ALGORITHMS FOR RATIONAL CHEBYSHEV APPROXIMATION

by

CHEE-MOU LEE

A THESIS SUBMITTED IN PARTIAL FULFILLMENT
OF THE REQUIREMENTS FOR THE DEGREE OF

MASTER OF SCIENCE

in the Department

of

Mathematics

Accepted for the Faculty
of Graduate Studies
[redacted]
Dean pro tem
12 May, 1971

We accept this thesis as conforming
to the required standard

[redacted]

UNIVERSITY OF VICTORIA

April 1971

ABSTRACT

Supervisor: Dr. F.D.K. Roberts

A common problem in numerical analysis is to obtain a convenient approximation to a prescribed function. Rational approximating functions are frequently selected for this purpose. This thesis is a survey of several methods that have been proposed for obtaining best rational Chebyshev approximations to functions defined on finite point sets. Most of the methods formulate the problem as a mathematical programming problem. Numerical examples are given to compare the efficiency of the methods.

Examiners: [redacted]

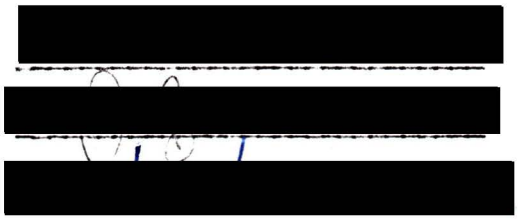

TABLE OF CONTENTS

LIST OF TABLES

ACKNOWLEDGEMENTS

I wish to express my sincere gratitude to my supervisor Dr F.D.K. Roberts for his guidance and encouragement. I also wish to thank the members of the Department of Mathematics for their assistance, and Mrs. L. Stewart for her careful typing of the final manuscript.

CHAPTER I

INTRODUCTION

The problem of determining a rational approximation to a continuous function has received much attention in the literature. The motivation for studying rational approximations is the problem of representing functions efficiently by easily computed expressions. The rational functions

$$R(x) = \frac{P_n(x)}{Q_m(x)} = \frac{p_0 + p_1 x + \ldots + p_n x^n}{q_0 + q_1 x + \ldots + q_m x^m}$$

have been found to be very useful for this purpose. The curve fitting ability of $R(x)$ is roughly the same as that of a polynomial of degree $n+m$. However, $R(x)$ can be evaluated with at most $\max(m,n)$ long operations (multiplications and divisions) whereas a polynomial of degree $n+m$ usually requires $m+n$ multiplications (Cheney, 1966, p. 151). A rational function $R(x)$ can be converted to a continued fraction by performing a series of divisions. For example,

$$R(x) = \frac{2x^4 - 4x^3 - 2x^2 + 12x - 4}{x^3 - 2x^2 - x + 5}$$

$$= 2x + \frac{2x - 4}{x^3 - 2x^2 - x + 5}$$

$$= 2x + \frac{2}{(x^3 - 2x^2 - x + 5)/(x-2)}$$

$$= 2x + \frac{2}{x^2 - 1 + 3/(x-2)}$$

This expression can be evaluated with two multiplications and two divisions. Thus, if division and multiplication are equally time-consuming, a rational function is preferable to a polynomial approximation.

The early attempts to obtain rational approximations were concerned with Padé approximations and continued fractions. The Padé approximation problem is, for given m and n, to choose P_n and Q_m so that $f(x)$ and $P_n(x)/Q_m(x)$ are equal at $x = 0$ and have as many derivatives as possible equal at $x = 0$. Here we assume that the interval of interest contains zero. If it does not, a simple change of variable will transform the interval to contain zero. The problem is then solved by expanding $f(x)$ in its MacLaurin series and comparing the coefficients with those of $R(x)$. Some Padé approximations to the function e^x are listed below:

$$\frac{P_4(x)}{Q_0(x)} = 1 + x + \frac{1}{2}x^2 + \frac{1}{6}x^3 + \frac{1}{24}x^4$$

$$\frac{P_3(x)}{Q_1(x)} = \frac{24 + 18x + 6x^2 + x^3}{24 - 6x}$$

$$\frac{P_2(x)}{Q_2(x)} = \frac{12 + 6x + x^2}{12 - 6x + x^2}$$

$$\frac{P_0(x)}{Q_4(x)} = \frac{1}{1 - x + \frac{1}{2}x^2 - \frac{1}{6}x^3 + \frac{1}{24}x^4} \quad .$$

The disadvantage of Padé approximations is that although they are good approximations close to the origin, the approximations are generally poor at points distant from the origin.

Another approach is the use of a continued fraction. Given a function $f(x)$ and a set $\{x_1, x_2, \ldots, x_N\}$, we define functions $\nu_r(x)$ by the recurrence relations.

$$\nu_r(x) = \nu_r(x_r) + \frac{x - x_r}{\nu_{r+1}(x)} \quad ,$$

$$\nu_0(x) = f(x) \quad .$$

Thus we obtain

$$f(x) = \nu_0(x)$$

$$= \nu_0(x_0) + \frac{x - x_0}{\nu_1(x)}$$

$$= \nu_0(x_0) + \cfrac{x - x_0}{\nu_1(x) + \cfrac{x - x_1}{\nu_2(x)}} \quad , \text{ etc.}$$

If we terminate this process after N terms, we obtain the continued fraction approximation

$$\nu_0(x_0) + \cfrac{x - x_0}{\nu_1(x_1) + \cfrac{x - x_1}{\nu_2(x_2) + \ddots + \cfrac{x - x_{N-1}}{\nu_N(x_N)}}} \quad .$$

For convenience, we rewrite this in the more compact form

$$\nu_0(x_0) + \frac{x - x_0}{\nu_1(x_1) +} \; \frac{x - x_1}{\nu_2(x_2) +} \cdots \frac{x - x_{N-1}}{\nu_N(x_N)} \quad .$$

This continued fraction interpolates $f(x)$ at the points $x_1, x_2, \ldots, x_N$. It can be rearranged as a rational function. However, for computational purposes, it is more efficient to retain it in a continued fraction form. The constants $\nu_r(x_r)$ are calculated by the recurrence relation

$$\nu_{r+1}(x) = \frac{x - x_r}{\nu_r(x) - \nu_r(x_r)} \quad .$$

If we let all the points x_i tend to zero, we obtain the continued fraction approximation (Handscomb, 1966, p. 114)

$$c_0 + \frac{x}{c_1 +} \; \frac{x}{c_2 +} \cdots \frac{x}{c_n}$$

where $\nu_r(x_r) \to C_r$ as $x_r \to 0$. As an example, let $f(x) = e^x$. The constants C_r are calculated to be

$$C_0 = 1 \ , \ C_1 = 1 \ .$$

$$C_{2r} = (-1)^r 2$$

$$C_{2r+1} = (-1)^r (2r + 1) \ , \ r = 1,2,\ldots \ .$$

We thus have the continued fraction approximation for e^x,

$$1 + \frac{x}{1+} \frac{x}{-2+} \frac{x}{-3+} \frac{x}{2+} \frac{x}{5+} \cdots \ .$$

The Chebyshev rational approximation problem is to determine a rational approximation which best approximates a given continuous function in the Chebyshev sense, that is in the sense of minimizing the maximum absolute difference between the function and the rational approximation.

The Chebyshev rational approximation problem can be stated as follows:

Given m,n and $f(x) \in C[a,b]$, determine polynomials $P_n(x)$ and $Q_m(x)$ which minimize

$$\max_{x \in [a,b]} |f(x) - P_n(x)/Q_m(x)| \ .$$

The problem was first discussed by Chebyshev (1859). Existence of a best rational approximation was proved by Walsh (1931). The characterization theorem was proved by Achieser (1930). This states

(with several restrictions on the function $f(x)$) that $P_n(x)/Q_m(x)$ is a best approximation to $f(x)$ if and only if the error curve $f(x) - P_n(x)/Q_m(x)$ alternates at least $m+n+2$ times. Uniqueness of best approximation was also shown by Achieser (1947).

The early algorithms for determining the best rational Chebyshev approximation were the Remes first and second algorithms.

The first algorithm is as follows:

At the kth stage, we are given a finite subset $X^k \subset [a,b]$. We then determine $R(x)$ to minimize

$$\max_{x \in X^k} \{|f(x) - R(x)|\} .$$

A point $x^k \in [a,b]$ is chosen such that $|f(x^k) - R(x^k)| \geq |f(x) - R(x)|$ for all $x \in [a,b]$. The method continues with $X^{k+1} = X^k \cup \{x^k\}$. The algorithm terminates when two successive approximations are in close agreement.

The second algorithm, which is based on the characterization theorem, is described fully in Chapter III.

More recently, the discrete rational Chebyshev problem has been investigated. In practical computation, it is often convenient to replace the interval $[a,b]$ by a finite set of points and then determine the best approximation on that finite point set. The discrete problem is usually more tractable computationally than the continuous one.

The discrete rational Chebyshev problem can be stated as follows:

Given m,n and $f(x)$ defined on $X = \{x_1, x_2, \ldots, x_N\}$, determine $P_n(x)$ and $Q_m(x)$ which minimize

$$\max_{x_i \in X} |f(x_i) - P_n(x_i)/Q_m(x_i)|$$

Existence of a best approximation is not guaranteed in general for the discrete problem (see example in Chapter III).

Thus far, we have replaced the interval [a,b] by the finite subset X and have considered the best approximation on X. However, we would like that solution which not only best approximates f(x) on X, but also remains pole-free throughout the interval [a,b]. That is, we require Q(x) to be of one sign on [a,b]. With this restriction, the uniqueness of best approximations can be proved. The discrete problem has the same characterization theorem as the continuous problem.

The purpose of this thesis is to compare seven algorithms which have been proposed for solving the discrete Chebyshev rational problem. Since several of these techniques involve linear programming, in Chapter II we discuss the general linear programming problem and the revised simplex method for solving linear programming problems. Chapter III describes the algorithms for determining best rational Chebyshev approximations. Chapter IV includes the numerical results and compares the effectiveness of each of these algorithms.

CHAPTER II

LINEAR PROGRAMMING

II - 1. Introduction.

Since the development of the simplex method for linear programming in 1948, many problems in approximation theory have been solved by this technique (see for example Barrodale and Young (1966), Rabinowitz (1968), (1970), Stiefel (1960)).

As an example we consider the linear Chebyshev approximation problem. Let $\{(x_i,f_i) \mid i = 1,2,\ldots,N\}$ be a discrete data set which we wish to approximate. For a set $\{\phi_j(x)\}$ of n real-valued continuous functions, we form a linear approximating function

$$F(A,x) = \sum_{j=1}^{n} a_j \phi_j(x)$$

where A is the set $\{a_1,a_2,\ldots,a_n\}$. The linear Chebyshev approximation problem is to determine A^* such that

$$\max_{x_i} |F(A^*,x_i) - f_i| \leq \max_{x_i} |F(A,x_i) - f_i|$$

for all A.

This can be transformed into a linear programming problem as follows:

Let $w = \max_{x_i} |F(A,x_i) - f_i|$. The problem then becomes that of minimizing w subject to the linear constraints

$$a_1 \phi_1(x_i) + \ldots + a_n \phi_n(x_i) - w \leq f_i$$

$$-a_1 \phi_1(x_i) - \ldots - a_n \phi_n(x_i) - w \leq -f_i$$

$$i = 1,2,\ldots,N$$

Note that the a_j's are not restricted in sign. Computationally, it is more convenient to solve the dual problem, since this involves fewer constraints than the primal. The dual problem can be stated as follows:

$$\text{Minimize} \quad \sum_{i=1}^{N} (s_i - t_i) f_i$$

subject to

$$\sum_{i=1}^{N} \phi_j (x_i)(s_i - t_i) = 0$$

$$j = 1,2,\ldots,n$$

$$\sum_{i=1}^{N} (s_i + t_i) = 1$$

$$s_i \geq 0 \ , \ t_i \geq 0 \quad \text{for} \quad i = 1,2,\ldots,N \ .$$

The solution to the primal problem can be obtained from the final simplex tableau of the dual

Similarly the Chebyshev rational approximation problem can be converted to a nonlinear programming problem. Now the approximating function is

$$R(x) = \frac{P_n(x)}{Q_m(x)} = \frac{\sum_{j=0}^{n} p_j x^j}{\sum_{j=0}^{m} q_j x^j} .$$

If we let $w = \max_{x_i} \left| \frac{P_n(x_i)}{Q_m(x_i)} - f_i \right|$, the problem then becomes

minimize w

subject to

$$\frac{P_n(x_i)}{Q_m(x_i)} - w \leq f_i$$

$$- \frac{P_n(x_i)}{Q_m(x_i)} - w \leq -f_i \quad , \quad i = 1,2,\ldots,N .$$

Several of the methods we discuss in this thesis attempt to convert this nonlinear programming problem into a sequence of linear programming problems.

In the remainder of this chapter we discuss the theory of linear programming and also the simplex and revised simplex algorithms for solving linear programming problems.

II - 2. Linear Programming.

Let

$$X = \begin{pmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{pmatrix} \quad , \quad c = \begin{pmatrix} c_1 \\ c_2 \\ \vdots \\ c_n \end{pmatrix} \quad , \quad b = \begin{pmatrix} b_1 \\ b_2 \\ \vdots \\ b_m \end{pmatrix}$$

and

$$A = \begin{pmatrix} a_{11} & \cdots & a_{1n} \\ \cdot & & \cdot \\ \cdot & & \cdot \\ \cdot & & \cdot \\ a_{m1} & \cdots & a_{mn} \end{pmatrix} \quad , \quad n > m \ .$$

The general linear programming problem is to find a vector X that minimizes the objective function

$$c'X \qquad \text{(II - 1)}$$

subject to the linear constraints

$$AX = b \qquad \text{(II - 2)}$$

and

$$X \geq 0 \ . \qquad \text{(II - 3)}$$

If we let P_j $(j = 1,2,\ldots,n)$ be the j^{th} column of matrix A and

$P_0 = b$, then the problem can be rewritten as

minimize $c'X$

subject to

$$x_1P_1 + x_2P_2 + \ldots + x_nP_n = P_0 \qquad \text{(II - 4)}$$

$$X \geq 0 \quad .$$

Definition. A vector X which satisfies (II-2) and (II-3) is called a feasible solution to the linear programming problem.

Definition. A basic solution for (II-3) is a solution obtained by setting (n-m) variables equal to zero and solving for the remaining m variables, provided the corresponding matrix is non-singular.

Definition. A basic feasible solution is a basic solution which is also feasible, i.e. nonnegative.

Let K be the set of all feasible solutions to a linear programming problem. The following theorems are stated without proof. The proofs can be found in Gass (1969).

Theorem. K is a convex set.

Theorem. If there exists a feasible solution to (II-2) and (II-3), then there exists a basic feasible solution.

Theorem. The objective function (II-1) assumes its minimum value at an extreme point of the convex set K.

Theorem. $X' = (x_1, x_2, \ldots, x_n)$ is an extreme point of K if and only if the vectors P_i associated with nonzero x_i are linearly independent. Therefore there are at most m values of x_i which are nonzero.

From the above theorems, we see that each basic feasible solution $X' = (x_1, x_2, \ldots, x_n)$ is an extreme point of K and vice versa. Therefore in searching for a minimum solution, we need only investigate the extreme points of the set K.

The simplex procedure finds an initial extreme point and determines whether it is a minimum. If it is not, the procedure moves to a neighbouring extreme point whose objective function value is not greater than that of the previous one.

The computational procedure is as follows:
Suppose we are given an extreme point $X = (x_1, x_2, \ldots, x_m, 0, \ldots, 0)$, i.e.

$$x_1 P_1 + x_2 P_2 + \ldots + x_m P_m = P_0 \qquad \text{(II - 5)}$$

$$x_1, x_2, \ldots, x_m \geq 0$$

and $P_1, P_2, \ldots, P_m$ are linearly independent. The value of the objective function is given by

$$z_0 = c_1x_1 + c_2x_2 + \ldots + c_mx_m \quad .$$

Since $P_1P_2,\ldots,P_m$ form a basis for E_m, each vector P_j which is not in the basis can be written as a linear combination of the basis vectors. Thus we have

$$P_j = \sum_{i=1}^{m} x_{ij} P_i \qquad \text{(II - 6)}$$

Define the marginal cost of P_j to be

$$z_j - c_j = \sum_{i=1}^{m} x_{ij} c_i - c_j \quad .$$

If there is an index j such that $z_j - c_j > 0$, then we can introduce the vector P_j into the basis and eliminate a vector P_k where k is chosen such that $\frac{x_k}{x_{kj}} = \min_i \frac{x_i}{x_{ij}}$ for $x_{ij} > 0$. The objective function is reduced if the problem is not degenerate. If all $x_{ij} \leq 0$, the solution is unbounded.

The elimination is done as follows.

Multiply (II-6) by $\frac{x_k}{x_{kj}}$ and subtract from (II-5) to obtain

$$(x_1 - \frac{x_k}{x_{kj}} x_{1j})P_1 + (x_2 - \frac{x_k}{x_{kj}} x_{2j})P_2 + \ldots + (x_k - \frac{x_k}{x_{kj}} x_{kj})P_k + \ldots$$

$$+ (x_m - \frac{x_k}{x_{kj}} x_{mj})P_m + \frac{x_k}{x_{kj}} P_j = P_0$$

The vector P_k is then eliminated and all the other coefficients are greater than zero. Hence this forms a new basis. The objective function is

$$(x_1 - \frac{x_k}{x_{kj}} x_{1j})c_1 + \ldots + 0 \cdot c_k + \ldots + (x_m - \frac{x_k}{x_{kj}} x_{mn})c_m + \frac{x_k}{x_{kj}} c_j$$

$$= z_0 - \frac{x_k}{x_{kj}} (z_j - c_j) .$$

Since $\frac{x_k}{x_{kj}} (z_j - c_j) > 0$, the solution is improved.

Theorem. If for any feasible solution $X = (x_1, x_2, \ldots, x_m)$, the marginal costs $z_j - c_j \leq 0$ for all $j = 1,2,\ldots,n$, then X is an optimal solution.

The simplex method enables us to start with a basic feasible solution and to generate a set of new basic feasible solutions that converge to the optimal solution.

II - 3. The Revised Simplex Method.

Consider the linear programming problem

minimize $c'X$

subject to $AX = b$

and $X \geq 0$.

Let $B = (P_1, P_2, \ldots, P_m)$ be a basis of m-dimensional vectors. Then every other vector P_j not in the basis can be expressed as linear combination of vectors in B as

$$P_j = x_{1j}P_1 + x_{2j}P_2 + \ldots + x_{mj}P_m \quad ,$$

where $(x_{1j}, x_{2j}, \ldots, x_{mj}) = B^{-1}P_j$. The basic feasible solution corresponding to B is

$$X_0 = B^{-1}b \quad .$$

The marginal cost $z_j - c_j$ is given by

$$z_j - c_j = c'X_j - c_j$$

$$= c'B^{-1}P_j - c_j$$

This determines which vector is to be introduced into the basis. Hence, if at each stage, the corresponding B^{-1} is known, we may progress to an optimal solution.

The B^{-1} of each stage is calculated as follows:

Suppose we have $B = (P_1, P_2, \ldots, P_\ell, \ldots, P_m)$. At the next iteration, a vector P_k is introduced to replace P_ℓ. Let $\bar{B} = (P_1, P_2, \ldots, P_k, \ldots, P_m)$, $B^{-1} = (b_{ij})$ and $\bar{B}^{-1} = (\bar{b}_{ij})$. $\bar{b}_{ij}$ is given by the elimination formulae

$$\bar{b}_{ij} = b_{ij} - \frac{b_{ij}}{k_{\ell k}} x_{ik} \qquad \text{for } i \neq \ell \ ,$$

$$\bar{b}_{\ell j} = \frac{b_{\ell j}}{x_{\ell k}} \ .$$

This can be verified by direct multiplication of $\bar{B}^{-1} \cdot \bar{B}$.

In the revised simplex method only B^{-1} and the solution X_0 are transformed at each iteration.

The computational details which were developed by Dantzig (1953) and Orchard-Hays (1954) are as follows:

We first introduce a new variable x_{n+m+1} which is defined by

$$x_{n+m+1} = -c_1 x_1 - c_2 x_2 - \ldots - c_n x_n$$

i.e. the negative of the objective function. The linear programming problem can be restated as:

maximize x_{n+m+1}

subject to

$$AX = b$$

$$c_1x_1 + c_2x_2 + \ldots + c_nx_n + x_{n+m+1} = 0 \qquad \text{(II - 7)}$$

$$X \geq 0 \quad .$$

Note that x_{n+m+1} is not restricted in sign.

As in the simplex method, the procedure begins with a basis with identity matrix which corresponds to either real or artificial vectors. If it starts with an artificial basis, we must determine the feasibility of the problem. The method consists of two phases. Phase I determines the feasibility of the problem and Phase II develops the optimal solution.

We introduce one more constraint in Phase I, namely

$$a_{m+2,1}x_1 + a_{m+2,2}x_2 + \ldots + a_{m+2,n}x_n + x_{m+n+2} = b_{m+2} \quad ,$$

where

$$a_{m+2,j} = -\sum_{i=1}^{m} a_{ij} \quad , \quad j = 1,2,\ldots,n$$

$$b_{m+2} = -\sum_{i=1}^{m} b_i \quad .$$

Adding artificial variables $x_{n+1},x_{n+2},\ldots,x_{n+m}$ to (II-7), we obtain the following:

Maximize x_{n+m+1}

subject to

$$
\begin{aligned}
a_{11}x_1 + \ldots + a_{1n}x_n + x_{n+1} &= b_1 \\
a_{21}x_1 + \ldots + a_{2n}x_n + x_{n+2} &= b_2 \\
\vdots \qquad\quad \vdots \qquad\quad \ddots \qquad\quad \vdots & \qquad\qquad (II\text{-}8) \\
a_{m1}x_1 + \ldots + a_{mn}x_n + x_{n+m} &= b_m \\
c_1 x_1 + \ldots + c_n x_n + x_{n+m+1} &= 0 \\
a_{m+2,1}x_1 + \ldots + a_{m+2,n}x_n + x_{n+m+2} &= b_{m+2}
\end{aligned}
$$

Note that by summing the first m equations and the last one, we obtain

$$x_{n+1} + x_{n+2} + \ldots + x_{n+m} + x_{n+m+2} = 0 \quad .$$

Thus x_{n+m+2} is the negative sum of the artificial variables. Since $x_{n+i} \geq 0$ $(i = 1,2,\ldots,m)$, x_{n+m+2} cannot be positive.

In Phase I, we consider the problem:

maximize x_{n+m+2}

subject to the constraints in (II-8). x_{n+m+1} and x_{n+m+2} are not restricted in sign and they are always kept in the basis.

If the maximum value of x_{n+m+2} is zero, this implies that

$x_{n+i} = 0$ for all $i = 1,2,\ldots,m$. Therefore we have obtained a feasible solution for (II-8). If the maximum value of x_{n+m+2} is negative, then at least one of the artificial variables is still in the basis and hence there is no feasible solution for (II-8).

When we have obtained a feasible solution, we proceed with Phase II, where the problem is to maximize x_{n+m+1} subject to (II-8). The variable x_{n+m+1} is always kept in the basis.

At each iteration, we record only the variables in the basis, the values of these variables and the inverse matrix of the basis.

The computational detail is briefly described in the following:

$$\text{Let } \bar{A} = \begin{pmatrix} a_{11} & \cdots & a_{1n} \\ \vdots & & \vdots \\ a_{m1} & \cdots & a_{mn} \\ c_1 & \cdots & c_n \\ a_{m+2,1} & \cdots & a_{m+2,n} \end{pmatrix} .$$

The starting basis $B = (P_{n+1}, P_{n+2}, \ldots, P_{n+m}, P_{n+m+1}, P_{n+m+2})$ is an identity matrix, hence B^{-1} is also an identity matrix.

Since we are maximizing x_{m+n+2}, the cost coefficient in Phase I is $c_0' = (0,0,\ldots,0,1)$. Hence for $j = 1,2,\ldots,n$

$$\begin{aligned} z_j - c_{0j} &= z_j \\ &= c_0' B^{-1} \bar{A}_j \\ &= B^{-1}_{m+2} A_j \quad , \end{aligned}$$

where B_{m+2}^{-1} is the $(m+2)^{th}$ row vector of B^{-1} and $\bar{A}_j$ is the j^{th} column vector of $\bar{A}$. If $x_{m+n+2} = 0$, i.e. a feasible solution is obtained, we then transfer to Phase II. If $x_{m+n+2} < 0$ and all $z_j \geq 0$ for $j = 1,2,\ldots,n$, then x_{m+n+2} is at its maximum and therefore no feasible solution exists for (II-8). If there exists some j such that $z_j < 0$, then we choose k such that $z_k = \min_j z_j$. The variable x_k is to be introduced into the basis. The variable x_ℓ to be eliminated from the basis is determined by calculating $x_{ik} = B_i^{-1} \bar{A}_k$, for all x_i in the basis and then choosing ℓ such that

$$\frac{x_{\ell 0}}{x_{\ell k}} = \min_i \frac{x_{i0}}{x_{ik}} \quad , \quad \text{for} \quad x_{ik} > 0 \ .$$

The new solution is given by:

$$x_{i0}' = x_{i0} - \frac{x_{i0}}{x_{\ell k}} x_{ik} \qquad \text{for} \quad i \neq k \ ,$$

$$x_{k0}' = \frac{x_{\ell 0}}{x_{\ell k}}$$

The matrix $B^{-1} = (b_{ij})$ is transformed by

$$b_{ij}' = b_{ij} - \frac{b_{\ell j}}{x_{\ell k}} x_{ik} \quad , \quad \text{for} \quad i \neq \ell$$

$$b_{\ell j}' = \frac{b_{\ell j}}{x_{\ell k}} \ .$$

Phase I is repeated until either we determine there is no feasible solution for (II-6) or a feasible solution of (II-6) is obtained with no artificial variables in the basis.

In Phase II, we maximize x_{m+n+1} subject to (II-8). The marginal cost of each variable is given by:

$$z_j - c_j = z_j$$

$$= B_{m+1}^{-1} \bar{A}_j \qquad j = 1,2,\ldots,n$$

If all $z_j \geq 0$, then x_{m+n+1} is at its maximum value, i.e. we have obtained an optimal solution and the negative of x_{m+n+1} is the value of the original objective function.

If there exists a $z_j < 0$, then we choose $z_k = \min_j z_k$ the variable x_k is introduced into the basis. Similarly the variable x_ℓ to be eliminated from the basis is determined by calculating $x_{ik} = B_i^{-1} \bar{A}_k$ and then choosing ℓ such that $\frac{x_{\ell 0}}{x_{\ell k}} = \min_{x_{ik}>0} \frac{x_{i0}}{x_{ik}}$.

If all $x_{ik} \leq 0$, then the problem has an unbounded solution.

The new solution is given by:

$$x_{i0}' = x_{i0} - \frac{x_{i0}}{x_{\ell k}} x_{ik} \qquad \text{for } i \neq k \ ,$$

$$x_{k0}' = \frac{x_{\ell 0}}{x_{\ell k}} \ .$$

The matrix B^{-1} is transformed by

$$b_{ij}' = b_{ij} - \frac{b_{lj}}{x_{lk}} x_{ik} \qquad \text{for } i \neq l \ ,$$

$$b_{lj}' = b_{lj} / x_{lk} \ .$$

Phase II is repeated until either a finite optimal solution is obtained or an unbounded solution is determined.

We make the following observations:

1. If the problem

minimize $c'X$,

subject to

$$AX = b \ ,$$

$$X \geq 0 \ ,$$

is solved by the revised simplex method, the dual solution is contained in the $(m+1)^{th}$ row of the final tableau of B^{-1} with the sign changed.

2. Since the revised simplex method always deals with the original matrix, for problems whose matrix has a large number of zeros, the total amount of computational work is greatly reduced. In the simplex method, as the computation progresses, the zeros will be replaced by nonzero values.

In the revised simplex method, the amount of new information needed to be recorded at each stage is in general reduced, since we only r

t[illegible]lution vector and the inverse matrix. The simplex method has to record the whole tableau.

As the computation progresses, rounding errors will tend to increase. If these errors become too large, it is a simple matter to re-invert the current basis if the revised simplex method is used. This cannot be accomplished conveniently if the simplex method is used.

A FORTRAN subroutine RESIM for solving a linear programming problem by the revised simplex method is included in Appendix I.

CHAPTER III

ALGORITHMS FOR RATIONAL APPROXIMATION

III - 1. Introduction.

In this chapter we discuss seven methods which have been proposed for solving the rational Chebyshev approximation problem. Let

$$R(x) = \frac{P_n(x)}{Q_m(x)} = \frac{p_0 + p_1 x + \ldots + p_n x^n}{q_0 + q_1 x + \ldots + q_m x^m} \quad .$$

Given n, m and a function $f(x)$ defined on $[a,b]$, the continuous Chebyshev approximation problem is to determine $p_0, p_1, \ldots, p_n$, $q_0, q_1, \ldots, q_m$ such that

$$w = \max_{x \in [a,b]} \left| f(x) - \frac{P_n(x)}{Q_m(x)} \right|$$

is minimized. For the discrete approximation problem, the interval $[a,b]$ is replaced by a finite point set $X = \{x_1, x_2, \ldots, x_N\}$, where $x_1 < x_2 < \ldots < x_N$ and $X \subset [a,b]$. The problem of obtaining a best approximation on a discrete point set is usually easier to solve than the continuous problem. We shall only consider algorithms for obtaining best approximations in the discrete case.

Some basic properties about the problem are given below.

(i) Existence. Existence of a best approximation is not guaranteed in general. Consider for example the problem of determining a best rational approximation of the form $\frac{p_0}{q_0 + q_1 x}$ to the function $(0,1)$, $(1,0)$. Set $p_0 = q_0 = 1$ and q_1 arbitrarily large. Then $w = \max_{x \in X} |R(x) - f(x)|$ can be made arbitrarily small. However, no best approximation exists.

Each rational function $R(x)$ gives rise to an error function $\varepsilon(x) = R(x) - f(x)$. We denote the error function of the best approximation by $\varepsilon^*(x) = R^*(x) - f(x)$.

(ii) Characterization. Both the continuous and the discrete approximation problems have the following characterization theorem.

Theorem. Let

$$R^*(x) = \frac{P_n^*(x)}{Q_m^*(x)} = \frac{\sum_{j=0}^{n-\nu} p_{j+\nu} x^j}{\sum_{j=0}^{m-\mu} q_{j+\mu} x^j}$$

where $0 \leq \mu \leq m$, $0 \leq \nu \leq n$, $q_m, p_n \neq 0$ and $P_n^*(x) / Q_m^*(x)$ is irreducible. Then $R^*(x)$ is a best approximation to $f(x)$ if and only if the number of points in $[a,b]$ (or X) at which $\varepsilon^*(x) = R^*(x) - f(x)$ takes on its maximum value with alternating signs is not less than $(n+m+2-d)$, where $d = \min(\mu,\nu)$. (Ralston (1965, p. 297)).

In most of the practical applications, we have $\mu = \nu = 0$, that is, $E(x)$ has exactly $n+m+2$ extreme points.

(iii) Uniqueness. For the discrete approximation problem, we require that a solution be polefree in $[a,b]$, that is we assume $Q(x)$ does not change sign throughout the interval $[a,b]$.

The uniqueness is ensured by the following.

Theorem. If $R^*(x) = \dfrac{P_n^*(x)}{Q_m^*(x)}$ is a best approximation to $f(x)$ on X then

$$\max_{x\varepsilon X} |R^*(x) - f(x)| < \max_{x\varepsilon X} |R(x) - f(x)|$$

for all $R(x) = \dfrac{P_n(x)}{Q_m(x)} \neq R^*(x)$, (Rivlin (1969, p. 125)).

In the following sections of this chapter, we consider various methods for determinating best rational approximations. These methods can in general be divided into two types. The first type formulates the problem as a mathematical programming problem. The best approximations are then computed without reference to the characterization theorem. Methods of the second type determine the best approximation based on the characterization theorem.

III - 2. Loeb's Algorithm. (Weighted Minimax Algorithm).

The approximation problem is to minimize

$$\max_{x_i \epsilon X} |R(x_i) - f(x_i)| \qquad \text{(III - 1)}$$

Rewriting (III-1), we obtain

$$\max_{x_i \epsilon X} \frac{1}{|Q(x_i)|} |P(x_i) - f(x_i)Q(x_i)|$$

Loeb (1959) proposes the following iterative method:

At the k^{th} stage, determine $P^{(k)}(x)$ and $Q^{(k)}(x)$ which minimize

$$\max_{x_i \epsilon X} \frac{1}{|Q^{(k-1)}(x_i)|} |f(x_i)Q(x_i) - P(x_i)| .$$

The term $\frac{1}{|Q^{(k-1)}(x_i)|}$ acts as a weight factor. The algorithm terminates when two successive approximations are in close agreement.

We formulate this as a linear programming problem as follows:

Let $w = \max_{x_i \epsilon X} \frac{1}{|Q^{(k-1)}(x_i)|} |P(x_i) - f(x_i)Q(x_i)| .$

The linear programming problem is

minimize w

subject to the constraints

$$\frac{1}{|Q^{(k-1)}(x_i)|}\,(P(x_i) - f(x_i)Q(x_i)) - w \leq 0$$

$$\frac{-1}{|Q^{(k-1)}(x_i)|}\,(P(x_i) - f(x_i)Q(x_i)) - w \leq 0$$

$$i = 1,2,\ldots,N$$

In order to avoid the trivial solution $P^{(k)}(x) = Q^{(k)}(x) = 0$, we normalize $R(x)$ by setting $q_0 = 1$. Thus we obtain

minimize w

subject to

$$\frac{1}{|Q^{(k-1)}(x_i)|}\sum_{j=0}^{n} p_j x_i^{\,j} - \frac{f(x_i)}{|Q^{(k-1)}(x_i)|}\sum_{j=1}^{m} q_j x_i^{\,j} - w \leq \frac{f(x_i)}{|Q^{(k-1)}(x_i)|}$$

$$\frac{-1}{|Q^{(k-1)}(x_i)|}\sum_{j=0}^{n} p_j x_i^{\,j} + \frac{f(x_i)}{|Q^{(k-1)}(x_i)|}\sum_{j=1}^{m} q_j x_i^{\,j} - w \leq \frac{-f(x_i)}{|Q^{(k-1)}(x_i)|}$$

$$i = 1,2,\ldots,N$$

There are $m+n+2$ variables and $2N$ constraints. In practice, N is much larger than $n+m$, hence it is more efficient to solve the dual problem, which is

$$\text{minimize} \quad \sum_{i=1}^{N} \frac{f(x_i)}{|Q^{(k-1)}(x_i)|} (s_i - t_i)$$

subject to

$$\sum_{i=1}^{N} \frac{1}{|Q^{(k-1)}(x_i)|} x_i^j (s_i - t_i) = 0$$

$$j = 0,1,2,\ldots,n$$

$$\sum_{i=1}^{N} \frac{-f(x_i)}{|Q^{(k-1)}(x_i)|} x_i^j (s_i - t_i) = 0$$

$$j = 1,2,\ldots,m$$

$$\sum_{i=1}^{N} (s_i + t_i) = 1$$

$$s_i, t_i \geq 0$$

The revised simplex method is applied at each iteration of the procedure.

Conditions under which the algorithm is convergent are unknown. The algorithm may fail to converge, and may also converge to a non-best

approximation. This is reported in Barrodale and Mason (1970), and is also demonstrated in the test examples which we ran (see Chapter IV).

III - 3. The Linear Inequality Method.

The Chebyshev rational approximation problem can be stated in the following way.

Find the smallest value of w such that the following system of inequalities is consistent

$$w \geq \left| \frac{\sum_{j=0}^{n} p_j x_i^j}{\sum_{j=0}^{m} q_j x_i^j} - f(x_i) \right|$$

$$i = 1,2,\ldots,N .$$

If we restrict the denominator to be nonnegative, i.e. $\sum_{j=0}^{m} q_j x_i^j \geq 0$, $i = 1,2,\ldots,N$, the system of inequalities can be written in the form

$$- \sum_{j=0}^{n} p_j x_i^j + f(x_i) \sum_{j=0}^{m} q_j x_i^j - w \sum_{j=0}^{m} q_j x_i^j \leq 0$$

$$\sum_{j=0}^{n} p_j x_i^j - f(x_i) \sum_{j=0}^{m} q_j x_i^j - w \sum_{j=0}^{m} q_j x_i^j \leq 0$$

$$- \sum_{j=0}^{m} q_j x_i^j \leq 0$$

$$i = 1,2,\ldots,N$$

We normalize the rational function by setting $q_0 = 1$. The inequalities become

$$-\sum_{j=0}^{n} p_j x_i^j + (f(x_i) - w) \sum_{j=1}^{m} q_j x_i^j \leq -(f(x_i) - w)$$

$$\sum_{j=0}^{n} p_j x_i^j - (f(x_i) + w) \sum_{j=1}^{m} q_j x_i^j \leq (f(x_i) + w) \qquad \text{(III - 2}$$

$$-\sum_{j=1}^{m} q_j x_i^j \leq 1$$

$$i = 1,2,\ldots,N .$$

Let w^* be the value corresponding to the best approximation. If we substitute w^* in (III-2) and solve for $p_0,\ldots,p_n$, $q_1,\ldots,q_m$, we obtain the best approximation.

Now w^* lies between 0 and $\max_{x_i \in X} |f(x_i)|$, since $R = 0$ is an approximation with error $\max_{x_i \in X} |f(x_i)|$. In the linear inequality method we search in the interval $[\, 0, \max_{x_i \in X} |f(x_i)| \,]$ to obtain the value of w^*.

Initially we choose w to be the mid-point of the interval and test for consistency of the system of linear inequalities (III-2). This can be accomplished by using the revised simplex method. If the system is consistent, we restrict our search to the lower half of the interval. If it is inconsistent, we retrict our search to the upper half of the interval. The process is repeated until the relative width of the interval containing w^* is less than some prescribed accuracy.

At each iteration, if the system is consistent, a solution $P(x)/Q(x)$ is obtained which satisfies (III-2). As the value of w converges to w^*, $P(x)/Q(x)$ converges to the best approximation. This method is due to Loeb (1960).

In determining the consistency of the system (III-2), we have $(n+m+1)$ variables and $3N$ inequalities. In practice, we formulate this as a linear programming problem and solve the dual.

Consider the linear programming problem

Maximize the zero function subject to the linear system (III-2). For a given value of w, if the linear programming problem has a feasible solution then the system (III-2) is consistent and $P(x)/Q(x)$ can be obtained. If the linear programming problem has no feasible solution, then the system is inconsistent.

The dual problem of the linear programming problem can be stated as follows:

minimize

$$\sum_{i=1}^{N} -(f(x_i) - w)s_i + \sum_{i=1}^{N} (f(x_i) + w)t_i + \sum_{i=1}^{N} u_i$$

subject to

$$-\sum_{i=1}^{N} x_i^j t_i + \sum_{i=1}^{N} x_i^j s_i = 0$$

$$j = 0,1,\ldots,n$$

$$\sum_{i=1}^{N} (f(x_i) - w)x_i^j t_i + \sum_{i=1}^{N} (-f(x_i)-w)x_i^j s_i - \sum_{i=1}^{N} x_i^j u_i = 0$$

$$j = 1,2,\ldots,m$$

$$t_i,\ s_i,\ u_i \geq 0 \quad .$$

By the duality theorem of linear programming (see for example Gass (1969, p. 90)), if the dual has a finite solution, then (III-2) is consistent. If the dual has an unbounded solution then (III-2) is inconsistent. We note that the dual is always feasible since the zero vector satisfies the dual constraints. The number of iterations required to obtain the best approximation depends on the value $\max_{x_i \in X} |f(x_i)|$ and w^*. Let I be the number of iterations, c be the convergence criterion and h be the width of the final interval containing w^*. The algorithm is terminated when

$$\frac{h}{w^*} \leq c \quad .$$

Then we have

$$\frac{\max_{x_i \in X} |f(x_i)|}{2^I} \leq c\, w^*$$

which implies $2^I \geq (cw^*)^{-1} \max_{x_i \in X} |f(x_i)|$.

This method is guaranteed to generate the best approximation, but the convergence is only linear.

III - 4. Differential Correction Method.

Let $w(R) = \max_{x_i \in X} |R(x_i) - f(x_i)|$. Let $R^* = P^*/Q^*$ be the best approximation with deviation w^* and $Q^*(x_i) > 0$ for each $x_i \in X$. This algorithm generates a sequence $R^{(k)}$ such that $R^{(k)} \to R^*$.

The Algorithm is as follows:

Define $\delta_k(R) = \max_{x_i \in X} \{|f(x_i)Q(x_i) - P(x_i)| - w(R^{(k)})Q(x_i)\}$.

At the $(k+1)^{st}$ stage, determine $R^{(k+1)}$ to minimize δ_k with the normalization that the coefficients of $R^{(k+1)}$ lie in the standard cube, i.e. $|p_i| \leq 1$, $i = 0,1,\ldots,n$; $|q_j| \leq 1$, $j = 0,1,2,\ldots,m$. There is no loss of generality with this normalization, since

$$\frac{P(x)}{Q(x)} = \frac{\lambda P(x)}{\lambda Q(x)}$$

for $\lambda \neq 0$. The coefficients of the initial $R^{(0)}$ are also chosen to lie in the standard cube, with the denominator $Q^{(0)}$ chosen to be positive.

Lemma. For each k , either $R^{(k)}$ is the best approximation or $\delta_k(R^{(k+1)}) < 0$.

Proof. Suppose $R^{(k)}$ is not a solution. Then since $R^{(k+1)}$ minimizes $\delta_k(R)$,

$$\delta_k(R^{(k+1)}) \leq \delta(R^*)$$

$$= \max_{x_i \in X} \{Q^*(x_i)[|f(x_i) - R^*(x_i)| - w(R^{(k)})]\}$$

This expression is negative since $Q^*(x_i) > 0$ and $|f(x_i) - R^*(x_i)| < w(R^{(k)})$.

Corollary. If $P^{(k)}/Q^{(k)}$ is not the best approximation then $Q^{(k+1)}(x_i) > 0$ for each $x_i \in X$.

Proof. Suppose on the contrary that there exists an $x_0 \in X$ such that $Q^{(k+1)}(x_0) \leq 0$. Then

$$0 > \delta_k(R^{(k+1)})$$

$$= \max_{x_i \varepsilon X} \{|f(x_i)Q^{(k+1)}(x_i) - P^{(k+1)}(x_i)| - w(R^{(k)})Q^{(k+1)}(x_i)\}$$

$$\geq |f(x_0)\, Q^{(k+1)}(x_0) - P^{(k+1)}(x_0)| - w(R^{(k)})Q^{(k+1)}(x_0)$$

$$\geq 0$$

This is a contradiction which proves the corollary.

On the other hand, if there is an $x_0 \varepsilon X$ such that $Q^{(k+1)}(x_0) \leq 0$ then $P^{(k)}/Q^{(k)}$ is the best approximation.

Corollary. $\delta_k(R^{(k+1)}) = 0$ implies $R^{(k)}$ is the best approximation with $Q^{(k)}(x_i) > 0$ for each $x_i \varepsilon X$.

Proof.

$$0 = \delta_k(R^{(k+1)})$$

$$\leq \delta_k(R^*)$$

$$= \max_{x_i \varepsilon X} \{Q^*(x_i)\left[|f(x_i) - R^*(x_i)| - w(R^{(k)})\right]\}$$

$$\leq \max_{x_i \varepsilon X} \{Q^*(x_i)\}\left[w(R^*) - w(R^{(k)})\right]$$

Now we note that $Q^*(x_i) > 0$ and $\left[w(R^*) - w(R^{(k)})\right] \leq 0$ which implies that $\left[w(R^*) - w(R^{(k)})\right] = 0$, i.e. $w(R^*) = w(R^{(k)})$. Therefore $R^{(k)}$

is the solution.

Hence in the following proof, we can assume $Q^{(k+1)}(x_i) > 0$ and $\delta_k(R^{(k+1)}) < 0$.

Theorem. $w(R^{(k)})$ converges to $w(R^*)$ for $k = 1,2,\ldots$.

Proof.

We assume R^* exists and $Q^*(x_i) > 0$. Let

$$\alpha = \min_{x_i \in X} Q^*(x_i) \quad \text{and} \quad \beta = \sup_{|q_j| \leq 1} \max_{x_i \in X} Q(x_i) \ .$$

By assumption, α is greater than zero. We note that β is bounded since the standard cube is a compact set.

$$0 > \delta_k(R^{(k+1)})$$

$$= \max_{x_i \in X} \{Q^{(k+1)}(x_i)\left[|f(x_i) - R^{(k+1)}(x_i)| - w(R^{(k)})\right]\}$$

$$\geq \beta \max_{x_i \in X} \{|f(x_i) - R^{(k+1)}(x_i)| - w(R^{(k)})\}$$

$$\geq \beta \left[w(R^{(k+1)}) - w(R^{(k)})\right] \ .$$

However,

$$\delta_k(R^{(k+1)}) \leq \delta_k(R^*)$$

$$= \max_{x_i \in X} \{Q^*(x_i)[|f(x_i) - R^*(x_i)| - w(R^{(k)})]\}$$

$$\leq \max_{x_i \in X} \{Q^*(x_i)[w(R^*) - w(R^{(k)})]\}$$

$$\leq \alpha[w(R^*) - w(R^{(k)})]$$

Hence, $\beta[w(R^{(k+1)}) - w(R^{(k)})] \leq \alpha[w(R^*) - w(R^{(k)})]$

$$\Rightarrow \quad w(R^{(k+1)}) - w(R^{(k)}) \leq \frac{\alpha}{\beta}[w(R^*) - w(R^{(k)})]$$

$$\Rightarrow \quad w(R^{(k+1)}) - w(R^*) \leq (1 - \frac{\alpha}{\beta})[w(R^{(k)}) - w(R^*)]$$

Since $0 < \alpha \leq \beta$, we have $0 \leq (1 - \frac{\alpha}{\beta}) < 1$. Hence $w(R^{(k)})$ converges to $w(R^*)$ in at least a linear rate.

The computation can be accomplished by linear programming. At the $(k+1)^{st}$ stage, we wish to solve the following problem

minimize δ

subject to

$$f(x_i) \sum_{j=0}^{m} q_j x_i^j - \sum_{j=0}^{n} p_j x_i^j - w \sum_{j=0}^{m} q_j x_i^j - \delta \leq 0$$

$$-f(x_i) \sum_{j=0}^{m} q_j x_i^j + \sum_{j=0}^{n} p_j x_i^j - w \sum_{j=0}^{m} q_j x_i^j - \delta \leq 0$$

$$i = 1,2,\ldots,N$$

$$-q_j \leq 1$$

$$q_j \leq 1 \quad , \; j = 0,1,\ldots,m$$

$$-p_j \leq 1$$

$$p_j \leq 1 \quad , \; j = 0,1,\ldots,n$$

There are (m+n+3) variables in 2N+2(m+n+2) inequalities. In practice, it is more efficient to solve the dual problem.

The dual problem is

$$\text{minimize} \quad \sum_{j=0}^{m} s_j + \sum_{j=0}^{m} t_j + \sum_{j=0}^{n} u_j + \sum_{j=0}^{m} v_j$$

subject to

$$\sum_{i=1}^{N} (f(x_i) - w)x_i^j \, g_i - \sum_{i=1}^{N} (f(x_i) + w)h_i - s_j + t_j = 0$$

$$j = 0,1,2,\ldots,m$$

$$-\sum_{i=1}^{N} x_i^j g_i + \sum_{i=1}^{N} x_i^j h_i - u_j + v_j = 0$$

$$j = 0,1,2,\ldots,n$$

$$\sum_{i=1}^{N} g_i + \sum_{i=1}^{N} h_i = 1$$

$$g_i, h_i, s_j, t_j, u_j, v_j \geq 0$$

The revised simplex method is applied at each stage.

This algorithm was first discussed by Cheney and Loeb (1962).

III - 5. Modified Differential Correction Method.

In this method, the function $\delta_k(R)$ of the differential correction method is modified to be

$$\delta_k(R) = \max_{x_i \in X} \left\{ \frac{|f(x_i)Q(x_i) - P(x_i)| - w(R^{(k)})Q(x_i)}{Q^{(k)}(x_i)} \right\}$$

The computational procedure is similar to that in the differential correction method. At the $(k+1)^{st}$ stage, we choose $R^{(k+1)}$ in the standard cube to minimize $\delta_k(R)$. The algorithm then generates a sequence of approximations $R^{(k)}$, $k = 1,2,\ldots$, which converges to $R^* = P^*/Q^*$ with $Q^* > 0$.

The following theorems about the convergence of the method are proved in Barrodale, Powell and Roberts (1971).

Theorem. If $P^{(k)}(x)/Q^{(k)}(x)$ is not the best approximation, then

$$\max_{x_i \in X} |R^{(k+1)}(x_i) - f(x_i)| < \max_{x_i \in X} |R^{(k)}(x_i) - f(x_i)|$$

and $Q^{(k+1)}(x_i) > 0$ for all $x_i \in X$.

Theorem. The sequence of approximations $R^{(k)}(x)$ converges to $R^*(x)$.

Theorem. If $N \geq n+m+1$, if a best approximation exists, and if the best approximation is not defective, then the rate of convergence is at least quadratic.

The quadratic convergence property does not hold for the differential correction method. Thus the modified method is superior to the differential correction method. The rapid convergence is also illustrated in the numerical results in Chapter IV. This method was first proposed by Cheney and Loeb (1961).

III - 6. The Remes Algorithm

Consider the error function of the best approximation $R^*(x)$ defined by

$$\varepsilon^*(x) = R^*(x) - f(x) \ . \qquad \text{(III-3)}$$

The characterization theorem states that $\varepsilon^*(x)$ achieves its extreme value at least $(m+n+2)$ times with alternating sign and equal magnitude. In this section we assume that $\varepsilon^*(x)$ has exactly $(m+n+2)$ extreme points. The algorithm consists of two stages:

(1) Given a set of $(m+n+2)$ initial estimates of extreme points $Y = \{y_0, y_1, \ldots, y_{m+n+1}\} \subset X$, we calculate an approximation $R(x)$ such that its error function alternates $(m+n+2)$ times on Y and has the values $(-1)^\alpha \lambda$ at the points y_α, that is $f(y_\alpha) - R(y_\alpha) = (-1)^\alpha \lambda$, $\alpha = 0,1,\ldots,m+n+1$. By setting $q_0 = 1$, this is a system of $(m+n+2)$ non-linear equations with $(m+n+2)$ unknowns. We will discuss its solution later in this section.

(2) The extreme points of the error function $\varepsilon(x)$ corresponding to the $R(x)$ computed in stage 1 then yield new estimates of the extreme points of the best approximation. With this new set of points, we repeat stage 1 until the required accuracy is attained. Since we

consider only the discrete approximation, the new extreme points are easily obtained by checking through all the data points.

In solving the system of non-linear equations in stage 1, we apply an iterative method (Fraser and Hart (1962)), as follows:

We rewrite the equations in the form

$$\sum_{j=0}^{n} p_j y_\alpha^{\,j} - (f(y_\alpha) + (-1)^\alpha \lambda_{k-1}) \sum_{j=1}^{m} q_j y_\alpha^{\,j} - (-1)^\alpha \lambda_k = f(y_\alpha)$$

$$\alpha = 0,1,\ldots,m+n+1$$

For a given value of λ_{k-1}, this is a system of linear equations which can be solved for p_j, $j = 0,1,\ldots,n$, q_j, $j = 1,2,\ldots,m$ and λ_k. The procedure is repeated until two successive values of λ_k are in close agreement. Convergence to a solution without a pole is not ensured in this method.

In practical computation, if there are no better estimates for the initial extreme points, the extreme points of the $(m+n+1)^{st}$ Chebyshev polynomial can be used.

III - 7. Maehly's Method.

In this method, the error function is characterized by its zeros rather than by its extrema. Since $\varepsilon^*(x)$ alternates at least $m+n+2$ times, it has $m+n+1$ zeros $z_0^* < z_1^* < \ldots < z_{m+n}^*$ in the interval $[a,b]$. We shall assume that $\varepsilon^*(x)$ has exactly $m+n+1$ zeros. Therefore it may be written in the form

$$\varepsilon^*(x) = G(x) \prod_{k=0}^{m+n} (x - z_k^*)$$

where $G(x)$ is a positive function and z_k^* are the zeros.

The algorithm is composed of two stages,

(1) For an initial estimate of the zeros $Z = \{z_0, z_1, \ldots, z_{n+m}\}$ of the error function ε^*, we calculate an approximation $R(x)$ whose error function has zeros at z_k, that is

$$R(z_k) - f(z_k) = 0 \quad , \quad k = 0,1,\ldots,m+n \ .$$

This is a system of linear equations, therefore there is not much difficulty in solving for $R(x)$. However, if z_k are not close to z_k^*, it may fail to have a solution.

(2) With the approximation $R(x)$ from stage 1, we compute the extreme points of its error function and use these points to make a correction on the estimates of the zeros. We then return to stage 1.

The algorithm terminates when two successive approximations are in close agreement.

We now describe the method for correcting the zeros. First let us denote by $\varepsilon(x,Z)$ the error function with zeros $Z = \{z_0, z_1, \ldots, z_{m+n}\}$. Let $Y = \{y_0, y_1, \ldots, y_{m+n+1}\}$ be its extreme points. We wish to determine $\delta Z = \{\delta z_0, \delta z_1, \ldots, \delta z_{m+n}\}$, the corrections of the zeros, such that at the point y_α

$$\varepsilon(y_\alpha, Z+\delta Z) = (-1)^\alpha \lambda, \quad \alpha = 0,1,\ldots,m+n+1$$

Taking logarithms of the absolute values of both sides and expanding in the first term of Taylor's series, we obtain

$$\ell n|\varepsilon(y_\alpha,Z)| + \sum_{k=0}^{m+n} \left[\frac{\partial \ell nG}{\partial z_k} - \frac{1}{y_\alpha - z_k} \right] \delta z_k = \ell n|\lambda| \quad ,$$

$$\alpha = 0,1,\ldots,m+n+1$$

We make another assumption that the function $G(x)$ does not depend (very much) on the zeros z_k , i.e. $\frac{\partial \ell nG}{\partial z_k} = 0$, $k = 0,1,\ldots,m+n$.

We then obtain the following linear system of equations

$$\sum_{k=0}^{m+n} \frac{\delta z_k}{y_\alpha - z_k} = \ell n|\varepsilon(y_\alpha,Z)| - \ell n|\lambda|$$

$$\alpha = 0,1,\ldots,m+n+1 \quad .$$

Eliminating $|\lambda|$ by subtracting the equations with $\alpha = 0$, we have

$$\sum_{k=0}^{m+n} \left[\frac{1}{y_\alpha - z_k} - \frac{1}{y_0 - z_k} \right] \delta z_k = \ell n \left| \frac{\varepsilon(y_\alpha,Z)}{\varepsilon(y_0,Z)} \right|$$

$$\alpha = 1,\ldots,m+n+1 \quad .$$

Using the approximation

$$\ell n \left| \frac{\varepsilon(y_\alpha,Z)}{\varepsilon(y_0,Z)} \right| \simeq 2 \frac{|\varepsilon(y_\alpha,Z)| - |\varepsilon(y_0,Z)|}{|\varepsilon(y_\alpha,Z)| + |\varepsilon(y_0,Z)|}$$

we obtain the following system

$$\sum_{k=0}^{m+n} \frac{(y_0-y_\alpha)}{(y_\alpha-z_k)(y_0-z_k)}\,\delta z_k = 2\,\frac{|\varepsilon(y_\alpha,Z)| - |\varepsilon(y_0,Z)|}{|\varepsilon(y_\alpha,Z)| + |\varepsilon(y_0,Z)|}$$

$$\alpha = 1,\ldots,m+n+1$$

This system of linear equations is well conditioned for the largest elements are close to the diagonal.

In practical applications, when the data to be approximated is smooth, the assumption made on the function $G(x)$ is quite satisfactory. This method is due to Maehly (1963).

If there are no better estimates for the initial zeros, the zeros of the $(m+n+1)^{st}$ Chebyshev polynomial can be used.

III - 8. A Linearization Method

In this method, we calculate the best rational approximation as follows.

Suppose we are given an approximation $P(x)/Q(x)$, we then calculate a better approximation by determining the corrections $\delta P(x)$ and $\delta Q(x)$ such that the function

$$\max_{x_i \varepsilon X} \left| f(x_i) - \frac{P(x_i) + \delta P(x_i)}{Q(x_i) + \delta Q(x_i)} \right| \qquad \text{(III-4)}$$

is minimized.

However, (III-4) is nonlinear in $\delta Q(x)$. We shall replace it by a linear expression which approximates (III-4). This is done by expanding the denominator of (III-4) and keeping only the linear terms.

$$\left| f - \frac{P + \delta P}{Q + \delta Q} \right|$$

$$= \left| f - \frac{P + \delta P}{Q} \left(1 + \frac{\delta Q}{Q}\right)^{-1} \right|$$

$$\simeq \left| f - \frac{P + \delta P}{Q} \left(1 - \frac{\delta Q}{Q}\right) \right|$$

$$\simeq \left| f - \frac{P}{Q} + \frac{P\delta Q - Q\delta P}{Q^2} \right| \quad .$$

The algorithm can be stated as follows:

At the k^{th} iteration, determine polynomials $\delta P(x)$ and $\delta Q(x)$ which minimize

$$\max_{x_i \in X} \left| f(x_i) - \frac{P^{(k-1)}(x_i)}{Q^{(k-1)}(x_i)} + \frac{P^{(k-1)}(x_i)\delta Q(x_i) - Q^{(k-1)}(x_i)\delta P(x_i)}{\left[Q^{(k-1)}(x_i)\right]^2} \right|$$

The k^{th} approximation $P^{(k)}(x)/Q^{(k)}(x)$ is given by

$$\frac{P^{(k)}(x)}{Q^{(k)}(x)} = \frac{P^{(k-1)}(x) + \lambda\delta P(x)}{Q^{(k-1)}(x) + \lambda\delta Q(x)} \quad ,$$

where $\lambda \in \{-1(0.02)1\}$ is chosen such that

$$\max_{x_i \in X} \left| f(x_i) - \frac{P^{(k-1)}(x_i) + \lambda \delta P(x_i)}{Q^{(k-1)}(x_i) + \lambda \delta Q(x_i)} \right|$$

is minimized.

The determination of $P(x)$ and $Q(x)$ at each iteration can be formulated as a linear programming problem. Let

$$w = \max_{x_i \in X} \left| f(x_i) - \frac{P^{(k-1)}(x_i)}{Q^{(k-1)}(x_i)} + \frac{P^{(k-1)}(x_i)\delta Q(x_i) - Q^{(k-1)}(x_i)\delta P(x_i)}{\left[Q^{(k-1)}(x_i)\right]^2} \right| .$$

Then the linear programming problem is

minimize w

subject to

$$\frac{P^{(k-1)}(x_i)}{\left[Q^{(k-1)}(x_i)\right]^2} \sum_{j=1}^{m} \delta q_j x_i^j - \frac{1}{Q^{(k-1)}(x_i)} \sum_{j=0}^{n} \delta p_j x_i^j - w \leq -f(x_i) + \frac{P^{(k-1)}(x_i)}{Q^{(k-1)}(x_i)}$$

$$- \frac{P^{(k-1)}(x_i)}{\left[Q^{(k-1)}(x_i)\right]^2} \sum_{j=1}^{m} \delta q_j x_i^j + \frac{1}{Q^{(k-1)}(x_i)} \sum_{j=0}^{n} \delta p_j x_i^j - w \leq f(x_i) - \frac{P^{(k-1)}(x_i)}{Q^{(k-1)}(x_i)}$$

$$i = 1,2,\ldots,N .$$

Note that we have normalized the rational approximation by setting $q_0 = 1$; i.e. $\delta q_0 = 0$.

In practice, it is more efficient to solve the dual problem, which is

$$\text{minimize} \quad \sum_{i=1}^{N} \left(-f(x_i) + \frac{P^{(k-1)}(x_i)}{Q^{(k-1)}(x_i)}\right)(s_i - t_i)$$

subject to

$$\sum_{i=1}^{N} \frac{P^{(k-1)}(x_i)}{\left[Q^{(k-1)}(x_i)\right]^2} x_i^j (s_i - t_i) = 0$$

$$j = 1,2,\ldots,m$$

$$\sum_{i=1}^{N} \frac{-x_i^j}{Q^{(k-1)}(x_i)} (s_i - t_i) = 0$$

$$j = 0,1,2,\ldots,n$$

$$\sum_{i=1}^{N} (s_i + t_i) = 1$$

$$s_i, t_i \geq 0$$

Unfortunately, the method does not always converge. Usually it converges if the starting approximation is sufficiently good. The convergence rate is usually very fast when the method does converge.

CHAPTER IV

NUMERICAL RESULTS AND COMMENTS

IV - 1. Introduction.

To compare the algorithms numerically, we selected ten sets of data and approximated each set of data by five rational functions. The data sets are given in Table I. Each data set consists of 21 points with abscissae equally spaced over the interval [a,b]. The approximating functions used are P_1/Q_1 , P_2/Q_2 , P_1/Q_3 , P_4/Q_2 , and P_0/Q_2 , where the subscripts denote the degrees of the polynomials. The rational functions which give the best approximations to each data sets are listed in Tables II-VI. The number of iterations required for the algorithms to converge in each case is given in Tables VII-XIII. The convergence criterion being that the relative change in the error of two successive approximations is less than 10^{-7} . The entire study was performed in double precision arithmetic on an IBM 360/44.

IV - 2. Comments on the Results.

(1) Loeb's Algorithm.

The results in Table VII show that Loeb's algorithm is quite a satisfactory technique when applied to smooth data sets. It usually

converges in about 10 iterations. However, the convergence is not guaranteed in general. For example, when the algorithm is applied to determine best P_2/Q_2 and P_1/Q_3 approximations to the function $\sin x$ in the interval $[-3,3]$, there is no sign of convergence in the first fifty iterations. Also for the case where

$$f(x) = \begin{cases} e^x & , x \in [0,1] \\ e^{-x} - e^{-1} + e & , x \in (1,2] \end{cases}$$

is approximated by P_1/Q_1, the algorithm oscillates between two non-best approximations.

Another serious disadvantage of the method is that even when the algorithm does converge, it may not converge to the best approximation. This is illustrated by the following example using the approximation P_1/Q_3,

$$f(x) = \begin{cases} e^x & , x \in [0,1] \\ e^{-x} - e^{-1} + e & , x \in (1,2] \end{cases}$$

The algorithm converges to

$$\frac{1.16520 - 1.15480x}{1 - 1.7778x + 1.04815x^2 - 0.26627x^3},$$

which is not the best approximation, for its error function does not alternate a sufficient number of times. Other examples for which the algorithm converges to a non-best approximation are given in Table VII.

Conditions which are sufficient for this algorithm to converge to the best approximation are not known. From Table VII, it seems that this depends on the data set as well as the approximating function used. For smooth data sets, this algorithm appears to be quite satisfactory.

(2) The Linear Inequality Method.

For the linear inequality method, a solution without a pole can always be obtained. The number of iterations required depends on $\max_{x_i \epsilon X} |f(x_i)|$ and the degree of the rational approximating function. They are listed in Table VIII. Let

$$w^* = \min_{P_n, Q_m} \max_{x_i \epsilon X} \left| \frac{P_n(x_i)}{Q_m(x_i)} - f(x_i) \right| .$$

We know that w^* lies in the interval $[0, \max_{x_i \epsilon X} |f(x_i)|]$. The algorithm applies the method of bisection to search for w^* in this interval. The algorithm terminates when the interval containing w^* is less than $10^{-6} \times w^*$. We encountered no numerical difficulties with this algorithm. however, a disadvantage of the method is the large number of iterations required to obtain a solution.

(3) The Differential Correction Method and the Modified Differential Correction Method.

The differential correction algorithm is guaranteed to converge to the best approximation from any starting approximation. However, the

convergence rate is sometimes very slow. For example, when the function P_2/Q_2 is used to approximate $\sqrt{x}$ in the interval $[0,1]$, the algorithm takes 138 iterations to converge. From Table IX we note that when the degree of the approximating function is high, the convergence rate is slow. In particular, the convergence is poor for the approximating function P_4/Q_2. For all the unsmooth data sets, the solutions generated before 100 iterations are not close to the best approximations, while the results for P_0/Q_2 and P_1/Q_1 are quite satisfactory.

The modified differential correction algorithm converges with a much faster rate. All solutions are obtained within ten to fifteen iterations. It is proved in Barrodale, Powell and Roberts (1971) that the convergence rate is at least quadratic.

An advantage of these algorithms is that they do not require special handling for the nonstandard case, that is, for the case when the error function of the best approximation does not alternate exactly $(n+m+2)$ times.

To start the algorithms, we set $P_n/Q_m = 1/1$. However, these algorithms do not seem to depend very much on the starting value.

The fast convergence rate and easy handling make the modified differential correction method the most satisfactory algorithm among those we investigated in this thesis.

(4) Remes Algorithm.

In the Remes Algorithm, we take the extreme points of the $(m+n+1)^{st}$ Chebyshev polynomial as the initial estimates of the extreme

points of the error function. The algorithm is very efficient except in some cases where it converges to a solution which possesses a pole in the interval $[a,b]$.

The main computational difficulty of this method is in solving the following system of nonlinear equations

$$f(x_i) - \frac{P_n(x_i)}{Q_m(x_i)} = (-1)^i \lambda \quad ,$$

$$i = 1,2,\ldots,(n+m+2) \quad .$$

In general, there are several values of λ which will satisfy these equations. Since the best approximation is unique, only one of the values of λ will give rise to the best approximation. Other values correspond to approximations which also have the error alternation property, but possess a pole in the interval $[a,b]$.

The iterative method we used to solve this system of equations does not always converge. For examples when P_2/Q_2 is used to approximate $\operatorname{erf}(x)$ in $[0,2]$, convergence does not occur before 1,000 iterations. More disturbing, there are no signs of convergence when P_1/Q_3 and P_0/Q_2 are used to approximate

$$f(x) = \begin{cases} x & , x \in [0,1] \\ 0.5x + 0.4 & , x \in (1,2] \end{cases} .$$

More frequently, the algorithm converges to a value of λ for which the corresponding approximation possesses a pole in the interval

$[a,b]$.

In some cases, with different starting values for λ_0 and the estimated extreme points, the algorithm converges to different approximations. This is illustrated by the following example:

$\sqrt{x}$ is approximated by P_2/Q_2 in the interval $[0,1]$.

Starting with the initial extreme points

$\{0., 0.1, 0.35, 0.65, 0.9, 1.0\}$

and $\lambda_0 = 0$, the rational function obtained after six iterations is

$$\frac{P_2(x)}{Q_2(x)} = \frac{0.041916 + 3.18742x + 4.19565x^2}{1 + 1.16117x - 3.16976x^2} .$$

The corresponding error function alternates at

$\{0, 0.05, 0.4, 0.75, 0.8, 1.0\}$

with amplitude $\lambda = 0.047917$. This approximating function has a pole at $x = 0.77395$.

If we start with initial extreme points $\{0, 0.2, 0.4, 0.6, 0.8, 1.0\}$ and $\lambda_0 = 0$, the resulting approximating function is

$$\frac{0.042205 + 3.15433x - 5.30056x^2}{1 + 0.77978x - 3.97652x^2} .$$

The error function has extreme points at $\{0, 0.05, 0.4, 0.6, 0.65, 1.0\}$ and the amplitude is 0.042205. This approximating function also possesses a pole in the interval $[0,1]$.

The solution without a pole is given by

$$\frac{0.0019293 + 8.88576x + 33.79501x^2}{1 + 27.15010x + 14.61511x^2} \quad ,$$

where its error function alternates on the set {0., 0.05, 0.1, 0.3, 0.7, 1.0} and the amplitude is 0.00192938. However, when we use these extreme points as starting points and choose $\lambda_0 = 0$, the algorithm does not generate the above approximation. It converges to

$$\frac{0.041083 + 3.086694x - 13.69958x^2}{1 - 1.83024x - 10.19448x^2} \quad ,$$

where the error function has extreme points at {0., 0.05, 0.2, 0.25, 0.45, 1.0} and the amplitude is 0.041083. Again this solution has a pole in [0,1]. Only if we set the initial value of λ_0 very close to the value 0.001929, will the algorithm converge to the solution which is pole free in [0,1].

In general, the algorithm will converge to the best approximation if the initial approximation is close to the best approximation. In this case, four or five iterations usually suffice to obtain the solution. However, a good starting approximation is not always available.

(5) Maehly's Method.

In Maehly's method, we take the zeros of the $(m+n+1)^{st}$ Chebyshev polynomial as the initial estimates of the zeros of the error function

$\varepsilon(x)$. Usually the algorithm converges to the best approximation in about 10 iterations provided that starting values are good enough.

In some cases, due to poor starting values, we encountered some difficulties in the computation. Although the first approximation is determined by solving the system of linear equations,

$$\frac{P_n(z_j)}{Q_m(z_j)} = f(z_j) \quad , \quad j = 1,2,\ldots,(n+m+1) \quad ,$$

where z_j are the initial estimates of the zeros, it may have an error function which has more than $(n+m+1)$ zeros in the interval $[a,b]$. That is, $\varepsilon(x)$ alternates more than $(n+m+2)$ times. This causes difficulty in choosing the extreme points for calculating the dz_j (the corrections of the zeros).

Also, since we consider only the discrete approximation problems, $\varepsilon(x)$ may have less than $n+m+1$ sign changes over the discrete data points $x_i \in X$, i.e. there may be two zeros in $[x_i,x_{i+1}]$. In this case, we do not have sufficient extreme points for calculating the values of dz_j . In these cases, the algorithm usually fails to converge. For example, approximating the data

$$f(x) = \begin{cases} e^x & , x \in [0,1] \\ e^{-x} - e^{-1} + e & , x \in (1,2] \end{cases} ,$$

the method fails to converge for all five approximating functions when the zeros of the Chebyshev polynomials are chosen to be the initial zeros

Since the first approximation generated already has an error function which alternates $(m+n+2)$ times, it is close to the best approximation. The algorithm then performs an adjustment on the zeros to reduce the maximum error. In some cases, the improvement is not significant. For example, when we approximate $f(x) = \sin(x)$ by P_2/Q_2 in the interval $[-3,3]$, the initial zeros chosen are $\{-2.5, -1.0, 0.5, 1.0, 2.5\}$. The maximum error in each iteration is given below:

Iteration	Maximal Error
1	0.39230
2	0.40799
3	0.32815
4	0.36084
5	0.31528
6	0.34001
7	0.31085
8	0.32739
9	0.30867
10	0.31959

We see that the maximum error is reduced in every alternate iteration. The convergence rate is slow. The best approximation is obtained after 35 iterations where the maximum error is 0.306077.

The choice of zeros of the Chebyshev polynomials as initial zeros is quite satisfactory when the data to be approximated is smooth, for example e^x, $\log(1+x)$, $\mathrm{erf}(x)$, e^{-x^2} and $\Gamma(x)$. Best approximations are obtained without much difficulty in these cases. If the data is not

smooth, the algorithm seldom converges. For example, for the function

$$f(x) = \begin{cases} x & , x \in [0,1] \\ 0.5x + 0.4 & , x \in (1,2] \end{cases} ,$$

of the five approximating functions used, only P_0/Q_2 converges to the best approximation. Convergence occurs after 20 iterations. For the function

$$f(x) = \begin{cases} 1 & , x \in [0,.5) \\ 0 & , x = .5 \\ -1 & , x \in (.5,1] \end{cases} ,$$

the five approximating functions obtained are $P_2/Q_2 = -1$, $P_4/Q_2 = -1$, $P_1/Q_1 = 1$, $P_1/Q_3 = 1$, $P_0/Q_2 = 0$. We see that only P_0/Q_2 gives the best approximation. The method therefore appears to be quite successful for approximating smooth data, but it is usually unsuccessful otherwise.

(6) A Linearization Method.

In this method, convergence to a solution is not guaranteed in general. It depends on the starting values used and the approximating function. The results in Table XIII are obtained with the starting values $p_0 = q_0 = 1$ and $p_i = q_j = 0$ $(i = 1,2,\ldots,n,\ j = 1,2,\ldots,m)$. From the results, we note that the convergence rate of this method is very fast when it does converge, and all solutions are obtained within

ten iterations. However, it may converge to a solution with a pole. The numerical difficulties we encountered for this case are approximating $\sqrt{x}$ in $[0,1]$ by P_1/Q_3 and $\begin{cases} e^x & , x \in [0,1] \\ e^{-x} - e^{-1} + e & , x \in (1,2] \end{cases}$ by P_4/Q_2. The results in Table XIII also indicate that if the degree of the approximating rational function is high, without good initial values, this method is not a satisfactory one. As an example, consider the approximating function P_4/Q_2. Of the ten data sets used, only three best approximations were obtained. Using different starting values, where the initial values of p_i and q_j are obtained by a random number generator, the number of best P_4/Q_2 approximations obtained increases to six. These solutions are also obtained within ten iterations.

We did not include in the constraints the restrictions $Q(x_i) \geq 0$. These restrictions may overcome the possibility of convergence to an approximation with a pole. Also, in searching for the minimum of the expression

$$\max_i \left| f(x_i) - \frac{P(x_i) + \lambda\delta P(x_i)}{Q(x_i) + \lambda\delta Q(x_i)} \right|$$

we only used a grid of values for λ, $-1.0(.02)1.0$. If the minimum value occurred when $\lambda = 0$ then the algorithm was terminated. However, in these cases it appears that a finer grid of values for λ will produce a decrease in the error of approximation. This is suggested by Watson (1970).

We conclude that this method has a very fast convergence rate when it converges, and it usually will converge if the starting values are sufficiently good. The main disadvantage of the method is the necessity for a linear search at each stage.

IV - 3. Conclusions.

In this thesis, we have presented several algorithms for the computation of best discrete rational approximations in the Chebyshev sense. The criterion we use to compare the methods is the number of iterations required for each method to converge to a prescribed accuracy. Alternative criteria that could be used are the amount of computation or the C.P.U. time required to obtain a best approximation. However, for the following five algorithms, the amount of work involved in each iteration is roughly comparable. For Loeb's algorithm and the linearization method, we solve a $(n+m+2) \times (2N)$ linear programming problem at each stage. For the linear inequality method, the size of the matrix is $(n+m+2) \times (3N)$. For the differential correction methods it is of size $(n+m+3) \times 2(N+n+m+2)$. The number of constraints is approximately the same, and it is quite reasonable to compare these five algorithms based on the number of iterations required to obtain the solution. These methods are easy to implement on a computer. For Loeb's algorithm, convergence often occurs in just a few iterations. However, there is no guarantee that convergence is to the best approximation. For the linearization method, the convergence rate is usually very fast. Unfortunately, the need for the linear search reduces its efficiency.

For the linear inequality method and the differential correction method, convergence to the best approximation is ensured, but the convergence rate is not as good as with other methods. The numerical results indicate that the modified differential correction method is the most satisfactory one for its fast convergence rate and sure convergence from any starting values.

For Maehly's method, we solve two $(n+m+1) \times (n+m+1)$ systems of linear equations at each stage. For the Remes algorithm, due to the iterative method applied at each iteration, we have to solve about five $(n+m+2) \times (n+m+2)$ systems of linear equations. Therefore, less operations are involved in Maehly's method at each stage. These two methods are not as easy to implement as the linear programming methods since they were originally proposed for solving the continuous approximation problem. The numerical results show that the convergence rates are fast for these methods, but there is no guarantee of convergence from arbitrary starting values.

TABLE I

DATA SETS FOR THE NUMERICAL EXAMPLES

	Function f(x)	Abscissae $\{x_1, x_2, \ldots, x_N\}$	Number of Points N
1.	e^x	-1.0(0.1)1.0	21
2.	sin(x)	-3.0(0.3)3.0	21
3.	$\sqrt{x}$	0.0(0.05)1.0	21
4.	1	0.0(0.05)0.45	21
	0	0.5	
	-1	0.55(0.05)1.0	
5.	x	0.0(0.1)1.0	21
	0.5x + 0.4	1.1(0.1)2.0	
6.	e^x	0.0(0.1)1.0	21
	$e^{-x} - e^{-1} + e$	1.1(0.1)2.0	
7.	log(1+x)	0.0(0.05)1.0	21
8.	erf(x)	0.0(0.1)2.0	21
9.	e^{-x^2}	0.0(0.1)2.0	21
10.	$\Gamma(x)$	2.0(0.05)3.0	21

TABLE II

BEST P_0/Q_2 APPROXIMATION

DATA	p_0	q_1	q_2	Max Error
f_1	0.96544	-1.01932	0.37907	0.03405
f_2	0.00000	0.00000	0 00000	0.99749
f_3	0.18117	-2.07440	1.29567	0.18117
f_4	0.00000	0.00000	0.00000	1.00000
f_5	0.22539	-1.01462	0.30528	0.22539
f_6	1.20697	-0.80360	0.28421	0.20697
f_7	0.09280	-2.01010	1.17068	0.09280
f_8	0.19844	-1.08579	0.35515	0.19844
f_9	0.93024	-1.06464	2.94511	0.69757
f_{10}	0.48334	-0.26423	0.00373	0.00641

Note: $q_0 = 1$.

TABLE III

BEST P_1/Q_1 APPROXIMATION

DATA	p_0	p_1	q_1	Max Error
f_1	1.01704	0.51755	-0.43976	0.20954×10^{-1}
f_2	0.00000	0.25551	0.00000	0.62542
f_3	0.04297	3.18115	2.36889	0.42972×10^{-1}
f_4	1.81818	3.63636	0.00000	0.81818
f_5	-0.05873	1.57291	0.60867	0.58739×10^{-1}
f_6	0.69127	5.89193	1.73212	0.30872
f_7	0.00085	0.97819	0.41422	0.85788×10^{-3}
f_8	-0.04408	1.86315	1.27130	0.44084×10^{-1}
f_9	1.07282	-0.57479	0.20401	0.72827×10^{-1}
f_{10}	0.48287	0.00765	-0.24929	0.64253×10^{-2}

Note: $q_0 = 1$.

TABLE IV

BEST P_2/Q_2 APPROXIMATION

DATA	p_0	p_1	p_2	q_1	q_2	Max Error
f_1	1.00007	0.50839	0.08571	-0.49132	0.07781	0.84776×10^{-4}
f_2	0.00000	2.28165	0.00000	0.00000	1.58959	0.30607
f_3	0.00192	8.88576	33.79501	27.15010	14.16511	0.19293×10^{-2}
f_4	0.73076	-1.46153	0.00000	-3.63636	3.63636	0.26923
f_5	0.05423	-3.55914	46.10706	22.69320	21.35198	0.54235×10^{-1}
f_6	1.08650	-1.48277	1.78873	-1.54130	1.07040	0.86503×10^{-1}
f_7	0.00000	0.99987	0.38616	0.88478	0.11484	0.15412×10^{-5}
f_8	0.00137	1.08589	0.30382	0.10162	0.55146	0.13753×10^{-2}
f_9	1.00267	-0.77483	0.14643	-0.68823	0.71304	0.26728×10^{-2}
f_{10}	1.24144	-0.63659	0.19163	-0.08320	-0.02468	0.35930×10^{-4}

Note: $q_0 = 1$.

TABLE V

BEST P_1/Q_3 APPROXIMATION

DATA	p_0	p_1	q_1	q_2	q_3	Max Error
f_1	0.99987	0.25358	-0.74660	0.24520	-0.03749	0.12237×10^{-3}
f_2	0.00000	2.28165	0.00000	1.58959	0.00000	0.30607
f_3	0.00763	6.54812	11.55292	-10.79577	4.84901	0.76302×10^{-2}
f_4	0.73076	-1.46153	-3.63636	3.63636	0.00000	0.26923
f_5	-0.02139	0.71350	-1.35342	1.48737	-0.40878	0.45572×10^{-1}
f_6	0.90464	-0.26791	-1.53208	0.99792	-0.22307	0.95354×10^{-1}
f_7	0.00000	0.99951	0.49498	-0.06632	0.01336	0.72177×10^{-5}
f_8	0.00092	1.10080	-0.10817	0.47577	-0.05913	0.92930×10^{-3}
f_9	1.00422	-0.46451	-0.33467	0.40535	0.41167	0.42278×10^{-2}
f_{10}	-0.04175	-0.23428	-1.77036	0.64754	-0.06996	0.54115×10^{-4}

Note: $q_0 = 1$.

TABLE VI

BEST P_4/Q_2 APPROXIMATION

DATA	p_0	p_1	p_2	p_3	p_4	q_1	q_2	Max Error
f_1	1.00000	0.67030	0.20261	0.03412	0.00286	-0.32969	0.93231	0.20465×10^{-6}
f_2	0.00000	1.01770	0.00000	-0.10444	0.00000	0.00000	0.08155	0.66482×10^{-2}
f_3	0.00006	12.93265	160.70134	108.25243	-14.58386	64.93582	201.39383	0.63642×10^{-4}
f_4	1.07046	-5.19153	9.15180	-6.10120	0.00000	-3.90329	3.90329	0.70465×10^{-1}
f_5	0.00620	1.04233	-2.21734	1.34816	-0.16477	-1.96388	0.97863	0.11176×10^{-1}
f_6	0.96908	-0.42079	-2.10332	2.09490	-0.35223	-1.92414	0.99396	0.30919×10^{-1}
f_7	0.00000	0.99999	0.70773	0.03988	-0.00208	1.20772	0.31056	$0.55984\ 10^{-8}$
f_8	0.00004	1.13071	-0.23743	0.09543	-0.00051	-0.19297	0.36561	0.44515×10^{-4}
f_9	1.00000	-0.34335	-0.62654	0.43082	-0.07622	-0.34392	0.38934	0.47168×10^{-4}
f_{10}	2.36335	-2.16812	1.32883	-0.35734	0.05244	0.34480	-0.09172	0.17423×10^{-7}

Note $q_0 = 1$.

TABLE VII

NUMBER OF ITERATIONS FOR LOEB'S ALGORITHM

DATA	Number of Iterations				
	P_0/Q_2	P_1/Q_1	P_2/Q_2	P_1/Q_3	P_4/Q_2
f_1	8	7	6	6	5
f_2	2	2	*	*	6
f_3	23	11	9	8	8
f_4	3	**	**	14	11
f_5	31	*	26***	13	12
f_6	10	*	24	**	15
f_7	22	5	5	4	5
f_8	24	11	8	6	6
f_9	15	15	8	10	6
f_{10}	5	7	6	6	5

* No convergence

** Convergence to a non-best approximation

*** Convergence to a solution with a pole.

TABLE VIII

NUMBER OF ITERATIONS FOR THE LINEAR INEQUALITY METHOD

DATA	Number of Iterations				
	P_0/Q_2	P_1/Q_1	P_2/Q_2	P_1/Q_3	P_4/Q_2
f_1	25	26	35	32	42
f_2	20	22	22	22	30
f_3	24	25	30	28	34
f_4	21	22	22	22	24
f_5	23	25	26	26	25
f_6	24	24	25	24	27
f_7	23	30	38	37	46
f_8	22	25	30	30	34
f_9	26	25	29	28	35
f_{10}	29	30	36	33	41

TABLE IX

NUMBER OF ITERATIONS FOR THE DIFFERENTIAL CORRECTION METHOD

DATA	Number of Iterations				
	P_0/Q_2	P_1/Q_1	P_2/Q_2	P_1/Q_3	P_4/Q_2
f_1	41	20	28	42	55
f_2	11	2	24	25	28
f_3	19	32	*	44	*
f_4	11	6	*	*	*
f_5	26	11	32	16	*
f_6	20	20	45	46	*
f_7	25	14	28	18	24
f_8	21	21	37	26	37
f_9	37	13	46	46	33
f_{10}	17	17	16	18	17

*Number of iterations exceeds 100.

TABLL X

NUMBER OF ITERATIONS FOR THE MODIFIED DIFFERENTIAL CORRECTION MFTHOD

DATA	Number of Iterations				
	P_0/Q_2	P_1/Q_1	P_2/Q_2	P_1/Q_3	P_4/Q_2
f_1	7	6	9	9	11
f_2	8	2	7	8	11
f_3	7	8	12	11	15
f_4	2	5	10	10	15
f_5	6	6	8	9	16
f_6	6	7	9	10	13
f_7	9	6	6	6	9
f_8	8	7	9	8	10
f_9	7	7	9	9	11
f_{10}	7	8	12	10	13

TABLE XI

NUMBER OF ITERATIONS FOR THE REMES ALGORITHM

DATA	Number of Iterations				
	P_0/Q_2	P_1/Q_1	P_2/Q_2	P_1/Q_3	P_4/Q_2
f_1	4	4	4	4	3
f_2	**	3	2	4*	2
f_3	4	6	6*	5*	6*
f_4	2	3	5	5	20
f_5	5	**	4*	6	7
f_6	4	4	5	4	6
f_7	5	3	5	3	4
f_8	4	8*	**	4	3
f_9	**	3	4	**	4
f_{10}	3	3	3	5*	3

*Convergence to a solution with a pole

**No convergence.

TABLE XII

NUMBER OF ITERATIONS FOR THE MAEHLY'S METHOD

DATA	Number of Iterations				
	P_0/Q_2	P_1/Q_1	P_2/Q_2	P_1/Q_3	P_4/Q_2
f_1	8	6	7	7	6
f_2	*	6	36	*	8
f_3	10	22	25	23	20
f_4	2	*	*	*	*
f_5	20	*	*	*	*
f_6	*	*	*	*	*
f_7	15	6	6	6	7
f_8	19	12	9	9	8
f_9	20	9	10	9	20
f_{10}	6	6	5	6	5

*No convergence.

TABLE XIII

NUMBERS OF ITERATION FOR THE LINEARIZATION METHOD

DATA	Number of Iterations				
	P_0/Q_2	P_1/Q_1	P_2/Q_2	P_1/Q_3	P_4/Q_2
f_1	6	5	7	8	8
f_2	2	2	13	9	*
f_3	7	7	12**	8	*
f_4	6	2	*	10	*
f_5	6	6	13	8	*
f_6	6	6	*	7	13**
f_7	*	6	10	6	11
f_8	7	7	11	7	9
f_9	7	7	10**	*	*
f_{10}	8	10	*	*	*

*No convergence

**Convergence to a solution with a pole.

REFERENCES

ACHIESER, N.I. (1930). On extremal properties of certain rational functions. Doklady Akademii Nauk SSSR, pp. 495-499.

ACHIESER, N.I. (1947). Lectures on the theory of approximation. Gostekhizdat, Moscow.

BARRODALE, I. and MASON, J.C. (1970). Two simple algorithms for discrete rational approximation. Math. of Computation, Vol. 24, pp. 877-892.

BARRODALE, I., POWELL, M.J.D. and ROBERTS, F.D.K. (1971). The differential correction algorithm for rational L_∞ approximation. Math. Report No. 54, University of Victoria.

BARRODALE, I. and YOUNG, A. (1966). Algorithm for best L_1 and L_∞ linear approximations on a discrete set. Numer. Math., Vol. 8, pp. 295-306.

CHEBYSHEV, P.L. (1859). Sur les questions de minima qui se rattachent a la representation par la methode des moindres carres, Oeuvres I, pp. 473-498.

CHENEY, E.W. and LOEB, H.L. (1962). On rational Chebyshev approximation. Numer. Math., Vol. 4, pp. 124-127.

CHENEY, E.W. (1966). Introduction to approximation theory. McGraw-Hill, New York.

DANTZIG, G.B. (1953). Computational algorithm of the revised simplex method. RAND Report RM-1266, The RAND Corporation, Santa Monica, Calif.

FRASER, W., and HART, J.F. (1962). On the computation of rational approximations to continuous functions. Comm. ACM, Vol. 5, pp. 401-403.

GASS, I.S. (1969). Linear programming methods and applications. McGraw-Hill, New York.

HANDSCOMB, D.C. (1966). Methods of numerical approximation. Pergamon Press, Oxford.

LOEB, H.L. (1959). A note on rational function approximation. Con. Astro. App. Math. Series No. 27.

LOEB, P.I. (1960). Algorithms for Chebycheff approximation using the ratio of linear forms. J. SIAM, Vol. 8, pp. 458-465.

MAEHLY, H.J. (1963). Methods for fitting rational approximations, Part II and III. J. ACM., Vol. 10, pp. 219-231.

ORCHARD-HAYS, W. (1954). Background, development and extensions of the revised simplex method. RAND Report RM-1433, The RAND Corporation, Santa Monica, Calif.

RABINOWITZ, P. (1968). Application of linear programming to numerical analysis. SIAM Review, Vol. 10, pp. 121-159.

RABINOWITZ, P. (1970). Mathematical programming and approximation. Approximation Theory. Proceedings of a symposium held at Lancaster, July 1969. Edited by A. TALBOT, Academic Press, London.

RALSTON, A. (1965). A first course in numerical analysis. McGraw-Hill, New York.

RIVLIN, T.J. (1969). An introduction to the approximation of functions. Blaisdell, Waltham, Mass.

STIEFEL, E.L. (1960). Note on Jordan elimination, linear programming and Tchebysheff approximation. Numer. Math., Vol. 2, pp. 1-17.

WALSH, J.L. (1931). The existence of rational function of best approximation. AMS Transactions, Vol. 33, pp. 668-689.

WATSON, G.A. (1970). On an algorithm for nonlinear minimax approximation. Comm. ACM., Vol. 13, pp. 160-162.

```
      SUBROUTINE RESIM(M2,N1,P,ESP,BASIS,U,PC,IX)
C
C     RESIM SOLVES THE LINEAR PROGRAMMING PROBLEM
C       MINIMIZE C*X
C     SUBJECT TO  A*X=B
C     WHERE  A  IS  M*N  DIMENSIONAL MATRIX
C            B  IS  M    DIMENSIONAL VECTOR
C            C  IS  N    DIMENSIONAL VECTOR
C     PARAMETERS:
C            M2=M+2
C            N1=N+1
C            P      - (M+2)*(N+1) MATRIX
C            ESP    - A POSITIVE TOLERANCE LESS THAN WHICH QUANTITIES ARE
C                     CONSIDER TO BE NONPOSITIVE
C            BASIS  - (M+2) INTEGER VECTOR
C            U      - (M+2)*(M+2) MATRIX
C            PC     - (M+2) VECTOR
C            IX     - KEY WORD:
C                          1 - SOLUTION OBTIANED
C                          2 -UNBOUNDED SOLUTION
C                          3 -NO FEASIBLE SOLUTION
C     INITIALLY STORE  A(M,N)  IN  P(M,N)
C                      B(M)    IN  P(*,N+1)
C                      C(N)    IN  P(M+1,*)
C     FINAL SOLUTION IS IN  A(*,N+1)  INDICATED BY
C     THE INTEGER VECTOR BASIS(M)
C
      IMPLICIT REAL *8 (A-H,O-Z)
      DIMENSION  P(M2,N1),U(M2,M2),PC(M2)
      INTEGER BASIS(M2)
      LOGICAL SW
      SW=.FALSE.
      M=M2-2
      M1=M+1
      N=N1-1
      IX=1
      DO 1 I=1,M2
    1 U(I,J)=0.D0
      U(I,I)=1.D0
      BASIS(I)=N+I
      P(M1,N1)=0.D0
      DO 3 I=1,N1
    3 P(M2,I)=0.D0
```

```
      DO 3 J=1,M
    3 P(M2,I)=P(M2,I)-P(I,I)
      IPHAS=1
      MP=M2
  100 DMIN=1.D50
      IF(IPHAS.EQ.1.AND.P(M2,N1).GE.-ESP)GO TO 200
      DO 6 J=1,N
      D=0.D0
      DO 5 I=1,M2
    5 D=D+U(MP,I)*P(I,J)
      IF(D.GL.DMIN)GO TO 6
      DMIN=D
      K=J
    6 CONTINUE
      IF(DMIN.GT.-ESP)GO TO 13
    4 DO 7 J=1,M2
      PC(J)=0.D0
      DO 7 I=1,M2
    7 PC(J)=PC(J)+U(J,I)*P(I,K)
      IF(SW)GO TO 12
      DO 8 I=1,M
      IF(PC(I).LE.ESP)GO TO 8
      L=I
      GO TO 9
    8 CONTINUE
      IX=2
      WRITE(6,62)
   62 FORMAT('0***UNBOUNDED SOLUTION***')
      RETURN
    9 PMIN=P(L,N1)/PC(L)
      DO 10 I=1,M
      IF(PC(I).LE.ESP)GO TO 10
      IF(P(I,N1)/PC(I).GT.PMIN) GO TO 10
      PMIN=P(I,N1)/PC(I)
      L=I
   10 CONTINUE
   12 PMIN=P(L,N1)/PC(L)
      SW=.FALSE.
      DO 16 I=1,M2
      IF(I.EQ.L)GO TO 16
      DO 17 J=1,M
   17 U(I,J)=U(I,J)-U(L,J)/PC(L)*PC(I)
   16 CONTINUE
      DO 20 I=1,M
```

```
 20 U(L,I)=U(L,I)/PC(I)
    DO 15 I=1,M2
 15 P(I,N1)=P(I,N1)-PMIN*PC(I)
    P(L,N1)=PMIN
    BASIS(L)=K
    GO TO 100
 13 GO TO (18,19),IPHAS
 18 IX=3
    WRITE(6,61)
 61 FORMAT('0***NO FLASIBLE SOLUTION***')
 19 RETURN
200 CONTINUE
    DO 21 IJK=1,M
    IF(BASIS(IJK).LE.N)GO TO 21
    L=IJK
    DO 22 I=1,N
    PC(L)=0.DO
    K=I
    DO 23 KK=1,M
    IF(K.EQ.BASIS(KK))GO TO 22
 23 CONTINUE
    DO 24 J=1,M2
 24 PC(L)=PC(L)+U(L,J)*P(J,K)
    SW=.TRUE.
    IF(DABS(PC(L)).GT.ESP)GO TO 4
 22 CONTINUE
 21 CONTINUE
    IPHAS=2
    MP=M1
    GO TO 100
    END
```

VITA

Surname: LEE Given Names: CHEE-MOU

Place of Birth: MALAYSIA Date of Birth: NOVEMBER 30, 1946

Educational Institutions Attended, with Dates of Entering and Leaving:

NANYANG UNIVERSITY, SINGAPORE	1965	to	1969
		to	
		to	
		to	

Degrees, Diplomas, Etc., Awarded, with Dates and Names of Institutions:

B.Sc.	1969	Nanyang University, Singapore

Honors and Awards:

University of Victoria Graduate Scholarship, 1969/70 and 1970/71.

Publications:

THE UNIVERSITY OF VICTORIA LIBRARY

MANUSCRIPT THESIS

AUTHORITY TO DISTRIBUTE

AUTHOR: This dissertation may be lent or microfilm copies made available:

(a) Without restriction [redacted]

(b) With the restriction that, for a period of five years (until _____) the written approval of the following is required:

(1) The Chairman, School of Graduate Studies _____

(2) The Author _____

(3) both the Chairman, School of Graduate Studies, and the Author _____

BORROWERS: The borrower undertakes, by signing below, to give proper credit for any use made of the dissertation, and to obtain the consent of the author if it is proposed to make extensive quotations, or to reproduce the dissertation in whole or in part.

Signature of Borrower	Address	Date