ROCKY MOUNTAIN
JOURNAL OF MATHEMATICS
Volume 6, Number 2, Spring 1976

ON THE CONVERGENCE OF AN ALGORITHM FOR RATIONAL CHEBYSHEV APPROXIMATION*

RICHARD FRANKE

ABSTRACT. An algorithm for rational Chebyshev approximation based on computing the zeros of the error curve was investigated. At each iteration the proposed zeros are corrected by changing them toward the abscissa of the adjacent extreme of largest magnitude. The algorithm is formulated as a numerical solution of a certain system of ordinary differential equations. Convergence is obtained by showing the system is asymptotically stable at the zeros of the best approximation. With an adequate initial guess, the algorithm has never failed for functions which have a standard error curve.

1. **Introduction.** We will consider the problem of computing the best approximation to a continuous function $f(x)$ by a rational function $R_{mn} = \sum_{i=0}^{m} a_i x^i / \sum_{j=0}^{n} b_j x^j$, where the measure of the error is the weighted sup norm,

$$\|f - R_{mn}\| = \sup_{a \leqq x \leqq b} \left| \frac{f(x) - R_{mn}(x)}{w(x)} \right| ,$$

$w(x) > 0$ on $[a, b]$. A number of methods have been proposed, and a brief description and numerical comparison of some of them is given by Lee and Roberts [8].

We have implemented and analyzed an algorithm for computing the best rational approximation. This algorithm [7] has not yet appeared in the literature. The convergence rate is linear, and although the algorithm is relatively slow compared to some others, the algorithm was successful in instances where others, such as the Remes algorithm and Maehly's second method, fail. For functions which have a standard error curve, the algorithm has never failed to converge to the best approximation, provided the initial approximant did not have a pole in $[a, b]$.

The algorithm is formulated as the numerical solution of a system of $m + n + 1$ differential equations. The dependent variables are points of zero error in a proposed approximation. When there are $m + n + 2$ alternations in the error curve, the system of equations is at a rest point. Under the assumption of a relatively mild hypothesis, the algorithm is proved convergent by showing that the system of equations is asymptotically stable at that rest point.

Received by the editors on January 10, 1975.
*This research was supported by the Foundation Research Program.

2. **A method for computation of R_{mn}*.** We assume that the best approximation $R_{mn}^* = P^*/Q^*$ exists, and is in reduced form. The basis for several algorithms, such as that of Remes (see, for example [3]) and Maehly's second method [9] is the following characterization theorem, which can be found in many texts, e.g., Cheney [2].

THEOREM 1. *In order that the irreducible rational function P/Q be a best approximation to f of the form R_{mn}, it is necessary and sufficient that the error have at least* $2 + \max(m + \deg Q, n + \deg P)$ *alternations.*

As with Maehly's second algorithm, we assume that the error curve is "standard" in that it has exactly $K = m + n + 1$ points of zero error and $K + 1$ alternations. If the points of zero error are known, the best approximation can be computed as the rational function of the form R_{mn} which interpolates to the value of f at those points.

Let $z_1^* < z_2^* < \cdots < z_K^*$ be the points of zero error for

$$E^*(x) = \frac{f(x) - R_{mn}^*(x)}{w(x)}.$$

Let $Z = (z_1, \cdots, z_K)$ be an approximation to $Z^* = (z_1^*, \cdots, z_K^*)$. Let $R_{mn}(x)$ be the rational function which interpolates to f at $z_1, \cdots, z_K$. Let $z_0 = a$, $z_{K+1} = b$, and define $N_k = \sup_{z_{k-1} \leqq x \leqq z_k} |E(x)|$, where $E(x) = [f(x) - R_{mn}(x)]/w(x)$. If Z is close to Z^*, R_{mn} exists, and the N_k are finite. Let x_k be the point in $[z_{k-1}, z_k]$ such that $|E(x_k)| = N_k$, $k = 1, \cdots, K + 1$. For Z close to Z^* these points must be unique.

Any of the variables with superscript asterisks will denote that variable for the best approximation. Note that $N_{k+1}^* - N_k^* = 0$, $k = 1, 2, \cdots, K$. Under our assumptions, if $N_{k+1} - N_k = 0$, $k = 1, 2, \cdots, K$, then (at least for Z in some neighborhood of Z^*) $Z = Z^*$, although in general this is not true.

In a talk given at the joint SIAM-MAA meeting for the Northern California sections in February 1972, Dr. Milton W. Green of the Stanford Research Institute discussed an algorithm based on the above ideas [7]. Given an initial guess Z at Z^*, one computes R_{mn}, and then $N_1, \cdots, N_{K+1}$. The value of each z_k is then corrected by changing it so that its new value is nearer to x_{k+1}, or x_k, as $N_{k+1} - N_k$ is positive or negative, respectively. That is, z_k is changed toward the point of largest (in magnitude) error in the interval $[z_{k-1}, z_{k+1}]$. Dr. Green reported that he had had good success with the algorithm.

In an attempt to systematize the method and to make it amenable to analysis for its convergence properties, we considered the basic idea in the following form. We formulated the method as a continuous (in

the corrections to the z_k) rather than a discrete problem. Consider the system of ordinary differential equations,

$$\dot{z}_k = N_{k+1} - N_k,\ k = 1, \cdots, K,$$

where z_k and N_k are as defined previously, and $Z = Z^{(0)}$ at $t = 0$ is an approximation to Z^*, used as the initial condition. This system, when solved by Euler's method, yields an algorithm similar to that proposed by Dr. Green.

The algorithm we study is based on a slightly different system of equations. Although the convergence properties are similar, we wished to remove the effect of linear transformations, and to incorporate some indication when the z_k^* bunch together, as happens when f has a large slope at some point. Consequently, we considered two somewhat modified systems of equations, neither of which seems to yield results markedly superior to the other. The first, and the one we analyze, is the system

$$\dot{z}_k = \frac{N_{k+1} - N_k}{\bar{N}}(x_{k+1} - x_k),\ k = 1, \cdots, K, \tag{1}$$

where x_k is as defined previously, and $\bar{N} = \max_{1 \leqq k \leqq K+1} N_k$, again with $Z = Z^{(0)}$ at $t = 0$. The other system was

$$\dot{z}_k = \frac{N_{k+1} - N_k}{\bar{N}}|z_k - y_k|,\ k = 1, \cdots, K, \tag{2}$$

where

$$y_k = \begin{cases} x_k, & \text{if } N_{k+1} - N_k \leqq 0, \\ x_{k+1}, & \text{if } N_{k+1} - N_k > 0, \end{cases}$$

and again $Z = Z^{(0)}$ at $t = 0$.

The factors $x_{k+1} - x_k$, and $|z_k - y_k|$ are both an attempt to "slow" z_k when the z_k^* bunch together. In some cases (2) is superior to (1), and in others (1) is superior to (2). We choose to analyze (1) because it seems to be more consistent in the optimum "time step" when solved by Euler's method. The analysis of (2) is nearly the same.

The point Z^* is clearly a rest point of the system (1), and in the next section we give an analysis showing that under appropriate assumptions, (1) is asymptotically stable at $Z = Z^*$.

There are many algorithms now possible, depending on the numerical method used to solve (1). Any method could be used, subject only to the appropriate choice of "time step". However, we should bear in mind that the goal is not necessarily to solve (1) accurately, but rather to approach Z^* closely. Thus, the use of Euler's method for the

solution. The choice of "time step" appears to be a matter of experience. It is desired to use a near optimal "time step", one which yields R_{mn}* to the desired accuracy in a minimum number of time steps, or iterations. It is seen that the "time step" is a parameter similar to the parameter in the solution of elliptic boundary value problems by the alternation-direction implicit method [**10**]. It is doubtful that it can be taken arbitrarily large in our case, however.

3. **Convergence.** The convergence of the algorithms possible in the setting of § 2, when an appropriate "time step" is used, is determined by whether or not the system (1) is asymptotically stable at $Z = Z^*$. We will make use of the following theorem, which is paraphrased slightly from the way it appears in most references, e.g., [1].

THEOREM 2. *The system of equations*

$$\dot{Z} = A \cdot (Z - Z^*) + H(Z - Z^*), \tag{3}$$

where A is a constant $K \times K$ matrix and $H(Z - Z^*)$ is a vector function which is small compared to $Z - Z^*$, is uniformly asymptotically stable at the point $Z = Z^*$ if the eigenvalues of A all have negative real parts.

Thus , in order to analyze the system (1) we must put it in the form (3). Denote sign$(E(x))$ on (z_{k-1}, z_k) by σ_k. By our assumptions about $E(x)$, we can write

$$E(x) = G(x) \prod_{i=1}^{K} (x - z_i),$$

where $G(x)$ is continuous and single signed if Z is close to Z*. Then we have $N_k = \sigma_k G(x_k) \prod_{i=1}^{K} (x_k - z_i)$, $k = 1, \cdots, K + 1$. Recall also that $a = z_0 < x_1 < z_1 < \cdots < x_K < z_K < x_{K+1} = b$.

We now make an assumption about the dependence of $G(x)$ on the z_i. As was done by Maehly and Witzgall [**9**], we assume that near the point Z*, the function $G(x)$ does not depend very much on the z_i. (Note: In the case of approximating x^K by a polynomial of degree $\leqq K - 1$, with $w(x) \equiv 1$, we have $G(x) \equiv 1$.) Then we have, for a given value of x,

$$\frac{\partial E(x)}{\partial z_j} \approx -G(x) \prod_{\substack{i=1 \\ i \neq j}}^{k} (x - z_i) = -\frac{E(x)}{x - z_j}.$$

Then, since x_k is the point of extreme error, even though x_k may change significantly with z_j, it is seen that the extreme value does not. Thus

we have

$$\frac{\partial N_k}{\partial z_j} \approx -\frac{\sigma_k E(x_k)}{x_k - z_j} \approx \frac{-\bar{N}}{x_k - z_j},$$

since near Z^*, $N_k \approx \bar{N}$. Then we have

$$\frac{N_{k+1} - N_k}{\bar{N}}(x_{k+1} - x_k) = \frac{x_{k+1}^* - x_k^*}{\bar{N}}$$

$$\sum_{j=1}^{K} \left(-\frac{\bar{N}}{x_{k+1}^* - z_j^*} + \frac{\bar{N}}{x_k^* - z_j^*} \right)(z_j - z_j^*) + H_k, k = 1, \cdots, K,$$

where

$$H = \begin{pmatrix} H_1 \\ \cdot \\ \cdot \\ \cdot \\ H_K \end{pmatrix}$$

is small compared to $Z - Z^*$. Further simplification gives the expression

$$(x_{k+1}^* - x_k^*)^2 \sum_{j=1}^{K} \frac{z_j - z_j^*}{(x_{k+1}^* - z_j^*)(x_k^* - z_j^*)} + H_k, k = 1, \cdots, K.$$

Thus, we have the system (1) in the form (3) with

$$A = \left(\frac{(x_{k+1}^* - x_k^*)^2}{(x_{k+1}^* - z_j^*)(x_k^* - z_j^*)} \right)_{k,j=1,\cdots,K}. \tag{4}$$

Now we must investigate the location of the eigenvalues of the matrix A. To simplify the notation, we drop the asterisks from the variables. We will show that the matrix $-A$ is a matrix of class **K** (see Fiedler & Pták [6]. These matrices are also known as M-matrices.)

Let B be a square matrix with nonpositive elements, except possibly on the diagonal. Matrices of class **K** are a subset of such matrices, characterized by many equivalent properties [6]. The two equivalent properties we shall use are given in

THEOREM 3. *The following properties of B are equivalent.* (i) *The real part of every eigenvalue of B is positive,* (ii) *Every leading principal minor of B is positive.*

LEMMA 4. *The matrix A defined by* (4) *has negative elements on the diagonal and positive elements elsewhere. Further, the principal minors of* $-A$,

$$\operatorname{Det}\left(-\frac{(x_{k+1}-x_k)^2}{(x_{k+1}-z_j)(x_k-z_j)}\right)_{k,j=1,\cdots,\ell}$$

are positive for $\ell = 1, 2, \cdots, K$, *whenever* $x_1 < z_1 < x_2 < z_2 < \cdots < x_K < z_K < z_{K+1}$.

PROOF. We consider the element $a_{kj} = (x_{k+1}-x_k)^2/(x_{k+1}-z_j)(x_k - z_j)$. The numerator is positive, and if $j < k$, then $z_j < x_{j+1} \leqq x_k < x_{k+1}$, and hence $a_{kj} > 0$. If $j > k$, we have $x_k < x_{k+1} \leqq x_j < z_j$, and again $a_{kj} > 0$. If $j = k$, $x_k = x_j < z_j < x_{j+1} = x_{k+1}$, hence $a_{kj} < 0$. We now consider the ℓth principal minor of $-A$,

$$\operatorname{Det}\left(-\frac{(x_{k+1}-x_k)^2}{(x_{k+1}-z_j)(x_k-z_j)}\right)_{k,j=1,\cdots,\ell}.$$

Removal of the factor $(x_{k+1}-x_k)^2$ from each row does not change the sign of the determinant, so we consider

$$\operatorname{Det}\left(-\frac{1}{(x_{k+1}-z_j)(x_k-z_j)}\right)_{k,j=1,\cdots,\ell} = D_\ell$$

The value of the latter determinant being nonzero is equivalent to the existence of a unique solution to the following interpolation problem: With functions $g_k(y) = -1/(x_{k+1}-y)(x_k-y)$, $k = 1, \cdots, \ell$, and given points (y_j, w_j), $j = 1, \cdots, \ell$, find constants α_k, $k = 1, \cdots, \ell$ *such* that

$$G_\ell(y_j) = \sum_{k=1}^{\ell} -\frac{\alpha_k}{(x_{k+1}-y_j)(x_k-y_j)} = w_j.$$

Here we assume that none of the y_j coincide with any of the x_k, which we have guaranteed in the case of interest. The above discussion says that D_ℓ is nonzero if the set of functions $g_k(y)$, $k = 1, \cdots, \ell$ is unisolvent on the permitted set of points. See Davis [5] for further discussion.

The interpolation problem is known to be uniquely solvable, if and only if any linear combination of the $g_k(y)$, $k = 1, \cdots, \ell$ which is zero at ℓ distinct points is identically zero. Consider $G_\ell(y) = \sum_{k=1}^{\ell} \alpha_k g_k(y)$. Now

$$G_\ell(y) = -\frac{1}{\prod_{k=1}^{\ell+1}(x_k-y)} \sum_{k=1}^{\ell} \alpha_k \prod_{\substack{j=1 \\ j\neq k \\ j\neq k+1}}^{\ell+1} (x_j - y) = \frac{P_{\ell-1}(y)}{Q_{\ell+1}(y)},$$

where $P_{\ell-1}$ and $Q_{\ell+1}$ are polynomials of degree $\ell - 1$ and $\ell + 1$, respectively. We see that $P_{\ell-1}$, and hence G_ℓ can have at most $\ell - 1$ distinct zeros, unless G_ℓ is identically zero. Hence $D_\ell \neq 0$.

Now let $y_1, \cdots, y_\ell$ be variable, but such that each y_j satisfies the condition stated for z_j in the lemma, i.e., $x_j < y_j < x_{j+1}$. Let $y_1, \cdots, y_\ell$ replace $z_1, \cdots, z_\ell$, respectively, in D_ℓ, and note that $x_k - y_k$ appears only on the diagonal. Hence by choosing y_k sufficiently close to x_k, the determinant is diagonally dominant, and is thus positive. By the continuity of D_ℓ as a function of $z_1, \cdots, z_\ell$, and the fact that D_ℓ is never zero, we conclude that $D_\ell > 0$. This completes the proof of the lemma.

Application of Theorems 3 and 2 to the system (1) yields the fact that (1) is asymptotically stable at $Z = Z^*$.

4. **Numerical Implementation.** A version of the algorithm, which we will call "Algorithm G", was implemented in double precision Fortran on the IBM 360/67 at the Naval Postgraduate School. We used Euler's Method to solve (1) numerically, using the initial guess $Z^{(0)}$ to be the zeros of the Chebyshev polynomial of degree K, translated to the interval $[a, b]$. The rational function R_{mn} was found by solving the linear system obtained by requiring R_{mn} to interpolate to f at the z_j. The IMSL routine LEQTIF was used to solve the system. (IMSL — International Mathematical and Statistical Libraries, Inc., 6200 Hillcroft, Suite 510, Houston, Texas 77036.)

The extreme values N_k were found by the method suggested by Maehly and Witzgall [9]. A search is made for a "turning point" using the previous value of x_k as an initial estimate, using steps of $h_k = .015(z_k - z_{k-1})$. When three points have been computed so that $|E(x)|$ is largest at the middle point, the value of x_k is approximated by passing a parabola through the three points, and finding its extreme point. The alternative is to convert the problem to a discrete one by evaluating the error at a fixed number of points, as did Lee and Roberts [8]. We feel our method is probably faster, especially in the latter stages, although it assumes the error curve has but one local extreme in each interval (z_{k-1}, z_k). We believe the method to be more accurate for smooth problems, as well as preserving the continuity of the original problem.

Algorithm G was tested by running a variety of problems. The same problems were also run using the Remes algorithm (The IMSL routine IRATCU was used. This is a Fortran version of procedure *Chebyshev* due to Cody, Fraser, and Hart [3].) and Maehley's second

method; the latter program was written by the author. The problems ranged from "easy", such as $\exp(x)$, $\Gamma(x)$, and $\log x$, to "hard" problems such as $\sqrt{x}$ on $[0, 1]$ and $\sqrt{x}$ on $[1/4, 1]$ and $[1/16, 1]$, the latter two with weight function $w(x) = \sqrt{x}$ for relative fits. In addition, the function $r(x) = (\arctan 8x)\ \sqrt{(8x-1)^2 + 1/8x}$, with $w(x) \equiv 1$, on $[-1, 1]$ was attempted. This latter function is due to Rutishauser (see Cody & Stoer [4], p. 179) and is difficult because the initial guesses usually used (data from appropriate Chebyshev polynomial) with the Remes algorithm lead to approximants with a pole in $[-1, 1]$.

The approximations of the form R_{11}, R_{13}, R_{22}, and R_{42} were usually attempted, although in some specific instances others were computed. In the case of the "easy" functions all three algorithms worked well, with the Remes algorithm converging very rapidly, of course. The Remes algorithm and Maehley's method failed on one or more of the "hard" problems, while Algorithm G, for appropriate "time step", did not fail. We note that one iteration of the Remes algorithm requires the solution of a non-linear system; Maehly's method requires the solution of two linear systems, while Algorithm G requires the solution of one linear system.

The difficulties with Rutishauser's function $r(x)$ were investigated more closely. Aside from the problem of poles, the function has a large slope near $x = 0$, which apparently causes difficulties. The programs for the Remes algorithm and Maehly's method were modified to accept input initial guesses. For initial guesses at the extremes which were accurate to seven significant digits, the Remes algorithm failed. For initial guesses at the interpolation points which were accurate to seven significant digits, Maehly's method failed. No particular difficulties were experienced by Algorithm G.

With regard to the possibility of poles in the initial approximant, we have discovered that while theoretically the method fails, the numerical algorithm may be able to recover. For example, when approximating $\sin x$ on $[0, 4.1]$ with $R_{01} = P_0/Q_1$, using the Chebyshev points as initial guesses gives an initial approximant with a pole near $x = 1.7$. However, because of the approximation to the extremes, the routine recovers, forces the pole out of the approximation, and then converges to the correct result. This may not be a general rule, however, but is an interesting example of how robust the algorithm can be.

Having satisfied ourselves that the algorithm works quite well, a study was made of how one should choose the "time step". We found that "time step" $\Delta t = .20$ was usually (not always) small enough for

convergence. The optimum value for Δt is dependent on the problem, and sometimes is significantly larger than .20. As might be expected, the optimum value is larger than required for accurate solution of (1), and it is better to underestimate Δt than to overestimate it. Table 1 gives the number of time steps (or iterations) required for convergence of various approximations to $\exp(x)$ on $[0, 1]$, $\sqrt{x}$ on $[0, 1]$, and $r(x)$ on $[-1, 1]$, versus Δt. The iterations were stopped when successive approximations were obtained whose respective coefficients had relative differences of less than 10^{-7}.

Because the solution of (1) is close to the solution of $\dot{Z} = A(Z - Z^*)$ for large times, the convergence rate is seen to be linear. Further, we can see in Table 1 that convergence is slow even for smooth functions such as $\exp(x)$, where 10–20 ieterations are required. In the case of $\sqrt{x}$, where most of the z_k tend to bunch near zero, significantly more iterations are required, the number increasing with K.

Table 1

SEARCH FOR OPTIMUM Δt

	$\exp(x)$		$\sqrt{x}$		$r(x)$		
Δt	P_1/Q_1	P_4/Q_2	P_1/Q_1	P_2/Q_2	P_1/Q_1	P_2/Q_2	P_4/Q_2
.10	36		64				
.125			51	>100	29	64	61
.15	23	47	41	97	22	49	53
.175			44	82	17	39	42
.20	16	35	>100	>100	22	failed	failed
.225	14				32		
.25	12	28					
.275	19						
.30	36	23					
.325							
.35		19					
.375		18					
.40		16					
.425		16					

5. **Conclusion.** Algorithm G appears to be very robust. When coupled with the appropriate "time step", it is likely infallible, provided the initial guess did not yield an approximant with a pole in the interval. Even then, the version of the algorithm we implemented has recovered in specific instances.

While this algorithm cannot compete with the Remes algorithm on the basis of speed, our tests show its speed to be comparable to Maehly's algorithm, which shows up quite well, on that basis, in the Lee and Roberts study. One suspects the Lee and Roberts timing is biased since cases where the algorithm was successful were likely to be the relatively easier (and faster) cases.

The simplicity of the algorithm coupled with its high success rate, make it worthy of consideration for computing approximations which result in failure of the Remes algorithm. The principal disadvantage seems to be the assumption that there are no more than $m + n + 1$ zeros of the error curve. Degenerate cases can be handled by increasing the degree of the numerator and/or denominator so that the error curve becomes standard. If he cause for more than $m + n + 1$ zeros is not degeneracy, the algorithm may fail.

6. **Acknowledgment.** The author wishes to acknowledge helpful discussions with Professors A. L. Schoenstadt and D. A. Archer.

References

1. S. Barnett and C. Storey, *Matrix methods in stability theory,* Barnes and Noble, Inc., New York, 1970.

2. E. W. Cheney, *Introduction to approximation theory,* McGraw-Hill Book Co., New York, 1966.

3. W. J. Cody, W. Fraser and J. F. Hart, *Rational Chebyshev approximation using linear equations,* Numerische Mathematik **12** (1968), 242–251.

4. W. J. Cody and J. Stoer, *Rational Chebyshev approximation using interpolation,* Numerische Mathematik **9** (1966), 177–188.

5. Philip J. Davis, *Interpolation and approximation,* Blaisdell Publishing Co., Waltham, Mass., 1963.

6. Miroslav Fiedler and Vlastimil Ptàk, *On matrices with non-positive off-diagonal elements and positive principal minors,* Czechoslovak Mathematical Journal **12** (1962), 382–400 (English).

7. Milton W. Green, *A new algorithm for rational approximation,* delivered at joint meeting of the Northern California sections of SIAM and MAA, February 5, 1972.

8. C. M. Lee and F. D. K. Roberts, *A comparison of algorithms for rational ℓ_∞ approximation,* Math. Comput. **27** (1973), 111–121.

9. Hans J. Maehly and Christoph Witzgall, *Methods for fitting rational approximations, Parts* II *and* III, Journal ACM **10** (1963), 257–277.

10. D. W. Peaceman and H. H. Rachford, *The numerical solution of parabolic and elliptic differential equations,* J. SIAM 3 (1955), 28–41.

Naval Postgraduate School, Monterey, California 93940