Contents

1

Matrix functions

Andreas Frommer[*] and Valeria Simoncini[†]

No Institute Given

1.1 Introduction

In this chapter, we give an overview on methods to compute functions of a (usually square) matrix A with particular emphasis on the matrix exponential and the matrix sign function. We will distinguish between methods which indeed compute the entire matrix function, i.e. they compute a matrix, and those which compute the action of the matrix function on a vector. The latter task is particularly important in the case where we have to deal with a very large (and possibly sparse) matrix A or in situations, where A is not available as a matrix but just as a function which returns Ax for any input vector x. Computing the action of a matrix function on a vector is a typical model reduction problem, since the resulting techniques usually rely on approximations from small-dimensional subspaces.

This chapter is organized as follows: In section 1.2 we introduce the concept of a matrix function $f(A)$ in detail, essentially following [38] and [27]. Section 1.3 gives an assessment of various general computational approaches for either obtaining the whole matrix $f(A)$ or its action $f(A)v$ on a vector v. Sections 1.4 and 1.5 then give much more details for two specific functions, the exponential and the sign functions, which, as we will show, are particularly important in many areas like control theory, simulation of physical systems and other application fields involving the solution of certain ordinary or partial differential equations. The applicability of matrix functions in general, and of the exponential and the sign functions in particular, is vast. However, we will limit our discussion to characterizations and to application problems that are mostly related to Model Order Reduction. For a comprehensive analysis of matrix functions and their computation we refer to the recent book by Nick Higham [34].

[*] Fachbereich Mathematik und Naturwissenschaften, Universität Wuppertal, D-42097 Wuppertal, Germany, frommer@math.uni-wuppertal.de

[†] Dipartimento di Matematica, Università di Bologna, Piazza di Porta S. Donato 5, I-40127 Bologna, and CIRSA, Ravenna, Italy valeria.simoncini@unibo.it

1.2 Matrix Functions

In this section we address the following general question: Given a function $f : \mathbb{C} \to \mathbb{C}$, is there a canonical way to extend this function to square matrices, i.e. to extend f to a mapping from $\mathbb{C}^{n\times n}$ to $\mathbb{C}^{n\times n}$? If f is a polynomial p of degree d, $f(z) = p(z) = \sum_{k=0}^{d} a_k z^k$, the canonical extension is certainly given by

$$p : \mathbb{C}^{n\times n} \to \mathbb{C}^{n\times n}, \quad p(A) = \sum_{k=0}^{d} a_k A^k.$$

If $f(z)$ can be expressed by a power series, $f(z) = \sum_{k=0}^{\infty} a_k z^k$, a natural next step is to put

$$f(A) = \sum_{k=0}^{\infty} a_k A^k, \tag{1.1}$$

but for (1.1) to make sense we must now discuss convergence issues. The main result is given in the following theorem, the proof of which gives us valuable further information on matrix functions. Recall that the spectrum $\operatorname{spec}(A)$ is the set of all eigenvalues of A.

Theorem 1. *Assume that the power series $f(z) = \sum_{k=0}^{\infty} a_k z^k$ is convergent for $|z| < \rho$ with $\rho > 0$ and assume that* $\operatorname{spec}(A) \subset \{z \in \mathbb{C} : |z| < \rho\}$. *Then the series (1.1) converges.*

Proof. Let T be the transformation matrix occuring in the Jordan decomposition

$$A = TJT^{-1}, \tag{1.2}$$

with

$$J = \begin{bmatrix} J_{m_1}(\lambda_1) & & 0 \\ & \ddots & \\ 0 & & J_{m_\ell}(\lambda_\ell) \end{bmatrix} =: \operatorname{diag}(\, J_{m_1}(\lambda_1), \ldots, J_{m_\ell}(\lambda_\ell)\,).$$

Here, $\lambda_1, \ldots, \lambda_\ell$ are the (not necessarily distinct) eigenvalues of A and m_j is the size of the jth Jordan block associated with λ_j, i.e.

$$J_{m_j}(\lambda_j) = \begin{pmatrix} \lambda_j & 1 & 0 & \cdots & 0 \\ 0 & \lambda_j & 1 & \ddots & \vdots \\ \vdots & \ddots & \ddots & \ddots & 0 \\ \vdots & & \ddots & \ddots & 1 \\ 0 & \cdots & \cdots & 0 & \lambda_j \end{pmatrix} =: \lambda_j I + S_{m_j} \in \mathbb{C}^{m_j \times m_j},$$

and $\sum_{j=1}^{\ell} m_j = n$. For each λ_j, the powers of $J_{m_j}(\lambda_j)$ are given by

$$J_m(\lambda_j)^k = \sum_{\nu=0}^{k} \binom{k}{\nu} \lambda_j^{k-\nu} \cdot S_{m_j}^{\nu}.$$

Note that $S^{\nu}_{m_j}$ has zero entries everywhere except for the ν-th upper diagonal, whose entries are equal to 1. In particular, $S^{\nu}_{m_j} = 0$ for $\nu \geq m_j$. Therefore,

$$f(J_{m_j}(\lambda_j)) = \sum_{k=0}^{\infty} a_k \sum_{\nu=0}^{k} \binom{k}{\nu} \lambda_j^{k-\nu} \cdot S^{\nu}_{m_j},$$

and for ν and j fixed we have

$$\sum_{k=0}^{\infty} a_k \binom{k}{\nu} \lambda_j^{k-\nu} = \sum_{k=0}^{\infty} \frac{1}{\nu!} \cdot a_k \cdot (k \cdot \ldots \cdot (k-\nu+1)) \lambda_j^{k-\nu} = \frac{1}{\nu!} f^{(\nu)}(\lambda_j).$$

Note that the last equality holds in the sense of absolute convergence because λ_j lies within the convergence disk of the series. This shows that the series $f(J_{m_j}(\lambda_j))$ converges. Plugging these expressions into the series from (1.1) we obtain the value of the original (now convergent) series,

$$\begin{aligned} f(A) &= T\text{diag}\left(f(J_{m_1}(\lambda_1)), \ldots, f(J_{m_\ell}(\lambda_\ell))\right)) T^{-1} \\ &= T\text{diag}\left(\sum_{\nu=0}^{m_1-1} \frac{1}{\nu!} f^{(\nu)}(\lambda_1) \cdot S^{\nu}_{m_1}, \ldots, \sum_{\nu=0}^{m_\ell-1} \frac{1}{\nu!} f^{(\nu)}(\lambda_\ell) \cdot S^{\nu}_{m_\ell} \right) T^{-1} \quad (1.3) \end{aligned}$$

□

It may happen that a function f cannot be expressed by a series converging in a large enough disk. If f is sufficiently often differentiable at the eigenvalues of A, then the right-hand side of (1.3) is still defined. We make it the basis of our final definition of a matrix function.

Definition 1. *Let $A \in \mathbb{C}^{n \times n}$ be a matrix with $\text{spec}(A) = \{\lambda_1, \ldots, \lambda_\ell\}$ and Jordan normal form*

$$J = T^{-1}AT = \text{diag}(\, J_{m_1}(\lambda_1), \ldots, J_{m_\ell}(\lambda_\ell)\,).$$

Assume that the function $f : \mathbb{C} \to \mathbb{C}$ is $m_j - 1$ times differentiable at λ_j for $j = 1, \ldots, \ell$. Then the matrix function $f(A)$ is defined as $f(A) = Tf(J)T^{-1}$ where

$$f(J) = \text{diag}(f(J_{m_1}(\lambda_1)), \ldots, f(J_{m_\ell}(\lambda_\ell))) \quad \text{with } f(J_{m_j}(\lambda_j)) = \sum_{\nu=0}^{m_j-1} \frac{1}{\nu!} f^{(\nu)}(\lambda_j) \cdot S^{\nu}_{m_j}.$$

This definition makes explicit use of the Jordan canonical form and of the associated transformation matrix T. Neither T nor J are unique, but it can be shown – as is already motivated by (1.1) – that $f(A)$ as introduced in Definition 1 does not depend on the particular choice of T or J.

As a first consequence of Definition 1 we note the following important property.

Proposition 1. *With the notation above, it holds $f(A) = p(A)$, where p is the polynomial of degree not greater than $n-1$ which interpolates the eigenvalues λ_j of A in the Hermite sense (i.e. $f^{(\nu)}(\lambda_j) = p^{(\nu)}(\lambda_j)$ for all relevant ν's and j's).*

The polynomial p in Proposition 1 will not only depend on f, but also on A or, more precisely, on the minimal polynomial of A (of which the multiplicity of an eigenvalue λ determines the maximal block size m_j for the Jordan blocks corresponding to this eigenvalue). When A is normal, T is an orthogonal matrix and all Jordan blocks have size one, i.e. we have

$$J = \mathrm{diag}(\lambda_1, \ldots, \lambda_n).$$

So, in this particular case, we do not need any differentiability assumption on f.

A further representation of $f(A)$ can be derived in the case when f is analytic in a simply connected region Ω containing $\mathrm{spec}(A)$. Let γ be a curve in Ω with winding number +1 w.r.t. a point $z \in \Omega$. The Residue Theorem tells us

$$\frac{f^{(\nu)}(z)}{\nu!} = \frac{1}{2\pi i} \oint_\gamma \frac{f(t)}{(t-z)^{\nu+1}} dt. \tag{1.4}$$

Let $J_{m_j}(\lambda_j)$ be a Jordan block associated with λ_j and let $z \neq \lambda_j$. Then

$$(zI - J_{m_j})^{-1} = ((z-\lambda_j)I - S_{m_j})^{-1} = \frac{1}{z-\lambda_j} \cdot \sum_{\nu=0}^{m_j-1} \left(\frac{1}{z-\lambda_j} \cdot S_{m_j} \right)^\nu,$$

from which we get

$$\begin{aligned} \frac{1}{2\pi i} \oint_\gamma f(z)(zI - J_{m_j})^{-1} dz &= \sum_{\nu=0}^{m_j-1} \frac{1}{2\pi i} \oint_\gamma \frac{f(z)}{(z-\lambda_j)^{\nu+1}} S_{m_j}^\nu dz \\ &= \sum_{\nu=0}^{m_j-1} \frac{f^{(\nu)}(\lambda_j)}{\nu!} \cdot S_{m_j}^\nu, \end{aligned}$$

the second line holding due to (1.4). Using this for each Jordan block in Definition 1 and recombining terms we obtain the following integral representation of $f(A)$,

$$f(A) = \frac{1}{2\pi i} \oint_\gamma f(t)(tI - A)^{-1} dt. \tag{1.5}$$

1.3 Computational aspects

It is not necessarily a good idea to stick to one of the definitions of matrix function given in the previous section when it comes to numerically compute a matrix function $f(A)$. In this section we will discuss such computational issues, describing several numerical approaches having their advantages in different situations, basically depending on spectral properties of A, on the dimension and sparsity of A and on whether we really want to obtain the matrix $f(A)$ rather than "just" its action $f(A)v$ on a vector v.

1.3.1 Normal matrices

A matrix $A \in \mathbb{C}^{n\times n}$ is said to be *normal* if it commutes with its adjoint, $AA^{\mathtt{H}} = A^{\mathtt{H}}A$. Normal matrices may also be characterized as being unitarily diagonalizable, i.e. we have the representation

$$A = Q\Lambda Q^{\mathtt{H}} \quad \text{with } Q^{-1} = Q^{\mathtt{H}},\ \Lambda = \mathrm{diag}(\lambda_1, \ldots, \lambda_n),\ \mathrm{spec}(A) = \{\lambda_1, \ldots, \lambda_n\}.$$

This representation is also the Jordan decomposition of A from (1.2), so that

$$f(A) = Qf(\Lambda)Q^{\mathtt{H}}, \qquad f(\Lambda) = \mathrm{diag}(f(\lambda_1), \ldots, f(\lambda_n)). \tag{1.6}$$

Normal matrices have the very attractive property that their eigenvalues λ_i and the corresponding invariant subspaces are well conditioned (see [16], for example), i.e. small changes in A yield only small changes in Λ and Q. Therefore, if we use a numerically (backward) stable algorithm to compute Λ and Q, like, for example, the standard Householder reduction to upper Hessenberg form followed by the QR-iteration, we may safely use the so computed Λ and Q to finally compute $f(A)$ via (1.6). The computational cost of this approach is $\mathcal{O}(n^3)$ due to the various matrix-matrix multiplications and to the cost for computing the eigendecomposition.

If A is not normal, its eigenvalues are not necessarily well conditioned, the condition number being related to $\|T\|_2 \cdot \|T^{-1}\|_2$ with T from the Jordan decomposition (1.2). It is also important to realize that the size of the Jordan blocks may widely vary under infinitesimal perturbations in A. Therefore, if A is not normal, Definition 1 does not provide a numerically stable means for computing $f(A)$.

1.3.2 Quadrature rules

Assume that f is analytic in Ω and that γ and Ω are as in (1.5) so that we have

$$f(A) = \frac{1}{2\pi i}\oint_\gamma f(t)(tI - A)^{-1}dt.$$

We apply a quadrature rule with m nodes $t_j \in \gamma$ and weights ω_j to the right-hand side to get

$$\frac{1}{2\pi i}\oint_\gamma \frac{f(t)}{t-z}dt = \sum_{j=1}^{m}\omega_j\frac{f(t_j)}{t_j - z} + r.$$

This shows that we can approximate

$$f(A) \approx \sum_{j=1}^{m}\omega_j f(t_j)\cdot(t_jI - A)^{-1}. \tag{1.7}$$

For such quadrature rules, the approximation error r can be expressed or bounded using higher derivatives of f. Actually, since we integrate over a closed curve, taking the right nodes the quadrature error is usually much smaller than what one would expect from quadrature formulas over finite (real) intervals, and the accuracy often increases exponentially with the number of nodes, see [14], [15]. In principle, this can then be used to obtain bounds on the approximation error in (1.7), but to do so we usually need some knowledge about the norms of T and T^{-1} in (1.2), as well as on the size of the eigenvalues of A. See also section 1.3.6.

For specific functions, other integral representations may be used. For example, for $z \in \mathbb{C}$, z not on the non-positive real line, we have (see [14])

$$\log(z) = \int_0^1 (z-1)[t(z-1)+1]^{-1} dt,$$

so that using a quadrature rule for the interval $[0,1]$, we can use the approximation

$$\log(A) \approx \sum_{j=1}^{m} \omega_j \cdot (A-I)[t_j(A-I)+I]^{-1}.$$

As another example, for $z > 0$ we can write

$$z^{-1/2} = \frac{2}{\pi} \cdot \int_0^\infty \frac{1}{t^2+z}\, dt,$$

and use a quadrature rule on $[0,\infty]$ to approximate $A^{-1/2}$ when $\mathrm{spec}(A) \subset (0,\infty]$.

Similar approaches have been proposed for various other functions like the p-th root or the sign function, see [6], [58], for example.

Within this quadrature framework, the major computational cost will usually be due to the inversion of several matrices. As is explained in [14], this cost can often be reduced if we first compute a unitary reduction to upper Hessenberg form (which can be done in a numerically stable manner using Householder transformations), i.e.

$$A = QHQ^{\mathtt{H}}, \;\; Q \text{ unitary }, \;\; H \text{ zero below the first subdiagonal.}$$

Then, for example,

$$(t_jI - A)^{-1} = Q \cdot (t_jI - H)^{-1} \cdot Q^{\mathtt{H}} \text{ for all } j,$$

with the inversion of the matrix $t_jI - H$ having cost $\mathcal{O}(n^2)$ rather than $\mathcal{O}(n^3)$.

1.3.3 Matrix iterations

Sometimes, it is convenient to regard $f(z)$ as the solution of a fixed point equation $g_z(f) = f$ with g_z being contractive in a neighbourhood of the fixed point $f(z)$. The method of successive approximations

$$f_{k+1} = g_z(f_k) \tag{1.8}$$

can then be turned into a corresponding matrix iteration

$$F_{k+1} = g_A(F_k). \tag{1.9}$$

Approaches of this kind have, for example, been proposed for the matrix square root [31], [32], where Newton's method

$$f_{k+1} = \frac{1}{2} \cdot \left(f_k + \frac{z}{f_k} \right) \tag{1.10}$$

to compute $\sqrt{z}$ results in the iteration

$$F_{k+1} = \frac{1}{2} \cdot \left(F_k + A \cdot F_k^{-1} \right). \tag{1.11}$$

Similar other iterations, not always necessarily derived from Newton's method, have been proposed for the matrix p-th root [6] or for the matrix sign function [41]. A major catch with these approaches is that numerical stability of the matrix iteration (1.9) is not always guaranteed, even when the scalar iteration (1.8) is perfectly stable. Then, some quite subtle modifications, like e.g. the coupled two-term iteration for the square root analyzed in [31] must be used in order to achieve numerical stability. The iteration (1.9) is usually also quite costly. For example, (1.11) requires the inversion of F_k at every step, so that each step has complexity $\mathcal{O}(n^3)$. Therefore, for these methods to be efficient, convergence should be fast, at least superlinear.

1.3.4 Rational approximations

Polynomial approximations for a function f often require a quite high degree of the approximating polynomial in order to achieve a reasonable quality of approximation. *Rational* approximations typically obtain the same quality with substantially fewer degrees of freedom.

Assume that we have the rational approximation

$$f(z) \approx \frac{\mathcal{N}_{\mu\nu}(z)}{\mathcal{D}_{\mu\nu}(z)},$$

where $\mathcal{N}_{\mu\nu}, \mathcal{D}_{\mu\nu}$ are polynomials of degree μ and ν, respectively. (The use of the two indices μ and ν in both polynomials may appear abusive at this point, but it will be very convenient when discussing Padé approximations to the exponential in section 1.4.2). Then

$$f(A) \approx \mathcal{N}_{\mu\nu}(A) \cdot \left(\mathcal{D}_{\mu\nu}(A)\right)^{-1}.$$

Assume that A is diagonalizable. If we know

$$\left| f(z) - \frac{\mathcal{N}_{\mu\nu}(z)}{\mathcal{D}_{\mu\nu}(z)} \right| \leq \epsilon \text{ for } z \in \operatorname{spec}(A),$$

for some $\epsilon > 0$, we get

$$\| f(A) - \mathcal{N}_{\mu\nu}(A) \cdot \left(\mathcal{D}_{\mu\nu}(A)\right)^{-1} \|_2 \leq \epsilon \cdot \|T\|_2 \cdot \|T^{-1}\|_2$$

which further simplifies when A is normal, since then T is unitary so that $\|T\|_2 \cdot \|T^{-1}\|_2 = 1$. Rational functions can be expressed as partial fraction expansions. Simplifying our discussion to the case of single poles, this means that we can expand

$$\frac{\mathcal{N}_{\mu\nu}(z)}{\mathcal{D}_{\mu\nu}(z)} = p(z) + \sum_{j=1}^{\nu} \frac{\omega_j}{z - \tau_j},$$

with $p(z)$ being a polynomial of degree $\mu - \nu$ if $\mu \geq \nu$ and $p \equiv 0$ if $\mu < \nu$. This representation is particularly useful if we are interested only in $f(A)v$ for some vector v, as we will discuss later in section 1.3.6. Note also that the quadrature rules from (1.7) immediately give a partial fraction expansion, so that the two approaches are very closely related. For a recent investigation, see [66].

1.3.5 Krylov subspace approaches

When A has large dimension, the action of $f(A)$ on a vector v, namely $f(A)v$, may be effectively approximated by projecting the problem onto a subspace of possibly much smaller dimension. The Krylov subspace

$$K_k(A, v) = \mathrm{span}\{v, Av, \ldots, A^{k-1}v\}$$

has been extensively used to this purpose, due to its favourable computational and approximation properties, see, e.g., van der Vorst [68], [69] for a discussion for general f. Let V_k be a full column rank $n \times k$ matrix whose columns span $K_k(A, v)$, and assume the following Arnoldi type recurrence holds for V_k,

$$AV_k = V_{k+1}H_{k+1,k} = V_kH_k + h_{k+1,k}v_{k+1}e_k^T. \tag{1.12}$$

An approximation to $x = f(A)v$ may be obtained as

$$x_k = V_kf(H_k)e_1\|v\|. \tag{1.13}$$

The procedure amounts to projecting the matrix onto the much smaller subspace $K_k(A, v)$, by means of the representation matrix H_k and $v = V_ke_1\|v\|$. If V_k has orthonormal columns then $H_k = V_k^{\mathtt{H}}AV_k$. If in addition A is Hermitian, the iteration (1.12) reduces to the Lanczos three-term recurrence, in which case H_k is tridiagonal and Hermitian.

The functional evaluation is carried out within this reduced space, and the obtained solution is expanded back to the original large space. Assume now that $k = n$ iterations can be carried out, so that the square matrix V_n is orthogonal. Then (1.12) gives $AV_n = V_nH_n$ and thus $A = V_nH_nV_n^{\mathtt{H}}$. Using this relation, for $k < n$, the approximation in $K_k(A, v)$ may be viewed as a problem order reduction to the first k columns of V_n and corresponding portion of H_n as

$$x = f(A)v = V_nf(H_n)V_n^{\mathtt{H}}v \approx V_kf(H_k)V_k^{\mathtt{H}}v.$$

For k small compared to n, the quality of the approximation strongly depends on the spectral properties of A and on the capability of $K_k(A,v)$ to capture them. A first characterization in this sense is given by the following result, which can be deduced from Proposition 1 applied to the matrix H_k and the fact that $p(A)v = V_k p(H_k)v$ for all polyomials of degree less than or equal to $k-1$; see [61, Proposition 6.3]. This is a generalization of [60, Theorem 3.3].

Proposition 2. *Let the columns of V_k, with $V_k^{\mathrm{H}} V_k = I_k$ span $K_k(A,v)$ and let $H_k = V_k^{\mathrm{H}} A V_k$. Then, the approximation $V_k f(H_k) e_1 \|v\|$ represents a polynomial approximation $p(A)v$ to $f(A)v$, in which the polynomial p of degree $k-1$ interpolates the function f in the Hermite sense on the set of eigenvalues of H_k.*

Other polynomial approximations have been explored, see, e.g., [18]; approaches that interpolate over different sets have been proposed for the exponential function [53]. Note that the projection nature of the approach allows one to derive estimates for $\|f(A)\|$ as $\|f(A)\| \approx \|f(H_k)\|$ which may be accurate even for small k when A is Hermitian.

All these results assume exact precision arithmetic. We refer to [17] for an analysis of finite precision computation of matrix functions with Krylov subspace methods when A is Hermitian.

It should be mentioned that the projection onto a Krylov subspace does not require A to be stored explicitly, but it only necessitates a function that given v, returns the action of A, namely $y = Av$. This operational feature is of paramount importance in applications where, for instance, A is the (dense) product or other combination of sparse matrices, so that the operation $y = Av$ may be carried out by a careful application of the given matrix combination.

Another practical aspect concerns the situation where k, the dimension of the Krylov subspace, becomes large. Computing $f(H_k)$ with one of the methods presented in the previous sections can then become non-negligible. Moreover, we may run into memory problems, since approximating $f(A)v$ via (1.13) requires the whole matrix V_k to be stored. This is needed even when, for istance, A is Hermitian, in which case (1.12) is the Lanczos recurrence and H_k is tridiagonal. In such a situation, however, we can resort to a “two–pass” procedure which crucially reduces the amount of memory needed: In the first pass, we run the short-term recurrence Lanczos process. Here, older columns from V_k can be discarded, yet the whole (tridiagonal) matrix H_k can be built column by column. Once $f(H_k)$ has been generated, we compute $y_k = f(H_k)e_1 \cdot \|v\|$. Then we run the short-term recurrence Lanczos process once again to recompute the columns of V_k and use them one at a time to sum up $V_k f(H_k) e_1 = V_k y_k$. Of course, this two-stage approach essentially doubles the computational work for generating the Lanczos basis.

For a general matrix A the Arnoldi process cannot be turned into a short-term recurrence, so one must search for alternatives in the case that k gets too large. Recently, Eiermann and Ernst [20] have developed an interesting scheme that allows one to *restart* Krylov subspace methods for computing $f(A)v$, in the same flavour as with linear system solvers; in fact, the two approaches are tightly related; see [47]. Having computed a not yet sufficiently good approximation x_k via (1.13), the idea is to start again a Krylov subspace approximation based on the error $x_k - f(A)v$ which is expressed as a *new* matrix function of A. The algorithmic formulation is non-trivial, particularly since special care has to be taken with regard to numerical stability, see [20].

Other alternatives include acceleration procedures, that aim at improving the convergence rate of the approximation as the Krylov subspace dimension increases. Promising approaches have been recently proposed in the Hermitian case by Druskin and Knizhnerman [19], by Moret and Novati [52] and by Hochbruck and van den Eshof [37].

1.3.6 Krylov subspaces and rational approximations

As a last contribution to this section, let us turn back to rational approximations for f which we assume to be given in the form of a partial fraction expansion (no multiple poles for simplicity)

$$f(z) \approx p(z) + \sum_{j=1}^{\nu} \frac{\omega_j}{z - \tau_j}.$$

Then $f(A)v$ can be approximated as

$$f(A)v \approx p(A)v + \sum_{j=1}^{\nu} \omega_j (A - \tau_j I)^{-1} v. \tag{1.14}$$

Since evaluating $p(A)v$ is straightforward, let us assume $p \equiv 0$ in the sequel.

The computation of $(A - \tau_j I)^{-1} v$ means that we have to solve a linear system for each j, where all linear systems have the same right-hand side, while the coefficient matrix only differs for the shift. In general, shifts may be complex even for real and symmetric A, although they appear in conjugate pairs. Interestingly, the particular “shifted” structure of these systems can be exploited in practical computation. If we solve each system iteratively using a Krylov subspace method with initial zero guess for all j, the kth iterate for each system lies in $K_k(A - \tau_j I, v)$ which is identical to $K_k(A, v)$. The fact that Krylov subspaces are invariant with respect to shifts can now be exploited in various Krylov subspace solvers like CG, BiCG, FOM and QMR (and also with modifications in BiCGStab and restarted GMRES) to yield very efficient procedures which require only *one* matrix-vector multiplication with A, and possibly with $A^{\mathtt{H}}$, in order to update the iterates for *all* m systems simultaneously; see [63] for a survey of these methods for shifted systems and also [21], [22], [23], [24]. Denote by $x_k^{(j)}$ the iterate of the Krylov solver at step k for system j. Then the linear combination

$$x_k = \sum_{j=1}^{\nu} \omega_j x_k^{(j)} \in K_k(A, v) \tag{1.15}$$

is an approximation to $f(A)v$. In fact, it is an approximation to the action of the rational function approximating $f(A)$. Therefore, what we obtained in (1.15) is an approximation to $f(A)v$ in $K_k(A,v)$, which is different from (1.13) presented before. A special case is when f is itself a rational function. In such a situation, the two approaches may coincide if, for instance, a Galerkin method is used to obtain the approximate solutions $x_k^{(j)}$. Indeed, for $f = \mathcal{R}_{\mu\nu} = \mathcal{N}_{\mu\nu}/\mathcal{D}_{\mu\nu}$,

$$\begin{aligned} f(A)v = \mathcal{N}_{\mu\nu}(A)(\mathcal{D}_{\mu\nu}(A))^{-1}v &= \sum_{j=1}^{\nu} \omega_j (A-\tau_j I)^{-1}v \qquad (1.16) \\ &\approx \sum_{j=1}^{\nu} \omega_j V_k (H_k - \tau_j I)^{-1} e_1 \|v\| = V_k f(H_k) e_1 \|v\|. \end{aligned}$$

The approach outlined above has several attractive features for a general function f. Firstly, if we have a bound for the error between $x_k^{(j)}$ and the solution $(A-\tau_j)^{-1}v$ for each j, we can combine these bounds with the approximation error of the rational approximation to get an overall a posteriori bound for $\|f(A)v - x^{(k)}\|$. Sometimes, such bounds might be obtained quite easily. For example, if A is Hermitian and positive definite and all shifts τ_j are real and negative, the norm of the inverse $(A - \tau_j I)^{-1}$ is bounded by $1/|\tau_j|$. Since the residuals $r_k^{(j)} = (A-\tau_j I)x_k^{(j)} - v$ are usually available in the Krylov solver in use, we can use the bound

$$\|x_k^{(j)} - (A-\tau_j I)^{-1}v\|_2 \le \frac{1}{|\tau_j|}\|r_k^{(j)}\|_2.$$

Similar bounds that require estimates of the spectrum of A may be obtained also for complex poles τ_j, see [47].

Secondly, in the Hermitian case, the memory requirements of this approach only depend on m, the number of poles in the rational approximation, but not on k, the dimension of the Krylov subspace. Indeed, the symmetry of the problem can be exploited to devise a short-term recurrence which dynamically updates the solution x_k without storing the whole Krylov subspace basis. So even if k has to be sensibly large in order to get a good approximation, we will not run into memory problems. This is in contrast to the approach from section 1.3.5, although the two approaches are strictly related. Indeed, using x_k in (1.15), by the triangle inequality we have

$$\begin{aligned} \left| \|f(A)v - x_k\| - \|f(A)v - V_k f(H_k)e_1\|v\| \| \right| &\le \|V_k f(H_k)e_1\|v\| - x_k\| \\ &= \| \left(f(H_k) - \mathcal{R}_{\mu\nu}(H_k)\right) e_1\| \|v\| . \end{aligned}$$

Therefore, whenever the chosen rational function $\mathcal{R}_{\mu\nu}$ accurately approximates f, the two approaches evolve similarly as the Krylov subspace dimension increases.

1.4 The exponential function

We next focus our attention on methods specifically designed to approximate the matrix exponential, $\exp(A)$, and its action on a vector v. We start by briefly discussing the role of this function within Model Order Reduction applications. Depending on the setting, we shall use either of the two equivalent notations $\exp(A)$ and e^A. We explicitly observe that Definition 1 ensures that $\exp(A)$ is nonsingular for any matrix A.

1.4.1 The exponential matrix in model order reduction applications

In this section we briefly review some application problems whose numerical solution benefits from the approximate computation of the exponential.

Numerical solution of time-dependent differential equations. The numerical solution of ordinary and time-dependent partial differential equations (ODEs and PDEs, respectively) may involve methods that effectively employ the matrix exponential. Recent developments in the efficient approximation of $\exp(A)v$ have increased the use of numerical "exponential-based" (or just "exponential") techniques that allow one to take larger time steps. More precisely, consider the system of ODEs of the form

$$u'(t) = Au(t) + b(t), \qquad u(0) = u_0,$$

where A is a negative semidefinite matrix. The analytic solution is given by

$$u(t) = e^{tA}u_0 + \int_0^t e^{(\tau - t)A} b(\tau) d\tau.$$

Whenever a good approximation to the propagation operator e^{sA} is available, it is possible to approximate the analytic solution by simply approximating the integral above with convenient quadrature formulas, leading to stable solution approximations. The generalization of this approach to the numerical solution of partial differential equations can be obtained, for instance, by employing a semidiscretization (in space) of the given problem. Consider the following self-adjoint parabolic equation

$$\frac{\partial u(x,t)}{\partial t} = \mathrm{div}(a(x)\nabla u(x,t)) - b(x)u(x,t) + c(x),$$

with $x \in \Omega$, Dirichlet boundary conditions and $b(x) \geq 0$, $a(x) > 0$ in Ω, with a, b, c sufficiently regular functions. A continuous time – discrete space discretization leads to the ordinary differential equation

$$E\frac{d\mathbf{u}(t)}{dt} = -A\mathbf{u}(t) + \mathbf{c}, \quad t \geq 0,$$

where A, E are positive definite Hermitian matrices, so that the procedure discussed above can be applied; see, e.g., [11], [25], [51], [65], [70]. Further attempts to generalize this procedure to non-selfadjoint PDEs can be found in [25, section 6.2], although the theory behind the numerical behavior of the ODE solver in this case is not completely understood yet.

The use of exponential integrators is particularly effective in the case of certain stiff systems of nonlinear equations. Consider, e.g., the initial value problem

$$\frac{du(t)}{dt} = f(u), \qquad u(t_0) = u_0.$$

If the problem is stiff, standard integrators perform very poorly. A simple example of an exponential method for this system is the *exponentially fitted Euler* scheme, given by

$$u_1 = u_0 + h\phi(hA)f(u_0),$$

where h is the step size, $\phi(z) = \frac{e^z-1}{z}$, and $A = f'(u_0)$. The recurrence $\{u_k\}_{k=0,1,\dots}$ requires the evaluation of $\phi(hA)v$ at each iteration, for some vector v; see, e.g., [36].

An application that has witnessed a dramatic increase in the use of the matrix exponential is Geometric Integration. This research area includes the derivation of numerical methods for differential equations whose solutions are constrained to belong to certain manifolds equipped with a group structure. One such example is given by linear Hamiltonian problems of the form

$$\begin{cases} \dot{Y}(t) = \mathcal{J}A(t)Y(t), \\ Y(t_0) = Y_0, \end{cases}$$

where $\mathcal{J}$ is the matrix $[0, I; -I, 0]$, A is a continuous, bounded, symmetric matrix function, and $Y_0 \in \mathbb{R}^{N\times p}$ is symplectic, that is it satisfies $Y_0^{\mathrm{H}}\mathcal{J}Y_0 = \mathcal{J}$. The solution $Y(t)$ is symplectic for any $t \geq t_0$. Using the fact that $\mathcal{J}A$ is Hamiltonian, it can be shown that $\exp(\mathcal{J}A(t))$ is symplectic as well. Numerical methods that aim at approximating $Y(t)$ should also preserve its symplecticity property. This is achieved for instance by the numerical scheme $Y_{k+1} = \exp(h\mathcal{J}A(t_k))Y_k$, $t_{k+1} = t_k + h$, $k = 0, 1, \dots$. Structure preserving methods associated with small dimensional problems have received considerable attention, see, e.g., [10], [29], [39], [71] and references therein. For large problems where order reduction is mandatory, approximations obtained by specific variants of Krylov subspace methods can be shown to maintain these geometric properties; see, e.g., [48].

Analysis of dynamical systems. The exponential operator has a significant role in the analysis of linear time-invariant systems of the form

$$\begin{cases} \dot{x}(t) = Ax(t) + Bu(t) \\ y(t) = Cx(t) \end{cases} \tag{1.17}$$

where A, B and C are real matrices of size $n \times n$, $n \times m$ and $p \times n$, respectively. In the following we assume that A is stable, that is its eigenvalues are in the left half plane $\mathbb{C}^-$, and that the system is controllable and observable; see, e.g., [1].

The matrix of the states of the system for impulsive inputs is $x(t) = e^{tA}B$, whereas in general, for an initial state x_0 at time t_0, the resulting state at time $t \geq t_0$ is given by

$$x(t) = e^{(t-t_0)A}x_0 + \int_{t_0}^{t} e^{(t-\tau)A}Bu(\tau)d\tau.$$

Therefore, an approximation to the state involves the approximation of the matrix exponential. Moreover, the state function is used to define the first of the following two matrices which are called the controllability and the observability Gramians, respectively,

$$P = \int_0^\infty e^{tA} B B^{\mathrm{H}} e^{tA^{\mathrm{H}}} dt, \quad Q = \int_0^\infty e^{tA^{\mathrm{H}}} C^{\mathrm{H}} C e^{tA} dt. \tag{1.18}$$

The following result shows that these are solutions to Lyapunov equations.

Theorem 2. *Given the linear time-invariant system (1.17), let P, Q be as defined in (1.18). Then they satisfy*

$$AP + PA^{\mathrm{H}} + BB^{\mathrm{H}} = 0, \qquad A^{\mathrm{H}}Q + QA + C^{\mathrm{H}}C = 0.$$

Proof. The proof follows from substituting the definition of P and Q into the corresponding expressions $AP+PA^{\mathrm{H}}$, $A^{\mathrm{H}}Q+QA$. By using the fact that $e^{tA}A = \frac{d}{dt}(e^{tA})$ and integrating, we obtain, e.g., for Q,

$$\begin{aligned} QA + A^{\mathrm{H}}Q &= \int_0^\infty \left(e^{tA^{\mathrm{H}}} C^{\mathrm{H}} C e^{tA} A + A^{\mathrm{H}} e^{tA^{\mathrm{H}}} C^{\mathrm{H}} C e^{tA} \right) dt \\ &= \int_0^\infty \left(e^{tA^{\mathrm{H}}} C^{\mathrm{H}} C \frac{de^{tA}}{dt} + \frac{de^{tA^{\mathrm{H}}}}{dt} C^{\mathrm{H}} C e^{tA} \right) dt \\ &= \int_0^\infty \frac{d(e^{tA^{\mathrm{H}}} C^{\mathrm{H}} C e^{tA})}{dt} dt = \lim_{\tau \to \infty} (e^{tA^{\mathrm{H}}} C^{\mathrm{H}} C e^{tA}) \Big|_0^\tau = -C^{\mathrm{H}}C. \quad \square \end{aligned}$$

It can also be shown that the solution to each Lyapunov equation is unique. In a more general setting, the matrix $M := -(A^{\mathrm{H}}Q + QA)$ is not commonly given in factored form. In this case, if it can be shown that M is positive semidefinite and that the pair (A, M) is observable, then Q is positive definite (a corresponding result holds for P); see, e.g., [4], [1], [13].

The Lyapunov equation may be used to compute estimates for $\|e^{tA}\|$, which in turn provides information on the stability of the original system in the case of $C^{\mathrm{H}}C$ full rank; see, e.g., [13, Th. 3.2.2] for a proof.

Theorem 3. *Let A be stable and $C^{\mathrm{H}}C$ full rank. Then the unique solution Q to the Lyapunov equation $A^{\mathrm{H}}Q + QA + C^{\mathrm{H}}C = 0$ satisfies*

$$\|e^{tA}\| \leq \left(\frac{\lambda_{\max}(Q)}{\lambda_{\min}(Q)} \right)^{\frac{1}{2}} e^{-\alpha t},$$

where $\alpha = \lambda_{\min}(Q^{-1}C^{\mathrm{H}}C)/2 > 0$.

For large problems, other devices can be used to directly approximate $\|e^{tA}\|$ without first resorting to the solution of a Lyapunov equation; cf. section 1.3.5. We also refer to [46] for a general discussion on the norm $\|e^{tA}\|$ and some of its bounds.

1.4.2 Computing the exponential of a matrix

Over the years, several methods have been devised and tested for the computation of the matrix exponential; we refer to [50] for a recent survey of several approaches and for a more complete bibliographic account. The algorithmic characteristics may be very different depending on whether the matrix has small or large dimension, or whether it is dense or sparse; the structural and symmetry properties also play a crucial role; see, e.g., the discussion in [62]. In this section we discuss the case of small matrices. When A is normal, the spectral decomposition discussed in section 1.3.1 can be employed, namely $A = TJT^{\mathtt{H}}$ with T unitary. This gives $\exp(A) = T\exp(J)T^{\mathtt{H}}$, once the decomposition of A is computed.

In the non-normal case, one method has emerged in the last decade, for its robustness and efficiency: Padé approximation with scaling and squaring. The basic method employs a rational function approximation to the exponential function as

$$\exp(\lambda) \approx \mathcal{R}_{\mu\nu}(\lambda) = \frac{\mathcal{N}_{\mu\nu}(\lambda)}{\mathcal{D}_{\mu\nu}(\lambda)},$$

where $\mathcal{N}_{\mu\nu}, \mathcal{D}_{\mu\nu}$ are polynomials of degree μ and ν, respectively. One attractive feature of the $[\mu/\nu]$ Padé approximation is that the coefficients of the two polynomials are explicitly known, that is

$$\mathcal{N}_{\mu\nu}(\lambda) = \sum_{j=0}^{\mu} \frac{(\mu+\nu-j)!\mu!}{(\mu+\nu)!(\mu-j)!j!}\lambda^j, \qquad \mathcal{D}_{\mu\nu}(\lambda) = \sum_{j=0}^{\nu} \frac{(\mu+\nu-j)!\nu!}{(\mu+\nu)!(\nu-j)!j!}(-\lambda)^j.$$

These two polynomials have a rich structure. For example, one has the relation $\mathcal{N}_{\mu\nu}(\lambda) = \mathcal{D}_{\nu\mu}(-\lambda)$ as well as several other important properties which can be found, e.g., in [26, section 5.2].

Diagonal Padé approximation ($\mu = \nu$), is usually preferred because computing $\mathcal{R}_{\mu\nu}$ with say, $\mu > \nu$, is not cheaper than computing the more accurate $\mathcal{R}_{\nu_*\nu_*}$ where $\nu_* = \max\{\mu, \nu\}$. Nonetheless, because of their stability properties, Padé $[\nu + 1/\nu]$ approximations are used, together with $[\nu/\nu]$ approximations, in the numerical solution of initial value problems with one-step methods. Another attractive property of the diagonal Padé approximation is that if A has eigenvalues with negative real part, then the spectral radius of $\mathcal{R}_{\nu\nu}(A)$ is less than one, for any ν. In the following, diagonal rational approximation will be denoted by $\mathcal{R}_{\nu\nu} = \mathcal{R}_{\nu}$. The accuracy of the approximation can be established by using the following result.

Theorem 4. [26, Theorem 5.5.1] *Let the previous notation hold. Then*

$$e^{\lambda} - \mathcal{R}_{\mu\nu}(\lambda) = (-1)^{\nu} \frac{\mu!\,\nu!}{(\mu+\nu)!\,(\mu+\nu+1)!}\lambda^{\mu+\nu+1} + O(\lambda^{\mu+\nu+2}).$$

This error estimate shows that the approximation degrades as λ gets away from the origin. This serious limitation motivated the introduction of the scaling and squaring procedure. By exploiting the property $e^A = (e^{A/k})^k$, for any square matrix A

and scalar k, the idea is to determine k so that the scaled matrix A/k has norm close to one, and then employ the approximation

$$e^{A/k} \approx \mathcal{R}_\nu(A/k).$$

The approximation to the original matrix e^A is thus recovered as $e^A \approx \mathcal{R}_\nu(A/k)^k$. The use of powers of two in the scaling factor is particularly appealing. Indeed, by writing $k = 2^s$, the final approximation $\mathcal{R}_\nu(A/2^s)^{2^s}$ is obtained by repeated squaring. The scalar s is determined by requiring that $\|A\|_\infty/2^s$ is bounded by some small constant, say 1/2. In fact, this constant could be allowed to be significantly larger with no loss in stability and accuracy; see [33]. The approach oulined here is used in Matlab 7.1. [49].

A rational function that is commonly used in the case of symmetric negative semidefinite matrices, is given by the Chebychev rational function. The Chebychev approximation $\mathcal{R}^\star_{\mu\nu}$ determines the best rational function approximation in $[0, +\infty)$ to $e^{-\lambda}$ by solving the problem

$$\min_{\mathcal{R}_{\mu\nu}} \max_{\lambda\in[0,+\infty)} \left| e^{-\lambda} - \mathcal{R}_{\mu\nu}(\lambda) \right|,$$

where the minimum is taken over all rational functions. In particular, the cases $\mu = 0$ and $\mu = \nu$ have been investigated in greater detail, and the coefficients of the polynomials of $\mathcal{R}^\star_\nu$ have been tabulated first by Cody, Meinardus and Varga in [12] for $\nu \leq 14$ and then in [9] for degree up to 30. Setting $\mathcal{E}_\nu = \max_{\lambda\in[0,+\infty)} \left| e^{-\lambda} - \mathcal{R}^\star_\nu(\lambda) \right|$, great efforts in the approximation theory community have been devoted to show the following elegant result on the error asymptotic behavior,

$$\lim_{\nu\to\infty} \mathcal{E}_\nu^{1/\nu} = \frac{1}{9.28903...},$$

disproving the so-called "1/9" conjecture. From the result above it follows that $\sup_{\lambda\in[0,+\infty)} \left| e^{-\lambda} - \mathcal{R}_\nu(\lambda) \right| \approx 10^{-\nu}$.

Other rational function approximations that have recently received renewed interest are given by rational functions with real poles, such as $\mathcal{R}_{\mu\nu}(\lambda) = \mathcal{N}_\mu(\lambda)/(1+h\lambda)^\nu$; see, e.g., [7], [52], [55]. An advantage of these functions is that they avoid dealing with *complex* conjugate poles.

1.4.3 Reduction methods for large matrices

In many application problems where A is large, the action of $\exp(A)v$ is required, rather than $\exp(A)$ itself, so that the methods of section 1.3.5 and of section 1.3.6 can be used. We first discuss some general convergence properties, and then show the role of the Krylov subspace approximation to $\exp(A)v$ in various circumstances. Note that time dependence can, in principle, be easily acommodated in the Krylov approximation as, for instance, $\exp(tA)v \approx V_k \exp(tH_k)e_1\|v\|$. In the following, we shall assume that A already incorporates time dependence. In particular, estimates