Uncertainty Quantification in Linear Interpolation for Isosurface Extraction

Tushar Athawale and Alireza Entezari, *Member, IEEE*

Abstract—We present a study of linear interpolation when applied to uncertain data. Linear interpolation is a key step for isosurface extraction algorithms, and the uncertainties in the data lead to non-linear variations in the geometry of the extracted isosurface. We present an approach for deriving the probability density function of a random variable modeling the positional uncertainty in the isosurface extraction. When the uncertainty is quantified by a uniform distribution, our approach provides a closed-form characterization of the mentioned random variable. This allows us to derive, in closed form, the expected value as well as the variance of the level-crossing position. While the former quantity is used for constructing a stable isosurface for uncertain data, the latter is used for visualizing the positional uncertainties in the expected isosurface level crossings on the underlying grid.

Index Terms—Uncertainty quantification, linear interpolation, isosurface extraction, marching cubes

1 Introduction

Visualization has become an integral part of data analysis and continues to facilitate discovery and exploration in many disciplines such as scientific simulations, biomedical imaging, geophysics and climate studies, to name a few. As a growing number of application areas employ visualization methods, the issue of uncertainty quantification in visualizations has become one of the important challenges in the field [9, 10]. This challenge presents two distinct problems. Visualization of uncertainty is the problem of illustrating the uncertainties associated with the input data, and uncertainty of visualization refers to the challenge of identifying and quantifying the errors introduced in the visualization pipeline [2].

The uncertainties associated with data arise from the inaccuracies introduced in the measurement or the simulation process. Discretization errors are often due to limited quantization levels and sampling rates (resolution). In remote sensing, specifically in light radar (LIDAR) applications, there are uncertainties about the elevation as well as the location where the elevation was recorded. In fluid simulations, the discretization of the domain and errors accumulated during the time-integration are sources of uncertainty in the data. In magnetic resonance (MR) imaging, the inhomogeneity in the magnetic field introduces uncertainties in the recorded signal.

Errors introduced in each step of the visualization pipeline (e.g., Haber and McNabb model) are propagated throughout the process, and the accumulated errors impact the resulting visualization [2]. The filtering stage of the pipeline (i.e., reconstruction) introduces a level of uncertainty whose magnitude is determined by the mismatch between the interpolation model and the underlying function that was sampled. For example, a nearest-neighbor interpolation provides a piecewise constant interpolation model. The error introduced in the reconstruction stage of the pipeline is determined by the difference between the underlying function from its piecewise constant approximation. Higher-order interpolants provide more accurate models based on the assumption that the underlying function is smooth (i.e., weakly differentiable). This assumption is a key for employing higher-order methods. In fact, higher-order interpolants could make the problem worse, if the underlying function does not have the necessary degree of continuity for the order of the interpolation model. Other sources of uncertainty are introduced in the visualization pipeline as in the processes of (pre or post) classification, mapping and rendering [4, 10].

Errors that occur during data acquisition, simulation, or processing are often characterized by a probability density function (PDF). An accurate characterization of this density function is necessary for a realistic model of these errors. Gaussian distribution is used for modeling uncertainty in many problems in science and engineering [26, 30] , where the variance is known or can be reliably estimated from the physics of the problem. However, certain applications model uniformly distributed uncertainty. A major source of uniformly distributed errors is the discretization (e.g., quantization and sub-sampling) process [34]. Kernel density estimation (aka Parzen-Rosenblatt window method) offers a non-parametric approach for modeling the error density function [32]. In this approach, a kernel such as the uniform, triangular, biweight, Epanechnikov or quartic is associated with each error sample to estimate the shape of the PDF from which the samples were drawn.

Contribution: Our approach builds on the work of Pöthkow and Hege [26] to quantify the positional uncertainty of isosurfaces in indirect visualization. In this paper, we study the behavior of linear interpolation specifically for isosurface extraction from uncertain data with the Marching Cubes algorithm. When the uncertainty in the data is modeled with an additive uniform noise, we present an analytic characterization of the PDFs associated with the positional uncertainties of the vertices of the extracted isosurface. Our analytic characterization also serves as a building block for the analysis, where the uncertainty in the data is obtained, non-parametrically, from kernel density estimation approaches. Using the closed-form PDFs, our approach allows for the analytic computation of the expected position, as well as standard deviation of the positions of vertices of the isosurface. While the former information is used for constructing a stable isosurface, the latter is used to visualize the positional uncertainty of the level crossings on the underlying grid.

2 Related Work

The issue of uncertainty in visualization has been advocated by many as a top challenge [9, 17, 5, 24, 36, 6]. An excellent survey of the state-of-the-art in this subject was gathered in [2]. While there is a large body of research in this area (see references in [2]), in what follows, we briefly discuss the techniques specific to indirect visualization.

There have been many developments in the visualization of uncertainty. In the context of *displaying* isosurfaces, uncertainties can be visualized using color [29] and glyphs [20]. A point-based surface vi-

- *This research is supported by the AFOSR grant FA9550-12-1-0304, NSF grant CCF-1018149, and ONR grant N000141210862.*

- *Tushar Athawale is with the Department of Computer and Information Science and Engineering at the University of Florida. E-mail: athawale@cise.ufl.edu.*
- *Alireza Entezari is with the Department of Computer and Information Science and Engineering at the University of Florida. E-mail: entezari@cise.ufl.edu.*

Manuscript received 31 March 2013; accepted 1 August 2013; posted online 13 October 2013; mailed on 4 October 2013.
For information on obtaining reprints of this article, please send e-mail to: tvcg@computer.org.

sualization approach was proposed in [7] to illustrate the uncertainty of isosurfaces. [4] presents an approach to assess the behavior of various isosurface extraction algorithms – providing a verifiable approach for isosurface visualization [11]. A probabilistic animation was introduced in [16] for the visualization of uncertainty in biomedical volumetric data. Uncertainty in vector field data also poses significant challenges in flow visualization [13, 1, 35], where one needs to visualize the uncertainty of flow, as well as angular uncertainties over time [8]. Schultz et. al [31] not only mapped flow field topology concepts such as sources and sinks to the brain anatomy, but also visualized fuzzy features in the brain diffusion MRI data using probabilistic fiber tracking. Otto et al. [22] proposed a technique to visualize the topological segmentation of an uncertain 2D vector field by finding the probabilities of particles ending up in different topological components. This technique was further extended to an uncertain 3D vector field [23] with some performance improvements. Generally the notion of uncertainty brings a new dimension associated with the data that makes uncertainty-aware visualization a more challenging task.

Due to the highly noisy acquisition process, geophysics and seismic data are often associated with very large uncertainties. Many traditional techniques (e.g., Kriging [33]) have been developed in geoimaging [2] to deal with such levels of uncertainties. Motivated by these methods, Gaussian processes were proposed [30] to model scalar volumetric data. Based on the analysis of the variance of the interpolated function, the authors demonstrate that Gaussian regression is the suitable choice for the interpolation of uncertain data, when data uncertainty is modeled using an additive Gaussian noise [30].

In the context of isosurface extraction, the uncertainty in the data leads to uncertainty in the position, as well as the topology of isosurfaces. A visualization approach was proposed by Pöthkow and Hege [26] to present the positional uncertainty of the extracted isosurfaces from a scalar field. In this approach, the uncertainty is modeled by considering the data as a random field, where a random variable is associated with each grid point. In this work, the concept of *level-crossing probability* (LCP) is introduced, and its value is estimated assuming uncorrelated Gaussian distributed random variables. This approach was further generalized to the case of correlated random fields [28, 25], where the odds of an isosurface passing through cells are estimated from sampling the random variables associated with the vertices of each cell (i.e., a Monte Carlo method). For an accurate estimation, the distributions of random variables associated with a cell need to be sampled heavily, which leads to a significant per-cell cost. In [27], the authors further improve the estimation of LCP by building conditional probabilities and the spanning tree of the vertices of each cell.

In this paper, we examine the interaction of linear interpolation with level-crossing probability. In particular, we demonstrate a closed-form solution to the level-crossing probability when the data uncertainty is modeled by uniform distributions. This framework allows us to quantify and visualize errors caused by issues coming from sampling and quantization, since such errors are uniformly distributed [34].

3 Inverse Linear Interpolation

The isosurface extraction algorithms provide means for visualizing level-sets of multivariate functions by triangulating the set of points in the domain, on which the function attains the isovalue: $S = \{\mathbf{x} \in \mathbb{R}^d \mid f(\mathbf{x}) = c\}$. Often the underlying function is sampled on a Cartesian grid, and a tensor-product linear interpolation (e.g., bilinear, trilinear) is used for interpolation.

In the trivariate setting, the Marching Cubes (MC) algorithm [15] seeks to provide a piecewise linear surface (i.e., a triangular mesh) that is consistent with trilinear interpolation of the data. Since trilinear interpolation coincides with univariate linear interpolation along the edges of the grid, the MC algorithm forms its triangulation by choosing vertices that are on the edges of the grid. This piecewise linear simplification of the trilinear surface leads to topological ambiguities [3] that were addressed by various deciders [21, 19, 14].

The triangulation's vertex positions are computed as an inversion to univariate linear interpolation. Given the samples of the function x_1 and x_2 at the grid points $\mathbf{v}_1$ and $\mathbf{v}_2$, the location of the isosurface vertex, $\mathbf{v}_c$, on an edge connecting $\mathbf{v}_1$ and $\mathbf{v}_2$ is given by:

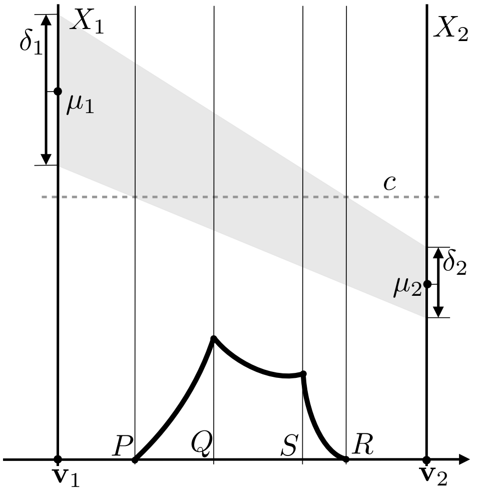


Fig. 1. Inverse linear interpolation provides a distribution for the vertex locations in the Marching Cubes algorithm. The probability density function of locations of the crossing point is determined by the five parameters: $\mu_1, \mu_2, \delta_1, \delta_2$ and the isovalue c.

$$\mathbf{v}_c = (1-z)\mathbf{v}_1 + z\mathbf{v}_2, \quad \text{where} \quad z = \frac{c - x_1}{x_2 - x_1}. \tag{1}$$

The uncertainty associated with the sampled data is modeled by a random variable associated with grid points. In this formulation, the location, Z, is a random variable that depends on X_1 and X_2 – the random variables associated to vertices $\mathbf{v}_1$ and $\mathbf{v}_2$:

$$Z = \frac{c - X_1}{X_2 - X_1}. \tag{2}$$

This inverse linear interpolation leads to a non-linear transformation on the random variables, which makes the characterization of Z difficult. Pöthkow and Hege's histograming approach is based on sampling X_1 and X_2 and estimating the *level-crossing probability* (i.e., $Pr(0 \leq Z \leq 1)$) by the ratio of the number of samples that cut the isovalue to those that do not. A reliable histograming approach requires a large number of samples from X_1 and X_2, which leads to a costly per-cell processing in the MC algorithm. We formulate a solution by deriving the *level-crossing probability density function*, $\mathrm{pdf}_Z(z)$, that fully characterizes the distribution of Z over [0,1]. The distribution of Z is qualitatively related to the isosurface condition analysis performed by Pöthkow and Hege. It shows that the upper bound on the positional uncertainty of the level crossing is inversely proportional to the gradient magnitude for a constant noise amplitude in the data [26].

Fig. 1 shows the random variables X_1 and X_2 and an isovalue c that determine the distribution of the crossing point on the edge between $\mathbf{v}_1$ and $\mathbf{v}_2$. This figure shows the distribution of Z as a piecewise PDF when X_1 and X_2 are uniformly distributed. As discussed in the following section, the pieces of the $\mathrm{pdf}_Z(z)$ are defined based on the order of four quantities P, Q, S, R. In what follows, we demonstrate that a closed-form characterization of $\mathrm{pdf}_Z(z)$ is possible when the data is uniformly distributed. Moreover, when data is not uniformly distributed (e.g., Gaussian), this approach can be employed together with numerical integration to derive the distribution of the crossing point.

We analyze the non-linear transformation in (2) by studying the joint distribution formed by the *dependent* variables $c - X_1$ and $X_2 - X_1$ forming the numerator and denominator of the ratio distribution describing Z. If given a sample from $c - X_1$, the value attained by the random variable $X_2 - X_1$ varies in a range determined solely by X_2. Therefore, when X_2 is independent from X_1, the joint probability density function is obtained by extruding the range of $c - X_1$ by X_2, which is the gray parallelogram shown in Fig. 2. If X_1 and X_2 are not independent, the corresponding joint pdf becomes a trapezoid (or a union of a number of trapezoids).

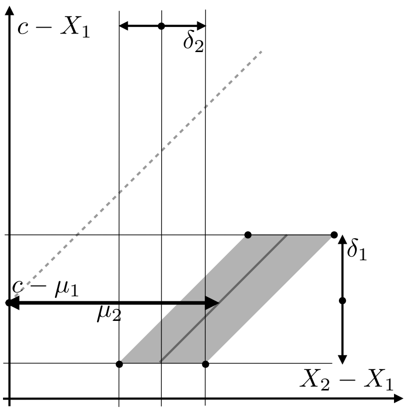


Fig. 2. The joint distribution between the dependent variables $c - X_1$ and $X_2 - X_1$ is the (normalized) indicator function of the parallelepiped formed by extending $c - X_1$ with the X_2 variable.

4 Ratio Distribution

Problem setting: Let μ_i and δ_i represent the mean and the width of the distribution associated with X_i random variables. In other words, X_1 is a uniform distribution over the interval $[\mu_1 - \delta_1, \mu_1 + \delta_1]$. For the notational convenience, we introduce a random variable Z that is a ratio of the random variables Z_1 and Z_2 defined by:

$$Z_1 := c - X_1, \quad Z_2 := X_2 - X_1 \quad \text{and} \quad Z := \frac{Z_1}{Z_2}. \tag{3}$$

When X_1 and X_2 are independent, parameters μ_1, μ_2, δ_1, δ_2 and c identify a unique parallelogram that is the support of the joint distribution of the random variables Z_1 and Z_2, as shown in the Fig. 3. In this setting, Z_1 assumes values in the interval $[c - \mu_1 - \delta_1, c - \mu_1 + \delta_1]$, while Z_2 varies in the interval $[\mu_2 - \mu_1 - \delta_1 - \delta_2, \mu_2 - \mu_1 + \delta_1 + \delta_2]$. After the parallelogram is fixed, we want to find the distribution of the ratio Z. Since we are using inverse linear interpolation to approximate the edge crossing in the case of MC algorithm, we restrict the domain of our ratio distribution to $[0,1]$. When Z assumes values outside of this interval it means that level crossing does not occur within the edge. Therefore, the cumulative distribution of Z within the $[0,1]$ determines the probability of level crossing on the edge between $\mathbf{v}_1$ and $\mathbf{v}_2$. Without any loss of generality, we assume that μ_2 is greater than μ_1. The setup for the ratio distribution is illustrated in Fig. 3 for the case $Z_1 \geq 0$ and $Z_2 \geq 0$, where the joint distribution of Z_1 and Z_2 is represented by a parallelogram $PQRS$ for a particular set of parameters.

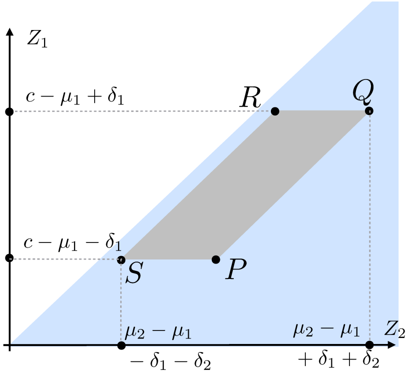


Fig. 3. Ratio distribution Z between two dependent variables Z_1 and Z_2 is supported on a parallelogram. Slopes ranging in $[0,1]$ (indicated in light blue) form the domain of integration for the cumulative distribution of Z.

4.1 Approach

Cumulative distribution of the ratio random variable, Z, is obtained by integration of the parallelogram within the domain of Z:

$$\begin{aligned} \text{cdf}_Z(z) &= Pr(0 \leq Z \leq z) \\ &= Pr(0 \leq Z_1 \leq z \cdot Z_2) + Pr(z \cdot Z_2 \leq Z_1 < 0). \end{aligned} \tag{4}$$

This means that the $\text{cdf}_Z(z)$ is the integral of the parallelogram below the line specified by:

$$Z_1 = z \cdot Z_2. \tag{5}$$

To construct the $\text{cdf}_Z(z)$ from the joint distribution of Z_1 and Z_2, we can sweep from line $Z_1 = 0$ to $Z_1 = z \cdot Z_2$. During the sweep, we visit the vertices of the parallelogram in a particular order depending upon how the parallelogram is set up. Fig. 4 shows a generic set up of the parallelogram with the domain of integration resulting in $0 \leq Z \leq 1$. Now we can find an expression for the area of the parallelogram swept.

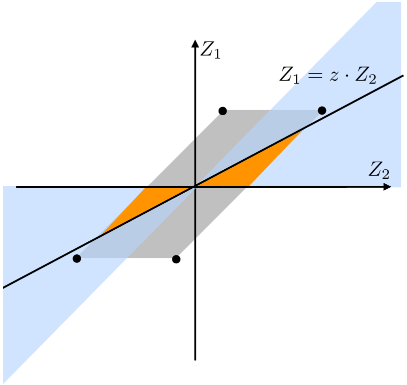


Fig. 4. The cumulative distribution of Z is determined by the overlap of the parallelogram with the domain of integration (indicated in light blue) for $0 \leq Z \leq 1$. The $\text{cdf}_Z(z)$ is the highlighted area under the line $Z_1 = z \cdot Z_2$.

The indicated area is a function of z variable and depends on μ_1, μ_2, δ_1, δ_2 and c, the isovalue. The probability density function, $\text{pdf}_Z(z)$, is obtained by the differentiation of $\text{cdf}_Z(z)$.

The geometry of the parallelogram is determined based on the range of the random variables Z_1 and Z_2. Depending on the (μ, δ) parameters of the uniform distributions, the order between the slopes associated with the vertices of the parallelogram differ. For example, the slope of S can be below or above that of Q in Fig. 3. Moreover, the parallelogram may or may not have an overlap with the domain of integration $0 \leq Z \leq 1$ (light blue region in Fig. 4).

We describe the relationship between Fig. 3 and Fig. 1 to depict our approach. In Fig. 3, as we sweep from $Z_1 = 0$ to $Z_1 = Z_2$, we traverse from $\mathbf{v}_1$ to $\mathbf{v}_2$ in Fig. 1. During the sweep, assume that we visit the parallelogram vertices in the order P, Q, S, R, which corresponds to the points on the edge joining $\mathbf{v}_1$ to $\mathbf{v}_2$. We get different expressions for $\text{cdf}_Z(z)$ between every pair of vertices. Therefore, upon differentiation, we get three different expressions for the $\text{pdf}_Z(z)$, which are plotted in Fig. 1.

Considering the uniform distributions of X_1 and X_2 with their intervals, we can identify three distinct interval-cases that affect the order of vertices in Fig. 3 and the integration in Fig. 4:

1. Non-overlapping intervals: The intervals in which X_1 and X_2 assume their values do not overlap. This is identified by: $\mu_1 + \delta_1 \leq \mu_2 - \delta_2$.
2. Overlapping intervals: This case happens when the intersection of the intervals is non-empty and is smaller than each individual interval. This case is identified by: $\mu_1 - \delta_1 < \mu_2 - \delta_2 \leq \mu_1 + \delta_1 < \mu_2 + \delta_2$.

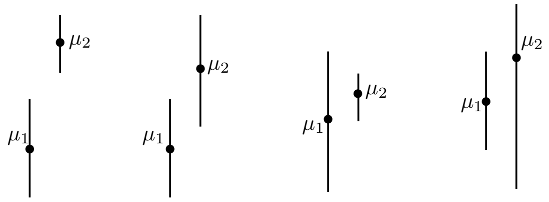


Fig. 5. Various configurations for intervals assuming $\mu_1 < \mu_2$. From left to right: non-overlapping, overlapping, X_2 contained in X_1 and X_1 contained in X_2.

3. Contained intervals: Either one of the intervals is a subset of the other. Since in our presentation we assumed $\mu_1 < \mu_2$, this leads to two distinct sub-cases: when X_2 is contained in X_1, and when X_1 is contained in X_2. While the former case is identified by $\mu_1 - \delta_1 \leq \mu_2 - \delta_2 \leq \mu_2 + \delta_2 \leq \mu_1 + \delta_1$, the latter case is identified by $\mu_2 - \delta_2 \leq \mu_1 - \delta_1 \leq \mu_1 + \delta_1 \leq \mu_2 + \delta_2$.

Fig. 5 depicts these interval configurations . Depending on where the isovalue c lies, the geometry of the parallelogram of the joint distribution between the dependent random variables Z_1 and Z_2 is determined. In each of the four interval-cases, the position of the isovalue c creates three distinct isovalue configurations.

Therefore, there are a total of 12 different configurations that determine the order of vertices in the parallelogram that are visited as a process of ratio Z sweeping from 0 to 1. Once we fix the geometry of the parallelogram, we approach the problem of finding the ratio distribution by first finding the cumulative distribution of the ratio Z, and then differentiating it to obtain the probability density function.

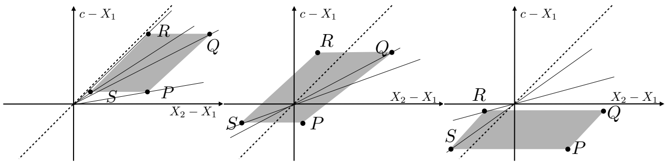
Each of the 12 configurations leads to a $\mathrm{cdf}_Z(z)$ that is a piecewise function in z, whose differentiation provides the distribution $\mathrm{pdf}_Z(z)$. In what follows, we present a single case (out of three possible configurations of isovalue) for each interval case.

4.2 Non-overlapping Intervals

When the intervals do not overlap, there are three distinct isovalue cases:

- $c \leq \mu_1 + \delta_1$
- $\mu_1 + \delta_1 < c \leq \mu_2 - \delta_2$
- $\mu_2 - \delta_2 \leq c$.

Each of the three cases leads to a piecewise function for the $\mathrm{cdf}_Z(z)$ depending upon the order in which the vertices are visited. For the sake of brevity, we present the middle isovalue case in this section. The other isovalue cases can be determined similarly[1]. Supposing that the parameters set up a parallelogram as shown in the left image in Fig. 6, the sweep from $Z_1 = 0$ to $Z_1 = Z_2$ visits the vertices of the parallelogram in the following order: P, Q, S and R. The density function of the random variable Z is a piecewise function consisting of three pieces.



Non-overlapping Overlapping Contained

Fig. 6. The parallelogram setting and vertex visit order represent the pieces of the $\mathrm{cdf}_Z(z)$ for different interval cases. The dotted line represents line $Z_1 = Z_2$. From left to right: Non-overlapping intervals: case $\mu_1 + \delta_1 < c \leq \mu_2 - \delta_2$ with vertex order P,Q,S,R. Overlapping intervals: case $\mu_2 - \delta_2 < c \leq \mu_1 + \delta_1$ with vertex order S,Q. Contained intervals: case $\mu_2 - \delta_2 < c \leq \mu_1 - \delta_1$ with vertex order R,S.

- For z values above the slope of P and below the slope of Q, the $\mathrm{cdf}_Z(z)$ is determined by the area of the triangular portion of the parallelogram swept. With differentiation, we obtain the probability density function:

$$\mathrm{pdf}_Z(z) = \frac{(\mu_2 + \delta_2 - c)^2 z^2 - (\mu_1 + \delta_1 - c)^2 (1-z)^2}{8\delta_1 \delta_2 z^2 (1-z)^2}.$$

- For z values above the slope of Q and below the slope of S, the $\mathrm{cdf}_Z(z)$ is determined by the area of the trapezoidal portion of the parallelogram swept. After differentiating the expression, we get the following piece:

$$\mathrm{pdf}_Z(z) = \frac{1}{2}(c - \mu_1)/(\delta_2 z^2).$$

For the sake of brevity, we detail the derivation for $\mathrm{pdf}_Z(z)$ in this particular case. All other expressions can be derived in a similar manner. Assume a line $Z_1 = zZ_2$, where slope of $Q < z <$ slope of S. We find the intersection points of this line with the edges PS and QR of the parallelogram in the non-overlapping case in Fig. 6. The coordinates for these intersection points are $((c - \mu_1 - \delta_1)/z, c - \mu_1 - \delta_1)$ and $((c - \mu_1 + \delta_1)/z, c - \mu_1 + \delta_1)$ respectively. We need to compute the trapezoidal area below the line $Z_1 = zZ_2$, which represents $\mathrm{cdf}_Z(z)$. The height of this trapezoid is $2\delta_1$, while its bases are of length $((\mu_2 - \mu_1 + \delta_2 - \delta_1) - (c - \mu_1 - \delta_1)/z)$ and $((\mu_2 - \mu_1 + \delta_2 + \delta_1) - (c - \mu_1 + \delta_1)/z)$. The $Pr(Z \leq z) = \mathrm{cdf}_Z(z)$ is obtained by the ratio of the area of the trapezoid and the area of the parallelogram ($4\delta_1\delta_2$) which, after algebraic simplification, is given by:

$$\mathrm{cdf}_Z(z) = \frac{\mu_2 - \mu_1 + \delta_2 - \frac{c - \mu_1}{z}}{2\delta_2}.$$

We then obtain the $\mathrm{pdf}_Z(z)$, as indicated above, after differentiation.

- For z values above the slope of S and below the slope of R, the $\mathrm{cdf}_Z(z)$ is the area of the triangular portion of the parallelogram not covered by sweeping up to S. After differentiating, we get the following piece:

$$\mathrm{pdf}_Z(z) = \frac{-(\mu_2 - \delta_2 - c)^2 z^2 + (\mu_1 - \delta_1 - c)^2 (1-z)^2}{8\delta_1 \delta_2 z^2 (1-z)^2}.$$

For example, where $\mu_1 = 3$, $\delta_1 = 3$, $\mu_2 = 12$, $\delta_2 = 4$, and $c = 7.8$, the ratio distribution has a piecewise probability density function as shown in the left figure in Fig. 7.

4.3 Overlapping Intervals

When the intervals overlap, there are three distinct isovalue cases:

- $c \leq \mu_2 - \delta_2$
- $\mu_2 - \delta_2 < c \leq \mu_1 + \delta_1$
- $\mu_1 + \delta_1 \leq c$.

[1] The tabulation of all the cases is included in the appendix in the supplementary material

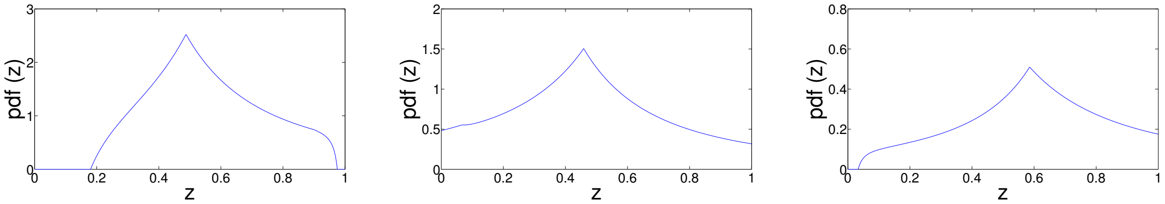


Fig. 7. Exemplar probability distribution functions, limited to the domain [0,1], for various interval cases. From left to right: non-overlapping ($\mu_1 = 3$, $\delta_1 = 3$, $\mu_2 = 12$, $\delta_2 = 4$, and $c = 7.8$), overlapping ($\mu_1 = 5$, $\delta_1 = 4$, $\mu_2 = 12$, $\delta_2 = 6$, and $c = 8.8$) and contained intervals ($\mu_1 = 7$, $\delta_1 = 2$, $\mu_2 = 8$, $\delta_2 = 6$, and $c = 4.9$).

Again, each of the three cases leads to a piecewise function for the $\text{cdf}_Z(z)$ depending upon the order in which the vertices are visited. Let us assume the case when $\mu_2 - \delta_2 < c \leq \mu_1 + \delta_1$. When the parameters set up the parallelogram as shown in the middle image in Fig. 6, the sweep from $Z_1 = 0$ to $Z_1 = Z_2$ visits the vertices of the parallelogram in the following order: S and Q. The density function of the random variable Z is a piecewise function consisting of three pieces.

- For z values above the slope of 0 and below the slope of S, the $\text{cdf}_Z(z)$ is determined by the area of portion of the parallelogram swept in the first and the third quadrant. With differentiation, we obtain the probability density function:

$$\text{pdf}_Z(z) = \frac{(c-\mu_2)^2 + \delta_2^2}{4(1-z)^2\delta_1\delta_2}.$$

- For z values above the slope of S and below the slope of Q, the $\text{cdf}_Z(z)$ is determined by the area of the portion of parallelogram swept. After differentiating the expression, we get the following piece:

$$\begin{aligned}\text{pdf}_Z(z) =& \frac{az^2 + b(1-z)^2}{8\delta_1\delta_2 z^2(1-z)^2}, \quad \text{where}\\ a =& (\mu_2 + \delta_2 - c)^2,\\ b =& (\mu_1 + \delta_1 - c)^2.\end{aligned}$$

- For z values above the slope of Q and below the slope of 1, the $\text{cdf}_Z(z)$ is the area of the portion of the parallelogram swept. After differentiating, we get the following piece:

$$\text{pdf}_Z(z) = \frac{(c-\mu_1)^2 + \delta_1^2}{4z^2\delta_1\delta_2}.$$

An example for $\mu_1 = 5$, $\delta_1 = 4$, $\mu_2 = 12$, $\delta_2 = 6$, and $c = 8.8$ leads to a piecewise function as the probability density function shown in the middle figure in Fig. 7.

4.4 Contained Intervals

Assume the case where X_1 is contained in X_2. When the one interval is contained in the other one, there are three distinct isovalue cases:

- $c \leq \mu_1 - \delta_1$
- $\mu_1 - \delta_1 < c \leq \mu_1 + \delta_1$
- $\mu_1 + \delta_1 \leq c$.

Again, each of the three cases leads to a piecewise function for the $\text{cdf}_Z(z)$ depending upon the order in which vertices are visited. Let us assume the case when $\mu_2 - \delta_2 < c \leq \mu_1 - \delta_1$. When the parameters form the parallelogram as shown in the right image in Fig. 6, the sweep from $Z_1 = 0$ to $Z_1 = Z_2$ visits the vertices of the parallelogram in the following order: R, S . The density function of the random variable Z is a piecewise function consisting of two pieces.

- For z values above the slope of 0 and below the slope of R, there is no portion of the parallelogram. For z values above the slope of R and below the slope of S, the $\text{cdf}_Z(z)$ is determined by the area of portion of the parallelogram swept in the third quadrant. With differentiation, we obtain the probability density function:

$$\begin{aligned}\text{pdf}_Z(z) =& \frac{az^2 - b(1-z)^2}{8\delta_1\delta_2 z^2(1-z)^2}, \quad \text{where}\\ a =& (\mu_2 - \delta_2 - c)^2,\\ b =& (\mu_1 - \delta_1 - c)^2.\end{aligned}$$

- For z values above the slope of S and below the slope of 1, the $\text{cdf}_Z(z)$ is determined by the area of portion of the parallelogram swept in the third quadrant. With differentiation, we obtain the probability density function:

$$\text{pdf}_Z(z) = \frac{1}{2}(\mu_1 - c)/(\delta_2 z^2).$$

An example in this case is illustrated in the right figure of Fig. 7, where $\mu_1 = 7$, $\delta_1 = 2$, $\mu_2 = 8$, $\delta_2 = 6$, and $c = 4.9$.

The analysis in this section can be considered as an extension of interval arithmetics for handling linear interpolation and its inverse – specifically for the ratio distribution.

While the uniform distribution leads to a constant function defined over the parallelogram, the Parzen-Rosenblatt window method uses kernels (e.g., triangular, biweight, triweight, or Epanechnikov) that introduce polynomial functions over the parallelogram. The analysis in this section can be extended to such windows, where instead of area calculation, one needs to integrate polynomials over parts of the parallelogram which is possible in closed form. Therefore, the approach in this section can be extended to handle uncertainty, where the noise in the data is estimated using the Parzen window method (i.e., kernel density estimation) [32].

4.5 Cell Crossing Probabilities

Pöthkow et al. volume rendered the maximum edge crossing probability for each grid cell to visualize the lower bound on the cell crossing probabilities [27]. In our framework, we can find the edge crossing probability using the following equation:

$$Pr_{\text{EdgeCrossing}} = PEC = \int_0^1 \text{pdf}_Z(z)\,dz.$$

Thus, we can similarly compute the maximum edge crossing probability per cell in the current framework. Since we have assumed an independent uniform noise model, the cell crossing probability can also be computed in closed form. So instead of rendering maximum edge crossing probability, we may directly render the cell crossing probability:

$$\begin{aligned}Pr_{\text{CellCrossing}}(c) = 1 &- Pr(X_1 > c, \ldots, X_8 > c)\\ &- Pr(X_1 < c, \ldots, X_8 < c),\end{aligned}$$

where $X_1, \ldots, X_8$ represent random variables associated with vertices of a cell. Under the assumption of independence, the joint densities above factor into the products $\prod_{i=1}^{8} P(X_i > c)$ and $\prod_{i=1}^{8} P(X_i < c)$, respectively. Each one of these probabilities are easily computed as:

$$Pr(X_i > c) = (\mu_i + \delta_i - c)/(2\delta_i)$$
$$Pr(X_i < c) = (c - (\mu_i - \delta_i))/(2\delta_i).$$

4.6 Distribution Properties

Once we have formulae for the distribution of the crossing random variable, Z, we can extract useful properties of the distribution such as expected value and variance. The ratio distribution is renormalized over the range [0,1] for the expected value and variance computations. We find the expected value of the ratio Z using the following formula:

$$E(Z) = \frac{\int_0^1 z \cdot \mathrm{pdf}_Z(z)\,\mathrm{d}z}{PEC}.$$

Expected value computation is essentially an indicator of the expected crossing point on the edge if the values at the endpoints of the edge have the uniform noise. We shall use the terms expected value of ratio Z and the 'expected crossing' interchangeably in the rest of the paper. Similarly, we calculate the variance as:

$$var(Z) = \frac{\int_0^1 z^2 \cdot \mathrm{pdf}_Z(z)\,\mathrm{d}z}{PEC} - (\frac{E(Z)}{PEC})^2.$$

Both the expected value and variance computation are possible in closed form (using a symbolic computation package such as Maple) using the $\mathrm{pdf}_Z(z)$ derived in the previous section. These quantities are determined based on the five parameters that determine the pdf_Z: $\mu_1, \mu_2, \delta_1, \delta_2$ and the isovalue c. Variance value computation quantifies the spatial uncertainty of expected crossing. In the experiments section, we explain how we exploit these properties for the uncertainty quantification in the sample data with uniform noise.

5 Experiments

Obtaining the distribution of the crossing points on the edges of the underlying grid allows us to integrate our approach to the Marching Cubes algorithm [2]. Our experiments compare the isosurfaces extracted when position of vertices of the triangulation is determined by (inverse) linear interpolation of the sample data versus the expected position of level crossing in the mean field. While the (inverse) linear interpolation (the former) approach only considers the sample data, our approach (the latter) considers the uniform distribution for the determination of the expected location of level crossings in the mean field. Moreover, the variance of the random variable modeling the position of level crossing gives us a way of visualizing the uncertainties associated with the position of level crossing.

We examined synthetic and real datasets in our experiments. The experiments were carried out for the extraction of candidate isovalues from datasets. Our first experiment considers the case when we inject noise (with uniform distribution) to the teardrop function [12]. Similar to the approaches discussed in [28, 34], we estimate μ and δ pairs for each data point. For each cell of the grid, we estimate the sample mean at each cell corner by simply taking the mean of the samples of the neighboring cells including that cell. Also we estimate the interval at each cell corner by computing the sample variance from the estimated mean. The left image in Fig. 8 shows the isosurface extracted from the ground truth for the isovalue of $c = -0.002$, while the second image uses inverse linear interpolation to find the position of level crossing on each edge of the grid. The third column in Fig. 8 shows the 'expected surface', whose vertex positions are determined by the expected position of level crossing (i.e., $E(Z)$). The fourth image shows the variance of the distribution for the positions of level crossing, $var(Z)$, color mapped (after histogram equalization) between highest (red) to mid (blue) and lowest (green) variances. The noise level that was injected in this experiment was $\delta = 0.01$. The first three images were rendered with flat shading to emphasize the geometry in the extracted isosurface, while the uncertainty image (the rightmost) is rendered with Gouraud shading, since the uncertainties are determined for the vertices of the triangulation. This smooth shading of the mesh visualizes the uncertainties in the level crossings throughout the extracted isosurface.

For our second experiment we used the Marschner-Lobb dataset [18]. The first image (from the left) in Fig. 10 shows the isosurface extracted from the ground truth for the isovalue of $c = 0.5$. The ground truth was injected with uniform noise of $\delta = 0.03$. The second image shows the result of direct linear interpolation used for extraction of vertex positions, while the third image uses the expected value of level crossing (i.e., $E(Z)$). The last image visualizes the uncertainties in the positions of vertices (i.e., level-crossing location) by color mapping the variance, $var(Z)$, (after histogram equalization) to red (highest), blue (mid) and green (lowest values). Similar to the previous study, flat shading was used for the first three images, while smooth shading was used to visualize the positional uncertainty of expected isosurface level crossings on the grid edges.

These experiments demonstrate that the isosurface extracted with the expected position of level crossing is more resilient to the noise level. At a high noise level as in Fig. 8, the reconstruction surface in linear interpolation breaks down as we can see a lot of unwanted geometry in the second image (from left), enlarging the thin region of the groundtruth teardrop dataset. However, the expected surface shows some stability in the structure, i.e., the thin region of the groundtruth teardrop dataset remains fairly thin in the expected surface reconstruction. When the noise level is significantly smaller than the isovalue, the effect of the expected surface computation is minimal. However, when the noise magnitude is high relative to the isovalue, it is easy to see the effects of expected level-crossing computations, which provide more stable isosurfaces as illustrated in Fig. 8 and Fig. 10. Generally, the distribution of errors is affected by the smoothness/frequency content of the signal. In smooth areas of the signal, the presence of green color confirms the accuracy of the surface location, and in "high-frequency" areas, the appearance of blue/red colors indicate the uncertainty of the location of isosurface. The Marschner-Lobb dataset is exhibiting aliasing on the diagonal directions, the natural pattern that shows up when sampled on the Cartesian lattice. These rings (aliasing due to critical sampling rate at the resolution of $41 \times 41 \times 41$) are documented in the original Marschner-Lobb paper [18]. This is the reason that the green areas are appearing on the X and Y axes in Fig. 10, where aliasing is not exhibited as much.

Our third experiment is on a fuel simulation dataset[3]. The first image in Fig. 9 (from the top) shows the ground truth surface corresponding to the isovalue of $c = 56$. When we increase the uncertainty in the data by injecting higher magnitudes of noise, the advantage of the expected crossing approach become clear. The second image in Fig. 9 (from the top) is the result of linear interpolation, when the magnitude of uncertainty is $\delta = 21$. The third image is the isosurface extracted by the expected level-crossing approach, and the fourth image shows the positional uncertainty of the vertices from triangulation in MC, found using expected crossing, by color mapping the variance of the level-crossing random variable, $var(Z)$. We can see some axis-aligned artifacts in the bottom image of Fig. 9. These are likely due to the axis-aligned features of the fuel dataset itself. The cylindrical structure appears to be reconstructed as a cylinder with an eight-sided (octagonal) base. The other end of the cylinder is exhibiting blue/red surfaces with high uncertainties without a particular axis-aligned orientation.

Similar results were observed on the bonsai tree dataset in Fig. 11 with an isovalue of $c = 84$, when noise of magnitude $\delta = 35$ is injected in the dataset. Fig. 11 in the third row provides a zoomed-in view of the branches of bonsai tree in this experiment. We can see in the zoomed-in view that branches break down in the case of linear interpolation, whereas they survive in the case of expected level crossing.

In our second set of experiments, we use expected surface compu-

[2] The ijkmcube package is courtesy of http://www.cse.ohio-state.edu/w̃enger/research/vis4d/documentation/ijkmcube.html.

[3] Datasets are courtesy of volvis.org.

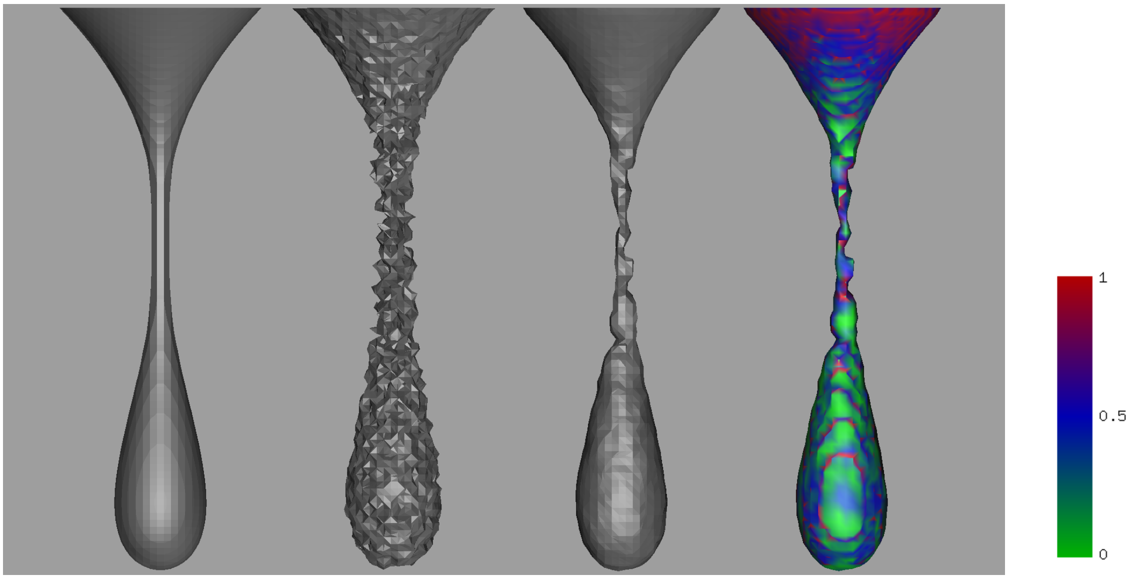


Fig. 8. Teardrop dataset visualized at the isovalue of $c = -0.002$. The leftmost image shows the result of linear interpolation in the original sampled data, which serves as the ground truth. Uniform noise of $\delta = 0.01$ is injected in the dataset, and linear interpolation is shown in the second image. The isosurface extracted using expected crossing is shown in the third image, which provides a more stable isosurface when noise level is high. The rightmost image visualizes the positional uncertainty of the vertices from triangulation in MC, found using expected crossing, by color mapping the ratio variance, $var(Z)$. Green indicates small variance, blue indicates moderate, while red indicates high variance; hence, high spatial uncertainty (quantiles after histogram equalization).

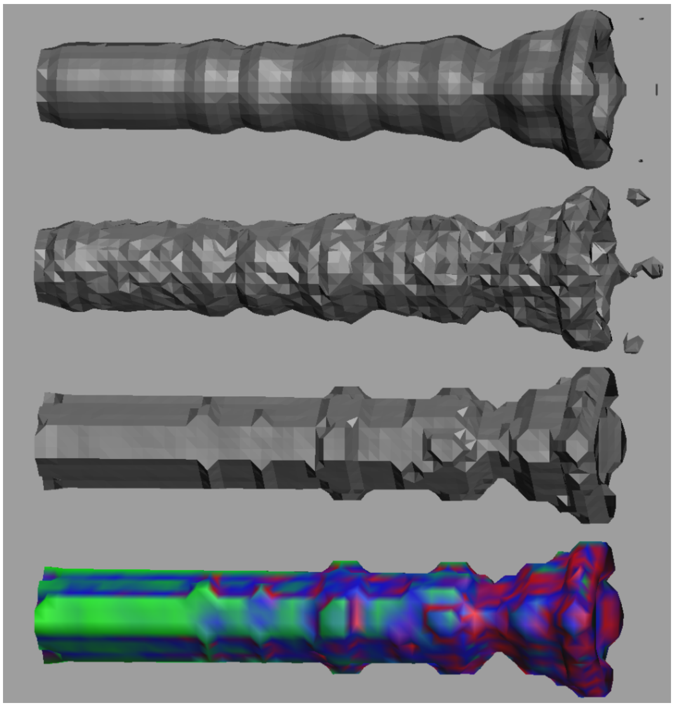


Fig. 9. Fuel dataset visualized at the isovalue of $c = 56$. The image on the top shows the result of linear interpolation in the original sampled data, which we consider as the ground truth. Uniform noise of $\delta = 21$ is injected in the dataset, and again the result of linear interpolation is shown in the next image. Isosurface extracted using expected crossing is shown in the third image. The bottom image visualizes the positional uncertainty of expected level crossing on the underlying grid by color mapping the ratio variance, $var(Z)$. Green indicates small variance, blue indicates moderate, while red indicates high variance; hence, high spatial uncertainty (quantiles formed after histogram equalization).

tations to visualize stable structures in various datasets by considering their local variations *without* introducing artificial noise. In these experiments, a local averaging is used to estimate the mean function and variation functions at each point. The top image in Fig. 12 is linear interpolation of the mean field estimated in the fuel dataset corresponding to the isovalue of $c = 56$. We assume this to be the ground truth. The expected surface computation provides stable structures in the underlying dataset obtained by smoothing out the original data. The second image in Fig. 12 is the expected isosurface corresponding to this isovalue after the smoothing operation when considering the estimated local variations (δ) field. The regions that are not smoothed out indicate parts of the isosurface that are stable and unaffected by the smoothing. Smoothed out areas in the expected surface indicate spatial locations, where the isosurface varies considerably. Third image in Fig. 12 visualizes stability of the linear interpolation level crossings by color mapping the distance between the vertices determined by linear interpolation (of mean) in the ground-truth dataset, and the vertices determined by expected level crossing as a result of the uncertainties in the data. These distances are color mapped after a histogram equalization step. Green color indicates the parts of the linear interpolation isosurface that are relatively stable, blue indicates parts of the isosurface that are moderately stable, and red indicates parts of the isosurface that are relatively unstable. Fig. 13 shows a similar study for the bonsai tree dataset with an isovalue of $c = 84$.

Our framework lends itself to an explicit computation of the cell crossing probabilities in MC. Fig. 14 shows a volume rendering of the probabilities computed in our framework for each cell of the grid for the engine and the fuel dataset. Isovalue of $c = 58$ is used in the engine dataset, and isovalue of $c = 56$ is used in the fuel dataset. Our color mapping uses green to indicate high cell crossing probability, blue to indicate moderate cell crossing probability, and red to indicate low cell crossing probability.

6 Conclusion & Future Work

We present an approach for studying the interaction between linear interpolation and data uncertainty quantified by the uniform distribution. Our approach includes a closed-form solution for the probability distribution of positions of the level crossing determined by inverse linear interpolation. With the probability density function, we compute

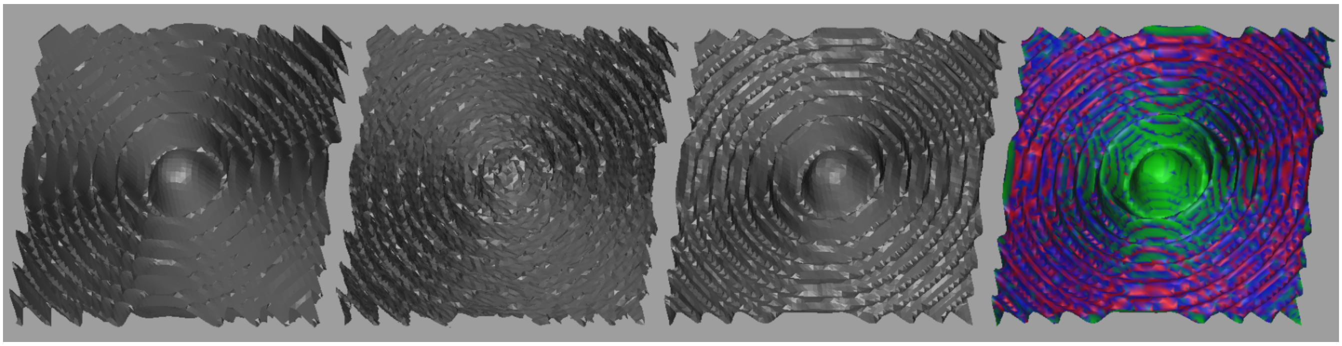


Fig. 10. Marschner-Lobb dataset visualized at the isovalue of $c = 0.5$. The leftmost image shows the result of linear interpolation in the original sampled data, which we consider as the ground truth. Uniform noise of $\delta = 0.03$ is injected in the dataset, and the result of linear interpolation is shown in the second image. The isosurface extracted using expected crossing is shown in the next image. Geometry of the mesh reconstruction tells us that expected crossing does relatively better reconstruction in a noisy dataset. The rightmost image visualizes the positional uncertainty of the vertices from triangulation in MC, found using expected crossing, by color mapping the ratio variance. Green indicates small variance, blue indicates moderate, while red indicates high variance; hence, the location of high spatial uncertainty (quantiles after histogram equalization).

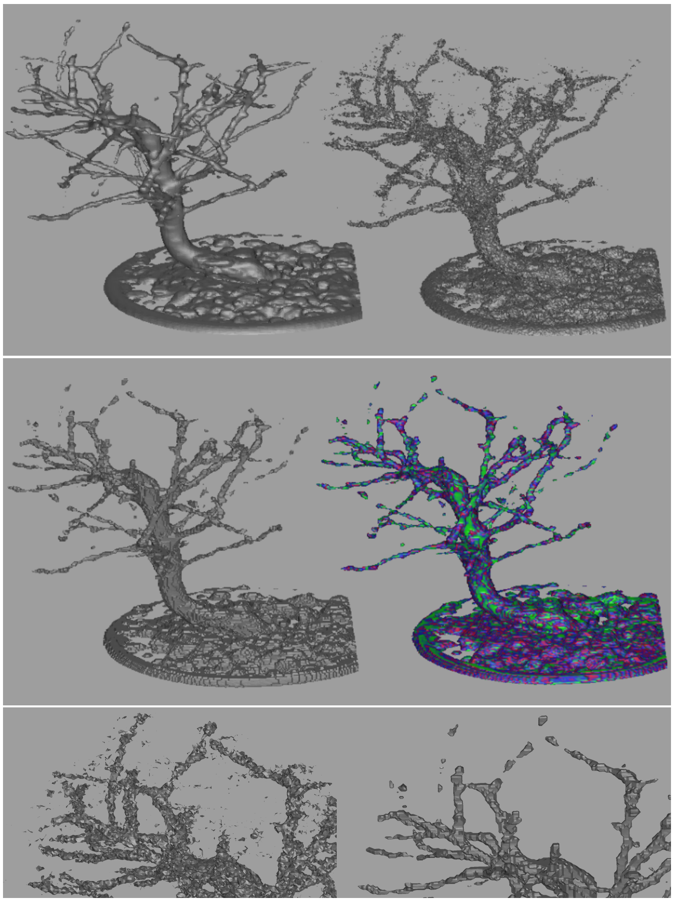


Fig. 11. The bonsai tree dataset. The top row: on the left, the ground truth for isovalue $c = 84$, and on the right, linear interpolation. The second row, the isosurface with the expected level-crossing positions on the left, and the uncertainties in the positions on the right. The third row is a close-up view of the linear interpolation isosurface and the expected level-crossing surface. The noise with uniform distribution of $\delta = 35$ was added to the ground truth.

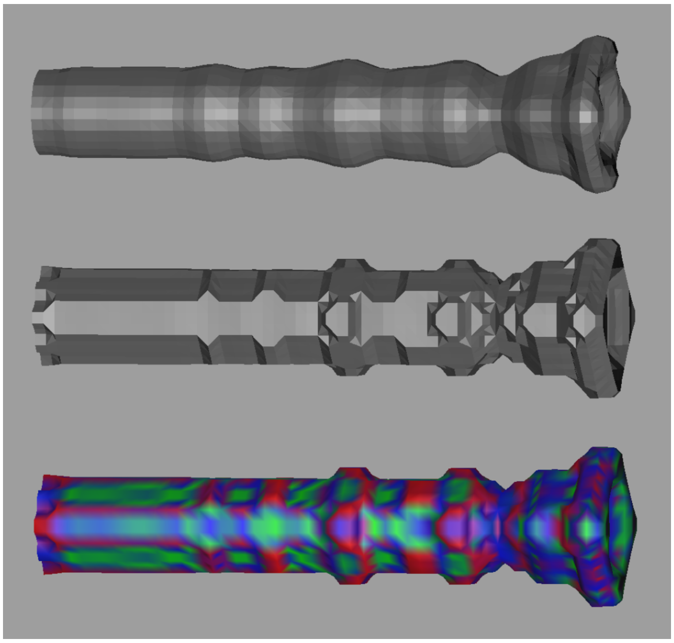


Fig. 12. The fuel dataset. The top figure shows the result of linear interpolation in the estimated mean field. The image in the middle shows the expected crossing result in the same mean field by considering the estimated local variations (δ) field. The bottom figure shows color-mapped distance between linear interpolation and expected crossing. Green color indicates the parts of the linear interpolation isosurface that are relatively stable, blue indicates parts that are moderately stable, and red indicates areas that are relatively unstable. Images taken for the isovalue $c = 56$.

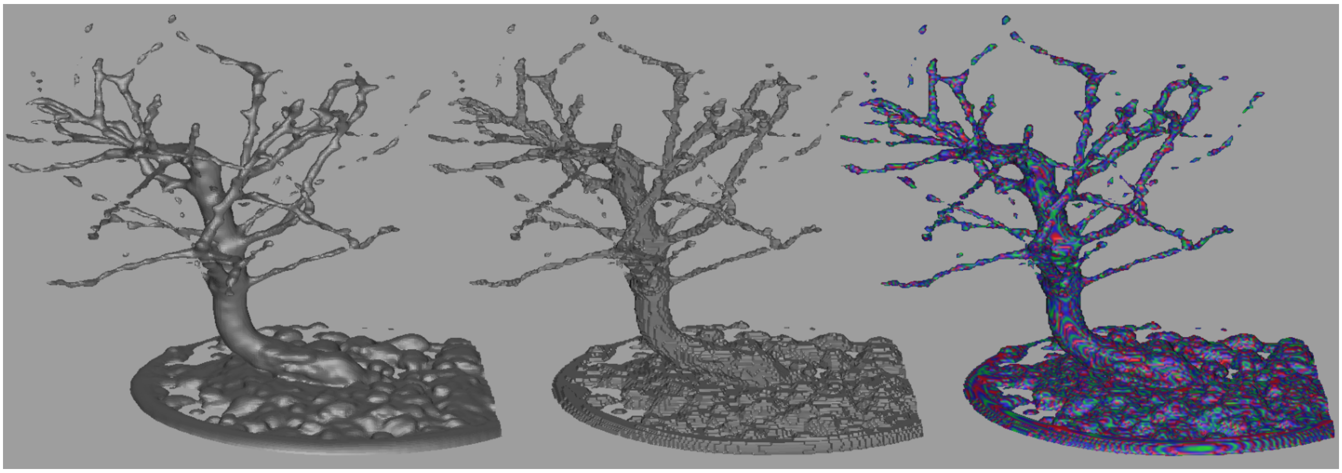


Fig. 13. The bonsai tree dataset. The left figure shows the result of linear interpolation in the estimated mean field. The middle figure pictures the expected crossing result in the same mean field by considering the estimated local variations (δ) field. The rightmost figure shows color-mapped distance between linear interpolation and expected crossing. Green color indicates the parts of the linear interpolation isosurface that are relatively stable, blue indicates parts that are moderately stable, and red indicates areas that are relatively unstable. Images taken for the isovalue $c = 84$.

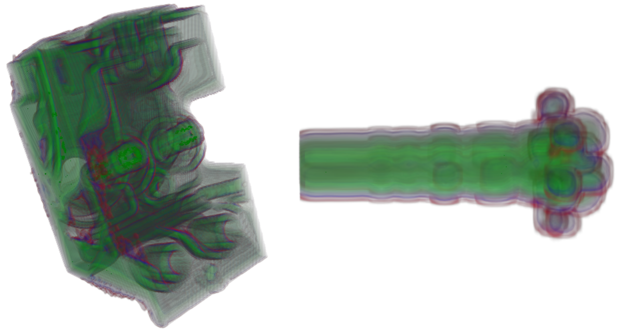


Fig. 14. Direct volume rendering of the cell crossing probabilities for engine (with the isovalue of $c = 58$) and fuel (with the isovalue of $c = 56$) datasets. Green indicates cells in the volume with high isosurface crossing probability, blue indicates areas with moderate probabilities, and red indicates areas with low crossing probability.

the expected position of isosurface vertices, as well as their variances. While the former quantity is used to construct a stable isosurface, the latter quantity is used to visualize the positional uncertainties in the expected level crossings on the underlying grid.

A limitation of the presented approach is the restrictive assumption of uniform distribution for modeling uncertainty. However, as discussed in Section 4, our parallelogram analysis is a building block for deriving the ratio distribution in inverse linear interpolation for more general probability densities that are estimated using the Parzen-Rosenblatt window method. We plan to extend our approach to cover more general kernel-based densities. Moreover, we plan to extend this framework to handle spatial correlations in the random field where the isosurface is being extracted from.

References

[1] R. P. Botchen, D. Weiskopf, and T. Ertl. Texture-based visualization of uncertainty in flow fields. In *Visualization, 2005. VIS 05. IEEE*, pages 647–654. IEEE, 2005.

[2] K. Brodlie, R. Allendes Osorio, and A. Lopes. A review of uncertainty in data visualization. *Expanding the Frontiers of Visual Analytics and Visualization*, pages 81–109, 2012.

[3] P. Cignoni, F. Ganovelli, C. Montani, and R. Scopigno. Reconstruction of topologically correct and adaptive trilinear isosurfaces. *Computers & Graphics*, 24(3):399–418, 2000.

[4] T. Etiene, C. Scheidegger, L. G. Nonato, R. M. Kirby, and C. Silva. Verifiable visualization for isosurface extraction. *Visualization and Computer Graphics, IEEE Transactions on*, 15(6):1227–1234, 2009.

[5] M. Goodchild, B. Buttenfield, and J. Wood. On introduction to visualizing data validity. *Visualization in geographical information systems*, pages 141–149, 1994.

[6] H. Griethe and H. Schumann. The visualization of uncertain data: Methods and problems. In *Proceedings of SimVis*, volume 6, pages 143–156, 2006.

[7] G. Grigoryan and P. Rheingans. Point-based probabilistic surfaces to show surface uncertainty. *Visualization and Computer Graphics, IEEE Transactions on*, 10(5):564–573, 2004.

[8] M. Hlawatsch, P. Leube, W. Nowak, and D. Weiskopf. Flow radar glyphs–static visualization of unsteady flow with uncertainty. *Visualization and Computer Graphics, IEEE Transactions on*, 17(12):1949–1958, 2011.

[9] C. Johnson. Top scientific visualization research problems. *Computer Graphics and Applications, IEEE*, 24(4):13–17, July-Aug. 2004.

[10] C. Johnson and A. Sanderson. A next step: Visualizing errors and uncertainty. *Computer Graphics and Applications, IEEE*, 23(5):6–10, Sept.-Oct. 2003.

[11] R. M. Kirby and C. T. Silva. The need for verifiable visualization. *Computer Graphics and Applications, IEEE*, 28(5):78–83, 2008.

[12] A. Knoll, Y. Hijazi, A. Kensler, M. Schott, C. Hansen, and H. Hagen. Fast ray tracing of arbitrary implicit surfaces with interval and affine arithmetic. In *Computer Graphics Forum*, volume 28, pages 26–40. Wiley Online Library, 2009.

[13] S. K. Lodha, A. Pang, R. E. Sheehan, and C. M. Wittenbrink. Uflow: Visualizing uncertainty in fluid flow. In *Visualization'96. Proceedings.*, pages 249–254. IEEE, 1996.

[14] A. Lopes and K. Brodlie. Interactive approaches to contouring and isosurfacing for geovisualization. *Exploring Geovisualization*, page 345, 2005.

[15] W. E. Lorensen and H. E. Cline. Marching cubes: A high resolution 3d surface construction algorithm. In *Computer Graphics (Proceedings of SIGGRAPH 87)*, volume 21, pages 163–169, July 1987.

[16] C. Lundstrom, P. Ljung, A. Persson, and A. Ynnerman. Uncertainty visualization in medical volume rendering using probabilistic animation. *Visualization and Computer Graphics, IEEE Transactions on*, 13(6):1648–1655, 2007.

[17] A. M. MacEachren, A. Robinson, S. Hopper, S. Gardner, R. Murray, M. Gahegan, and E. Hetzler. Visualizing geospatial information uncertainty: What we know and what we need to know. *Cartography and Geographic Information Science*, 32(3):139–160, 2005.

[18] S. R. Marschner and R. J. Lobb. An evaluation of reconstruction filters for volume rendering. In *Proc. of the IEEE Conf. on Visualization 1994*, pages 100–107. IEEE Computer Society Press, Oct. 1994.

[19] B. K. Natarajan. On generating topologically consistent isosurfaces from

uniform samples. *The Visual Computer*, 11(1):52–62, 1994.

[20] T. S. Newman and W. Lee. On visualizing uncertainty in volumetric data: techniques and their evaluation. *Journal of Visual Languages & Computing*, 15(6):463–491, 2004.

[21] G. M. Nielson and B. Hamann. The asymptotic decider: resolving the ambiguity in marching cubes. In *Proceedings of the 2nd conference on Visualization'91*, pages 83–91. IEEE Computer Society Press, 1991.

[22] M. Otto, T. Germer, H.-C. Hege, and H. Theisel. Uncertain 2d vector field topology. *Computer Graphics Forum*, 29(2):347–356, June 2010.

[23] M. Otto, T. Germer, and H. Theisel. Uncertain topology of 3d vector fields. *Pacific Visualization Symposium (PacificVis), 2011 IEEE*, pages 67–74, March 2011.

[24] A. T. Pang, C. M. Wittenbrink, and S. K. Lodha. Approaches to uncertainty visualization. *The Visual Computer*, 13(8):370–390, 1997.

[25] T. Pfaffelmoser, M. Reitinger, and R. Westermann. Visualizing the positional and geometrical variability of isosurfaces in uncertain scalar fields. In *Computer Graphics Forum*, volume 30, pages 951–960. Wiley Online Library, 2011.

[26] K. Pöthkow and H.-C. Hege. Positional uncertainty of isocontours: Condition analysis and probabilistic measures. *Visualization and Computer Graphics, IEEE Transactions on*, 17(10):1393–1406, 2011.

[27] K. Pöthkow, C. Petz, and H.-C. Hege. Approximate level-crossing probabilities for interactive visualization of uncertain isocontours. *International Journal for Uncertainty Quantification*, 2012.

[28] K. Pöthkow, B. Weber, and H.-C. Hege. Probabilistic marching cubes. In *Computer Graphics Forum*, volume 30, pages 931–940. Wiley Online Library, 2011.

[29] P. J. Rhodes, R. S. Laramee, R. D. Bergeron, T. M. Sparr, et al. Uncertainty visualization methods in isosurface rendering. In *Eurographics*, volume 2003, pages 83–88, 2003.

[30] S. Schlegel, N. Korn, and G. Scheuermann. On the interpolation of data with normally distributed uncertainty for visualization. *Visualization and Computer Graphics, IEEE Transactions on*, 18(12):2305–2314, 2012.

[31] T. Schultz, H. Theisel, and H.-P. Seidel. Topological visualization of brain diffusion MRI data. *IEEE Trans. on Visualization and Computer Graphics*, 13(6):1496–1503, 2007.

[32] J. S. Simonoff. *Smoothing methods in statistics*. Springer, 1996.

[33] H. Wendland. *Scattered data approximation*, volume 17 of *Cambridge Monographs on Applied and Computational Mathematics*. Cambridge University Press, Cambridge, 2005.

[34] B. Widrow and I. Kollár. *Quantization Noise: Roundoff Error in Digital Computation, Signal Processing, Control, and Communications*. Cambridge University Press, Cambridge, UK, 2008.

[35] C. M. Wittenbrink, A. T. Pang, and S. K. Lodha. Glyphs for visualizing uncertainty in vector fields. *Visualization and Computer Graphics, IEEE Transactions on*, 2(3):266–279, 1996.

[36] T. Zuk and S. Carpendale. Theoretical analysis of uncertainty visualizations. In *Proceedings of SPIE*, volume 6060, pages 66–79, 2006.