Optimal Bandwidth Selection for MLS Surfaces

Hao Wang Carlos E. Scheidegger Cláudio T. Silva
University of Utah

Abstract

We address the problem of bandwidth selection in MLS surfaces. While the problem has received relatively little attention in the literature, we show that appropriate selection plays a critical role in the quality of reconstructed surfaces. We formulate the MLS polynomial fitting step as a kernel regression problem for both noiseless and noisy data. Based on this framework, we develop fast algorithms to find optimal bandwidths for a large class of weight functions. We show experimental comparisons of our method, which outperforms heuristically chosen functions and weights previously proposed. We conclude with a discussion of the implications of the Levin's two-step MLS projection for bandwidth selection.

1 Introduction

There has been a large, recent interest in the area of surface reconstruction from point-sampled data. This work has been motivated by a set of important applications where the ability to define continuous surfaces out of a set of discrete point samples is necessary. The resolution and availability of current 3-D range scanners that output a very large set of unconnected points has driven the development of effective techniques to reconstruct surfaces directly from the point cloud data.

One of the main challenges in effectively using these data is dealing with the inherently noisy and irregular nature of the acquired dataset. The noise introduced by these point-of-view scanners is dependent on factors such as the material of the object being reconstructed, incident angle of the range-finder laser on the object, and distance to the scanner. This means there typically are different noise levels throughout the range scan, and reconstruction techniques must cope with these issues to be successful. Notably, these three-dimensional range scans are currently being used in digital archeology, where the acquired data is considered a historical artifact. In some of these scans, it is possible to recover details that elucidate the technique used by the artist [15]. It becomes very important, then, to recover as much detail as possible.

In particular, Moving Least-Squares surfaces [12, 3] have shown to be a powerful and popular surface reconstruction method. An MLS surface is defined by a point cloud P and a *projection operator* $f : \mathbb{R}^2 \to \mathbb{R}^2$, which takes a point r in the neighborhood of a C^∞ surface S (the *MLS surface*), and returns a point $f(r) \in S$ close to r. The points in P might generally not be in S, but in the limit of increasing density of P, it can be shown that P converges to S geometrically and topologically [8]. One of the main attractions of MLS is its natural resilience to noise. This is easily done by changing the *bandwidths* of the point samples — their influence radii, essentially.

In this paper, we address the problem of accurately selecting these bandwidths. In particular, we show that by reformulating part of the MLS procedure, it is possible to derive analytical optima for the bandwidths that minimize certain criteria we will make precise. An appropriate choice of bandwidth plays a critical role in the reconstruction of geometric detail. We start by reviewing related work. Then, we discuss our formulation of the problem in terms of kernel regression, and the derivation of optimal bandwidths. We present implementation details and results on the following section. Finally, we discuss some implications of the techniques and suggest future work.

2 Related Work

There have been many different proposed formulations of MLS surfaces. These include the projection operator of Levin and Alexa et al [12, 3] and its implicit surface formulation [6]. Simplified versions with favorable computational requirements have also been proposed, usually derived in as a combination of weighted centroids and normal fields [2, 10]. Fleishman et al. presented extensions that increase the method's robustness to outliers while also introducing sharp features in the surface [9]. In this work, we use the two-step projection procedure of Alexa et al. Zwicker et al. suggested the most popular variant of the k-nearest neighbor rule for bandwidth determination [25].

While much work has gone into different MLS surface formulations, relatively little attention has been paid

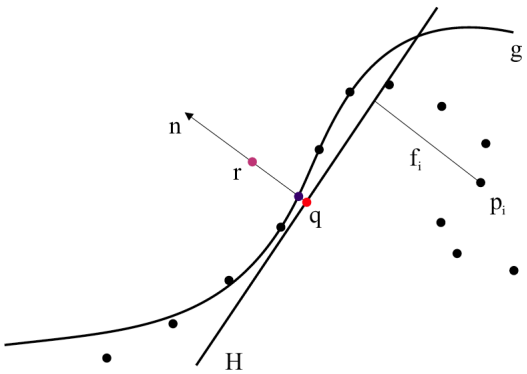

Figure 1. **An overview of the MLS surface projection operator we use in this paper. (from [4])**

to choosing bandwidths for the surface projection. Adamson et al. [1] originally proposed extending their weighted-centroid formulation to incorporate elliptical kernels, allowing the samples to conform to the principal curvatures of the surface. While they provide an argument for picking ellipsoidal kernels for clean samples, there is little discussion of the influence of noise in the anisotropy estimation. It is not clear, also, how to derive optimality criteria using those definitions. Lipman et al. show a tight error bound for the pointwise error in the MLS approximation formula [13]. They use this bound to numerically minimize the error for each projection. This mathematically sound approach outperforms heuristically chosen neighborhood sizes in accuracy. On the other hand, they depend on a search scheme that can be computationally costly.

3 Computing Optimal Bandwidths

In this section, we first review the MLS projection operator, (following the presentation in [4]) and, in particular, the polynomial fitting step. Then we will reformulate this second step as a kernel regression problem. This leads to the discussion of weight functions and optimal bandwidths for 2-D functional data, after which we move on to the generalization into 3-D data. Finally, we discuss how to incorporate the method into MLS projections.

3.1 Background

Given a set of input points $P = p_i \subset R^2$ and a point r to be projected on S, the MLS surface is defined in two steps. In the first step, we find a local approximating hyperplane H that minimizes a locally-weighted sum of squared distances from p_i to H. The weights are given as a funtion of the distance from the projection of r onto H (called q), as shown in Figure 1. The local plane $H = \{x|\langle n, x\rangle - D = 0, x \in \mathbb{R}^2\}, n \in \mathbb{R}^2, ||n|| = 1$ is found by minimizing

$$\sum_{i=1}^{N} \left(\langle n, p_i\rangle - D\right)^2 \theta_i(||p_i - q||) \tag{1}$$

where θ is the weight function, the principal subject of this paper. After H is found, a second step finds a local polynomial approximation g of the surface, by minimizing

$$\sum_{i=1}^{N} \left(g(x_i, y_i) - f_i\right)^2 \theta_i(||p_i - q||) \tag{2}$$

where (x_i, y_i) are the representations of q_i, the projections of p_i onto H expressed in the local coordinate system H, and f_i is the signed height of p_i over H. The same weighting functions are used, but notice that the weights now are not part of the optimization (q is fixed), so the optimization is linear. The most commonly used weighting function is a (possibly truncated) Gaussian:

$$w_h(r) = e^{-\frac{r^2}{h^2}} \chi_{[0,k)}(r) \tag{3}$$

k indicates a cutoff that is typically used for computational efficiency, limiting the distance query into the spatial data structures. Although this function works well in practice, it is empirically chosen and little work has been done with respect to which weight functions are valid and which ones are optimal. In addition, this weight function has confined the shape of the neighborhood of the reference point to be isotropic, which might not be geometrically justified.

3.2 2-D Kernel Regression

We modify the problem setting to transform the interpolation problem into a regression problem. To facilitate our discussion, we adopt the standard kernel regression terminology in statistics: Given random variables $X_1, ..., X_n$ with density $g(X)$ and response variables $Y_1, ..., Y_n$ that satisfy:

$$Y_i = f(X_i) + v^{1/2}\epsilon_i, \text{i = 1, .., n}$$

where v is the variance of the noise and ϵ_i are independent random variables for which

$$E(\epsilon_i|X_1, ..., X_n) = 0, Var(\epsilon_i|X_1, ..., X_n) = 1$$

The value of $f(x)$ at a specific point x is estimated by evaluating a polynomial $p(t)$ of degree d at $t = x$. $p(t)$ is defined as follows:

$$p(t) = \hat{\beta}_0 + \hat{\beta}_1(t - x) + ... + \hat{\beta}_d(t - x)^d \tag{4}$$

where $(\hat{\beta}_0, \hat{\beta}_1, ..., \hat{\beta}_d)$ minimizes

$$\sum_{i=1}^{n} (Y_i - p(x))^2 K_h(X_i - x) \tag{5}$$

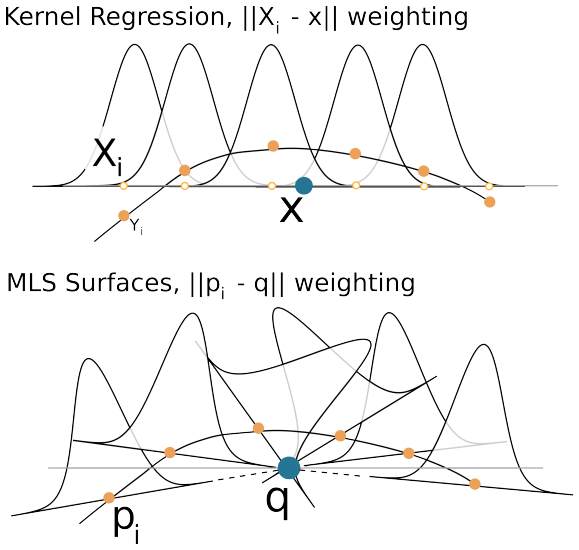

Figure 2. **Even though kernel regression and MLS surfaces both employ kernels, the way samples are weighted is different.**

Here $K_h(u) = \frac{1}{h}K(\frac{u}{h})$ is a weight function (kernel) which assigns large weights to points within some neighborhood of x and small weights outside of the neighborhood. The size of neighborhood is controlled by parameter h. Notice that the weighting in kernel regression is a function of the distance between the values in the domain of the functional, $||X_i - x||$, while in MLS surfaces the weighting is a function of the distance between the actual samples and the center of the reference frame, $||p_i - q||$. This is illustrated in Figure 2.

The kernel function is usually chosen to be a symmetric and unimodal probability density function [7]. Common choices of kernels include the *normal kernel*, *Epanechnikov kernel*, *biweight kernel*, etc. It has been proved [24] that in 2-D the bandwidth rather than the kernel plays the vital role in achieving high quality regression result. In other words, we can replace one kernel for another in regression without causing much loss of acuracy if we use the optimal bandwidth for each of them respectively. The Gaussian kernel we use is within 95% of the efficiency of the Epanechnikov kernel, the asymptotically optimal choice.

In the following discussions of this paper, we will focus on two error criteria for the evaluation of performance of kernel regression: the first one is the Mean Squared Error (MSE) which emphasizes the expected error of the local specific point of interest:

$$\text{MSE}(p(x)) = (E\left[p(x) - f(x)\right])^2 \tag{6}$$

and the other one is the integral of MSE over the functional domain which summarizes the overall expected errors:

$$\text{MISE}(p(x)) = \int \text{MSE}(x)g(x)dx \tag{7}$$

We have the following formula for MSE [18]:

$$\text{MSE}(p(x)) \simeq \frac{1}{4}h^4(f''(x))^2\mu_2(K)^2 + \frac{R(K)v}{nhg(x)} \tag{8}$$

where

$$\mu_2(K) = \int z^2K(z)dz,\ R(K) = \int K^2(z)dz$$

The error term of MSE in Equation 8 is $o_P(n^{-1}h^{-1} + h^2)$. Letting the derivative of the approximated MSE be 0 and solving for h, we get the h that minimizes MSE:

$$h_{opt} = C\left(\frac{v}{ng(x)(f''(x))^2}\right)^{1/5} \tag{9}$$

where $C = (R(K)/\mu_2(K)^2)^{\frac{1}{5}}$ is a constant dependent on the kernel, e.g, for *normal kernel* $C = 1/(2\sqrt{\pi})^{\frac{1}{5}}$.

Computing the optimal bandwidth involves calculating $g(x)$, $f''(x)$ and v, which are related to the underlying function that we do not know. Our solution is to use the "plug-in" method [22, 18, 21], which uses statistical inference to estimate these values and then plugs the estimator into the formula of optimal bandwidth. Among various ways to estimate the density $g(x)$, we choose to use kernel density estimation because of its accuracy and close relation to kernel regression:

$$g(x) = \frac{1}{nh}\sum_{i=1}^{n} K\left(\frac{x - X_i}{h}\right) \tag{10}$$

Once again, we need to choose the right bandwidth. We are only looking for an unbiased estimator of $g(x)$, so many choices are possible. We use the *normal scale rule* [24] to select the bandwidth:

$$h = \left(\frac{8\pi^{1/2}C}{3n}\right)^{1/5}\sigma \tag{11}$$

where C is defined as before and σ is the sample standard deviation. The normal scale rule essentially uses the optimal bandwidth for normal density as the bandwidth.

To estimate the second derivatives, we apply ordinary least squares quartic polynomial fitting to approximate the underlying functional. It has been shown [11] that it is necessary to divide the functional domain Ω into several "blocks" to make the method work for fast oscillating functionals. The "blocking method" divides the domain according to Mallow's C_p [14]. C_p is a statistic defined for the regression model of fitting each of the N blocks of data to

a $p-1$ degree of polynomial. The optimal N is the one that minimizes C_p. Technically, N is chosen from the set $\{1, 2, ..., N_{max}\}$ to minimize:

$$C_p(N) = \frac{RSS(N)(n - pN_{max})}{RSS(N_{max})} - (n - 2pN) \quad (12)$$

In our case, since we fit data to a quartic polynomial, $p = 5$ and the C_p becomes:

$$C_p(N) = \frac{RSS(N)(n - 5N_{max})}{RSS(N_{max})} - (n - 10N) \quad (13)$$

where $RSS(N)$ is the residual sum of squares over N blocks. In order to reduce the chance of overfitting, the following formula for N_{max} [18] has been suggested:

$$N_{max} = max\{min(\lfloor n/20 \rfloor, N^*), 1\} \quad (14)$$

where N^* is a user specified parameter which sets the upper limit of the number of blocks. As for variance, we use the estimator suggested by Ruppert et.al [18]:

$$v = \frac{RSS(N)}{n - 5N} \quad (15)$$

Assume $X_i \in [a, b]$, the optimal bandwidth based on MISE can be derived in an analogous way and the result is:

$$h_{opt} = C \left(\frac{v(b-a)}{n \int_a^b (f''(x))^2 g(x) dx} \right)^{1/5} \quad (16)$$

The only new difficulty that arises in the MISE based formula is estimation of the integral of the square of second derivative. We approximate the integral by Monte Carlo Integration:

$$\int_\Omega (f''(x))^2 g(x) dx \simeq \frac{b-a}{n} \sum_{i=1}^{N} \sum_{\{j: x_j \in \text{block i}\}} p''(x_j)^2 \quad (17)$$

3.3 3-D Kernel Regression

In 3-D space, we work on vectors $\mathbf{x} = (x_1, x_2)^T$ instead of scalars. The formulation of kernel regression problem in 3-D space is analogous to that in 2-D space except that the bandwidth B is now a matrix and the kernel function $K_B(\mathbf{x}) = |B|^{-1/2} K(B^{1/2}\mathbf{x})$. $K(\mathbf{x})$ can be constructed from a univarite kernel function in one of the two ways:

$$K(\mathbf{x}) = k(x_1)k(x_2) \text{ or } K(\mathbf{x}) = \frac{k((\mathbf{x}^T\mathbf{x})^{1/2})}{\int k((\mathbf{x}^T\mathbf{x})^{1/2}) d\mathbf{x}}$$

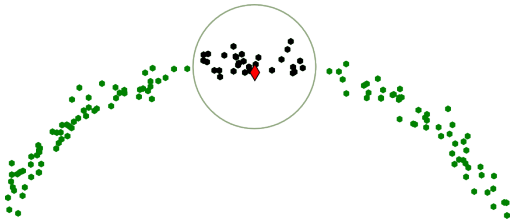

Figure 3. **Incorporating the kernel selector in MLS surfaces. A neighborhood around q (red diamond) is chosen among the sample data and kernel regression is applied to points within the neighborhood.**

If the univariate kernel is *normal*, we get the same bivariate Gaussian kernel in either way of construction:

$$K(\mathbf{x}) = \frac{1}{2\pi} e^{-\frac{1}{2}\mathbf{x}^T\mathbf{x}} \quad (18)$$

This property of Gaussian kernel motivates us to use it as our kernel function in the following discussions.

Different from 2-D kernel regression, the bandwidth matrix B not only controls the size of the neighborhood but also the shape. If B takes the following form:

$$\mathbf{B} = \begin{bmatrix} h^2 & 0 \\ 0 & h^2 \end{bmatrix} \quad (19)$$

the shape of the neighborhood would be circular on the XY plane; if B is of the following form

$$\mathbf{B} = \begin{bmatrix} h_1^2 & 0 \\ 0 & h_2^2 \end{bmatrix} \quad (20)$$

then the neighborhood on the XY plane is an ellipse with its axes parallel to the coordinate axes. The MSE for an arbitrary B [19] is:

$$\text{MSE} \simeq \frac{1}{4}\mu_2(K)^2 tr^2(\mathbf{B}\mathcal{B}_f(\mathbf{x})) + \frac{R(K)v}{ng(\mathbf{x})|\mathbf{B}|^{1/2}} \quad (21)$$

where

$$\mu_2(K)\mathbf{I} = \int \mathbf{z}\mathbf{z}^T K(\mathbf{z}) d\mathbf{z}, \; R(K) = \int K^2(\mathbf{z}) d\mathbf{z}$$

$\mathcal{B}_f(\mathbf{x})$ is the Hessian Matrix and $tr(\mathbf{B})$ is the trace of $\mathbf{B}$. The error term of MSE in Equation (21) is $o_P\{n^{-1}|\mathbf{B}|^{-1/2} + tr^2(\mathbf{B})\}$.

To find the optimal bandwidths, we plug in a particular form of B into approximated MSE and try to find the minimizer. For circular kernel based on MSE we have:

$$h_{opt} = \left(\frac{2R(K)v}{n\mu_2(K)^2 (\frac{\partial^2 f}{\partial {x_1}^2} + \frac{\partial^2 f}{\partial {x_2}^2})^2 g(\mathbf{x})} \right)^{1/6} \quad (22)$$

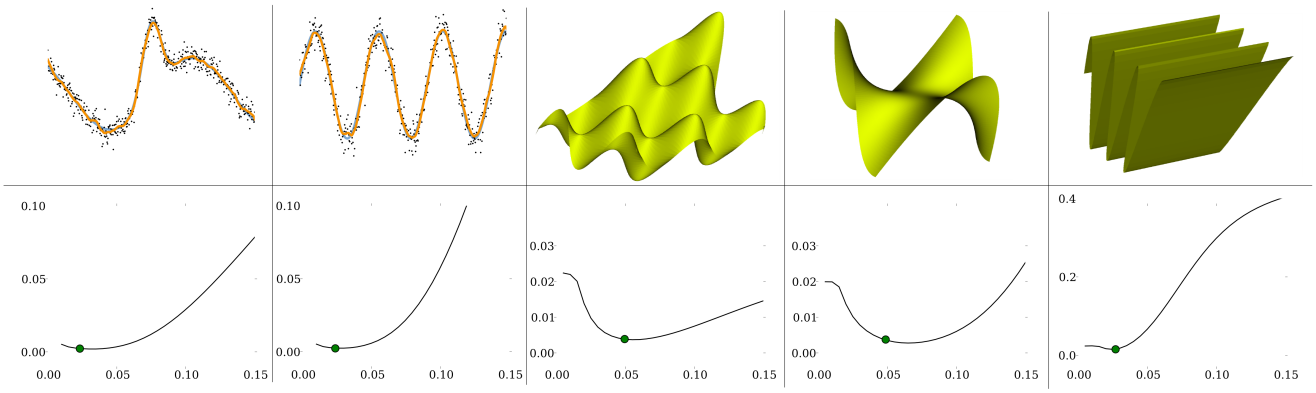

Figure 4. **Optimal bandwidths for 2-D and 3-D functional data, with circular kernels in the case of 3-D data. The underlying functionals are shown in Tables 1 and 2. The bottom row shows plots of bandwidth vs. Mean Squared Error. Our algorithm finds a value close to the real minimum in all cases.**

If we choose h based on MISE, then the optimal h is (Assume $X_i \in [a,b] \times [c,d]$.):

$$h_{opt} = \left(\frac{2R(K)v(b-a)(d-c)}{n\mu_2(K)^2 \int (\frac{\partial^2 f}{\partial {x_1}^2} + \frac{\partial^2 f}{\partial {x_2}^2})^2 g(\mathbf{x}) d\mathbf{x}} \right)^{1/6} \quad (23)$$

For elliptical kernel, let:

$$C_1 = \frac{R(K)v}{ng(x)}, C_2 = \mu_2(K)^2, d_1 = \frac{\partial^2 f}{\partial {x_1}^2}, d_2 = \frac{\partial^2 f}{\partial {x_2}^2}$$

When $d_1 d_2 > 0$, the optimal h_1 and h_2 satisfy:

$$h_1^2 = \sqrt{\frac{d_2}{d_1}} \left(\frac{C_1}{2C_2 d_1 d_2} \right)^{\frac{1}{3}} \quad (24)$$

$$h_2^2 = \sqrt{\frac{d_1}{d_2}} \left(\frac{C_1}{2C_2 d_1 d_2} \right)^{\frac{1}{3}} \quad (25)$$

When $d_1 d_2 < 0$, no minimizer exists for MSE. In this case, one can either choose to use the circular kernel or the MISE based optimal bandwidth matrix which satisfies:

$$h_1^2 = \left(\frac{I_3}{I_1} \right)^{1/4} \left(\frac{R(K)v(b-a)(d-c)}{n\mu_2(K)^2(\sqrt{I_1 I_3} + I_2)} \right)^{1/3} \quad (26)$$

$$h_2^2 = \left(\frac{I_1}{I_3} \right)^{1/4} \left(\frac{R(K)v(b-a)(d-c)}{n\mu_2(K)^2(\sqrt{I_1 I_3} + I_2)} \right)^{1/3} \quad (27)$$

where

$$I_1 = \int \left(\frac{\partial^2 f}{\partial {x_1}^2} \right)^2 g(\mathbf{x}) d\mathbf{x} \quad (28)$$

$$I_2 = \int \frac{\partial^2 f}{\partial x_1^2} \frac{\partial^2 f}{\partial {x_2}^2} g(\mathbf{x}) d\mathbf{x} \quad (29)$$

$$I_3 = \int \left(\frac{\partial^2 f}{\partial {x_2}^2} \right)^2 g(\mathbf{x}) d\mathbf{x} \quad (30)$$

and $X_i \in [a,b] \times [c,d]$.

As in 2-D, we approximate the underlying functional by ordinary quartic polynomial fitting. The integrals are approximated by Monte Carlo Integration and the densities are estimated by kernel density estimation with optimal bandwidth matrix $n^{-1/3}\Sigma$ (Σ is the sample covariance matrix) found again by the normal scale rule[23]. We also use the 3-D version of "blocking" method to adapt our method to fast-oscillating functionals.

3.4 MLS

We have investigated two ways to incorporate our method into MLS. We first chose a neighborhood for each q which ensures that all points in that neighborhood are sampled from a functional instead of, say, double-sheeted surfaces, and then applied kernel regression using kernel regression weighting within that neighborhood (Figure 3); We also used kernel regression with MLS weighting on the whole dataset but this time we employed local weighted polynomial fitting to estimate the second derivatives of the underlying surfaces, etc. to find the optimal bandwidth.

The advantage of the first method is that by simply using linear polynomial fitting, we can have a reconstruction whose precision is higher than heuristic approaches (Figure 6). Since it uses the kernel weighting scheme instead of the MLS weighting scheme (Figure 2), the distances between the sample points and underlying surface along the normal directions do not affect the weights assigned to each point. This helps reduce estimator bias: using MLS weighting, there is an inherent bias towards $f(x) = 0$. The nontrivial part of this method is how to choose a neighborhood for each q that quanrantees all the points selected are sampled from a functional. We empirically chose the size of the neighborhood for q in our experiments. However, this

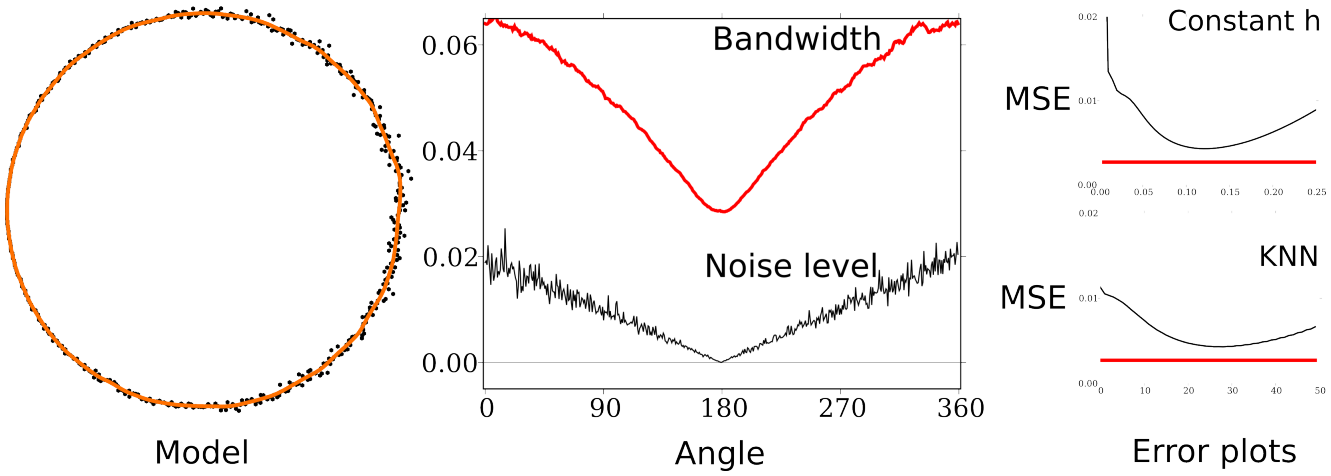

Figure 5. **MISE based circular kernel incoporated into *MLS*. We use as a model a circle with varying levels of noise (on the left). In the middle, the red line shows the selected bandwidth as a function of angle, and the black line shows the measured noise. On the right column, we show the results of choosing a constant h or a certain value of k for the k nearest neighbor heuristic. Notice that the optimal bandwidth around regions of no noise is not zero (see Section 5).**

choice is not critical: in places where the neighborhoods and reference frames are likely to not be functional, the MLS method will typically fail [5, 16].

As for the second method, there is no trouble of selecting a neighborhood for q. However, since Euclidean distances are used here, nonlinear regression instead of linear regression should be applied to alleviate the bias problem. It is comparatively easy to find optimal bandwidths for nonlinear polynomial fitting in 2-D [18], but we were unable to generalize the result to 3-D due to the complexity of mathematics. In the following sections, we present experimental results using the first method.

4 Experimental Results

Since the second step of MLS is polynomial fitting on functional data, testing our methods on functional data is enough for evaluation their effectiveness. For completeness, however, we present experimental results of the method's performance after it is incorporated into MLS. To test our methods for functional data, we compared our results with the real optimum; to test our methods in MLS, we compared our methods with heuristic approaches for bandwidth selection.

4.1 Functionals

For 2-D functionals, we tested our algorithm on points sampled from functions in Table 1. Sample size is 500 and the distribution of noises is $N(0, 0.2)$. To evaluate the performance of the algorithm, we checked 100 possible values of the bandwidth from the interval $[0, 1]$ and compared the integral squared error of the approximated curves using these bandwidths against the one produced by our algorithm. Figure 4 shows the comparison results. Our results are close to optimal, even in data sets with high noise level or fast oscillation.

Name	Function
F_1	$sin(8x-4)+e^{-16(4x-2)^2}$
F_2	$sin(6.5\pi x)$

Table 1. **2-D functionals in experiments**

Name	Function
F_3	$0.3cos(12x)sin(9y)+e^{-\frac{9}{4}(x^2+y^2)}$
F_4	$(2x-1)^3-3(2x-1)(2y-1)^2$
F_5	$cos(20x)$

Table 2. **3-D functionals in experiments**

For 3-D functionals, we chose to test MISE based circular kernel in the same setting as 2-D functionals. The underlying functionals are shown in Table 2. As in 2-D case, our reconstruction error is close to the real minimum (Figure 4).

4.2 MLS

We have evaluated our method quantitatively reconstructing a circle (Figure 5), a sphere and a torus (Figure 6). In our evaluation, we compared our method against two

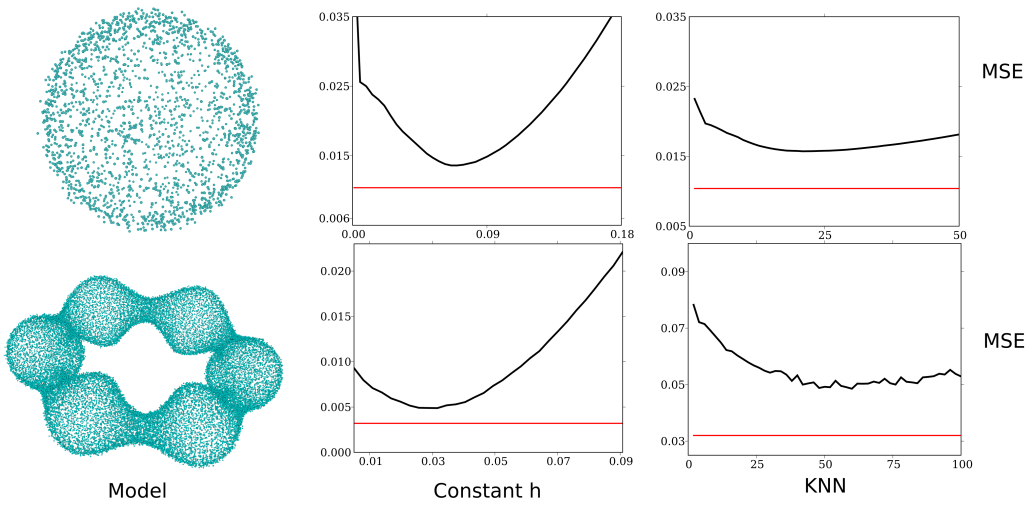

Figure 6. **MISE based circular kernel incorporated into *MLS*, using two synthetic models whose ground truth is known. The top model is a sphere, the bottom one a torus with varying radius. The middle column shows the mean squared error over the entire surface of varying h, and the right column shows the error using the k nearest neighbor heuristic. Our results are represented by the red lines in the plots.**

commonly used heuristic approaches for bandwidth selection - a constant h for all projected points and the k-Nearest Neighbor method which uses a third of the distance from q to its kth nearest neighbor as the standard deviation of the Gaussian kernel. As in our experiments for functional data, we enumerated possible values of h and k and compared the reconstruction errors (distances between the reconstructed surface and real underlying surface) of different methods.

As demonstrated by the experimental results, our method outperforms the heuristic approaches. First, it is not clear how to find the optimal h or k for all possible projected points, while we have a closed-form analytical solution for a large family of cases. Second, and more importantly, the results show that there isn't a particular h or k that is best suited for the entire model, even in the case of constant noise. As we argue in Section 5, our method outperforms both algorithms because it makes fewer assumptions about the neighborhood configurations.

We also tested our method on real world models. Figure 7 shows that a visually acceptable reconstruction by heuristic approaches may fail to capture the geometry precisely, but our approach is capable of achieving close approximation to the real surface.

Since we used the first method of incorporating the kernel selectors into MLS, we had to choose a neighborhood for each q. The size of this neighborhood affects the quality of reconstruction but we've found that for a wide range of neighbor sizes our method all produced reconstruction errors smaller than heuristic approaches. The neighborhoods used in the figures are not necessarily the best. For example, to reconstruct the circle (Figure 5), we sed $(maxX_i - minX_i)/10$ as the neighborhood size but $(maxX_i - minX_i)/5$ produced better results. The important point, however, is that both of them significantly outperform the heuristic approaches.

Our bandwidth selector is fast. We tested the data for Figure 5 on an 2.8G Hz Linux machine with AMD Athlon(tm) 64 X2 Dual Core Processor 5600+. The second step (polynomial fitting) of MLS took a total of 8.917 seconds, among which 1.160 seconds were used to calculate the bandwidths. The time consumption of bandwidth computation is only 13% of the total time spent on the second step of MLS.

5 Discussion

In this section, we discuss issues such as the impact of this technique, its limitations, its applicability in other MLS surface definitions, and others.

Other MLS definitions We have implemented the bandwidth selector for the classic MLS surface definition of Levin and Alexa et al, but there are many different and popular formulations. We believe the current framework can be easily extensible to many different variants. For example, Alexa and Anderson's implicit surface formulation defines the MLS surface as essentially the set of points whose

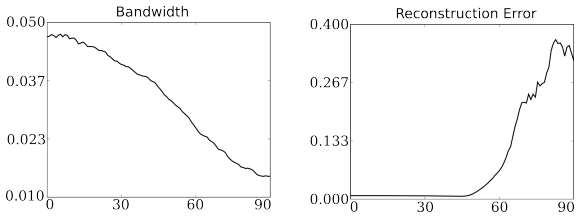

Figure 8. **Bandwidth and reconstruction error when the reference plane is rotated. Points are sampled from a circle with noise added. Left plot shows the bandwidth of a reference point when the rotation angle grows from 0 to 90 degrees; The right plot shows the reconstruction error associated with each reference point.**

local normal is perpendicular to the vector pointing to the neighborhood centroid [2]. The projection is usually implemented by a custom root-finding algorithm that builds a sequence of approximations of the normal and centroid. A natural way of incorporating the bandwidth selection is to use, at each iteration, the plane defined by the current best approximation of the projection and normal. While closer investigation is necessary, we believe that the method will converge to a point in the surface with optimal weights.

Kernel regression weighting vs MLS surface weighting We use kernel regression weights (Figure2) to determine the optimal bandwidths, but use regular MLS weighting to actually perform the second step of polynomial fitting. While our experimental results clearly outperform the popular methods for choosing bandwidths, it is still important to investigate possible extensions of the optimal bandwidth derivations using MLS weighting.

Reconstruction quality It is interesting to reflect on the Mean Squared Error results we have obtained. At first glance, we might expect that for the case of i.i.d. noise there should exist a single h that performs as well as any algorithm. However, neighborhoods of i.i.d. samples are random variables themselves, so they are subject to variability. Then, across any particular model, there will be different optimal bandwidths. The k-nearest neighbor heuristic fails for similar reasons: it is effective at determining the local density of points, but not so at estimating whether the region is densely sampled or noisy. Figure 5 clearly illustrates this. The optimal bandwidth does not go to zero with the noise, which means that as the angle goes to 180, the algorithm is "shifting its focus" from noise to actual sampling density around the circle.

Kernel shapes induced by B We have derived optima for the class of diagonal matrices with positive eigenvalues, but it would be desirable to find a general solution for all symmetric positive-definite matrices. We have solved the equations, but they involve a system of several quadratic equations that we have currently been unable to solve. However, it is easy to circumvent this problem by rotating the coordinate system before computing the optimal bandwidths. This can be done by computing the *unweighted* covariance matrix of a neighborhood around q, and using the matrix of eigenvectors as a coordinate frame transformation. This will align the covariance axes with the coordinate frames, allowing the diagonal matrix to accurately capture the anisotropy. This works in general because all symmetric matrices are diagonalizable.

$\mathbf{h}\mapsto$MISE($\mathbf{h}$) In some cases, we have observed the presence of more than one extrema in the function $h \mapsto \text{MISE}(h)$. While we have always observed a single minimum, we have occasionally seen maxima of the function. While it remains to be investigated under what conditions the function will have multiple extrema, in our experiments, the computed value of h is always extremely close to the minimum.

Bandwidths for the first step Recall that the MLS surface definition is based on a two-step approach. In current practice, the same bandwidths are used for both steps, but there's no particular reason for that decision. Perhaps the most interesting aspect of the results we have encountered is that the optimal bandwidths for the second step are significantly narrower than the bandwidths necessary for finding an appropriate reference frame. This means, essentially, that typical current MLS definitions tend to oversmooth the surfaces. A notable exception is the work of Pauly et al. [17], where Gaussian noise becomes a convolution of the relevant functions with the exact same noise distribution.

Nonlinear kernel regression We have been able to find the optimal bandwidth selector for nonlinear kernel regression in MLS in 2-D, but the derivation of optimal bandwidths for nonlinear kernel regression in 3-D is highly non-trivial [19]. In addition, the bandwidth selector for nonlinear kernel regression requires higher order derivative estimation, the accuracy of which is jeopardized by error propagation of numerical errors. In 2-D, a few steps of iteration is usually necessary to guarantee the precision of reconstruction [24].

Robustness of bandwidth selectors Our bandwidth selector is not integrated with the nonlinear optimization of MLS surfaces. To test the robustness of the method against incorrectly selected reference frames, we rotate the reference plane from 0 to 90 degrees and investigate the behavior of our bandwidth selectors. As shown in Figure 8, the bandwidth selector are capable of selecting bandwidth adaptively according to the local neighborhood up to 45 degrees - the bandwidths change with rotation angles while the reconstruction errors stay at the same level. This means our bandwidth selectors are insensitive to possible errors in the first step. In addition, the curve that plots bandwidth against angle approximates a cosine curve, which agrees with the fact that when the reference plane is rotated by angle θ approximately only $cos(\theta)$ of the original points remain in the

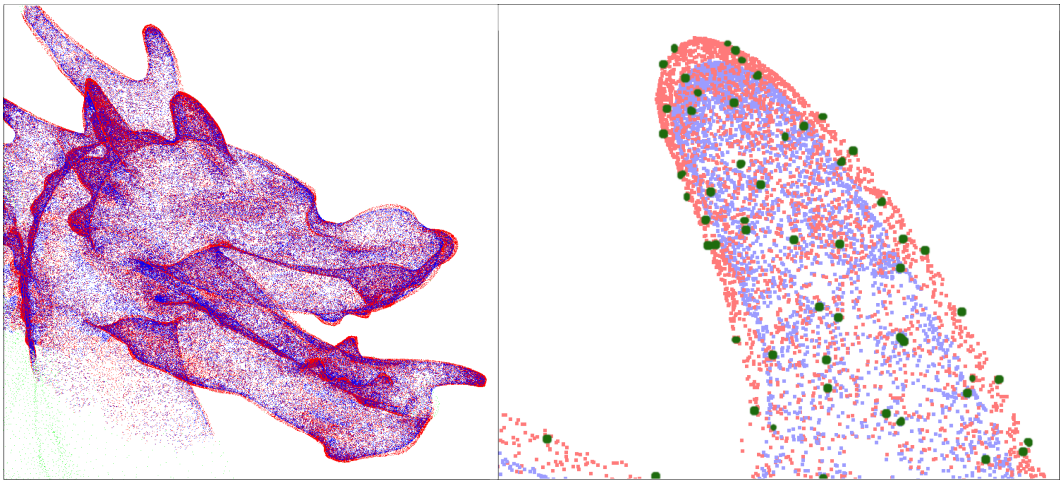

Figure 7. **Dragon head reconstructed from points sampled by afront [20] of the dragon model in the Stanford Scanning Model Repository. Red points are projections of reference points using our MISE-based kernel selector; blue points are projections using a constant bandwidth across the point set. Red points are close to real surface (green points) while the surface composed of blue points contracts.**

neighborhood.

6 Conclusion and Future Work

In this paper, we provide a solid theoretic foundation for the polynomial fitting step in the MLS procedure. We have discussed the possible choices of weight functions and we have proposed algorithms for choosing the optimal parameters for weight functions. We have tested our methods on functionals, surfaces and real world scanning models. Our methods work for clean data as well as for noisy data.

We are currently working on generalizing the kernel selector for higher degree polynomial fitting from 2-D to 3-D. In some cases it might be desirable to settle for numerical optimization, using nonlinear optimizers such as conjugate gradients to explicitly minimize the MISE instead of analytically deriving the optimal value.

References

[1] A. Adamson and M. Alexa. Anisotropic point set surfaces. In *Afrigaph '06: Proceedings of the 4th international conference on Computer graphics, virtual reality, visualisation and interaction in Africa*, pages 7–13, New York, NY, USA, 2006. ACM Press.

[2] M. Alexa and A. Adamson. On normals and projection operators for surfaces defined by point sets. In *Proceedings of Eurographics Symposium on Point-based Graphics*, pages 149–156, 2004.

[3] M. Alexa, J. Behr, D. Cohen-Or, S. Fleishman, D. Levin, and C. Silva. Computing and rendering point set surfaces. *IEEE Transactions on Visualization and Computer Graphics*, 9(1):3–15, 2003.

[4] M. Alexa, J. Behr, D. Cohen-Or, S. Fleishman, D. Levin, and C. T. Silva. Point set surfaces. In *VIS '01: Proceedings of the conference on Visualization '01*, pages 21–28, Washington, DC, USA, 2001. IEEE Computer Society.

[5] N. Amenta and Y. Kil. The domain of a point set surface. In *Eurographics Symposium on Point-Based Graphics*, pages 139–147, 2004.

[6] N. Amenta and Y. J. Kil. Defining point-set surfaces. *ACM Trans. Graph.*, 23(3):264–270, 2004.

[7] D. Cline. Admissibile kernel estimators of a multivariate density. *The Annals of Statistics*, 16(4):1421–1427, 1988.

[8] T. Dey and J. Sun. An adaptive mls surface for reconstruction with guarantees. In *Symposium on Geometry Processing*, pages 43–52, 2005.

[9] S. Fleishman, D. Cohen-Or, and C. Silva. Robust moving least-squares fitting with sharp features. *ACM Transactions on Graphics*, 24(3), 2005.

[10] G. Guennebaud and M. Gross. Algebraic point set surfaces. In *SIGGRAPH '07: ACM SIGGRAPH 2007 papers*, page 23, New York, NY, USA, 2007. ACM.

[11] W. Hardle and J. Marron. Fast and simple scatterplot smoothing. *Computational Statistics & Data Analysis*, 20(1):1–17, 1995.

[12] D. Levin. Mesh-independent surface interpolation. *Geometric Modelling for Scientific Visualization*, 2003.

[13] Y. Lipman, D. Chohen-Or, and D. Levin. Error bounds and optimal neighborhoods for mls approximation. *Eurographics Symposium on Geometry Processing*, pages 71–80, 2006.

[14] C. Mallows. Some comments on c_p. *Technometrics*, 15:661–675, 1973.

[15] The digital michelangelo project. http://graphics.stanford.edu/projects/mich/.

[16] T. Ochotta, C. Scheidegger, J. Schreiner, Y. Lima, R. M. Kirby, and C. Silva. A unified projection operator for mls surfaces. Technical Report UUSCI-2007-007, Scientific Computing and Imaging Institute, University of Utah, 2007.

[17] M. Pauly, N. Mitra, and L. Guibas. Uncertainty and variability in point cloud surface data. In *Proceedings of the Eurographics Symposium on Point-Based Graphics*, 2004.

[18] D. Ruppert, S. Sheather, and M. Wand. An effective bandwidth selector for local least squares regression. *Journal of the American Statistical Association*, 90(432):1257–1270, Dec. 1995.

[19] D. Ruppert and M. Wand. Multivariate locally weighted least squares regression. *The Annals of Statistics*, 22(3):1346–1370, Sept. 1994.

[20] J. Schreiner, C. Scheidegger, and C. Silva. afront. http://afront.sourceforge.net.

[21] W. Schucany. Adaptive bandwidth choice for kernel regression. *Journal of the American Statistical Association*, 90(430), 1995.

[22] S. Sheather. The performance of six popular bandwidth selection methods on some real data sets. *Computational Statistics*, (7):225–250, 1992.

[23] M. Wand and M. Jones. Comparison of smoothing parameterizations in bivariate kernel density estimation. *Journal of the American Statistical Association*, pages 520–528, 1993.

[24] M. Wand and M. Jones. *Kernel Smoothing*, volume 60 of *Monographs on Statistics and Applied Probability*. Chapman & Hall, 1995.

[25] M. Zwicker, M. Pauly, O. Knoll, and M. Gross. Pointshop 3d: an interactive system for point-based surface editing. *ACM Trans. Graph.*, 21(3):322–329, 2002.

A Appendix

In this section, we provide the derivation for optimal bandwidth formulas in 3-D.

A.1 Circular Kernel

By Equation (21) we have:

$$\text{MSE} \simeq \frac{\mu_2(K)^2 h^4}{4}(\frac{\partial^2 f}{\partial {x_1}^2} + \frac{\partial^2 f}{\partial {x_2}^2})^2 + \frac{R(K)v}{nh^2 g(\mathbf{x})} \quad (31)$$

By setting MSE' to be 0 and solving the equation, we obtain the optimal h:

$$h_{opt} = \left(\frac{2R(K)v}{n\mu_2(K)^2(\frac{\partial^2 f}{\partial {x_1}^2} + \frac{\partial^2 f}{\partial {x_2}^2})^2 g(\mathbf{x})}\right)^{1/6} \quad (32)$$

If we choose h based on MISE:

$$\text{MISE} \simeq \frac{\mu_2(K)^2 h^4}{4}\int(\frac{\partial^2 f}{\partial {x_1}^2} + \frac{\partial^2 f}{\partial {x_2}^2})^2 g(\mathbf{x})d\mathbf{x} + \frac{R(K)v}{nh^2} \quad (33)$$

then the optimal h is:

$$h_{opt} = \left(\frac{2R(K)v}{n\mu_2(K)^2\int(\frac{\partial^2 f}{\partial {x_1}^2} + \frac{\partial^2 f}{\partial {x_2}^2})^2 g(\mathbf{x})d\mathbf{x}}\right)^{1/6} \quad (34)$$

A.2 Elliptical Kernel

By Equation (21) we have:

$$\text{MSE} \simeq \frac{\mu_2(K)^2}{4}(h_1^2\frac{\partial^2 f}{\partial {x_1}^2} + h_2^2\frac{\partial^2 f}{\partial {x_2}^2})^2 + \frac{R(K)v}{nh_1h_2 g(\mathbf{x})} \quad (35)$$

To simplify demonstration of our derivation, let:

$$C_1 = \frac{R(K)v}{ng(x)}, C_2 = \mu_2(K)^2, d_1 = \frac{\partial^2 f}{\partial {x_1}^2}, d_2 = \frac{\partial^2 f}{\partial {x_2}^2}$$

then MSE can be rewritten as

$$\text{MSE} = \frac{C_1}{h_1h_2} + \frac{C_2}{4}(d_1h_1^2 + d_2h_2^2)^2 \quad (36)$$

Now we have:

$$\frac{\partial \text{MSE}}{\partial h_1} = -\frac{C_1}{h_1^2h_2} + C_2d_1h_1(d_1h_1^2 + d_2h_2^2) \quad (37)$$

$$\frac{\partial \text{MSE}}{\partial h_2} = -\frac{C_1}{h_1h_2^2} + C_2d_2h_2(d_1h_1^2 + d_2h_2^2) \quad (38)$$

When $d_1d_2 > 0$, by setting both equations to be 0 we get:

$$h_1^2 = \sqrt{\frac{d_2}{d_1}}\left(\frac{C_1}{2C_2d_1d_2}\right)^{1/3} \quad (39)$$

$$h_2^2 = \sqrt{\frac{d_1}{d_2}}\left(\frac{C_1}{2C_2d_1d_2}\right)^{1/3} \quad (40)$$

When $d_1d_2 < 0$, no minimizer exists for MSE: If $d_1(d_1h_1^2 + d_2h_2^2) < 0$, then $\frac{\partial \text{MSE}}{\partial h_1} < 0$ for $\forall h_1$; if $d_1(d_1h_1^2 + d_2h_2^2) > 0$, then $\frac{\partial \text{MSE}}{\partial h_2} < 0$ for $\forall h_2$. In this case, one can either choose to use the circular kernel or the MISE based optimal bandwidth matrix. By analogous analysis, the optimal h_1 and h_2 based on MISE satisfy:

$$h_1^2 = \left(\frac{I_3}{I_1}\right)^{1/4}\left(\frac{R(K)v}{n\mu_2(K)^2(\sqrt{I_1I_3} + I_2)}\right)^{1/3} \quad (41)$$

$$h_2^2 = \left(\frac{I_1}{I_3}\right)^{1/4}\left(\frac{R(K)v}{n\mu_2(K)^2(\sqrt{I_1I_3} + I_2)}\right)^{1/3} \quad (42)$$

where

$$I_1 = \int\left(\frac{\partial^2 f}{\partial {x_1}^2}\right)^2 g(\mathbf{x})d\mathbf{x} \quad (43)$$

$$I_2 = \int\frac{\partial^2 f}{\partial {x_1}^2}\frac{\partial^2 f}{\partial {x_2}^2}g(\mathbf{x})d\mathbf{x} \quad (44)$$

$$I_3 = \int\left(\frac{\partial^2 f}{\partial {x_2}^2}\right)^2 g(\mathbf{x})d\mathbf{x} \quad (45)$$