Implicit function reconstruction in the errors-in-variables context

Levente Hunyadi

Budapest University of Technology and Economics
Department of Automation and Applied Informatics
H-1111 Budapest, Goldmann György tér 3., Hungary
Phone: +36 1 463-2870; Fax: +36 1 463-2871
Email: hunyadi@aut.bme.hu

Abstract: A method is presented that approximates the zero-set of an unknown implicit function based on a set of noisy observations. The algorithm operates by adaptively decomposing the entire domain into partially overlapping subdomains, minimizing misfit over the subdomain and splitting the subdomain if necessary. A consistent estimator is applied to get a parametric polynomial model for a single subdomain, and a partition of unity approach is taken to blend local estimates into a global estimate.

Keywords: function approximation; implicit functions; errors-in-variables; iterative domain subdivision; parametric estimation; partition of unity

1 Introduction

Reconstructing a model from a set of observations is an important problem when one seeks to capture the internal laws that describe a system. In most cases, the model leads to a more compact representation of the relationship between the variables and simplifies a wide range of subsequent operations. For instance, laser scanning or other computer vision techniques produce a large set of (unorganized) data that is cumbersome to manipulate directly. A scanned version of an ellipse consists of thousands of points yet the simple relationship $f(\mathbf{x}_0) = \mathbf{v}_0^\top \mathbf{Q} \mathbf{v}_0 = 0$ with symmetric parameter matrix $\mathbf{Q}$ and $\mathbf{v}_0 = \begin{bmatrix} x_0 & y_0 & z_0 & 1 \end{bmatrix}$ perfectly describes the model. More generally, reconstructing a surface as the zero-set of a scalar-valued implicit function $f(\mathbf{x}_0)$ is desired, which greatly simplifies constructive solid geometry (CSG) operations such

as boolean union, difference and intersection. CSG operations are typically difficult to perform on polygonal meshes as one has to take precautions to ensure the compactness of the resultant surface but are straightforward on an implicit function representation. For instance, let $f_1(\mathbf{x}_0) = 0$ and $f_2(\mathbf{x}_0) = 0$ be implicit function representations of two objects, then $\max(f_1(\mathbf{x}_0),\ f_2(\mathbf{x}_0))$ defines the intersection, $\min(f_1(\mathbf{x}_0),\ f_2(\mathbf{x}_0))$ the union, provided that $f(\mathbf{x}_0) < 0$ is the inside of the object.

Nevertheless, a major problem in faithfully reconstructing an implicit function from acquired data is that data is polluted with measurement noise, possibly to a different degree for each variable. Estimation algorithms that do not take this into account are often biased or inconsistent. Errors-in-variables algorithms, however, specifically target those applications that seek to minimize the error in the reconstructed $f(\mathbf{x}_0) = 0$ even though only noise-contaminated observations $\mathbf{x}$ are at our disposal.

We propose a semi-parametric algorithm based on domain decomposition, local polynomial approximation, and blending of subdomain approximations to construct an implicit function $f(\mathbf{x}_0) = 0$ from the noisy observations $\mathbf{x}$. The algorithm starts with a polynomial function approximation on the domain. If the approximation error exceeds a threshold, the domain is subdivided and local approximations are constructed for each subdomain. This iterative procedure results in an approximation tree with local approximations in leaf nodes, which are merged to get a global approximation with a blending function.

The paper is structured as follows. The problem setting and notational conventions are introduced in Section 2, while Section 3 surveys related work. The proposed reconstruction algorithm is presented in Section 4. Some simulation results are shown in Section 5 before the paper concludes with Section 6.

In the rest of the paper, the following notational conventions are adopted. $\mathbb{R}$ denotes the set of real numbers, $\mathbb{R}^n$ is a column vector of dimension n. x is a scalar variable, $\mathbf{x}$ is a vector and $\mathbf{X}$ denotes a matrix; $f(\mathbf{x})$ is a scalar-valued vector function and $\mathbf{f}(\mathbf{x})$ is a vector-valued vector function. $\mathbb{E}x$ is the expected value of the random variable x. Given the noisy variable x, its (unobservable) noise-free counterpart is x_0, which generalizes to the vector versions x and x_0.

2 Problem setting

Let us consider a static system defined by the (unknown but approximated) implicit function $f : \mathbb{R}^n \to \mathbb{R}$ in n-dimensional space over a compact domain with $f(\mathbf{x}_{0,i}) = 0$ for each underlying data point $\mathbf{x}_{0,i}$ (i.e. each $\mathbf{x}_{0,i}$ is an element of the zero-set of f) where the subscript i in $\mathbf{x}_{0,i}$ denotes the ith observation, $i = 1, \ldots, N$. As the data points are observed with noise, $\mathbf{x}_{0,i} = \mathbf{x}_i - \tilde{\mathbf{x}}_i$ where the (unknown) measurement noise

contribution $\tilde{\mathbf{x}}_i \sim N(0,\ \mathbf{C}_\mathbf{x})$ with $\mathbf{C}_\mathbf{x}$ being the (known) noise covariance matrix:

$$\mathbf{C}_\mathbf{x} = \operatorname{diag}\left(\sigma_\mathbf{x}^2\right) = \mathbb{E}\left(\tilde{\mathbf{x}}\tilde{\mathbf{x}}^\top\right) \approx \frac{1}{N}\sum_{i=1}^{N} \tilde{\mathbf{x}}_i \tilde{\mathbf{x}}_i^\top$$

in which $\sigma_\mathbf{x}^2$ is a vector of variances for each component of $\tilde{\mathbf{x}}$. Introduce a polynomial approximation of f with degree d over a domain $\mathcal{D} \subset \mathbb{R}^n$ such that $p_\mathcal{D} = \theta^\top \mathbf{g}(\mathbf{x})$ where $\mathbf{g}(\mathbf{x})$ is a linearization of $\mathbf{x}$ and $\theta \in \mathbb{R}^{\frac{1}{6}(d+1)(d+2)(d+3)}$ is a parameter vector. For example, the 3D quadric polynomial

$$p_\mathcal{D}(x, y, z) = \sum_{0 \le i+j+k \le d} \theta_{i,j,k} x^i y^j z^k$$

takes the vectorial form $p_\mathcal{D} = \theta^\top \mathbf{g}(\mathbf{x})$ with $\theta \in \mathbb{R}^{10}$ and $\mathbf{g}(\mathbf{x}) : \mathbb{R}^3 \to \mathbb{R}^{10}$ where

$$\mathbf{g}(\mathbf{x}) = \begin{bmatrix} x^2 & y^2 & z^2 & xy & xz & yx & x & y & z & 1 \end{bmatrix}^\top,$$

while for linear approximation $\mathbf{g}(\mathbf{x})$ is basically an identity mapping so that

$$\mathbf{g}(\mathbf{x}) = \begin{bmatrix} \mathbf{x} & 1 \end{bmatrix}^\top.$$

Furthermore, denote with $\mathbf{h}(\mathbf{x})$ the linearization that omits the last, constant term.

As the implicit function f is reconstructed from noisy samples, a perfect fit, i.e. data interpolation is not desired. Instead, a function is approximated from the data that exhibits a misfit. The misfit may either be measured geometrically, i.e. the Euclidean distance of the noise-free data points $\mathbf{x}_{0,i}$ from the approximating function, or algebraically as

$$\left(\theta^\top \mathbf{g}(\mathbf{x})\right)^2$$

in a least-squares sense.

3 Related work

For three-dimensional surface reconstruction from a noisy point cloud, several algorithms have been devised employing Delaunay triangulation [4, 8, 10] but these produce a surface mesh rather than an implicit function that captures the relationship between points. Implicit function reconstruction with radial basis functions (RBFs) [3, 12] and domain decomposition [11] with moderately noisy conditions have also been investigated. A different approach is taken by Zhao et al. [15] who propose a gradient descent method to minimize a combined elastic and potential energy functional.

Guennebaud et al. [5] employ a weighed moving least squares (MLS) approach where the implicit function f is approximated from a compact support region. To evaluate f for any point $\mathbf{x}$ in n-dimensional space, a least-squares problem is solved on the point set within the support region, determined by a support radius R. First, a local parameter vector $\hat{\theta}$ is computed as

$$\hat{\theta}(\mathbf{x}) = \arg\min_{\theta} \left\| \mathbf{W}^{\frac{1}{2}}(\mathbf{x}) \mathbf{Z} \theta \right\|^2 \tag{1}$$

where $\mathbf{W}(\mathbf{x})$ is a diagonal weighing matrix whose entries are computed based on the distance of each point within the support region to the point $\mathbf{x}$ (the entries take a value of zero for points outside the support region), and $\mathbf{Z}$ is a matrix of linearizations of all points within the support region. Second, the estimate is computed as $f(\mathbf{x}) = \hat{\theta}^\top \mathbf{g}(\mathbf{x})$. They fit spheres, i.e. their linearization function is

$$\mathbf{g}(\mathbf{x}) = \begin{bmatrix} x^2 & y^2 & z^2 & x & y & z & 1 \end{bmatrix}^\top.$$

They also present an efficient method for the case when data points are given with normals, as well as estimation of sharp features and normal vector estimation for the approximated implicit surface in the absence of preliminarily specified normals.

As apparent from the necessity to solve a least squares problem for each point $\mathbf{x}$ for which we wish to evaluate the implicit function, the approach is computationally rather intensive. This is especially undesirable when the implicit function representation is to be converted into a polygonal mesh [2] for the sake of presentation. Ohtake et al. [11] propose a computationally less intensive method that decomposes the entire domain into overlapping subdomains, and the least squares problem is solved only for the subdomains. Whenever the implicit function f is to be evaluated, local estimates on the affected subdomains (which contain the point) are blended together using a partition of unity approach to obtain an estimate valid for the entire domain.

Regardless of the different approaches a local neighborhood is used to produce a global estimate, parametric estimation in a local neighborhood involves a least-squares optimization problem. When fitting higher-order surfaces in the errors-in-variables context, the nonlinearity in the model (e.g. due to terms x^2, y^2 and z^2 for sphere fitting) leads to inconsistent estimates for ordinary least-squares and orthogonal least-squares i.e. the estimates do not converge to their true value as the sample size increases [9]. Consequently, local estimators that are consistent [9, 13] are of paramount importance.

4 Reconstruction algorithm

Surface reconstruction algorithms [5, 11], which have been briefly discussed in Section 3, do not distinctively handle noisy data in their local approximation algorithms.

Notably, the least squares objective function (1) treats misfit as it were present in all components of $\mathbf{g}(\mathbf{x})$ (which accounts for a single row of $\mathbf{Z}$) equally, which, however, is truly not the case. Reformulating (1),

$$\hat{\theta}(\mathbf{x}) = \arg\min_{\theta} \left\| \mathbf{W}^{\frac{1}{2}}(\mathbf{x})\mathbf{Z}\theta \right\|^2 = \arg\min_{\theta} \theta^\top \mathbf{Z}^\top \mathbf{W}(\mathbf{x})\mathbf{Z}\theta$$

the least-squares estimates incorporates a weighed covariance matrix $\mathbf{D} = \mathbf{Z}^\top \mathbf{W}(\mathbf{x})\mathbf{Z}$. Investigating the entries of $\mathbf{D}$ for $\mathbf{W} = \mathbf{I}$ and

$$\mathbf{g}(\mathbf{x}) = \begin{bmatrix} x^2 & x & 1 \end{bmatrix}^\top,$$

the self-covariance for the linear component is

$$\mathbb{E}x^2 = \mathbb{E}(x_0 + \tilde{x})^2 = \mathbb{E}(x_0^2 + 2x_0\tilde{x} + \tilde{x}^2) = \mathbb{E}x_0^2 + \sigma_x^2$$

but for a quadratic term is

$$\begin{aligned} \mathbb{E}x^4 &= \mathbb{E}\left(x_0 + \tilde{x}\right)^4 = \mathbb{E}\left(x_0^4 + 4x_0^3\tilde{x} + 6x_0^2\tilde{x}^2 + 4x_0\tilde{x}^3 + \tilde{x}^4\right) \\ &= \mathbb{E}x_0^4 + 6\sigma_x^2\mathbb{E}x_0^2 + 3\sigma_x^4. \end{aligned}$$

Notice how additional noise terms and dependence on unobservable noise-free data appear. In this section, we propose a reconstruction algorithm that takes these discrepancies into account thereby incorporating a consistent local estimator.

4.1 Local approximation

For local approximation over a subdomain $\mathcal{D}$, we employ a parametric estimation method combined with model selection. By selecting a model, we commit ourselves to the particular parameter structure to use, i.e. we choose a linearization $\mathbf{g}(\mathbf{x})$, for which the corresponding parameter vector θ is subsequently sought. For example, for a quadratic model, a linearization

$$\mathbf{g}_{quad}(\mathbf{x}) = \begin{bmatrix} x^2 & y^2 & z^2 & xy & xz & yx & x & y & z & 1 \end{bmatrix}$$

is constructed but for a linear model

$$\mathbf{g}_{lin}(\mathbf{x}) = \begin{bmatrix} x & y & z & 1 \end{bmatrix}.$$

Once a model has been chosen, the problem reduces to a parameter estimation problem that is linear in parameters θ as $f(\mathbf{x}) \approx \theta^\top \mathbf{g}(\mathbf{x})$.

For the sake of local approximation, we apply a "reverse thinking" and estimate parameters over the same data set with multiple model assumptions and choose the particular model that minimizes an error measure. The goal is to avoid overfitting the data,

i.e. to incorporate perturbations into the reconstructed implicit function that are otherwise only attributable to noise. Conventionally, this is performed using cross-validation but the error that measures the bias from the known noise model can also serve as an error measure, as we shall later see in this section.

In order to estimate parameters for a particular linearization, we employ a simple parametric estimation method based on data covariance matrices. For brevity, let $\mathbf{z} = \mathbf{h}(\mathbf{x})$, $\theta = \begin{bmatrix} \bar{\theta} & b \end{bmatrix}$ (where b pairs with the constant in $\mathbf{g}(\mathbf{x})$) and introduce the noisy and noise-free observation sample and noise covariance matrices as

$$\begin{aligned} \mathbf{D} &= \mathbb{E}\left(\mathbf{z}\mathbf{z}^\top\right) \approx \frac{1}{N}\sum_{i=1}^{N}\mathbf{z}_i\mathbf{z}_i^\top \\ \mathbf{D}_0 &= \mathbb{E}\left(\mathbf{z}_0\mathbf{z}_0^\top\right) \approx \frac{1}{N}\sum_{i=1}^{N}\mathbf{z}_{0,i}\mathbf{z}_{0,i}^\top \\ \mathbf{C} &= \mathbb{E}\left(\tilde{\mathbf{z}}\tilde{\mathbf{z}}^\top\right) \approx \frac{1}{N}\sum_{i=1}^{N}\tilde{\mathbf{z}}_i\tilde{\mathbf{z}}_i^\top. \end{aligned} \tag{2}$$

As $\bar{\theta}^\top\mathbf{z}_0 = -b$ for those data points that satisfy the polynomial implicit function approximation and

$$\mathbf{D} = \mathbb{E}\left(\mathbf{z}\mathbf{z}^\top\right) = \mathbb{E}\left(\mathbf{z}_0\mathbf{z}_0^\top + \tilde{\mathbf{z}}\tilde{\mathbf{z}}^\top\right) = \mathbf{D}_0 + \mathbf{C},$$

the (full-rank) sample covariance matrix $\mathbf{D}$ comprising of noisy observations can be decomposed into a (rank-deficient) noise-free component $\mathbf{D}_0$ and a noise component $\mathbf{C}$

$$\theta^\top\mathbf{D}\theta = \theta^\top\mathbf{D}_0\theta + \theta^\top\mathbf{C}\theta = \theta^\top\mathbf{C}\theta$$

in which finding θ entails minimizing the objective function

$$J(\theta) = \theta^\top\mathbf{D}\theta - \theta^\top\mathbf{C}\theta. \tag{3}$$

We get an alternative formulation for $\mathbf{C}$ as compared to (2) if we write its entries as polynomials of terms that are a product of noise "magnitude" and "structure". For instance, let

$$\mathbf{z} = \begin{bmatrix} x^2 & x \end{bmatrix}^\top,$$

for which the covariance of the linear term is

$$C_{lin} = \mathbb{E}x^2 - \mathbb{E}x_0^2 = \sigma_x^2 = \mu\bar{\sigma}_x^2$$

where μ is the noise "magnitude", and of the quadratic term is

$$C_{quad} = \mathbb{E}x^4 - \mathbb{E}x_0^4 = 3\mu^2\bar{\sigma}_x^4 - 6\mu\bar{\sigma}_x^2\mathbb{E}x_0^2.$$

This scheme generalizes to arbitrary entries, such that $\mathbf{C} = \mathbf{C}(\mu)$ is a polynomial in μ with the coefficients being matrices, and is called a nonlinear extension to the Koopmans method by [13]. As a result, (3) can be effectively tackled by solving the eigenvector decomposition problem

$$(\mathbf{D} - \mathbf{C}(\mu))\,\theta = 0$$

where $\mathbf{C}(\mu)$ is a noise covariance matrix known up to a scalar μ (if only linear components are present) or a matrix polynomial of μ (in general). The solutions are the smallest nonzero real eigenvalue and the corresponding eigenvector. If a linear model is fitted to the data, this scheme produces maximum likelihood estimates obtained with

$$\det(\mathbf{D}_0) = \det(\mathbf{D} - \mu\mathbf{C}_x) = 0$$

so that the model parameter vector θ is found by solving a generalized eigenvector problem on the matrix pair $(\mathbf{D},\ \mathbf{C}_x)$. For a polynomial degree $d > 1$, the estimates are still statistically consistent [13] even if not maximum likelihood. In addition, $\mathbf{C} \neq \mathbf{C}_x$ for the higher-order polynomial case and due to the linearization $\mathbf{g}$ involved will depend not only on noise properties but also on noise-free observations. For $d = 2$, the eigenvector problem to solve is

$$\left(\mathbf{D} - \mu\mathbf{C} - \mu^2\mathbf{C}\right)\theta = 0$$

in which $\mathbf{C}$ is a function of the set of noise-free data $\mathbf{Z}_0$. As noise-free data are not available, $\mathbf{C}$ has to be estimated from noisy observations. For example, $\mathbf{C}$ for a quadric in 2D looks as follows:

$$\begin{aligned}
\mathbf{C}(\mu) &= \mu^2 \begin{bmatrix} 3\bar{\sigma}_x^4 & \bar{\sigma}_x^2\bar{\sigma}_y^2 & 0 & 0 & 0 \\ \bar{\sigma}_x^2\bar{\sigma}_y^2 & 3\bar{\sigma}_y^4 & 0 & 0 & 0 \\ 0 & 0 & \bar{\sigma}_x^2\bar{\sigma}_y^2 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix} \\
&- \mu \begin{bmatrix} 6\bar{\sigma}_x^2\bar{x}^2 & \bar{x}^2\bar{\sigma}_y^2 + \bar{\sigma}_x^2\bar{y}^2 & 3\bar{y}\bar{x}\bar{\sigma}_x^2 & 3\bar{x}\bar{\sigma}_x^2 & \bar{y}\bar{\sigma}_x^2 \\ \bar{x}^2\bar{\sigma}_y^2 + \bar{\sigma}_x^2\bar{y}^2 & 6\bar{\sigma}_y^2\bar{y}^2 & 3\bar{x}\bar{y}\bar{\sigma}_y^2 & \bar{x}\bar{\sigma}_y^2 & 3\bar{y}\bar{\sigma}_y^2 \\ 3\bar{y}\bar{x}\bar{\sigma}_x^2 & 3\bar{x}\bar{y}\bar{\sigma}_y^2 & \bar{x}^2\bar{\sigma}_y^2 + \bar{\sigma}_x^2\bar{y}^2 & \bar{y}\bar{\sigma}_x^2 & \bar{x}\bar{\sigma}_y^2 \\ 3\bar{x}\bar{\sigma}_x^2 & \bar{x}\bar{\sigma}_y^2 & \bar{y}\bar{\sigma}_x^2 & \bar{\sigma}_x^2 & 0 \\ \bar{y}\bar{\sigma}_x^2 & 3\bar{y}\bar{\sigma}_y^2 & \bar{x}\bar{\sigma}_y^2 & 0 & \bar{\sigma}_y^2 \end{bmatrix}
\end{aligned}$$

where the notations $\sigma_x^2 = \mu\bar{\sigma}_x^2$, $\sigma_y^2 = \mu\bar{\sigma}_y^2$, $\bar{x} = \frac{1}{N}\sum_{i=1}^{N} x_i$ and $\bar{x}^2 = \frac{1}{N}\sum_{i=1}^{N} x_i^2$ have been used.

Once the eigenvalue problem has been solved, the parameter b for the constant term is found as the mean of the error vector $\mathbf{e} = \mathbf{Z}\theta$.

Insofar, the covariance matrix has been split into "magnitude" and "structure", and the scalar variable μ has been treated as if it were also an unknown, which however is not

the case. Consequently, model validation is possible by comparing the noise magnitude $\hat{\mu}$ as emitted by the estimator given a particular (linear or higher-order) model to the true value. The particular model to use can be found out by minimizing $|\mu - \hat{\mu}|$.

4.2 Global approximation

Local approximation yields a polynomial $p_{\mathcal{D}} = \theta^{\top}\mathbf{g}(\mathbf{x})$ that approximates f over a small subdomain $\mathcal{D}$. In order to reconstruct an f that is valid over the entire domain, a decomposition of the domain into local subdomains as well as a blending of local approximations into a global approximation are necessary. The former we attain with iterative subdivision, and the latter by partition of unity blending. The decomposition algorithm is as follows:

1. Initialize $\mathcal{D} = \mathcal{C}$ where $\mathcal{C}$ is a bounding cube for the entire compact domain of interest. Set the support radius $R = \alpha d$ where d is the diameter of the bounding cube.

2. Test whether the domain contains enough data to obtain a reliable local estimate, i.e. $|\mathcal{D}| \geqq N_{\min}$. If not, enlarge the domain $\mathcal{D}$ accordingly.

3. Compute a local approximation over the possibly enlarged domain.

   (a) If the approximation error exceeds a threshold ε, subdivide the domain into 2^n equal parts where n is the dimensionality of the domain. (For 3D, this is an octree subdivision.) Repeat from step 2 for each subdomain with updated support radius R.

   (b) Otherwise save the estimated parameters θ to be local estimates for the (original) domain $\mathcal{D}$.

The principle of partition of unity blending is that for any $\mathbf{x}$ for which the value of the implicit function f is sought, a weighed estimate $q_{\mathcal{D}} = w_{\mathcal{D}} p_{\mathcal{D}}$ is computed with $w_{\mathcal{D}} \geqq 0$ and $\sum_{\mathcal{D}} w_{\mathcal{D}} = 1$. One possible choice for $w_{\mathcal{D}}$ is to measure

$$\bar{w}_{\mathcal{D}} = b\left(\frac{3\,\|\mathbf{x} - \mathbf{c}_{\mathcal{D}}\|}{2R_{\mathcal{D}}} + \frac{3}{2}\right)$$

where b is the third order B-spline blending function

$$b = \begin{cases} \frac{1}{2}t^2 & 0 \leqslant t \leqslant 1 \\ \frac{1}{2}(-2t^2 + 6t - 3) & 1 \leqslant t \leqslant 2 \\ \frac{1}{2}(t^2 - 6t + 9) & 2 \leqslant t \leqslant 3 \end{cases}$$

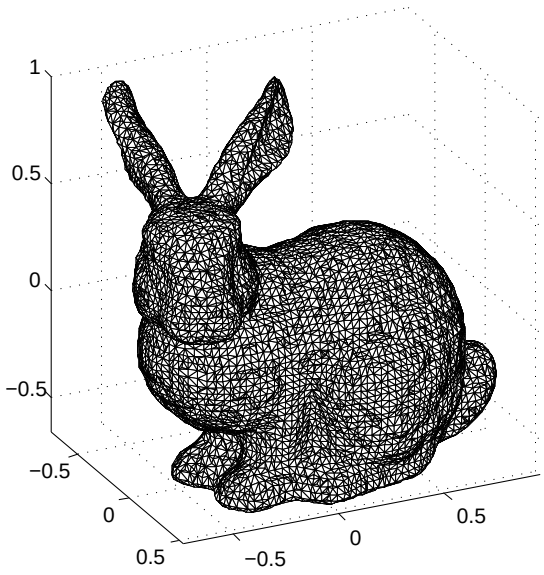

Figure 1: The Stanford bunny model.

or use Wendland's weighing function [14]

$$\bar{w}_{\mathcal{D}} = \left(1 - \frac{\|\mathbf{x} - \mathbf{c}_{\mathcal{D}}\|}{R_{\mathcal{D}}}\right)^4 \left(\frac{4\,\|\mathbf{x} - \mathbf{c}_{\mathcal{D}}\|}{R_{\mathcal{D}}} + 1\right)$$

and set

$$w_{\mathcal{D}} = \frac{\bar{w}_{\mathcal{D}}}{\sum_{\mathcal{D}} \bar{w}_{\mathcal{D}}}.$$

5 Simulation results

We demonstrate the proposed method on the Stanford bunny model, taken from [1] (Figure 1). We chose a model that has 8171 3D points, obtained with a laser scanner from multiple directions and blended into a single model. For the sake of function reconstruction, point normals have been discarded and points have been rescaled to fit into a unit cube, centered around the origin. Noise of equal magnitude $\sigma = 0.01$ has been used as measurement noise in all scaled 3D vector components x, y and z. As the proposed local estimation scheme delivers parameter estimates up to sign (i.e. it makes no decision as to which is the "inside" and the "outside" of the surface), the original

Points used (%)	Sum of absolute error
40	98.5703
60	77.0111
80	60.9651
100	53.7400

Table 1: Consistency of the model reconstruction scheme.

normals have been used to obtain the proper sign but a simple method, such as the one used in [6] could also have been applied. Simulations have been performed with the Fræser errors-in-variables estimation and simulation framework [7], which is written partly in MatLab and (for the sake of improved performance) partly in C and C++.

The consistency of the estimation scheme is illustrated by taking only a specific proportion of the data points into account, constructing a model based on these randomly selected points, and assessing accuracy on the entire noise-free data set. The results are shown in Table 1. The sum of absolute error

$$e = \sum_i |f_r(\mathbf{x}_{0,i})|$$

where $f_r : \mathbb{R}^n \to \mathbb{R}$ is the reconstructed implicit function, shows a decreasing tendency as a higher percentage of noisy data points is utilized.

6 Conclusion

We have proposed a method to reconstruct an unknown implicit function from a set of noisy observations. The method approximates the zero-set of the function in a two-stage process: (1) local estimates are computed with a consistent errors-in-variables estimator over a compact domain, and (2) global estimates are combined from local estimates with distance-based weight function. The method exhibits moderate computational complexity owing to the iterative domain partitioning, yet has a fair accuracy inherited from the local estimator that directly incorporates the errors-in-variables context into the estimation strategy. Future work includes more rigorous comparison of different global estimation strategies that utilize the local errors-in-variables estimator and possible application to dynamic systems.

References

[1] The Stanford 3D Scanning Repository. Available online: `http://graphics.stanford.edu/data/3Dscanrep/`, September 2009.

[2] Jules Bloomenthal. Polygonization of implicit surfaces. *Comput. Aided Geom. Des.*, 5(4):341–355, 1988.

[3] J. C. Carr, R. K. Beatson, J. B. Cherrie, T. J. Mitchell, W. R. Fright, B. C. McCallum, and T. R. Evans. Reconstruction and representation of 3d objects with radial basis functions. In *SIGGRAPH '01: Proceedings of the 28th annual conference on Computer graphics and interactive techniques*, pages 67–76, New York, NY, USA, 2001. ACM.

[4] Tamal K. Dey and Samrat Goswami. Provable surface reconstruction from noisy samples. In *SCG '04: Proceedings of the twentieth annual symposium on Computational geometry*, pages 330–339, New York, NY, USA, 2004. ACM.

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

[6] Hugues Hoppe, Tony DeRose, Tom Duchamp, John Alan McDonald, and Werner Stuetzle. Surface reconstruction from unorganized points. In *SIGGRAPH*, pages 71–78, 1992.

[7] Levente Hunyadi. Fræser: a MatLab framework for estimating errors-in-variables systems. Available online: `http://fraeser.sourceforge.net`, June 2009.

[8] Ravikrishna Kolluri, Jonathan Richard Shewchuk, and James F. O'Brien. Spectral surface reconstruction from noisy point clouds. In *SGP '04: Proceedings of the 2004 Eurographics/ACM SIGGRAPH symposium on Geometry processing*, pages 11–21, New York, NY, USA, 2004. ACM.

[9] Ivan Markovsky, Alexander Kukush, and Sabine Van Huffel. Consistent least squares fitting of ellipsoids. *Numer. Math.*, 98(1):177–194, 2004.

[10] Boris Mederos, Nina Amenta, Luiz Velho, and Luiz Henrique de Figueiredo. Surface reconstruction from noisy point clouds. In *SGP '05: Proceedings of the third Eurographics symposium on Geometry processing*, page 53, Aire-la-Ville, Switzerland, Switzerland, 2005. Eurographics Association.

[11] Yutaka Ohtake, Alexander Belyaev, Marc Alexa, Greg Turk, and Hans-Peter Seidel. Multi-level partition of unity implicits. *ACM Trans. Graph.*, 22(3):463–470, 2003.

[12] Greg Turk and James F. O'Brien. Modelling with implicit surfaces that interpolate. *ACM Trans. Graph.*, 21(4):855–873, 2002.

[13] István Vajk and Jenõ Hetthéssy. Identification of nonlinear errors-in-variables models. *Automatica*, 39:2099–2107, 2003.

[14] Holger Wendland. Computational aspects of radial basis function approximation. In Kurt Jetter, Martin Buhmann, Werner Haussmann, Robert Schaback, and Joachim Stoeckler, editors, *Topics in Multivariate Approximation and Interpolation*, volume 12 of *Studies in Computational Mathematics*, pages 231–256. Elsevier Science Inc., New York, NY, USA, 2005.

[15] Hong-Kai Zhao, Stanley Osher, Barry Merriman, and Myungjoo Kang. Implicit and nonparametric shape reconstruction from unorganized data using a variational level set method. *Computer Vision and Image Understanding*, 80(3):295–314, 2000.