27
3-31-76
2-
up to TIS

UCID- 17059

Lawrence Livermore Laboratory

PIECEWISE POLYNOMIAL FUNCTIONS

R.E. Carlson

MASTER

January 1976

This is an informal report intended primarily for internal or limited external distribution. The opinions and conclusions stated are those of the author and may or may not be those of the laboratory.

Prepared for U.S. Energy Research & Development Administration under contract No. W-7405-Eng-48.

DISTRIBUTION OF THIS DOCUMENT IS UNLIMITED

DISCLAIMER

This report was prepared as an account of work sponsored by an agency of the United States Government. Neither the United States Government nor any agency Thereof, nor any of their employees, makes any warranty, express or implied, or assumes any legal liability or responsibility for the accuracy, completeness, or usefulness of any information, apparatus, product, or process disclosed, or represents that its use would not infringe privately owned rights. Reference herein to any specific commercial product, process, or service by trade name, trademark, manufacturer, or otherwise does not necessarily constitute or imply its endorsement, recommendation, or favoring by the United States Government or any agency thereof. The views and opinions of authors expressed herein do not necessarily state or reflect those of the United States Government or any agency thereof.

DISCLAIMER

Portions of this document may be illegible in electronic image products. Images are produced from the best available original document.

TABLE OF CONTENTS

1. Introduction	1
2. Univariate Functions	3
3. Cubic Splines	8
4. Bivariate Functions	18
5. Blending Functions	25
6. Nonrectangular Domains	29
Appendix	33
References	39

NOTICE

This report was prepared as an account of work sponsored by the United States Government. Neither the United States nor the United States Energy Research and Development Administration, nor any of their employees, nor any of their contractors, subcontractors, or their employees, makes any warranty, express or implied, or assumes any legal liability or responsibility for the accuracy, completeness or usefulness of any information, apparatus, product or process disclosed, or represents that its use would not infringe privately owned rights.

PIECEWISE POLYNOMIAL FUNCTIONS

R.E. Carlson*

1. INTRODUCTION

This report is the written version of a series of talks presented Summer 1975 at LLL on the general topic of piecewise polynomial functions. Such functions offer simple, accurate, economical, and flexible procedures for solving a wide variety problems. Our concern lies primarily with curve fitting, surface fitting, and applications to finite element approximations. This discussion is not intended as an all-inclusive, general presentation of piecewise polynomials. Rather, we focus on the basics of cubic (and bicubic) interpolation and approximation, since they represent a good first choice for many problems when increased accuracy over linear (or bilinear) approximations is desired.

The following table outlines specifically the spaces of functions to be considered in this report.

<u>Functions</u>	<u>Univariate</u>	<u>Rectangles</u>	<u>Bivariate Polygons</u>	<u>Curved Boundaries</u>
Hermite	Yes	Yes	Yes	No
Spline	Yes	Yes	No	No
Blending	No	Yes	Yes	Yes
Quasi-Polynomials	No	No	Yes	Yes

The blending functions referred to above were first developed by Dr. William Gordon of General Motors Research Laboratories in the late 1960's. Quasi-polynomials were developed by the author in 1970 and represent the only functions in the report not discussed in the literature.

*Current address: Department of Mathematics, Grove City College, Grove City, PA 16127.

Since our goal is an easy to read, easy to understand discussion of the topic of piecewise polynomial and related functions, no proofs of theorems are included. Instead, the appropriate papers from the literature are noted along with some occasional comments. An extensive bibliography of related reference material is included also.

Before beginning our discussion of univariate functions there are two terms relevant to interpolation schemes which must be defined.

Definition 1: An interpolation scheme is said to be algebraically well set if, for any appropriately defined smooth function f and partition π of the domain D , the interpolant of f exists and is unique.

Definition 2: An interpolation scheme is said to be analytically well set if, as the partition π is successively refined, the associated sequence of interpolants of f converges to f .

2. UNIVARIATE FUNCTIONS

Let I denote the closed interval $[a, b]$ and let π be the partition $a = x_0 < x_1 < \dots < x_n = b$. The x_i are called mesh points, nodes, knots, or joints, and are distinct. By a piecewise polynomial function on (I, π) we mean a function $p(x)$ which is a polynomial on each subinterval, i.e.

$$p(x) \equiv p_i(x) \text{ for } x_{i-1} \leq x < x_i, \quad i = 1, 2, \dots, n$$

where $p_i(x)$ is a polynomial.

Under the usual definition of addition and scalar multiplication, these functions form a vector space over the reals. If the degree of each $p_i(x)$ is restricted to be less than some positive integer M , then the vector space is finite dimensional. Of particular interest here is the case $M=3$, the space of piecewise cubics (or less). Let us denote this space by $P^2(I, \pi)$.

Remark: In a more general setting $P^n(I, \pi)$ denotes the space of piecewise polynomials of degree $2n-1$.

It is easily seen that $\dim \left(P^2(I, \pi) \right) = 4n = 4 \cdot (\text{number of intervals})$. In general, functions in $P^2(I, \pi)$ are piecewise analytic, but discontinuous at the knots.

Two subspaces of $P^2(I, \pi)$ are of interest to us. The first is the cubic Hermite space defined by

$$H^2(I, \pi) = P^2(I, \pi) \cap C^1[I]$$

where $C^1[I]$ is the space of all functions with domain I which are continuous and have continuous first derivatives.

It is clear that in each subinterval $[x_i, x_{i+1}]$, $u(x) \in H^2(I, \pi)$ is uniquely determined by the values $u(x_i^+)$, $u(x_{i+1}^-)$, $u'(x_i^+)$, $u'(x_{i+1}^-)$ where x_i^+ means $x \rightarrow x_i$ from above and x_{i+1}^- means $x \rightarrow x_{i+1}$ from below. However, $u(x_i^+) = u(x_i^-)$ and $u'(x_i^+) = u'(x_i^-)$, $i = 1, 2, \dots, n-1$ because $u(x) \in C^1[I]$; thus the + and - are superfluous and will be dropped.

The above representation for $u(x)$ suggests the following algebraically well set cubic Hermite interpolation scheme.

Definition 3: Let f be an appropriate function with domain I and partition π . Then the unique cubic Hermite polynomial $u_f(x)$ satisfying

$$(1) \quad \left. \begin{aligned} u_f(x_i) &= f(x_i) \\ u'_f(x_i) &= f'(x_i) \end{aligned} \right\} i = 0, 1, 2, \dots, n$$

is called the cubic Hermite interpolant of f .

Remark: f need not be a function but merely a set of values interpreted as in (1) to define $u_f(x)$.

It is an immediate consequence of Definition 3 that

$$(2) \quad \text{Dim} \left(H^2(I, \pi) \right) = 2(n+1).$$

Since $u_f(x)$ interpolates not only values of $f(x)$ but the first derivative of $f(x)$ at the nodes (i.e. u_f has second order contact with f), Hermite interpolation is a special case of osculatory interpolation (see [16]).

The next question that arises is how "good" of an approximation is u_f to f ? Obviously the answer depends upon what is meant by "good" and what properties f satisfies. One answer to this question for smooth

functions f is contained in

Theorem 1: [9] If $f \in C^4[I]$, then $\|(u_f - f)^{(k)}\|_{\infty} \leq \alpha_k \|f^{(4)}\|_{\infty} h^{4-k}$,
 $k = 0, 1, 2, 3$ where $h = \max_{1 \leq i \leq n} (x_i - x_{i-1})$, $\alpha_0 = 1/384$, $\alpha_1 = \sqrt{3}/216$, $\alpha_2 = 1/12$
 and $\alpha_3 = 1/2$.

Remark: The above theorem states that u_f is a fourth order approximation to f and the k th derivative of u_f is a $(4-k)$ th order approximation to $f^{(k)}$, provided $f \in C^4[I]$. If less smoothness is required of f , then the order of the approximation will decrease correspondingly (for details, see [10,14]).

Having defined an algebraically well set interpolation scheme in (1) and having shown that it is analytically well set (Theorem 1), the next step is to consider how to represent u_f in a convenient form. This problem is equivalent to finding an appropriate basis for $H^2(I, \pi)$.

An interpolation basis is obtained by setting one of the interpolation parameters in (1) equal to 1 and all others equal to zero. Two types of functions are included in this basis. The first type occurs when one of the function values is set equal to one, the second when one of the derivatives is set equal to one. The former is defined by

$$b_i(x_k) = \delta_{ik}$$

$$i = 0, 1, \dots, n, \quad k = 0, 1, \dots, n$$

$$b'_i(x_k) = 0$$

Then $b_i(x)$ is nonzero over (at most) two subintervals and is shown below for $i \neq 0, n$.

Figure 1. $b_i(x)$

The second type of basis function is defined by

$$\left. \begin{aligned} c_i(x_k) &= 0 \\ c_i'(x_k) &= \delta_{ik} \end{aligned} \right\} \quad i = 0, 1, \dots, n; \quad k = 0, 1, \dots, n$$

Then $c_i(x)$ is nonzero over (at most) two subintervals and is shown below.

Figure 2. $c_i(x)$

The Hermite basis functions defined above have the following important properties:

1. The basis functions are local in the sense they are nonzero over (at most) two subintervals.

$$2. \quad u_f(x) = \sum_{i=0}^n \{f(x_i)h_i(x) + f'(x_i)c_i(x)\}$$

3. In the interval $[x_k, x_{k+1}]$ u_f can be written in the form

$$u_f(x) = f(x_k) H_1(x) + f(x_{k+1}) H_2(x) + f'(x_k) H_3(x) + f'(x_{k+1}) H_4(x)$$

where

$$H_1(x) = \left(\frac{1}{\Delta x_k} \right)^3 \left(2(x-x_k)^3 - 3\Delta x_k (x-x_k)^2 + (\Delta x_k)^3 \right)$$

$$\begin{aligned}
 H_2(x) &= \left(\frac{1}{\Delta x_k} \right)^3 \left(-2(x-x_k)^3 + 3\Delta x_k(x-x_k)^2 \right) \\
 (3) \quad H_3(x) &= \left(\frac{1}{\Delta x_k} \right)^2 \left((x-x_k)^3 - 2\Delta x_k(x-x_k)^2 + (\Delta x_k)^2(x-x_k) \right) \\
 H_4(x) &= \left(\frac{1}{\Delta x_k} \right)^2 \left((x-x_k)^3 - \Delta x_k(x-x_k)^2 \right)
 \end{aligned}$$

and $\Delta x_k = x_{k+1} - x_k$. (The functions $H_i(x)$ are simply the basis functions $b_i(x)$ and $c_i(x)$ restricted to one subinterval.)

There is an obvious generalization to the cubic Hermite space defined above, and that is to consider piecewise polynomial functions of degree $2m-1$. In this case we define

$$H^m(I, \pi) = P^m(I, \pi) \cap C^{m-1}[I]. \quad \text{The interpolation conditions are}$$

$$u_f^{(k)}(x_i) = f^{(k)}(x_i) \quad , \quad i = 0, 1, \dots, n; \quad k = 0, 1, \dots, m-1.$$

The convergence results are of the form

$$\|u_f - f\|_\infty \leq C \|f^{(2m)}\|_\infty h^{2m}.$$

For details, see [10].

3. CUBIC SPLINES

Spline functions were first studied by Schoenberg in the 1940's [31]. Little was done with splines until the 1960's when much interest began to develop — both from a practical and a theoretical point of view. Our approach here differs somewhat from that of Schoenberg. We emphasize the relationship between splines and Hermite functions and do not emphasize the variational properties. We begin with the following definition of cubic splines.

Definition 4: The space $Sp^2(I, \pi)$ of cubic splines is the subspace of $H^2(I, \pi)$ defined by

$$Sp^2(I, \pi) = H^2(I, \pi) \cap C^2[I].$$

That is, cubic splines are cubic Hermite functions whose second derivatives are continuous across mesh points. To see what this condition implies, consider two subintervals $[x_{i-1}, x_{i+1}]$ shown below.

Figure 3

The cubic Hermite function $u(x)$ in $[x_{i-1}, x_{i+1}]$ is completely determined by the six parameters $u(x_k)$, $k = i-1, i, i+1$ and $u'(x_k)$, $k = i-1, i, i+1$. The second derivative of $u(x)$ for x in $[x_{i-1}, x_i]$ evaluated at $x = x_i$ is

$$u''(x_i^-) = \frac{6u_{i-1}}{h_{i-1}^2} - \frac{6u_i}{h_{i-1}^2} + \frac{2}{h_{i-1}} u'_{i-1} + \frac{4}{h_{i-1}} u'_i$$

Similarly, for the interval $[x_i, x_{i+1}]$ we obtain

$$u''(x_i^+) = \frac{-6}{h_i^2} u_i + \frac{6}{h_i^2} u_{i+1} - \frac{4}{h_i} u_i' - \frac{2}{h_i} u_{i+1}'$$

Equating $u''(x_i^-)$ with $u''(x_i^+)$ and multiplying by $\frac{h_{i-1} h_i}{2}$ yields

$$(4) \quad h_i u_{i-1}' + 2(h_{i-1} + h_i) u_i' + h_{i-1} u_{i+1}' = -3 \frac{h_i}{h_{i-1}} u_{i-1} + 3 \left(\frac{h_i}{h_{i-1}} - \frac{h_{i-1}}{h_i} \right) u_i + 3 \frac{h_{i-1}}{h_i} u_{i+1}$$

This equation must be valid for each interior mesh point, i.e., $i = 1, 2, \dots, n-1$. Therefore, by requiring the second derivative of a cubic Hermite polynomial to be continuous, we introduce $n-1$ conditions on the $2(n+1)$ Hermite coordinates, and we conclude $\dim Sp^2(I, \pi) = 2(n+1) - (n-1) = n+3$. Thus, an algebraically well set spline interpolation scheme involves specifying $n+3$ conditions on the Hermite coordinates such that (4) is satisfied, and a unique function in $Sp^2(I, \pi)$ is determined. One such scheme is given below.

Definition 5: Let f be a function defined on the interval I which has been partitioned by a mesh π . Then the cubic spline interpolant of f is defined by

$$(5) \quad \begin{aligned} s_f(x_i) &= f(x_i) & i &= 0, 1, \dots, n \\ s_f'(x_j) &= f'(x_j) & j &= 0, n. \end{aligned}$$

Given the interpolation data above, the remainder of the Hermite coordinates of the spline (i.e. $s_f'(x_i)$, $i = 1, 2, \dots, n-1$) are obtained by solving the system of $n-1$ linear equation defined by (4). This system is

tri-diagonal and diagonally dominant, hence very stable. Once the Hermite coordinates of $s_f(x)$ are found for any subinterval $[x_i, x_{i+1}]$, we can write

$$s_f(x) = s_f(x_i)H_1(x) + s_f(x_{i+1})H_2(x) + s'_f(x_i)H_3(x) + s'_f(x_{i+1})H_4(x)$$

$$\text{where } s_f(x_j) = f(x_j) \quad j = i, i+1$$

$$s'_f(x_j) = f'(x_j) + \epsilon_j \quad j = i, i+1.$$

Note that $s_f(x)$ does not interpolate the derivatives of $f(x)$ at the interior nodes. Bounds for the error in $s'_f(x_j)$, i.e. ϵ_j , can be derived from the matrix problem. To show that the interpolation scheme is analytically well set for appropriate functions f , one considers

$$||s_f - f|| = ||s_f - u_f + u_f - f|| \leq ||s_f - u_f|| + ||u_f - f||$$

A bound for $||u_f - f||$ is given in Theorem 1. At a point $x \in [x_i, x_{i+1}]$,

$$s_f(x) - u_f(x) = \epsilon_i H_3(x) + \epsilon_{i+1} H_4(x)$$

By calculating a bound for $|\epsilon_j|$ and knowing $H_3(x)$ and $H_4(x)$ explicitly, one is able to construct a bound for $||s_f - u_f||$. (see [24]). Specifically, we have

Theorem 2: If $f \in C^4[I]$, then

$$|| (s_f - f)^{(k)} ||_{\infty} < \beta_k || f^{(4)} ||_{\infty} h^{4-k}, \quad k = 0, 1, 2, 3$$

where $h = \max_{1 \leq i \leq n} (x_i - x_{i-1})$, $\beta_0 = 5/384$, $\beta_1 = \frac{9+\sqrt{3}}{216}$, $\beta_2 = \frac{3\mu+1}{12}$,

$\beta_3 = \frac{\mu^2+1}{2}$ and μ is the mesh ratio, $\mu = \left(\frac{h}{\min_{1 \leq j \leq n} (x_j - x_{j-1})} \geq 1 \right)$.

Remark: The above theorem states that s_f is a fourth order approximation to f and that the k th derivative is a $(4-k)$ th order approximation is $f^{(k)}$, provided $f \in C^4[I]$. These results exactly match those of the cubic Hermite case, despite the fact that the number of interpolation conditions (i.e. $\dim Sp^2(I, \pi)$) is roughly half of that of $H^2(I, \pi)$. This fact (among others) has accounted for the increased use of spline functions to solve many types of problems.

Our next consideration is the construction of an interpolation basis for cubic splines. This basis is constructed by setting one of the interpolation parameters equal to 1 and all others 0. Two types of basis functions arise from this procedure. Type 1, consisting of $n+1$ basis functions which satisfy

$$\phi_i(x_j) = \delta_{ij}, \quad i, j = 0, 1, \dots, n, \quad \phi_i'(x_0) = \phi_i'(x_n) = 0.$$

The second type of basis function has derivative one at a boundary point and has value 0 at each node. These two basis functions are denoted by $\phi_{-1}(x)$ and $\phi_{n+1}(x)$ where

$$\phi_{-1}'(x_0) = 1$$

$$\phi_{-1}(x_i) = 0 \quad i = 0, 1, \dots, n$$

$$\phi_{-1}'(x_n) = 0$$

$$\phi'_{n+1}(x_0) = 0$$

$$\phi_{n+1}(x_i) = 0 \quad i = 0, 1, 2, \dots, n$$

$$\phi'_{n+1}(x_n) = 1$$

Therefore, we write

$$s_f(x) = \sum_{i=-1}^{n+1} f(x_i) \phi_i(x)$$

where $f(x_{-1}) = f'(x_0)$ and $f(x_{n+1}) = f'(x_n)$. A typical type one basis function, $\phi_k(x)$, is pictured below.

Figure 4. $\phi_k(x)$

The two basis functions which interpolate derivatives at the end points have a somewhat different appearance, and $\phi_{-1}(x)$ is shown below.

Figure 5. $\phi_{-1}(x)$

Remark: Unlike the Hermite case the spline basis functions are not local since each one is nonzero over all of I . In evaluating $s_f(x)$ at some point $\bar{x} \in I$, one would not use the formula

$$s_f(\bar{x}) = \sum_{i=-1}^{n+1} f(x_i) \phi_i(\bar{x}).$$

Instead, (assuming storage is no problem) one solves the tri-diagonal system (4) for $s'_f(x_j)$, $j = 1, 2, \dots, n-1$. Having these values the subinterval containing $\bar{x}$ is located and the following formula is used.

$$s_f(\bar{x}) = f(x_i) H_1(\bar{x}) + f(x_{i+1}) H_2(\bar{x}) + s'_f(x_i) H_3(\bar{x}) + s'_f(x_{i+1}) H_4(\bar{x})$$

for $\bar{x} \in [x_i, x_{i+1}]$.

The question is then asked, is there a "local basis" for $Sp^2(I, \pi)$? Such a basis must be identically zero outside a minimum number of subintervals. Over the first subinterval in which it is nonzero (say $[x_i, x_{i+1}]$) the basis spline has the property $b^{(k)}(x_i) = 0$ $k = 0, 1, 2$ but $b^{(3)}(x_i) \neq 0$. Similarly, in the last subinterval over which $b(x) \neq 0$, $[x_{i+l-1}, x_{i+l}]$, $b^{(k)}(x_{i+l}) = 0$, $k = 0, 1, 2$, $b^{(3)}(x_{i+l}) \neq 0$. It is easy to show that the smallest value of l , (i.e. the smallest number of subintervals over which a spline can be nonzero) is 4 for cubics. Such a basis function is called a B-spline and is shown below

Figure 6 B-spline

In order to get enough basis functions to span $Sp^2(I, \pi)$ the domain may be extended beyond the interval $I = [a, b]$ to include points x_{-1} , x_{-2} , x_{-3} ,

x_{n+1} , x_{n+2} and x_{n+3} for the purpose of constructing B-splines similar to the one pictured above. The spline function $s_f(x)$ is then represented in the form

$$s_f(x) = \sum_{i=1}^{n+3} \alpha_i b_i(x).$$

The α_i are calculated from the interpolation conditions. This results in a system of linear equations

$$f(x_j) = s_f(x_j) = \sum_{i=1}^{n+3} \alpha_i b_i(x_j) \quad j = 0, 1, 2, \dots, n$$

$$f'(x_0) = s'_f(x_0) = \sum_{i=1}^{n+3} \alpha_i b'_i(x_0)$$

$$f'(x_n) = s'_f(x_n) = \sum_{i=1}^{n+3} \alpha_i b'_i(x_n)$$

With proper ordering this system will be tri-diagonal and very stable. After calculating the coefficients α_i above, one evaluates $s_f^{(k)}(x)$ at a point $x = \bar{x} \in I$ by the formula

$$s_f^{(k)}(\bar{x}) = \sum_{i=1}^{n+3} \alpha_i b_i^{(k)}(\bar{x})$$

This method is very efficient since at most 4 of the $b_i^{(k)}(x)$ are nonzero at $x = \bar{x}$. In addition, one need only store the $n+3$ coefficients α_i . (Using the interpolation basis $2(n+1)$ words of storage are needed.)

Remark: A more complete discussion of B-splines and their application will be contained in a subsequent paper by the author and Dr. Robert Dickinson. This paper is particularly relevant in the bivariate case discussed in Section 4.

In addition to the interpolation scheme of Definition 4 one could construct many other algebraically well set interpolation schemes for cubic splines. Unfortunately, not all of these are analytically well set. For example, consider modifying (5) by replacing

$$s'(x_n) = f'(x_n)$$

with the condition

$$s'(c) = f'(c)$$

for some point $a < c < b$ in I . Thus, we have

$$s(x_i) = f(x_i) \quad i = 0, 1, \dots, n$$

$$s'(x_0) = f'(x_0)$$

$$s'(c) = f'(c)$$

For simplicity, let us assume $c = x_k$ for some $0 < k < n$. In the interval $[a, c]$ $s(x)$ is the usual spline interpolant of f . Outside of that interval (i.e. in $[c, b]$) we are extrapolating under the condition $s(x_i) = f(x_i)$, $i = k+1, \dots, n$.

This interpolation scheme is extremely unstable in $[c,b]$ and an example is included in the appendix to substantiate this observation. This illustration is included because such interpolation conditions arise when attempting to use spline interpolation in two-dimensional regions which are nonrectangular domains (see sec. 6). Furthermore, the instability is not obvious when using a non-interpolatory basis such as B-splines. The reason for the instability is discussed in the Appendix.

There are a number of generalizations suggested for spline functions. One obvious approach is to consider higher degree polynomials, and define

$$S_p^n(I,\pi) = H^n(I,\pi) \cap C^{2n-2}[I].$$

For $n=3$ this gives rise to quintic splines which are in $C^4[I]$. Quintic Hermite functions are in $C^2[I]$, leaving an "intermediate" space of $H^3(I,\pi) \cap C^3[I]$.

One could also consider spaces of even degree splines (of degree $2m$) defined by

$$S_{pe}^{2m} = P^{2m}(I,\pi) \cap C^{2m-1}[I].$$

A different approach to spline functions involves variational consideration. For example, there is Holladay's Theorem for "natural" cubic splines satisfying

$$s(x_i) = g(x_i) \quad i = 0, 1, 2, \dots, n$$

$$s''(x_0) = s''(x_n) = 0$$

Theorem 3: (Holladay) Of all functions $f(x)$ satisfying $f(x_i) = g(x_i)$, $i = 0, 1, \dots, n$ and $f''(x_0) = f''(x_n) = 0$, the cubic spline minimizes the integral

$$\int_{x_0}^{x_n} |f''(x)|^2 dx.$$

Generalizations of this result have led to the definition of "spline" spaces which are not piecewise polynomials (see [32]).

There are a number of important considerations involved in the practical application of splines to interpolation and curve-fitting problems. These considerations are discussed in detail in Section 4 and in the Appendix.

4. BIVARIATE FUNCTIONS

The problem of fitting a surface to specified data is conveniently separated into two cases. The first involves constructing a surface over a rectangular domain and the second over a nonrectangular domain. Most of our discussion will focus on the former.

Let R be a rectangle in the x - y plane with sides parallel to the coordinate axes. Then $R = \{(x,y) | a \leq x \leq b, c \leq y \leq d\}$ is the Cartesian product $I_1 \times I_2$ where $I_1 = \{x | a \leq x \leq b\}$ and $I_2 = \{y | c \leq y \leq d\}$. Let $\pi_1: a = x_0 < x_1 < \dots < x_m = b$ and $\pi_2: c = y_0 < y_1 < \dots < y_n = d$ be partitions of I_1 and I_2 respectively. Then $\pi = \pi_1 \times \pi_2$ partitions R into subrectangles with mesh points $\{(x_i, y_j) | 0 \leq i \leq m, 0 \leq j \leq n\}$.

For bivariate functions let us consider in detail the analogue to piecewise cubics, i.e. piecewise bicubic polynomials. There are two approaches to the bivariate problem. The first involves proceeding exactly as in the univariate case with piecewise bicubic polynomials and then imposing smoothness constraints on the derivatives. The second involves using tensor products of spaces of univariate functions. The latter is chosen because of the ease in which theoretical results can be derived.

The space $H^2(R, \pi)$ of bicubic Hermite polynomials is defined as the tensor product of $H^2(I_1, \pi_1)$ and $H^2(I_2, \pi_2)$, i.e.

$$H^2(R, \pi) = H^2(I_1, \pi_1) \otimes H^2(I_2, \pi_2)$$

In each rectangular element of (R, π) $u(x,y) \in H^2(R, \pi)$ has the form

$$\sum_{j=0}^3 \sum_{i=0}^3 \alpha_{ij} x^i y^j$$

A basis for $H^2(R, \pi)$ is simply the tensor product of basis functions of $H^2(I_1, \pi_1)$ and $H^2(I_2, \pi_2)$. There are four possibilities:

$$b_i(x)\hat{b}_j(y), c_i(x)\hat{b}_j(y), b_i(x)\hat{c}_j(y), c_i(x)\hat{c}_j(y).$$

The properties of these basis functions lead to the following definition of the bicubic Hermite interpolant.

Definition 6: Let f be an appropriate function with domain R and partition π . Then the unique bicubic Hermite polynomial $u_f(x, y)$ satisfying

$$\begin{aligned} u_f(x_i, y_j) &= f(x_i, y_j) \\ \frac{\partial u_f}{\partial x}(x_i, y_j) &= \frac{\partial f}{\partial x}(x_i, y_j) \\ (6) \quad \frac{\partial u_f}{\partial y}(x_i, y_j) &= \frac{\partial f}{\partial y}(x_i, y_j) \\ \frac{\partial^2 u_f}{\partial x \partial y}(x_i, y_j) &= \frac{\partial^2 f}{\partial x \partial y}(x_i, y_j) \end{aligned} \quad \begin{aligned} i &= 0, 1, \dots, m \\ j &= 0, 1, \dots, n \end{aligned}$$

is the bicubic Hermite interpolant of f .

Clearly $u_f(x, y)$ can be written in the form

$$u_f(x, y) = \sum_{j=0}^n \sum_{i=0}^m \left\{ f(x_i, y_j) b_i(x) \hat{b}_j(y) + \frac{\partial f}{\partial x}(x_i, y_j) c_i(x) \hat{b}_j(y) + \frac{\partial f}{\partial y}(x_i, y_j) \cdot b_i(x) \hat{c}_j(y) + \frac{\partial^2 f}{\partial x \partial y}(x_i, y_j) c_i(x) \hat{c}_j(y) \right\}$$

In order to evaluate $u_f(x, y)$ at some point $(\bar{x}, \bar{y}) \in R$ one first locates the subrectangle $[x_1, x_{1+1}] \times [y_j, y_{j+1}]$ containing $(\bar{x}, \bar{y})$, and, in that subrectangle, $u_f(x, y)$ has the form

$$\begin{aligned}
 u_f(x,y) = & f(x_i,y_j)H_1(x)G_1(y) + f(x_{i+1},y_j)H_2(x)G_1(y) + f(x_i,y_{j+1})H_1(x)G_2(y) + \\
 & f(x_{i+1},y_{j+1})H_2(x)G_2(y) + \frac{\partial f}{\partial x}(x_i,y_j)H_3(x)G_1(y) + \frac{\partial f}{\partial x}(x_{i+1},y_j)H_4(x)G_1(y) + \\
 & \frac{\partial f}{\partial x}(x_i,y_{j+1})H_3(x)G_2(y) + \frac{\partial f}{\partial x}(x_{i+1},y_{j+1})H_4(x)G_2(y) + \frac{\partial f}{\partial y}(x_i,y_j)H_1(x)G_3(y) + \\
 & \frac{\partial f}{\partial y}(x_{i+1},y_j)H_2(x)G_3(y) + \frac{\partial f}{\partial y}(x_i,y_{j+1})H_1(x)G_4(y) + \frac{\partial f}{\partial y}(x_{i+1},y_{j+1})H_2(x)G_4(y) + \\
 & \frac{\partial^2 f}{\partial x \partial y}(x_i,y_j)H_3(x)G_3(y) + \frac{\partial^2 f}{\partial x \partial y}(x_{i+1},y_j)H_4(x)G_3(y) + \frac{\partial^2 f}{\partial x \partial y}(x_i,y_{j+1})H_3(x)G_4(y) + \\
 & \frac{\partial^2 f}{\partial x \partial y}(x_{i+1},y_{j+1})H_4(x)G_4(y)
 \end{aligned}$$

where the $H_i(x)$ are given in (3) and the $G_i(y)$ are the analogues in y .

The next result shows that u_f is a "good" approximation to f if f is smooth. Specifically,

Theorem 4: [10] Let $f \in C^4[R]$, then, for r and $s = 0,1$,

$$\left\| \frac{\partial^{r+s}}{\partial x^r \partial y^s} (u_f - f) \right\| \leq C_{r,s} \left\| f^{(4)} \right\| h^{4-r-s}$$

Remark: It follows from the tensor product formulation that $u_f \in C^{(1,1)}[R]$, i.e. that

$$u_f, \frac{\partial u_f}{\partial x}, \frac{\partial u_f}{\partial y} \text{ and } \frac{\partial^2 u_f}{\partial x \partial y}$$

are continuous in R . In general, the other second partial derivatives are discontinuous across mesh lines. Explicit error bounds for bicubic Hermite interpolation are contained in a separate report (UCRL 77516).

Bicubic Hermite interpolation is seldom used in practice because of the large number of interpolation parameters, the frequent unavailability of the required derivatives and the storage requirements. Instead, a subspace of $H^2(R, \pi)$ is often used, that of bicubic splines. As in the univariate case, we could proceed by considering the space of bicubic splines to be all those Hermite functions with two continuous derivatives, that is $Sp^2(R, \pi) = H^2(R, \pi) \cap C^2[R]$. However, it is more convenient to use the equivalent tensor product definition given below.

Definition 7: Let (I_1, π_1) and (I_2, π_2) be the partitioned intervals $a=x_0 < x_1 < \dots < x_m=b$; $c=y_0 < y_1 < \dots < y_n=d$ respectively. For $R = I_1 \times I_2$ and $\pi = \pi_1 \times \pi_2$ the space of bicubic splines over (R, π) is defined by

$$Sp^2(R, \pi) = Sp^2(I_1, \pi_1) \otimes Sp^2(I_2, \pi_2)$$

It follows from the tensor product definition that

$$\begin{aligned} \dim Sp^2(R, \pi) &= \dim Sp^2(I_1, \pi_1) \cdot \dim Sp^2(I_2, \pi_2) \\ &= (m+3)(n+3) \end{aligned}$$

A cardinal basis for $Sp^2(R, \pi)$ consists of products of cardinal basis functions from the univariate spaces (see sec. 3)

$$\left\{ \phi_i(x) \psi_j(y) \mid -1 \leq i \leq m+1, \quad -1 \leq j \leq n+1 \right\}.$$

The nature of these basis functions gives rise to the bicubic spline interpolation scheme.

Definition 8: Let f be an appropriate function with domain R . Then the bicubic spline interpolant of f is the unique bicubic spline satisfying

$$s_f(x_i, y_j) = f(x_i, y_j) \quad , \quad 0 \leq i \leq m; \quad 0 \leq j \leq n$$

$$\frac{\partial s_f}{\partial x}(x_i, y_j) = \frac{\partial f}{\partial x}(x_i, y_j) \quad , \quad i = 0, m; \quad 0 \leq j \leq n$$

$$\frac{\partial s_f}{\partial y}(x_i, y_j) = \frac{\partial f}{\partial y}(x_i, y_j) \quad , \quad 0 \leq i \leq m; \quad j = 0, n$$

$$\frac{\partial^2 s_f}{\partial x \partial y}(x_i, y_j) = \frac{\partial^2 f}{\partial x \partial y}(x_i, y_j) \quad , \quad i = 0, m; \quad j = 0, n$$

Thus, the bicubic spline interpolant of f interpolates to f at each mesh point, the normal derivative of f at each boundary mesh point and the first mixed partial derivative of f at the four corners of R .

The interpolation scheme of Definition 8 is analytically well set as noted in the theorem below (see [14]).

Theorem 5: Let $f \in C^4[R]$ and let s_f be the bicubic spline interpolant of f of Definition 8. Then

$$(8) \quad \left\| \frac{\partial^{r+s}(s_f - f)}{\partial x^r \partial y^s} \right\|_{\infty} \leq c_{rs} \left\| f^{(4)} \right\| h^{4-r-s}$$

where the c_{rs} are given in [14].

Remark: The notation in Theorem 5 has the following interpretation:

$f \in C^4[R]$ means that $\frac{\partial^{r+s} f}{\partial x^r \partial y^s}$ is continuous in R where $r+s = 4$; $\|f^{(4)}\| =$

$$\max \left\{ \left\| \frac{\partial^4 f}{\partial x^4} \right\|, \left\| \frac{\partial^4 f}{\partial x^3 \partial y} \right\|, \left\| \frac{\partial^4 f}{\partial x^2 \partial y^2} \right\|, \left\| \frac{\partial^4 f}{\partial x \partial y^3} \right\|, \left\| \frac{\partial^4 f}{\partial y^4} \right\| \right\}; \text{ and}$$

$$h = \max_{i,j} \{x_{i+1} - x_i, y_{j+1} - y_j\}.$$

In order to evaluate s_f at some point (x, y) based on the above discussion, one would calculate the Hermite coordinates of s_f and proceed as though $s_f \in H^2(R, \pi)$ as discussed earlier.

Remark: From the interpolation conditions $(m+3)(n+3)$ of the Hermite coordinates have been specified from Definition 8. The remaining coordinates are uniquely determined by these conditions and that $s_f \in C^2[R]$. To compute $\frac{\partial s_f}{\partial x}(x_i, y_j)$, $1 \leq i \leq m-1$, $0 \leq j \leq n$, we use the fact that for $y=y_j$, $s_f(x, y_j)$ is a cubic spline in x . The value of this spline at each mesh point and its first derivative at the end points is given in (5). The derivatives, $\frac{\partial s_f}{\partial x}(x_i, y_j)$ are computed by solving the tri-diagonal system (4). By solving $(n+1)$ such systems, $\frac{\partial s_f}{\partial x}$ is computed at each node. The same procedure is used to calculate $\frac{\partial s_f}{\partial y}$ at each node, using the cubic spline $s_f(x_i, y)$. The first mixed partial derivative is first calculated on two opposite boundaries, using the fact that $\frac{\partial s_f}{\partial x}(x_i, y)$ $i=0, m$ (or $\frac{\partial s_f}{\partial y}(x, y_j)$, $j=0, n$) are cubic splines. This procedure is repeated along each horizontal (vertical) mesh line for $\frac{\partial s_f}{\partial y}(x, y_j)$ to calculate $\frac{\partial^2 s_f}{\partial x \partial y}$ at each node. (See [17] for details.)

This procedure suffers from the handicap that four words of storage are required for each mesh point. Using a different basis — B-splines — one can reduce the storage to $(m+3)(n+3)$ words. In this case $s_f(x, y)$ is written

$$(9) \quad s_f(x, y) = \sum_{j=-1}^{n+1} \sum_{i=-1}^{m+1} \alpha_{ij} b_i(x) \hat{b}_j(y)$$

where $b_i(x)$ and $\hat{b}_j(y)$ are univariate B-splines. The coefficients α_{ij} are computed from the interpolation conditions (7)

$$s_f(x_k, y_\ell) = f(x_k, y_\ell) = \sum_{j=-1}^{n+1} \sum_{i=-1}^{m+1} \alpha_{ij} b_i(x_k) \hat{b}_j(y_\ell)$$

This leads to a single system of linear equations of order $(m+3)(n+3)$. Using a factorization technique due to Bob Dickinson solving this matrix problem can be reduced to solving several smaller systems. In addition, by modifying the B-splines slightly, the smaller systems can be made to be

tri-diagonal and diagonally dominant. Details of this factorization will be contained in another report (UCRL-77894).

5. BLENDING FUNCTIONS

In the late 1960's Dr. William Gordon of General Motors Research Laboratories developed a new approximation technique called blending functions. They are not piecewise polynomials, but are closely related. In most practical applications the method is modified to a two-level approximation which does result in a piecewise polynomial function. Blending functions are used to get extremely accurate interpolation results.

With Hermite and spline approximation a finite number of values of f and various derivatives of f were interpolated at a discrete set of points in the domain (R, π) . Using blending functions one is able to match arbitrary functions along mesh lines.

To illustrate, let f be an appropriate function with domain (R, π) . The functions $f(x_i, y)$, $i = 0, 1, \dots, m$ and $f(x, y_j)$, $j = 0, 1, \dots, n$ are univariate functions. We shall construct a function $v(x, y)$ such that $v(x_i, y) = f(x_i, y)$, $y_0 \leq y \leq y_n$ and $v(x, y_j) = f(x, y_j)$, $x_0 \leq x \leq x_m$. (Note that the functions $f(x_i, y)$ and $f(x, y_j)$ need not be piecewise polynomials. For our purposes the only requirement other than smoothness is that

$$\left. f(x_i, y) \right|_{y=y_j} = \left. f(x, y_j) \right|_{x=x_i} .)$$

The function $v(x, y)$ will be the cubic spline blended interpolant of f . Other types of blending are possible, but this represents a good first choice for many problems and it serves to illustrate the method.

Let us now consider three functions:

$$v_1(x, y) = \sum_{i=-1}^{m+1} f(x_i, y) \phi_i(x)$$

where $\{\phi_i(x) \mid -1 \leq i \leq m+1\}$ is the set of cardinal basis functions discussed in sec. 3. Similarly, let

$$v_2(x,y) = \sum_{j=-1}^{n+1} f(x,y_j) \psi_j(y)$$

where $\{\psi_j(y) \mid -1 \leq j \leq n+1\}$ are cardinal splines in the y-direction.

Finally, let

$$v_3(x,y) = \sum_{i=-1}^m \sum_{j=-1}^n f(x_i,y_j) \phi_i(x) \psi_j(y).$$

Thus, $v_3(x,y)$ is the usual bicubic spline interpolant of f . The cubic spline blended interpolant of f , $v_f(x,y)$, is defined by

$$v_f(x,y) = v_1(x,y) + v_2(x,y) - v_3(x,y)$$

To verify the interpolation properties of v_f consider

$$\begin{aligned} v_f(x,y) \Big|_{x=x_k} &= \sum_{i=-1}^{m+1} f(x_i,y) \phi_i(x) \Big|_{x=x_k} + \sum_{j=-1}^{n+1} f(x,y_j) \psi_j(y) \Big|_{x=x_k} \\ &\quad - \sum_{i=-1}^{m+1} \sum_{j=-1}^{n+1} f(x_i,y_j) \phi_i(x) \psi_j(y) \Big|_{x=x_k} \\ &= f(x_k,y) + \sum_{j=-1}^{n+1} f(x_k,y_j) \phi_j(y) - \sum_{j=-1}^{n+1} f(x_k,y_j) \psi_j(y) \\ &= f(x_k,y) \end{aligned}$$

since $\phi_i(x_k) = \delta_{ik}$. Similarly, it can be shown that $v(x, y_j) = f(x, y_j)$,
 $j = 0, 1, \dots, n$

$$\frac{\partial v_f}{\partial x}(x_i, y) = \frac{\partial f}{\partial x}(x_i, y), \quad i = 0, m$$

$$\frac{\partial v_f}{\partial y}(x, y_j) = \frac{\partial f}{\partial y}(x, y_j), \quad j = 0, n$$

$$\frac{\partial^2 v_f}{\partial x \partial y}(x_i, y_j) = \frac{\partial^2 f}{\partial x \partial y}(x_i, y_j), \quad i = 0, m; j = 0, n.$$

Because we are interpolating f along lines and not just at discrete points it would seem reasonable that v_f would be a better approximation to f than either the spline or the Hermite. This is in fact the case and explicit error bounds for v_f are given in the following theorem.

Theorem 6: [14, Theorem 2] Let $f \in C^{(4,4)}[R]$, then

$$\|v_f - f\|_{\infty} \leq \left(\frac{5}{384}\right)^2 \|f^{(4,4)}\| h^4 k^4$$

where $h = \max_{0 \leq i \leq m-1} (x_{i+1} - x_i)$ and $k = \max_{0 \leq j \leq n-1} (y_{j+1} - y_j)$.

Therefore, v_f is an eighth order approximation to f . (Similar results are available for $\frac{\partial^{r+s} v_f}{\partial x^r \partial y^s}$ as an approximation to $\frac{\partial^{r+s} f}{\partial x^r \partial y^s}$.)

It was noted earlier that in practice spline blending results in a two-level approximation. This is necessary because $f(x_i, y)$ and $f(x, y_j)$ are not usually known explicitly. These are univariate functions defined along mesh lines and a fine mesh on these lines can be constructed to get extremely accurate approximations to $f(x_i, y)$ and $f(x, y_j)$.

The resulting partition is shown in the figure below.

If $\underline{h} \approx h^2$ and $\underline{k} \approx k^2$, then one still gets an approximation of order $h^4 k^4$ using cubic splines to approximate $f(x_i, y)$ and $f(x, y_j)$. Note that this is equivalent to what one obtains by imposing the fine mesh over all of R .

Remark: Using this two level approximation v_f is a bicubic spline over the fine partition but does not interpolate to f on the interior of the coarse mesh.

Gordon's results have been extended to non-rectangular domains by means of domain transformations (see [21]).

6. NONRECTANGULAR DOMAINS

The use of piecewise polynomial functions in nonrectangular domains creates many difficult problems because of what happens along the boundary, and the fact that tensor product formulations are no longer valid. However, some meaningful results are available and I would like to discuss them rather briefly.

Perhaps the simplest nonrectangular domain is a rectangular polygon, P . If P is partitioned into a union of rectangles (assuming mesh lines terminate only at the boundary and not at an interior mesh line), we can define $H^2(P, \pi)$ as the space of piecewise bicubic polynomials in $C^1[P]$. The local nature of bicubic Hermite polynomials makes them suitable for this case. The nonlocal nature of bicubic splines makes them unsuitable even for very simple nonrectangular domains. First, one must determine the dimension of the spline space $Sp(P, \pi)$, (which could be defined as $H^2(P, \pi) \cap C^2[P]$). Then one would be faced with other problems such as constructing analytically well set interpolation schemes. What complicates the problem is that in two-dimensional applications B-splines are frequently used and the problems are much more subtle and less easily recognized.

Polygonal domains frequently arise in attempts to approximate curved boundaries. Bicubic Hermite interpolation can be extended into polygons which can be partitioned into rectangles on the interior and right triangles along the boundary. Three types of extensions are given in a paper by Hall [25]. Unfortunately, not all polygons can be subdivided in this manner, so the application is somewhat limited.

For general polygons one could partition the region into triangular elements and use a lower order interpolation scheme such as piecewise linear functions. Until the last ten years or so this was a popular method of partitioning for finite element approximations in the area of structural mechanics.

More recently several methods have been developed for domains with curved boundaries. In this case one has essentially two alternatives.

- i. Transform the domain into a rectangle, solve the problem over the rectangle and map the solution to the original domain; or
- ii. construct new types of functions which can accommodate curved sides.

For most applications the first alternative has been chosen. In finite element approximations the isoparmetric element developed by Ergatoudis, Irons and Zienkiewicz [18] has become the standard. Gordon and Hall [21] have extended their results about blending functions to domains with curved boundaries using this approach. These methods seem to work well provided the transformation is sufficiently well behaved. Frequently it is important to have good interaction with the computer in constructing the partition.

The second alternative, constructing new types of functions, is the approach I took several years ago. The functions I came up with are really quite simple, and, for lack of a better name, are called quasi-polynomials.

Consider a domain D partitioned by a sequence π of horizontal and vertical lines as shown below.

Figure 7

Let f be an appropriate function with domain D , and construct the bicubic quasi-polynomial interpolant, q_f , as follows.

In each rectangular element q_f is the usual bicubic Hermite interpolant of f . For the boundary elements let us consider a typical element E :

Figure 8

For simplicity we shall assume the lower left corner is the origin and $(\bar{x}, \bar{y})$ is an arbitrary point in E . Also, it is necessary to assume that the boundary curve can be represented by a function of the form

$$x_b = g(y), \quad 0 \leq y \leq k.$$

Thus, $x_b = g(\bar{y})$ is the length of the dashed line in the figure for that particular value of y .

The quasi-polynomial $q_f(x, y)$ is defined as follows:

$$(10) \quad q_f(x,y) = q_f(0,y) H_1(x,g(y)) + q_f(x_b,y) H_2(x,g(y)) \\ + \frac{\partial q_f}{\partial x}(0,y) H_3(x,g(y)) + \frac{\partial q_f}{\partial x}(x_b,y) H_4(x,g(y))$$

Let us examine each term above in more detail. The $H_i(x,g(y))$ are the cubic Hermite basis functions defined in (3) where Δx_i is replaced by x_b , the length of the interval indicated in Figure 8. By setting $q_f(x_b,y) = f(x_b,y)$ and $\frac{\partial q_f}{\partial x}(x_b,y) = \frac{\partial f}{\partial x}(x_b,y)$ we discover q_f interpolates to f and its normal derivative on the boundary. Along the left edge of the element $q_f(0,y)$ and $\frac{\partial q_f}{\partial x}(0,y)$ are simply the cubic Hermite interpolants of $f(0,y)$ and $\frac{\partial f}{\partial x}(0,y)$. This assures us that $q_f \in C^1[D]$. It is a simple matter to show that this interpolation scheme is analytically well set and that q_f is a fourth order approximation to $f \in C^4[D]$.

More details of this scheme will be included in a subsequent paper. This method looks particularly applicable to finite element approximations, however bilinear quasi polynomials would probably represent a better choice to begin evaluating the technique.

A brief discussion of some numerical results and comments about applying piecewise polynomials to practical problems is contained in the Appendix.

APPENDIX

The purpose of the appendix is to consider some practical methods for implementing some of the ideas developed in the main body of the report. We begin by discussing curve fitting where only data is specified at the mesh points (no derivatives).

Given values $f(x_i)$, $i = 0, 1, \dots, n$ the problem is to construct a curve passing through these $(n+1)$ points. (For our purposes we shall assume a piecewise cubic is desired.) Our first choice might be a cubic spline, however, there are $n+3$ conditions necessary to construct a spline and only $n+1$ conditions are available (derivatives at the end points are missing).

There are several alternatives available. One could specify that the second derivative of the spline is zero at each end point (this is called a "natural spline") or that the third derivative is zero at each end point. An alternate approach would be to use a finite difference approximation to the first derivative at the end points. A four point (third order) approximation would be consistent with the error bounds in Theorem 2.

Remark: The latter approach has been the most accurate in a series of tests conducted this summer. A discussion of these tests and the results will be contained in a subsequent paper with Bob Dickinson (UCRL-77893).

Another approach would be to use Hermite functions with an approximation to the derivative at each node. Two types of approximations were considered, a simple three point difference and a method due to Akima [3]. Neither of these turned out to be as good as the spline in our evaluations.

A frequent concern in practical problems is the effect of "noise" in the data. That is, $f(x_i)$ is really $f(x_i) + \epsilon_i$ where, hopefully, ϵ_i is small. Spline interpolation is very stable and such errors do not propagate and grow. The cubic spline which interpolates to $f(x_i) + \epsilon_i$ will differ from that which interpolates to $f(x_i)$ only by the order of ϵ_i — which is as good as one could hope for. This result is readily obtained by considering the matrix problem in (4) and the Hermite basis functions. Errors in the derivatives of order n_j at the end points result in errors of order hn_j in function values, and die out exponentially away from the boundary.

If there is considerable error in the data, one may want to fit a spline in the least squares sense and not a spline of interpolation. This technique has proven quite successful.

Another common problem is using splines to fit curves which are either discontinuous or have discontinuous derivatives. Difficulties obviously will occur in intervals near the discontinuity where a "wobble effect" is noticed, although the "wiggles" die out quite rapidly. If the points of discontinuity are known, one could use a piecewise spline to maintain a high level of accuracy and avoid the above difficulty. An attempt to remove the "wiggles" in a different way was made by D.G. Schweikert using "Splines in Tension." A rigorous analysis of this technique has not yet been made.

The problem of spline extrapolation was referred to in Sec. 3. I would like to consider the following example to show the disastrous results which can occur.

Example 1: Let $f(x) = x^6$, $0 \leq x \leq 2$. The usual spline interpolant, $s_f(x)$ satisfies

$$s_f(x_i) = f(x_i) = x_i^6, \quad i = 0, 1, \dots, n$$

$$s'_f(0) = f'(0) = 0$$

$$s'_f(2) = f'(2) = 6 \cdot 2^5 = 192$$

Let $s(x)$ be the spline that satisfies

$$s(x_i) = f(x_i), \quad i = 0, 1, \dots, n$$

$$s'(0) = f'(0) = 0$$

$$s'(1) = f'(1) = 6$$

Both interpolation schemes are algebraically well set. Writing both splines in terms of the Cardinal basis we obtain

$$s_f(x) = f'(0) \phi_{-1}(x) + \sum_{i=0}^n f(x_i) \phi_i(x) + f'(2) \phi_{n+1}(x)$$

and
$$s(x) = f'(0) \phi_{-1}(x) + \sum_{i=0}^n f(x_i) \phi_i(x) + s'(2) \phi_{n+1}(x)$$

where $s'(2)$ is uniquely determined (but, at this point, unknown). Subtracting,

we obtain

$$s_f(x) - s(x) = \left(f'(2) - s'(2) \right) \phi_{n+1}(x)$$

It is known (Theorem 2) that $s_f(x)$ is a fourth order approximation to x^6 . Therefore, as the mesh is refined, $s(x)$ will converge to x^6 if and only if $s'(2) \rightarrow 192$. The following table demonstrates what happens.

h	$s'(2)$
1/2	195
1/4	195
1/8	228.75
1/16	86,637.188
1/32	$\sim 7 \times 10^{12}$

Therefore, as $h \rightarrow 0$, $s'(2) \rightarrow \infty$. This phenomena is due to the fact that the derivative of $\phi_{n+1}(x)$ approaches 0 exponentially as x moves away from 2. Therefore, a "small error" at $x = 1$ compared with $s'_f(1)$ results in a "large error" at 2.

More than anything else this is a testimony for the stability of spline interpolation, because despite the large error near $x = 2$, extremely accurate results are obtained for $x \leq 1$ where the approximation does converge.

In the bivariate case let us assume we are trying to fit a surface to a set of $(m+1)(n+1)$ points $f(x_i, y_j)$ arranged over a rectangular grid (x_i, y_j) , $0 \leq i \leq m$, $0 \leq j \leq n$. If we attempt to construct a spline surface through these points, we need to supply $2(m+1) + 2(n+1) + 4$ more conditions, i.e. the missing normal derivatives on the boundary and cross derivatives at the

corners. These derivatives can be approximated using the four point difference scheme mentioned above. Lacking additional information about the surface, this technique probably represents the best choice available. Some inaccuracies can be anticipated near the boundary since exact derivatives are not being supplied. Numerical results using this approximation will be contained in a subsequent report with Bob Dickinson.

Blending functions are rather new and I would like to include a numerical example to illustrate the technique. Consider the rectangular domain $[0,2] \times [0,2]$ with an initial partition of $H = 1/4$. Along each mesh line construct a partition with mesh size $h = 1/16 = (1/4)^2$ as shown in Figure 9.

Figure 9

Let $f(x,y) = x^6 y^6$ and construct cubic spline approximations to

$$f(x_i, y) = (x_i)^6 y^6 \text{ for } x_i = 0, 1/4, 1/2, \dots, 7/4, 2$$

$$\text{and } f(x, y_j) = x^6 (y_j)^6 \text{ for } y_j = 0, 1/4, \dots, 7/4, 2$$

$$\frac{\partial f}{\partial x}(0, y) = 0, \frac{\partial f}{\partial x}(2, y), \frac{\partial f}{\partial y}(x, 0) \text{ and } \frac{\partial f}{\partial y}(x, 2),$$

using a mesh size of $h = 1/16$. Substitute these cubic splines for $f(x_i, y)$ and $f(x, y_j)$ in the formula

$$v(x,y) = \sum_{i=-1}^9 f(x_i, y) \phi_i(x) + \sum_{j=-1}^9 f(x, y_j) \psi_j(y) \\ \sum_{j=-1}^9 \sum_{i=-1}^9 f(x_i, y_j) \phi_i(x) \psi_j(y)$$

This gives the cubic spline blended approximation to $f(x,y)$. Numerical results on spline blending will be presented in another paper with Bob Dickinson.

REFERENCES

1. Ahlberg, J.H., Nilson, E.N., "Convergence Properties of the Spline Fit," SIAM J. Applied Math., 11 (1963), 95-104.
2. Ahlberg, J.H., Nilson, E.N., and Walsh, J.L., The Theory of Splines and Their Applications, Academic Press, New York (1967).
3. Akima, H., "A New Method of Interpolation and Smooth Curve Fitting Based on Local Procedures," J. A.C.M., 17 (1970), 589-602.
4. Birkhoff, G., "Piecewise Bicubic Interpolation and Approximation in Polygons," Approximations with Special Emphasis on Spline Functions, ed. I.J. Schoenberg, Academic Press, New York, (1969), 185-221.
5. Birkhoff, G. and de Boor, C.R., "Piecewise Polynomial Interpolation and Approximation," Approximation of Functions, ed. H.L. Garabedian, Elsevier, New York, (1965), 164-190.
6. Birkhoff, G. and de Boor, C.R., "Error Bounds for Spline Interpolation," J. Math. Mech., 13 (1964), 827-835.
7. Birkhoff, G., Cavendish, J.C. and Gordon, W.J., "Multivariate Approximation by Locally Blended Univariate Interpolants," Proc. Nat. Academy Sci., 71 (1974), 3423-3425.
8. Birkhoff, G. and Garabedian, H.L., "Smooth Surface Interpolation," J. Math. and Phys., 39 (1960), 353-368.
9. Birkhoff, G. and Priver, A., "Hermite Interpolation Errors for Derivatives," J. Math. and Phys., 46 (1967), 440-447.
10. Birkhoff, G., Schultz, M.H. and Varga, R.S., "Piecewise Hermite Interpolation in One and Two Variables with Applications to Partial Differential Equations," Num. Math., 11 (1968), 232-256.
11. Birkhoff, G. and Gordon, W.J., "The Draftsman's and Related Equations," J. Approx. Theory, 1 (1968), 199-208.
12. Carlson, R.E. and Hall, C.A., "On Piecewise Polynomial Approximation of Smooth Functions in Rectangular Polygons," J. Approx. Theory, 4 (1971), 37-53.
13. Carlson, R.E. and Hall, C.A., "Bicubic Spline Interpolation in Rectangular Polygons," J. Approx. Theory, 6 (1972), 366-377.
14. Carlson, R.E. and Hall, C.A., "Error Bounds for Bicubic Spline Interpolation," J. Approx. Theory, 7 (1973), 41-47.
15. Cavendish, J.C., "Local Mesh Refinement Using Rectangular Blended Finite Elements," to appear, J. Comp. Phys.

16. Conte, S.D. and de Boor, C.R., Elementary Numerical Analysis, McGraw-Hill Book Co., New York, (1972).
17. de Boor, C.R., "Bicubic Spline Interpolation," J. Math. and Phys. 41 (1962), 212-218.
18. Ergatoudis, I., Irous, B.M. and Zienkiewicz, O.C., "Curved, Isoparametric, "Quadrilateral" Elements for Finite Element Analysis," Int. J. Solids Structures, 4 (1968), 31-42.
19. de Boor, C.R., "Best Approximation Properties of Spline Functions of Odd Degree," J. Math. Mech., 12 (1963), 747-750.
20. Gordon, W.J., "Spline-Blended Interpolation Through Curve Networks," J. Math. Mech., 18 (1969), 931-952.
21. Gordon, W.J. and Hall, C.A., "Transfinite Element Methods: Blending-Function Interpolation over Arbitrary Curved Element Domains," Num. Math. 21 (1973), 109-129.
22. Gordon, W.J., "Blending Function Methods of Bivariate and Multivariate Interpolation and Approximation," SIAM J. Numer. Anal., 8 (1971), 158-177.
23. Greville, T.N.E. (ed.), Theory and Applications of Spline Functions, Academic Press, New York, (1969).
24. Hall, C.A., "On Error Bounds for Spline Interpolation," J. Approx. Theory, 1 (1968), 209-218.
25. Hall, C.A., "Bicubic Interpolation over Triangles," J. Math. Mech., 19 (1969), 1-11.
26. Hall, C.A., "Error Bounds for Periodic Quintic Splines," Com. A.C.M., 12 (1969), 450-452.
27. Hall, C.A., "Uniform Convergence of Cubic Spline Interpolants," J. Approx. Theory, 7 (1973), 71-75.
- Kämmerer, W.J., Reddien, G.W. and Varga, R.S., "Quadratic Interpolatory Splines," Num. Math., 22 (1974), 241-259.
- Kämmerer, W.J., Reddien, G.W., "Local Convergence of Smooth Cubic Spline Interpolation," SIAM J. Numer. Anal., 9 (1972), 687-694.
- Marsden, M.J., "Cubic Spline Interpolation of Continuous Functions," J. Approx. Theory, 10 (1974), 103-111.
- Meir, A. and Sharma, A., "On Uniform Approximation by Cubic Splines," J. Approx. Theory, 2 (1969), 270-274.
- Schoenberg, I.J., "Contributions to the Problem of Approximation of Equidistant Data by Analytic Functions," Quart. Applied Math., 4 (1946), 45-99.

32. Schoenberg, I.J. (ed), Approximations with Special Emphasis on Spline Functions, Academic Press, New York, 1969.
33. Sharma, A. and Meir, A., "Degree of Approximation of Spline Interpolation," J. Math. Mech. 15 (1966), 759-767.
34. Walsh, J.S., Ahlberg, J.H. and Nilson, E.N., "Best Approximation Properties of the Spline Fit," J. Math. Mech., 11 (1962), 225-234.

REC/jat

DISTRIBUTION

Internal

R. Abbott, L-307
M. Axelrod, L-531
R. Bell, L-60
A. Blum, L-156
J. Boyd, L-388
T. Buckholtz, L-71
G. Carlson, L-24
J. Chang, L-71
R. Dickinson, L-310
J. Dent, L-46
S. Devoto, L-388
G. Dorough, L-82
W. Feng, L-90
S. Fernbach, L-61
F. Fritsch, L-310
P. Gluckman, L-61
H. Graboske, L-504
P. Gray, L-307
A. Greenbaum, L-547
A. Hall, L-310
R. Hausman, L-310
N. Hetherington, L-421
A. Hindmarsh, L-310

M. Hummell, L-307
D. Jones, L-531
J. Killeen, L-318
R. Lee, L-142
D. Logan, L-24
N. Madsen, L-310
J. Matthews, L-307
F. McMurphy, L-402
G. Myers, L-152
W. Nervik, L-61
B. Olness, L-504
D. Pehrson, L-60
R. Pexton, L-310
A. Poggio, L-156
G. Rodrigue, L-310
R. Sani, L-142
N. Smiriga, L-310
T. Sullivan, L-142
R. von Holdt, L-60
C. Wilgus, L-307
T. Wilson, L-117
A. Winslow, L-71
NMG Files - 10 Copies
TID - 15 Copies

External

U.S. Energy Research & Develop. Admin.
Technical Information Center
P.O. Box 62
Oak Ridge, Tennessee 37830 (27)

Library
Lawrence Berkeley Laboratory
University of California
Bldg. 50, Room 134
Berkeley, California 94720

Elon Close
Mathematics & Computing Group
Lawrence Berkeley Laboratory
University of California
Berkeley, California 94720

J. Young
Mathematics & Computing Group
Lawrence Berkeley Laboratory
University of California
Berkeley, California 94720

Melvin R. Scott
Applied Mathematics Division 2642
Sandia Laboratories
Sandia Base
Albuquerque, New Mexico 87115

Larry F. Shampine
Numerical Analysis Division 5121
Sandia Laboratories
Sandia Base
Albuquerque, New Mexico 87115

Robert J. Thompson
Applied Mathematics Division 5122
Sandia Laboratories
Sandia Base
Albuquerque, New Mexico 87115

Burton Wendroff
Division T-7
Los Alamos Scientific Laboratory
P.O. Box 1663
Los Alamos, New Mexico 87545

Bill Puzbee, Kirby Fong
Division C-3, MS 265
Los Alamos Scientific Laboratory
P.O. Box 1663
Los Alamos, New Mexico 87545

R. Huddleston, J. Rogers
and J. Lathrop
Numerical Applications
Division 8441
Sandia Laboratories
P.O. Box 969
Livermore, CA 94550

W. J. Cody, Jr.,
Wayne R. Cowell, and
J.C.T. Pool
Applied Mathematics Div.
Argonne National Laboratory
9700 South Cass Avenue
Argonne, Illinois 60439

Lee W. Lucas
Code 6075
Naval Weapons Center
China Lake, CA 93555

Software Services Section
Attn: Gene Anderson
Code 5137
Naval Weapons Center
China Lake, CA 93555

C.W. Gear
Computer Science Dept.
University of Illinois
Urbana, Illinois 61801

Richard R. Sincovec
Computer Science Dept.
Kansas State University
Manhattan, Kansas 66506

Henry C. Thacher, Jr.
Dept. of Computer Science
University of Kentucky
Lexington, Kentucky 40506

Cleve B. Moler
Dept. of Mathematics
University of New Mexico
Albuquerque, New Mexico 87106

George D. Byrne
Mathematics Department
University of Pittsburgh
Pittsburgh, Pennsylvania 15213

External (Continued)

Joseph F. Traub
Computer Science Department
Carnegie-Mellon University
Pittsburgh, Pennsylvania 15213

Gene H. Golub
Computer Science Department
Stanford University
Stanford, California 94305

Brian Ford
NAG Central Office
13 Banbury Road
Oxford
ENGLAND

R.F. Hausman
Lockheed Missile & Space Co.
Dept. 6213, Bldg. 104
P.O. Box 504
Sunnyvale, California 94088

Myron Ginsberg
Dept. of Computer Science
and Operations Res.
Southern Methodist University
Institute of Technology
Dallas, Texas 75275

Robert Voigt
ICASE
M/S 132C
NASA-Langley Research Center
Hampton, Virginia 23665

Dr. Douglass B. Morris
Dept. of Mathematics
Talladega College
Talladega, AL 35160

J.W. Spellmann
Department of Mathematics
Pan American University
Edinburg, Texas 78539

Dr. Ralph Carlson (20 Copies)
Dept. of Mathematics
Grove City College
Grove City, Pennsylvania 16127

NOTICE

"This report was prepared as an account of work sponsored by the United States Government. Neither the United States nor the United States Energy Research & Development Administration, nor any of their employees, nor any of their contractors, subcontractors, or their employees, makes any warranty, express or implied, or assumes any legal liability or responsibility for the accuracy, completeness or usefulness of any information, apparatus, product or process disclosed, or represents that its use would not infringe privately-owned rights."

Printed in the United States of America
Available from

National Technical Information Service
U.S. Department of Commerce
5285 Port Royal Road
Springfield, Virginia 22151

Price: Printed Copy \$ ___*; Microfiche \$2.25

<u>* Pages</u>	<u>NTIS Selling Price</u>
1-50	\$4.00
51-150	\$5.45
151-325	\$7.60
326-500	\$10.60
501-1000	\$13.60

Technical Information Department
LAWRENCE LIVERMORE LABORATORY
University of California | Livermore, California | 94550