Two Simple Algorithms for Discrete Rational Approximation

By I. Barrodale and J. C. Mason

Abstract. This paper reports on computational experience with algorithms due to Loeb and Appel for rational approximation on discrete point sets. Following a brief review of the linear discrete approximation problem, the two rational algorithms are stated in a general setting. Finally, several numerical examples of applications to l_1, l_2, and l_∞ approximation are supplied and discussed.

1. Introduction. Rational functions can often provide very satisfactory approximations to discrete data. However, as with most other nonlinear approximating functions, effective algorithms that produce best discrete rational approximations are few in number and are often complicated or time-consuming. The purpose of this paper is to give further exposure to two rational algorithms due to Loeb and Appel, to extend their applicability to each of the three norms l_1, l_2, and l_∞, and to test their effectiveness on a variety of problems. Both methods are simple in the sense that they employ only a linear approximation algorithm and possibly a straightforward iteration.

For the sake of completeness, the remainder of this introductory section consists of some remarks on the general problem of best approximation on a discrete point set.

Given a set $X = \{x_1, x_2, \cdots, x_N\}$ of real numbers and a function $f(x)$ defined on X, we choose an approximating function $F(A, x)$ and select a particular form $F(A^*, x)$ which approximates $f(x)$ satisfactorily on X, according to some criterion. Here, $A = \{a_1, a_2, \cdots, a_n\}$ is a set of free parameters, and $F(A, x)$ is a linear approximating function only if it depends linearly upon these parameters. Thus, a rational function $F(A, x) = (a_1 + a_2 x)/(1 + a_3 x)$ is nonlinear, and the most general linear function is $F(A, x) = \sum_{j=1}^{n} a_j \phi_j(x)$, where the $\phi_j(x)$'s are given linearly independent functions defined on X. $F(A^*, x)$ is called a best approximation in a norm $||\cdot||$ if, for all choices of A, $||f(x) - F(A^*, x)|| \leq ||f(x) - F(A, x)||$. The three norms used in practice are:

$$l_1: \quad ||w(x)[f(x) - F(A, x)]||_1 = \sum_{i=1}^{N} w(x_i) |f(x_i) - F(A, x_i)|,$$

$$l_2: \quad ||w(x)[f(x) - F(A, x)]||_2 = \left\{ \sum_{i=1}^{N} w(x_i)[f(x_i) - F(A, x_i)]^2 \right\}^{1/2},$$

$$l_\infty: \quad ||w(x)[f(x) - F(A, x)]||_\infty = \max_{1 \leq i \leq N} w(x_i) |f(x_i) - F(A, x_i)|,$$

where $\{w(x_i)\}$ is a prescribed set of positive weights.

In the linear case best approximations exist in all three norms, but only l_2 ap-

proximations are necessarily unique. The actual determination of best l_1 and l_∞ linear approximations may be effected by linear programming (see Barrodale and Young [3]), while l_2 approximations may be found by solving the relevant normal equations. If the discrete data is affected by noise then the appropriate choice of norm depends upon the distribution of these random errors. In practice one chooses the l_∞, l_2, or l_1 norm, respectively, according as the errors are very small relative to the error of approximation, normally distributed, or subject to wild points. (Rice and White [7] give evidence which supports this viewpoint.)

Progress towards developing algorithms for nonlinear approximation has been slow: much information concerning the state of the art to date is contained in Rice [6]. The theory of best nonlinear approximation on intervals is more advanced than that for discrete sets, where existence cannot even be guaranteed for mildly nonlinear functions. Rational approximation appears to be the best understood nonlinear approximation problem, even though the examples presented in this paper of rational l_1 approximations are among the first to appear in the literature from this norm.

2. The Loeb Algorithm. Suppose that a generalized rational approximation is required to $f(x)$ of the form

$$F(A, x) = \frac{P_m(x)}{Q_n(x)} = \frac{p_0\phi_0(x) + p_1\phi_1(x) + \cdots + p_m\phi_m(x)}{q_0\psi_0(x) + q_1\psi_1(x) + \cdots + q_n\psi_n(x)},$$

where $\{\phi_i(x)\}$ and $\{\psi_i(x)\}$ are both linearly independent sets of given functions. The parameter set $A = \{p_0, \cdots, p_m, q_0, \cdots, q_n\}$ can be normalized by, for example, setting $q_0 = 1$. Then the problem of determining a best approximation in a given norm is to choose A^* which minimizes $||f(x) - P_m(x)/Q_n(x)||$.

The Loeb algorithm is an iterative procedure which consists of minimizing at the kth stage the quantity

(A)
$$\left\| \frac{1}{Q^{(k-1)}} [fQ^{(k)} - P^{(k)}] \right\|$$

by appropriately choosing the coefficients $p_i^{(k)}$ and $q_i^{(k)}$. Here, $f \equiv f(x)$, $Q^{(k)} \equiv Q_n^{(k)}(x)$, $P^{(k)} \equiv P_m^{(k)}(x)$, and $1/Q^{(k-1)}$ is regarded as a known weight function. We fix $q_0 = 1$ and normally start the iteration by putting $Q^{(0)} = 1$. Then the minimization of (A) is a linear approximation problem which can be tackled in any of the norms l_1, l_2, or l_∞, by the techniques referred to in Section 1. The iteration is continued until the computed parameters of best approximation in (A) converge to within some prescribed tolerance.

This algorithm was proposed by Loeb [4] for the l_∞ norm and by Wittmeyer [8] for the l_2 norm. Our own experience in using the algorithm, some of which is summarized in Section 5, suggests that it is often a satisfactory technique for producing good rational approximations. However, we have encountered examples in all three norms where the procedure has not converged even after a large number of iterations. More seriously, we have also discovered examples in each norm where the algorithm does converge, but not to a best approximation.

Wittmeyer [8] states that if the l_2 algorithm converges, then it necessarily converges to a best approximation. This claim is untrue: indeed, in general there is almost

no chance that Loeb's algorithm will converge to a best least-squares approximation, unless there is no error in the approximation!

We are indebted to Mr. M. J. D. Powell for this last remark which can be justified as follows. Loeb's algorithm has converged when $P^{(k-1)}/Q^{(k-1)} = P^{(k)}/Q^{(k)}$, which in the l_2 norm requires that the partial derivatives of the square of expression (A) with respect to each p_j and q_i be all zero at the approximation $P^{(k)}/Q^{(k)}$. But these conditions on $P^{(k)}/Q^{(k)}$ are different from the system of equations which is generated by the normal equations for the square of $||f(x) - P_m(x)/Q_n(x)||_2$, although this difference diminishes as the error of approximation decreases. Thus if Loeb's algorithm converges, it is likely to reach a point that is not a solution to the best least-squares rational approximation problem.

3. Approximation by $(P/Q)^r$. Any algorithm that produces discrete best approximations by functions of the form $G(A, x)$ can be used to compute near-best approximations by $F(A, x) = (G(A, x))^r$. The prescribed quantity r is usually a positive integer, and the resulting approximations are "near-best" in the sense of the analysis below.

In general, we wish to minimize $||f(x) - F(A, x)|| = ||(g(x))^r - (G(A, x))^r||$, where $g(x) = (f(x))^{1/r}$. Suppose that for each point $x \in X$ we define $\epsilon(x) = f(x) - F(A, x)$. then

(B)
$$G(A, x) = [(g(x))^r - \epsilon(x)]^{1/r} = g(x)[1 - \epsilon(x)/f(x)]^{1/r}$$

$$= g(x)[1 - \epsilon(x)/rf(x)] + O\left(\frac{\epsilon(x)}{f(x)}\right)^2.$$

If $\epsilon(x)$ is small compared to $f(x)$ on the discrete point set X, then we may neglect second order terms in (B) and write

$$g(x) - G(A, x) = \frac{1}{r}\frac{g(x)}{f(x)}\epsilon(x),$$

thus,

$$||f(x) - F(A, x)|| = ||w(x)[g(x) - G(A, x)]||,$$

where $w(x) = |r(f(x)/g(x))|$.

Hence, the minimum of $||f(x) - F(A, x)||$ can be found approximately by solving the known problem of minimizing $||w(x)[g(x) - G(A, x)]||$.

In particular, approximations of the form

$$F(A, x) = \left(\frac{P_m(x)}{Q_n(x)}\right)^r = \left(\frac{p_0\phi_0(x) + p_1\phi_1(x) + \cdots + p_m\phi_m(x)}{q_0\psi_0(x) + q_1\psi_1(x) + \cdots + q_n\psi_n(x)}\right)^r$$

may be determined by applying the Loeb algorithm to

$$\left|\left|w(x)\left[(f(x))^{1/r} - \frac{P_m(x)}{Q_n(x)}\right]\right|\right|,$$

where $w(x) = |r(f(x))^{1-1/r}|$.

4. The Appel Algorithm. Clearly, a rational function with a fixed denominator is just a linear approximating function. More interestingly, Appel [1] has shown

that if the numerator is fixed but the denominator varies, then a rational approxi-
mation can be computed by using a weighted linear algorithm. This is a noniterative process which produces a near-best approximation. Such a form of rational
approximation is particularly appropriate for fitting data which tends rapidly to
zero as x tends to infinity.

We are concerned with functions of the form

$$F(A, x) = P_m(x)/(q_0\psi_0(x) + q_1\psi_1(x) + \cdots + q_n\psi_n(x))^r,$$

where the parameter set is $A = \{q_0, q_1, \cdots, q_n\}$, the prescribed quantity r is usually
a positive integer, and $P_m(x)$ is a given fixed function. Normally, $P_m(x)$ is chosen
to ensure that $P_m(x)/f(x)$ is positive for all $x \in X$. Appel's analysis, which is very
similar to that of Section 3, was given in [1] for the l_2 norm, but it extends trivially to
any other discrete norm.

For any $x \in X$ we define

$$g(x) = (P_m(x)/f(x))^{1/r},$$

$$G(A, x) = (P_m(x)/F(A, x))^{1/r} = q_0\psi_0(x) + q_1\psi_1(x) + \cdots + q_n\psi_n(x)$$

and $\epsilon(x) = f(x) - F(A, x)$. Then, clearly

(C) $\qquad G(A, x) = g(x)\left[1 - \frac{\epsilon(x)}{f(x)}\right]^{-1/r} = g(x)\left[1 + \frac{1}{r}\frac{\epsilon(x)}{f(x)}\right] + O\left(\frac{\epsilon(x)}{f(x)}\right)^2.$

If $\epsilon(x)$ is small compared to $f(x)$ on the discrete point set X, then we may neglect
second order terms in (C) and write

$$g(x) - G(A, x) = -\frac{1}{r}\frac{g(x)}{f(x)}\epsilon(x),$$

thus,

$$||f(x) - F(A, x)|| = ||w(x)[g(x) - G(A, x)]||,$$

where $w(x) = |r(f(x)/g(x))|$.

Hence, the minimum of $||f(x) - F(A, x)||$ can be found approximately by solving
the known problem of minimizing

(D) $\qquad\qquad\qquad\qquad ||w(x)[g(x) - G(A, x)]||.$

Since $G(A, x)$ is a linear approximating function, the original problem is thus solved
approximately by one application of a linear algorithm.

5. Numerical Results. This section contains complete details of several rational
approximations obtained by the methods described above. Tables 1A and 1B show
approximations in each of the norms l_1, l_2, and l_∞, to four smooth functions. Three
different rational approximating functions were used and Loeb's algorithm was
employed. Tables 2A and 2B record our attempts to approximate six functions
whose behavior might be described as "kinky". The analytic expressions for these
six functions show that they are either discontinuous, or they have rapidly changing
derivatives, at some points in the intervals containing their abscissae. Loeb's algo-
rithm, all three norms, and the approximating function $P_2(x)/Q_2(x)$, were used
here. Table 3 is a collection of l_2 approximations by $P_2(x)/Q_2(x)$ to various standard

TABLE 1A. Smooth data: Summary

Loeb's algorithm   Convergence criterion: relative change in each parameter $< 10^{-4}$

$f(x)$ Function	Domain	N Number of data points	$F(A,x)$ Form of approximation	Number of iterations to convergence		
				ℓ_1	ℓ_2	ℓ_∞
A-1. $\exp(x)$	$[-1,1]$	21	$P_2(x)/Q_2(x)$	3	3	4
A-2. $\tan(x)$	$(0,\pi)$	20	$P_2(x)/Q_2(x)$	3	4	3
A-3. Thomas–Fermi	$[0,20]$	18	$\left[P_1(x^{1/2})/Q_4(x^{1/2})\right]^2$	3	5	5
A-4. Dawson Integral	$[0,10]$	20	$x\,P_3(x^2)/Q_4(x^2)$	5	7	12

Note: (i) Dawson Integral defined by $f(x) = \exp(-x^2) \displaystyle\int_0^x \exp(t^2)\,dt$

(ii) Thomas–Fermi defined by $x(f''(x))^2 = (f(x))^3$ with $f(0) = 1$, $f(\infty) = 0$

(This approximation requires the special version of Loeb's algorithm described in §3)

TABLE 1B. Smooth data: Parameters, and error measures.

Loeb's algorithm   Define $\varepsilon_i = \varepsilon(x_i) = f(x_i) - F(A^*,x_i)$

ℓ_1 norm				
Function	A-1	A-2	A-3	A-4
$\frac{1}{N}\Sigma|\varepsilon_i|$	$(5.5)10^{-5}$	$(1.1)10^{-2}$	$(1.3)10^{-5}$	$(4.4)10^{-5}$
p_0	1.00006	0.08883	1.00000	0.99951
p_1	0.50876	0.67740	1.04287	0.06945
p_2	0.08603	-0.21967		0.03081
p_3				0.00706
q_1	-0.49103	-0.66249	1.04569	0.73146
q_2	0.07780	0.01646	0.77072	0.26573
q_3			0.24200	0.04560
q_4			0.07890	0.01427

ℓ_2 norm				
Function	A-1	A-2	A-3	A-4
$\{\frac{1}{N}\Sigma\varepsilon_i^2\}^{1/2}$	$(6.3)10^{-5}$	$(1.4)10^{-2}$	$(1.7)10^{-5}$	$(6.9)10^{-5}$
p_0	1.00006	0.06042	1.00000	0.99979
p_1	0.50900	0.72913	1.06087	0.08275
p_2	0.08609	-0.23294		0.03265
p_3				0.00764
q_1	-0.49077	-0.64277	1.06292	0.74635
q_2	0.07761	0.00391	0.77504	0.27460
q_3			0.24694	0.05041
q_4			0.08004	0.01538

ℓ_∞ norm				
Function	A-1	A-2	A-3	A-4
$\max|\varepsilon_i|$	$(8.5)10^{-5}$	$(2.0)10^{-2}$	$(2.4)10^{-5}$	$(9.0)10^{-5}$
p_0	1.00007	0.04712	1.00001	0.99978
p_1	0.50840	0.75196	1.07545	0.10119
p_2	0.08571	-0.23936		0.03246
p_3				0.00961
q_1	-0.49133	-0.63662	1.07717	0.76350
q_2	0.07781	0.00000	0.77790	0.29193
q_3			0.25140	0.04986
q_4			0.08086	0.01927

TABLE 2A. Kinky data: Summary

Loeb's algorithm   Form of approximation: $P_2(x)/Q_2(x)$

I = number of iterations to convergence, with relative change in each parameter $< \delta$

f(x) Function	Domain	N Number of data points	ℓ_1		ℓ_2		ℓ_∞	
			I	δ	I	δ	I	δ
B-1. e^x ; $e^{-x}-1+e$	$[0,1]$; $[1,2]$	21	3	10^{-4}	24	10^{-4}	15	10^{-4}
B-2. $(x-2)x(x+1)$	$[-5,5]$	21	4	10^{-4}	25	$(1.5)10^{-2}$	No convergence	
B-3. 1 ; 0 ; -1	$[0,.5)$; $x=.5$; $(.5,1]$	(a) 21	No convergence		25	$(3.6)10^{-3}$	No convergence but oscillates between two good approximations	
		(b) 7	Converges to good (non-best approximation)		10	10^{-4}	No convergence	
B-4. 1 ; 0 ; -1	$[0,.75)$; $x=.75$; $(.75,1]$	21	No convergence		22	10^{-4}	Converges to poor (non-best) approximation	
B-5. x ; $.5x + .4$	$[0,1]$; $(1,2]$	21	No convergence but oscillates between two good approximations		18	10^{-4}	25	$(1.5)10^{-3}$
B-6. e^x ; $e^{-x}-1+e+.3$	$[0,1]$; $(1,2]$	21	3	10^{-4}	19	10^{-4}	15	10^{-4}

Note: "No convergence" means that no sign of convergence is evident within the first twenty-five iterations.

TABLE 2B. Kinky data: Parameters, and error measures.

Loeb's algorithm.   Define $\epsilon_1 = \epsilon(x_1) = f(x_1) - P_2^*(x_1)/Q_2^*(x_1)$

ℓ_1 norm

Function $\frac{1}{N}\Sigma\lvert\epsilon_1\rvert$	B-1 $(4.2)10^{-2}$	B-2 $(6.8)10^{0}$	B-3(b) $(8.4)10^{-2}$	B-5 $(1.6)10^{-2}$	B-5 $(1.6)10^{-1}$	B-5** $(1.5)10^{-2}$	B-6 $(4.8)10^{-2}$
p_0	1.06808	0.00000	1.00000	-0.01922	-0.02278	-0.02077	1.03605
p_1	-0.72730	5.95455	-2.41667	1.24981	1.28849	1.26664	-0.27026
p_2	1.09168	-0.59091	0.83333	-1.23367	-1.21540	-1.21977	0.53633
q_1	-1.19763	-0.01515	-3.41667	-0.61876	-0.57009	-0.59252	-0.99072
q_2	0.76644	-0.03030	3.33333	-0.38432	-0.37960	-0.38138	0.49248

ℓ_2 norm

Function $\{\frac{1}{N}\Sigma\epsilon_1^2\}^{\frac{1}{2}}$	B-1 $(5.9)10^{-2}$	B-2 $(7.8)10^{0}$	B-3(a) $(2.2)10^{-1}$	B-3(b) $(1.1)10^{-1}$	B-4 $(1.7)10^{-1}$	B-5 $(2.1)10^{-2}$	B-6 $(6.5)10^{-2}$
p_0	1.07453	-1.77204	0.76612	0.88418	0.88947	-0.02361	1.06892
p_1	-0.96063	6.41934	-1.78356	-1.92017	-1.92908	1.31795	-0.70335
p_2	1.23840	-0.52302	0.56383	-0.33051	0.99923	-1.26376	0.85071
q_1	-1.30947	-0.01366	-3.42950	-3.31839	-2.41857	-0.55873	-1.16566
q_2	0.83563	-0.02994	3.17934	3.17342	1.49006	-0.41090	0.62095

ℓ_∞ norm

Function $\max\lvert\epsilon_1\rvert$	B-1 $(8.7)10^{-2}$	B-3(a) $(5.6)10^{-1}$	B-3(a) $(5.6)10^{-1}$	B-3(a)** $(2.7)10^{-1}$	B-4 $(1.0)10^{0}$	B-5 $(3.2)10^{-2}$	B-6 $(9.3)10^{-2}$
p_0	1.08651	0.59526	0.82091	0.73077	$(1.0)10^{-7}$	-0.03213	1.09321
p_1	-1.48281	-0.66786	-1.98949	-1.46153	$(3.0)10^{-7}$	1.37993	-1.29127
p_2	1.78878	-1.16148	0.77262	-0.00002	$(-5.0)10^{-7}$	-1.30555	1.46471
q_1	-1.54132	-4.04816	-3.36243	-3.63637	-2.34203	-0.52908	-1.39066
q_2	1.07042	4.55147	3.02763	3.63637	1.37117	-0.43001	0.85104

Note: ** denotes that here these approximations were computed by direct search.

TABLE 3. Various ℓ_2 approximations: Summary

Loeb's algorithm.   Form of approximation: $P_2(x)/Q_2(x)$

Convergence criterion: relative change in each parameter $< 10^{-4}$

Define $\varepsilon_i = \varepsilon(x_i) = f(x_i) - P_2^*(x_i)/Q_2^*(x_i)$

	f(x) Function	Domain	N Number of data points	Number of iterations to convergence	$\{\frac{1}{N}\Sigma\varepsilon_i^2\}^{\frac{1}{2}}$
C-1.	$\log(1 + x)$	$[0,1]$	21	3	$(1.3)10^{-6}$
C-2.	$\log(x)$	$[.1,1]$	10	5	$(2.6)10^{-4}$
C-3.	$\sqrt{x}$	$[0,1]$	21	7	$(1.3)10^{-3}$
C-4.	$\sqrt[3]{x}$	$[0,1]$	21	7	$(1.5)10^{-3}$
C-5.	$\sin(x)$	$[0,3]$	21	3	$(6.2)10^{-4}$
C-6.	$\sin(x)$	$[-3,3]$	21	No convergence	
C-7.	$\arcsin(x)$	$[0,1]$	11	5	$(1.5)10^{-3}$
C-8.	$\arctan(x)$	$[0,1]$	21	4	$(4.0)10^{-5}$
C-9.	$\arctan(x)$	$[0,10]$	21	5	$(4.2)10^{-4}$
C-10.	$\sinh(x)$	$[0,1]$	11	4	$(3.8)10^{-5}$
C-11.	$\tanh(x)$	$[0,1]$	11	3	$(2.4)10^{-5}$
C-12.	$\Gamma(1 + x)$	$[0,1]$	11	3	$(1.1)10^{-5}$
C-13.	$\log \Gamma(1 + x)$	$[0,1]$	11	3	$(1.1)10^{-5}$
C-14.	$\operatorname{erf}(x)$	$[0,2]$	11	5	$(1.1)10^{-3}$
C-15.	$\exp(-x^2)$	$[0,2]$	11	4	$(2.0)10^{-3}$
C-16.	$\begin{cases} x \\ 1-x \end{cases}$	$\begin{bmatrix} [0,.5] \\ [.5,1] \end{bmatrix}$	21	7	$(1.4)10^{-2}$
C-17.	$\begin{cases} 4x \\ 4(1-x) \end{cases}$	$\begin{bmatrix} [0,.5] \\ [.5,1] \end{bmatrix}$	21	7	$(5.5)10^{-2}$

functions: Loeb's algorithm was used throughout. Tables 4A and 4B contain four approximations by $1/(Q_n(x))^r$. Appel's algorithm was used, and results are shown for each norm. Table 5 lists the data points used to define each function approximated in this study.

The functions and forms of approximation in Tables 1 and 4 were chosen on the basis of previous experience in Mason [5]. The standard functions in Table 3

TABLE 4A.   Appel's algorithm: Summary

Domain: $[0, \infty)$

Form of approximation: $\dfrac{1}{(Q_n(x))^r} = (q_0 + q_1 z + \ldots + q_n z^n)^{-r}$

Define $\varepsilon_i = \varepsilon(x_i) = f(x_i) - 1/(Q_n^*(x_i))^r$

Function	N Number of data points	z	n	r	ℓ_1 $\frac{1}{N} \Sigma \lvert \varepsilon_i \rvert$	ℓ_2 $\{\frac{1}{N} \Sigma \varepsilon_i^2\}^{\frac{1}{2}}$	ℓ_∞ $\max \lvert \varepsilon_i \rvert$
D-1. exp(-x)	20	x	4	4	$(4.6)10^{-6}$	$(2.5)10^{-5}$	$(3.9)10^{-5}$
D-2. 1 - erf(x)	20	x	5	4	$(1.1)10^{-5}$	$(5.3)10^{-5}$	$(1.0)10^{-4}$
D-3. Blasius	20	x	5	4	$(2.3)10^{-5}$	$(1.2)10^{-4}$	$(1.6)10^{-4}$
D-4. Thomas- Fermi	18	$x^{\frac{1}{2}}$	6	1	$(5.9)10^{-5}$	$(7.4)10^{-5}$	$(1.0)10^{-4}$

Note:

(i)   erf(x)   defined by $f(x) = \dfrac{2}{\sqrt{\pi}} \displaystyle\int_0^x \exp(-t^2)dt$

(ii)   Blasius defined by $f(x) = g(x) + 1.7208 - 2x$ where
      $g'''(x) + g(x) g''(x) = 0$ with $g(0) = g'(0) = 0$, $g'(\infty) = 2$

(iii)   D-4. defined same as A-3 (see Table 1A).

were chosen at the suggestion of Mr. I. Farkas. All l_1 and l_∞ results were computed in double precision arithmetic on an IBM 360/44 at Victoria; improved versions of the linear algorithms given in [3] were used to minimize the quantities (A) and (D) in the algorithms of Loeb and Appel, respectively. All l_2 results were computed in single precision on an IBM 7094 at Toronto; a double precision subroutine (unpublished) by W. Kahan for linear least-squares approximation was used to minimize (A) and (D) in this case.

We shall finish by making several remarks on particular approximations shown in these tables.

The results in Tables 1A and 1B give a good indication of how well Loeb's algorithm can behave in practice. In all cases convergence to the accuracy shown has occurred within just a few iterations. For example, in the case A-1 the total time required to compute a nonlinear approximation, in any of the three norms, is less than five times that which is required to compute a linear approximation with the same number of free parameters. The l_∞ approximations produced for this paper by Loeb's algorithm almost all satisfy the characteristic equioscillation property of best approximations, whereas most of the l_2 approximations do not satisfy the necessary normal equations. (There is no known characterization theorem for l_1 rational approximation.) Each approximation computed in A-3 is certainly of high

TABLE 4B. Appel's algorithm: Parameters

ℓ_1 norm			
Function	D-1	D-2	D-3	D-4
q_0	1.00000	1.00000	0.87311	1.00000
q_1	0.24996	0.28353	0.25443	0.02928
q_2	0.03148	0.18539	0.09369	1.33427
q_3	0.00233	0.11521	0.05685	-0.40641
q_4	0.00026	-0.02907	-0.01018	0.45413
q_5		0.03266	0.01343	-0.06865
q_6				0.01580

ℓ_2 norm			
Function	D-1	D-2	D-3	D-4
q_0	1.00001	0.99999	0.87310	0.99998
q_1	0.24996	0.28322	0.25470	0.02180
q_2	0.03146	0.18790	0.09168	1.37286
q_3	0.00235	0.10951	0.06145	-0.47086
q_4	0.00026	-0.02395	-0.01392	0.49910
q_5		0.03105	0.01440	-0.08176
q_6				0.01705

ℓ_∞ norm			
Function	D-1	D-2	D-3	D-4
q_0	1.00001	0.99998	0.87309	0.99990
q_1	0.24989	0.28372	0.25496	0.02160
q_2	0.03158	0.18423	0.08996	1.37701
q_3	0.00230	0.11837	0.06521	-0.48130
q_4	0.00026	-0.03203	-0.01704	0.50798
q_5		0.03341	0.01526	-0.08453
q_6				0.01730

TABLE 5. Definitions of discrete data sets

Note: In the last column the word "subroutine" indicates that the ordinates were generated in a computer subroutine to at least 7 decimals.

Function	Abscissae	Accuracy of ordinates
A-1	-1.0(0.1)1.0	subroutine
A-2	kπ/21 for k = 1(1)20	subroutine
A-3	0.0,0.05,0.1,0.2,0.45,0.7,0.95,1.2,1.6,2.1, 2.6,3.2,4.2,5.5,8,11,16,20	5 sig. figs.
A-4	0.2,0.4,0.6,0.8,0.9,1.0(0.2)2.0,2.5(0.5)6.0,10	4 decimals
B-1	0.0(0.1)2.0	subroutine
B-2	-5.0(0.5)5.0	subroutine
B-3(a)	0.0(0.05)1.0	subroutine
B-3(b)	0.0,0.2,0.4,0.5,0.6,0.8,1.0	subroutine
B-4	0.0(0.05)1.0	subroutine
B-5,B-6	0.0(0.1)2.0	subroutine
C-1	0.0(0.05)1.0	subroutine
C-2	0.1(0.1)1.0	subroutine
C-3,C-4	0.0(0.05)1.0	subroutine
C-5	0.0(0.15)3.0	subroutine
C-6	-3.0(0.3)3.0	subroutine
C-7	0.0(0.1)1.0	subroutine
C-8	0.0(0.05)1.0	subroutine
C-9	0.0(0.5)10.0	subroutine
C-10,C-11,C-12,C-13	0.0(0.1)1.0	subroutine
C-14,C-15	0.0(0.2)2.0	subroutine
C-16,C-17	0.0(0.05)1.0	subroutine
D-1	0.0(0.1)0.6,0.8,1.0,1.2,1.6,2.0,2.5,3.0, 3.5,4,5,6,7,9	4 decimals
D-2	0.0(0.1)1.0,1.2(0.2)2.8	4 decimals
D-3	0.0(0.1)0.6,0.8(0.2)3.0,3.4	4 decimals
D-4	same as A-3	5 sig. figs.

accuracy. Note that in the case A-2 all three approximations reproduce the pole at $x = \pi/2$ to an accuracy of at least four decimal places.

The examples in Tables 2A and 2B provide a more severe test for Loeb's algorithm. In the l_2 norm, convergence does not always occur to the desired accuracy within twenty-five iterations; nevertheless the convergence rates in this norm are quite satisfactory both here and in Table 3. It can be seen that in the l_1 and l_∞ norms the results are far from satisfactory. In both cases there are examples shown where no sign of convergence is evident within the first twenty-five iterations. From further tests performed on these examples, we can report that this behavior usually persists after fifty or more iterations and appears to be independent of the starting value $Q^{(0)}$. A more definite trend is established in B-3(a) for the l_∞ norm and B-5 for the l_1 norm, where in both examples the algorithm oscillates between two good approximations.

The special nature of the given functions in B-3 and B-4 causes Loeb's algorithm to break down almost completely in the l_1 and l_∞ norms. The functions which are approximated are pairs of constant functions, and here there is a tendency for Loeb's algorithm to produce successive approximations which possess poles in the interval of interest. If they occur close to abscissae from the discrete point set, these poles may have a disruptive effect on the sequence of approximations, since the successive weight functions then attain very large values. In particular, consider the case B-3(a). Since $f(x) = 1$ for almost half the points, and in each iteration the weighted quantity $||fQ - P||$ is minimized, a reasonable strategy for the algorithm is to set $P \equiv Q$ and minimize $||Q||$ over the remaining points. In the l_1 norm this is particularly appropriate since a best approximation tends to interpolate as many points of the given function as possible. Moreover, with interpolation at just three points where $f(x) = 1$, it follows that the quadratics P and Q must be identically equal. The problem then reduces to minimizing the quantity $||Q||$, and this minimum is achieved by a function which varies very little from the quadratic Chebyshev polynomial of the second kind for the interval $[.5, 1]$. Therefore, Q has two roots, and the approximation thus produced has two poles, in this interval. In the next iteration this weight function causes P and Q to coincide on $(.5, 1]$, thus producing a new approximation with poles in $[0, .5)$. The algorithm is thus continually thrown off track and often fails to converge. This argument can also be applied in the l_∞ norm, where there is a good possibility that the characteristic equioscillation property of the curve $(fQ - P)$ will occur at least four times when $f(x) = 1$. This implies that $(Q - P)$ has at least three zeros and hence, $Q \equiv P$. The optimum choice for Q in this norm is a function closely resembling the quadratic Chebyshev polynomial of the first kind, and again poles appear in the interval of interest. The algorithm seems able to avoid these pitfalls in the l_2 norm; this is possibly due to the less stereotyped behavior of best approximations in this norm.

Further disturbing results are B-3(b) for the l_1 norm and B-4 for the l_∞ norm. Here, in both cases Loeb's algorithm converges to approximations which are definitely not best approximations. The rational function produced in B-3(b) has an error of approximation that is almost optimal, and in fact there are two distinct best l_1 approximations for this problem. (This case was previously reported in [2] and some further details can be found therein.) The function produced for B-4 is a poor l_∞ approximation, being almost zero everywhere except around two poles enclosing the point $x = 0.85$.

Tables 1, 2, and 3 together emphasize the fact that Loeb's algorithm almost always seems to converge in the l_2 norm. We feel therefore that the user would be well advised to avoid the l_1 and l_∞ norms in those instances where the choice of norm is unimportant. Even the failure to converge in the l_2 example C-6 is not too disturbing, since the even function $P_2(x)/Q_2(x)$ is obviously an inappropriate choice of approximating form for the odd function $f(x) = \sin x$ on $[-3, 3]$.

Tables 4A and 4B clearly demonstrate that Appel's algorithm can provide very accurate approximations. The assumption inherent in this method, that the error of approximation $\epsilon(x)$ at any discrete point be small compared to the given function $f(x)$, may not be too restrictive in practice. Indeed, in some applications of the algorithm this assumption has been violated and yet good approximations have still been produced. However, in the case D-4 we originally included two extra data points with small ordinates, and here Appel's algorithm produced approximations far less satisfactory than those shown. A technique which can accommodate situations like this, where the error of approximation $\epsilon(x)$ is too large at some points, is to introduce an additional weight function that reduces $\epsilon(x)$ on these critical points. The choice of an appropriate additional weight function is discussed in Mason [5].

6. Summary. Loeb's algorithm is simple to implement, and it often converges in just a few iterations. Sometimes, however, it does not converge at all, and, even when it does converge, there is no guarantee that convergence is to a best approximation. In spite of these weaknesses the algorithm will often produce quite satisfactory rational approximations, particularly in the l_2 norm.

Appel's algorithm requires only one application of a linear algorithm, it computes near-best approximations, and it appears to be reliable and capable of producing very accurate results.

Department of Mathematics
University of Victoria
Victoria, British Columbia
Canada

Department of Computer Science
University of Toronto
Toronto, Ontario
Canada

1. K. APPEL, "Rational approximation of decay-type functions," *Nordisk Tidskr. Informationsbehandling*, v. 2, 1962, pp. 69–75.
2. I. BARRODALE, "On computing best L_1 approximations," *Approximation Theory*, edited by A. Talbot, Academic Press, London, 1970, pp. 205–215.

3. I. BARRODALE & A. YOUNG, "Algorithms for best L_1 and L_∞ linear approximations on a discrete set," *Numer. Math.*, v. 8, 1966, pp. 295–306. MR 33 #5096.

4. H. L. LOEB, *On Rational Fraction Approximations at Discrete Points*, Convair Astronautics, Math. Preprint #9, 1957.

5. J. C. MASON, *Some New Approximations for the Solution of Differential Equations*, Doctoral Thesis, Oxford, 1965.

6. J. R. RICE, *The Approximation of Functions*. Vol. 2, Addison-Wesley, Reading, Mass., 1969.

7. J. R. RICE & J. S. WHITE, "Norms for smoothing and estimation," *SIAM Rev.*, v. 6, 1964, pp. 243–256. MR 29, #5334.

8. L. WITTMEYER, "Rational approximation of empirical functions," *Nordisk Tidskr. Informationsbehandling*, v. 2, 1962, pp. 53–60.