SIAM J. MATRIX ANAL. APPL.
Vol. 22, No. 4, pp. 1126–1135

EVALUATING PADÉ APPROXIMANTS OF THE MATRIX LOGARITHM*

NICHOLAS J. HIGHAM†

Abstract. The inverse scaling and squaring method for evaluating the logarithm of a matrix takes repeated square roots to bring the matrix close to the identity, computes a Padé approximant, and then scales back. We analyze several methods for evaluating the Padé approximant, including Horner's method (used in some existing codes), suitably customized versions of the Paterson–Stockmeyer method and Van Loan's variant, and methods based on continued fraction and partial fraction expansions. The computational cost, storage, and numerical accuracy of the methods are compared. We find the partial fraction method to be the best method overall and illustrate the benefits it brings to a transformation-free form of the inverse scaling and squaring method recently proposed by Cheng, Higham, Kenney, and Laub [*SIAM J. Matrix Anal. Appl.*, 22 (2001), pp. 1112–1125]. We comment briefly on how the analysis carries over to the matrix exponential.

Key words. matrix logarithm, Padé approximation, inverse scaling and squaring method, Horner's method, Paterson–Stockmeyer method, continued fraction, partial fraction expansion

AMS subject classification. 65F30

PII. S0895479800368688

1. Introduction.

Any nonsingular matrix $A \in \mathbb{R}^{n\times n}$ having no eigenvalues on the negative real axis has a real logarithm, that is, a real matrix W such that $e^W = A$ [12, Thm. 6.4.15], [13]. Among all real logarithms there is a unique one whose eigenvalues have imaginary parts lying strictly between $-\pi$ and π; this is the principal logarithm, which we denote by $\log A$.

One of the most effective ways to compute $\log A$ is by inverse scaling and squaring combined with Padé approximation. The idea is to compute $Z = A^{1/2^k}$, with k large enough so that Z is close to the identity, and then to compute a Padé approximant of $\log Z$. The logarithm of A is then obtained from the identity [5], [13]

$$\log A = 2^k \log A^{1/2^k}. \tag{1.1}$$

We will refer to this method as the inverse scaling and squaring method. The method was proposed by Kenney and Laub [13], who suggested obtaining the square roots by computing a Schur decomposition of A and then taking square roots of the triangular Schur factor, using the methods of [2], [10]. Recently, Cheng, Higham, Kenney, and Laub [5] developed a transformation-free form of the inverse scaling and squaring method in which the square roots are approximated using a matrix iteration and certain parameters are chosen dynamically to minimize the computational cost subject to achieving a specified accuracy. This new version can be implemented using only matrix multiplication, LU factorization, and matrix inversion. The methods of [5] and [13] must evaluate a diagonal Padé approximant

$$r_m(x) = p_m(x)/q_m(x) = \log(1+x) + O(x^{2m+1})$$

*Received by the editors March 7, 2000; accepted for publication (in revised form) by D. Calvetti November 4, 2000; published electronically March 13, 2001.
http://www.siam.org/journals/simax/22-4/36868.html

†Department of Mathematics, University of Manchester, Manchester, M13 9PL, England (higham@ma.man.ac.uk, http://www.ma.man.ac.uk/~higham/). This work was supported by Engineering and Physical Sciences Research Council grant GR/L94314 and a Royal Society Leverhulme Trust Senior Research Fellowship.

at a matrix argument X with $\|X\| < 1$. Here, p_m and q_m are polynomials of degree m whose coefficients are known, and $m \leq 16$ in practice. The norm is any subordinate matrix norm. The question we consider here is how to evaluate the Padé approximant for a given m.

Evaluation of $r_m(X)$ by applying Horner's method to the numerator and denominator polynomials is the most obvious approach and was used in [13] and during the initial work of [5]. However, several alternatives are available and a hint that the use of a different representation of the rational r_m may be profitable is given by Dieci, Morini, and Papini [7], who comment that "for diagonal Padé approximants, it might instead be more desirable to pass to their quadrature formula equivalent ... to avoid ill-conditioning in the denominator of the rational function."

In the next section we describe the Paterson–Stockmeyer method for evaluating the p_m/q_m form and Van Loan's variant of it, together with methods based on continued fraction and partial fraction representations. We count the operations and storage required. The effect of rounding errors on the methods is described in section 3 and numerical experiments are given in section 4. We finish, in section 5, with a recommendation on the choice of method and a brief discussion of how the analysis carries over to the evaluation of Padé approximants of the matrix exponential.

2. Methods of evaluation. We consider methods of evaluating the Padé approximant $r_m(X)$ at $X \in \mathbb{R}^{n\times n}$ based on three representations. We note that in several of our equations matrices can be reordered, since rational functions of a matrix X commute, but such changes have no effect on the computational cost or accuracy. When counting storage we will include that for X and $r_m(X)$ and assume that X cannot be overwritten.

2.1. Rational evaluation. In this method the polynomials $p_m(X)$ and $q_m(X)$ are evaluated and $Y = r_m(X)$ is computed by solving $q_m Y = p_m$. We consider three possibilities. First, Horner's method can be used for the polynomial evaluations, as in [5], [13]. Thus

$$p_m(X) = \sum_{k=0}^{m} b_k X^k \tag{2.1}$$

is evaluated by

$S_m = b_m X + b_{m-1} I$
for $j = m-2\colon -1\colon 0$
  $S_j = X S_{j+1} + b_j$
end
$p_m = S_0$

and similarly for $q_m(X)$. The total cost is $2(m-1)M + I$, where we denote by M the cost of a matrix multiplication and I the cost of a matrix inversion or of solving a linear system with n right-hand sides.

Instead of using Horner's method we could explicitly compute the powers X^2, ..., X^m and evaluate p_m and q_m as linear combinations of the powers, at a cost of $(m-1)M + I$ (note that if the polynomial coefficients were matrices rather than scalars, this method would cost 50 percent more than Horner's method). However, a potentially greater reduction in cost over Horner's method is offered by a method of Paterson and Stockmeyer [9, sect. 11.2.4], [16]. It writes p_m as

$$p_m(X) = \sum_{k=0}^{r} B_k \cdot (X^s)^k, \qquad r = \mathrm{floor}(m/s), \tag{2.2}$$

where s is an integer parameter and

$$B_k = \begin{cases} b_{sk+s-1}X^{s-1} + \cdots + b_{sk+1}X + b_{sk}I, & k = 0\colon r-1, \\ b_m X^{m-sr} + \cdots + b_{sr+1}X + b_{sr}I, & k = r. \end{cases}$$

The powers $X^2, \ldots, X^s$ are computed, then the B_k, and finally (2.2) is evaluated by Horner's method. The cost of evaluating p_m is

$$(s + r - 1 - f(s,m))M, \qquad f(s,m) = \begin{cases} 1 & \text{if } s \text{ divides } m, \\ 0 & \text{otherwise.} \end{cases} \tag{2.3}$$

The cost of evaluating r_m by the Paterson–Stockmeyer method is $\big(s + 2r - 1 - 2f(s,m)\big)M + I$, which is approximately minimized[1] by $s = \sqrt{2m}$. We therefore take for s whichever of floor($\sqrt{2m}$) and ceil($\sqrt{2m}$) yields the smaller operation count. Unfortunately, the method requires $(s+2)n^2$ elements of storage. This can be reduced to $4n^2$ by computing p_m and q_m a column at a time, as shown by Van Loan [18], though the cost of evaluating r_m then increases to $(2s + 2r - 3 - 2f(s,m))M + I$. Since $s = \sqrt{m}$ approximately minimizes the cost of Van Loan's variant we take for s whichever of floor($\sqrt{m}$) or ceil($\sqrt{m}$) yields the smaller operation count.

2.2. Continued fraction. The Padé approximant r_m to $\log(1+x)$ has the continued fraction expansion [1, p. 174]

$$r_m(x) = \cfrac{c_1 x}{1 + \cfrac{c_2 x}{1 + \cfrac{c_3 x}{\cdots \cfrac{}{1 + \cfrac{c_{2m-1}x}{1 + c_{2m}x}}}}},$$

where

$$c_1 = 1, \qquad c_{2j} = \frac{j}{2(2j-1)}, \quad c_{2j+1} = \frac{j}{2(2j+1)}, \qquad j = 1, 2, \ldots.$$

This expansion can be evaluated at the matrix X in two ways. Top-down evaluation (which converts the continued fraction to rational form) is effected by the recurrence [3]

$A_1 = c_1 X$, $B_1 = I$, $A_2 = c_1 X$, $B_2 = I + c_2 X$
for $j = 3\colon 2m$
  $A_j = A_{j-1} + c_j X A_{j-2}$
  $B_j = B_{j-1} + c_j X B_{j-2}$
end
$r_m = A_{2m} B_{2m}^{-1}$.

The cost is $2(2m-2)M + I$.

Using bottom-up evaluation, $r_m(X)$ is evaluated by

$Y_{2m} = c_{2m} X$
for $j = 2m-1\colon -1\colon 1$
  Solve $(I + Y_{j+1})Y_j = c_j X$ for Y_j.
end
$r_m = Y_0$.

[1] In [7] $s = \sqrt{m}$ is chosen, which minimizes the cost of evaluating p_m or q_m alone, but not both together.

TABLE 1
Cost of evaluating $r_m(X)$. The optimal s are described in the text and f is defined in (2.3).

Method	Computational cost	Storage
Horner	$2(m-1)M + I$	$3n^2$
Paterson–Stockmeyer	$\big(s + 2r - 1 - 2f(s,m)\big)M + I \gtrsim (2\sqrt{2}\sqrt{m} - 1)M + I$	$(s+2)n^2$
Van Loan	$(2s + 2r - 3 - 2f(s,m))M + I \gtrsim (4\sqrt{m} - 3)M + I$	$4n^2$
Continued fraction	top-down: $2(2m-2)M + I$	$5n^2$
	bottom-up: $(2m-1)I$	$3n^2$
Partial fraction	mI	$3n^2$

This evaluation costs $(2m-1)I$.

Although the top-down evaluation is computationally expensive, it merits further consideration as it is well suited to situations in which the whole sequence $r_1(X)$, $r_2(X)$, ..., needs to be evaluated; in this case the bottom-up evaluation has to start afresh each time.

2.3. Partial fraction. The Padé approximant r_m can be expressed in partial fraction form as

$$r_m(x) = \sum_{j=1}^{m} \frac{\alpha_j^{(m)} x}{1 + \beta_j^{(m)} x}, \tag{2.4}$$

where the $\alpha_j^{(m)}$ are the weights and the $\beta_j^{(m)}$ the nodes of the m-point Gauss–Legendre quadrature rule on $[0,1]$ [7, Thm. 4.3]. The connection with quadrature stems from the integral representation

$$\log(1+x) = x \int_0^1 \frac{dt}{1+xt}.$$

Codes for computing the $\alpha_j^{(m)}$ and $\beta_j^{(m)}$ are given in [6, App. 2], [8], [17, sect. 4.5]; these computations are of negligible cost if $m \ll n$ and the coefficients can of course be precomputed and stored. The cost of evaluating (2.4) at the matrix X is mI. An advantage of (2.4) is its suitability for parallel evaluation; see [4] for a discussion and extensive bibliography on parallel evaluation of matrix partial fraction expansions.

Table 1 summarizes the cost of the methods. The Paterson–Stockmeyer and Van Loan methods clearly require the least computation for large m, since their costs grow as $\sqrt{m}$ for the optimal s rather than linearly with m as for the other methods. In fact, both methods are more efficient than Horner's method and the continued fraction methods for all m, as shown by Figure 1, in which the total number of matrix multiplications and inversions is plotted against m. For the range of m of interest the partial fraction method is competitive with the $O(\sqrt{m})$ methods.

The sensitivity of the methods to rounding errors is another important factor in the choice of method and we examine it in the next section.

3. Effects of rounding errors. Before beginning the error analysis we state some properties of $r_m = p_m/q_m$ that will be needed [14]. First, $q_m(x)$ is an increasing, positive function of x for $x > -1$. Second, the coefficients of p_m and q_m (with the normalization $q_m(0) = 1$) are nonnegative. To illustrate, in unnormalized form,

$$r_3(x) = \frac{60x + 60x^2 + 11x^3}{60 + 90x + 36x^2 + 3x^3}.$$

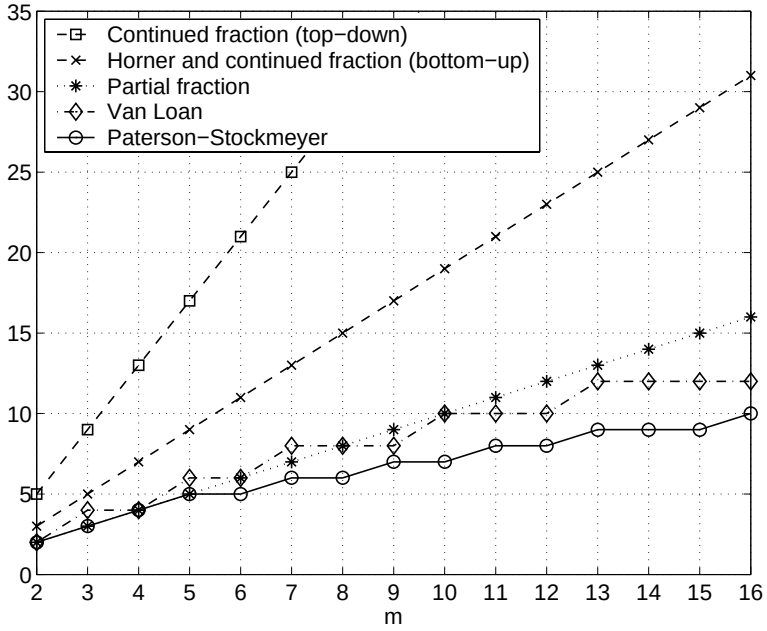

FIG. 1. *Total number of matrix multiplications and inversions to evaluate* $r_m(X)$.

It is straightforward to derive an error bound for Horner's method for evaluating a polynomial p_m of the form (2.1). The following result is a generalization of one for the scalar case [11, sect. 5.1]. We use the standard model of floating point arithmetic with unit roundoff u [11, sect. 2.2].

LEMMA 3.1. *The computed polynomial* $\widehat{p}_m$ *from Horner's method applied to* (2.1) *satisfies*

$$\|\widehat{p}_m - p_m\| \le m(n+1)u\,\widetilde{p}_m(\|X\|) + O(u^2),$$

where $\widetilde{p}_m(X) = \sum_{i=0}^m |b_k| X^k$.

The bound in the lemma is not the sharpest that can be obtained, but it is adequate for our application, in which $\|X\| < 1$.

In view of the lemma, the system that is solved to determine $Y = r_m(X)$ is

$$(q_m + \Delta Q)Y = p_m + \Delta P,$$
$$\|\Delta Q\| \le m(n+1)u q_m(\|X\|) + O(u^2), \quad \|\Delta P\| \le m(n+1)u p_m(\|X\|) + O(u^2),$$

where we have used the fact that our particular p_m and q_m have nonnegative coefficients. Assuming the system is solved by a stable method, the overall forward error bound will be of the form

$$\frac{\|Y - \widehat{Y}\|}{\|Y\|} \le d_1(m,n)u\kappa(q_m)\eta(X) + O(u^2), \tag{3.1}$$

where $d_j(m,n)$ denotes a constant depending on m and n and η is given by

$$\eta_1(X) = \left(\frac{p_m(\|X\|)}{\|q_m\|\|Y\|} + \frac{q_m(\|X\|)}{\|q_m\|}\right) \ge 1. \tag{3.2}$$

Kenney and Laub [14] show that

$$\kappa(q_m(X)) \le \frac{q_m(\|X\|)}{q_m(-\|X\|)}, \quad \|X\| < 1, \tag{3.3}$$

and this bound is easily evaluated for particular m and x.

For the Paterson–Stockmeyer and Van Loan methods it is not difficult to show that a bound of the same form as that in Lemma 3.1 holds, but with different constants. Therefore (3.1) applies to these methods too.

Next, we consider top-down evaluation of the continued fraction. We can express the recurrence for the B_j as

$$\begin{aligned}\begin{bmatrix} B_j \\ B_{j-1} \end{bmatrix} &= \begin{bmatrix} I & c_jX \\ I & 0 \end{bmatrix} \begin{bmatrix} B_{j-1} \\ B_{j-2} \end{bmatrix} \\ &= \begin{bmatrix} I & c_jX \\ I & 0 \end{bmatrix} \cdots \begin{bmatrix} I & c_2X \\ I & 0 \end{bmatrix} \begin{bmatrix} I \\ I \end{bmatrix}.\end{aligned}$$

From a standard error bound for matrix multiplication [11, Prob. 3.8] we have

$$\|\widehat{B}_{2m} - B_{2m}\| \le d_2(m,n)u \prod_{j=2}^{2m} (1 + c_j\|X\|).$$

Similarly,

$$\|\widehat{A}_{2m} - A_{2m}\| \le d_3(m,n)uc_1\|X\| \prod_{j=3}^{2m} (1 + c_j\|X\|).$$

Therefore (3.1) holds with η given by

$$\eta_2(X) = \frac{\prod_{j=3}^{2m}(1 + c_j\|X\|)}{\|q_m\|} \left(\frac{c_1\|X\|}{\|Y\|} + 1 + c_2\|X\| \right). \tag{3.4}$$

For the bottom-up evaluation of the continued fraction, in which Y_j is computed by solving $(I + Y_{j+1})Y_j = c_jX_j$, errors in Y_{j+1} can potentially be magnified by $\kappa(I + Y_{j+1})$ in passing to Y_j. Therefore it is essential that $\max_j \kappa(I+Y_j)$ is small. Assuming $\|Y_j\| < 1$, we have

$$\kappa(I + Y_j) \le \frac{1 + \|Y_j\|}{1 - \|Y_j\|}, \tag{3.5}$$

and the $\|Y_j\|$ satisfy, with $\|Y_{2m}\| = c_{2m}\|X\|$,

$$\|Y_j\| \le \frac{|c_j|\|X\|}{1 - \|Y_{j+1}\|}, \qquad j = 2m-1\colon -1\colon 1. \tag{3.6}$$

For a particular bound on $\|X\|$ we can therefore compute a bound on $\kappa(I + Y_j)$ and the overall error will be roughly bounded by $\max_j \kappa(I + Y_j)u$.

For the partial fraction method the accuracy is again dependent on the condition of the linear systems that are solved, and we expect the normwise relative error to be bounded approximately by $d_4(m,n)u\phi$, where

$$\phi = \max_j \left[\alpha_j^{(m)} \kappa(I + \beta_j^{(m)} X) \right] \tag{3.7}$$

TABLE 2

Terms from error analysis. $\epsilon(m,\|X\|)$ is defined in (3.8); η_1 in (3.2) and η_2 in (3.4) are terms from the Horner and top-down continued fraction methods; the bound for $\kappa(q_m(X))$ is from (3.3) and that for $\kappa(I+Y_j)$ from (3.5) and (3.6); ϕ for the partial fraction method is defined in (3.7).

			Approx. to		Bounds for		
$\|X\|$	m	$\epsilon(m,\|X\|)$	$\eta_1(X)$	$\eta_2(X)$	$\kappa(q_m(X))$	$\max_j \kappa(I+Y_j)$	ϕ
tol $= 2^{-24} \approx 6\times 10^{-8}$							
0.99	16	7.7e-3	6.8e2	1.9e3	4.5e10	8.3e0	1.8e0
0.95	16	1.9e-6	5.6e2	1.5e3	1.7e9	5.3e0	7.9e-1
0.90	14	3.5e-8	2.2e2	4.8e2	1.3e7	4.1e0	6.2e-1
0.75	8	4.7e-8	1.9e1	2.6e1	1.0e3	2.7e0	5.8e-1
0.50	5	2.3e-8	5.3e0	5.8e0	1.4e1	1.8e0	5.4e-1
0.25	3	5.7e-8	2.7e0	2.7e0	2.1e0	1.3e0	5.7e-1
0.10	3	5.1e-11	2.3e0	2.3e0	1.4e0	1.1e0	4.9e-1
tol $= 2^{-53} \approx 1\times 10^{-16}$							
0.90	16	2.9e-9	4.4e2	1.1e3	1.4e8	4.1e0	5.5e-1
0.75	16	2.6e-12	2.1e2	4.1e2	1.1e6	2.7e0	3.1e-1
0.50	16	3.4e-14	5.5e1	7.7e1	4.7e3	1.8e0	1.8e-1
0.25	7	0.0e0	4.2e0	4.4e0	5.9e0	1.3e0	2.7e-1
0.10	5	1.4e-17	2.5e0	2.5e0	1.7e0	1.1e0	3.1e-1
Largest $\|X\|$, m permitted in earlier version of [5].							
0.50	8	5.9e-13	1.0e1	1.2e1	6.9e1	1.8e0	3.5e-1

(note that $\alpha_j^{(m)} > 0$ and $\sum_j \alpha_j^{(m)} = 1$). We have

$$\kappa(I+\beta_j^{(m)}X) \le \frac{1+|\beta_j^{(m)}|\|X\|}{1-|\beta_j^{(m)}|\|X\|},$$

and since $\beta_j^{(m)} \in (0,1)$ the condition number is guaranteed to be small provided that $\|X\|$ is not too close to 1.

The two key parameters to consider when investigating the accuracy of the methods are the degree m of the Padé approximant and the norm of the matrix argument, X. In practice, these parameters are chosen so that $r_m(X)$ approximates $\log(I+X)$ to the desired accuracy, with either a fixed choice of m [7], [13] or a dynamic choice intended to minimize the overall computation time [5]. For a given X with $\|X\| < 1$ the bound

$$\|r_m(X) - \log(I+X)\| \le |r_m(-\|X\|) - \log(1-\|X\|)| =: \epsilon(m,\|X\|) \tag{3.8}$$

from [14] enables a suitable m to be determined.

In Table 2 we compare approximations to and bounds for the quantities arising in our analysis for a range of $\|X\|$ and m, with m chosen as the smaller of 16 and the minimal value for which $\epsilon(m,\|X\|) \le$ tol, where tol is a tolerance. The values of tol used for the table correspond to single and double precision accuracy in the Padé approximant, and for the η values we approximated $\|Y\| = \|\log(I+X)\| \approx \|X\|$ and $\|q_m(X)\| \approx q_m(0) = 1$.

The table implies that the effect of rounding errors on the bottom-up evaluation of the continued fraction and the partial fraction methods is negligible for all m and $\|X\|$ of interest. But Horner's method, the Paterson–Stockmeyer method, Van Loan's method, and the continued fraction evaluated top-down are all potentially unstable unless $\|X\|$ is much less than 1, as the denominator polynomial q_m has a condition number bound that grows rapidly with $\|X\|$ and the η terms from the error bounds

TABLE 3
Normwise relative errors. The pairs $(\|X\|, m)$ *correspond to those in Table* 2.

			Paterson–		Continued fraction		Partial
$\|X\|$	m	Horner	Stockmeyer	Van Loan	top-down	bottom-up	fraction
0.99	16	6.7e-12	3.5e-11	1.3e-11	2.9e-12	1.5e-16	4.2e-16
0.95	16	1.4e-15	3.0e-15	2.7e-15	1.2e-14	1.4e-16	3.1e-16
0.90	14	9.4e-14	5.9e-14	4.0e-14	7.9e-14	9.1e-17	2.0e-16
0.75	8	5.9e-16	1.0e-15	1.0e-15	1.6e-15	1.9e-16	3.7e-16
0.50	5	2.7e-16	2.3e-16	1.8e-16	3.9e-16	1.0e-16	5.7e-17
0.25	3	1.8e-16	7.9e-17	2.6e-16	1.7e-16	6.1e-17	4.1e-16
0.10	3	9.8e-17	1.0e-16	1.0e-16	9.8e-17	1.7e-16	3.2e-16
0.90	16	2.8e-13	2.8e-13	9.9e-14	2.1e-13	9.1e-17	2.6e-16
0.75	16	6.0e-15	1.3e-14	8.6e-15	1.1e-14	1.7e-16	3.4e-16
0.50	16	1.7e-15	1.2e-14	6.0e-15	1.6e-14	1.4e-16	4.1e-16
0.25	7	1.5e-16	1.8e-16	1.9e-16	4.3e-16	1.4e-16	4.5e-16
0.10	5	5.2e-17	1.4e-16	4.0e-17	2.8e-16	8.1e-17	8.6e-17

also become significant for $\|X\|$ close to 1. The last line of the table justifies a restriction on $\|X\|$ and m used in an earlier version of [5] in conjunction with Horner evaluation of r_m.

In the next section we check the actual errors via numerical experiments.

4. Numerical experiments. We report numerical experiments carried out in MATLAB, for which $u = 2^{-53} \approx 1 \times 10^{-16}$.

First we test the predictions from the analysis of the previous section. For random 4×4 matrices X with elements from the normal $N(0,1)$ distribution we computed the normwise relative errors $\|\widehat{Y} - Y\|_2 / \|Y\|_2$ in $Y = r_m(X)$ for a range of values of $\|X\|_2$ and m corresponding to Table 2. The "exact" logarithm was obtained using the variable precision arithmetic of MATLAB's Symbolic Math Toolbox. The results are shown in Table 3.

The results confirm that the Horner, Paterson–Stockmeyer, Van Loan, and top-down continued fraction methods do indeed suffer instability when $\|X\|$ is close to 1 and m is large, though the level of instability is much less than the bounds for $\kappa(q_m(X))$ in Table 2 would suggest. The actual $\kappa(q_m(X))$ values in this experiment are less than the square root of the bounds, showing that the bound (3.3) can be very weak. As expected, the bottom-up continued fraction and partial fraction methods give perfect accuracy.

Next we illustrate how the choice of method for evaluating the Padé approximant can affect the efficiency of Cheng, Higham, Kenney, and Laub's version of the inverse scaling and squaring method [5]. The implementation in [5] uses the partial fraction expansion with the restrictions that $\|X\| \le 0.99$ and $m \le 16$. An earlier implementation used Horner's method with the stronger restrictions that $\|X\| \le 1/2$ and $m \le 8$. In view of our analysis in the previous section and the value of ϕ in the first line of Table 2 these two implementations should have similar accuracy properties. We used both implementations to compute the logarithm of the 7×7 Frank matrix (MATLAB's `gallery('frank',7)`). The results are shown in Table 4 for two choices of tolerance in the method corresponding to approximation of the logarithm to single precision and double precision accuracy (all computations are carried out in double precision arithmetic). The partial fraction-based implementation is about 10 percent more efficient than the Horner-based implementation in this example. The improvement accrues from the algorithm being able to take fewer square roots and use a higher degree Padé approximant, as well as from the more efficient evaluation

TABLE 4
Comparison of current and earlier implementations of method from [5]. *"Roots" is the number of square roots, "Cost" the total number of matrix multiplications and matrix inversions, and m the degree of Padé approximant chosen.*

	$\text{tol} = 2^{-24}$			$\text{tol} = 2^{-53}$		
	Roots	Cost	Degree m	Roots	Cost	Degree m
Earlier (Horner)	9	63	5	9	93	8
Current (partial fraction)	8	58	8	9	86	8

of the Padé approximant.

5. Conclusions, and comments on the matrix exponential. We have analyzed alternatives to Horner's method for evaluating Padé approximants to the matrix logarithm. All but two of the alternatives are less expensive than Horner's method and the bottom-up continued fraction method and the partial fraction method have more favorable accuracy properties. Based on operation counts the choice narrows down to the Paterson–Stockmeyer method, Van Loan's version of it, and partial fraction expansion. For the degrees m of practical interest ($m \le 16$), the methods have similar computational cost, but the Paterson–Stockmeyer and partial fraction methods are rich in level 3 BLAS operations whereas Van Loan's method is inherently level 2 BLAS-based. If storage of size $(\sqrt{2m}+2)n^2$ is not available then the Paterson–Stockmeyer method must be ruled out. The partial fraction method has the advantage of being readily parallelizable and of allowing $\|X\|$ to be much closer to 1 without any loss of stability. Therefore the partial fraction expansion emerges as the best overall method.

In special cases a different choice may be appropriate. For example, if matrix multiplication is significantly faster than matrix inversion, as may be the case on certain high-performance machines, if sufficient storage is available, and if $\|X\|$ can be kept significantly less than 1, the Paterson–Stockmeyer method may be the most attractive choice.

An investigation similar to that given here can be done for the matrix exponential. Padé approximants $r_m = p_m/q_m$ of the matrix exponential e^A need to be evaluated in the scaling and squaring method, which approximates $e^{A/2^k}$ by $r_m(A/2^k)$ in the expression $e^A = (e^{A/2^k})^{2^k}$, where k is chosen so that $\|A/2^k\| \le 1$ [19] or $\|A/2^k\| \le 1/2$ [15], [9, sect. 11.3]. We briefly summarize some pertinent facts concerning the evaluation of $r_m(A/2^k)$. The coefficients $\alpha_j^{(m)}$ and $\beta_j^{(m)}$ in the partial fraction expansion (2.4) of r_m are not known explicitly, and the $\alpha_j^{(m)}$ can be very large [4], leading to numerical instability in the evaluation of the expansion. However, the techniques of [4] can be used to obtain an incomplete partial fraction expansion with suitably bounded coefficients. Ill conditioning of the denominator polynomial q_m is not an issue, as $\kappa(q_m(B)) < 5$ for $\|B\| \le 1$ [19, Thm. 1]. Finally, $q_m(X) = p_m(-X)$, and advantage can be taken of this when applying the Paterson–Stockmeyer and Van Loan methods. For the matrix exponential, then, the Paterson–Stockmeyer and Van Loan methods have the advantage over the partial fraction expansion except, possibly, in a parallel computing context.

Acknowledgments. Charlie Kenney suggested the possibility of using the continued fraction and partial fraction representations to evaluate r_m during our work on [5]. I thank Peter Graves-Morris for helpful comments on the manuscript.

REFERENCES

[1] G. A. Baker, Jr. and P. Graves-Morris, *Padé Approximants*, 2nd ed., Encyclopedia Math. Appl., Cambridge University Press, Cambridge, UK, 1996.

[2] A. Björck and S. Hammarling, *A Schur method for the square root of a matrix*, Linear Algebra Appl., 52/53 (1983), pp. 127–140.

[3] G. Blanch, *Numerical evaluation of continued fractions*, SIAM Rev., 6 (1964), pp. 383–421.

[4] D. Calvetti, E. Gallopoulos, and L. Reichel, *Incomplete partial fractions for parallel evaluation of rational matrix functions*, J. Comput. Appl. Math., 59 (1995), pp. 349–380.

[5] 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.

[6] P. J. Davis and P. Rabinowitz, *Methods of Numerical Integration*, 2nd ed., Academic Press, Orlando, FL, 1984.

[7] L. Dieci, B. Morini, and A. Papini, *Computational techniques for real logarithms of matrices*, SIAM J. Matrix Anal. Appl., 17 (1996), pp. 570–593.

[8] W. Gautschi, *Algorithm* 726: *ORTHPOL—A package of routines for generating orthogonal polynomials and Gauss-type quadrature rules*, ACM Trans. Math. Software, 20 (1994), pp. 21–62.

[9] G. H. Golub and C. F. Van Loan. *Matrix Computations*, 3rd ed., Johns Hopkins University Press, Baltimore, MD, 1996.

[10] N. J. Higham, *Computing real square roots of a real matrix*, Linear Algebra Appl., 88/89 (1987), pp. 405–430.

[11] N. J. Higham, *Accuracy and Stability of Numerical Algorithms*, SIAM, Philadelphia, 1996.

[12] R. A. Horn and C. R. Johnson, *Topics in Matrix Analysis*, Cambridge University Press, London, 1991.

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

[14] C. Kenney and A. J. Laub, *Padé error estimates for the logarithm of a matrix*, Internat. J. Control, 50 (1989), pp. 707–730.

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

[16] 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.

[17] W. H. Press, S. A. Teukolsky, W. T. Vetterling, and B. P. Flannery, *Numerical Recipes in FORTRAN: The Art of Scientific Computing*, 2nd ed., Cambridge University Press, London, 1992.

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

[19] R. C. Ward, *Numerical computation of the matrix exponential with accuracy estimate*, SIAM J. Numer. Anal., 14 (1977), pp. 600–610.