SIAM J. Numer Anal.
Vol 12, No. 1, March 1975

A MULTIPLE EXCHANGE ALGORITHM FOR MULTIVARIATE CHEBYSHEV APPROXIMATION*

G. A. WATSON†

Abstract. A Remes-type algorithm based on linear programming is presented for computing linear best Chebyshev approximations to multivariate functions. Numerical results are given for some 2-variable examples.

1. Introduction. Until fairly recently, the development of algorithms for computing best linear Chebyshev approximations to functions of one real variable x on an interval $[a, b]$ was influenced to a large extent by the requirement that the set of approximating functions should form a Chebyshev set in $[a, b]$. By this we mean that if there are n functions $\{\phi_i(x)\}$, then no linear combination of these should vanish at more than $n - 1$ points in $[a, b]$. The importance of this "classical" restriction is that certain degenerate cases are excluded; in particular, we know that the maximum error in the best approximation occurs at $n + 1$ points in $[a, b]$ with alternating sign and that this best approximation is unique (see, for example, [3]). In fact, if the interval $[a, b]$ is replaced by at least $n + 1$ distinct points from $[a, b]$, then the resulting best approximation is also characterized in this way.

Two ascent algorithms, due to Remes (see [3]), are often used to solve the Chebyshev approximation problem on an interval. One of these, the second algorithm of Remes or multiple exchange algorithm, makes use of the sign alternation property and thus requires the classical assumptions. The other, the first algorithm of Remes, although also developed within the classical framework, may be formulated independently of the classical assumptions [3, p. 96], and thus is applicable in a more general situation. Both algorithms involve the solution of a sequence of discrete problems: at the kth step, a finite discrete subset X^k of the interval $[a, b]$ is defined, and the best Chebyshev approximation obtained on this point set. In the second algorithm of Remes, successive subsets contain precisely $n + 1$ points: an initial set of $n + 1$ distinct points is replaced by $n + 1$ local extrema of the current error function in a process which has been shown by Veidinger [12] to have quadratic convergence. For the first algorithm of Remes, the initial point set contains at least $n + 1$ points, and these are supplemented at each step by the global maximum of the error in the current approximation.

If the functions $\{\phi_i(x)\}$ form a Chebyshev set, then it follows that the matrix A of the discrete problem (i.e., the matrix with (i, j) element $\phi_i(x_j)$, where $x_j \in X^k$) has all its $n \times n$ minors nonsingular. In this case, A is said to satisfy the Haar condition, and again algorithms developed within the classical framework are relevant, for example, the exchange algorithm of Stiefel [11]. It is known, however, that a linear programming formulation of this algorithm can be applied without requiring the Haar condition to be satisfied. In fact, the standard simplex method of linear programming may be used to solve the discrete problem provided only

* Received by the editors October 26, 1973, and in revised form February 26, 1974.

† Department of Mathematics, University of Dundee, Dundee, Scotland.

that the rank of A is n (see, for example, [2], [9]). An exchange algorithm which may be applied in the absence of the Haar condition is also discussed in [4].

The important role which the classical assumptions have continued to play in the actual computation of Chebyshev approximations is undoubtedly a factor contributing to the lack of algorithms for multivariate problems: it was first shown by Mairhuber [8] that Chebyshev sets do not exist in spaces of dimension greater than one. The only implementation known to the author is that of a method due to Fletcher, Grant and Hebden [5], who present a promising approach based on the Pólya algorithm (see [3]). The possibility of applying Remes-type ascent algorithms, however, seems not to have been investigated, and it is the purpose of this paper to outline and numerically illustrate the use of such a method based on the linear programming approach.

It will be necessary to make use of certain linear programming results, details of which may be found in [7]. In particular, it is a feature of the application of the simplex algorithm to this problem (see §2) that each basic feasible solution identifies $n + 1$ points of maximum deviation of the error function. Therefore methods of this type have a potential difficulty when used to solve the Chebyshev problem on a continuum X when the optimal error curve has less than $n + 1$ extrema in X. We define such problems to be *singular*. In the one-variable case, singular problems are relatively uncommon, though they arise frequently in more than one dimension. This is perhaps due to the intrinsically more complex nature of multivariate problems and the absence of useful characterization theorems even for simple polynomial functions.

The application of the linear programming approach to singular problems in one variable is discussed in [10] and is illustrated by an example which shows that the reduction in the number of extrema is obtained by certain of these points coalescing. Successive optimal basis matrices become more and more singular, and thus the condition of the linear programming problem deteriorates. This would appear to be a serious difficulty with the approach, but we justify our investigation on the grounds that:

(a) not all multivariate problems are singular, and the method should work well in nonsingular cases;

(b) even if a problem is singular, an approximation can still be obtained;

(c) numerical evidence is always valuable, particularly in view of the aforementioned lack of alternative algorithms.

It is, of course, possible to obtain an *approximation* to the solution of any continuous problem by straightforward discretization. As already indicated, if a solution is obtained by the simplex algorithm of linear programming, the optimal basis matrix will define $n + 1$ points of maximum deviation of the error, and for the approximation to be a good one, these points must all lie close to extrema of the continuous problem. Thus if the continuous problem is singular, these $n + 1$ points can only satisfy this criterion if at least 2 of them lie close together. If equispaced points are used, it follows that a large number is required. Further, since adjacent points will be represented in the optimal basis matrix, again this matrix will be nearly singular. We are concerned here, however, with the solution to the continuous problem rather than to the discrete analogue of it.

The basic algorithm (which for the one-variable restricted case is given in

[13]) is outlined in the next section, and the implementation of a modified version discussed. In §3, some numerical results with 2-variable problems, many of them singular, are presented.

2. The algorithm. Let X be the real m-dimensional continuum defined by $[a_1, b_1] \times [a_2, b_2] \times \cdots \times [a_m, b_m]$. Let $C[X]$ denote the Banach space of all real-valued continuous functions defined on X with the norm $\|f\| = \max \{|f(\mathbf{x})| : \mathbf{x} \in X\}$ and let $\phi_i(\mathbf{x}) \in C[X]$, $i = 1, 2, \cdots, n$, be given linearly independent functions. Let $f(\mathbf{x}) \in C[X]$ and define, for all $\mathbf{x} \in X$,

$$r(\boldsymbol{\alpha}, \mathbf{x}) = f(\mathbf{x}) - p(\boldsymbol{\alpha}, \mathbf{x}), \tag{2.1}$$

where

$$p(\boldsymbol{\alpha}, \mathbf{x}) = \sum_{i=1}^{n} \alpha_i \phi_i(\mathbf{x}), \tag{2.2}$$

with $\boldsymbol{\alpha} = (\alpha_1, \alpha_2, \cdots, \alpha_n)^T$. Then the best Chebyshev approximation to $f(\mathbf{x})$ in X by functions of the form (2.2) is obtained by finding a vector $\boldsymbol{\alpha}^*$ such that

$$\|r(\boldsymbol{\alpha}^*, \mathbf{x})\| = \inf \|r(\boldsymbol{\alpha}, \mathbf{x})\|.$$

It is well known that a solution to this problem exists, but as already mentioned, uniqueness cannot be guaranteed for $m \geqq 2$.

Assuming that linear programming is used to solve the discrete subproblems, the basic Remes algorithm may be defined as follows.

Step 1. Let X^1 be a subset of X consisting of r distinct points $\mathbf{x}_i$, $i = 1, 2, \cdots, r$, where $r \geqq n + 1$, and let the matrix A be defined by

$$A_{ij} = \phi_j(\mathbf{x}_i), \qquad i = 1, 2, \cdots, r, \quad j = 1, 2, \cdots, n, \tag{2.3}$$

such that the rank of A is n. Let $\mathbf{f}$ be an r-vector with ith component $f(\mathbf{x}_i)$, and form

$$\mathbf{c}_1^T = [\mathbf{f}^T, -\mathbf{f}^T], \qquad C_1 = \begin{bmatrix} A^T & -A^T \\ \mathbf{e}^T & \mathbf{e}^T \end{bmatrix},$$

where $\mathbf{e}$ is a vector, each component of which is 1. Set $k = 1$.

Step 2. Define the linear programming problem

$$\text{minimize } h \quad \text{subject to} \quad C_k^T \begin{bmatrix} \boldsymbol{\alpha} \\ h \end{bmatrix} \geqq \mathbf{c}_k. \tag{2.4}$$

This may conveniently be solved by going to the dual formulation

$$\text{maximize } \mathbf{c}_k^T \mathbf{w} \quad \text{subject to} \quad C_k \mathbf{w} = \mathbf{e}_{n+1}, \quad \mathbf{w} \geqq 0, \tag{2.5}$$

where each component of $\mathbf{e}_{n+1}$ is zero except the $(n + 1)$st, which is 1. Let the solution (to the primal) be $\boldsymbol{\alpha} = \boldsymbol{\alpha}^k$, $h = e^k$.

Step 3. If $\|r(\boldsymbol{\alpha}^k, \mathbf{x})\| = e^k$, go to Step 4. Otherwise, let $\mathbf{y}^k$ be a value of $\mathbf{x}$ at which $\|r(\boldsymbol{\alpha}^k, \mathbf{x})\|$ is attained, and define

$$\mathbf{d}_k = [\phi_1(\mathbf{y}^k), \phi_2(\mathbf{y}^k), \cdots, \phi_n(\mathbf{y}^k)]^T,$$

$$C_{k+1} = \left[\begin{array}{c|cc} C_k & \mathbf{d}_k & -\mathbf{d}_k \\ & 1 & 1 \end{array}\right],$$

and $\mathbf{c}_{k+1}^T = [\mathbf{c}_k^T, f(\mathbf{y}^k), -f(\mathbf{y}^k)]$. Increase k by 1 and go to Step 2.

Step 4. The problem has been solved. A solution is given by $\boldsymbol{\alpha}^k$, with e^k equal to the limiting value of $\|r(\boldsymbol{\alpha}, \mathbf{x})\|$.

If the matrix A has rank n, the matrices C_k have rank $n + 1$, and so the linear programming problem (2.5) has an optimal (basic feasible) solution for all k (see [13]). Convergence of the algorithm may be proved by making minor modifications to the proof for the one-variable case given by Cheney [3].

The algorithm was not, in fact, implemented precisely as described, but the following modification introduced. The matrix C_{k+1} was formed from C_k by adding not only those columns based on a point $\mathbf{y}^k$, but also a number of additional pairs of columns corresponding to as many local maxima of $|r(\boldsymbol{\alpha}^k, x)|$ (greater than e^k) as could be obtained in X. Of course, provided that the global maximum is included at each step, the convergence proof is still valid, and the only effect of the modification is to accelerate the convergence. This is important, as the calculation of $\mathbf{y}^k$ (which involves the calculation of local maxima anyway) can be time-consuming.

For each successive value of k, a new optimal solution to the dual problem is obtained starting from the previous one. A column of C_{k+1} corresponding to the newly found global maximum point is, in fact, immediately introduced into the optimal basis matrix for the previous iteration, so that no searching need be done at this step (see [13]). Subsequently, the simplex algorithm is allowed to progress to optimality in the usual way. Thus the total number of simplex steps at each iteration (after the first) is relatively low. Now it is a standard programming result that if a variable is in the dual basis, the corresponding primal constraint holds with equality. Thus each basis identifies $n + 1$ "reference points" where the current maximum value of the objective function is attained, and so each step of the simplex method corresponds to an exchange step analogous to a single Remes step. Consequently, each iteration of the modified algorithm can be thought of as a *multiple* exchange step.

The major problem in implementing the algorithm lies in the computation of the local maxima of $|r(\boldsymbol{\alpha}^k, \mathbf{x})|$. However, the following simple scheme is suggested, and for the case $m = 2$, has proved to be remarkably effective. Starting from a reference point $\boldsymbol{\xi}$, say, the local maximum in the neighborhood of $\boldsymbol{\xi}$ is obtained. This may be efficiently done using Newton's method to find a zero of $\nabla r(\boldsymbol{\alpha}^k, \mathbf{x})$. However, if the Newton iteration gives an initial step which gives a decrease in $|r(\boldsymbol{\alpha}^k, \mathbf{x})|$, or moves outside X, this iteration is abandoned. If the iteration converges to a point $\mathbf{y}$ such that $|r(\boldsymbol{\alpha}^k, \mathbf{y})| > e^k$, then the iteration is successful. This procedure is applied to each of the $n + 1$ reference points in turn, and the limit points of successful iterations (removing duplicates) is recorded. Finally, a grid search is conducted on the edges of X in order to span local maxima of $|r(\boldsymbol{\alpha}^k, \mathbf{x})|$ occurring there, and these maxima are then estimated by quadratic fits.

The matrix C_{k+1} is formed from C_k by the addition of pairs of columns based on all the points obtained by both the above procedures, $\mathbf{c}_{k+1}$ is analogously defined and the next iteration is commenced. It is tempting to restrict the size of the matrices C_k by dropping columns, for example, those corresponding to points at which the error is small. However, these are comparatively few in number, as after the first one or two steps the points tend to cluster round the extrema. If the initial set of points X^1 is chosen to be equispaced in X, their continuing presence is also of value if the current reference points are confined to an unrepresentative portion (for example, the boundary) of X, for then the deviations are still examined at points throughout the remainder. Difficulties in guaranteeing convergence also arise if points are to be discarded.

The size of the matrices C_k depends to a certain extent on the value of r, the number of points in X^1, and so this value should not be too large. However, the larger r is, the nearer are the $n + 1$ extrema of the discrete solution likely to be to the current local maxima on the first iteration, and there is less likelihood of the Newton iterations failing to converge. Typically, we have chosen the set X^1 to be equispaced in X, and $2n < r < 4n$.

The implementation of the algorithm as outlined imposes the additional requirement on $f(\mathbf{x})$ and the functions $\{\phi_i(\mathbf{x})\}$ that they be twice differentiable with respect to $\mathbf{x}$, in order that Newton's method may be applied. Some alternative scheme would be required if this condition is not satisfied. Finally, we note that the algorithm is in fact valid if X is a compact subset of a real m-dimensional continuum. However, the task of dealing with any more complicated region would introduce severe problems for this (and probably any other) method.

3. Numerical results. All the numerical results were obtained on the Elliott 4130 of the University of Dundee, which gives about 11 decimal place accuracy using floating-point arithmetic. Basically, the same ALGOL program was used for all the calculations. The linear programming problems were solved by the inverse matrix formulation, primarily because the code for this, imbedded in a similar Remes-type method, was already available. Because of the potential ill-conditioned nature of many of the problems, a stable linear programming code based on the LU factorization (for example, [1], [6]) is advisable, although for the examples attempted, no difficulties were experienced. The stable code would, however, be an essential part of any definitive implementation of the method.

Convergence was accepted when the calculated maximum error on X differed from the current e^k by less than some prescribed tolerance. For the examples quoted, this was taken to be 0.5×10^{-6}, although 0.5×10^{-8} was also used in some cases and attained without difficulty.

The first example is that solved in [5], where e^{-x^2-y} is approximated by the quadratic

$$\alpha_1 + \alpha_2 x + \alpha_3 y + \alpha_4(2x^2 - 1) + \alpha_5 xy + \alpha_6(2y^2 - 1)$$

in the square $(0 \leqq x \leqq 1, 0 \leqq y \leqq 1)$. In Table 1, we show the progress of the algorithm, starting with 16 equispaced points. The global maximum (modulus) is denoted by g, l denotes the number of local maxima, and s is the number of simplex steps. Convergence was obtained in 16.2 seconds. The coefficients were still changing in the third decimal place, but the final values were

$$\alpha_1 = 0.985738, \quad \alpha_2 = -0.347940, \quad \alpha_3 = -0.902683,$$
$$\alpha_4 = -0.144633, \quad \alpha_5 = 0.424562, \quad \alpha_6 = 0.112918.$$

The final set of reference points (with deviation signs) was

$$(0.2720, 0)-, \quad (1, 0.6221)-, \quad (0.6781, 1)+,$$
$$(0,0.2176)+, \quad (0, 1)\cdot -, \quad (0.8362, 0)+, \quad (0.6756, 1)+.$$

The third and seventh points are coalescing.

TABLE 1
Approximation to e^{-x^2-y}

k	s	e^k	g	l
1	13	0.024620	0.034219	5
2	3	0.026886	0.028296	5
3	3	0.027204	0.027530	5
4	2	0.027250	0.027334	5
5	5	0.027273	0.027292	5
6	3	0.027274	0.027278	3
7	4	0.027274	0.027275	1
8	2	0.027275	0.027275	

Polynomial approximations were obtained to a number of simple functions of 2 variables in the region —1 $\leqq x, y \leqq 1$. For the unsymmetric functions, the approximating polynomials were of the form

$$\sum_{i=0}^{t} \sum_{j=0}^{t} \alpha_{ij} x^i y^j$$

for given t, and for the symmetric functions, symmetry was used to reduce the number of unknown coefficients. A summary of results for some of these problems (which are mostly singular—denoted by S) is given in Table 2. The value of N gives the number of iterations required to obtain convergence using a tolerance of 0.5×10^{-6}, and e^* is the limiting value of e^k. As is to be expected, the nonsingular

TABLE 2
Approximations to some simple functions

$f(\mathbf{x})$	n	r	N	e^*	
$(x + 2y + 4)^{1/2}$	9	16	9	0.011401	S
	16	25	15	0.002747	S
$\exp(x^2 + xy)$	9	16	12	0.735469	S
$\sin(x^2 + y)$	9	16	7	0.071228	NS
$1/(x + 2y + 4)$	9	16	10	0.058359	S
$\exp(xy)$	6	16	4	0.045017	NS
	10	25	4	0.005528	NS
	15	36	7	0.000547	NS
$1/(x + y + 3)$	6	16	8	0.026137	S
	10	25	7	0.006889	S
	15	36	10	0.001826	S
$(x + y + 3)^{1/2}$	6	16	7	0.003896	S
	10	25	5	0.000622	S
	15	36	7	0.000114	S

problems tend to converge faster than the singular ones. In fact, although there is no proof, it is possible that the rate of convergence for nonsingular problems could be quadratic, since the method is then analogous to the second algorithm of Remes. A check on the validity of all these approximations was made by evaluating the errors on a 40 × 40 grid of points in the region X. On no occasion was an error greater than e^* (to the specified tolerance) obtained.

4. Concluding remarks. Obviously, it is not our intention to suggest that the solution of a (possibly) well-conditioned problem by a sequence of increasingly ill-conditioned ones is a particularly desirable strategy. However, the algorithm as we have described it, is easy to apply and easy to automate, and our numerical experiments, though by no means extensive, give results which indicate its reliability even when the approximation problems turn out to be singular. For such problems, the salvation of the method in a practical sense seems to be that reasonably accurate approximations may be obtained while the coalescing points are still significantly separated. In other words, the error function can be well-determined while the parameters $\boldsymbol{\alpha}$ are not. Thus although convergence tends to be retarded in such problems because of the nature of the exchange, the approach of singularity is slow enough to permit good approximations to be obtained without ever requiring the solution of a problem of a too severely ill-conditioned nature.

REFERENCES

[1] R. H. Bartels and G. H. Golub, *The simplex method of linear programming using LU decomposition*, Comm. ACM, 12 (1969), pp. 266–268.

[2] L. Bittner, *Das Austauschverfahren der linearen Tschebyscheff-Approximation bei nicht erfüllter Haarscher Bedingung*, Z. Angew, Math. Mech., 41 (1961), pp. 238–256.

[3] E. W. Cheney, *Introduction to Approximation Theory*, McGraw-Hill, New York, 1966.

[4] J. Descloux, *Dégénérescence dans les approximations de Tschebyscheff linéaires et discrètes*, Numer. Math., 3 (1961), pp. 180–187.

[5] R. Fletcher, J. A. Grant and M. D. Hebden, *Linear minimax approximation as the limit of best L_p-approximation*, this Journal, 11 (1974), pp. 123–136.

[6] P. E. Gill and W. Murray, *A numerically stable form of the simplex algorithm*, J. Linear Algebra Appl. 7 (1973), pp. 99–138.

[7] G. Hadley, *Linear Programming*, Addison-Wesley, Reading, Mass., 1962.

[8] J. Mairhuber, *On Haar's theorem concerning Chebyshev approximation problems having unique solutions*, Proc. Amer. Math. Soc., 7 (1956), pp. 609–615.

[9] M. R. Osborne and G. A. Watson, *On the best linear Chebyshev approximation*, Comput. J., 10 (1967), pp. 172–177.

[10] ———, *A note on singular minimax approximation problems*, J. Math. Anal. Appl., 25 (1969), pp. 692–700.

[11] E. L. Stiefel, *Uber diskrete und lineare Tschebyscheff-Approximationen*, Numer. Math., 1 (1959), pp. 1–28.

[12] L. Veidinger, *On the numerical determination of the best approximations in the Chebyshev sense*, Ibid., 2 (1960), pp. 99–105.

[13] G. A. Watson, *The calculation of best restricted approximations*, this Journal, 11 (1974), pp. 693–699.