# Mesh-Independent Surface Interpolation

David Levin

School of Mathematical Sciences, Sackler Faculty of Exact Sciences, Tel Aviv University, Tel Aviv 69978, Israel *levin@post.tau.ac.il*

**Summary.** Smooth interpolation of unstructured surface data is usually achieved by joining local patches, where each patch is an approximation (usually parametric) defined on a local reference domain. A basic mesh-independent projection strategy for general surface interpolation is proposed here. The projection is based upon the 'Moving-Least-Squares' (MLS) approach, and the resulting surface is $C^\infty$ smooth. The projection involves a first stage of defining a local reference domain and a second stage of constructing an MLS approximation with respect to the reference domain. The approach is presented for the general problem of approximating a $(d-1)$-dimensional manifold in $\mathbb{R}^d$, $d \geq 2$. The approach is applicable for interpolating or smoothing curve and surface data, as demonstrated here by some graphical examples.

## 1 Introduction

The problem of interpolating or approximating a function on $\mathbb{R}^d$ using scattered data values has many nice and well established solutions [4], [10]. Some of the methods use meshing strategies, but the preferred methods are those which are mesh-free, such as polynomial approximation, approximations by shifts of radial basis functions [3], and moving least-squares approximations [8].

The situation is quite different in the problem of surface approximation in $\mathbb{R}^d$, given scattered points on a surface. In general, it is not possible to find a natural **global** reference domain, or a parametric domain, which may be used as a base for launching one of the standard global approximation tools. The common practice in this case is to use a collection of **local** reference domains, related to some meshing of the data (e.g. triangulation), and the approximating surface is obtained as the collection of patches defined over those reference domains. The patches are usually defined as piecewise polynomial or piecewise rational parametric patches, smoothly joined together, and, in most cases, the resulting surface depends upon the specific mesh defining the patches. This long established approach works very well in numerous applications, and the Statue of Liberty is a fine example of a surface generated by patches [2].

The main goal of this work is to develop a mesh-independent method for smooth surface interpolation (or approximation) from unstructured scattered data. Such data sets are common in reverse engineering processes, where the surface of a sculptured object is measured by a laser-scanner or by a coordinate measurement machine. A mesh-independent surface approximation may be valuable, for example, serving as a reference surface for comparing different patching approximations of the surface.

To achieve the above goal we present in this work a different paradigm for surface approximation, namely, a projection procedure. This is an approach which seems more complex on the one hand, but, as shown later, may also be considered as simpler and more natural, on the other hand. It is based upon the basic notions of surfaces in differential geometry, namely a local reference system and a local mapping function for each point of the surface. The main tool used here for realizing this approach is based on the moving least-squares idea, which seems appropriate since it uses local approximations.

Let us first recall the definition of the moving least-squares approximation for the case of function approximation [8]. Let $\{x_i\}_{i\in I}$ be a set of distinct data points in $\mathbb{R}^d$, and let $\{f(x_i)\}_{i\in I}$ be some data values at these points. The moving least-squares approximation of degree $m$ at a point $x \in \mathbb{R}^d$ is the value $\tilde{p}(x)$ where $\tilde{p} \in \Pi_m^d$ is minimizing, among all $p \in \Pi_m^d$, the weighted least-squares error

$$\sum_{i\in I}(p(x_i) - f(x_i))^2\theta(\|x - x_i\|) \ . \tag{1.1}$$

Throughout the paper $\theta$ is a non-negative weight function, $\|\cdot\|$ is the Euclidean distance in $\mathbb{R}^d$ and $\Pi_m^d$ is the space of polynomials of total degree $m$ in $\mathbb{R}^d$. The approximation is made local if $\theta(s)$ is rapidly decreasing as $s \to \infty$, or is of finite support, and interpolation is achieved if $\lim_{s\to 0}\theta(s) = \infty$.

The above function approximation method is adopted here for defining a surface approximation strategy which we name MLS (the initials MLS may stand for Moving Least-Squares, or for Moving Local System, or for Meshless Surface). To prepare the presentation of the MLS approach we start in Section 2 with a basic MLS procedure for data smoothing, which is interesting and powerful by itself. Already here it is made clear that the MLS approach is applicable in any dimension in the most natural way. In section 3 we present a modified strategy, resulting is an MLS projection procedure. This procedure is the basis for the smoothing and the interpolation methods defined in Section 4. It is argued that the resulting approximating surfaces are mesh-independent, localized and infinitely smooth. We stress that the purpose of this work is to present the projection idea for the construction of surface approximants. The MLS approach should be viewed as one of the possible tools for implementing this idea. Some theoretical problems concerning the projection idea and the MLS approach remain open at this stage.

## 2 Smoothing Noisy Surface Data

Let $S$ be a $(d-1)$-dimensional manifold in $\mathbb{R}^d$, let $\{r_i\}_{i\in I}$ be points on $S$ or situated near $S$, e.g., points obtained from some measurements of $S$. An interpolating approximation is a manifold passing through $\{r_i\}_{i\in I}$, while a smoothing approximation is a manifold passing 'near' the data points $\{r_i\}_{i\in I}$. Considering the smoothing problem, rather than looking for a smoothing manifold, let us try to approximate the projection of the data points $r_j$, $j \in I$, and of points near the data set, onto $S$. The projection procedure suggested here involves two steps: Given a point $r$ near $S$, we first find a local approximation to $S$ by a hyperplane in $\mathbb{R}^d$. Then we 'project' the point $r$ on a local polynomial approximation of $S$, defined over that hyperplane. Formally, the process is as follows:

**The Basic MLS Procedure:**

**Step 1 – The local approximating hyperplane.** Find a hyperplane $H_r = \{x \mid \langle a, x\rangle - D = 0\ ,\ x \in \mathbb{R}^d\}$, $a \in \mathbb{R}^d$, $\|a\| = 1$, such that the following quantity is minimized,

$$\sum_{i\in I}(\langle a, r_i\rangle - D)^2\theta(\|r_i - r\|)\ , \tag{2.1}$$

where $\langle\cdot,\cdot\rangle$ is the standard inner product in $\mathbb{R}^d$. Since the weights $\{\theta(\|r_i - r\|\}$ decrease as the distance $\|r_i - r\|$ increases, the resulting hyperplane $H_r$ approximates a tangent hyperplane to $S$ near the point $r$. In general, there may be several local minima of (2.1). We choose the one which is the closest to $r$, namely, such that $|\langle a, r\rangle - D|$ is the smallest.

**Step 2 – The approximated projection $\tilde{P}_m$.** Let $\{x_i\}_{i\in I}$ be the orthogonal projections of the points $\{r_i\}_{i\in I}$ onto $H_r$, represented in a specific orthonormal coordinate system on $H_r$, and let $f_i = \langle r_i, a\rangle - D$, $i \in I$, be the heights of the points $\{r_i\}_{i\in I}$ over $H_r$ . Also, let $q$ be the orthogonal projection of $r$ onto $H_r$, and let us choose the origin of the orthonormal coordinate system on $H_r$ to be at $q$. We define a local approximation of degree $m$ to $S$ by a polynomial $\tilde{p} \in \Pi_m^{d-1}$ minimizing, among all $p \in \Pi_m^{d-1}$, the weighted least-squares error

$$\sum_{i\in I}(p(x_i) - f_i)^2\theta(\|r_i - r\|)\ . \tag{2.2}$$

The value $\tilde{p}(0)$ approximates the height of $S$ over $H_r$ at the origin, hence the point $\tilde{r} = q + \tilde{p}(0)a$ is defined to be the approximation of the projection of $r$ on $S$. The result may be denoted in an operator notation as $\tilde{r} = \tilde{P}_m(r)$.

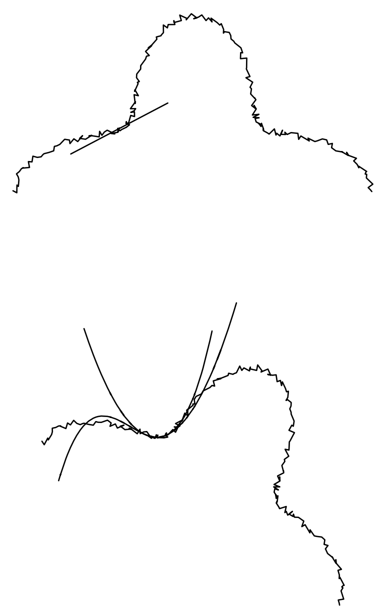
*Remark 1.* The operator $\tilde{P}_m$ defined by the basic MLS procedure is not a projection.

It is essential to note that the distances defining the weights in (2.2) are defined by the distances from the data points $\{r_i\}$, rather than the distances from the points $x_i$ as in (1.1). Also, the distances are taken from the point

$r$ which is not on $H_r$. The last issue plays a central role later on. It actually implies that the above projection step is not really a projection, namely, in general, the projection of $\tilde{r}$ is not going to stay $\tilde{r}$.

*Example 1.* (Curve smoothing) The applications in mind are of course for surfaces in $\mathbb{R}^3$, but the procedure is best understood when applied to the approximation of curves in $\mathbb{R}^2$. In the upper part of Figure 1 we display the data points $\{r_i\}$, drawn with the polygonal line connecting them, and the local approximating line $H_r$ for some data point $r = r_j$. In the lower part the points are rotated so that the $y$- coordinate is in the direction of the local normal to $H_r$, and the local approximating quadratic and cubic polynomial approximations are drawn. In all our examples, the weight function used in the computation of the MLS approximations is $\theta(s) = e^{-s^2/h^2}$ where $h$ is an average separation distance between the data points.

**Fig. 1.** Upper part: the noisy data and a local approximating line. Lower part: the data points rotated to the local normal, and the local 2nd and 3rd degree polynomial approximations.



## 3 The MLS Projection

We recall that the main objective of this work is to develop a method for smooth surface interpolation from unstructured scattered data. As shown later this is achieved by using a proper projection procedure. The basic MLS procedure for approximated projection described above is very effective for thinning data sets. I.e., given a cloud of points representing noisy curve or surface data, the cloud is made thinner by applying the basic MLS procedure to each of the data points. This has already been implemented in [6]. Yet, as explained in Remark 1, the basic MLS procedure does not actually define a projection operator. Another important argument against the basic MLS procedure, related to Remark 1, is the following:

*Remark 2.* The basic MLS procedure in $\mathbb{R}^d$ is not a mapping onto a $d-1$ dimensional manifold.

Instead of proving this statement in general, let us try to explain it and to demonstrate it for the case $d=3$. Let $\{r_i\}$ be data points near a surface $S$ in $\mathbb{R}^3$, and consider the application of the basic MLS procedure for the approximate projection of a point $r$ onto $S$, $\tilde{r} = \tilde{P}_m(r)$. Let $B(s,\delta)$ denote the closed ball of radius $\delta$ centered at $s$, and consider the image of $B(r,\delta)$ under $\tilde{P}_m$:

$$\tilde{P}_m(B(s,\delta)) = \{\tilde{P}_m(r) \mid r \in B(s,\delta)\} \ .$$

The statement in Remark 2 claims that $\tilde{P}_m(B(s,\delta))$ is not a two dimensional manifold. In general it contains interior points. To visualize this consider $S$ to be the $x$-$y$ plane in $\mathbb{R}^3$ and the points $\{r_i\}$ to be $\{r_i\} = \{(i_1 h, i_2 h, (-1)^{i_1+i_2} h)\}_{i=(i_1,i_2)\in\mathbb{Z}^2}$ with $h << \delta$. Also, let $m=0$ and $s=(0,0,0)$. As we project a point $r \in B(s,\delta)$, the weights used for determining the local plane fit are decaying with the distance from $r$. Hence, the best fitted plane is going to be above the $x$-$y$ plane if $r$ is above the $x$-$y$ plane, and vice versa. The same applies to the next step of a local approximation by a constant (m=0). As a result, the image set $\tilde{P}_0(B(s,\delta))$ of the ball $B(s,\delta)$ is going to be lentil-shaped.

Remarks 1 and 2 set the goals in defining a modified MLS procedure for approximated projection in $\mathbb{R}^d$ below; First, is should be a projection, second, it should project the points onto a $d-1$ dimensional manifold.

**The MLS Projection Procedure:**

Given a data set of points $\{r_i\}_{i\in I}$ on a $d-1$ hypersurface $S$ in $\mathbb{R}^d$, or near $S$, and given a point $r$ near $S$, the projection of $r$ with respect to $\{r_i\}_{i\in I}$ is defined as follows:

**Step 1 – The local approximating hyperplane.** Find a hyperplane $H = \{x \mid \langle a, x\rangle - D = 0 \ , \ x \in \mathbb{R}^d\}$, $a \in \mathbb{R}^d$, $\|a\| = 1$, and a point $q$ on $H$, i.e., $\langle a, q\rangle = D$, such that $(r - q) \parallel a$, i.e.,

$$q = r + ta \ , \ t \in \mathbb{R} \ , \tag{3.1}$$

and such that the following quantity is locally minimized,

$$\sum_{i\in I}(\langle a, r_i\rangle - D)^2\theta(\|r_i - q\|) \ . \tag{3.2}$$

Reformulating this, we look for a direction $a \in \mathbb{R}^d$, $\|a\| = 1$, and a finite distance $t \in \mathbb{R}$ such that

$$\sum_{i\in I}\langle a, r_i - r - ta\rangle^2\theta(\|r_i - r - ta\|) \ , \tag{3.3}$$

is locally minimized. Usually, there may exist more than one pair $\{a, t\}$ locally minimizing the above quantity. The pair $\{a, t\}$ is then chosen to be the one with the minimal $|t|$.

For later use we introduce the notation $q = Q(r)$ and $a = A(r)$.

**Step 2 – The MLS projection $P_m$.** Let $\{x_i\}_{i\in I}$ be the orthogonal projections of the points $\{r_i\}_{i\in I}$ onto $H$, represented in an orthonormal coordinate system on $H$ so defined that $r$ is projected to the origin. Also, let $f_i = \langle r_i, a\rangle - D$, $i \in I$, be the heights of the points $\{r_i\}_{i\in I}$ over $H$. Find a polynomial $\tilde{p} \in \Pi_m^{d-1}$ minimizing, among all $p \in \Pi_m^{d-1}$, the weighted least-squares error

$$\sum_{i\in I}(p(x_i) - f_i)^2\theta(\|r_i - q\|) \ . \tag{3.4}$$

The projection of $r$ is defined as

$$P_m(r) \equiv q + \tilde{p}(0)a \ . \tag{3.5}$$

An important observation, related to the projection property, follows directly from (3.3):

**Proposition 1.** *If the pair $\{a, t\}$ minimizes (3.3) for some $r = r^* \in \mathbb{R}^d$, then the pair $\{a, t - s\}$ minimizes (3.3) for $r = r^* + sa$.*

**Proposition 2.** *$Q$ is a projection operator. Furthermore, if there exists a subset $U \subset \mathbb{R}^d$ such that for any $r \in U$ the minimization problem (3.3) has a unique global solution, and $P_m : U \to U$, then $P_m|_U$ is also a projection operator.*

*Proof.* Since $q = r + ta$ then, by Proposition 1, $A(q) = A(r)$ and $Q(q) = Q(r) = q$. Also, note that $P_m(r) = q + \tilde{p}(0)a = r + ta + \tilde{p}(0)a = r + sa$ with

$s = t + \tilde{p}(0)$. Hence, by Proposition 1, and the uniqueness assumption, it follows that

$$A(P_m(r)) = A(r) \ ,$$

and

$$Q(P_m(r)) = P_m(r) + (t - (t + \tilde{p}(0)))a = q + \tilde{p}(0)a - \tilde{p}(0)a = q = Q(r) \ .$$

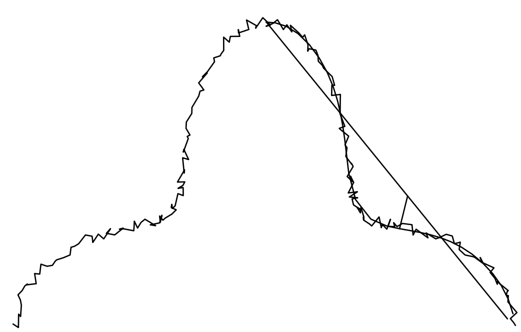
Now, since only $a = A(r)$ and $q = Q(r)$ are used to define $P_m(r)$, the result follows. ∎

The subset $U$ in Proposition 2 is going to be a neighborhood of the hypersurface $S$ to be approximated.

Let us now repeat the previous examples, using the MLS projection.

*Example 2.* (Curve smoothing by the MLS projection) In Figure 2 we depict a noisy data set (the same one used in Example 1), a line segment $L$ near the data set, and its MLS projection $P_2(L)$. The data points are drawn with the polygonal line connecting them, and the projected curve is drawn as the polygonal line connecting the projections of 30 equidistant points on $L$. To visualize the projection, one of the points on the line segment is connected with its projection.

**Fig. 2.** The noisy data, a line segment, and its MLS projection



Let $S$ be a smooth hypersurface in $\mathbb{R}^d$, and let $\{r_i\}_{i \in I}$ be points on $S$. We say that the data set $R = \{r_i\}_{i \in I}$ has a mesh size $h$ if $h$ is the minimal value for which $R \cap B(s, h/2) \neq \emptyset$ for any $s \in S$, where $B(s,\delta) \equiv \{x \mid \|x-s\| \leq \delta\}$. A point $P_m(r)$ defined by the MLS projection is a generic point on the approximating hypersurface defined by the data set $\{r_i\}_{i\in I}$. Based upon Proposition 2, $P_m(r) = P_m(Q(r)) = P_m(Q(q)) = P_m(q)$. For a data set $R = \{r_i\}_{i\in I}$ of a mesh size $h$ on a smooth hypersurface $S$, we introduce the following definition:

**Definition.** (The MLS approximating hypersurface) *For a given data set $R$ of mesh size $h$ let*

$$\tilde{Q} = \{r \mid Q(r) = r \ , \ R \cap B(r,h) \neq \emptyset\} \ . \tag{3.7}$$

*Then the MLS approximating hypersurface induced by the data set $\{r_i\}_{i\in I}$ is defined as*

$$\tilde{S} = \{P_m(r) \mid r \in \tilde{Q}\} \ . \tag{3.8}$$

The above formal definition seems quite strange, and it raises many questions, such as:

- Is $\tilde{S}$ really a $d-1$ dimensional hypersurface?
- How smooth is this hypersurface?
- What is the approximation order?
- How can one reconstruct $\tilde{S}$?

Before we approach some of these questions, let us try to gain some intuition for the above definition. We note that in case the underlying hypersurface $S$ is a hyperplane, then $\tilde{Q} = S$ and $\tilde{S} = S$. Also, for any $r \in \mathbb{R}^d$, $Q(r)$ is just the orthogonal projection of $r$ onto $S$ and $P_m(r) = Q(r)$. If the hypersurface $S$ is smooth, and the mesh size $h$ tends to zero, while the weight function $\theta$ is of finite support of size $O(h)$, then it can be shown that $\tilde{Q} \to S$ linearly in $h$.

Let us consider the system of equations defining the points in $\tilde{Q}$. These are derived by considering the minimization problem (3.3), and with the extra condition that the minimum is attained for $t = 0$. The necessary conditions for a minimum provide $d+2$ equations for the $2d+1$ unknowns which are the components of $a$ and of $r$ and the Lagrange multiplier for the side condition $\|a\| = 1$. A naive counting implies that the solution is a $d-1$ parameter family, and thus $\tilde{Q}$ is expected to be a $d-1$ dimensional manifold.

The issues of smoothness and convergence rate are not analyzed here. Yet, based upon the results in [7] we expect the following:

*Conjecture* (Smoothness and approximation order). Let us assume $S$ is $C^{m+1}$ and that the mesh size $h$ tends to zero, while the weight function $\theta$ is infinitely smooth and is of finite support of size $O(h)$. Then $\tilde{S}$ is a $C^\infty$ surface and $\tilde{S} \to S$ at a rate $O(h^{m+1})$, where $m$ is the degree of the polynomials used in the MLS procedure.

## 4 The MLS Interpolation Scheme and Mesh Independence in $\mathbb{R}^3$

In this section we aim at the two goals set in the title of this paper, namely, interpolation and mesh-independence. As remarked in the Introduction, in the case of function approximation, interpolation is obtained if the weight function is chosen so that $\lim_{s\to 0}\theta(s)=\infty$. Doing just this is not enough for achieving interpolation in the case of surface data. In addition, Step 2 of the MLS projection method should be revised as follows:

Let $\{r_i\}_{i\in I}$, $\{x_i\}_{i\in I}$, $\{f_i\}_{i\in I}$, $q=Q(r)$ be as defined in the MLS projection procedure, and define $q_i=Q(r_i)$, $i\in I$.

In the revised MLS projection procedure the local polynomial approximation $\tilde{p}\in\Pi_m^{d-1}$ is obtained by minimizing the weighted least-squares defined with distances measured from the points $\{q_i\}_{i\in I}$, namely,

$$\sum_{i\in I}(p(x_i)-f_i)^2\theta(\|q_i-q\|)\ . \tag{4.1}$$

The projection of $r$ is thus defined as

$$P_m^*(r)\equiv q+\tilde{p}(0)a\ . \tag{4.2}$$

Assume we are given a data set $R=\{r_i\}_{i\in I}$, of mesh size $h$, of points on a hypersurface $S$ in $\mathbb{R}^d$, and let $\tilde{Q}$ and $\tilde{S}$ be defined by (3.7) and (3.8). If $r\in\tilde{Q}$ then, following Proposition 1, $Q(r+tA(r))=r$ for a small enough $|t|$. This observation induces the following definition:

**Definition.** (Equivalence sets) *The equivalence set $E(r)$ of a point $r\in\tilde{Q}$ is the connected set containing $r$ of all the points of the form $r+tA(r)$ for which $Q$ is uniquely defined and $Q(r+tA(r))=r$. Consequently, the equivalence set $E(\tilde{Q})$ is defined as $E(\tilde{Q})=\cup_{r\in\tilde{Q}}E(r)$.*

**Proposition 3.** (MLS interpolation) *Let $R=\{r_i\}_{i\in I}$ be a data set on a $d-1$ hypersurface $S$ in $\mathbb{R}^d$ and let the MLS projection be defined as above, by (4.2), using (4.1) with a decreasing weight function $\theta$ satisfying $\lim_{s\to 0}\theta(s)=\infty$. Then,*

$$P_m^*(r)=r_j\quad\forall\ r\in E(Q(r_j)),\quad j\in I\ . \tag{4.3}$$

*Proof.* The proof follows directly from the fact that $q=Q(r)=Q(r_j)=q_j$ for any $r\in E(Q(r_j))$. Together with the condition $\lim_{s\to 0}\theta(s)=\infty$, this enforces $\tilde{p}$ to satisfy $\tilde{p}(0)=\tilde{p}(x_j)=f_j$, and thus $P_m^*(r)=q+\tilde{p}(0)a=q_j+f_ja=r_j$. ∎

The other goal set in the title is to achieve a method which is mesh-independent. Here we restrict the discussion to surfaces in $\mathbb{R}^3$. Assume we are given a data set $R=\{r_i\}_{i\in I}$, of mesh size $h$, of points on a surface $S$ in $\mathbb{R}^3$, or near $S$. In practice the MLS approximating surface is to be defined by using some underlying parametrization domain, but the resulting surface is going to be independent of the choice of the parametrization domain.

Consider a proper triangulation $\mathcal{T} = \{T_k\}_{k\in K}$ of the data set $R$. I.e., each $T_k$, $k \in K$, is a triangle in $\mathbb{R}^3$ with vertices in $\{r_i\}_{i\in I}$, and the vertices and edges in $\mathcal{T}$ define a planar graph. We denote the boundary of $\mathcal{T}$ by $\partial\mathcal{T}$, i.e., the set of all edges in $\mathcal{T}$ which belong to one triangle in $\mathcal{T}$ only.

The linear surface defined by a triangulation $\mathcal{T}$ is an approximation of $S$, also denoted by $\mathcal{T}$, and it may serve as a base for defining the MLS approximating surface.

**Definition.** (The MLS approximation $\tilde{S}_{\mathcal{T}}$) *Let $\mathcal{T}$ be a triangulation of $R = \{r_i\}_{i\in I}$ and assume $\mathcal{T} \subset E(\tilde{Q})$. The MLS approximation based upon $\mathcal{T}$ is defined as*

$$\tilde{S}_{\mathcal{T}} = \{P_m(x) \mid x \in \mathcal{T}\} \equiv P_m(\mathcal{T})$$

*i.e., the collection of the MLS projections of all the points on $\mathcal{T}$, with respect to the data set $\{r_i\}_{i\in I}$.*

Practically, we have $\mathcal{T} \subset E(\tilde{Q})$ if there are enough data points where the underlying surface $S$ has high curvature or is nearly self intersecting, and if the edges in $\mathcal{T}$ are short enough. In particular, we assume that $R \not\subset \partial\mathcal{T}$.

**Proposition 4.** (Mesh independence) *Assume $\tilde{S}$ is a two-dimensional submanifold. Let $\mathcal{T}_1$ and $\mathcal{T}_2$ be two simply connected triangulations of $R = \{r_i\}_{i\in I}$ sharing the same boundary, $\partial\mathcal{T}_1 = \partial\mathcal{T}_2$, such that $\mathcal{T}_j \subset E(\tilde{Q})$, and $\forall r \in \partial\mathcal{T}_j$ $E(Q(r)) \cap \mathcal{T}_j = r$, $j = 1,2$. Then $P_m(\mathcal{T}_1) = P_m(\mathcal{T}_2)$.*

*Proof.* Clearly,

$$\tilde{S}_{\mathcal{T}_j} = P_m(\mathcal{T}_j) \subset \tilde{S} \quad , \ j = 1,2 \ ,$$

and

$$P_m(\partial\mathcal{T}_1) = P_m(\partial\mathcal{T}_2) \ .$$

Also we note that the MLS projection is a continuous operator over a triangulation $\mathcal{T}_j$ ,$j = 1,2$. That is, $\|P_m(r_1) - P_m(r_2)\| < \varepsilon$ if $\|r_1 - r_2\| < \delta(\varepsilon)$ and $r_1, r_2 \in \mathcal{T}_j$. Hence, $P_m(\partial\mathcal{T}_j)$ is a simple closed curve in $\mathbb{R}^3$. The condition $E(Q(r)) \cap \mathcal{T}_j = r$ $\forall r \in \partial\mathcal{T}_j$ implies that each point in $P_m(\partial\mathcal{T}_j)$ has a unique source in $\mathcal{T}$. In particular, $P_m(\partial\mathcal{T}_j) \cap P_m(int(\mathcal{T}_j)) = \emptyset$ where $int(\mathcal{T}) \equiv \mathcal{T} \backslash \partial\mathcal{T}$. It also implies that $P_m(\partial\mathcal{T}_j)$ is a simple closed curve on $\tilde{S}$. Given a continuous map on a simply connected domain, which is a one-to-one mapping on the boundary, we know from differential topology [5], that its image is also a simply connected domain. It thus follows that $P_m(T_j)$, $j = 1,2$, are both simply connected domains, sharing the same boundary. Since they also share some interior points, and are both subsets of $\tilde{S}$ which is assumed to be a submanifold, they must be identical. ∎

*Remark 3.* The conditions of Proposition 4 seem quite difficult to verify beforehand, but they can be checked while computing the projection.

*Example 3.* (Surface approximation by the MLS projection) The application of the MLS projection for surface approximation is presented in Figure 3. The data points are depicted by small circles, and in the upper figure we also see a rectangular plane segment $M$ near the data points. This rectangular domain is used as a local parametric base domain for the projection procedure. In the lower figure we see the mesh of points which is the $P_2$ projection of a rectangular mesh on $M$.

**Some computational hints.** Step 1 of the MLS projection procedure involves minimizing the quantity in (3.3), subject to $\|a\| = 1$, and with the additional constraint that $|t|$ is small. This is a non-linear optimization problem, and we have used an iterative method to solve it. To handle the constraints we represent the unknown parameters in terms of unconstrained parameters $u$, $v$ and $w$. The constant $h$ defined the maximal value we allow for $|t|$.

$$a = \big(\cos(u)\cos(v), \sin(u)\cos(v), \sin(v)\big) \ , \quad t = h \cdot \sin(w) \ . \tag{4.4}$$

## 5 Discussion and Conclusions

We have presented a projection approach to surface approximation from unstructured point cloud data. The method is motivated by the MLS approach, where both a local coordinate system and a local polynomial approximation are computed by MLS for the projection of each point. The method has already proved to be useful in practical computer graphics applications for the visualization and simplification of point-sampled surfaces [1] [9]. Both [1] and [9] present strategies for making the ideas presented here more practical, and they also present very nice figures of surface data smoothing by the MLS projection.

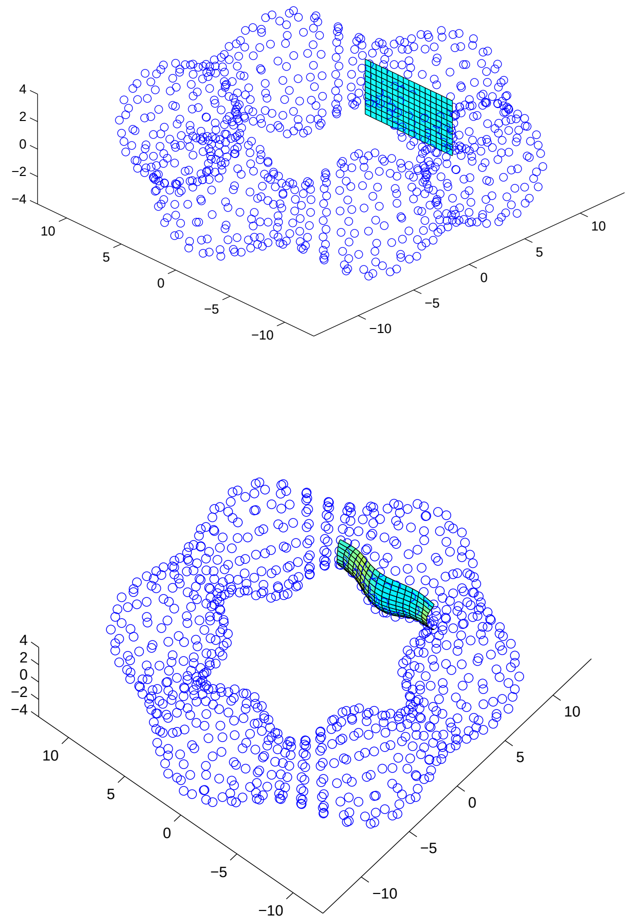
From a theoretical point of view, there is an open question regarding the assumption in Proposition 2 about the existence of a subset $U$ which satisfies certain implicit properties. It is not clear under what circumstances this assumption holds, or how to check it, and this calls for further investigation. Yet, the main contribution of the paper is in presenting the idea of a projection operator for surface reconstruction. The challenge set here is to find other projection operators which may be more robust or easier to compute.

*Acknowledgement.* I thank Ed Nadler for helpful discussions on data smoothing, and I am grateful to my son Adi Levin for his valuable assistance with the computer applications.

## References

1. Alexa, M., Behr, J., Cohen-Or, D., Fleishman, S., Levin, D., Silva, T. (2001): Point set surfaces. IEEE Visualization 2001

**Fig. 3.** Upper part - the data points, and a plane segment $L$ near it. Lower part - the projection $P_2(L)$.





2. Boehm, W., Farin, G., Kahmann, J. (1984): A survey of curve and surface methods in CAGD. Computer Aided Geometric Design, **1**, 10–60
3. Dyn, N. (1987): Interpolation and approximation by radial and related functions. In: Chui, C.K., Schumaker, L.L., Ward, J.D. (eds) Approximation Theory VI. Academic Press, pp. 211–234
4. Franke, R. (1982): Scattered data interpolation: Tests of some methods. Math. Comp., **38**, No. 157, 181–200
5. Hirsch, M.W. (1976): Differential Topology. Springer-Verlag
6. Lee, I.-Q. Curve reconstruction from unorganized points. Computer Aided Geometric Design, to appear

7. Levin, D. (1998): The approximation power of moving least-squares. Math. Comp., **67**, 1517–1531
8. McLain, D.H. (1976): Two dimensional interpolation from random data. The Computer Journal, **19**, 178–181
9. Pauly, M., Gross, M., Kobbelt, L. (2002): Efficient simplification of point-sampled surfaces. IEEE Visualization 2002
10. Powell, M.J.D. A review of methods for multivariate interpolation at scattered data points. Report DAMTP 1996/NA11, University of Cambridge