

# Surfaces Generated by Moving Least Squares Methods

By P. Lancaster and K. Salkauskas

**Abstract.** An analysis of moving least squares (m.l.s.) methods for smoothing and interpolating scattered data is presented. In particular, theorems are proved concerning the smoothness of interpolants and the description of m.l.s. processes as projection methods. Some properties of compositions of the m.l.s. projector, with projectors associated with finite-element schemes, are also considered. The analysis is accompanied by examples of univariate and bivariate problems.

**1. Introduction.** While the theory and practice of interpolation and approximation of functions of a single variable on the basis of a finite amount of information is well developed, the same is not true for functions of several variables. When the information consists of function values at the points of a rectangular grid, tensor product and blended interpolants based on univariate schemes can be employed. For irregularly distributed function-value data, the situation is much worse. Finite-element techniques are of value, although, in order to produce  $C^1$  interpolants, more than just function-value information is required. As well, if the distribution of data is irregular, only triangular elements seem feasible. Their use has become more attractive in view of the recent development of fast and efficient triangulation algorithms. Nevertheless, the additional nodal information consisting of derivative data must somehow be concocted.

Least squares approximation by polynomials is in widespread use and is, of course, not expected to produce an interpolant. However, as has been shown by McLain [7], Gordon and Wixom [4], Barnhill [1], and more recently Lancaster [5], least squares approximation ideas can be applied to generate interpolants by introducing the notion of *moving least squares approximation* together with appropriate singularities in the weights used in such approximations. This method includes the metric interpolation technique of Shepard [11]. To date the theory of such moving least squares approximants and interpolants is meagre. Here we develop the method in a way which allows it to be seen in the context of inner-product spaces. This leads to theorems concerning the differentiability class of such approximants as well as some other properties of the projection defined by this line of attack. These go beyond the results reported by Lancaster [5]. We do not give a convergence theory; attention is focussed on geometrical properties of the interpolants.

It is shown that the method determines a projector, say  $G$ , from the space of continuous functions on a (closed) domain  $\bar{D}$  onto a space of  $m$  times differentiable

---

Received October 31, 1979; revised July 3, 1980.

1980 *Mathematics Subject Classification.* Primary 41A05, 65D05.

© 1981 American Mathematical Society  
0025-5718/81/0000-0111/\$05.50

functions on  $D$ , with  $m$  determined by the basis and other details of the method. An interpolating projector  $P$  related to  $G$  is also introduced, which can result in reduced computational effort. Particular versions of the latter process have previously been applied in such programs as the SURFACE II package of the Kansas Geological Survey. After the preparation of this paper, the recent work of Franke and Nielson [3] came to our attention. In [3] too, the projector  $P$  is derived as an alternative to  $G$ , and some practical guidance to actual computation is given.

In order to isolate certain features of the analysis, we begin by considering the moving least squares approximation method without attempting to interpolate. It is assumed that a function  $f: \bar{D} \rightarrow \mathbf{R}$  is to be approximated and that its values  $f_i$  at the points  $z_i \in \bar{D}$ ,  $i = 1, \dots, N$ , are given. The domain  $\bar{D}$  is assumed to be the closure of a simply connected subset  $D$  of  $\mathbf{R}^n$ . For ease of presentation, we shall assume  $n = 2$ ; it is then easy to see that the results apply for any number of variables. The development of an approximation  $Gf$  to  $f$ , on the basis of the available information about  $f$ , is the subject of the next section.

Thereafter, singularities are introduced into the weight functions employed in order to ensure the interpolation conditions  $Gf(z_i) = f_i$ ,  $i = 1, 2, \dots, N$ , and, in Section 6, the variant of the interpolation process mentioned above is examined. In Section 9, we discuss composite methods obtained by generating nodal data for finite-element methods from local least squares interpolants.

Since the methods introduced have not been systematically investigated previously, the analysis is quite liberally illustrated with examples of both univariate and bivariate problems. The graphs are intended to convey qualitative information only and are of limited accuracy.

**2. Noninterpolating Moving Least Squares Methods.** Given a function  $f$ , as in the previous section, we determine a global approximating function  $Gf$  by first forming, at each point  $\hat{z} \in \bar{D}$ , a local approximant  $L_{\hat{z}}f$ , defined in terms of some basis  $\{b^{(i)}\}_{i=1}^n$ ,  $n \leq N$ , and a local  $L^2$ -norm. It turns out that  $Gb^{(i)} = b^{(i)}$  (see Section 5), so it is natural to include the constant function in the basis. If this is the only function in the basis,  $Gf$  is a moving average and, in the interpolating formulation of Section 4,  $Gf$  becomes the Shepard interpolant [11]. We assume that the basis has the properties

- (2.1) (i)  $b^{(1)} \equiv 1$ ,  
(ii)  $b^{(i)} \in C^m(D)$ ,  $i = 1, \dots, n$ ,  
(iii)  $\{b^{(i)}\}_{i=1}^n$  is independent over some set of  $n$  of the given  $N$  points in  $\bar{D}$ .

Now, for each  $\hat{z} \in \bar{D}$ , find coefficients  $a_i(\hat{z})$ ,  $i = 1, \dots, n$ , so that the function

$$(2.2) \quad L_{\hat{z}}f := \sum_{i=1}^n a_i(\hat{z})b^{(i)}$$

is in a certain least squares sense the best approximation to  $f$ . Then define, for any  $\hat{z} \in \bar{D}$ ,

$$(2.3) \quad Gf(\hat{z}) = L_{\hat{z}}f(\hat{z}) = \sum_{i=1}^n a_i(\hat{z})b^{(i)}(\hat{z}).$$

We will now be more specific about the "least squares sense" and show that  $G$  is well defined by (2.3). For the approximation of  $f$  by  $L_{\hat{z}}f$ , we employ a weighted,

discrete,  $L^2$ -norm induced by a  $\hat{z}$ -dependent inner product for  $\tilde{C}^0(\bar{D})$ , where the tilde indicates that any two functions that do not differ on the set of given points  $\{z_i\}_{i=1}^N$  shall be deemed equivalent. With the notation

$$\mathbf{u} = (u(z_1), u(z_2), \dots, u(z_N))^T, \quad u \in \tilde{C}^0(\bar{D}),$$

the inner product at  $\hat{z}$  is given by

$$(2.4) \quad (u, v)_{\hat{z}} := \mathbf{u}^T W(\hat{z}) \mathbf{v}, \quad \forall u, v \in \tilde{C}^0(\bar{D}).$$

We take for  $W(\hat{z})$  an  $N \times N$  diagonal matrix,  $W(\hat{z}) = \text{diag}(w^{(1)}(\hat{z}), \dots, w^{(N)}(\hat{z}))$ , defined for each  $\hat{z} \in \bar{D}$ , with positive diagonal elements (but see also Section 7). The corresponding  $\hat{z}$ -norm is of course

$$(2.5) \quad \|u\|_{\hat{z}} = (u, u)_{\hat{z}}^{1/2}.$$

Now, for any fixed  $\hat{z} \in \bar{D}$ , the unique best approximation  $L_{\hat{z}}f$  (cf. (2.2)) to  $f$  from  $\text{span}\{b^{(i)}\}_{i=1}^n$  in the  $\hat{z}$ -norm satisfies

$$(f - L_{\hat{z}}f, b^{(i)})_{\hat{z}} = 0, \quad i = 1, \dots, n,$$

which implies that the coefficients  $a_i(\hat{z})$  satisfy the normal equations

$$\sum_{i=1}^n a_j(\hat{z})(b^{(i)}, b^{(j)})_{\hat{z}} = (f, b^{(j)})_{\hat{z}}, \quad j = 1, \dots, n.$$

With the convention that  $a_k$  denotes the  $k$ th component of the vector  $\mathbf{a}$ , these normal equations may be written in the form

$$(2.6) \quad BW(\hat{z})B^T \mathbf{a} = BW(\hat{z})\mathbf{f},$$

where  $B$  is an  $n \times N$  matrix whose  $j$ th row is  $(\mathbf{b}^{(j)})^T = (b^{(j)}(z_1), \dots, b^{(j)}(z_N))$  and  $\mathbf{f} = (f(z_1), \dots, f(z_N))^T$ . The independence of the vectors  $\mathbf{b}^{(i)}$ , which follows from independence of  $\{b^{(i)}\}_{i=1}^n$ , together with the positive-definiteness of  $W(\hat{z})$ , implies that  $BW(\hat{z})B^T$  is positive-definite. Hence the coefficients  $a_i(\hat{z})$  in (2.2) are uniquely determined and the definition of  $G$  by (2.3) is justified.

This definition implies that, unless  $W(\hat{z})$  is in fact a constant matrix, a new vector  $\mathbf{a}(\hat{z})$  must be determined for each  $\hat{z} \in \bar{D}$ . The  $\hat{z}$  may be replaced by  $z$  at this stage and  $Gf$  is said to be the approximation to  $f$  generated by a *moving least squares method*. If  $W(z)$  is constant, then  $Gf$  is a classical, nonmoving, weighted least squares approximation or regression function for  $f$ .

As to the smoothness of  $Gf$ , it is elementary to prove that if the  $b^{(i)} \in C^m(D)$ ,  $i = 1, \dots, n$ , and the (diagonal) elements of  $W$ ,  $w^{(j)} \in C^l(D)$ ,  $j = 1, \dots, N$ , then  $Gf \in C^k(D)$ ,  $k = \min\{l, m\}$ . This follows immediately from the invertibility of  $BWB^T$  at every point of  $\bar{D}$  and the representation (2.3).

For future reference, we look more closely at the case  $n = 1$ . Here,  $L_{\hat{z}}f$  consists of the projection of  $f$ , through the  $\hat{z}$ -inner product, on  $b^{(1)}$ . Therefore

$$(2.7) \quad L_{\hat{z}}f = (f, b^{(1)})_{\hat{z}} \frac{b^{(1)}}{(b^{(1)}, b^{(1)})_{\hat{z}}} = \frac{\sum_{i=1}^N w^{(i)}(\hat{z})f_i}{\sum_{i=1}^N w^{(i)}(\hat{z})},$$

and so (cf. (2.1) and (2.4))

$$(2.8) \quad L_{\hat{z}}f = \sum_{i=1}^N f_i v^{(i)}(\hat{z}),$$

where we have put

$$(2.9) \quad v^{(i)}(\hat{z}) = \frac{w^{(i)}(\hat{z})}{\sum_{j=1}^N w^{(j)}(\hat{z})}, \quad i = 1, \dots, N.$$

It is clear that the  $v^{(i)}(\hat{z})$  satisfy  $0 < v^{(i)}(\hat{z}) < 1$  and  $\sum v^{(i)}(\hat{z}) = 1$ . Hence, for each  $\hat{z}$ ,  $L_{\hat{z}}f(\hat{z}) = Gf(\hat{z})$  is just a weighted average of the given values of  $f$ . In this case, the moving least squares method is a moving weighted average method. Therefore, when  $n = 1$ ,

$$\min\{f_1, \dots, f_N\} \leq Gf(z) \leq \max\{f_1, \dots, f_N\}.$$

In anticipation of the results of Section 4, where a choice of singular functions for the  $w^{(i)}(z)$  will result in  $Gf$  interpolating the data, we define

$$(2.10) \quad Sf := \sum_{i=1}^N f_i v^{(i)},$$

with the  $v^{(i)}$  given by (2.9). In Section 4 this will be the Shepard interpolant [11].

**3. An Alternate Representation of the Local Approximation.** In this section, we develop a representation of  $L_{\hat{z}}f$  which will be important in the proof of Theorem 4.1 concerning the smoothness and representation of interpolating surfaces. The idea is just to represent  $L_{\hat{z}}f$  in a basis for  $\text{span}\{b^{(i)}\}_{i=1}^n$  consisting of a multiple of  $b^{(1)}$  and a basis for the  $\hat{z}$ -orthogonal complement of  $b^{(1)}$  in  $\text{span}\{b^{(i)}\}_{i=1}^n$ .

First normalize  $b^1$  and write

$$(3.1) \quad \beta^{(1)}(\hat{z}; \cdot) = \frac{b^{(1)}}{\|b^{(1)}\|_{\hat{z}}} = \frac{1}{\left[ \sum_{j=1}^N w^{(j)}(\hat{z}) \right]^{1/2}}.$$

Now, for  $i = 2, \dots, n$ , generate functions orthogonal to  $\beta^{(1)}(\hat{z}; \cdot)$  by subtracting from  $b^{(i)}(\cdot)$  its  $\hat{z}$ -projection on  $\beta^{(1)}(\hat{z}; \cdot)$ . Thus, we define

$$\begin{aligned} u^{(i)}(\hat{z}; \cdot) &= b^{(i)}(\cdot) - (b^{(i)}, \beta^{(1)}(\hat{z}; \cdot))_{\hat{z}} \beta^{(1)}(\hat{z}; \cdot) \\ &= b^{(i)}(\cdot) - \sum_{j=1}^N v^{(j)}(\hat{z}) b^{(i)}(z_j), \end{aligned}$$

using (3.1) and then (2.9). Or, using (2.10), we can write

$$(3.2) \quad u^{(i)}(\hat{z}; \cdot) = b^{(i)}(\cdot) - S b^{(i)}(\hat{z}), \quad i = 2, 3, \dots, n,$$

and observe that, for a fixed  $\hat{z}$ ,  $u^{(i)}(\hat{z}; \cdot)$  differs from  $b^{(i)}(\cdot)$  by a constant function.

It is easily seen that the  $n - 1$  functions of (3.2) determine linearly independent vectors  $\mathbf{u}^{(i)}(\hat{z}), i = 2, \dots, n$ , where

$$\mathbf{u}^{(i)}(\hat{z})^T = [u^{(i)}(\hat{z}, z_1), \dots, u^{(i)}(\hat{z}, z_N)].$$

Hence, the  $(n - 1) \times N$  matrix  $U(\hat{z})$ , having  $\mathbf{u}^{(i)}(z)^T$  as its  $(i - 1)$ st row, is of full rank.

Now, consider the least squares procedure of Section 2 applied using the new basis  $\beta_1, u_2, \dots, u_n$ . Because of the orthogonality conditions used in the construction, a reduced set of  $n - 1$  normal equations is obtained, and we have the

representation

$$L_{\hat{z}}f(\cdot) = (f, \beta^{(1)}(\hat{z}; \cdot))_{\hat{z}}\beta^{(1)}(\hat{z}; \cdot) + \sum_{i=2}^n (f, u^{(i)}(\hat{z}; \cdot))_{\hat{z}}u^{(i)}(\hat{z}; \cdot).$$

Using (3.1) followed by (2.9) and (2.10), the first term on the right is seen to be  $Sf(\hat{z})$ , and, writing  $\alpha_{i-1}(\hat{z}) = (f, u^{(i)}(\hat{z}; \cdot))_{\hat{z}}$ ,  $\alpha(\hat{z}) = (\alpha_1(\hat{z}), \dots, \alpha_{n-1}(\hat{z}))$ , we arrive at

LEMMA 3.1. *The local approximation  $L_{\hat{z}}f$  of  $f$  has the representation*

$$(3.3) \quad L_{\hat{z}}f(\cdot) = Sf(\hat{z}) + \sum_{i=2}^n \alpha_{i-1}(\hat{z})u^{(i)}(\hat{z}; \cdot),$$

where  $u^{(i)}(\hat{z}; \cdot)$  is given by (3.2), and  $\alpha(\hat{z})$  is the unique solution of

$$(3.4) \quad U(\hat{z})W(\hat{z})U^T(\hat{z})\alpha = U(\hat{z})W(\hat{z})\mathbf{f}.$$

On replacing  $\hat{z}$  by  $z$ , and evaluating at  $z$ , (3.3) gives, for the global approximant,

$$(3.5) \quad Gf(z) = Sf(z) + \sum_{i=2}^n \alpha_{i-1}(z)g^{(i)}(z),$$

where, for  $i = 2, \dots, n$ ,

$$(3.6) \quad g^{(i)}(z) = u^{(i)}(z, z) = b^{(i)}(z) - Sb^{(i)}(z).$$

**4. Interpolating Moving Least Squares Methods.** Evidently, the moving least squares approximant  $Gf$ , discussed in the previous sections, need not interpolate the data. However, an idea apparently due to Shepard [11] and discussed in some detail by Gordon and Wixom [4], Lancaster [6], and Barnhill [1] can be employed to ensure that  $Gf$  interpolates at some or all data points. The principle involved is to make  $w^{(k)}$  become infinite at the data point  $z_k$  if  $Gf$  is to interpolate there. For the purposes of this and the next three sections, it will be assumed that  $Gf$  is to interpolate at every data point. Since the values of  $Gf$  are computed from those of  $L_{\hat{z}}f$ , as in (2.3), it is obvious that the local approximant must satisfy the interpolation conditions

$$(4.1) \quad L_{z_k}f(z_k) = f_k, \quad k = 1, \dots, N.$$

The case  $n = 1$  is of particular importance. In that case, (3.5) shows that  $Gf = Sf$ , which is called the *Shepard interpolant*. First we introduce singularities of an analytically convenient kind into the weight functions and indicate some useful properties of the resulting normalized weight functions defined by (2.9). A wider class of weight functions is considered in Section 7.

LEMMA 4.1. *Let  $w^{(i)}(z) = |z - z_i|^{-\alpha}$ , with  $\alpha$  an even positive integer,  $z \notin \{z_i\}_{i=1}^N$ . Then  $v^{(i)} \in C^\infty(D)$  and hence  $Sf \in C^\infty(D)$ . Furthermore,  $Sf$  satisfies  $Sf(z_i) = f_i$ ,  $i = 1, \dots, N$ , and the  $v^{(i)}$  satisfy*

- (i)  $v^{(i)}(z_j) = \delta_{ij}$ ,  $i, j = 1, \dots, N$ ,
- (ii)  $0 \leq v^{(j)}(z) < 1$ ,  $\forall z$ , and  $v^{(j)}(z) = 0$  if and only if  $z = z_i$ ,  $i \neq j$ ,
- (iii)  $\sum_{j=1}^N v^{(j)}(z) = 1$ ,  $\forall z$ ,
- (iv)  $v^{(j)}(z) \rightarrow 1/N$  as  $|z| \rightarrow \infty$ .

*Proof.* The hypotheses concerning the  $w^{(i)}$  and the definitions (2.9) of the  $v^{(i)}$  show the latter to be rational functions (in  $x$  and  $y$ ) with nonvanishing denominator. Hence  $v^{(i)} \in C^\infty(D)$ . The remaining properties of the  $v^{(i)}$  then follow easily. Since  $Sf = \sum_{i=1}^N f_i v^{(i)}$ ,  $Sf \in C^\infty(D)$ , and property (i) above shows that  $Sf(z_i) = f_i$ ,  $i = 1, \dots, N$ .  $\square$

**COROLLARY.**  $\text{grad } v^{(i)}(z_j) = 0$ ,  $i, j = 1, \dots, N$ .

This follows from the fact that  $v^{(i)} \in C^\infty(D)$  and properties (i) and (ii) in the lemma.  $\square$

Since  $Sf = \sum f_i v^{(i)}$ , the property of the Corollary is transmitted to  $Sf$ , i.e.  $\text{grad } Sf(z_i) = 0$ ,  $i = 1, \dots, N$ . This is the ‘‘flat spot’’ phenomenon associated with the Shepard interpolant, which suggests that  $Sf$  has limited application to surface fitting in general. The flat spots generally disappear, however, when  $n > 1$ . Having seen the effect on  $Sf$  of the introduction of the singularities in the weights, we consider the effect on the remaining terms of  $L_z f$  and  $Gf$  (cf. (3.3), (3.5)). It is clear that if  $\hat{z} \neq z_i$ , then the calculation of  $L_z f$  and  $Gf(\hat{z})$  will not involve a singular weight in  $W(\hat{z})$ . Thus both  $L_z f$  and  $Gf$  are in  $C^m(D \setminus \{z_i\}_{i=1}^N)$ , since  $b^{(i)} \in C^m(D)$ .

When  $n > 1$ , the presence of singularities in the weight functions leads to coefficients of the matrix  $BWB^T$  of (2.6), which become infinite as  $\hat{z} \rightarrow z_k$ , and hence to some difficulty in analysis of the smoothness of the surface defined by (2.3). We now show how representation of the surface in the form (3.5) yields a set of normal equations having no such singularities in the coefficients. In terms of the notation of Section 3, we prove first

**LEMMA 4.2.** *The functions  $u^{(i)}$  and  $g^{(i)}$ , defined by (3.2) and (3.6), respectively, belong to  $C^m(D)$  whenever the  $w^{(i)}$  are defined as in Lemma 4.1 and the  $b^{(i)} \in C^m(D)$ . Furthermore, for every  $h \in \tilde{C}^0(D)$  (cf. comments following (2.3)), the inner products  $(u^{(i)}(\hat{z}; \cdot), h)_z$  are  $C^\infty$  functions of  $\hat{z}$  on  $D$ .*

*Proof.* It follows immediately from (3.2) and Lemma 4.1 that  $u^{(i)}(\hat{z}; \cdot) \in C^m(D)$  for every  $\hat{z} \in D$ . Similarly,  $g^{(i)} \in C^m(D)$ . As for  $(u^{(i)}(\hat{z}; \cdot), h)_z$ , the only term of the inner product that could cause a singularity is given, in a sufficiently small deleted neighborhood of a data point  $z_k$ , by

$$w^{(k)}(\hat{z})u^{(i)}(\hat{z}; z_k)h(z_k) = w^{(k)}(\hat{z}) \left\{ b^{(i)}(z_k) - \frac{\sum_{j=1}^N w^{(j)}(\hat{z})b^{(i)}(z_j)}{\sum_{j=1}^N w^{(j)}(\hat{z})} \right\} h(z_k)$$

$$= h(z_k)v^{(k)}(\hat{z}) \sum_{\substack{j=1 \\ j \neq k}}^N w^{(j)}(\hat{z}) [b^{(i)}(z_k) - b^{(i)}(z_j)].$$

Since  $v^{(k)} \in C^\infty(D)$ , and the sum is void in  $w^{(k)}(\hat{z})$ , and the  $w^{(i)} \in C^\infty(D)$ , the inner product can be extended to a  $C^\infty$  function (of  $\hat{z}$ ) on the whole neighborhood of  $z_k$ .  $\square$

The local approximation  $L_z f$  is now determined by the orthogonality conditions

(4.2) (a)  $(f - L_z f, \beta^{(1)}(\hat{z}; \cdot))_z = 0,$

(4.2) (b)  $(f - L_z f, u^{(j)}(z; \cdot))_z = 0, \quad j = 2, \dots, n,$

as in Section 3. However, care must be taken in view of the singular weight functions in the inner product. With

$$L_{\hat{z}}f = \alpha_0 \beta^{(1)}(\hat{z}; \cdot) + \sum_{i=2}^n \alpha_{i-1} u^{(i)}(\hat{z}; \cdot),$$

and with a temporary and obvious simplification of notation, the orthogonality conditions above yield

$$(4.3) \quad \begin{aligned} (a) \quad & \alpha_0 (\beta^{(1)}, \beta^{(1)})_{\hat{z}} + \sum_{i=2}^n \alpha_{i-1} (u^{(i)}, \beta^{(1)})_{\hat{z}} = (f, \beta^{(1)})_{\hat{z}}, \\ (b) \quad & \alpha_0 (\beta^{(1)}, u^{(j)})_{\hat{z}} + \sum_{i=2}^n \alpha_{i-1} (u^{(i)}, u^{(j)})_{\hat{z}} = (f, u^{(j)})_{\hat{z}}, \quad j = 2, \dots, n. \end{aligned}$$

With the exception of  $(f, \beta^{(1)})_{\hat{z}}$  which is unbounded as  $\hat{z} \rightarrow z_k$ , a data point, all other inner products are well defined on all of  $\bar{D}$  by virtue of Lemma 4.2 and the definition of  $\beta^{(1)}$ . But, in a sufficiently small deleted neighborhood of  $z_k$ , the system is equivalent, by elimination of  $\alpha_0$  from (4.3)(b), to

$$(4.4) \quad \begin{aligned} (a) \quad & \alpha_0 = (f, \beta^{(1)})_{\hat{z}}, \\ (b) \quad & \sum_{i=2}^n \alpha_{i-1} (u^{(i)}, u^{(j)})_{\hat{z}} = (f - \alpha_0 \beta^{(1)}, u^{(j)})_{\hat{z}} \\ & = (f - Sf(\hat{z}), u^{(j)})_{\hat{z}}, \quad j = 2, \dots, n. \end{aligned}$$

By Lemmas 4.1 and 4.2, the coefficient matrix and right-hand side of (4.4)(b) consists of  $C^\infty$  elements. The coefficient matrix is positive-definite throughout  $\bar{D}$ , and hence a unique  $C^\infty$  solution  $(\alpha_1, \dots, \alpha_{n-1}) = \alpha^T$  exists for all  $z \in \bar{D}$ . Although  $\alpha_0$  is unbounded as  $\hat{z} = z_k$ ,  $\alpha_0 \beta^{(1)}(\hat{z}; \cdot) = Sf(\hat{z}) \in C^\infty(D)$ . In terms of the notation of (3.5), and using Lemma 4.2, we thus have

**THEOREM 4.1.** *Let  $b^{(j)} \in C^m(D)$ ,  $j = 1, \dots, n$ , and let  $w^{(i)}$  be defined as in Lemma 4.1. Then both the local approximant  $L_{\hat{z}}f$  and global interpolant  $Gf$  belong to  $C^m(D)$  and have the forms*

$$(4.5) \quad L_{\hat{z}}f(z) = Sf(\hat{z}) + \sum_{i=2}^n \alpha_{i-1}(\hat{z}) u^{(i)}(\hat{z}; z),$$

$$(4.6) \quad Gf(z) = Sf(z) + \sum_{i=2}^n \alpha_{i-1}(z) g^{(i)}(z),$$

where the  $u^{(i)}$  and  $g^{(i)}$  are defined by (3.2) and (3.6), respectively, and the coefficient vector  $\alpha$  satisfies

$$(4.7) \quad U(z)W(z)U^T(z)\alpha = U(z)W(z)(\mathbf{f} - Sf(z)\mathbf{b}^{(1)}).$$

The discussion leading to this theorem shows that in working with the basis  $b^{(1)}, u^{(2)}(\hat{z}; \cdot), \dots, u^{(n)}(\hat{z}; \cdot)$  for the determination of  $L_{\hat{z}}f$ , the burden of coping with the singular weights falls solely on  $b^{(1)}$  and the associated Shepard interpolant whose form can be seen in (2.8) and in the first term of the sum in (3.2). Equation (3.1) shows that  $\beta^{(1)}(\hat{z}; \cdot) \rightarrow 0$  as  $\hat{z} \rightarrow z_k$ , a data point. At the same time  $(f, \beta^{(1)}(\hat{z}; \cdot))_{\hat{z}} \rightarrow \infty$ . However, the product defining  $Sf(\hat{z})$  remains finite.

Theorem 4.1 generalizes an earlier smoothness result of Lancaster [5], provides a more natural line of argument, and does not require the "very well-posed" notion

introduced in that paper. Furthermore, as we shall see in Section 7, the conclusions of the theorem continue to hold under much weaker hypotheses on the nature of the weight function.

We remark that, at the data points  $z = z_k, k = 1, 2, \dots, N$ , the system (4.7) can also be obtained by forming a weighted least squares approximation with an interpolation constraint and then eliminating the Lagrange multiplier.

Following Lemma 4.1 we referred to the “flat spot” property of  $Sf$ . This can be considered in conjunction with the fact that  $Sf$  is produced by making a least squares fit by a constant at every point  $z$  of  $\bar{D}$ . This constant is  $L_z f$  with  $n = 1$ . Thus  $L_z f$  is tangent to  $Sf$  at the data points. A similar tangency property can be demonstrated when  $n > 1$ .

**THEOREM 4.2.**  $\text{grad } L_{z_k} f(z_k) = \text{grad } G(z_k)$ .

The proof of this result requires a little care, but is quite straightforward and is therefore omitted.

**5. Interpolating Moving Least Squares as a Projection Method.** From the results of the previous section, it is clear that  $G$  is a linear operator from  $\tilde{C}^0(\bar{D})$  into  $C^m(D)$ . Consequently,  $Gf$  has a representation in terms of cardinal functions  $E^{(k)}, k = 1, \dots, N$ , having the properties

$$E^{(k)}(z_j) = \delta_{kj}, \quad j, k = 1, \dots, N,$$

and

$$Gf = \sum_{k=1}^N f_k E^{(k)}.$$

Any  $E^{(k)}$  can be computed by applying  $G$  to a function  $e^{(k)}$  defined on  $\bar{D}$  such that  $e^{(k)}(z_j) = \delta_{kj}$  for  $j, k = 1, \dots, N$ .

The fact that  $Gf$  interpolates  $f$  and uses only the values  $f_1, \dots, f_N$  shows that  $G^2 f = Gf$ . Thus  $G$  is a projector from  $\tilde{C}^0(\bar{D})$  onto  $M$ , a linear space of dimension  $N$ , i.e.  $\text{Im } G = M$  (where  $\text{Im}$  denotes the image, or range). It is also easy to see that  $M \supset \text{span}\{b^{(i)}\}_{i=1}^n$ . For, with  $z \neq z_k$ , we may use the normal equations (2.6) to determine the coefficient vector  $\mathbf{a}$ . Putting  $\mathbf{f} = \mathbf{b}^{(k)}$ , we have  $\mathbf{f} = B_{\cdot,k}^T$ , the  $k$ th column of  $B^T$ . Then (2.6) has the unique solution  $\mathbf{a} = \mathbf{e}^{(k)}$ , so that  $G\mathbf{b}^{(k)} = \mathbf{b}^{(k)}$  on  $\bar{D} \setminus \{z_i\}_{i=1}^N$ . By continuity, this can be extended to all of  $\bar{D}$ . The results are summarized in

**THEOREM 5.1.** *The interpolation scheme described in Theorem 4.1 determines a map  $G: \tilde{C}^0(\bar{D}) \rightarrow C^m(D)$  which is a projector onto the  $N$ -dimensional space  $M = \text{span}\{E^{(k)}\}_{k=1}^N \subset C^m(D)$ . Furthermore,  $\text{span}\{b^{(i)}\}_{i=1}^n$  is an  $n$ -dimensional subspace of  $M$ .*

*Example 1.* Consider the interpolating moving least squares (IMLS) method applied with one independent variable and four points of interpolation symmetrically placed on a parabolic arc. The basis functions chosen are  $b^{(i)}(x) = x^{i-1}, i = 1, 2, \dots, n$ . In Figure 5.1(a) and (b) interpolants are shown with  $n = 1, 2$ , and 3 and inverse second and fourth power weight functions, respectively ( $\alpha = 2, 4$  in Lemma 4.1). These graphs illustrate the necessity of including the full quadratic



basis if “dimpling” is to be avoided at local maxima and minima of a curve when there is only sparse data near the stationary value. The data and fitted curves are symmetric with respect to the line  $x = 0$ .

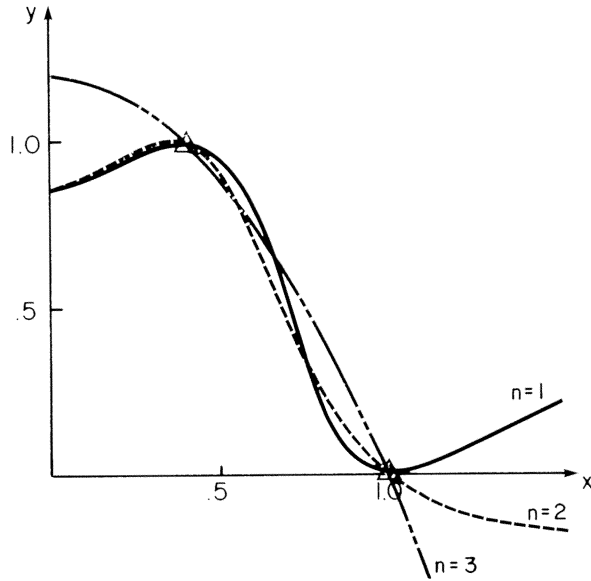


FIGURE 5.1

The “dimpling” phenomenon at local maxima.

(a) Inverse 2nd power weight function

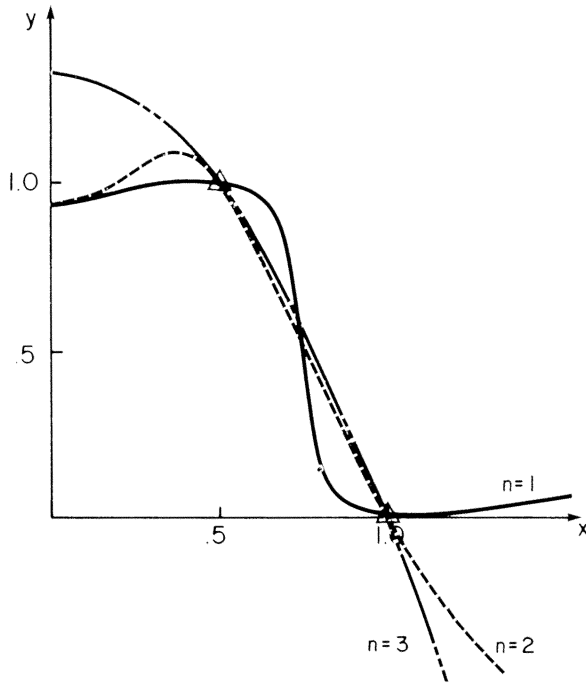


FIGURE 5.1

The “dimpling” phenomenon at local maxima

(b) Inverse 4th power weight function

*Example 2.* The IMLS method (with one independent variable) is applied to six pieces of data from a step function as indicated in Figure 5.2(a) and (b). The choice of weight functions and basis functions  $b^{(i)}$  is as in Example 1. Cubic spline interpolants are also shown for comparison. The data and fitted curves are symmetric with respect to the origin of coordinates.

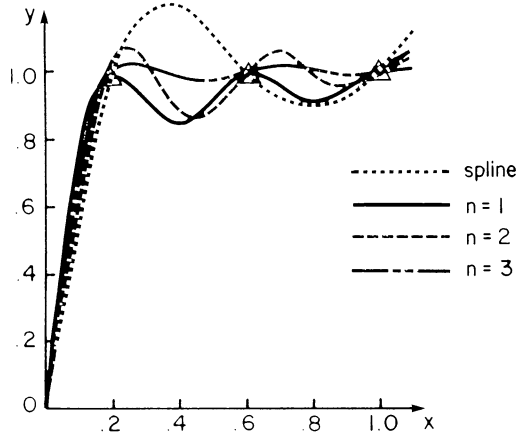


FIGURE 5.2  
(a) *Inverse second power weights*

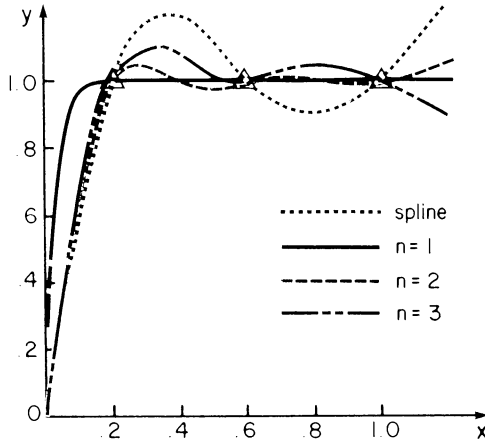


FIGURE 5.2  
(b) *Inverse fourth power weights*

*Example 3.* This is again a case of IMLS interpolants with one independent variable. The data is from eleven uniformly spaced points in  $[-1, 1]$  at which  $e^x$  is evaluated. The basis functions are as in Example 1 above and an inverse-square weight function is used. We draw the error curves,  $e^x - (\text{interpolant})$ , in Figure 5.3.

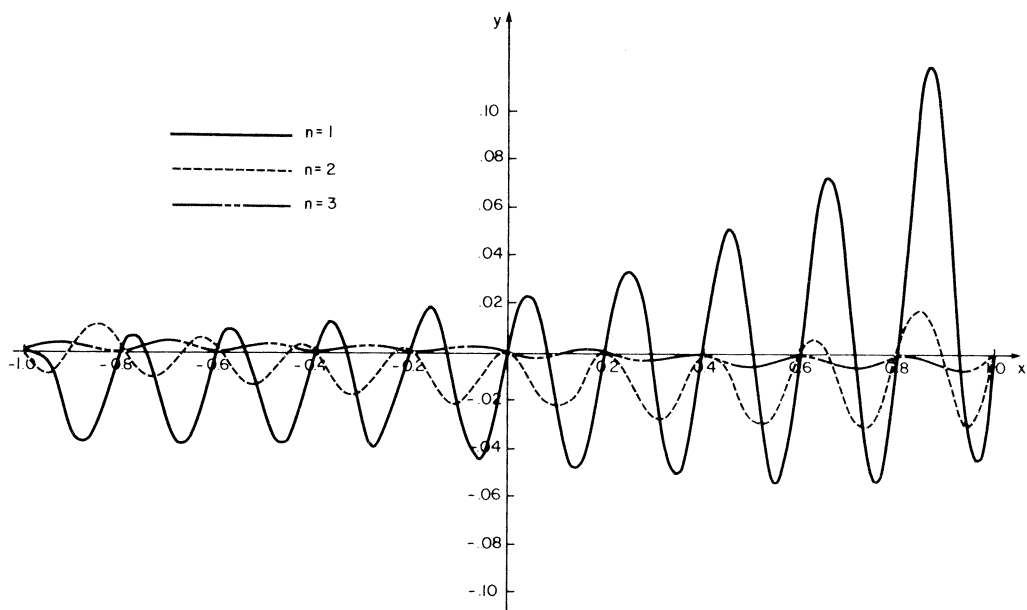


FIGURE 5.3

*The error in interpolants to the exponential function*

The relatively large errors in the case  $n = 1$  (the Shepard interpolant), and their oscillatory nature, are to be expected. They are a direct consequence of the “flat-spot” phenomenon.

*Example 4.* Consider an IMLS interpolant with two independent variables and  $n = 6$ , the basis functions  $b^{(i)}$  being  $1, x, y, x^2, xy, y^2$ . In this case a mathematically defined function made up of planes and a “mountain”, illustrated in Figure 5.4, is defined on a rectangular domain. Data points are generated by the random selection of 150 points in the rectangle (Figure 5.5) and the function sampled at these points to determine the data  $f_j, j = 1, \dots, 150$ . The resulting IMLS surface is illustrated in Figure 5.6.

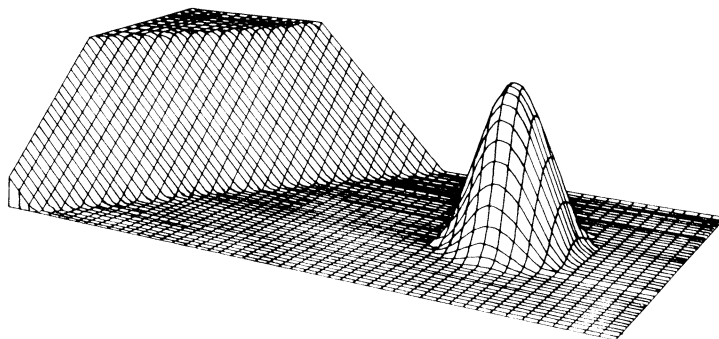


FIGURE 5.4

*A model problem*

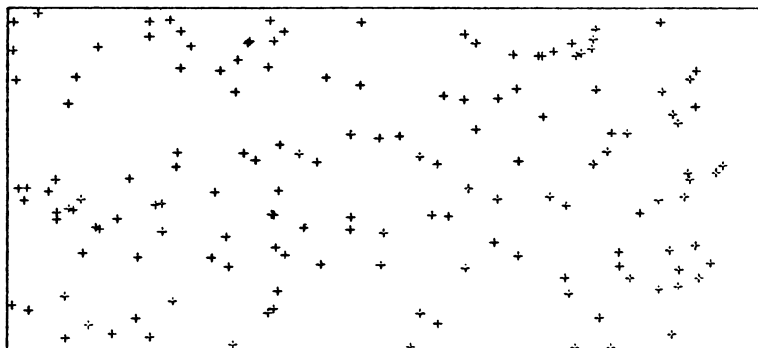


FIGURE 5.5  
150 randomly selected data points

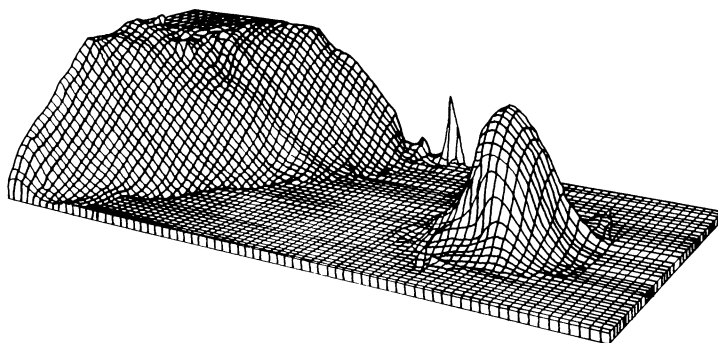


FIGURE 5.6  
An interpolating moving least squares surface

**6. A Related Projection Method.** It is evident that  $Gf$  does not have a convenient representation for computational purposes and that a new system of equations must be solved for each new value  $z$ . The following method requires the solution of  $n - 1$  equations (for  $n \geq 2$ ) at the  $N$  data points  $z_k, k = 1, 2, \dots, N$ , and an analogue of Theorem 4.2 holds for it as well.

Let the operator  $P: \tilde{C}^0(\bar{D}) \rightarrow C^m(D)$  be defined by

$$(6.1) \quad Pf(z) := \sum_{j=1}^N v^{(j)}(z) L_{z_j} f(z).$$

Thus, the  $N$  local approximating surfaces  $L_{z_j} f(z)$  in  $\text{span}\{b^{(i)}(z)\}_{i=1}^n$  are formed (implicitly) and  $Pf(z)$  is a weighted mean of these surfaces using the normalized weight functions. In view of Lemma 4.1 and Theorem 4.1,  $Pf \in C^m(D)$ . The cardinality properties of the  $v^{(j)}$  given by (i) in Lemma 4.1 show that  $Pf(z_k) = L_{z_k} f(z_k) = f_k, k = 1, \dots, N$ . In addition,

$$\begin{aligned} \frac{\partial}{\partial x} Gf(z) \Big|_{z=z_k} &= \sum_{j=1}^N \left[ \frac{\partial}{\partial x} v^{(j)}(z) L_{z_j} f(z) + v^{(j)}(z) \frac{\partial}{\partial x} L_{z_j} f(z) \right]_{z=z_k} \\ &= \frac{\partial}{\partial x} L_{z_k} f(z) \Big|_{z=z_k}, \quad k = 1, \dots, N, \end{aligned}$$

because of the cardinality properties of the  $v^{(j)}$  and the corollary to Lemma 4.1. A similar calculation for the  $y$ -derivative completes the demonstration that  $\text{grad } Pf(z_k) = \text{grad } L_{z_k} f(z_k)$ .

Since  $Pf$  interpolates and uses only the values  $f_k$ , it satisfies  $P^2f = Pf$  and is a projector. Furthermore,

$$Pb^{(k)} = \sum_{j=1}^N v^{(j)} L_{z_j} b_k^{(k)} = b^{(k)} \sum_{j=1}^N v^{(j)} = b^{(k)}, \quad k = 1, \dots, n,$$

because  $L_{z_j} b^{(k)} = b^{(k)}$ , and  $\sum_{j=1}^n v^{(j)} = 1$  (Lemma 4.1).  $P$  is a linear operator, and  $Pf$ , like  $Gf$ , admits a representation

$$(6.2) \quad Pf = \sum_{k=1}^N f_k F^{(k)},$$

where  $F^{(k)} = Pe^{(k)}$ , and the  $e^{(k)}$  are as in Section 5. We have proved

**THEOREM 6.1.** *Given the hypotheses of Theorem 4.1, the interpolation scheme defined by (6.1) defines a projector  $P$  onto  $N = \text{span}\{F^{(k)}\}_{k=1}^N$ , an  $N$ -dimensional subspace of  $C^m(D)$ , and  $\text{span}\{b^{(j)}\}_{j=1}^n$  is an  $n$ -dimensional subspace of  $N$ . Furthermore,  $\text{grad } Gf(z_k) = \text{grad } Pf(z_k) = \text{grad } L_{z_k} f(z_k)$ ,  $k = 1, \dots, N$ .*

In conclusion, it should be noted that  $Pf$  can be defined in the context of Section 2 by means of (6.1), in which case it is a moving least squares approximant to  $f$ . Discussion of numerical examples is postponed until the end of Section 7. We remark only that, if the surface  $Gf$  or  $Pf$  is evaluated at many more than  $N$  points, then the  $P$  projector will be the more economical. However, there is an added expense in that an additional  $N \times (n - 1)$  array must be stored (to retain the solution of (4.7) at the  $N$  data points).

**7. Choice of Weight Functions.** In formulating Lemma 4.1, specific hypotheses were made concerning the weight functions which admit a concise formulation of the results. In numerical practice other choices may be preferred, and we comment here on some natural selections. First, if  $N$  is very large, it may be deemed expedient to truncate the weight functions so that distant points enter with zero weight. This is to be done in such a way as to ensure the existence of  $Gf$  and maintain its continuity class. To this end, let  $m^{(i)}$ ,  $i = 1, \dots, N$ , be functions having the following properties:

- (i) The support of  $m^{(i)}$  is  $\bar{D}_i$ , where  $\bar{D}_i$  is simply connected and compact,  $z_i \in D_i$ ,  $m^{(i)}(z) > 0$  for  $z \in D_i$ , and  $m^{(i)} \in C^l(D)$ .
- (7.1) (ii) For any  $z \in \bar{D}$  there exist indices  $i_1, \dots, i_k$ ,  $k \geq n$ , dependent on  $z$ , such that  $z \in \bigcap_{j=1}^k D_{i_j}$  and  $\{z_{i_1}, \dots, z_{i_k}\}$  contains a subset of  $n$  points on which  $\{b^{(j)}\}_{j=1}^n$  is independent.

For the noninterpolating case one might choose  $w^{(i)} = m^{(i)}$ . Then (2.9) ensures that the normalized weight functions  $v^{(i)}$  are in  $C^l(D)$  and are nonnegative. Although  $W(\hat{z})$  (cf. (2.6)) may contain some zero diagonal elements, condition (ii) ensures the positive-definiteness of the coefficient matrix of the normal equations, and the conclusions of Sections 2 and 3 remain valid. In particular,  $Gf \in C^k(D)$ , where  $k = \min\{l, m\}$ .

In applications it may be convenient to take for each  $D_i$  an open disc of some radius  $\rho$ , centered at  $z_i$ . Then the only data points entering the calculation of  $Gf(z)$  are those satisfying  $|z - z_i| < \rho$ . Clearly,  $\rho$  must be chosen large enough so that conditions (7.1) are satisfied. If a standard regression surface is desired,  $\rho$  can be taken so large that  $D \subseteq D_i$ , and  $m^{(i)} = 1, i = 1, \dots, N$ . It may also be useful to let  $\rho$  depend on  $i$  in order to cope with problems in which the density of points shows large variations from one region to another.

In the case when interpolation is desired at some data point  $z_i$ , a suitable singular weight must be introduced for that point. This is sufficient, for the procedure is local. Thus, as a generalization of the choice made in Lemma 4.1, an appropriate choice is

$$(7.2) \quad w^{(i)}(z) = m^{(i)}(z)|z - z_i|^{-\alpha}, \quad \alpha_i < 0 \text{ and even,}$$

where  $m^{(i)}$  satisfies (7.1). It is necessary to check the differentiability class of  $v^{(i)}$  when several terms in the denominator, as well as the numerator, can have singularities of the type shown in (7.2). This can be accomplished, and one finds that, indeed,  $v^{(i)} \in C^l(D), i = 1, \dots, N$ , and  $Sf \in C^l(D)$ . As in Section 4, the orthogonalization process of Section 3 ensures the absence of singularities from the calculation of  $Gf$ . Hence the conclusions of Theorems 4.1, 4.2, 5.1, 6.1 hold, with the minor modification resulting from the differentiability class of the  $m^{(i)}$ . We find that  $L_x f, Gf$ , and  $Pf$  belong to  $C^k(D), k = \min\{l, m\}$ .

In most applications it will be convenient to define the weights  $w^{(i)}(z)$  as translates of a standard weight function  $w(z)$ . When interpolation is necessary, this standard weight function has its singularity at the origin. Then  $w^{(i)}(z) = w(z - z_i), i = 1, \dots, N$ . This is the procedure we adopt.

The following numerical examples are in one independent variable and the weight function used is

$$(7.3) \quad w(x) = \begin{cases} \frac{\rho^2}{x^2} \cos^2\left(\frac{\pi x}{2\rho}\right), & \text{for } |x| \leq \rho, \\ 0, & \text{for } |x| > \rho. \end{cases}$$

Similar effects are expected with the computationally less expensive choice

$$w(x) = \frac{\rho}{x^2} \left(1 - \frac{|x|}{\rho}\right)_+^2,$$

where the suffix  $+$  denotes truncation of the argument to zero when  $(1 - |x|/\rho) < 0$ .

*Example 5.* Here we calculate some cardinal functions on the abscissas  $-1$  (0.25) to 1. In each case the weight function (7.3) is used with  $\rho = 1$ . Two Shepard cardinal functions are illustrated in Figure 7.1, and we recall that this is the case  $n = 1$ , common to both the  $G$  and  $P$  projectors. With a truncated weight function, the cardinal functions must clearly have compact support. The rate of attenuation of amplitude is of interest and is influenced by the choice of  $\rho$ . One expects higher attenuation as  $\rho$  decreases, but recall that there is generally a lower bound on  $\rho$  determined by the choice of  $n$ .

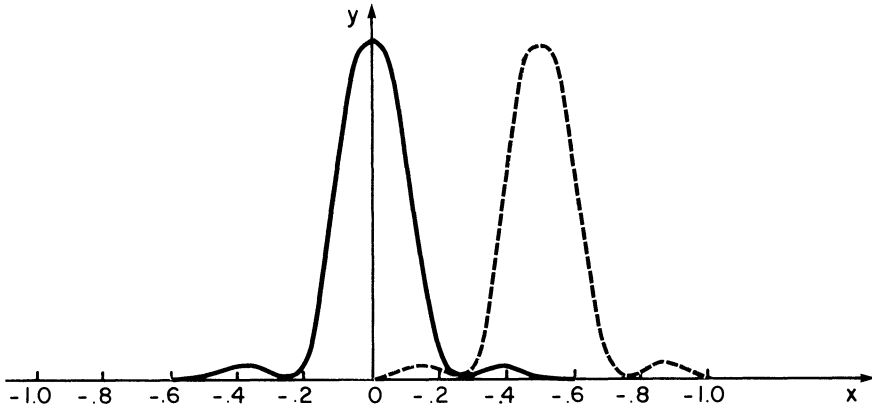


FIGURE 7.1

*Shepard cardinal functions on 9 data points,  $\rho = 1$*

For the  $G$  projector with  $n = 2$  and  $3$ , there is little visual change in these graphs. With  $n = 2$  there is just a little “overshoot” to negative function values on the side of nodes away from the data value  $+1$ , this effect being more pronounced with  $n = 3$  than in the case  $n = 2$ .

For the  $P$  projector with  $n = 2$  and  $3$ , the graphs are almost identical with those for  $G$ , the swing to negative function values being more pronounced in the case  $n = 2$  for  $P$  than for  $G$ .

*Example 6.* Data and basis functions are as in Example 2, and the weight function in (7.3) with  $\rho = 1$ . In Figure 7.2 we indicate the results for the  $G$  projector with  $n = 1, 2, 3$ . These results should be compared with Figure 5.2(a). The  $P$  projector gives similar results. When  $n = 3$ , it cannot be distinguished from the graph  $n = 3$  in Figure 7.2. The case  $n = 2$  gives a somewhat higher maximum than the case  $n = 3$ , between  $x = .3$  and  $.4$ .

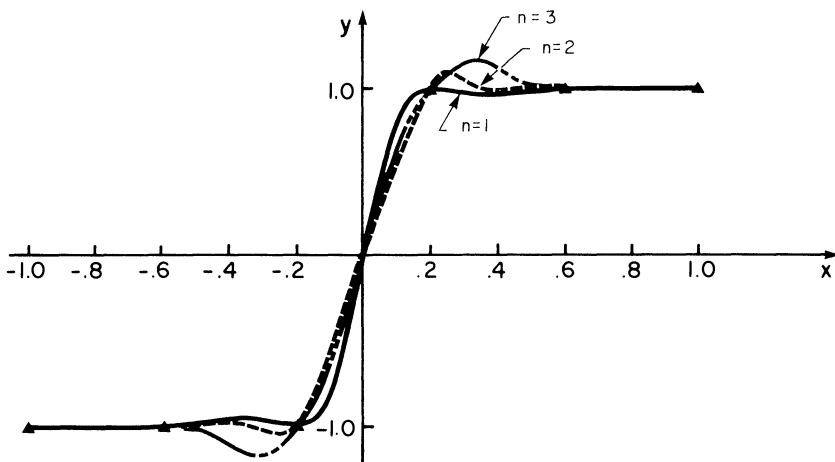


FIGURE 7.2

*The  $G$  projection with truncated weight function (cf. Figure 5.2(a))*

*Example 7.* The data and basis functions are as in Example 3, in which approximations for  $e^x$  on  $[-1, 1]$  are considered. We describe the numerical results qualitatively by reference to Figure 5.3. Calculations are described using weight function 7.3, with  $\rho = 1$ , and the projectors  $G$  and  $P$ . When  $n = 1$ , the errors behave very much as in Figure 5.3, as one must expect since the flat-spot phenomenon is retained. The magnitude of the error is similar, if a little larger.

When  $n = 2$ , the error is of a similar oscillatory nature but reduced in magnitude by about a half. When  $n = 3$ , there is a more dramatic reduction in error—with a ratio of about ten. These remarks apply to the  $G$  projection. The errors for  $P$  are comparable to those for  $G$ , although generally larger up to a factor of two. It would be interesting to know if a theoretical explanation can be found for this phenomenon.

**8. Asymptotic Behavior.** For all the weight functions discussed, the following observation is valid: If we define the diagonal matrix  $V = (\text{Tr } W)^{-1}W$ , then  $\lim_{|\hat{z}| \rightarrow \infty} V$  exists; denote it by  $V_0 = \text{diag}(v_0^{(1)}, \dots, v_0^{(N)})$ . For example, with  $W$  as in Lemma 4.1, part (iv) of that lemma gives  $V_0 = N^{-1}I$ .

For sufficiently large  $|\hat{z}|$ , both sides of (2.6) can be divided by  $\text{Tr } W$  yielding

$$BV(\hat{z})B^T \mathbf{a} = BV(\hat{z})\mathbf{f}.$$

These equations hold whether or not interpolation is being carried out. The matrix  $B$  is independent of  $\hat{z}$ ; thus  $\mathbf{a}$  is asymptotic to a constant vector  $\mathbf{a}^{(0)}$ . It follows that  $L_{\hat{z}}f \sim \sum_{k=1}^N a_i^{(0)}b^{(i)}$  and therefore  $Gf \sim \sum_{i=1}^n a_i^{(0)}b^{(i)}$ . On the other hand,  $Pf \sim \sum_{k=1}^N v_0^{(k)}L_{z_k}f$ . This is a weighted average of the local approximants  $L_{z_k}f$  at the data points, for  $\sum_{k=1}^N v_0^{(k)} = 1$  implies also  $\sum_{k=1}^N v_0^{(k)} = 1$ . It follows that  $Pf$  is also asymptotic to a linear combination of the  $b^{(i)}$ ,  $i = 1, \dots, n$ .

These observations show that if a polynomial basis is used, then “extrapolation” outside, say, the convex hull of the data points can display the usual undesirable polynomial behavior.

**9. Composite Methods.** The projectors  $G$  and  $P$  introduced above are computationally very expensive. In numerical practice they have been, and are likely to be, used as a first step in a two-stage process whose implementation will, in fact, restrict the amount of direct computation with  $G$  or  $P$ . An approach in common use is to determine a regular grid of points whose convex hull contains  $\bar{D}$  and use information about  $Gf$ , say, at the points of the *regular* grid in order to produce a smooth surface by a more familiar tensor product, or finite-element scheme. The information required on the regular grid may be values of  $Gf$  and its first few partial derivatives. The use of such derivative information requires a smoothness result like Theorem 4.1, of course.

If the second stage of interpolation, on the regular grid, is a projection method determined by a projector  $Q$ , then the surface generated from samples  $f_1, \dots, f_N$  of a function  $f$  is a representation of the function  $QGf$ . However, unless the points  $z_1, \dots, z_N$  are also vertices of the regular grid,  $QGf$  will not generally be an interpolant for  $f$  at these points. In this case it is clear that the operator  $QG$  will not be a projector. This process is often described as “moving the data” onto a regular grid of points.



For many workers applying those techniques the loss of the interpolating property is seen as a serious disadvantage. A technique which is gaining in popularity is to first form a triangulation of the plane based on the vertices  $\{z_i\}_{i=1}^N$ , then generate the data  $f_i$  at these points, together with derivative data sampled from  $Gf$  at the same points, and use such information to generate, triangle by triangle, a  $C^1$  surface over the whole plane. This is, of course, a "finite-element" procedure. Again, there will be an underlying finite-element projector  $Q$ , but this time  $QGf$  will interpolate  $f$ . It follows immediately that the composite operator  $QG$  determines a projector. It should be noted, however, that it will not generally be the case that  $QG = GQ$  (see [6] for further discussion).

Consider, for the moment, the selection of an appropriate  $C^1$  finite-element scheme for use on a general triangulation. Numerical experience shows that computation of second order partial derivatives for  $Gf$  should be avoided; cf. [10]. In general, while  $Gf$  may be a good approximation for  $f$ , the derivatives of  $Gf$  become increasingly poor approximations for the derivatives of  $f$  as the order of derivative increases. Thus, a finite-element scheme, using only function-value and first-derivative information at the nodes and having  $C^1$ -interelement continuity, is desirable. Candidates for such a scheme are associated with Clough and Tocher [2], Mansfield [8], and Powell and Sabin [9].

*Example 8.* The data and basis functions are as in Example 4. In Figures 9.1(a) and (b) we illustrate surfaces determined by a composition  $QG$  which is *not* a projector. The projector  $Q$  is defined by a piecewise quadratic rectangular finite element requiring only function and derivative evaluations of  $Gf$  at the vertices of the rectangles (this element is developed from a triangular element of Powell and Sabin [9] and is described in detail in the work of Ritchie [10]). First the smoothing effect of  $Q$  is clear (cf. Figure 5.6 which is a picture of  $Gf$ ), and the improved resolution in going from 8 rectangles (45 nodes) in 9.1(a) to 32 rectangles (135 nodes) in 9.1(b) is clear. Note that, for these particular examples, the use of  $Pf$  rather than  $Gf$  would not be justified since the number of data points exceeds the number of samplings of  $Pf$  needed to generate  $QPf$ .

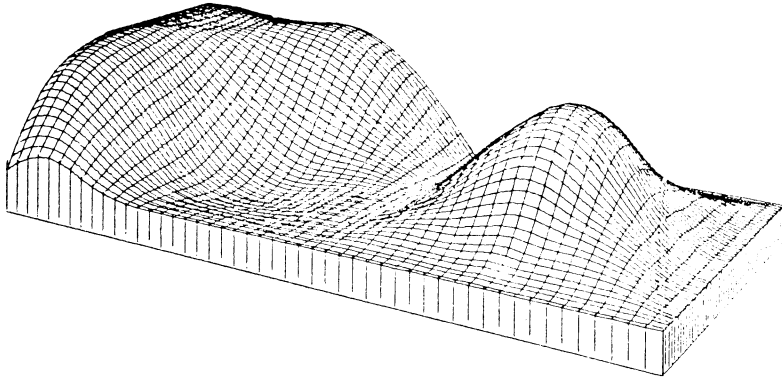


FIGURE 9.1(a)

*Composition of a moving least squares projector with a piecewise quadratic rectangular finite-element projector,  $4 \times 2$  rectangular grid, 45 nodal values*

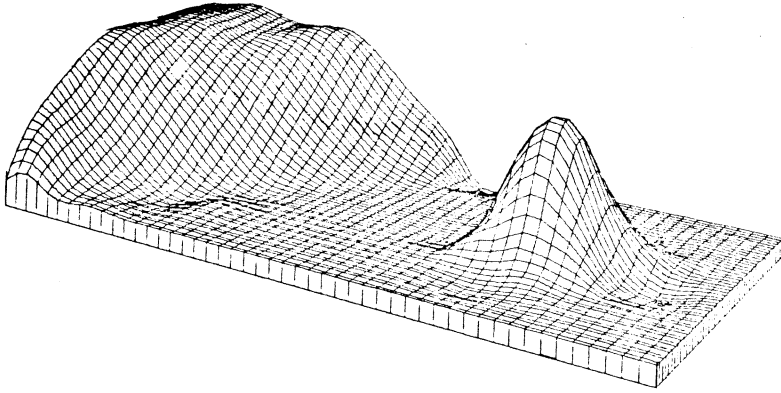


FIGURE 9.1(b)

*Composition of a moving least squares projector with a piecewise quadratic rectangular finite-element projector,  $8 \times 4$  rectangular grid, 135 nodal values*

Department of Mathematics and Statistics  
The University of Calgary  
Calgary, Alberta, Canada T2N 1N4

1. R. E. BARNHILL, *Representation and Approximation of Surfaces, Mathematical Software III*, Academic Press, New York, 1977, pp. 69–120.
2. R. W. CLOUGH & J. L. TOCHER, "Finite element stiffness matrices for analysis of plates in bending," in *Proc. Conf. Matrix Methods in Structural Mechanics*, Wright-Patterson A.F.B., Ohio, 1965.
3. R. FRANKE & G. NIELSON, *Smooth Interpolation of Large Sets of Scattered Data*, Technical Report #NPS-53-79-005, Naval Postgraduate School, Monterey, Calif., 1979.
4. W. J. GORDON & J. A. WIXOM, "Shepard's method of 'metric interpolation' to bivariate and multivariate data," *Math. Comp.*, v. 32, 1978, pp. 253–264.
5. P. LANCASTER, "Moving weighted least-squares methods," in *Polynomial and Spline Approximation* (B. N. Sahney, Ed.), NATO Advanced Study Institute Series C, Reidel, Dordrecht, 1979, pp. 103–120.
6. P. LANCASTER, "Composite methods for generating surfaces," in *Polynomial and Spline Approximation* (B. N. Sahney, Ed.), NATO Advanced Study Institute Series C, Riedel, Dordrecht, 1979, pp. 91–102.
7. D. H. MCLAIN, "Drawing contours from arbitrary data points," *Comput. J.*, v. 17, 1974, pp. 318–324.
8. L. MANSFIELD, "Higher order compatible triangular finite elements," *Numer. Math.*, v. 22, 1974, pp. 89–97.
9. M. J. D. POWELL & M. A. SABIN, "Piecewise quadratic approximation on triangles," *ACM Trans. Math. Software*, v. 3, 1977, pp. 316–325.
10. S. RITCHIE, *Representation of Surfaces by Finite Elements*, M.Sc. Thesis, University of Calgary, 1978.
11. D. SHEPARD, *A Two-Dimensional Interpolation Function for Irregularly Spaced Points*, Proc. 1968 A.C.M. Nat. Conf., pp. 517–524.