CONDITIONING OF THE EXPONENTIAL OF A BLOCK TRIANGULAR MATRIX.

LUCA DIECI AND ALESSANDRA PAPINI

Abstract. We propose a new measure of conditioning for the exponential of a block triangular matrix. We also show that different "condition numbers" must be used to assess the accuracy of different algorithms which implement diagonal Padé with scaling and squaring.

1. Introduction

In this work, we consider conditioning of the matrix function e^A for a 2×2 block triangular matrix $A \in \mathbb{R}^{n\times n}$:

$$A \;=\; \begin{bmatrix} A_{11} & A_{12} \\ 0 & A_{22} \end{bmatrix}, \tag{1.1}$$

where the diagonal blocks are square matrices; of course, we are interested in the case $A_{12} \neq 0$. We will use the notation $F(A)$ to denote e^A. The norm used is the 2-norm, unless otherwise stated.

It is well known that the exponential of A, e^A, is given by the series

$$e^A \;=\; \sum_{k=0}^{\infty} \frac{A^k}{k!}\,. \tag{1.2}$$

As a consequence of (1.2), if A is as in (1.1), then $F(A)$ has the same block structure as A:

$$F(A) \;=\; \begin{bmatrix} F_{11} & F_{12} \\ 0 & F_{22} \end{bmatrix}, \tag{1.3}$$

$$\text{where} \quad F_{ii} = e^{A_{ii}}\,, \;\; i = 1,2\,, \qquad \text{and} \qquad F_{12} \;=\; \int_0^1 e^{A_{11}(1-s)} A_{12} e^{A_{22}s} ds\,. \tag{1.4}$$

Given the form (1.3), a reasonable algorithm to approximate $F(A)$ should also give a block triangular approximation, call it $\hat{F}(A)$. This is the case for diagonal Padé rational functions, to which we will restrict from now on. Here, $\hat{F}(A) = R(A)$, $R(A) = P(A)Q^{-1}(A)$, and numerator and denominator are polynomials of degree s chosen so to match $2s+1$ terms in (1.2). Since Padé approximations are more accurate if the matrix A

1991 *Mathematics Subject Classification.* 65F30, 65F35, 15A24.

Key words and phrases. Conditioning, exponential of a matrix, diagonal Padé, scaling and squaring.

This work was supported in part under NSF Grant DMS-9973266 and CNR-GNIM. Performed while the first author was in visit to the Dep. Energetica, Univ. Florence. The authors are grateful to the referees for several helpful suggestions.

is close to 0, some rescaling of A is needed. The most popular scaling strategy is by progressive divisions of A by 2. The resulting method is known as scaling and squaring with diagonal Padé approximations. This is the "least dubious" of the celebrated 19 dubious ways of Moler & Van Loan, [11], and is the approach of Ward, [13]. The method is at the core of the `Matlab` built in function `expm`, which implements the following algorithm.

Algorithm 1 (`expm`)

(1) Choose k to be the smallest integer so that $A_k := A/2^k$ satisfies $\|A_k\| < 0.5$.
(2) Use the (6,6) Padé to approximate $F(A_k)$, call $\hat{F}(A_k)$ the obtained answer.
(3) Approximate $F(A)$ with $\hat{F}(A)$ by squaring k times $\hat{F}(A_k)$, i.e., $\hat{F}(A) = (\hat{F}(A_k))^{2^k}$.

In [1], we recently showed that the above technique may give unsatisfactory results when A is as in (1.1), because of overscaling, and proposed the following simple modification.

Algorithm 2

(1') Choose k to be the smallest integer for which $A_k := A/2^k$ satisfies $\|A_{ii}/2^k\| \leq 0.4$, $i = 1, 2$.
(2)-(3) As in Algorithm 1.

Quite clearly, the only difference is that with Algorithm 2 only the norms of the diagonal blocks determine the scaling of A, while with Algorithm 1 it is the norm of A to determine the scaling factor.

Example 1.1. This test problem is revealing. Take

$$A = \begin{bmatrix} \omega & x \\ 0 & \omega \end{bmatrix}, \quad e^A = e^\omega \begin{bmatrix} 1 & x \\ 0 & 1 \end{bmatrix}, \qquad \text{and fix} \qquad x = 10^6 \ .$$

The interesting case is when $|\omega| \ll x$, say $\omega = O(1)$. (On the diagonal of A we could use two different values, ω_1 and ω_2, as long as they are of $O(1)$). Using the two algorithms above, we get the following relative errors $\frac{\|\hat{F}(A)-F(A)\|}{\|F(A)\|}$ (in `Matlab`, the machine precision EPS is about 2.2×10^{-16}).

ω	Algorithm 1	Algorithm 2
2.1	1.4×10^{-10}	5.7×10^{-16}
4.1	7.6×10^{-11}	1.9×10^{-15}
6.1	1.9×10^{-10}	1.1×10^{-15}

As it turns out, and in spite of the obvious difference in performance, both algorithms give results which are better than those predicted by classical conditioning theory for $F(A)$: trying to understand the reason for this fact has been a motivation for this work and will lead us to a more refined conditioning measure for e^A when A is as in (1.1).

Computation and conditioning of $F(A)$ are well understood in case A is a normal matrix (see [3] and [9]), but not when A is not normal. Major contributions on conditioning of functions of a matrix have been made by Kenney and Laub in a series of influential papers: see [5, 6, 7], and references there. However, the case of A not normal[1] remains elusive

[1] the structure (1.1) is prototypical of the non-normal case

and gives rise to intriguing and poorly understood phenomena, such as the well known "hump" of [11]. It should be further appreciated that computation of $F(A)$ is expected to be the harder the farther A is from a normal matrix, see [2, (11.3.2)]. Accordingly, it is appropriate (but not strictly necessary for what follows) to think that in (1.1) we have $\|A_{11}\|$ and $\|A_{22}\| \ll \|A_{12}\| \approx \|A\|$. Finally, it is convenient (but, again, not necessary) to think that the spectra of A_{11} and A_{22} are close to one another. If not, it is our belief that approximation of $F(A)$ should be carried out differently than using a single Padé approximant; namely, by first separately computing F_{11} and F_{22}, and then solving the Sylvester equation

$$A_{11}F_{12} - F_{12}A_{22} = F_{11}A_{12} - A_{12}F_{22}$$

to approximate F_{12}. This approach, known as Parlett's method, has been discussed in several places, e.g., see [2], but also [4] for algorithmic aspects of this method on triangular matrices.

In the next section, we review some recent results on diagonal Padé approximations for e^A, and specialize conditioning results to the case of matrices as in (1.1). Finally, we briefly discuss the case of general (not block triangular) A.

Remark 1.2. A common occurrence of block triangular structure is when a matrix gets transformed to triangular form, for example via Schur reduction. Indeed, this is a generally advocated first step in algorithms which approximate e^A (e.g., see [7], but also [4]). On the other hand, block triangular structure arises naturally also on its own rights, as the following problems in engineering applications exemplify.

(1) In [10], a discrete optimal control problem is studied and the need arises to compute exponentials of block triangular matrices. There, Algorithm 1 is used. We observe that the block triangular structure of [10] fits the case of interest in this paper: the diagonal blocks have identical eigenvalues."

(2) In linear systems theory, block triangular structure plays a key theoretical and practical role; see [8]. E.g., it is the canonical form for controllability and reconstructibility. In these canonical cases, one has a linear differential systems with (2×2) block upper triangular structure. In the time invariant case, computation of the exponential of this matrix is required.

(3) One of the important interconnections of linear systems (see [8, pp. 43 & ss.]) is the so called *serial connection*. In this case, the resulting structure for the augmented state variables is precisely block upper triangular. Again, one ends up with differential systems with block upper triangular coefficient matrices and needs to compute the exponential of these matrices.

(4) As it is well known, Lyapunov and Sylvester differential equations play a key role in decoupling of linear systems and in studies of asymptotic stability of linear systems. The Lyapunov equation plays also a fundamental role in study of systems driven by white noise; see [8, p. 101 & ss.]. Recall that the Lyapunov equation is the matrix differential equation

$$\dot{X} = AX + XA^T + C\,, \quad X(0) = X_0\,,$$

where $C = C^T$ and $X_0 = X_0^T$. The coefficients A and C can be time dependent or time independent. In the latter case, the solution of the problem is:

$$X = YZ^{-1} \ , \quad \begin{bmatrix} \dot{Y} \\ \dot{Z} \end{bmatrix} = \begin{bmatrix} A & C \\ 0 & -A^T \end{bmatrix} \begin{bmatrix} Y \\ Z \end{bmatrix} \ , \ Y(0) = X_0 \ , \ \ Z(0) = I \ ,$$

and one needs to compute the exponential of the block triangular matrix $\begin{bmatrix} A & C \\ 0 & -A^T \end{bmatrix}$, precisely the structure under study here. Similar situation occurs for the Sylvester equation $\dot{X} = AX - XB + C$, which is associated to the triangular structure $\begin{bmatrix} A & C \\ 0 & B \end{bmatrix}$.

(5) As further illustration of the engineering relevance of block triangular structure, we point out that many of the model problems in control engineering are in block triangular structure. For example, see the *stirred tank* and the *inverted pendulum* problems in [8].

To sum up, a refined conditioning theory for block triangular matrices will serve a dual purpose: (i) to specialize conditioning results to a class of matrices which arises naturally in applications, and (ii) to understand the relative merits of algorithms which end up computing e^A with A (block) triangular (say, Algorithm 2 versus Algorithm 1, or the Fréchet phase of the Schur–Fréchet method of [7], or the Parlett's recursion phase of the algorithm in [4]).

2. Errors, Conditioning, and Condition Numbers

We begin with an observation. On the rescaled matrices, we can assume (see [1, Theorem 3.2]) that both Algorithms 1 and 2 give relative errors in $\hat{F}(A_k)$ of size EPS:

$$\frac{\left\| F(A_k) - \hat{F}(A_k) \right\|}{\|F(A_k)\|} \approx \text{EPS} \, .$$

Therefore, the difference between the two algorithms has to be found in the final squaring phase. This is correct, and to appreciate why we need to resort to [1, Theorem 3.6] (see also [11] for Algorithm 1). There, we showed that Algorithms 1 and 2 compute the exponential of a matrix $A + E$:

$$\hat{F}(A) = e^{A+E} \qquad \text{where} \qquad E = \left[\begin{smallmatrix} E_{11} & E_{12} \\ 0 & E_{22} \end{smallmatrix} \right] \ , \quad \text{and} \quad \|E\| \approx \text{EPS}\|A\| \, . \tag{2.1}$$

Moreover, if $\omega_i = \|A_{ii}\|/2^k$, $i = 1, 2$, satisfy $\omega_i \leq 0.4$, we also derived the following bounds:

$$\begin{gathered} \|E_{ii}\| \leq \gamma_i \|A_{ii}\| \ , \ i = 1, 2 \ , \ \ \|E_{12}\| \leq \gamma_3 \|A_{12}\| \, , \\ \gamma_i \approx \frac{\text{EPS}}{\omega_i} + O(\text{EPS}^2) \ , \ i = 1, 2, \ \ \gamma_3 \approx \text{EPS} + O(\text{EPS}^2) \, . \end{gathered} \tag{2.2}$$

What do (2.1)-(2.2) mean? At first glance, they look as good as one can reasonably hope for. However, if ω_1, and/or ω_2, are very small, then E_{11}, and/or E_{22}, are not $O(\text{EPS})$ perturbations of A_{11}, and/or A_{22}, and hence we are not guaranteed to have a small backward error in a block sense. This is precisely what can happen for Algorithm 1: in case in which $\|A\|$ is large compared to $\|A_{ii}\|$, $i = 1, 2$, to reduce the norm of A one may have unduly reduced the norm of the A_{ii}'s at the price of an unstable (in a backward, and block, sense) algorithm. But, should we expect that $\|E_{ii}\|$ are $O(\text{EPS})$ magnification of $\|A_{ii}\|$? After all, (2.1) shows that both Algorithms are backward stable, in the sense that $\|E\| \approx \text{EPS}\|A\|$.

Now, classical arguments tell us that, if $F(A)$ and $F(A+E)$ are exact and perturbed values, then

$$\frac{\|F(A+E)-F(A)\|}{\|F(A)\|} \leq \|F'(A)\|\frac{\|A\|}{\|F(A)\|}\;\frac{\|E\|}{\|A\|} + O(\|E\|^2)\,, \tag{2.3}$$

where $\|F'(A)\| = \max_{\|B\|=1}\|F'(A)B\|$. This leads to what has been termed *condition number of F at A*:

$$\kappa(F(A)) \;:=\; \|F'(A)\|\frac{\|A\|}{\|F(A)\|}\;,\quad \|F'(A)\| = \max_{\|B\|=1}\|F'(A)B\|\,, \tag{2.4}$$

which can also be written (see [9]) as

$$\kappa(F(A)) \;=\; \max_{\|E\|=1}\left\|\int_0^1 e^{A(1-t)}Ee^{At}dt\right\|\;\frac{\|A\|}{\|F(A)\|}\,. \tag{2.5}$$

Example 2.1. Consider again Example 1.1. A little algebra (or see [4], where explicit formulas are given) gives $\kappa(F(A)) \approx x^2 = 10^{12}$. Recalling the results from Table 1, we may consider ourselves lucky that we only lost six digits with Algorithm 1, and extremely lucky that we lost no digits with Algorithm 2. But the caveat is that the condition number (2.4) has taken into no account the extra structure of E, and has been derived by looking at perturbations everywhere around A; so doing, it has been penalized by a worse case sensitivity analysis that is of no help in assessing the goodness of the obtained answers.

We are ready to take into account the added triangular structure of A to arrive at an improved condition number for $F(A)$, which in turn will help assessing the net worth of Algorithms 1 and 2. We define the condition number following the classical approach given in [12] and [5].

Let the set S_b be given by

$$\begin{aligned} S_b(A) = \{B = \left[\begin{smallmatrix} B_{11} & B_{12} \\ 0 & B_{22}\end{smallmatrix}\right] : \|B\| = 1,\ \|B_{ii}\| \leq \frac{\max(\|A_{11}\|, \|A_{22}\|)}{\|A\|}\ ,\ i = 1, 2, \\ \text{and}\quad \|B_{12}\| \leq \frac{\|A_{12}\|}{\|A\|}\}\,. \end{aligned} \tag{2.6}$$

For the (inherent) condition number of $F(A)$, A as in (1.1), we propose the following:

$$\kappa_b(F(A)) \;=\; \lim_{\delta\to 0^+}\;\max_{B\in S_b(A)}\frac{\|F(A+\delta B)-F(A)\|}{\delta}\;\frac{\|A\|}{\|F(A)\|}\,. \tag{2.7}$$

This is readily seen to be equivalent to

$$\kappa_b(F(A)) \;=\; \max_{B\in S_b(A)}\|F'(A)B\|\,\frac{\|A\|}{\|F(A)\|}\,. \tag{2.8}$$

We also notice that, since the diagonal subproblems are decoupled, the diagonal subproblems have their own conditioning, defined in the standard way from (2.5):

$$\kappa(F(A_{ii})) = \max_{\|E_{ii}\|=1} \left\| \int_0^1 e^{A_{ii}(1-t)} E_{ii} e^{A_{ii}t} dt \right\| \frac{\|A_{ii}\|}{\|F(A_{ii})\|}\,, \quad i = 1, 2\,. \tag{2.9}$$

Next, we want to better characterize the structure of the perturbation matrices produced by Algorithms 1 and 2. This will allow us to understand in which set of perturbations we should define "condition numbers" (similarly to (2.8)) for the two algorithms, so to obtain some feedback on the relative errors obtained. Ideally, these sets should coincide with S_b.

So, for A as in (1.1), and when either Algorithm 1 or 2 is used, we would like to measure the following quantity

$$\|F'(A)B\| \frac{\|A\|}{\|F(A)\|}\,, \; B = \frac{E}{\|E\|}\,, \; E : \hat{F}(A) = F(A+E)\,. \tag{2.10}$$

Clearly, by the way B is defined, one has $B = \begin{bmatrix} B_{11} & B_{12} \\ 0 & B_{22} \end{bmatrix}$, $B_{ii} = \frac{E_{ii}}{\|E\|}$, $i = 1, 2$, and $B_{12} = \frac{E_{12}}{\|E\|}$. Moreover, from (2.2) and the fact that $\|E\| \approx \text{EPS}\|A\|$, we have the following approximate bounds:

$$\frac{\|E_{ii}\|}{\|E\|} \leq \frac{\gamma_i}{\text{EPS}} \frac{\|A_{ii}\|}{\|A\|}\ , i = 1, 2, \;\; \frac{\|E_{12}\|}{\|E\|} \leq \frac{\gamma_3}{\text{EPS}} \frac{\|A_{12}\|}{\|A\|}\ .$$

Now, consider Algorithm 1. Here, k is chosen so to obtain $\|A\| < 2^{k-1}$, and therefore $k \geq \log_2 \|A\|$. Thus, using the form of the constants γ_i from (2.2), one gets $\gamma_i \approx \text{EPS}\frac{\|A\|}{\|A_{ii}\|}$, $i = 1, 2$, whereas $\gamma_3 \approx \text{EPS}$. Therefore, for Algorithm 1, one obtains the approximate bounds (at first order in EPS)

$$\|B_{ii}\| \leq 1\ , i = 1, 2, \quad \text{and} \quad \|B_{12}\| \leq \frac{\|A_{12}\|}{\|A\|}\,. \tag{2.11}$$

In other words, the appropriate value of "condition number" to measure relative errors for Algorithm 1 is given by

$$\kappa_1(F(A)) \;=\; \lim_{\delta \to 0^+} \; \max_{B \in \hat{S}_b(A)} \frac{\|F(A+\delta B) - F(A)\|}{\delta} \; \frac{\|A\|}{\|F(A)\|}\,, \tag{2.12}$$

where

$$\hat{S}_b(A) = \{B = \begin{bmatrix} B_{11} & B_{12} \\ 0 & B_{22} \end{bmatrix} : \; \|B\| = 1, \; \|B_{ii}\| \leq 1, \, i = 1, 2, \;\; \|B_{12}\| \leq \frac{\|A_{12}\|}{\|A\|}\}\,.$$

For Algorithm 2, instead, k is chosen to ensure that $\frac{\max(\|A_{11}\|, \|A_{22}\|)}{2^k} < 2/5$, and thus $k > \log_2(5\max(\|A_{11}\|, \|A_{22}\|)) - 1$. Thus, we now get $\gamma_i \approx \frac{5}{2}\text{EPS}\frac{\max(\|A_{11}\|, \|A_{22}\|)}{\|A_{ii}\|}$, $i = 1, 2$, and $\gamma_3 \approx \text{EPS}$. Therefore, for Algorithm 2, one ends up with the following approximate bounds

$$\|B_{ii}\| \leq \frac{5}{2}\frac{\max(\|A_{11}\|, \|A_{22}\|)}{\|A\|}\ , i = 1, 2, \quad \text{and} \quad \|B_{12}\| \leq \frac{\|A_{12}\|}{\|A\|}\,. \tag{2.13}$$

Thus, an appropriate value of "condition number" to measure relative errors for Algorithm 2 is essentially the same as (2.8).

Remark 2.2. Observe that (2.13), and (2.6)–(2.8), hints that a possible loss of precision on the diagonal blocks may be experienced whenever these blocks have widely different norms.

Example 2.3. Consider once more our test problem, Example 1.1. Let us first assess its (inherent) conditioning. With some algebra, we obtain the following outcome

- from (2.8): $\kappa_b(F(A)) \approx |\omega| = O(1)$; from (2.9), $\kappa(F(A_{ii})) \approx |\omega| = O(1)$.

In other words, the restricted search for the norm of the Fréchet derivative in the direction of the allowed block triangular B's in (2.8) gives very different outcome than the unrestricted search amongst all possible directions B's: the problem is perfectly well conditioned with respect to (2.8).

As far as the errors we should have experienced when using Algorithm 1 or 2, we have the following situation. For simplicity, fix $\omega = 2.1$. Recall that –with either Algorithm 1 or 2– we have computed the exact exponential e^{A+E}, where $E = \left[\begin{smallmatrix} a & b \\ 0 & c \end{smallmatrix}\right]$.

(1) From (2.10) and (2.11) (i.e., relatively to Algorithm 1), we anticipate a relative error on $F(A)$ of size x EPS $= O(\text{EPS}\|A\|)$. Indeed, in this case the error matrix E has $|a|, |c| \approx \text{EPS}\,\frac{\omega}{\omega_1}$, where $\omega_1 = \frac{\omega}{2^k}$ is defined before (2.2), and $|b| \approx \text{EPS}\; x$. With this, and some algebra, one gets $\frac{\|e^{A+E}-e^A\|}{\|e^A\|} \approx 2^k\text{EPS}$. When $k = 21$, the value needed to get $\frac{\|A\|}{2^k} < 0.5$, this predicts a loss of six digits, in agreement with (2.12) and the results of Table 1.

(2) From (2.10) and (2.13) (i.e., relatively to Algorithm 2) we anticipate a relative error of size $|\omega|\text{EPS} = O(\text{EPS})$. Indeed, in this case for the error matrix E we have $|a|, |c| \approx \text{EPS}\;\omega$, and $|b| \approx \text{EPS}\; x$. Again, with a bit of algebra, one now gets

$$\frac{\|e^{A+E} - e^A\|}{\|e^A\|} \approx \text{EPS}\,.$$

That is, no loss of precision should be experienced, in agreement with (2.8) and the results in Table 1.

In a similar way to what we have done on the above example, we now look for upper bounds on $\|F'(A)B\|$ in the general case of A as in (1.1). These bounds can then be used in (2.8) and (2.10) to obtain upper bounds on the conditioning of the problem and on the "condition numbers" of Algorithms 1 and 2. We will use the following well known inequalities, which can be found in [9]:

$$\|e^{At}\| \le M(t)\, e^{at}, \qquad \text{and} \qquad \|F'(At)B\| \le \|B\| M^2(t) e^{at}, \quad t \ge 0, \quad \text{where} \tag{2.14}$$

(i) *norm estimates*: $a = \|A\|$, $M(t) \equiv 1$; or

(ii) *logarithmic norm estimates*: $a = \mu(A)$, $M(t) \equiv 1$, and $\mu(A)$ is the logarithmic norm of A, that is the largest eigenvalue of $(A + A^T)/2$; or

(iii) *Schur form estimates*: $a = \alpha(A)$, $M(t) = \sum_{k=0}^{n-1} \frac{\|N\|^k t^k}{k!}$, where $\alpha(A)$ is the spectral abscissa of A, that is the largest real part of the eigenvalues of A, and N is the strictly upper triangular part in a Schur form of A.

The following result gives three different upper bounds on $\|F'(A)B\|$. We stress that these are upper bounds, and may severely overestimate the true value of $\|F'(A)B\|$.

Theorem 2.4. *Let* $A = \begin{bmatrix} A_{11} & A_{12} \\ 0 & A_{22} \end{bmatrix}$ *and* $B = \begin{bmatrix} B_{11} & B_{12} \\ 0 & B_{22} \end{bmatrix}$. *Then*

$$\|F'(A)B\| \le \|D_1\; F'(|A|_a)|B|_b\, D_2\;\| \,, \tag{2.15}$$

where we have set $|A|_a = \begin{bmatrix} a_1 & \|A_{12}\| \\ 0 & a_2 \end{bmatrix}$, $|B|_b = \begin{bmatrix} M_1\|B_{11}\| & \|B_{12}\| \\ 0 & M_2\|B_{22}\| \end{bmatrix}$, $D_1 = \begin{bmatrix} M_1 & 0 \\ 0 & 1 \end{bmatrix}$, $D_2 = \begin{bmatrix} 1 & 0 \\ 0 & M_2 \end{bmatrix}$, *and*

(i) $a_i = \|A_{ii}\|$ *and* $M_i = 1$, $i = 1, 2$, *or*
(ii) $a_i = \mu(A_{ii})$ *(logarithmic norm of* A_{ii}*), and* $M_i = 1$, $i = 1, 2$, *or*
(iii) $a_i = \alpha(A_{ii})$ *(spectral abscissa of* A_{ii}*), and* $M_i = M_{ii}(1) = \sum_{k=0}^{n_i-1} \frac{\|N_{ii}\|^k}{k!}$, *with* N_{ii} *the strict upper triangular part of a Schur form of* A_{ii}, $i = 1, 2$.

Proof. For the diagonal blocks we have

$$[F'(A)B]_{ii} = F'(A_{ii})B_{ii}\,, \quad \text{and} \quad \|F'(A_{ii})B_{ii}\| \le M_i^2\|B_{ii}\|e^{a_i} = M_i[F'(|A|_a)|B|_b]_{ii}\,.$$

For the $(1, 2)$ block we have

$$\begin{aligned} &F_{12}(A + hB) - F_{12}(A) \;= \\ &\int_0^1 e^{(A_{11}+hB_{11})(1-s)}(A_{12} + hB_{12})e^{(A_{22}+hB_{22})s}ds - \int_0^1 e^{A_{11}(1-s)}A_{12}e^{A_{22}s}ds \;= \\ &\int_0^1 e^{(A_{11}+hB_{11})(1-s)}hB_{12}e^{(A_{22}+hB_{22})s}ds + \int_0^1 \left[e^{(A_{11}+hB_{11})(1-s)} - e^{A_{11}(1-s)}\right]A_{12}e^{(A_{22}+hB_{22})s}ds \\ &\qquad + \int_0^1 e^{A_{11}(1-s)}A_{12}\left[e^{(A_{22}+hB_{22})s} - e^{A_{22}s}\right]ds. \end{aligned}$$

Now divide by h and take the limit as $h \to 0$ under the integral signs:

$$\begin{aligned} [F'(A)B]_{12} = &\int_0^1 e^{A_{11}(1-s)}B_{12}e^{A_{22}s}ds \\ &+ \int_0^1 \big[F'(A_{11}(1-s))B_{11}(1-s)\big]A_{12}e^{A_{22}s}ds + \int_0^1 e^{A_{11}(1-s)}A_{12}\big[F'(A_{22}s)B_{22}s\big]ds. \end{aligned}$$

Taking norms and using (2.14) we get:

$$\begin{aligned} \|[F'(A)B]_{12}\| \le \|B_{12}\| &\int_0^1 M_{11}(1-s)e^{a_1(1-s)}M_{22}(s)e^{a_2s}ds \\ &+ \|B_{11}\|\,\|A_{12}\| \int_0^1 [M_{11}(1-s)]^2e^{a_1(1-s)}(1-s)M_{22}(s)e^{a_2s}ds \\ &+ \|B_{22}\|\,\|A_{12}\| \int_0^1 M_{11}(1-s)e^{a_1(1-s)}[M_{22}(s)]^2e^{a_2s}sds\,. \end{aligned}$$

Putting together the three terms of the last inequality and using $M_{ii}(t) \le M_{ii}(1) = M_i$, $t \in [0, 1]$, we get

$$\|[F'(A)B]_{12}\| \le M_1M_2 \int_0^1 e^{a_1(1-s)}\left[\|B_{12}\| + \|A_{12}\|\left(M_1\|B_{11}\|(1-s) + M_2\|B_{22}\|s\right)\right]e^{a_2s}ds.$$

Letting $b_{ii} = M_i\|B_{ii}\|$, $b_{12} = \|B_{12}\|$, $a_{ii} = a_i$ and $a_{12} = \|A_{12}\|$ one finally has

$$\|[F'(A)B]_{12}\| \le M_1M_2 \int_0^1 e^{a_{11}(1-s)}(b_{12}+a_{12}(b_{11}(1-s)+b_{22}s))e^{a_{22}s}ds = M_1M_2[F'(|A|_a)|B|_b]_{12}$$

and (2.15) is proven. □

Example 2.5. To illustrate the theorem, let us consider (2.15)-(i) in the case of $\|A_{11}\| = \|A_{22}\|$ (similar estimates to those below are obtained in case $\|A_{11}\| \ne \|A_{22}\|$). For the (1,2) entry of (2.15) we have

$$(F'(|A|_a)|B|_b)_{12} = e^{\|A_{11}\|}\left(\|B_{12}\| + \frac{1}{2}(\|B_{11}\|\|A_{12}\| + \|B_{22}\|\|A_{12}\|)\right),$$

so that

$$F'(|A|_a)|B|_b = e^{\|A_{11}\|}\begin{bmatrix} \|B_{11}\| & \|B_{12}\|+\frac{1}{2}\|A_{12}\|(\|B_{11}\|+\|B_{22}\|) \\ 0 & \|B_{22}\| \end{bmatrix}.$$

Now, if $\|A_{12}\| \le 2$, since $\|B_{ij}\| \le 1$, then we have: $\|F'(|A|_b)|B|_b\| \le e^{\|A_{11}\|}\| \begin{bmatrix} \|B_{11}\| & \sqrt{3} \\ 0 & \|B_{22}\| \end{bmatrix} \|$ $\approx \sqrt{3}e^{\|A_{11}\|}$. Instead, if $\|A_{11}\| = O(1) \ll \|A_{12}\| \approx \|A\|$, then we recover the situation of Example 1.1.

At this point, we can draw a summary of what are the implications of adopting the revised measure (2.8) to assess the (inherent) conditioning of e^A when A is block upper triangular.

(1) Each algorithm needs a "condition number" tailored to the class of perturbations which the algorithm has produced. In particular, for a 2×2 block triangular matrix, Algorithm 1 requires us to look at (2.12), which –compared with (2.8)– betrays potential instabilities of Algorithm 1 in case $\|A_{11}\| \approx \|A_{22}\| \ll \|A_{12}\| \approx \|A\|$. For Algorithm 2, instead, (2.8) provides an adequate measure of condition number.

(2) Problems in which $\|A_{11}\|$ and $\|A_{22}\|$ are very different are potentially ill conditioned, in the sense that (2.8) may be large. Instead, problems for which $\|A_{11}\| \approx \|A_{22}\| = O(1) \ll \|A\| \approx \|A_{12}\|$ are perfectly conditioned, in sharp contrast to what is predicted by classical conditioning theory.

(3) For problems where $\|A_{11}\|$ and $\|A_{22}\|$ are of very different magnitude, consideration of (2.9) and of the relative error bounds (2.2) says that it may be possible to approximate more accurately F_{ii}, $i = 1, 2$, than $F(A)$. If this is warranted by the particular application, then it may be justified to use a more expensive algorithm which separately computes F_{11}, F_{22}, and F_{12}. For example, this can be achieved with the Fréchet phase of the algorithm of [7], or also using diagonal Padé approximations (with distinct scaling factors) to separately approximate F_{11} and F_{22}, and then use Algorithm 2 to approximate $F(A)$, only retaining the approximation of F_{12}.

Case of p blocks.

Next, we would like to extend the previous considerations to the case of A block triangular with p blocks:

$$A = \begin{bmatrix} A_{11} & A_{12} & \dots & A_{1p} \\ 0 & A_{22} & \dots & A_{2p} \\ \vdots & \ddots & \ddots & \vdots \\ 0 & \dots & 0 & A_{pp} \end{bmatrix} . \tag{2.16}$$

Again, $F(A)$ will have the same block structure, and so will an approximation $\hat{F}(A)$ produced by Padé approximation with scaling and squaring. Since the diagonal blocks of $\hat{F}(A)$ have no eigenvalues with negative real part, then $\hat{F}(A)$ has a unique real logarithm, call it $\hat{L}$, with all eigenvalues with imaginary part in $(-\pi, \pi)$. Obviously, then, $\hat{F}(A) = e^{A+E}$, where $E = \hat{L} - A$. A detailed characterization of the backward error matrix E is not available, although the characterization resulting from a 2×2 block partitioning of (2.16) and E as in (2.1) and (2.2) is still possible. Moreover, some simple observations can still be made: (i) E has the same block triangular structure as A; (ii) the blocks E_{ij}, $i = 1, \dots, p,\ j = i+1, \dots, p$, depend only on the sub-matrices $A(i:j)$ of A: $A(i:j) := \begin{bmatrix} A_{ii} & \dots & A_{ij} \\ & \ddots & \vdots \\ & & A_{jj} \end{bmatrix}$. Because of these two facts, an appropriate measure of conditioning of $F(A)$ now should at the very least read as

$$\kappa_b(F(A)) \ = \max_{B=\begin{bmatrix} B_{11} & \dots & B_{1p} \\ & \ddots & \vdots \\ & & B_{pp} \end{bmatrix}, \|B\|=1} \|F'(A)B\| \frac{\|A\|}{\|F(A)\|} . \tag{2.17}$$

Unfortunately, we do not have a good understanding of the relative order of magnitudes of the blocks B_{ij} which can lead us to a definition similar to the one of the set S_b in (2.6), and this forces a maximization amongst (the most favorable) 2×2 block partitioning of B as in (2.8). In any case, it should be appreciated that in the process of approximating $F(A)$, we have also approximated the subproblems $F(A(i:j))$, each of which has its own condition number[2]

$$\begin{aligned} \kappa_b(F(A(i:j))) \ &= \max_{\|B(i:j)\|=1} \|F'(A(i:j))B(i:j)\| \frac{\|A(i:j)\|}{\|F(A(i:j))\|} , \\ \text{for} \quad & i = 1, \dots, p,\ j = i+1, \dots, p , \end{aligned} \tag{2.18}$$

where again there will be order of magnitude restrictions on $B(i:j)$ resulting from a 2×2 block partitioning of it, as in (2.8) and (2.15). Once more, (2.18) reveals that some pieces of $F(A)$, namely $F(A(i:j))$ for some $i > 1$ and/or $j < p$, may be better conditioned –and thus can be approximated more accurately– than $F(A)$. We exemplify this occurrence on the following example.

Example 2.6. Take $A = \begin{bmatrix} 1 & x & x^2/2 \\ 0 & 1 & x \\ 0 & 0 & 1 \end{bmatrix}$, with $x \gg 1$. (Here, $e^A = e \begin{bmatrix} 1 & x & x^2 \\ 0 & 1 & x \\ 0 & 0 & 1 \end{bmatrix}$.) In this case, one obtains that $\kappa_b(F(A)) \approx x\kappa_b(F(A(1:2))) \approx x$. $\square$

Impact of Schur reduction.

So far, see (2.8), we have relied on having A in the form (1.1). Although this structure arises naturally in some applications (see [10]), if one has a general matrix $A \in \mathbb{R}^{n \times n}$ then

[2] in (2.18), $B(i:j)$ is a block triangular matrix of same structure as $A(i:j)$

block triangular structure is usually arrived at via Schur reduction. Let us briefly consider this situation.

Let Q be an orthogonal matrix giving a block Schur reduction of A to the form (1.1). Then, $F(A) = QF(R)Q^T$ and we can use Algorithm 1 or 2 to compute $F(R)$:
Schur–Padé strategy

(i) Let orthogonal Q such that $Q^TAQ = R = \left[\begin{smallmatrix} R_{11} & R_{12} \\ 0 & R_{22} \end{smallmatrix}\right]$.
(ii) Use Algorithm 1 or 2 to approximate $F(R)$ with $\hat{F}(R)$.
(iii) Let $\hat{F}(A) = Q\hat{F}(R)Q^T$.

What is an appropriate condition number for this algorithm? Optimistically, we would like to say that that the backward error matrix produced by the Schur–Padé algorithm has the structure QEQ^T with E produced by Algorithm 1 or 2 to compute $F(R)$. But, in general, this is not true. There are two error matrices which are produced:

(a) the Schur reduction process (see [2]) delivers a matrix R such that $Q^T(A+E_1)Q = R$, and –at best– all we can reasonably assume on E_1 is that $\|E_1\| \approx \text{EPS}\|A\|$;
(b) by Algorithm 1 or 2, one then computes $\hat{F}(R) = F(R + E_R)$, where E_R has same block triangular structure as R, and $\|E_R\| \approx \text{EPS}\|R\|$ (possibly also in a block sense).

Eventually, thus, at first order in EPS, one computes $QF(R + E_R)Q^T = F(QRQ^T + QE_RQ^T) = F(A + E_1 + E_2) = F(A + E)$, where $E_2 = QE_RQ^T$ and $E = E_1 + E_2$. At this point, in general, E does not have any particularly exploitable structure, and all we are able to say is that (at best) $\|E\| \approx \text{EPS}\|A\|$. Therefore, in this case, (2.4) remains an appropriate measure of condition number for an algorithm based on the Schur–Padé strategy.

Example 2.7. Let us revisit Example 1.1. We take

$$A = \frac{1}{2}\left[\begin{smallmatrix} 1.9-x & x+0.1 \\ -x-0.1 & 1.9+x \end{smallmatrix}\right], \quad A = QRQ^T, \quad R = \begin{bmatrix} 1 & x \\ 0 & 0.9 \end{bmatrix}, \quad x = \sqrt{3} \times 10^6, \; Q = \frac{\sqrt{2}}{2}\begin{bmatrix} 1 & -1 \\ 1 & 1 \end{bmatrix}.$$

Clearly, $e^A = Qe^RQ^T$, and $e^R = \begin{bmatrix} e & 10x(e - e^{0.9}) \\ 0 & e^{0.9} \end{bmatrix}$. With some work, we now obtain the estimate $\kappa(F(A)) \approx x^2 = O(\|A\|^2)$. Indeed, if we use either Algorithm 1 or 2 to approximate $F(R)$ in step (ii) of the Schur Padé strategy, we eventually lose eleven digits on $F(A)$, which is in good agreement with the value of $\kappa(F(A))$. Of course, Algorithm 2 is less expensive and must be the preferred choice.

3. Conclusion

We revisited conditioning for the exponential of a block triangular matrix. Our measure (2.8) is an improvement over the classical measure of conditioning (2.4), and provides more reliable feedback on the goodness of an answer obtained by a stable algorithm. In Theorem 2.4 we have given three different upper bounds on the new condition number, but we have not discussed how to compute practical estimates for it. This may be an interesting task for future work.

References

[1] L. Dieci and A. Papini. Padé approximation for the exponential of a block triangular matrix. *Linear Algebra Applic.*, 308:183–202, 2000.

[2] G. H. Golub and C. F. Van Loan. *Matrix Computations*. The Johns Hopkins University Press, 2nd edition, 1989.

[3] B. Kagström. Bounds and perturbation bounds for the matrix exponential. *BIT*, 17:39–57, 1977.

[4] B. Kagström. Computation of matrix functions. Technical Report Report UMINF-58.77, Dept. Inf. Processing, Univ. of Umea, Umea, Sweden, 1977.

[5] C. Kenney and A. J. Laub. Condition estimates for matrix functions. *SIAM J. Matrix Anal. Appl.*, 10:191–209, 1989.

[6] C. Kenney and A. J. Laub. Small–sample statistical condition estimates for general matrix functions. *SIAM J. Sci. Comput.*, 15:36–61, 1994.

[7] C. Kenney and A. J. Laub. A Schur-Fréchet algorithm for computing the logarithm and exponential of a matrix. *SIAM J. Matrix Anal. Appl.*, 19:640–663, 1998.

[8] U. Kwakernaak and R. Sivan. *Linear Optimal Control Systems*. Wiley-Interscience, 1972.

[9] C. Van Loan. The sensitivity of the matrix exponential. *SIAM J. Numer. Anal.*, 14:971–981, 1977.

[10] C. Van Loan. Computing integrals involving the matrix exponential. *IEEE. Trans. Autom. Control*, 23:395–404, 1978.

[11] C. B. Moler and C. Van Loan. Nineteen dubious ways to compute the exponential of a matrix. *SIAM Rev.*, 20:801–836, 1978.

[12] J. Rice. A theory of condition. *SIAM J. Numer. Anal.*, 3:287–310, 1966.

[13] R. C. Ward. Numerical computation of the matrix exponential with accuracy estimates. *SIAM J. Numer. Anal.*, 14:600–610, 1977.

School of Mathematics, Georgia Institute of Technology, Atlanta, GA 30332 U.S.A.
E-mail address: dieci@math.gatech.edu

Dep. Energetica S. Stecco, Univ. of Florence, via C. Lombroso 6/17, 50134 Florence, Italy
E-mail address: papini@de.unifi.it