The Complexity of Linear Approximation Algorithms

Martin H. Schultz

Research Report #21

This research was supported in part by the Office of Naval Research, N0014-67-A-0097-0016. The article was presented at the American Mathematical Society's "Symposium on the Complexity of Real Computational Processes," held in April 1973.

May 1973

§1. Introduction

The computation of simple approximations to general functions or data is a very common activity at most computing centers. In this paper, we discuss the complexity of such computations. We discuss questions which are direct analogues of those currently being discussed in "concrete" complexity and computational combinatorics. Specifically we will concentrate on four themes: (1) the general problem is computationally difficult; (2) adaptive or artificial intelligence algorithms are computationally difficult; (3) subproblems are computationally easy; and (4) computationally easy "approximate" algorithms exist.

In this regard, the emphasis of this paper is somewhat different from that of the recent paper of J. Rice [14] on a similar topic and the work of J. Traub and others on "analytic computational complexity," cf. [18].

In Section 2, we show that for all reasonable mathematical models, linear approximation algorithms have infinite computational complexity (for the worst case analysis). Moreover, we show that nonlinear, adaptive algorithms are of no assistance in the worst case.

In Section 3, we derive <u>lower</u> and <u>upper</u> bounds for the error in approximating an important class of smooth functions defined on the unit interval [0,1]. For the class of functions under consideration, we show that the subspace of continuous, piecewise linear polynomials with n uniformly spaced knots is an essentially optimal n-dimensional subspace. This demonstrates theme (3).

In Sections 4 and 5, we concentrate on theme (4) and study computationally easy, approximate mappings into subspaces of continuous, piecewise linear polynomials. In Section 4, we introduce and study the mapping which yields a discrete Tchebycheff approximation and in Section 5 we consider the familiar interpolation and least squares projection mappings. Finally in Section 6, we consider the extension of the material of Section 5 to the approximation of functions of two variables defined on a square domain.

Most of the results of this paper can be extended to subspaces of piecewise polynomials of arbitrary degree. What remains is a verification of many technical details many of which have already been provided by deBoor, cf. [2] and [3]. However, our goal in this paper is to present a point of view rather than mathematical generality and virtuosity. Hence, we consider only the technically simple case of piecewise linear polynomials.

§2. A Discouraging Complexity Result

The general mathematical framework for our study of linear approximation algorithms will be an infinite dimensional real Banach space B, i.e., an infinite dimensional complete, normed, vector space over the real field. Our prime example will be the space of all real-valued, continuous functions f defined on the unit interval [0,1] with the maximum norm $\|f\| \equiv \max\{|f(x)| \mid 0 \leq x \leq 1\}$.

If S is an index set, an algorithm for linear approximation in B consists of a set of finite dimensional subspaces of B, $\{B(s) \mid s \in S\}$, and a set of associated mappings $\{M(s) \mid s \in S\}$ such that $M(s): B \to B(s)$. Our prime example of S will be the set of all ordered n-tuples $\Delta: 0 = x_1 < x_2 < \ldots < x_n = 1$ and our prime example of B(s) will be the n-dimensional space

$$L(\Delta) \equiv \{\ell(x) \in C[0,1] \mid \ell(x) \text{ is a linear polynomial on each element } [x_i, x_{i+1}],\ 1 \leq i \leq n-1\},$$

where $n \geq 2$.

We input as data to the algorithm the element $s \in S$ and the element $b \in B$. As output, we obtain M(s)b which we hope is a "good" approximation. By "good" we mean that the error $E(b,s) \equiv \|b - M(s)b\|$ is sufficiently small.

Generally we are given a tolerance $\varepsilon > 0$ and we must select s to guarantee that

(2.1) $E(b,s) \leqq \varepsilon.$

For a <u>worst case analysis</u>, we wish to have (2.1) for all $b \in B$. Hence, we want

(2.2) $E(s) \equiv \sup\{E(b,s) \mid b \in B, \|b\| = 1\} \leqq \varepsilon.$

Clearly we may view ε^{-1} as a fairly accurate parameterization of the computational difficulty of the approximation problem. If $\dim B(s) > \dim B(t)$ for $s,t \in S$, we expect that $E(s) < E(t)$ and $\text{cost } M(s) > \text{cost } M(t)$.

Recent work in complexity theory leads us to investigate the dependence of $E(s)$ on the dimension of $B(s)$. In particular, we would like to find a space B and a linear approximation algorithm such that $E(s) = O((\dim B(s))^{-t})$ as $\dim B(s) \to \infty$, thus giving polynomial complexity. However, it may be that the best we can do is $E(s) = O((\log \dim B(s))^{-1})$ as $\dim B(s) \to \infty$, thus giving exponential complexity.

Unfortunately this problem is a disaster; it has infinite complexity! We will show that we always have $E(s) = 1$. This will show that no matter how clever we are a priori and no matter how much computer time we invest, there will be inputs for which our algorithm computes approximations which are no better than the zero of the Banach space.

Theorem 2.1. For all B, $\{B(s) \mid s \in S\}$, and $s \in S$
$E(s) \equiv \sup\{\|b-M(s)b\| \mid b \in B, \|b\| = 1\} = 1.$

Proof. It suffices to show that for all $s \in S$, there exists $b \in B$ such that $\|b\| = 1$ and $E(s,b) = 1$. Since $B(s)$ is a closed, proper subspace of B, there exists a vector $y \notin B(s)$. If z denotes a best approximation to y in B(s), then the vector $b \equiv (y-z)/\|y-z\|$ has the necessary properties.

QED

There are a number of valuable lessons to be learned from this result. First, we must take our input data from dense, nonclosed subspaces of Banach spaces. Second, worst case analyses may be misleading -- after all people do successfully use linear approximation algorithms in practical situations.

We might hope to be rescued from our difficulties by resorting to nonlinear, adaptive algorithms. However, we will show that these approaches won't help as far as a worst case analysis goes.

We can model nonlinear, adaptive algorithms by assuming the algorithm "chooses" both $s \in S$ and $Mb \in B(s)$. For example, we may consider $L(\Delta)$, where Δ is a set of n knots, and allow our algorithm to vary the n-2 internal knots.

However, under reasonable conditions (which are satisfied in the above example), we can prove an analogue of Theorem 2.1.

Theorem 2.2. If the closure of $B(S) \equiv \bigcup_{s \in S} B(s)$ is a proper subset of B and $M: B \to B(S)$, then

(2.4) $$E(S) \equiv \sup\{\|b-Mb\| \mid b \in B, \|b\| = 1\} = 1.$$

Proof. It suffices to show that there exists $b \in B$ with $\|b\| = 1$

and

$$(2.5)\qquad d(b,B(S)) \equiv \inf\{\|b-y\| \mid y \in B(s),\ s \in S\} = 1.$$

Assume (2.5) is false, i.e., there exists $\delta < 1$ such that

$$(2.6)\qquad d(b,B(S)) \leq \delta$$

for all $b \in B$ with $\|b\| = 1$. Since the closure of $B(S)$, $\overline{B(S)}$, is a proper subset of B, there exists $y \notin \overline{B(S)}$. Let $\{y_k\}_{k=1}^{\infty} \subset B(S)$ be such that $\|y-y_k\| \to d(y,B(S))$ as $k \to \infty$.

The sequence of vectors $v_k \equiv (y-y_k)/\|y-y_k\|$, $k \geq 1$, has $\|v_k\| = 1$ and by (2.6)

$$(2.7)\qquad d(v_k,B(S)) \leq \delta.$$

Thus, $d(y-y_k,B(S)) \leq \delta\|y-y_k\|$. Hence,

$d(y,B(S)) = d(y-y_k,B(S)) \leq \delta\|y-y_k\|$ and taking the limit as $k \to \infty$, we obtain $d(y,B(S)) \leq \delta d(y,B(S)) < d(y,B(S))$, which is a contradiction.

QED

Thus, we have shown that for a worst case analysis nonlinear, adaptive algorithms do not help us. Of course, for particular classes of problems they are very effective, cf. [13].

§3. Lower Bounds

The results of Section 2 suggest that we should restrict our inputs f to our approximation algorithm if we hope to achieve some reasonable results. In 1936, the Russian mathematician Kolmogorov, cf. [8], had the brilliant idea of studying the quantities

$$(3.1) \qquad d_n(A) \equiv \inf_{B_n} \sup_{b \in A} \inf_{b_n \in B_n} \|b-b_n\|,$$

where A is the set of allowable inputs, n is a positive integer, and the infimum is over all n dimensional subspaces B_n of B. Once we know the quantities $d_n(A)$, we have a hold on lower bounds on the complexity of linear approximation algorithms.

For the remainder of this paper we will restrict ourselves to the special case of $B \equiv C[0,1]$ with norm $\|f\| \equiv \max\{|f(x)| \mid 0 \leq x \leq 1\}$ and $A \equiv \{f \mid f \in W^{1,\infty}(0,1) \text{ and } \|Df\| \leq 1\}$, i.e., A is the set of absolutely continuous functions f with $\|Df\| \leq 1$. Following a technique given in [9], we may prove a lower bound due to Tihomirov [17].

Theorem 3.1. $d_n(A) \geq \frac{1}{2n}$.

Proof. Let $B_n \subset C[0,1]$ be any n-dimensional subspace spanned by $\phi_1(x), \ldots, \phi_n(x)$ and Δ_{n+1}: $0 = x_1 < x_2 < \ldots < x_{n+1} = 1$ be the uniform partition with uniformly spaced knots, $x \equiv (i-1)/n$, $1 \leq i \leq n+1$. If A is the $n \times (n+1)$ matrix given by $A \equiv [a_{ij}] \equiv [\phi_i(x_j)]$, the linear system

$$(3.2) \qquad A\underline{c} = \underline{0}$$

has a nontrivial solution $\underline{\tilde{c}}$ such that $\sum_{i=1}^{n+1} |\tilde{c}_i| = 1$.

If $\lambda_i \equiv \text{sign } c_i$, $1 \leq i \leq n+1$, choose $\ell(x) \in L(\Delta_{n+1})$ such that sign $\ell(x_i) \equiv \lambda_i$, $1 \leq i \leq n+1$, and $|\ell(x_i)| \equiv \frac{1}{2n}$. Clearly $\ell(x) \in A$.

Moreover, for all $\underline{a} \in R^n$, we have

$$\|\ell - \sum_{k=1}^{n} a_k\phi_k\| \geq \sum_{i=1}^{n+1} |\tilde{c}_i| \; |\ell(x_i) - \sum_{k=1}^{n} a_k\phi_k(x_i)|$$

$$\geq |\sum_{i=1}^{n+1} \tilde{c}_i \ell(x_i) - \sum_{k=1}^{n} a_k \sum_{i=1}^{n+1} \tilde{c}_i \phi_k(x_i)| = |\sum_{i=1}^{n+1} \tilde{c}_i \ell(x_i)|$$

$$\geq \frac{1}{2n} \sum_{i=1}^{n+1} |\tilde{c}_i| = \frac{1}{2n}.$$

QED

A particular n-dimensional subspace is $L(\Delta_n)$ and as a corollary of the Peano Kernel Theorem, cf. [16], we have the following upper bound.

Theorem 3.2. $\sup_{f \in A} \inf_{\ell \in L(\Delta_n)} \|f - \ell\| \leq \frac{1}{2(n-1)}$.

Since $\frac{1}{2(n-1)} = \frac{1}{2n} + \frac{1}{2n(n-1)}$, we have that $L(\Delta_n)$ is essentially an optimal n-dimensional subspace of $C[0,1]$ with respect to A.

§4. Discrete Tchebycheff Approximation

In view of the results of Section 3, we will concentrate on finding "good" mappings $M(\Delta)$: $C[0,1] \to L(\Delta)$ for all sets of knots Δ: $0 = x_1 < x_2 < \ldots < x_n = 1$. By "good" we mean that (a) there exists a positive constant K such that

(4.1) $\quad \|f-M(\Delta)f\| \leq K\, d(f,L(\Delta))$, for all $f \in C[0,1]$ and all Δ,

and (b) $M(\Delta)f$ is inexpensive to compute, i.e., the number of function evaluations and arithmetic operations needed to compute $M(\Delta)f$ is $O(n^p)$, p a positive integer, for all $f \in C[0,1]$.

In this Section, we describe and analyze an algorithm, which in the context of polynomial subspaces is due to de la Vallée Poussin (1919), cf. [4]. If Ω is any subset of $[0,1]$ and $f \in C[0,1]$, we will use the notation $\|f\|_\Omega \equiv \max\{|f(x)| \mid x \in \Omega\}$. In particular, if $Y \equiv \{y_k \mid 0 \leq k \leq m\} \subset [0,1]$ is a discrete point set, then we may consider the discrete Tchebycheff problem by finding $\ell_Y \in L(\Delta)$ which minimizes $\|f-\ell\|_Y \equiv \max_{y \in Y} |f(y)-\ell(y)|$ over all $\ell \in L(\Delta)$ and we define $M_Y(\Delta)f \equiv \ell_Y$. The problem of constructing ℓ_Y is equivalent to a standard linear programming problem, cf. [15], which may be solved by the simplex algorithm.

We now analyze this algorithm.

Theorem 4.1. If $Y_i \equiv Y \cap [x_i,x_{i+1}] \neq 0$, $|Y_i| \equiv \max_{x \in [x_i,x_{i+1}]} \min_{y \in Y_i} |x-y|$, and $|Y_i| < \frac{1}{2}(x_{i+1}-x_i)$, $1 \leq i \leq n-1$, then

$$\|f-M_Y(\Delta)f\| \leq [2(1-2(x_{i+1}-x_i)^{-1}|Y_i|)^{-1}+1]d(f,L(\Delta)) \tag{4.2}$$

for all $f \in C[0,1]$.

To prove this result we use a basic inequality of A. A. Markov (1889), cf. [11], for polynomials. We state and prove Markov's inequality for the deceptively simple case of linear polynomials.

Lemma. If $\ell(x)$ is a linear polynomial on $[a,b]$, then

$$\|D\ell\|_{[a,b]} \leq 2(b-a)^{-1}\|\ell\|_{[a,b]}. \tag{4.3}$$

Proof. We clearly have

$$|D\ell(x)| = (b-a)^{-1}|\ell(b)-\ell(a)| \leq (b-a)^{-1}(|\ell(b)|+|\ell(a)|)$$
$$\leq 2(b-a)^{-1}\|\ell\|_{[a,b]}.$$

QED

Proof of Theorem 4.1. Let $t \in [x_i,x_{i+1}]$ be such that $|\tilde{\ell}(t)-\ell_Y(t)| = \|\tilde{\ell}-\ell_Y\|$ where $\tilde{\ell}$ is a best approximation to f in $L(\Delta)$. By the Mean Value Theorem,

$$\|\tilde{\ell}-\ell_Y\|_{[x_i,x_{i+1}]} = |\tilde{\ell}(t)-\ell_Y(t)| \leq \|\tilde{\ell}-\ell_Y\|_{Y_i}+|Y_i|\;\|D(\tilde{\ell}-\ell_Y)\|_{[x_i,x_{i+1}]}. \tag{4.4}$$

Using Markov's inequality (4.3) to bound the right hand side of (4.4), we have

$$\|\tilde{\ell}-\ell_Y\|_{[x_i,x_{i+1}]} \leq \|\tilde{\ell}-\ell_Y\|_{Y_i}+2(x_{i+1}-x_i)^{-1}|Y_i|\;\|\tilde{\ell}-\ell_Y\|_{[x_i,x_{i+1}]} \tag{4.5}$$

and hence

$$\|\tilde{\ell}-\ell_Y\|_{[x_i,x_{i+1}]} \leq (1-2(x_{i+1}-x_i)^{-1}|Y_i|)^{-1}\|\tilde{\ell}-\ell_Y\|_{Y_i}. \tag{4.6}$$

Since $\|\tilde{\ell}-f\|_{Y_i} \leq \|\tilde{\ell}-f\|$ and $\|\ell_Y-f\|_{Y_i} \leq \|\tilde{\ell}-f\|_Y \leq \|\tilde{\ell}-f\|$, we have from inequality (4.6),

$$(4.7)\qquad \|\tilde{\ell}-\ell_Y\| = \|\tilde{\ell}-\ell_Y\|_{[x_i,x_{i+1}]}$$
$$\leqq (1-2(x_{i+1}-x_i)^{-1}|Y_i|)^{-1} \cdot (\|\tilde{\ell}-f\|_{Y_i}+\|f-\ell_Y\|_{Y_i})$$
$$\leqq (1-2(x_{i+1}-x_i)^{-1}|Y_i|)^{-1}(\|\tilde{\ell}-f\|+\|f-\ell_Y\|)$$
$$\leqq 2(1-2(x_{i+1}-x_i)^{-1}|Y_i|)^{-1}\|\tilde{\ell}-f\|.$$

Combining (4.7) and the triangle inequality, we have
$\|f-\ell_Y\| \leqq \|f-\tilde{\ell}\|+\|\tilde{\ell}-\ell_Y\| \leqq [1+2(1-2(x_{i+1}-x_i)^{-1}|Y_i|)^{-1}]\ \|f-\tilde{\ell}\|$, which yields (4.2).

QED

Unfortunately we can show that the simplex algorithm for constructing ℓ_Y will require $O(n^2)$ arithmetic operations (not counting function evaluations). As for function evaluations, there are two regimes to investigate. The first is where we can compute f at arbitrary points and we wish to economize, i.e., we want to minimize the number of function evaluations. The second is where we are a priori given large quantities (relative to n) of data or approximate function evaluations — a situation typically arising in the analysis of experimental data which we wish to smooth and compress.

We will discuss the first regime in the remainder of this Section. The second regime will be discussed in Section 5.

By Theorem 4.1, we need $|Y_i| < \frac{1}{2}(x_{i+1}-x_i)$ for all $1 \leqq i \leqq n-1$, which implies that we need at least two function evaluations in the interior of each element $[x_i,x_{i+1}]$. By symmetry, we minimize the coefficient of the right-hand side of (4.2) by evaluating the function f at the points $\frac{1}{4}(3x_i+x_{i+1})$ and $\frac{1}{4}(x_i+3x_{i+1})$. With this choice of evaluation points for each element, $(x_{i+1}-x_i)^{-1}|Y_i| = \frac{1}{4}$ and using (4.2) we obtain the following result.

Corollary. If $Y \equiv \{y_k\}_{k=1}^{2n}$ where $y_{2i-1} \equiv \frac{1}{4}(3x_i+x_{i+1})$ and $y_{2i} \equiv \frac{1}{4}(x_i+3x_{i+1})$, $1 \leqq i \leqq n$,

$$\|f-M_Y(\Delta)f\|_\infty \leqq 5\ d(f,L(\Delta)). \tag{4.8}$$

This algorithm requires 2n evaluations of f.

It might be rather surprising that we can obtain a better result with essentially half of the function evaluations and no arithmetic operations! Suppose we evaluate f only at the knots $\{x_i\}_{i=1}^{n}$, i.e., $Y \equiv \Delta$. Then the preceeding analysis does not quite hold since $|Y_i| = \frac{1}{2}(x_{i+1}-x_i)$. However, in this case the discrete Tchebycheff problem is trivial to solve. In fact, its solution is the piecewise linear interpolate $I_{L(\Delta)}f$ of f. That is, if $x \in [x_i,x_{i+1}]$,
$M_\Delta(\Delta)f(x) \equiv I_{L(\Delta)}f(x) \equiv (x_{i+1}-x_i)^{-1}[f(x_i)(x_{i+1}-x)+f(x_{i+1})(x-x_i)]$.

Fortunately, by a different technique, we can prove a version of (4.2) for this set of data.

Theorem 4.2. If $f \in C[0,1]$,

$$\|f-I_{L(\Delta)}f\| \leq 2d(f,L(\Delta)). \tag{4.9}$$

Proof. If $f \in L(\Delta)$, the result is trivial. Otherwise, let $x \in [x_i, x_{i+1}]$ be such that $|f(x) - I_{L(\Delta)} f(x)| = \|f - I_{L(\Delta)} f\|$.

If $e(x) \equiv f(x) - I_{L(\Delta)} f(x)$, then clearly $e(x_i) = e(x_{i+1}) = 0$ and $\inf_{\ell \in L(\Delta)} \|f-\ell\| = \inf_{\ell \in L(\Delta)} \|e-\ell\|$. Thus, it suffices to show that for all $\ell \in L(\Delta)$

$$\|e-\ell\| \geqq \|e-\ell\|_{[x_i, x_{i+1}]} \geqq \frac{1}{2}\|e\|_{[x_i, x_{i+1}]} = \frac{1}{2}\|e\| .$$

If $|e(x) - \ell(x)| \geqq \frac{1}{2}\|e\|_{[x_i, x_{i+1}]}$ for all $\ell \in L(\Delta)$, we are done.

Otherwise, $e(x)$ and $\ell(x)$ have the same sign and $|\ell(x)| \geqq \frac{1}{2}\|e\|_{[x_i, x_{i+1}]}$. This implies that $|\ell(x_k)| \geqq \frac{1}{2}\|e\|_{[x_i, x_{i+1}]}$ for k = either i or $i+1$.

Since $e(x_i) = e(x_{i+1}) = 0$, this implies that either $|e(x_i) - \ell(x_i)| \geqq \frac{1}{2}\|e\|_{[x_i, x_{i+1}]}$ or $|e(x_{i+1}) - \ell(x_{i+1})| \geqq \frac{1}{2}\|e\|_{[x_i, x_{i+1}]}$.

QED

This is the ideal situation for those problems in which we can evaluate a function arbitrarily. This interpolation algorithm has additional desirable features, such as linearity, which will be considered in detail in the next Section.

§5. Least Squares Algorithms

The problems in which we have a large quantity of data which we wish to smooth and compress have not been satisfactorily resolved as yet. Moreover, it is of further interest to have a projection algorithm; that is, one based on projection mappings. To be precise, a linear approximation algorithm is said to be a linear projection algorithm if and only if the associated mappings $M(s)$ are linear and are such that $M(s)y = y$ for all $y \in B(s)$, i.e., the mappings $M(s)$ are linear projectors.

Following ideas of Kantorovich, Lax, and deBoor, cf. [7], we have the following equivalence result for linear projection algorithms.

Theorem 5.1. Let the sequence of subspaces $\{B_n \mid 1 \leq n < \infty\}$ be such that $\lim_{n\to\infty} \inf_{b_n \in B_n} \|b-b_n\| = 0$ for all $b \in B$ and the sequence of mappings $\{M_n \mid 1 \leq n < \infty\}$ be linear projectors which are consistent, i.e., there exists a dense subspace, D, of B such that $\lim_{n\to\infty} \|M_n b-b\| = 0$ for all $b \in D$. The following conditions are equivalent: (a) there exists $\varepsilon > 0$ such that $\|M_n\| \leq \varepsilon$ for all n; (b) $\|b-M_n b\| \leq (1+\varepsilon)d(b,B_n)$, for all n and $b \in B$; and (c) $\|b-M_n b\| \to 0$ as $n \to \infty$ for all $b \in B$.

Proof. <u>(a) ⇒ (b)</u>: For all $g \in B_n$,

$$\|b-M_n b\| \leq \|b-g\|+\|g-M_n b\| = \|b-g\|+\|M_n(g-b)\| \leq (1+\varepsilon)\|b-g\|.$$

<u>(b) ⇒ (c)</u>: Obvious.

<u>(c) ⇒ (a)</u>: If $b \in B$, then $\{\|M_n b\| \mid n \geq 1\}$ is bounded. For

otherwise there exists a sequence of elements $M_{n_1}b, \ldots, M_{n_j}b$ whose norms tend to infinity which contradicts convergence. Condition (a) follows from the uniform boundedness principle of functional analysis.

QED

This equivalence theorem is quite general. Let us consider some elementary applications.

Examples:

(1) $B \equiv C[0,1]$, $B_\Delta \equiv L(\Delta)$, and $M(\Delta) \equiv \mathcal{I}_{L(\Delta)}$. Then $\|\mathcal{I}_{L(\Delta)}\| = 1$ and hence $\|f-\mathcal{I}_{L(\Delta)}f\| \leqq 2d(f,L(\Delta))$, which is the result of Theorem 4.2

(2) Let $B \equiv C(U)$, $\rho \equiv \Delta_x \times \Delta_y$ be the product grid on the unit square, U, and consider the tensor product space $B_\rho \equiv L(\rho) \equiv L(\Delta_x) \otimes L(\Delta_y)$. If $\mathcal{I}_{L(\rho)}$ denotes the two dimensional interpolation mapping into $L(\rho)$, cf. [16], then $\|\mathcal{I}_{L(\rho)}\| = 1$ and hence $\|f-\mathcal{I}_{L(\rho)}f\| \leqq 2d(f,L(\rho))$.

(3) Let T be a triangulated polygon, L(T) be the space of continuous, piecewise linear polynomials with respect to T, and $\mathcal{I}_{L(T)}$ be the obvious mapping of $B \equiv C(T)$ into $B_T \equiv L(T)$. Then, once again $\|\mathcal{I}_{L(T)}\| = 1$ and $\|f-\mathcal{I}_{L(T)}f\| \leqq 2d(f,L(T))$.

We now consider least squares algorithms. Given $f \in C[0,1]$, we determine $\tilde{\ell} \in L(\Delta)$ which minimizes $\int_0^1\int_0^1 (f(x)-\ell(x))^2dx \equiv \|f-\ell\|_2^2$ over $\ell \in L(\Delta)$; i.e., $\tilde{\ell} \equiv P(\Delta)f$ is the orthogonal projection of f onto $L(\Delta)$. Expressing $\tilde{\ell}(x) = \sum_{i=1}^{n} \tilde{\beta}_i \ell_i(x)$, where $\ell_i(x)$ is the unique element in $L(\Delta)$ defined by $\ell_i(x_j) = \delta_{ij} \equiv \begin{cases} 1 & \text{if } i = j \\ 0 & \text{otherwise} \end{cases}$, the minimization problem is equivalent to solving

$$A\underset{\sim}{\beta} = \underline{k}, \tag{5.1}$$

where A is the n×n matrix given by

$$A \equiv \begin{bmatrix} \frac{1}{3}x_2 & \ddots & & & 0 \\ \ddots & \ddots & \ddots & & \\ \frac{1}{6}(x_i-x_{i-1}) & & \frac{1}{3}(x_{i+1}-x_{i-1}) & & \frac{1}{6}(x_{i+1}-x_i) \\ & \ddots & & \ddots & \ddots \\ 0 & & & & \frac{1}{3}(1-x_{n-1}) \end{bmatrix} \tag{5.2}$$

and

$$\underline{k} \equiv [\int_0^1 f(x)\ell_i(x)dx]. \tag{5.3}$$

For the case of a uniform mesh with $h \equiv x_{i+1}-x_i$, $1 \leq i \leq n-1$.

$A = \frac{h}{6}\begin{bmatrix} 2 & 1 & & & 0 \\ 1 & 4 & \ddots & & \\ & \ddots & \ddots & \ddots & \\ & & \ddots & 4 & 1 \\ 0 & & & 1 & 2 \end{bmatrix}$. Using the elementary fact that

$\|\sum_{i=1}^{n} \beta_i \ell_i\| = \|\underline{\beta}\|_\infty \equiv \max_{1\leq i\leq n} |\beta_i|$ for all $\underline{\beta} \in R^n$, we can obtain the following

result for the mappings P_Δ.

Theorem 5.2. For all $f \in C[0,1]$, $\|f-P_\Delta f\| \leq 4d(f,L(\Delta))$.

Proof. If $D \equiv [d_{ij}]$ is the diagonal matrix with $d_{11} \equiv 3x_2^{-1}$, $d_{ii} \equiv 3(x_{i+1}-x_{i-1})^{-1}$, $2 \leq i \leq n-1$, and $d_{nn} \equiv 3(1-x_{n-1})^{-1}$, i.e., $\tilde{D}^{-1}$ is the diagonal of A, then $A\underset{\sim}{\beta} = \underline{k}$ implies

$$\text{(5.4)} \qquad D A \tilde{\underline{\beta}} = D \underline{k},$$

where $DA \equiv I+M$, $\|M\|_\infty \equiv \max_{1 \leq i \leq n} \sum_{j=1}^{n} |m_{ij}| = \frac{1}{2}$, and $\|D\underline{k}\|_\infty \leq \frac{3}{2}\|f\|$.

It follows that $\|(DA)^{-1}\|_\infty = \|(I+M)^{-1}\|_\infty \leq 2$, cf. [1], and hence that $\|\tilde{\underline{\beta}}\|_\infty \leq 3\|f\|$. The result now follows from Theorem 5.1.

QED

What about computing (5.4) and its solution? If we can compute the integrals needed to evaluate $\underline{k}$, then we need only $O(n)$ arithmetic operations to compute the linear system (5.4). Moreover, we can solve the tridiagonal linear system in $O(n)$ arithmetic operations by means of Gaussian elimination for tridiagonal matrices, cf. [1].

From the viewpoint of round-off error analysis, it is important to know about the conditioning of DA. In this case, it turns out that the condition number of DA, i.e., $\|(DA)^{-1}\|_\infty \|DA\|_\infty$, is uniformly bounded (independent of Δ) by $2 \cdot 3 = 6$. Thus, we have an ideal situation.

To handle discrete data, we suggest an algorithm of Patent, cf. [12]. We note that the data occurs only in the right-hand side of the linear system (5.1). Thus, if the data is at Y, we choose $\Delta \subset Y$, interpolate f at Y by $\mathcal{I}_{L(Y)}f$, replace f in the right-hand side of (5.1), and compute the resulting integrals whose integrands are piecewise quadratics by either symbolic methods or Gaussian quadrature with two nodes in each element defined by Y. That is we solve the linear system

$$\text{(5.4)} \qquad A\hat{\underline{\beta}} = \hat{\underline{k}} \equiv [\textstyle\int_0^1 \mathcal{I}_{L(Y)} f(x) \ell_i(x)\,dx],$$

and let $\hat{M}(\Delta)f \equiv \sum_{i=1}^{n} \hat{\beta}_i \ell_i(x)$. We can prove the following error bound for this procedure.

Theorem 5.3. If $\Delta \subset Y$, then for all $f \in C[0,1]$,

$$\|f-\hat{M}(\Delta)f\| \leq 4d(f,L(\Delta)). \tag{5.5}$$

Proof. By Theorem 5.1, it suffices to show that $\|\hat{M}_\Delta\| \leq 3$. Since $\Delta \subset Y$, $L(\Delta) \subset L(Y)$ and $\hat{M}(\Delta) = P_\Delta \mathcal{I}_{L(Y)}$.

Hence, $\|\hat{M}(\Delta)\| \leq \|P_\Delta\| \; \|\mathcal{I}_{L(Y)}\| \leq 3 \cdot 1 = 3$.

QED

§6. A Two Dimensional Extension

In this Section, we consider the problem of approximating a function $f(x,y)$ of two variables on the unit square U by means of bilinear functions in $L(\rho)$, where $\rho \equiv \Delta_x \times \Delta_y$ is a product rectangular mesh. Our least squares algorithm seeks $P_\rho f \equiv \tilde{\ell}(x,y) \in L(\rho)$ which minimizes

$$(6.1) \qquad \int_0^1 \int_0^1 (f(x,y)-\ell(x,y))^2 dxdy$$

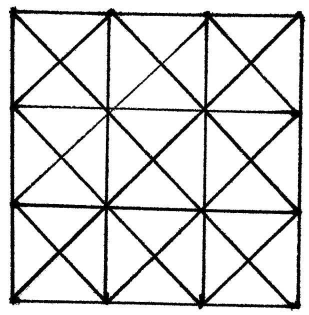
over $\ell \in L(\rho)$. If $\tilde{\ell}(x,y) = \sum_{i=1}^{n} \sum_{j=1}^{n} \tilde{\beta}_{ij} \ell_i(x) \ell_j(y)$, then it is easy to show that $\underline{\tilde{\beta}} \equiv \tilde{\beta}_{n(i-1)+j} \equiv \tilde{\beta}_{ij}$, $1 \leq i,\ j \leq n$, is the unique solution of the linear system

$$(6.2) \qquad A_x \otimes A_y \underline{\tilde{\beta}} = \underline{k},$$

where A_x and A_y are the one dimensional least squares matrices with respect to $L(\Delta_x)$ and $L(\Delta_y)$ respectively and $A_x \otimes A_y$ is the Kronecker product, i.e.,

$$[a_{ij}] \otimes [b_{ij}] \equiv \begin{bmatrix} a_{11}[b_{ij}] & \cdots & a_{1n}[b_{ij}] \\ \vdots & \ddots & \vdots \\ a_{n1}[b_{ij}] & \cdots & a_{nn}[b_{ij}] \end{bmatrix}.$$

Note that we have used the "natural ordering along vertical lines of ρ" for the solution vector $\underline{\tilde{\beta}}$. The matrix $A_x \otimes A_y$ is sparse. In fact, it has only nine nonzero diagonals and a bandwidth of n+2. For the special case of n = 2, its associated graph is



Of course, we would like to prove an error bound which is a two dimensional analogue of Theorem 5.2. We may do this with the aid of some elementary results about Kronecker products.

Theorem 6.1. For all $f \in C(U)$,

(6.3) $\|f-P_\rho f\| \leqq 10d(f,L(\rho))$.

Proof. If D_x and D_y are the diagonal matrices defined in the proof of Theorem 5.2, then $(D_xA_x \otimes D_yA_y)\underset{\sim}{\beta} = (D_x \otimes D_y)(A_x \otimes A_y)\underset{\sim}{\beta} = (D_x \otimes D_y)\underline{k}$, where we have used the fact that the product of tensor products is the tensor product of the products, cf. [10]. As in the proof of Theorem 5.2, $D_xA_x \equiv I+M_x$ and $D_yA_y \equiv I+M_y$ where $\|M\|_\infty = \frac{5}{4}$. Hence, $D_xA_x \otimes D_yA_y$ is <u>not</u> diagonally dominant. However, the inverse of a tensor product is the tensor product of the inverses and the ∞-norm of a tensor product is less than or equal to the product of the ∞-norms.

Thus, $\|(D_xA_x \otimes D_yA_y)^{-1}\|_\infty \leqq \|(D_xA_x)^{-1}\|_\infty \|(D_yA_y)^{-1}\|_\infty \leqq 2\cdot 2 = 4$. Moreover, $\|\underline{k}\|_\infty \leqq \frac{9}{4}\|f\|$. Hence, $\|P_\rho f\| = \|\underline{\beta}\|_\infty \leqq 4\cdot\frac{9}{4}\|f\| = 9\|f\|$, and the result follows from Theorem 5.1.

QED

If the data is given as a discrete function on a rectangular grid Y including ρ as a subset, we may prove a two dimensional analogue of Theorem 5.3. The details are left to the reader.

An interesting and important issue is the choice of algorithms for solving linear systems of the form (6.2). If we use band or profile Gaussian elimination, we need $O(n^3)$ storage locations and $O(n^4)$ arithmetic operations, cf. [6]. If we use sparse matrix techniques, then the best we can do is $O(n^2 \ln n)$ storage locations and $O(n^3)$ arithmetic operations, cf. [5] and [6]. These latter results hold for J. A. George's "nested ordering" of the unknowns.

However, using the special structure of the equations, we can achieve an "alternating direction" direct method which requires $O(n^2)$ storage locations and $O(n^2)$ arithmetic operations. To start we observe that it suffices to solve the coupled systems

$$(6.4)\qquad (I\otimes A_y)\underline{w} = \underline{k}$$

$$(6.5)\qquad (A_x\otimes I)\underset{\sim}{\underline{\beta}} = \underline{w}\ .$$

In fact, if $\underline{w}$ and $\underset{\sim}{\underline{\beta}}$ satisfy (6.4) and (6.5)

$$(I\otimes A_y)(A_x\otimes I)\underset{\sim}{\underline{\beta}} = (I\otimes A_y)\underline{w} = \underline{k}.$$

But $I\times A_y \equiv \begin{bmatrix} A_y & 0 \\ 0 & \ddots\, A_y \end{bmatrix}$ is an n×n block diagonal matrix with n×n blocks.

If we partition $\underline{w}$ and $\underline{k}$ into the corresponding n-block vectors, we have

$$(6.6)\qquad A_y\underline{w}_\ell = \underline{k}_\ell,\ 1 \leq \ell \leq n.$$

Moreover, each system in (6.6) can be solved with $O(n)$ storage locations and $O(n)$ arithmetic operations. Since there are n such systems, we need a total of $O(n^2)$ storage locations and $O(n^2)$ arithmetic operations to compute $\underline{w}$.

To solve (6.5) efficiently, we define $\hat{\underline{\beta}}$ and $\hat{\underline{w}}$ by reordering the components of $\tilde{\underline{\beta}}$ and $\underline{w}$ to correspond to the "natural ordering along horizontal lines of ρ," e.g. $\hat{\beta}_{n(j-1)+i} \equiv \tilde{\beta}_{ij}$, $1 \leqq i, j \leqq n$. Then, (6.5) may be rewritten as

$$(6.7) \qquad (I \otimes A_x)\hat{\underline{\beta}} = \hat{\underline{w}}.$$

This system may be solved the same way we solved (6.6) with $O(n^2)$ storage locations and $O(n^2)$ arithmetic operations, and $\tilde{\beta}_{ij}$, $1 \leqq i, j \leqq n$, may be reconstructed from $\hat{\underline{\beta}}$.

A moral of this analysis is that a sparse linear system with a special structure may often be solved more efficiently by means of a special algorithm than by general sparse matrix algorithms.

Acknowledgement

The author is grateful to Professor S. C. Eisenstat for many helpful discussions regarding the content of this paper.

References

[1] Blum, E. K., Numerical Analysis and Computation: Theory and Practice. Addison-Wesley Publishing Co., Reading (1972).

[2] deBoor, C., On the convergence of odd-degree spline interpolation. Journal of Approximation Theory 1, 452-463 (1968).

[3] deBoor, C., On uniform approximation by splines. Journal of Approximation Theory 1, 219-235 (1968).

[4] de la Vallée Poussin, C., Lecons sur l'Approximation des Fonctions d'une Variable Réelle. Gauthier-Villars, Paris (1919).

[5] George, J. A., Nested disection of a regular finite element mesh. SIAM Journal on Numerical Analysis 10, 345-363 (1973).

[6] Hoffman, A. J., M. S. Martin, and D. J. Rose, Complexity bounds for regular finite difference and finite element grids. SIAM Journal on Numerical Analysis 10, 364-369 (1973).

[7] Kantorovich, L. V., Functional analysis and applied mathematics. Uspekhi Matematicheskikh Nauk 3, 89-185 (1948).

[8] Kolmogorov, A. N., Über die beste Annäherung von Funktionen einer gegebenen Funktionenklasse. Ann. of Math. 37, 107-111 (1936).

[9] Lorentz, G. G., Approximation of Functions. Holt, Rinehart, and Winston, New York (1966).

[10] Marcus, M., and H. Minc, A Survey of Matrix Theory and Matrix Inequalities. Allyn and Bacon, Inc., Boston (1964).

[11] Markov, A. A., On a problem of D. I. Mendeleev. St. Petersburg, Izv. Akad. Nauk 62, 1-24 (1889).

[12] Patent, P. D., Least square polynomial spline approximation. Doctoral Thesis, California Institute of Technology (1972).

[13] Rice, J. R., On the degree of convergence of nonlinear spline approximation. Approximations with Special Emphasis on Spline Functions (1969).

[14] Rice, J. R., On the computational complexity of approximation operators. Austin meeting on Approximation Theory, January (1973).

[15] Schultz, M. H., Discrete Tchebycheff approximation for multivariate splines. Journal of Computer and System Sciences 6, 298-304 (1972).

[16] Schultz, M. H., Spline Analysis. Prentice-Hall, Inc. Englewood Cliffs (1973).

[17] Tihomirov, V. M., Widths of sets in functional spaces and the theory of best approximations. Uspehi 15, 81-120 (1960).

[18] Traub, J. F., Computational complexity of iterative processes. SIAM Journal on Computing 1, 167-179 (1972).