Optimality of the Paterson-Stockmeyer method for evaluating matrix polynomials and rational matrix functions

Fasi, Massimiliano

2018

ISSN 1749-9097

Optimality of the Paterson–Stockmeyer Method for Evaluating Matrix Polynomials and Rational Matrix Functions[*]

Massimiliano Fasi[†]

Abstract

Many state-of-the-art algorithms reduce the computation of transcendental matrix functions to the evaluation of polynomial or rational approximants at a matrix argument. This task can be accomplished efficiently by resorting to the Paterson–Stockmeyer method, an evaluation scheme originally developed for matrix polynomials that extends quite naturally to rational functions. An important feature of this technique is that the number of matrix multiplications required to evaluate an approximant of order n grows slower than n itself, with the result that different approximants yield the same asymptotic computational cost. We analyze the number of matrix multiplications required by the Paterson–Stockmeyer method and by two widely used generalizations, one for evaluating diagonal Padé approximants of general functions and one specifically tailored to those of the exponential. In all three cases, we identify the approximants of maximum order for any given computational cost.

Keywords: Paterson–Stockmeyer method, polynomial evaluation, matrix polynomial, matrix rational function, matrix function.

2010 MSC: 15A16, 13M10, 65F60.

1 Introduction

Several numerical methods for evaluating matrix functions, including the state-of-the-art algorithms for computing the exponential [1], [13], [14, Chap. 10], the logarithm [2], [7], trigonometric [3] and hyperbolic functions, and their inverses [5], rely on rational approximation. The special case of polynomial approximants is of particular interest, as it usually yields simpler formulae and often leads to elementary proofs of theoretical results. In the literature, algorithms based on polynomial approximation have been proposed for computing the matrix exponential [6], [8], [9], [20], [21], the matrix logarithm [10], and trigonometric matrix functions [4], [19].

In order to compute $f(A)$, where $f\colon \mathbb{C}^{n\times n} \to \mathbb{C}^{n\times n}$ and $A \in \mathbb{C}^{n\times n}$, these algorithms typically perform three main steps. First, a series of transformations is applied to A,

[*] Version of 30th March 2019. **Funding:** This work was supported by MathWorks and the Istituto Nazionale di Alta Matematica, INdAM–GNCS Project 2018. The opinions and views expressed in this publication are those of the author, and not necessarily those of the funding bodies.

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

in order to obtain a matrix B for which a suitable polynomial or rational approximant to f is guaranteed to deliver a prescribed level of accuracy. This approximant is then evaluated at the matrix B, and an approximation of $f(A)$ is obtained by exploiting algebraic properties of f in order to reverse the transformations initially applied to A.

Let us consider the polynomial

$$p(A) = \sum_{i=0}^{k} c_i A^i, \tag{1}$$

where $k \in \mathbb{N}$ and $c_0, c_1, \ldots, c_k \in \mathbb{C}$. As a polynomial is nothing but a linear combination of powers of its argument, one can evaluate $p(A)$ by explicitly computing the first k powers of A, scaling them by the corresponding coefficients of p, and summing them up. If all the powers $A^2, A^3, \ldots, A^k$ are computed, this algorithm requires $k-1$ matrix multiplications, k matrix scalings, k matrix sums, and one diagonal update of the form $A \leftarrow A + \alpha I$, for $\alpha \in \mathbb{C}$, which can be performed efficiently without explicitly forming the diagonal matrix αI. This technique requires at least $2n^2$ additional elements of storage, as it is necessary to keep track of the intermediate powers of A and of the accumulated partial sum.

A second evaluation scheme for (1) is the matrix version of Horner's method. This is the algorithm of choice for scalar polynomials, as it reduces the number of multiplications to be performed without affecting that of scalar sums. In order to employ this scheme, we define the recursion

$$\begin{aligned} P_{k-1} &= c_k A + c_{k-1} I, \\ P_i &= P_{i+1} A + c_i I, \qquad i = k-2, k-3, \ldots, 0, \end{aligned} \tag{2}$$

and evaluate $p(A) = P_0$ by computing P_i for i from $k-1$ down to 0. For dense polynomials, this method requires $k-1$ matrix multiplications, but only one matrix scaling and k diagonal updates, and can be implemented in a memory efficient way that requires only a half of the additional storage needed by the algorithm that evaluates $p(A)$ by explicitly computing powers of A.

In order to reduce the number of matrix multiplications needed to form $p(A)$, Paterson and Stockmeyer [17] proposed a less straightforward approach, which for $k \geqslant 4$ yields an operation count lower than that of the two techniques discussed thus far. By collecting powers of A in a suitable fashion, for $s \in \mathbb{N}^+ := \mathbb{N} \setminus \{0\}$ we obtain

$$p(A) = \sum_{i=0}^{\nu} (A^s)^i B_i^{[p]}(A), \qquad \nu = \left\lfloor \frac{k}{s} \right\rfloor, \tag{3}$$

where

$$B_i^{[p]}(A) = \begin{cases} \displaystyle\sum_{j=0}^{s-1} c_{si+j} A^j, & i = 0, 1, \ldots, \nu - 1, \\ \displaystyle\sum_{j=0}^{|k|_s} c_{si+j} A^j, & i = \nu, \end{cases}$$

Here $|a|_b$ denotes, for two integers a and b, the reminder of the integer division of a by b. In other words, if $|a|_b = \delta \in \mathbb{N}$, then $a = \gamma b + \delta$ for some $\gamma \in \mathbb{N}$. If $\delta = 0$, that is, if a is an integer multiple of b, we write $b \mid a$.

The scheme (3) requires $k - r + 1$ matrix scalings and sums, and $r + 1$ diagonal updates; computing A^2, A^3, $\dots$, A^s requires $s - 1$ matrix multiplications, and, at the price of storing these $s - 1$ additional matrices, no extra multiplication is needed to compute $B_i^{[p]}(A)$, for $i = 0, \dots, \nu$. By evaluating (3) à la Horner, we obtain the recursion

$$\begin{aligned} \widetilde{P}_{\nu-1} &= \begin{cases} c_k A^s + B_{\nu-1}^{[p]}(A), & s \mid k, \\ A^s B_\nu^{[p]}(A) + B_{\nu-1}^{[p]}(A), & s \nmid k, \end{cases} \\ \widetilde{P}_i &= A^s P_{i+1} + B_i^{[p]}(A), \qquad\qquad i = \nu - 2, \nu - 3, \dots, 0, \end{aligned} \tag{4}$$

and computing $p(A) = \widetilde{P}_0$ requires $\nu - 1$ additional matrix multiplications if k is a multiple of s, and ν if it is not. Therefore, evaluating (1) by means of (3) requires

$$C_s^p(k) := s - 1 + \left\lfloor \frac{k}{s} \right\rfloor - [s \mid k] \tag{5}$$

matrix multiplications, where $[\,\cdot\,]$ denotes the Iverson bracket, defined, for a proposition $\mathcal{P}$, by

$$[\mathcal{P}] = \begin{cases} 1, & \text{if } \mathcal{P} \text{ is true}, \\ 0, & \text{if } \mathcal{P} \text{ is false}. \end{cases}$$

Taking the derivative of (5) with respect to s shows that the continuous relaxation of $C_s^p(k)$ is minimized by taking

$$s^\star = \sqrt{k}. \tag{6}$$

As s must be an integer, we can choose either $s = \lfloor \sqrt{k} \rfloor$ or $s = \lceil \sqrt{k} \rceil$. These two choices, together with the evaluation scheme (3), give two variants of the Paterson–Stockmeyer method. Note that this evaluation scheme is not defined for $k = 0$. Hargreaves [12, Thm. 1.7.4] proved that, in fact, these two algorithms have the same cost for any $k \in \mathbb{N}$. In the next section, we provide a new proof of this result, in which we establish the notation and present techniques we will rely on later on.

It is important to pinpoint that this approach trades off memory for computational efficiency, since $s + 1$ additional matrices need to be stored, for a space complexity of $O(\sqrt{k}n^2)$. Van Loan [22] showed that, by computing $p(A)$ one column at a time, it is possible to reduce the storage requirement of the algorithm to $3n^2$ additional elements, at the price of $(\alpha \log_2 s - 1)n^3$ additional flops, where α is a small constant that depends only on s. How to implement the original Paterson–Stockmeyer algorithm and this variant in a memory and communication efficient way has been recently discussed by Hoffman, Schwartz, and Toledo [15].

We note that the Paterson–Stockmeyer method is not the fastest known algorithm for evaluating polynomials of matrices: Paterson and Stockmeyer [17] discuss a technique that requires fewer matrix multiplications than the algorithm above, and an alternative approach for reducing the number of matrix multiplications to evaluate polynomials of matrices has recently been proposed by Sastre [18]. These algorithms evaluate several appropriately chosen polynomials of lower degree, whose coefficients are obtained from those of the original polynomial by means of various techniques. This preprocessing stage may introduce numerical instabilities, thus the new coefficients must be carefully chosen on a case-by-case basis, as done for example in [20] for the truncated Taylor approximants to the exponential of order 8, 15, 24, and 30.

Polynomials of the form (1) often arise when computing matrix functions by relying on Padé approximation. A rational function $r_{km} = p_{km}/q_{km}$, for $k, m \in \mathbb{N}$, is the $[k/m]$ Padé approximant to f at 0 if p_{km} and q_{km} are polynomials of degree k and m, respectively, $q_{km}(0) = 1$, and the first $k+m$ terms in the series expansion of $f(x) - r_{km}(x)$ at 0 are zero. In particular, we focus on truncated Taylor series, for which $m = 0$, and diagonal Padé approximants, for which $m = k$, since these are the two families of Padé approximants most commonly encountered in the literature. Subdiagonal Padé approximants are also considered [11], [16], but the partial fraction form is usually preferred for their evaluation.

The scheme (3) readily generalizes to the evaluation of rational matrix functions: after computing the first s powers of A, for some $s \in \mathbb{N}^+$, one can evaluate numerator and denominator separately, by means of (3), and then solve a multiple right-hand side linear system. An approximately optimal value for s can be determined by minimizing the continuous relaxation of the corresponding cost function.

Since the cost of matrix multiplications is asymptotically higher than that of matrix scalings and matrix sums, we follow the customary practice of measuring the efficiency of algorithms for evaluating polynomials of matrices by counting the numbers of matrix multiplications that need to be performed [14, Chap. 4]. The goal of this work is twofold. On the one hand, we study the optimality of the Paterson–Stockmeyer method amongst all methods of the form (3); on the other, we give several results that can aid in developing numerical algorithms for the computation of matrix functions. Now we summarize our contribution while outlining the structure of the following sections.

It has been observed [14, p. 74] that the Paterson–Stockmeyer method minimizes the number of matrix multiplications required to evaluate polynomials of degree between 2 and 16 by means of the scheme (3). In section 2.1 we show that this is in fact the case for polynomials of any degree.

When matrix functions are approximated by means of polynomials, it is customary not to consider all possible approximants, but only those that maximize the approximation degree for a given number of matrix multiplications. For example, since $C_s^p(11) \geqslant 5$ and $C_s^p(12) \geqslant 5$ for any $s \in \mathbb{N}^+$, there is little point in considering an approximant of degree 11 when that of degree 12 is likely to deliver a more accurate approximation at the same cost. The following definition allows us to make this notion precise and extend it to the case of rational approximants.

Definition 1 (Optimal orders of an evaluation scheme)**.** Let $C(k)$, for $k \in \mathbb{N}$, be the number of matrix multiplications required by a scheme $\mathcal{S}$ to evaluate an approximant of order k. Then $k' \in \mathbb{N}$ is an optimal order (or degree, if the approximant is a polynomial) for $\mathcal{S}$ if there exists $\zeta \in \mathbb{N}$ such that

$$k' = \max\{k \in \mathbb{N} : C(k) = \zeta\}.$$

When working with fixed precision arithmetic, the order of the highest approximant that may be needed to achieve the required accuracy, $k_{\max}$ say, is typically known when the algorithm is being designed, and only the optimal orders smaller than $k_{\max}$ are needed. These can be found by inspecting the values of $C(k)$ for $k \leqslant k_{\max}$, as was done in [14, Table 4.1] and [6, Table 1] for polynomial approximants and in [14, Table 10.3] for the diagonal Padé approximants to the exponential. In arbitrary precision floating-point environments, however, depending on the working precision and the desired accuracy, an approximant of arbitrarily high order may be needed, and alternative techniques to efficiently find all optimal degrees become necessary.

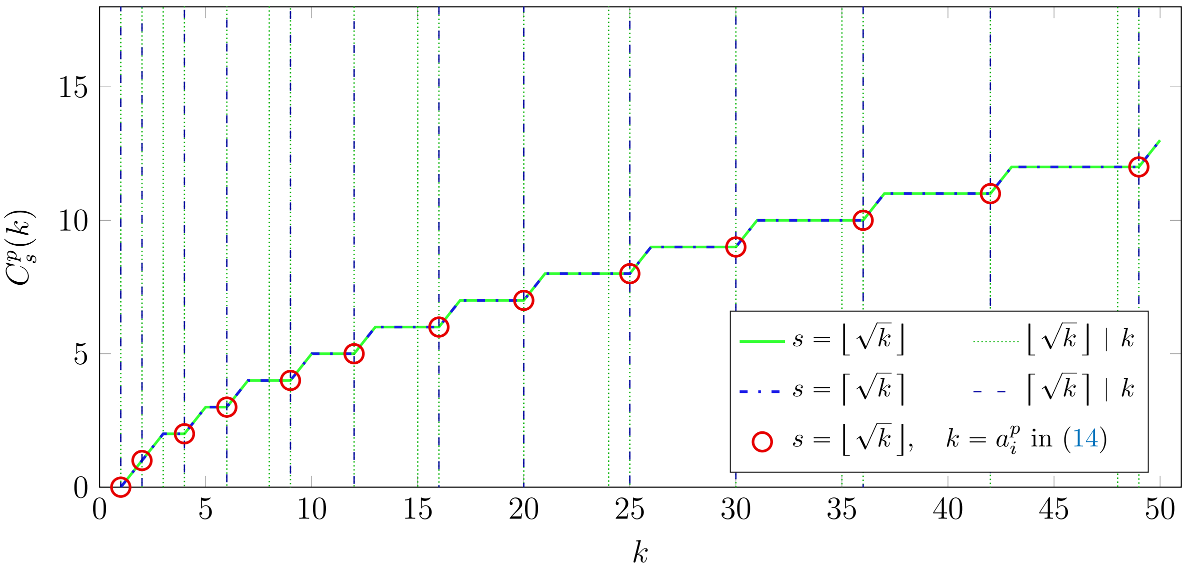

Figure 1: Number of matrix multiplications required to evaluate a polynomial of degree k, for k between 1 and 50, by means of the scheme (3) with $s = \lfloor \sqrt{k} \rfloor$ and $s = \lceil \sqrt{k} \rceil$. Dashed and dotted lines mark the values of k that are integer multiples of $\lfloor \sqrt{k} \rfloor$ and $\lceil \sqrt{k} \rceil$, respectively; the circles mark the number of matrix multiplications required to evaluate polynomials of optimal degree (in the sense of Definition 1) for the Paterson–Stockmeyer method.

In section 2.2, we derive a formula for the sequences of optimal degrees for the Paterson–Stockmeyer method for polynomial evaluation. In section 3, we obtain closed formulae for the optimal orders of the Paterson–Stockmeyer-like scheme for evaluating rational functions whose numerator and denominator have same degree, and in section 4 we consider the special case of the diagonal Padé approximants to the exponential.

Finally, in section 5 we summarize our findings and outline possible directions for future work.

2 Evaluation of matrix polynomials

Figure 1 shows the value of the cost function (5) for the two canonical variants of the Paterson–Stockmeyer method, which differ only in the direction $\sqrt{k}$ is rounded in order to obtain the parameter s in (3). It is well known that both choices yield the same computational cost for the evaluation of a polynomial of any degree, and in section 2.1 we show that this is the minimum value for $C_s^p(k)$ among all choices of $s \in \mathbb{N}^+$. The values marked with a red circle are discussed in section 2.2.

2.1 Optimality of the Paterson–Stockmeyer method

Most of the results that follow stem from a couple of simple observations. If $s = \lfloor \sqrt{k} \rfloor$, then by definition of the floor operator, we have that

$$s \leqslant \frac{k}{s} < \frac{(s+1)^2}{s} = s + 2 + \frac{1}{s}, \tag{7}$$

where the first inequality holds strictly if $\lfloor \sqrt{k} \rfloor \neq \lceil \sqrt{k} \rceil$. It follows that $\lfloor \frac{k}{s} \rfloor = s + t$, where t can only be 0, 1, or 2, and in fact it is convenient to split (7) into the three subcases

$$s + t \leqslant \frac{k}{s} < s + t + 1, \qquad t = 0, 1, 2. \tag{8}$$

Combining (7) and (8) for $t = 2$ with the fact that k is an integer, reveals that $\lfloor \frac{k}{s} \rfloor = s+2$ only if $s \mid k$, that is, only if $k = s(s+2)$.

Theorem 1 (Hargreaves, [12, Thm. 1.7.4])**.** *Let $A \in \mathbb{C}^{n \times n}$ and let p be a polynomial of degree $k \in \mathbb{N}^+$. The two methods obtained by setting s in (3) to $s_f = \lfloor \sqrt{k} \rfloor$ and $s_c = \lceil \sqrt{k} \rceil$ require the same number of matrix multiplications to evaluate $p(A)$.*

Proof. We need to prove that $C^p_{s_f}(k) = C^p_{s_c}(k)$ for any $k \in \mathbb{N}^+$. If k is a perfect square, then $s_f = s_c$ and the result follows immediately. Otherwise, one has that $s := s_f = s_c - 1$, and thus that

$$\Delta(k) := C^p_{s_f}(k) - C^p_{s_c}(k) = \left\lfloor \frac{k}{s} \right\rfloor - [s \mid k] - 1 - \left\lfloor \frac{k}{s+1} \right\rfloor + [s + 1 \mid k]. \tag{9}$$

If $s \mid k$ and $k \neq s^2$, then (7) implies that $\nu = \lfloor \frac{k}{s} \rfloor = \frac{k}{s}$ is either $s+1$ or $s+2$. If $\nu = s + 1$, then $k = s(s+1)$ and $s + 1 \mid k$, and substituting into (9) gives $\Delta(k) = 0$. If $\nu = s + 2$, then

$$\frac{k}{s+1} = \frac{s(s+2)}{s+1} = s + 1 - \frac{1}{s+1},$$

hence $\lfloor \frac{k}{s+1} \rfloor = s$ and $s + 1 \nmid k$, and once again substituting into (9) shows that $\Delta(k) = 0$. When $s + 1 \mid k$, multiplying (7) by $\frac{s}{s+1}$ gives

$$s - 1 + \frac{1}{s+1} < \frac{k}{s+1} < s + 1,$$

which leads back to the case $k = s(s+1)$.

Finally, if $s \nmid k$ and $s + 1 \nmid k$, then $\lfloor \frac{k}{s} \rfloor = s + t$, where t is either 0 or 1, and multiplying (8) by $\frac{s}{s+1}$ gives

$$s + t - 1 - \frac{t-1}{s+1} \leqslant \frac{k}{s+1} < s + t - \frac{t}{s+1},$$

which implies that

$$\left\lfloor \frac{k}{s+1} \right\rfloor = s + t - 1 = \left\lfloor \frac{k}{s} \right\rfloor - 1. \tag{10}$$

Substituting (10) into (9) concludes the proof. ∎

In view of the result in Theorem 1, we can drop the subscript and adopt the notation $C^p(k)$ to indicate the number of matrix multiplications required by the Paterson–Stockmeyer method.

Next, we show that the Paterson–Stockmeyer method is the cheapest algorithm that arises from the evaluation scheme (3). Note that this result is not an obvious consequence of the optimality of $s^\star$ in (6), since the continuous relaxation of (5) does not take into account the discontinuities induced by the floor operator in $\lfloor \frac{k}{s} \rfloor$ and the non-continuous term $[s \mid n]$.

Proposition 1. *Let $A \in \mathbb{C}^{n\times n}$ and let p be a polynomial of degree $k \in \mathbb{N}^+$. The Paterson–Stockmeyer method minimizes the number of matrix multiplications required to evaluate $p(A)$ by means of the evaluation scheme* (3).

Proof. Let $s = \lfloor \sqrt{k} \rfloor$. In view of Theorem 1, it suffices to show that $C^p_{s+\ell}(k) \leqslant C^p_s(k)$, for all $\ell \in \mathbb{Z}$ such that $\ell > -s$. The proof is by exhaustion since, by (7), ν can take only the three values s, $s+1$, and $s+2$. For $t = 0, 1$, or 2, we have that

$$C^p_s(k) = 2s + t - 1 - [s \mid k], \tag{11}$$

and since

$$\frac{k}{s+\ell} \geqslant \frac{s(s+t)}{s+\ell} = s - \ell + t + \eta^\ell_t, \qquad \eta^\ell_t := \frac{\ell(\ell - t)}{s+\ell}, \tag{12}$$

we can conclude that

$$C^p_{s+\ell}(k) = s + \ell - 1 + \left\lfloor \frac{k}{s+\ell} \right\rfloor - [s+\ell \mid k] \geqslant 2s + t - 1 + \lfloor \eta^\ell_t \rfloor - [s+\ell \mid k].$$

For $\nu = s$, η^ℓ_0 is nonnegative, and $C^p_{s+\ell}(k)$ can be strictly smaller than $C^p_s(k)$ only if $s+\ell \mid k$ and $\lfloor \eta^\ell_0 \rfloor = 0$ but $s \nmid k$. By taking the floor of (12), we see that the first condition is satisfied only if $k = (s+\ell)(s-\ell) = s^2 - \ell^2$ for some ℓ. However, k cannot be smaller than s^2, thus the only admissible value for ℓ is 0, in which case $C^p_s(k) = C^p_{s+\ell}(k)$.

For $\nu = s+1$, η^ℓ_1 is nonnegative, and $C^p_{s+\ell}(k) < C^p_s(k)$ only if $k = (s+\ell)(s-\ell+1)$ and $s \nmid k$. Since k must be larger than $s(s+1)$, the only two admissible values for ℓ are 0 and 1, but in both cases we have that $k = s(s+1)$, and thus that $s \mid k$.

Finally, for $t = 2$ and $k = s(s+2)$, observe that $C^p_{s+\ell}(k) \geqslant C^p_s(k)$ unless $\lfloor \eta^\ell_2 \rfloor = -1$ and $s+\ell \mid k$. The former condition is satisfied if and only if $\ell = 1$, but in this case $s+1 \nmid s(s+2)$, since

$$\frac{s(s+2)}{s+1} = s + \frac{s}{s+1}$$

cannot be integer for $s > 0$. □

2.2 Optimal degrees for the Paterson–Stockmeyer method

We can characterize the degrees that are optimal for the Paterson–Stockmeyer method in the sense of Definition 1. In order to accomplish this task, we need to show that the cost function (5) is non-decreasing in k. Again, this result is not obvious because of the terms $\lfloor \frac{k}{s} \rfloor$ and $[s \mid k]$ in (5).

Lemma 1. *The number of matrix multiplications required by the Paterson–Stockmeyer method to evaluate a matrix polynomial is non-decreasing in the degree of the polynomial.*

Proof. We want to show that, for $k \in \mathbb{N}^+$,

$$C^p(k) \leqslant C^p(k+1). \tag{13}$$

As floor and ceiling yield the same operation count, we can restrict ourselves to considering only $s = \lfloor \sqrt{k} \rfloor$ and $s' = \lfloor \sqrt{k+1} \rfloor$. If $s = s'$, then we only need to prove that $\lfloor \frac{k}{s} \rfloor \leqslant \lfloor \frac{k+1}{s} \rfloor$. By adding $\frac{1}{s}$ to all the terms in (8), we get that that, if $\lfloor \frac{k}{s} \rfloor = s + t$, then

$$s + t + \frac{1}{s} \leqslant \frac{k+1}{s} < s + t + 1 + \frac{1}{s},$$

and thus that $\left\lfloor \frac{k+1}{s} \right\rfloor$ is either $s+t$ or $s+t+1$, and cannot be smaller than $\left\lfloor \frac{k}{s} \right\rfloor$. Otherwise, we must have that $s' = s+1$.

If $s \mid k$, then $k = s(s+t)$, for $t = 0, 1$, or 2, and observing that

$$\frac{k+1}{s+1} = \frac{s^2+st+1}{s+1} = s+t-1+\frac{2-t}{s+1},$$

we can conclude that $\left\lfloor \frac{k+1}{s+1} \right\rfloor = s+t-2$. Therefore, if $t = 0$ or 1, then $s+1 \nmid k+1$ and the inequality (13) holds strictly, whereas if $t = 2$, then $k+1 = (s+1)^2$ and the equality is satisfied.

If $s+1 \mid k+1$, then $\left\lfloor \sqrt{k+1} \right\rfloor = s+1$ and $\left\lfloor \sqrt{k} \right\rfloor = s$, which implies that $(s+1)^2 \leqslant k+1$ and $k+1 < (s+1)^2+1$, respectively. By dividing both inequalities by $s+1$, we get that $\frac{k+1}{s+1} = s+1$, which can be rewritten as $k = (s+1)^2 - 1$, and readily implies that $\frac{k}{s} = s+2$. Substituting these values into (13) shows that equality holds in this case.

Finally, when $s \nmid k$ and $s+1 \nmid k+1$, by multiplying all the terms in (8) by s, incrementing them by one, and dividing them by $s+1$, one gets

$$s+t-1+\frac{s}{s+1} \leqslant \frac{k+1}{s+1} < s+t+\frac{1-t}{s+1},$$

which implies that $\left\lfloor \frac{k+1}{s+1} \right\rfloor$ can be either $s+t-1$ or $s+t$. Substituting into (13) shows that the former satisfies the equality and the latter the strict inequality. □

Recall that an integer a is a quarter-square, a perfect square, or an oblong number, if there exists $b \in \mathbb{N}$ such that $a = \lfloor b^2/4 \rfloor$, $a = b^2$, or $a = b(b+1)$, respectively.

Proposition 2. *The degree of a polynomial is optimal for the Paterson–Stockmeyer algorithm if and only if it is a positive quarter-square.*

Proof. By Lemma 1, a degree $k \in \mathbb{N}^+$ is optimal if and only if $C^p(k) < C^p(k+1)$. Since positive quarter-squares are either positive perfect squares or positive oblong numbers, we need to prove only that $C^p(k) < C^p(k+1)$ if and only if $k = s^2$ or $k = s(s+1)$ for some $s \in \mathbb{N}^+$. We have that $\left\lfloor \sqrt{k} \right\rfloor = \left\lfloor \sqrt{k+1} \right\rfloor = s$, and it is straightforward to verify that $C^p(s^2) = 2s-2 < 2s-1 = C^p(s^2+1)$ and $C^p(s(s+1)) = 2s-1 < 2s = C^p(s(s+1)+1)$, and thus that s^2 and $s(s+1)$ are optimal degrees for all $s \in \mathbb{N}^+$.

Conversely, let $k \in \mathbb{N}^+$ be an optimal degree for the Paterson–Stockmeyer method, and let $s = \lfloor \sqrt{k} \rfloor$. Note that if k is not an integer multiple of s, then a polynomial with $s - (k \mod s)$ more terms can be evaluated with the same number of matrix multiplications. Therefore, if k is optimal, then $s \mid k$ and, as a consequence of (7), k must be of the form $s(s+t)$, where $t = 0, 1$, or 2. We already known that if $t = 0$ or $t = 1$, then k is optimal, and we need to show only that $k' := s(s+2)$ is not. Since $k'+1 = (s+1)^2$, we have that $\sqrt{k'+1} \mid k'+1$, and thus that $C^p(k') = 2s = C^p(k'+1)$, which shows that k' is not optimal. □

Therefore, the sequence of optimal degrees for the Paterson–Stockmeyer method is $(a_i^p)_{i \in \mathbb{N}}$, where

$$a_i^p = \left\lfloor \frac{(i+2)^2}{4} \right\rfloor. \tag{14}$$

By observing that $C^p(a_i^p) = i$, we can conclude that the polynomial of highest degree that can be evaluated with i matrix multiplications is that of degree a_i^p.

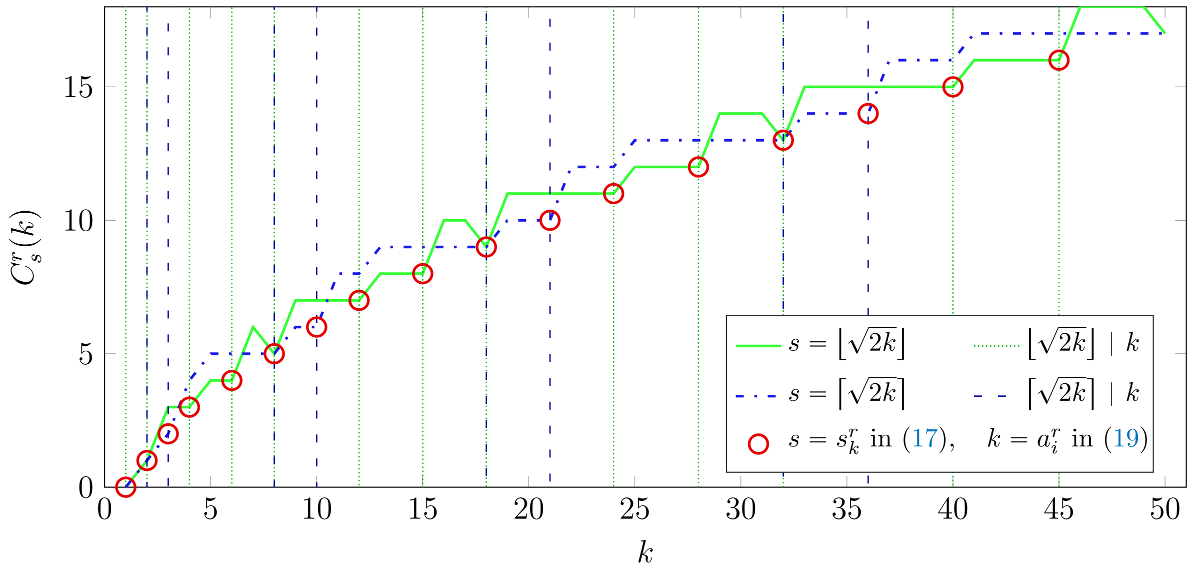

Figure 2: Number of matrix multiplications required to evaluate a rational function of order $[k/k]$, for k between 1 and 50, by means of the scheme (15), for $s = \lfloor\sqrt{2k}\rfloor$ and $s = \lceil\sqrt{2k}\rceil$. The dotted and dashed lines mark the values of k that are integer multiples of $\lfloor\sqrt{2k}\rfloor$ and $\lceil\sqrt{2k}\rceil$, respectively; the circles mark the number of matrix multiplications required to evaluate rational matrix functions of optimal order (in the sense of Definition 1) for the evaluation scheme (15).

3 Rational matrix functions of order $[k/k]$

A rational function is the quotient of two polynomials and, in the matrix case, it can be interpreted as the solution to a multiple right-hand side linear system whose coefficients and constant term are both matrix polynomials. Therefore, the value of a rational function at a matrix argument can be computed by relying on a suitable modification of the scheme (3) capable of minimizing the number of matrix multiplications required to evaluate at once two polynomials at the same matrix argument.

Since in algorithms for computing matrix functions the evaluation of diagonal approximants is typically needed in this section we focus on the evaluation of rational matrix functions of order $[k/k]$. Let us consider the task of evaluating $r(A) = q(A)^{-1}p(A)$, where both p and q are polynomials of degree $k \in \mathbb{N}^+$. We can rewrite numerator and denominator of this rational function as polynomials in A^s, which gives

$$p(A) = \sum_{i=0}^{\nu} B_i^{[p]}(A)(A^s)^i, \qquad q(A) = \sum_{i=0}^{\nu} B_i^{[q]}(A)(A^s)^i, \qquad \nu = \left\lfloor \frac{k}{s} \right\rfloor. \tag{15}$$

If this scheme is used and A^2, A^3, ..., A^s are computed only once, then evaluating $r(A)$ requires the solution of one multiple right-hand side linear system and

$$C_s^r(k) := s - 1 + 2\left\lfloor \frac{k}{s} \right\rfloor - 2[s \mid k] \tag{16}$$

matrix multiplications. The continuous relaxation of (16) is minimized by taking $s = \sqrt{2k}$, but, as Figure 2 shows, depending on k, either taking the floor or the ceiling of this quantity may yield the lowest flop count. Therefore, for $k \in \mathbb{N}^+$, we define use of

$$s_k^r := \arg\min \left\{ C^r_{\lfloor\sqrt{2k}\rfloor}(k), C^r_{\lceil\sqrt{2k}\rceil}(k) \right\}. \tag{17}$$

Figure 2 seems to suggests that if either rounding of $\sqrt{2k}$ divides k, then setting s to it in (15) will give $C^r_{s_k^r}(k)$. In the following we prove that, when that happens, s_k^r in fact minimizes the cost function $C_s^r(k)$ among all possible choices of s.

Lemma 2. *Let $A \in \mathbb{C}^{n\times n}$ and let p and q be polynomials of degree $k \in \mathbb{N}^+$. If $\lfloor\sqrt{2k}\rfloor \mid k$ or $\lceil\sqrt{2k}\rceil \mid k$, then setting s in (15) to $\lfloor\sqrt{2k}\rfloor$ or $\lceil\sqrt{2k}\rceil$, respectively, minimizes the number of matrix multiplications required to evaluate both $p(A)$ and $q(A)$ by the scheme (15).*

Proof. Let $\bar{s} = \lfloor\sqrt{2k}\rfloor$. By definition of the floor operator, $\bar{s}^2 \leqslant 2k < (\bar{s}+1)^2$, and thus

$$\frac{\bar{s}}{2} \leqslant \frac{k}{\bar{s}} < \frac{\bar{s}}{2} + 1 + \frac{1}{2\bar{s}}.$$

Since $\bar{s} \mid k$, we have that $\frac{k}{\bar{s}} = \frac{\bar{s}+t}{2}$, where $t = 0$ or 2 if $\bar{s}$ is even and $t = 1$ if $\bar{s}$ is odd, and thus that $C^r_{\bar{s}}(k) = 2\bar{s} + t - 3$. In order to determine the number of multiplications required when setting $s \neq \bar{s}$ in (15), note that for $\ell \in \mathbb{N}$ such that $\ell > -\bar{s}$, we have

$$\frac{k}{\bar{s}+\ell} = \frac{\bar{s}(\bar{s}+t)}{2(\bar{s}+\ell)} = \frac{1}{2}\left(\bar{s} - \ell + t + \eta_t^\ell\right), \qquad \eta_t^\ell := \frac{\ell^2 - t\ell}{\bar{s}+\ell}. \tag{18}$$

If $\bar{s}+\ell \mid k$, then $\eta_t^\ell \geqslant -\frac{1}{s+1} > -1$, thus $\left\lfloor \frac{k}{\bar{s}+\ell} \right\rfloor \geqslant \frac{\bar{s}-\ell+t}{2}$ and $C^r_{\bar{s}+\ell}(k) \geqslant 2\bar{s}+t-3 = C^r_{\bar{s}}(k)$. On the other hand, if $\bar{s}+\ell \nmid k$, then $\left\lfloor \frac{k}{\bar{s}+\ell} \right\rfloor \geqslant \frac{\bar{s}-\ell-t-1}{2}$, and $C^r_{\bar{s}+\ell}(k) \geqslant 2\bar{s}+t-2 > C^r_{\bar{s}}(k)$.

The proof for $\bar{s} = \lceil\sqrt{2k}\rceil$ is rather similar. From $(\bar{s}-1)^2 < k \leqslant \bar{s}^2$ we have that

$$\frac{\bar{s}}{2} - 1 - \frac{1}{2\bar{s}} < \frac{k}{\bar{s}} \leqslant \frac{\bar{s}}{2},$$

and since $\bar{s} \mid k$, that $\frac{k}{\bar{s}} = \frac{\bar{s}+t}{2}$, for $t = 0, 1,$ or 2. For $\ell > -\bar{s}$, one has that $\frac{k}{\bar{s}+\ell} = \frac{1}{2}(\bar{s} - \ell - t + \eta^\ell_{-t})$, and we can argue as above that if $\bar{s}+\ell \mid k$ then $C^r_{\bar{s}+\ell}(k) \geqslant 2\bar{s}-t-3 = C^r_{\bar{s}}(k)$, while if $\bar{s}+\ell \nmid k$, then $C^r_{\bar{s}+\ell}(k) \geqslant 2\bar{s}-t-2 > C^r_{\bar{s}}(k)$. $\square$

In order to characterize the optimal degrees for the scheme (15), we need to define the cost function $C^r(k) = \min_{1\leqslant s\leqslant k}\{C_s^r(k)\}$, which represents the number of matrix multiplications needed to evaluate a diagonal rational function by means of (15) over all reasonable choices of s. In analogy with quarter-squares, we say that $a \in \mathbb{N}$ is an eight-square if there exists $b \in \mathbb{N}$ such that $a = \lfloor b^2/8 \rfloor$.

Proposition 3. *The degree of numerator and denominator of a rational function is optimal for the evaluation scheme (15) if and only if it is a positive eight-square.*

Proof. Let $r = p/q$, where p and q are polynomials of degree $k \in \mathbb{N}^+$. Note that when $s \nmid k$, then adding $s - (k \mod s)$ more terms to p and q does not increase the number of matrix multiplications required by the scheme (15), thus we only need to consider cases where k is an integer multiple of s.

Let us begin by showing that if k is a positive eight-square then it is optimal. Note that $k = x(2x + t)$, for some $x \in \mathbb{N}^+$, if $k \equiv t \pmod 4$ and $t = 0, 1$, or 2, and that $k = (2x + 1)(x + 1)$ for some $x \in \mathbb{N}$, if $k \equiv 3 \pmod 4$. We consider the four cases separately. In the following, we always assume that $\ell \in \mathbb{Z}$ is such that $\ell > -s$ and that $j \in \mathbb{N}$.

If $k = 2x^2$, then $s = \sqrt{2k} = 2x$, and since $s \mid k$, by Lemma 2 the minimum number of matrix multiplications required to evaluate $r(A)$ is $C_s^r(k) = 2s - 3$. Since

$$\frac{k+j}{s+\ell} = \frac{1}{2}\left(s - \ell + \eta_j^\ell\right), \qquad \eta_j^\ell := \frac{\ell^2 + 2j}{s+\ell},$$

and $\eta_j^\ell > 0$, we have that $s + \ell \mid k + j$ only if $\eta_j^\ell \geqslant 1$, which implies that $C_{s+\ell}^r(k + j) \geqslant 2s - 2 > C_s^r(k)$.

If $k = x(2x+1)$, then k is an integer multiple of $s = \lceil\sqrt{2k}\rceil = 2x+1$, thus $C_s^r(k) = 2s-4$ and

$$\frac{k+j}{s+\ell} = \frac{1}{2}\left(s - \ell - 1 + \eta_j^\ell\right), \qquad \eta_j^\ell := \frac{\ell^2 + \ell + 2j}{s+\ell},$$

Since it is strictly positive, η_j^ℓ must be at least 1 for $s + \ell$ to divide $k + j$, which implies that $C_{s+\ell}^r(k + j) \geqslant 2s - 3 > C_s^r(k)$.

If $k = 2x(x + 1)$, then $s = \lfloor\sqrt{2k}\rfloor = 2x$, and $C_s^r(k) = 2s - 1$. On the other hand,

$$\frac{k+j}{s+\ell} = \frac{1}{2}\left(s - \ell + 2 + \eta_j^\ell\right), \qquad \eta_j^\ell := \frac{\ell^2 - 2\ell + 2j}{s+\ell},$$

where as before $\eta_j^\ell > 0$. In order to have $s + \ell \mid k + j$, we have that η_j^ℓ must be at least 1, which in turn gives that $C_{s+\ell}^r(k + j) = 2s > C_s^r(k)$.

Finally, if $k = (2x + 1)(x + 1)$, then $s = \lceil\sqrt{2k}\rceil = 2x + 1$ and $C_s^r(k) = 2s - 2$ Moreover

$$\frac{k+j}{s+\ell} = \frac{1}{2}\left(s - \ell + 1 + \eta_j^\ell\right), \qquad \eta_j^\ell := \frac{\ell^2 - \ell + 2j}{s+\ell},$$

where $\eta_j^\ell > 0$. As before, since $s + \ell \mid k + j$ only if $\eta_j^\ell \geqslant 1$, we have that $C_{s+\ell}^r(k + j) = 2s - 1 > C_s^r(k)$.

We have established that all eight-squares are optimal degrees for the evaluation scheme (15). In order to prove that all optimal degrees are eight-squares, it suffices to note that for all $n \in \mathbb{N}$ there exists an eight-square k such that $C^r(k) = n$. By Definition 1, optimal orders must be unique, therefore all optimal degrees must be eight-squares. ∎

In view of this result, the sequence of optimal orders for the evaluation scheme (15) with $s = s_k^r$ in (17) is $(a_i^r)_{i \in \mathbb{N}}$, where

$$a_i^r = \left\lfloor \frac{(i+3)^2}{8} \right\rfloor. \tag{19}$$

Moreover, since $C^r(a_i^r) = i$, the rational function of highest order that can be evaluated with i matrix multiplications is that of order $[a_i^r/a_i^r]$.

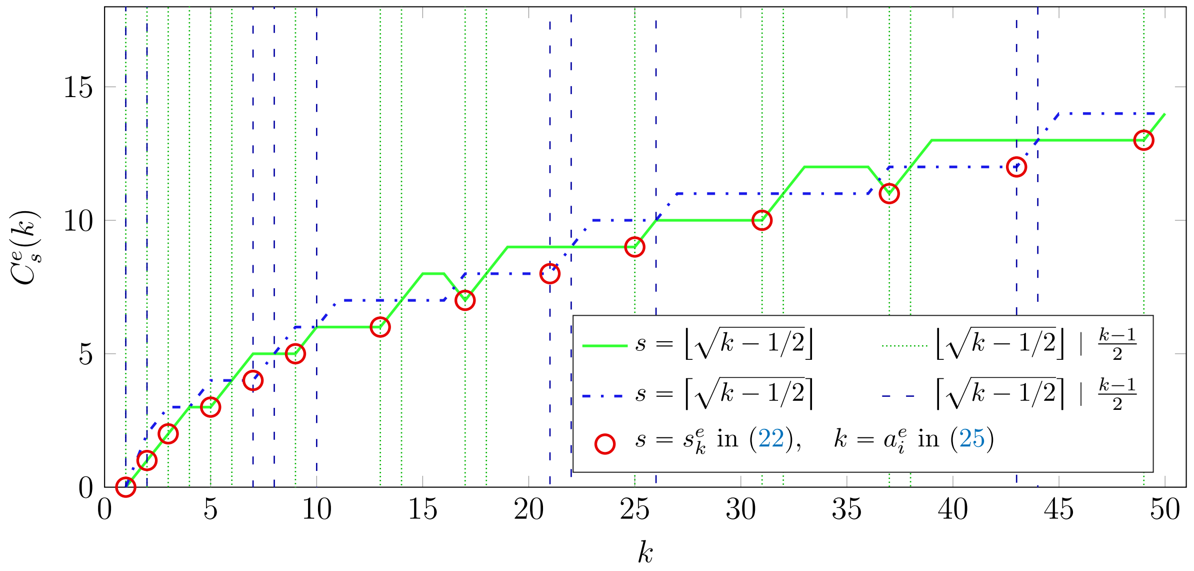

Figure 3: Number of matrix multiplications required to evaluate $[k/k]$ Padé approximant to the matrix exponential, for k between 1 and 50, by means of the scheme (20), for $s = \lfloor\sqrt{k-1/2}\rfloor$ and $s = \lceil\sqrt{k-1/2}\rceil$. The dotted and dashed lines mark the values of k for which $\frac{k-1}{2}$ is an integer multiple of $\lfloor\sqrt{k-1/2}\rfloor$ and $\lceil\sqrt{k-1/2}\rceil$, respectively; the circles mark the number of matrix multiplications required to evaluate the diagonal Padé approximants to the matrix exponential of optimal order (in the sense of Definition 1) for the evaluation scheme (20).

4 Diagonal Padé approximants to the matrix exponential

Let $r = p/q$ be the $[k/k]$ diagonal Padé approximant to the exponential. The evaluation of these rational matrix functions deserves special attention, as the identity $q(x) = p(-x)$ allows for a much faster evaluation of r at a matrix argument. Let $\mu_k^e = \lfloor k/2 \rfloor$ and $\mu_k^o = \lfloor (k-1)/2 \rfloor$. By separating the $\mu_k^e + 1$ powers of A of even degree from the $\mu_k^o + 1$ powers of odd degree, we can write

$$\begin{aligned} p(A) &= \sum_{i=0}^{k} c_i A^i = \sum_{i=0}^{\mu_k^e} c_{2i} A^{2i} + A \sum_{i=0}^{\mu_k^o} c_{2i+1} A^{2i} =: U_e\left(A^2\right) + A U_o\left(A^2\right), \\ q(A) &= p(-A) = U_e\left(A^2\right) - A U_o\left(A^2\right), \end{aligned}$$

which shows that once $U_e\left(A^2\right)$ and $AU_o\left(A^2\right)$ are available, evaluating $p(A)$ and $q(A)$ requires no additional matrix multiplication.

As $U_e\left(A^2\right)$ and $U_o\left(A^2\right)$ are polynomials in A^2, they can be evaluated by means of the scheme

$$U_e\left(A^2\right) = \sum_{i=0}^{\nu_e} B_i^{[U_e]}\left(A^2\right)\left(A^{2s}\right)^i, \qquad U_o\left(A^2\right) = \sum_{i=0}^{\nu_o} B_i^{[U_o]}\left(A^2\right)\left(A^{2s}\right)^i, \tag{20}$$

where $\nu_e = \lfloor \mu_k^e/s \rfloor$ and $\nu_o = \lfloor \mu_k^o/s \rfloor$, and the powers of A^2 are computed only once. Computing $A^2, A^4, \ldots, A^{2s}$ requires s matrix multiplications, evaluating the polynomials $U_e\left(A^2\right)$ and $U_o\left(A^2\right)$ require $\left\lfloor \frac{\mu_k^e}{s} \right\rfloor - [s \mid \mu_k^e]$ and $\left\lfloor \frac{\mu_k^o}{s} \right\rfloor - [s \mid \mu_k^o]$, respectively, and one additional matrix multiplication is needed to compute $AU_o(A^2)$. Therefore evaluating $r(A)$ requires one matrix inversion and

$$C^e_s(k) := s + 1 + \left\lfloor \frac{\mu^e_k}{s} \right\rfloor + \left\lfloor \frac{\mu^o_k}{s} \right\rfloor - [s \mid \mu^e_k] - [s \mid \mu^o_k] \tag{21}$$

matrix multiplications. The continuous relaxation of (21) is approximately minimized by taking $s = \sqrt{k - \frac{1}{2}}$, and as in (17) we define

$$s^e_k := \arg\min \left\{ C^e_{\left\lfloor \sqrt{k-\frac{1}{2}} \right\rfloor}(k), C^e_{\left\lceil \sqrt{k-\frac{1}{2}} \right\rceil}(k) \right\}. \tag{22}$$

Lemma 3. *Let $A \in \mathbb{C}^{n \times n}$, let $k \in \mathbb{N}^+$ be odd, let p and q be the numerator and denominator of the $[k/k]$ Padé approximant to the exponential, respectively, and let $s_f = \left\lfloor \sqrt{k - 1/2} \right\rfloor$ and $s_c = \left\lceil \sqrt{k - 1/2} \right\rceil$. If $s_f \mid \frac{k-1}{2}$ or $s_c \mid \frac{k-1}{2}$, then setting s to s_f or s_c, respectively, minimizes the number of matrix multiplications required to evaluate both $q(A)$ and $p(A)$ by means of the scheme (20).*

Proof. If k is odd, then $\mu^e_k = \mu^o_k = \frac{k-1}{2}$. For s_f, we have

$$\frac{k-1}{2s_f} = \frac{s_f + t}{2}, \tag{23}$$

where $t = 0$ or 2, if s_f is even, and $t = 1$, if s_f is odd, and it is easy to see that $C^e_{s_f}(k) = 2s_f + t - 1$.

From (23), we have that $k - 1 = s_f(s_f + t)$, thus for $\ell > -s_f$

$$C^e_{s_f+\ell}(k) \geqslant \begin{cases} s_f + \ell + 2\left\lfloor \theta^\ell_t \right\rfloor - 1, & s_f \mid \theta^\ell_t, \\ s_f + \ell + 2\left\lfloor \theta^\ell_t \right\rfloor + 1, & s_f \nmid \theta^\ell_t, \end{cases} \qquad \theta^\ell_t := \frac{s_f - \ell + t + \eta^\ell_t}{2}, \qquad \eta^\ell_t := \frac{\ell^2 - t\ell}{s_f + \ell}.$$

If $s_f + \ell \mid \theta^\ell_t$, then $C^e_{s_f+\ell}(k) \geqslant C^e_{s_f}(k)$ if and only if $\left\lfloor \theta^\ell_t \right\rfloor \geqslant \frac{s_f - \ell + t}{2}$. Note that, for $\alpha, \beta \in \mathbb{R}^+$, we have that $\lfloor \alpha \rfloor < \beta$ if and only if $\alpha < \lceil \beta \rceil$, and since $s_f + t$ is even, $s_f - \ell + t$ has the same parity as ℓ. Therefore, we only need to show that there exists no $\ell > -s_f$ such that

$$\theta^\ell_t < \left\lceil \frac{s_f - \ell + t}{2} \right\rceil = \begin{cases} \dfrac{s_f - \ell + t}{2}, & \ell \text{ is even}, \\ \dfrac{s_f - \ell + t + 1}{2}, & \ell \text{ is odd}. \end{cases}$$

These two conditions are equivalent to η^ℓ_t being strictly smaller than 0 and 1, respectively. However, since $s_f + \ell \mid \theta^\ell_t$, the quantity η^ℓ_t must be an integer and have the same parity as ℓ, and we need to ensure only that there are no values of ℓ such that $\eta^\ell_t \leqslant -2$ or $\eta^\ell_t \leqslant -1$. It is easy to check that for t between 0 and 2, $\eta^\ell_t \leqslant -2$ is equivalent to $\ell^2 + (2-t) + 2s_f \leqslant 0$, which has no even solutions, whereas $\eta^\ell_t \leqslant -1$ is equivalent to $\ell^2 + (1-t) + s_f \leqslant 0$, which has no odd solutions.

If $s_f + \ell \nmid \theta^\ell_t$, then by the same argument we conclude that we need to prove that there exists no $\ell > -s_f$ such that

$$\theta^\ell_t < \left\lceil \frac{s_f - \ell + t - 2}{2} \right\rceil = \begin{cases} \dfrac{s_f - \ell + t - 2}{2}, & \ell \text{ is even}, \\ \dfrac{s_f - \ell + t - 1}{2}, & \ell \text{ is odd}. \end{cases}$$

These two conditions lead to the inequalities $\eta_t^\ell < -2$ and $\eta_t^\ell < -1$, which have no solution for t between 0 and 2, as discussed above.

The proof for s_c is similar. In this case, we have that $s_c \mid \frac{k-1}{2}$ if and only if

$$\frac{k-1}{2s_c} = \frac{s_c - 1}{2},$$

and thus that $C_{s_c}^e(k) = 2s_c - 2$. It is easy to show that, for $\ell > -s_c$,

$$C_{s_c+\ell}^e(k) \geqslant \begin{cases} s_c + \ell + 2\left\lfloor \theta^\ell \right\rfloor - 1, & s_c \mid \theta^\ell, \\ s_c + \ell + 2\left\lfloor \theta^\ell \right\rfloor + 1, & s_c \nmid \theta^\ell, \end{cases} \qquad \theta^\ell := \frac{s_c - \ell - 1 + \eta^\ell}{2}, \qquad \eta^\ell := \frac{\ell^2 + \ell}{s_c + \ell}.$$

Therefore, if $s_c \mid \theta^\ell$, we only have to prove that there exists no $\ell > -s_c$ such that

$$\theta^\ell < \left\lceil \frac{s_c - \ell - 1}{2} \right\rceil = \begin{cases} \dfrac{s_c - \ell - 1}{2}, & \ell \text{ is even}, \\ \dfrac{s_c - \ell}{2}, & \ell \text{ is odd}, \end{cases}$$

or, in other words, that $\eta^\ell < 0$ if ℓ is even, and $\eta^\ell < -1$ if ℓ is odd. Both conditions are trivially satisfied, since $\eta^\ell \geqslant 0$ for $|\ell| \geqslant 1$. Finally, if $s_c \nmid \theta^\ell$, we obtain the conditions $\eta^\ell < -1$ if ℓ is even and $\eta^\ell < -2$ if ℓ is odd, both of which clearly satisfy since η^ℓ is nonnegative. □

We are now ready to characterize the optimality of the Paterson–Stockmeyer method for the diagonal Padé approximants to the matrix exponential.

Proposition 4. *A degree $k \in \mathbb{N}^+$ is optimal for the evaluation scheme (20) if and only if $k = 2$ or*

$$k = 2\left\lceil \frac{y}{4} \right\rceil \left(y - 2\left\lfloor \frac{y-1}{4} \right\rfloor \right) + 1, \tag{24}$$

for some $y \in \mathbb{N}$.

Proof. First, note that for k to be optimal, both μ_k^e and μ_k^o must be integer multiples of s, since otherwise, we could add more terms at no cost until both conditions are satisfied. This implies that, if either μ_k^e or μ_k^o is greater than 1, then k must be odd: if it were not, then $s \in \mathbb{N}^+$ could not divide both μ_k^o and $\mu_k^e = \mu_k^o + 1$.

It is easy to show that $k = 2$ is an optimal degree for the evaluation scheme (20). We have that $s = 1$, $\mu_k^o = 0$, and $\mu_k^e = 1$, which gives $C_1^e(2) = 1$, and

$$\frac{2+j}{2(1+\ell)} = \frac{1}{2}\left(1 - \ell + \eta_j^\ell\right), \qquad \eta_j^\ell := \frac{2\ell^2 + j + 1}{1 + \ell}.$$

Since η_j^ℓ is strictly positive, if $1 + \ell \nmid \frac{2+j}{2}$, then $C_{1+\ell}^e(2+j) \geqslant 2 > C_1^e(2)$, whereas if $1 + \ell \mid \frac{2+j}{2}$, then η_j^ℓ must be an integer larger than 2, which again gives $C_{1+\ell}^e(2+j) \geqslant 2 > C_1^e(2)$.

It is convenient to split the expression for k into four cases that allow us to get rid of the floor and ceiling operators in (24). To that end, we note that if $k \equiv \widetilde{t} \pmod 4$, then $k = 2x(2x + t) + 1$, for some $x \in \mathbb{N}$ and $t = \widetilde{t} - 2$.

The three cases $|t| \leqslant 1$ can be addressed together. We have that $s = 2x + t$ or, equivalently, that $x = \frac{s-t}{2}$, and since $\frac{k-1}{2s} = x$, we can conclude that $C_s^e(k) = 4x + t - 1$. Now let $\ell \in \mathbb{Z}$ be such that $\ell > -s$ and let $j \in \mathbb{N}^+$. We have that

$$\frac{k+j-1}{2(s+\ell)} = \frac{1}{2}\left(\frac{s(s-t)+j}{s+\ell}\right) = \frac{1}{2}\left(s - \ell - t + \eta_{t,j}^{\ell}\right), \qquad \eta_{t,j}^{\ell} := \frac{\ell^2 - t\ell + j}{s+\ell}.$$

Note that $\eta_{t,j}^{\ell} > 0$. If $s + \ell \nmid \frac{k+j-1}{2}$, then $C_{s+\ell}^e(k+j) \geqslant 4x + t + 1 > C_s^e(k)$. On the other hand, if $s + \ell \mid \frac{k+k-1}{2}$, then $\eta_{t,j}^{\ell}$ must be a positive integer in order for $\frac{k+j-1}{2(s+\ell)}$ to be integer, which gives that $C_{s+\ell}^e(k+j) = 4x + t > C_s^e(k)$.

Finally we consider the case $t = 2$. From $s = 2x$, we get that $x = \frac{s}{2}$ and $k-1 = s(s+2)$, which gives $C_s^e(k) = 4x + 1$. We have that

$$\frac{k+j-1}{2(s+\ell)} = \frac{1}{2}\left(\frac{s(s+2)+j}{s+\ell}\right) = \frac{1}{2}(s - \ell + 2 + \eta_j^{\ell}), \qquad \eta_j^{\ell} := \frac{\ell^2 - 2\ell + j}{s+\ell}.$$

It is easy to see that η_j^{ℓ} is nonnegative, and in particular that $\eta_j^{\ell} = 0$ only if $j = 1$ and $\ell = 1$. Thus, if $s + \ell \nmid \frac{k+j-1}{2}$, then $C_{s+\ell}^e(k+j) \geqslant 4x + 3 > C_s^e(k)$. When $s + \ell \mid \frac{k+j-1}{2}$, on the other hand, since $s + 1 \nmid \frac{k}{2}$ and η_j^{ℓ} is positive, in particular η_j^{ℓ} must be larger than 1 for $\frac{k+j-1}{2}$ to be an integer multiple of $s + \ell$. Therefore, we have that $C_{s+\ell}^e(k+j) \geqslant 4x + 2 > C_s^e(k)$.

The converse follows from the same argument as that used in the proof of the analogous result in Proposition 3. ∎

In view of Proposition 4, the sequence of optimal degrees for the evaluation scheme (20) is $(a_i^e)_{i \in \mathbb{N}}$, where

$$\begin{aligned} a_0^e &= 1, \\ a_1^e &= 2, \\ a_i^e &= 2\left\lceil \frac{i-1}{4} \right\rceil \left(i - 3\left\lfloor \frac{i-1}{4} \right\rfloor\right) + 1, \qquad i \geqslant 2. \end{aligned} \tag{25}$$

Moreover, we have that $C^e(a_i^e) = i$ and that the diagonal Padé approximant to the matrix exponential of highest order that can be evaluated with i matrix multiplications is that of degree $[a_i^e/a_i^e]$.

5 Conclusion

The scheme (3), which gives rise to the Paterson–Stockmeyer method, and the related evaluation schemes (15) and (20), are customary tools for evaluating truncated Taylor series and diagonal Padé approximants. They all feature a parameter, s, which is usually chosen by approximately solving an optimization problem over the integers. For the evaluation of matrix polynomials, we showed that the Paterson–Stockmeyer choices $s = \lfloor \sqrt{k} \rfloor$ and $s = \lceil \sqrt{k} \rceil$ always minimize the number of matrix multiplications required to evaluate a polynomial of degree k. For the evaluation of diagonal approximants, we gave sufficient conditions for the parameter s to minimize the computational cost of the corresponding evaluation schemes. Tests not reported here suggest that, for all $k \in \mathbb{N}^+$, the choices $s = s_k^r$ in (17) and $s = s_k^e$ in (22) minimize the number of matrix multiplications required by the schemes (15) and (20), respectively, and we believe that exploring this question further might lead to results similar to that in Proposition 1 for the Paterson–Stockmeyer method.

When relying on polynomial or rational approximation to evaluate matrix functions, one is usually interested only in approximants whose order is maximal for a given computational cost. By exploiting the results discussed above, we showed that the sequences of optimal orders (in the sense of Definition 1) for the three evaluation schemes (3), (15), and (20), are (14), (19), and (25), respectively. We wonder whether similar results can be derived for rational functions of any order, and more generally, for schemes that require the evaluation of three or more polynomials of any degree. This will be the subject of future work.

Acknowledgements

The author thanks Stefan Güttel, Nicholas J. Higham, Bruno Iannazzo, Froilán Dopico, and the two anonymous referees for reading early drafts of the manuscript and providing comments that greatly improved the presentation of this work.

References

[1] A. H. Al-Mohy and N. J. Higham, *A new scaling and squaring algorithm for the matrix exponential*, SIAM J. Matrix Anal. Appl., 31 (2009), pp. 970–989.

[2] ——, *Improved inverse scaling and squaring algorithms for the matrix logarithm*, SIAM J. Sci. Comput., 34 (2012), pp. C153–C169.

[3] A. H. Al-Mohy, N. J. Higham, and S. D. Relton, *New algorithms for computing the matrix sine and cosine separately or simultaneously*, SIAM J. Sci. Comput., 37 (2015), pp. A456–A487.

[4] P. Alonso, J. Ibáñez, J. Sastre, J. Peinado, and E. Defez, *Efficient and accurate algorithms for computing matrix trigonometric functions*, J. Comput. Appl. Math., 309 (2017), pp. 325–332.

[5] M. Aprahamian and N. J. Higham, *Matrix inverse trigonometric and inverse hyperbolic functions: Theory and algorithms*, SIAM J. Matrix Anal. Appl., 37 (2016), pp. 1453–1477.

[6] M. Caliari and F. Zivcovich, *On-the-fly backward error estimate for matrix exponential approximation by Taylor algorithm*, J. Comput. Appl. Math., 346 (2019), pp. 532–548.

[7] S. H. Cheng, N. J. Higham, C. S. Kenney, and A. J. Laub, *Approximating the logarithm of a matrix to specified accuracy*, SIAM J. Matrix Anal. Appl., 22 (2001), pp. 1112–1125.

[8] E. Defez, J. Ibáñez, J. Sastre, J. Peinado, and P. Alonso, *A new efficient and accurate spline algorithm for the matrix exponential computation*, J. Comput. Appl. Math., 337 (2018), pp. 354–365.

[9] M. Fasi and N. J. Higham, *An arbitrary precision scaling and squaring algorithm for the matrix exponential*, MIMS EPrint 2018.36, Manchester Institute for Mathematical Sciences, The University of Manchester, UK, 2018.

[10] M. Fasi and N. J. Higham, *Multiprecision algorithms for computing the matrix logarithm*, SIAM J. Matrix Anal. Appl., 39 (2018), pp. 472–491.

[11] S. Güttel and Y. Nakatsukasa, *Scaled and squared subdiagonal Padé approximation for the matrix exponential*, SIAM J. Matrix Anal. Appl., 37 (2016), pp. 145–170.

[12] G. Hargreaves, *Topics in Matrix Computations: Stability and Efficiency of Algorithms*, PhD thesis, University of Manchester, Manchester, England, 2005.

[13] N. J. Higham, *The scaling and squaring method for the matrix exponential revisited*, SIAM J. Matrix Anal. Appl., 26 (2005), pp. 1179–1193.

[14] ——, *Functions of Matrices: Theory and Computation*, Society for Industrial and Applied Mathematics, Philadelphia, PA, USA, 2008.

[15] N. Hoffman, O. Schwartz, and S. Toledo, *Efficient evaluation of matrix polynomials*, Lect. Notes Comput. Sci., (2018), pp. 24–35.

[16] D. Kressner and R. Luce, *Fast computation of the matrix exponential for a Toeplitz matrix*, SIAM J. Matrix Anal. Appl., 39 (2018), pp. 23–47.

[17] M. S. Paterson and L. J. Stockmeyer, *On the number of nonscalar multiplications necessary to evaluate polynomials*, SIAM J. Comput., 2 (1973), pp. 60–66.

[18] J. Sastre, *Efficient evaluation of matrix polynomials*, Linear Algebra Appl., 539 (2018), pp. 229–250.

[19] J. Sastre, J. Ibáñez, P. Alonso, J. Peinado, and E. Defez, *Two algorithms for computing the matrix cosine function*, J. Comput. Appl. Math., 312 (2017), pp. 66–77.

[20] J. Sastre, J. Ibáñez, and E. Defez, *Boosting the computation of the matrix exponential*, Appl. Math. Comput., 340 (2019), pp. 206–220.

[21] J. Sastre, J. Ibáñez, E. Defez, and P. Ruiz, *New scaling-squaring Taylor algorithms for computing the matrix exponential*, SIAM J. Matrix Anal. Appl., 37 (2015), pp. A439–A455.

[22] C. Van Loan, *A note on the evaluation of matrix polynomials*, IEEE Trans. Automat. Control, 24 (1979), pp. 320–321.