

7. Conclusions. We have considered several numerical algorithms to approximate $(A\#_t B)v$ 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 boils down to an estimation of the number and considerations on the 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 built subspace, for Krylov methods. Note that this number can be efficiently estimated *a priori* in the former case, by applying the method to the scalar case.

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, which despite requiring more linear systems than `RatAdapt` and `RatFit`, is faster than them when the space 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 factorisation, 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 $A^{-1}B$. In extremely well-conditioned cases, we expect all the methods to converge in very few iterations, and the best choice is simply to build a polynomial Krylov space to approximate the solution. For mildly ill-conditioned matrices, `Extended` generates a Krylov space which is not too large, and the overhead introduced by the factorisations is balanced by the reduction in execution time of the single iterations.

For severely ill-conditioned matrices, the most efficient algorithms are to be found among quadrature methods. In particular, when $t = 1/2$ or close to 0 and 1, `Elliptic` seems to be the best choice,

Table 5: Comparison of the algorithms presented in the paper, when applied to (33), in terms of CPU time (in seconds) and number of linear systems to be solved (between parentheses).

	a	Poly	Ext	RatAdapt	RatFit	Quad1	Quad2	Elliptic
1	0	47.7 (0326)	3.8 (132)	1.6 (049)	**0.6 (020)**	2.0 (648)	0.8 (064)	0.6 (023)
1	2	181.7 (0549)	5.3 (180)	1.8 (055)	0.6 (022)	15.9 (636)	**0.3 (043)**	0.3 (021)
1	4	180.4 (0566)	6.2 (172)	1.0 (043)	0.8 (021)	14.0 (554)	8.3 (201)	**0.3 (025)**
2	0	129.8 (0335)	16.6 (156)	8.3 (070)	2.0 (019)	7.0 (1134)	1.2 (090)	**1.1 (021)**
2	2	1076.2 (0888)	40.1 (250)	10.1 (079)	2.0 (022)	36.4 (1118)	**1.1 (049)**	1.4 (023)
2	4	1568.4 (1032)	44.5 (260)	8.4 (070)	2.5 (022)	34.7 (1045)	1.2 (056)	**0.9 (024)**
3	0	104.1 (0152)	65.1 (150)	174.5 (163)	6.7 (017)	50.9 (1954)	4.3 (077)	**3.9 (022)**
3	2	7244.0 (1239)	265.8 (322)	234.4 (183)	5.1 (020)	111.3 (1928)	**4.2 (047)**	4.3 (020)
3	4	15372.0 (1731)	349.2 (366)	197.5 (168)	5.4 (021)	116.7 (1834)	**4.3 (042)**	4.7 (022)
4	0	33.6 (0036)	56.5 (062)	425.0 (131)	17.4 (014)	250.1 (2000)	20.9 (109)	**9.1 (001)**
4	2	16610.9 (1106)	1730.7 (374)	1659.3 (268)	18.3 (017)	265.5 (2000)	16.4 (059)	**14.0 (019)**
4	4	59405.2 (2000)	2840.6 (494)	1752.3 (268)	17.5 (019)	272.0 (2000)	**14.9 (033)**	15.6 (020)

whereas for intermediate values of t `Quad2` appears to be preferable. 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 Graham–Schmidt orthogonalisation 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 optimisation is currently under investigation and is the topic of some future work.

Acknowledgements. The authors are grateful to Valeria Simoncini, who advised the first author during his MSc thesis, providing guidance and countless suggestions for the improvement to the implementation of Krylov methods, and Mario Berljafa, for fruitful discussions about the RKFIT algorithm and the use of the Rational Krylov Toolbox. The authors wish to thank Nicholas J. Higham for providing feedback on the manuscript and useful comments which improved the presentation of the paper and Daniel Loghin who provided suggestions on the matrices to be used in the numerical tests on large-scale matrices.

REFERENCES

[1] Milton Abramowitz and Irene Stegun, *Handbook of Mathematical Functions*, Dover Publications, New York, NY, USA, 10th ed. ed., 1972.
[2] Mario Arioli, Drosos Kourounis, and Daniel Loghin, *Discrete fractional Sobolev norms for domain decomposition preconditioning*, IMA J. Numer. Anal., 33 (2011), pp. 318–342.
[3] Mario Arioli and Daniel Loghin, *Discrete interpolation norms with applications*, SIAM J. Numer. Anal., 47 (2009), pp. 2924–2951.
[4] ———, *Spectral analysis of the anisotropic Steklov–Poincaré matrix*, Linear Algebra Appl., 488 (2016), pp. 168–183.
[5] Mario Berljafa 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/`.
[6] ———, *Generalized rational Krylov decompositions with an application to rational approximation*, SIAM J. Matrix Anal. Appl., 36 (2015), pp. 894–916.
[7] Mario Berljafa 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.
[8] Rajendra Bhatia, *Matrix Analysis*, vol. 169 of Graduate Texts in Mathematics, Springer-Verlag, New York, NY, USA, 1997.
[9] ———, *Positive Definite Matrices*, Princeton Series in Applied Mathematics, Princeton University Press, Princeton, NJ, USA, 2007.
[10] Dario Bini and Bruno Iannazzo, *The Matrix Means Toolbox*, Last accessed: 05–02–2016.
[11] 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.
[12] Jacopo Castellini, *Krylov iterative methods for the geometric mean of two matrices times a vector*, (*Submitted for pubblication*).
[13] Timothy A. Davis and Yifan Hu, *The University of Florida sparse matrix collection*, ACM Trans. Math. Softw., 38 (2011), pp. 1:1–1:25.
[14] Tobin A. Driscoll, *Algorithm 756: A MATLAB toolbox for Schwarz–Christoffel mapping*, ACM Trans. Math. Softw., 22 (1996), pp. 168–186.

[15] ———, *Algorithm 843: Improvements to the Schwarz–Christoffel toolbox for MATLAB*, ACM Trans. Math. Softw., 31 (2005), pp. 239–251.
[16] Tobin A. Driscoll, Nicholas Hale, and Lloyd N. Trefethen, *Chebfun Guide*, Pafnuty Publications, 2014.
[17] Tobin A. Driscoll and Lloyd N. Trefethen, *Schwarz–Christoffel Mapping*, Cambridge Monographs on Applied and Computational Mathematics, Cambridge University Press, Cambridge, UK, 2002.
[18] 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.
[19] Claudio Estatico and Fabio Di Benedetto, *Shift-invariant approximations of structured shift-variant blurring matrices*, Numer. Algorithms, 62 (2013), pp. 615–635.
[20] 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.
[21] Walter Gautschi, *A survey of Gauss-Christoffel quadrature formulae*, in E. B. Christoffel (Aachen/Monschau, 1979), Birkhäuser, Basel, Switzerland, 1981, pp. 72–147.
[22] 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.
[23] Gene H. Golub and Charles F. Van Loan, *Matrix Computations*, Johns Hopkins University Press, Baltimore, MD, USA, 4rd ed. ed., 2013.
[24] Stefan Güttel and Leonid Knizhnerman, *Automated parameter selection for rational Arnoldi approximation of Markov functions*, PAMM, 11 (2011), pp. 15–18.
[25] ———, *A black-box rational Arnoldi variant for Cauchy–Stieltjes matrix functions*, BIT, 53 (2013), pp. 595–616.
[26] 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.
[27] 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.
[28] Nicholas J. Higham, *Functions of Matrices: Theory and Computation*, Society for Industrial and Applied Mathematics, Philadelphia, PA, USA, 2008.
[29] 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.
[30] Bruno Iannazzo, *The geometric mean of two matrices from a computational viewpoint*, Numer. Linear Algebra Appl., 23 (2015), pp. 208–229.
[31] Bruno Iannazzo and Carlo Manasse, *A Schur logarithmic algorithm for fractional powers of matrices*, SIAM J. Matrix Anal. Appl., 34 (2013), pp. 794—-813.
[32] Bruno Iannazzo and Beatrice Meini, *Palindromic matrix polyomials, matrix functions and integral representations*, Linear Algebra Appl., 434 (2011), pp. 174–184.
[33] Carl Jagels and Lothar Reichel, *The extended Krylov subspace method and orthogonal Laurent polynomials*, Linear Algebra Appl., 431 (2009), pp. 441–458.
[34] ———, *Recursion Relations for the Extended Krylov Subspace Method*, Linear Algebra Appl., 431 (2009), pp. 441–458.
[35] 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.
[36] 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.
[37] 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.
[38] Anthony Ralston and Philip Rabinowitz, *A First Course in Numerical Analysis*, Dover Publications, New York, NY, USA, 2nd ed. ed., 1978.
[39] Axel Ruhe, *Rational Krylov sequence methods for eigenvalue computation*, Linear Algebra Appl., 58 (1984), pp. 391–405.
[40] ———, *Rational Krylov algorithms for nonsymmetric eigenvalue problems. II. matrix pairs*, Linear Algebra Appl., 197–198 (1994), pp. 283–295.
[41] ———, *Rational Krylov: A practical algorithm for large sparse nonsymmetric matrix pencils*, SIAM J. Matrix Anal. Appl., 19 (1998), pp. 1535–1551.
[42] Valeria Simoncini, *A new iterative method for solving large-scale lyapunov matrix equations*, SIAM J. Sci. Comput., 29 (2007), pp. 1268–1288.
[43] Francesco Tudisco, *Personal communication.*
[44] Björn von Sydow, *Error estimates for Gaussian quadrature formulae*, Numer. Math., 29 (1977/78), pp. 59–64.