Variational Level-Set Detection of Local Isosurfaces from Unstructured Point-based Volume Data

Vladimir Molchanov[1], Paul Rosenthal[1], and Lars Linsen[1]

1 Visualization and Computer Graphics Laboratory
Jacobs University Bremen
Campus Ring 1, Bremen, Germany

Abstract

A standard approach for visualizing scalar volume data is the extraction of isosurfaces. The most efficient methods for surface extraction operate on regular grids. When data is given on unstructured point-based samples, regularization can be applied but may introduce interpolation errors. We propose a method for smooth isosurface visualization that operates directly on unstructured point-based volume data avoiding any resampling. We derive a variational formulation for smooth local isosurface extraction using an implicit surface representation in form of a level-set approach, deploying Moving Least Squares (MLS) approximation, and operating on a kd-tree. The locality of our approach has two aspects: first, our algorithm extracts only those components of the isosurface, which intersect a subdomain of interest; second, the action of the main term in the governing equation is concentrated near the current isosurface position. Both aspects reduce the computation times per level-set iteration. As for most level-set methods a reinitialization procedure is needed, but we also consider a modified algorithm where this step is eliminated. The final isosurface is extracted in form of a point cloud representation. We present a novel point completion scheme that allows us to handle highly adaptive point sample distributions. Subsequently, splat-based or mere (shaded) point rendering is applied. We apply our method to several synthetic and real-world data sets to demonstrate its validity and efficiency.

1998 ACM Subject Classification Algorithm/Technique

Keywords and phrases Level-set, isosurface extraction, visualization in astrophysics, particle simulations.

Digital Object Identifier 10.4230/DFU.Vol2.SciViz.2011.222

1 Introduction

Many applications in modern engineering and science provide techniques that produce unstructured point-based volume data. Such data are, for instance, generated by measurements with sensors that are irregularly placed in space. An even larger class of unstructured data generation is given by simulations with Lagrangian methods in engineering, hydrodynamics, astrophysics, etc. A classical representative of such methods is the approach of Smoothed Particle Hydrodynamics (SPH) developed by Lucy [16] and Gingold and Monaghan [8].

SPH is currently used for numerous astrophysical simulations of galaxy mergers, star formation, supernova explosions, and other processes. The method involves no regular spatial grid. Objects are represented as an ensemble of interacting particles, which permanently change their positions according to some governing equations. The particles carry all quantities of interest, which may remain unchanged (mass of a particle) or may vary with time (velocity, density, temperature, etc.). The number of particles in modern simulations

Scientific Visualization: Interactions, Features, Metaphors. *Dagstuhl Follow-Ups, Vol. 2.*
Editor: Hans Hagen; pp. 222–239
DAGSTUHL FOLLOW-UPS Dagstuhl Publishing
Schloss Dagstuhl – Leibniz Zentrum für, Germany

is in the order of several millions. Thus, a large amount of information needs to be saved, transformed, analyzed, and rendered. As a consequence, there exists a constantly growing need in fast, robust, and effective visualization techniques.

A standard approach to visualize scalar volume data is to extract and render isosurfaces. A huge variety of algorithms are known for gridded data. To detect an isosurface from unstructured data, a common procedure is to first interpolate the scalar field to vertices of a regular (rectangular) grid. Such a regularization simplifies the extraction algorithm but inevitably introduces an interpolation error. This error can significantly affect the result, especially in regions with low sample density or highly adaptive sample distribution. The particular strength of Lagrangian simulations such as SPH is its adaptivity. Therefore, visualization methods should make an effort to avoid interpolation to a regular grid whenever possible [23]. Our intent is to perform all operations exclusively on scattered samples, i.e., on unstructured point-based volume data without any given connectivity information.

A high-quality smooth isosurface visualization requires accurate computation of gradients to the isosurface. This becomes a problem for some real-world data sets, as the sampled scalar function may contain noise and may not be sufficiently smooth. There exist two principle approaches to extract smooth isosurfaces:

- The original scalar function f is regularized in a neighborhood of the isosurface and the surface position and the gradient field are found with respect to the modified function;
- An auxiliary smooth (signed distance) function ψ is initialized at the samples and iteratively updated to approach f maintaining desired regularity. Subsequently, an isosurface is extracted due to the final values of ψ.

The first scheme works well, if an a priory information on noise parameters is available. Moreover, the behavior of the scalar function close to the isosurface may be non-uniform: for the real astrophysical data we work with, the ratio of the largest to the smallest gradient norm can be of order 10^3, which makes the first approach hardly applicable. The competitive second scheme being a flexible multitasking tool is referred to as a level-set approach and is widely used in edge detection, image segmentation problems, etc.

We propose a novel method for smooth local isosurface extraction from point-based volume data using an implicit surface representation. Locality in the context means that we are only interested in extracting those components of the isosurface that have a non-empty intersection with a given subdomain (region of interest). This feature is highly desirable when operating on many-object astrophysical data, where different components of an isosurface may occlude each other from a viewer. To derive the governing equation we introduce a cost functional, that is minimized during level-set evolution. After convergence, we extract the detected isosurface in point-cloud representation, which can be made arbitrarily dense for point or splat-based rendering.

The main contributions of this work are:

- a new main term (governing equation) of the cost function, which significantly softens the dependence of parameters on given data;
- a variational approach to the minimization problem;
- an accurate and fast estimation of differential operators based on the Moving Least Squares (MLS) algorithm;
- investigation of a reinitialization-free formulation using the functional proposed by Li et al. [14];
- iterative extraction of isopoints using MLS approximation and Newton's method;

- a new procedure for point-cloud completion resulting in dense, accurate, and smooth point-based rendering of the isosurface; and
- numerical tests involving both synthetical and real-world data sets indicating the behavior of several quantities measuring the quality of the solution.

Our method is designed for but not limited to the visualization of data coming from astrophysical simulations using Smoothed Particle Hydrodynamics.

2 Related Work

Level-set methods are successfully used in various areas of research including visualization. They find their application in image segmentation [5, 17], object detection [6], shape reconstruction [28], isosurface generation [7], multiphase motion [27], etc. The methods are applied to data coming from scanners [17], satellites [25], or tomography [11, 12].

Most level-set algorithms are designed to operate on a grid, which may be uniform, adaptive, octree-based, or of other form. If scattered data are allowed as input, they are typically interpolated to an auxiliary underlying mesh. We refer to recent results in the area of scattered data interpolation that allow for efficient large-scale data processing [20, 21].

MLS is a well-studied tool for reconstructing smooth function from scattered data. The approach finds many applications in numerical simulations (especially, particle methods) and computer graphics. Recently, MLS was applied to volume rendering of shaded isosurfaces from regular and irregular volume data [10]. The high-quality shading requires accurate approximation of derivatives. There exist two methods for gradient computation using MLS: the faster one was proposed by Nayroles et al. [18] and was used in the Diffuse Element Method (DEM) to solve compressible fluid equations in $\mathbb{R}^d$. The more accurate but time-consuming method proposed by Belytschko et al. [3] was used in the Element-Free Galerkin Method (EFG).

A PDE-based technique for smooth isosurface extraction from volume data sampled on particles was developed by Rosenthal and Linsen [23, 24, 15]. Our method also uses original irregularly sampled data to propagate an implicit surface according to a level-set equation. However, in contrast to the approach proposed by Rosenthal and Linsen, we suggest to minimize a cost functional by variational methods. Also, we use the non-morphological formulation to make the method local and achieve better efficiency [19]. The advantages of local level-set methods are reduction of computational efforts, simplicity of implementation, and generality [22, 26]. The data sets we work with represent systems of interacting astrophysical objects. Thus, we prefer to use a local method to be able to extract isosurfaces associated with one object or a group of objects.

We extract isosurfaces in point cloud representation, which we render using point-based rendering methods. The main problem we face is that the point cloud representation of the surface is inhomogeneously dense due to the irregularity of the data samples. Thus, a sampling of additional isosurface points is required. We refer to recent approaches [13, 1, 2] and references therein for some algorithms. For instance, adaptive splatting is a well-known rendering method allowing for high-quality images from point-set surfaces [29]. Our approach to produce additional samples is not based on the extracted point cloud, but is derived from the underlying data field, which allows for a more precise positioning.

3 Problem Setting and Data Structure

In astrophysical simulations, when using a particle-based method like SPH, all quantities of interest are periodically recorded in files. The data usually contains vector fields (current particles positions, velocities, pressure, magnetic field) and scalar functions (mass density, particles' interaction radii, temperature, chemical component concentrations, etc.), typically ordered by particle indices. Required information should then be extracted from the data, transformed, processed, and displayed to facilitate visual exploration in order to obtain a deeper understanding of the modeled processes. In our work, we focus on the detection and rendering of isosurfaces — surfaces corresponding to some prescribed value of a scalar field.

Given a set of N irregularly distributed points $\mathbf{x}_i \in D \subset \mathbb{R}^3$ with associated scalar values $f_i \approx f(\mathbf{x}_i) \in \mathbb{R}$ representing some smooth field $f : D \to \mathbb{R}$ in a bounded domain D, we want to detect and render all components of a smooth isosurface $\Gamma_{\text{iso}} = \{\mathbf{x} \in D : f(\mathbf{x}) = f_{\text{iso}}\}$ with respect to a given real isovalue f_{iso} that have non-empty intersections with a subdomain $D' \subset D$. We assume $\min_i f_i < f_{\text{iso}} < \max_i f_i$. Our method produces a finite set of points $\mathbf{p}_k \in \Gamma_{\text{iso}}$, which are used for surface rendering. From now on, we refer to these points as *isopoints.*

A three-dimensional kd-tree T is used to store space coordinates of points together with pointers to the scalar functions used in the surface extraction algorithm. This data structure is an efficient and flexible tool for fast handling of irregular data and search queries [23, 10]. In a pre-processing step, an indexing of the samples is applied. Subsequently, a major part of the required operations on the data is reduced to a fast search over the indices. In particular, we use the generated kd-tree to compute and store the n nearest neighbors for each sample point. In all our tests, we used $n = 26$.

Although it should be obvious, we still want to point out that the proposed methods can be applied to structured data sets, as well. For structured data, the neighborhood information can be retrieved much easier, i.e., without using the kd-tree structure, which also allows for a more accurate estimation of differential operators on samples and a more precise evaluation of isopoint positions.

4 Variational Formulation and Level-Set Equation

The problem of detecting isosurface locations has much in common with problems in image segmentation. In fact, a simple transformation $f \to \text{sgn}(f - f_{\text{iso}})$ of the data converts the surface defined by a prescribed isovalue into a boundary with a sharp gradient. Then, one can apply any known edge-detection algorithm to capture the surface. However, our goal is to develop an algorithm for isosurface detection that is more efficient, simple, and robust for this task than the transformed image segmentation algorithms.

4.1 Cost Function

We start with a construction of a functional E, which is the target function that we want to minimize. E depends on the given data f, the constant f_{iso} representing the isovalue, and a level-set function φ together with its derivatives of first order, i.e., $E = E(\varphi, \nabla\varphi; f, f_{\text{iso}})$. The total functional consists of two weighted terms

$$E = E_1 + \lambda E_2, \tag{1}$$

responsible for accuracy and smoothness of the solution, respectively. We propose to use

$$E_1 = \frac{1}{4}\int_D (\operatorname{sgn}(\varphi(\mathbf{x})) - \operatorname{sgn}(f(\mathbf{x}) - f_{\text{iso}}))^2 \,\mathrm{d}\mathbf{x} \tag{2}$$

and

$$E_2 = \int_D \delta(\varphi(\mathbf{x}))\,|\nabla\varphi(\mathbf{x})|\,\mathrm{d}\mathbf{x}. \tag{3}$$

Here, we use the standard definitions of the sign function

$$\operatorname{sgn}(x) = \begin{cases} 0 & \text{for} \quad x = 0 \\ x/|x| & \text{for} \quad x \neq 0 \end{cases}$$

and the Dirac function

$$\delta(x) = \frac{\mathrm{d}}{\mathrm{d}x}H(x), \qquad H(x) = \begin{cases} 1 & \text{for} \quad x \geq 0 \\ 0 & \text{for} \quad x < 0 \end{cases} .$$

$H(x)$ denotes the Heaviside function and the derivative is used in the sense of distributions. A discussion of the proposed functional E in comparison with previous work is provided in Section 7.

A function φ_∞ minimizing a functional of the form $\int L(\varphi, \nabla\varphi)\,\mathrm{d}\mathbf{x}$ satisfies the Euler-Lagrange equation

$$\left(\frac{\partial L}{\partial\varphi} - \sum_i \frac{\partial}{\partial x_i}\frac{\partial L}{\partial\varphi_i}\right)\Bigg|_{\varphi=\varphi_\infty} = 0, \tag{4}$$

where φ_i is the i-th component of $\nabla\varphi$. We derive the Euler-Lagrange equations for E_1 by

$$\frac{1}{2}(\operatorname{sgn}(\varphi(\mathbf{x})) - \operatorname{sgn}(f(\mathbf{x}) - f_{\text{iso}}))\ \operatorname{sgn}'(\varphi(\mathbf{x})) = 0 \tag{5}$$

and for E_2 by

$$\begin{aligned} 0 &= \delta'(\varphi(\mathbf{x}))|\nabla\varphi(\mathbf{x})| - \nabla\cdot\left[\delta(\varphi(\mathbf{x}))\,\frac{\nabla\varphi(\mathbf{x})}{|\nabla\varphi(\mathbf{x})|}\right] \\ &= \delta'(\varphi)|\nabla\varphi| - \delta'(\varphi)\nabla\varphi\cdot\frac{\nabla\varphi}{|\nabla\varphi|} - \delta(\varphi)\,\nabla\cdot\left[\frac{\nabla\varphi}{|\nabla\varphi|}\right] = -\,\delta(\varphi(\mathbf{x}))\,\nabla\cdot\left[\frac{\nabla\varphi(\mathbf{x})}{|\nabla\varphi(\mathbf{x})|}\right]. \end{aligned} \tag{6}$$

The idea of a level-set approach is to reach φ_∞ being a fixed point of an evolution equation for $\varphi = \varphi(\mathbf{x}, t)$ minimizing E. Here, t is an artificial time parameterizing the minimization process $\varphi(\mathbf{x}, t) \to \varphi_\infty(\mathbf{x})$ as $t \to \infty$. To construct the PDE, one equates the left-hand side of Equation (4) with $-\partial\varphi/\partial t$. For the functional E, it reads [6, 9]

$$\frac{\partial\varphi}{\partial t} = \delta(\varphi)\ (\operatorname{sgn}(f - f_{\text{iso}}) - \operatorname{sgn}(\varphi)) + \lambda\,\delta(\varphi)\ \nabla\cdot\left(\frac{\nabla\varphi}{|\nabla\varphi|}\right). \tag{7}$$

In this derivation, we used

$$\operatorname{sgn}(x) = 2H(x) - 1, \qquad \operatorname{sgn}'(x) = 2H'(x) = 2\delta(x) \tag{8}$$

in the sense of distributions. In the following, Equation (7) will be regularized and discretized in space and time, leading to an iterative process for the value of φ at each sample point.

4.2 Spatial and Temporal Discretization

In practice, the Dirac function $\delta(x)$ in Equation (7) is smoothed on a small scale ϵ. Recall that $\delta(\varphi)$ in the first term of Equation (7) appears as a derivative of the sgn-function. Therefore, in order to be consistent with the differentiation, one should also smooth $\mathrm{sgn}(\varphi)$. It is worth to note that in many papers (e.g., [6, 14, 19]) the smoothed functions contain in their definitions trigonometric expressions, which are expensive to compute. Our choice is to use polynomial splines

$$\delta_\epsilon(x) = \frac{1}{\epsilon}\hat{\delta}\left(\frac{x}{\epsilon}\right), \qquad \hat{\delta}(x) = \frac{1}{6}\begin{cases} 4 - 6x^2 + 3x^3 & \text{for} \quad |x| \le 1 \\ (2-x)^3 & \text{for} \quad 1 < |x| \le 2 \\ 0 & \text{for} \quad |x| > 2 \end{cases},$$

$$\mathrm{sgn}_\epsilon(\epsilon x) = \frac{\mathrm{sgn}(x)}{12}\begin{cases} 16|x| - 8|x|^3 + 3x^4 & \text{for} \quad |x| \le 1 \\ 12 - (|x|-2)^4 & \text{for} \quad 1 < |x| \le 2 \\ 12 & \text{for} \quad |x| > 2 \end{cases}.$$

Here $\mathrm{sgn}'_\epsilon(x) = 2\delta_\epsilon(x)$ similar to Equation (8).

The quantity 4ϵ characterizes the width of a band around the current zero-level surface, in which the terms of Equation (7) operate. The smaller the value of parameter ϵ is, the stronger is the localization of the update. We want to mention that Equation (7) contains two delta-functions, which could be smoothed on different scales ϵ and ϵ'. In grid-based methods, the smoothing parameter is chosen to be in the order of several grid steps. In our case, there is no such characteristic, since the samples are scattered. Thus, the optimal ϵ and ϵ' depend on local samples distribution and the parameters can be corrected during simulations.

To discretize Equation (7) in space we need a technique to approximate the differential operators (i.e., gradient and divergence) on the unstructured data. For each point of interest $\mathbf{x}_i$, we use the kd-tree to find its n nearest neighbors from the whole set of samples. Let V_i be the set of the n nearest neighbors of $\mathbf{x}_i$ including the point itself. Then, for any scalar function g and sample $\mathbf{x}_i$ we can determine the parameters of the linear function $g_i(\mathbf{x}) = \mathbf{a}_i \cdot \mathbf{x} + b_i$, $g_i : \mathbb{R}^3 \to \mathbb{R}$ from the condition

$$\sum_{\mathbf{x}_j \in V_i} w_i^2(\mathbf{x}_i - \mathbf{x}_j)(g(\mathbf{x}_j) - g_i(\mathbf{x}_j))^2 \to \min$$

with some weight function $w_i(\mathbf{x})$. In other words, g_i is the local MLS approximation of the scalar field g in the neighborhood of $\mathbf{x}_i$. Practically, the unknown coefficients are found as the solution of the overdetermined system of linear equations

$$w_i(\mathbf{x}_i - \mathbf{x}_j)(\mathbf{a}_i \cdot \mathbf{x}_j + b_i) = w_i(\mathbf{x}_i - \mathbf{x}_j)g(\mathbf{x}_j), \qquad \mathbf{x}_j \in V_i,$$

using QR-decomposition [4] or by explicit formulae that are analogous to the ones proposed by Rosenthal and Linsen [24] for least-squares approximation. The least-squares approximation used by Rosenthal and Linsen can be viewed as a particular case of MLS with the weight function $w(\mathbf{x}) \equiv 1$. A simple one-dimensional test comparing the two approaches is presented in Figure 1. It is shown that MLS produces a smoother interpolation when compared to least squares.

For all our tests we use

$$w_i(\mathbf{x}) = \begin{cases} (1 - \|\mathbf{x}\|^2/h_i^2) & \text{for} \quad |x| \le h_i \\ 0 & \text{else} \end{cases},$$

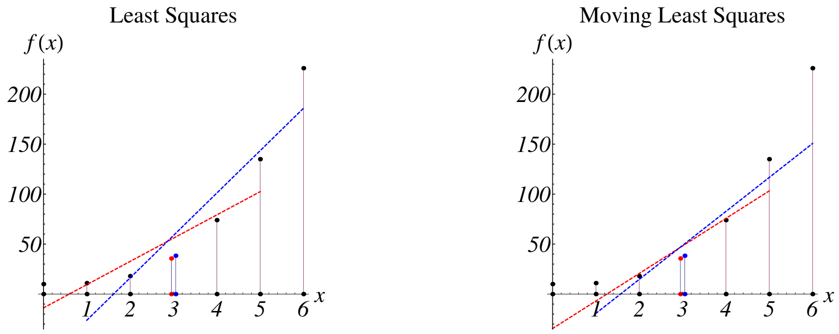

Figure 1 Comparison of least squares and MLS approximation for two close samples in one spatial dimension. Samples are located at $\{0, 1, 2, 2.95, 3.05, 4, 5, 6\}$ with values due to the function $f(x) = x^3 + 10$. We used 6 nearest neighbors in both cases and the weight function $w(x) = (1 - (x/3.5)^2)$ for MLS. MLS produces a smoother interpolation.

where h_i is the support size chosen to be equal to the distance between the ith sample and the farthest of the 26 nearest neighbors.

After having determined an MLS linear approximation in the neighborhood of the ith sample, we can set

$$\nabla g(\mathbf{x}_i) \approx \mathbf{a}_i.$$

We note that we stay rather in the frame of the DEM than the EFG formulation. Similar to Ledergerber et al. [10] we prefer the faster method, since the improvement by more accurate derivative estimation is negligible. To estimate the divergence of a vector field (g^1, g^2, g^3) at a sample point $\mathbf{x}_i$ we construct the linear least-square approximations g_i^k for each k-th component and then sum up the coefficients as

$$\nabla \cdot (g^1,\ g^2,\ g^3)(\mathbf{x}_i) = (\mathbf{a}_i^1)_1 + (\mathbf{a}_i^2)_2 + (\mathbf{a}_i^3)_3.$$

For the time integration we use an explicit Euler scheme. Hence, the iteration step

$$\varphi^{k+1} = \varphi^k + \Delta t \left[\delta_\epsilon(\varphi^k)\ (\operatorname{sgn}(f - f_{\text{iso}}) - \operatorname{sgn}_\epsilon(\varphi^k)) + \lambda\, \delta_{\epsilon'}(\varphi^k)\ \nabla \cdot \left(\frac{\nabla \varphi^k}{|\nabla \varphi^k|} \right) \right] \tag{9}$$

is applied to all sample points $\mathbf{x}_i$. The upper index k stands for the k-th layer in time. The use of the explicit first-order scheme ensures that the iterations are stable for a sufficiently small Δt depending on the spatial resolution. However, since the sampling is irregular, the choice of a globally stable Δt can be small. We refer the reader to the discussion about the choice of Δt and asynchronous time integration by Rosenthal and Linsen [24].

4.3 Reinitialization

Stability of evolution of φ according to a level-set equation requires closeness of φ to a signed distance function [22]. The width of the band involved in computations due to Equation (9) is determined by the size of the support of the smeared-out delta function $\delta_\epsilon(\varphi)$. In the neighborhood of an isopoint $\mathbf{p} \in \Gamma_{\text{iso}}$, it can be roughly estimated as $4\epsilon|\nabla\varphi(\mathbf{p})|$. Thus, a high gradient of φ makes the band narrow and slows down propagation of Γ_{iso}. However, small values of the gradient norm make the computations in Equation (9) unstable due to the division by $|\nabla\varphi|$. This observation explains the need for keeping $|\nabla\varphi| \approx 1$ during computations.

An exact calculation of the signed distance function is extremely time consuming. A standard approach is to trigger a reinitialization routine after several evolution steps [19]. This procedure drives the auxiliary function ψ towards a signed distance function of the zero-isosurface as in the Eikonal equation

$$\frac{\partial\psi(\mathbf{x},\tau)}{\partial\tau} = \operatorname{sgn}(\varphi(\mathbf{x},t))\,\left(1 - |\nabla\psi(\mathbf{x},\tau)|\right) . \tag{10}$$

We discretize and solve Equation (10) for $\tau > 0$ with the initial condition $\psi(\mathbf{x}, 0) = \varphi(\mathbf{x}, t)$. A few iterations suffice. After the reinitialization procedure, we set $\varphi = \psi$ and continue the computations with respect to the level-set equation.

If the level-set function remains close to the signed distance function during evolution, the governing equation could be simplified [19]. For example, the mean curvature term could be reduced to the Laplacian of φ. In our approach, no simplified equation is used. Therefore, we may moderate the requirement for φ to $c < |\nabla\varphi(\mathbf{x}_i, t)| < C$ with some constants c and $C \in \mathbb{R}^+$ for all $1 \leq i \leq N$ and $t \geq 0$. Hence, it is enough to have a mechanism that suppresses large and low gradients. Such a tool was proposed by Li et al. [14] in form of an additional diffusion term. This term minimizes a functional controlling the closeness of the level-set function to the signed distance function. It eliminates the need to solve the Eikonal equation. We discuss this enhancement of our level-set method in Section 5.

4.4 Choice of Parameters and Level-Set Function Initialization

The parameters ϵ and ϵ' define a band around the isosurface, where the level-set function values are to be updated. Usually, they are chosen to be comparable with the cell size. In our experiments, we set $\epsilon = \epsilon' \in [1.0l, 1.5l]$, where l is the local average distance between samples. This choice results in a width of the band of 4ϵ due to the definition of the δ_ϵ function. Small values prevent isosurface propagation, large smoothing parameters decrease efficiency of the method.

Next, we choose a sufficiently small time step Δt to ensure that the main term in Equation (9) does not destroy local monotonicity of φ and provides stable computations. In Figure 2, we show how the choice of Δt influences the shape of $\varphi^1(x)$ initialized as $\varphi^0(x) = -x$. The one-dimensional examples show that large time steps cause instabilities, whereas small steps Δt slow down the propagation of the profile. Scaling analysis gives $\Delta t \sim \epsilon^2$. In our tests we use $\Delta t \in [0.3\epsilon^2,\ 0.6\epsilon^2]$. It is possible to derive an exact estimation for the time stepping as it is shown for simple one-dimensional regularized functions in the appendix.

The last parameter to be set is λ. We refer the reader to Rosenthal and Linsen [24] for the stability analysis of the smoothing term in Equation (9) for regular and unstructured grids. Small λ may result in non-smooth surface, large values make computations unstable. For our tests we use $\Delta t\lambda < 0.5$.

Then we should specify an initial condition for Equation (9) to obtain a Cauchy problem. For the reasons discussed in the previous section, it is desirable to set $\varphi(\mathbf{x}, 0)$ as a signed distance function, i.e., $|\nabla\varphi(\mathbf{x}, 0)| = 1$. In our tests, we use the scalar function f representing mass density or temperature distribution in an astrophysical object. These quantities necessarily vanish on the boundary of the domain D. Thus, their isosurfaces are closed hypersurfaces with gradients pointing inside the closed volume. Therefore, it is preferable to initialize the level-set function with a closed zero-level surface embedded in D and containing the subdomain of interest D' inside the bounded volume. A simple way to achieve this, is to use the ansatz

$$\varphi(\mathbf{x}, 0) = r_0 - |\mathbf{x} - \mathbf{x}_0|$$

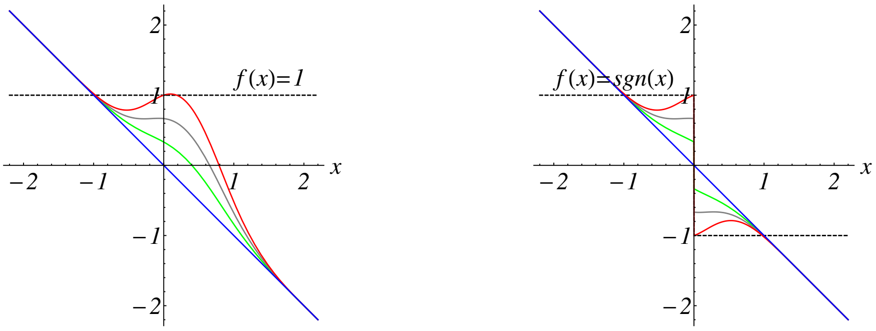

Figure 2 Dependance of the profile of the level-set function after one time step on the choice of Δt using the one-dimensional underlying scalar fields $f(x) = 1$ (left) and $f(x) = sgn(x)$ (right). Initial level-set function $\varphi^0(x) = -x$ (blue line) and level-set function $\varphi^1(x)$ after first time step for $\Delta t = 0.5,\ 1.0,\ 1.5$ (green, gray, and red lines, respectively) are shown with parameter $\epsilon = 1$. Large time steps destroy the local monotonicity of φ.

with some parameters $r_0 > 0$ and $\mathbf{x}_0 \in D$ representing the radius and center of a sphere, respectively. Let $\Omega^+(g) = \{\mathbf{x} \in D : \ g(\mathbf{x}) > 0\}$ and $\Omega^-(g) = \Omega^+(-g)$ for any scalar function g. Let the region $\Omega^+(f - f_{\text{iso}})$ have several components. Our method recovers all isosurfaces, for which the intersection of $\Omega^+(f - f_{\text{iso}})$ with $\Omega^+(\varphi)$ is non-empty.

5 Level-Set Method without Reinitialization

A new variational formulation for active contours was proposed by Li et al. [14]. The main idea is to add the term

$$E_3 = \frac{1}{2} \int_D (|\nabla\varphi(\mathbf{x})| - 1)^2 \, \mathrm{d}\mathbf{x} \tag{11}$$

added with some weight μ to the functional E. This term penalizes deviation of the level-set function from a signed distance function. It aims to make the algorithm more efficient by eliminating the costly reinitialization routine.

The associated Euler-Lagrange equation reads

$$0 = -\ \nabla \cdot \left[(|\nabla\varphi(\mathbf{x})| - 1) \frac{\nabla\varphi(\mathbf{x})}{|\nabla\varphi(\mathbf{x})|} \right] . \tag{12}$$

We add the weighted term to Equation (7), discretize it in space and time, and derive the formulation

$$\begin{aligned} \varphi^{k+1} = \varphi^k + \Delta t\ \Big[& \delta_\epsilon(\varphi^k)\ (\operatorname{sgn}(f - f_{\text{iso}}) - \operatorname{sgn}_\epsilon(\varphi^k)) \\ & + \lambda\, \delta_{\epsilon'}(\varphi^k)\ \nabla \cdot \left(\frac{\nabla\varphi^k}{|\nabla\varphi^k|} \right) + \mu\, \nabla \cdot \left((|\nabla\varphi^k| - 1) \frac{\nabla\varphi^k}{|\nabla\varphi^k|} \right) \Big] . \end{aligned} \tag{13}$$

One of our tasks was to test the algorithm without reinitialization on the problems of smooth isosurface extraction in comparison to the standard variant with the Eikonal equation.

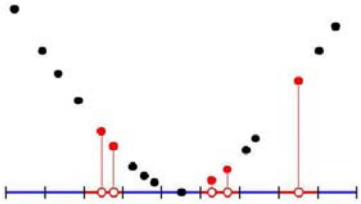

Figure 3 Scheme of the local completion of the point-based surface. The sample of interest and its neighbors (dark), added rough estimates of additional samples (red non-filled), and corrected additional points (red filled) are shown.

6 Isopoints Computation

Following steps are performed after finishing an iteration due to the level-set equation:

1. a rough estimate for a set of isopoints $\mathbf{p}_k$ is computed;
2. positions of $\mathbf{p}_k$ are improved;
3. the set of isopoints is completed and rendered.

In the first step, we find all pairs of neighboring samples $\mathbf{x}_i$ and $\mathbf{x}_j$ such that $\operatorname{sgn}(\varphi_i) \neq \operatorname{sgn}(\varphi_j)$. Using an angle criterion [23], some pairs are rejected. Isopoints are found by means of linear interpolation between the pairs of detected samples.

In the second step, we refine the position of each isopoint $\mathbf{p}_k$ using MLS and Newton's method as follows:

$$\mathbf{p}_k^{i+1} = \mathbf{p}_k^i - \frac{\varphi(\mathbf{p}_k^i)}{|\nabla\varphi(\mathbf{p}_k^i)|^2}\nabla\varphi(\mathbf{p}_k^i), \qquad \mathbf{p}_k^0 = \mathbf{p}_k. \tag{14}$$

Iterations are performed until $|\mathbf{p}_k^{i+1} - \mathbf{p}_k^i|$ is sufficiently small.

The resulting point set is quite accurate but in most cases has low (and non-uniform) density for rendering. We complete the set as it is shown in Figure 3. In the neighborhood of each $\mathbf{p}_k$ we compute a linear approximation to the isosurface. The plane is divided in cells of small size. Then, neighbors of $\mathbf{p}_k$ are orthogonally projected to the plane. We add new samples in empty cells and refine them by Newton's method (14). Normals at isopoints are computed during the refinement procedure. Assuring high sampling rate, the final completed point set can be rendered using mere point renderings (with local shading). Alternatively, one can use a somewhat lower sampling rate and use the standard splat-based rendering approach

7 Comparison with Literature

There are three aspects that distinguish our approach from the one proposed by Rosenthal and Linsen [24], namely: the main part of the cost function, the variational derivation of the level-set equation, and the non-morphological approach. In addition, we consider an algorithm with a proper diffusive term to make the reinitialization procedure obsolete.

The functional

$$\bar{E} = \int_D |\varphi(\mathbf{x}) - (f(\mathbf{x}) - f_{\text{iso}})| \,\mathrm{d}\mathbf{x} \tag{15}$$

was proposed by Rosenthal and Linsen [23] to measure the accuracy of isosurface extraction. The proposed evolution level-set equation has the form

$$\frac{\partial \varphi(\mathbf{x},t)}{\partial t} + (\varphi(\mathbf{x},t) - (f(\mathbf{x}) - f_{\text{iso}}))\,|\nabla\varphi(\mathbf{x},t)| = 0\,. \tag{16}$$

This equation pushes φ towards $f(\mathbf{x}) - f_{\text{iso}}$ on the global scope, whereas the reinitialization step tries to restore the signed distance function. As a result, φ converges to some compromise, possibly being far away from both. Moreover, if we scale the field $f(\mathbf{x}) - f_{\text{iso}}$ by a large constant factor k, the isosurface obviously remains unchanged, but the discretized Equation (16) requires a much smaller time step Δt. When nevertheless choosing larger time steps, the reinitialization procedure needs to be called after each iteration and needs to perform many iterations on the whole data set. Obviously, the computational costs increase significantly.

It is absolutely not necessary to achieve $\varphi(\mathbf{x},t) \to f(\mathbf{x}) - f_{\text{iso}}$ to be able to detect the isosurface Γ_{iso}. It suffices to make the sets $\Omega^{\pm}(\varphi)$ and $\Omega^{\pm}(f - f_{\text{iso}})$ coincide. The closeness of the sets can be measured with the functional E_1. In contrast to $\bar{E}$, our functional and level-set equation parameters are not sensible to the magnitude of f, which makes our approach preferable. In practice, the minimization of E_1 on a discrete set of nodes results in the detection of the isosurface that is accurate up to the local internodal scale. Correction of the isosurface on the subgrid scale is then dominated by the smoothing and reinitialization procedures.

Finally, we underline the fact that our goal is to extract local isosurfaces. The method automatically detects those parts of a multi-component isosurface, which intersect with the subdomain of interest. This property will be demonstrated in experiments below.

8 Results and Discussion

8.1 Measurement Functionals

The following functionals are computed and recorded during our tests to measure the quality of the solution.

$$P(t) = \frac{1}{2N}\sum_{i=1}^{N} |\text{sgn}(\varphi(\mathbf{x}_i,t)) - \text{sgn}(f(\mathbf{x}_i) - f_{\text{iso}})|$$

computes the relative number of samples, at which the sign of the level-set function is different to the sign of the underlying scalar field. P can remain large, if not all isosurface components are extracted. Constant P indicates no propagation of the isosurface.

$$MG(t) = \max_j \left\{|\nabla\varphi(\mathbf{p}_j,t)|,\ |\nabla\varphi(\mathbf{p}_j,t)|^{-1}\right\}$$

and

$$AG(t) = \frac{1}{M}\sum_{j=1}^{M} \max\left\{|\nabla\varphi(\mathbf{p}_j,t)|,\ |\nabla\varphi(\mathbf{p}_j,t)|^{-1}\right\}$$

measure the maximal and the average deviation of φ from the signed distance function at M isopoints. If the analytical form of f is known, we can estimate the average distance from isopoints to the exact isosurface as follows

$$L(t) = \frac{1}{M}\sum_{j=1}^{M} \frac{|f(\mathbf{p}_j) - f_{\text{iso}}|}{|\nabla f(\mathbf{p}_j)|}.$$

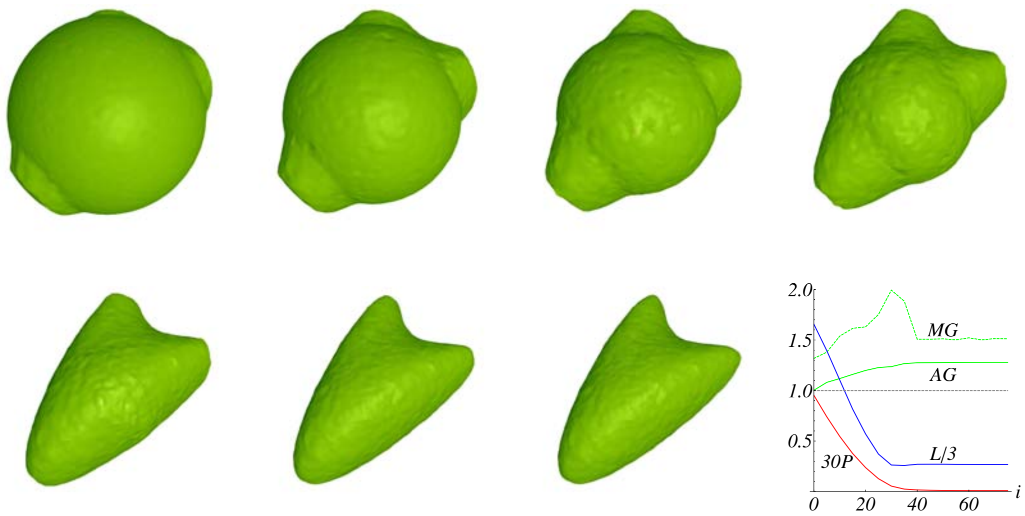

Figure 4 Level-set evolution of the isosurface for a one-component synthetic data set: steps 5, 10, 15, 20, 30, 50, and 75 are presented. Splat-based rendering of refined isopoint sets is performed. Parameters: total number of samples in the box $[0, 128]^3$ is 2M, $\triangle t = 1.0$, $\epsilon = 4.0$, $\epsilon' = 5.0$, and $\lambda = 0.15$. Two reinitialization steps are performed after each iteration.

Here, we used

$$|\mathbf{p}_j - \mathbf{y}| = \frac{|f(\mathbf{p}_j) - f(\mathbf{y})|}{|f(\mathbf{p}_j) - f(\mathbf{y})|} \|\mathbf{p}_j - \mathbf{y}\| \approx \frac{|f(\mathbf{p}_j) - f_{\text{iso}}|}{|\nabla f(\mathbf{p}_j)|},$$

where $\mathbf{y}$ is the point closest to $\mathbf{p}_j$ on the exact isosurface: $f(\mathbf{y}) = f_{\text{iso}}$. The approximation is accurate for small $|\mathbf{p}_j - \mathbf{y}|$.

Beside the quantities above, we are interested in the efficiency of the methods, which is related to the amount of time needed to perform one temporal iteration. We have performed several tests based on synthetic and real-world data presented in the following.

8.2 One-component Synthetic Data

In our first experiment, we investigated the accuracy and convergence of our method. We use a synthetic volume data given by the analytic scalar field $f(\mathbf{x}) = -((ax + 0.05(ay)^3)^2 + 0.7(ay)^2 + 0.3(az - 0.7(ya)^2)^2 - 4.0)/a^2$ with $a = 0.1$. Extracted is the isosurface $\Gamma = \{\mathbf{x} : \ f(\mathbf{x}) = 0\}$. There are 2M samples randomly distributed within a cube of edge size 128. The isosurface is initialized as a sphere. Positions of the isosurface during the level-set iterations as well as the evolution of the main process characteristics are presented in Figure 4. One iteration of our level-set approach takes about 15s on a standard PC with an Intel Xeon (3.2GHz, quad-core) processor. The final average distance between isopoints and the exact isosurface is of order 0.7, which is less than the average sample distance (~ 1) and the characteristic length of the isosurface (~ 100). Here, as well as in the next experiments we intentionally perform more iterations than necessary to document convergence of our method.

8.3 Multi-component Synthetic Data

The second experiment shows the extraction of a subset of isosurface components and illustrates the flexibility of the method with respect to the initial surface setting. The results are shown in Figure 5. The initial isosurface represents two intersecting spheres.

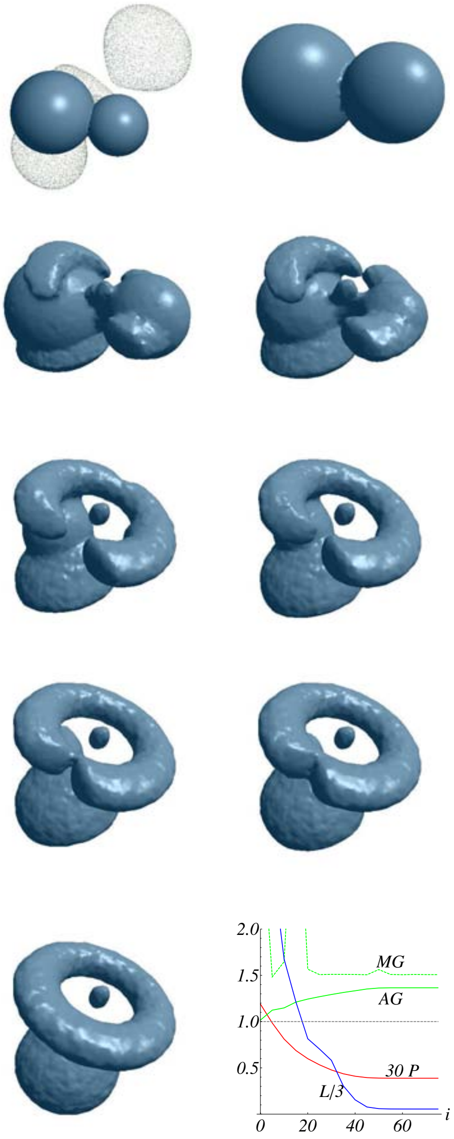

Figure 5 Level-set evolution of the isosurface for a multi-component synthetic data set: steps 0, 3, 5, 8, 10, 20, 30, 50 and 100 are shown. Point rendering is used. Parameters: total number of samples in the box $[0, 128]^3$ is 2M, $\triangle t = 1.25$, $\epsilon = 3.5$, $\epsilon' = 5.0$, and $\lambda = 0.01$. Two reinitialization steps are performed after each iteration.

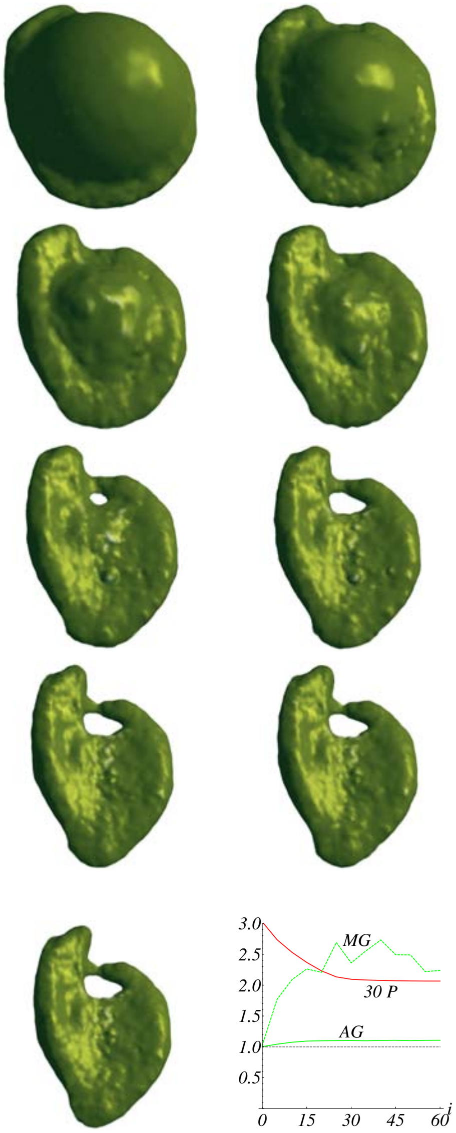

Figure 6 Level-set evolution of the isosurface for the White Dwarf data set: steps 0, 3, 5, 8, 10, 20, 30, 50 and 100 are shown. Point rendering is used. Parameters: total number of samples in the box $[0, 128]^3$ is 2M, $\triangle t = 1.25$, $\epsilon = 3.5$, $\epsilon' = 5.0$, and $\lambda = 0.01$. Four reinitialization steps are performed after each iteration. Data set courtesy of Stephan Rosswog, Jacobs University, Bremen, Germany.

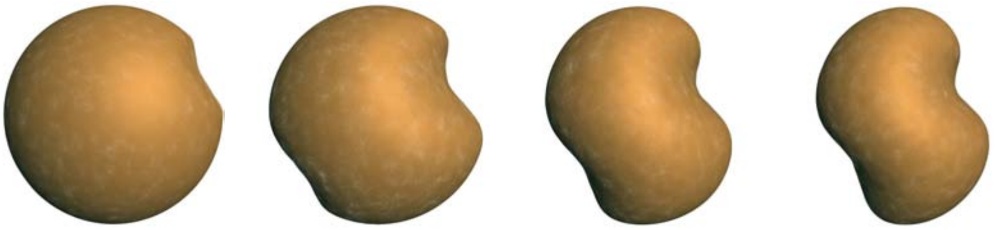

Figure 7 Level-set evolution of the isosurface without reinitialization for the data set with two White Dwarfs: steps 10, 50, 100 and 150 are shown. Splat-based rendering is used. Parameters: total number of samples is 2.5M, $\triangle t = 0.0025$, $\epsilon = 1.0$, $\epsilon' = 0.2$, $\lambda = 10^{-8}$, and $\mu = 10^{-6}$. Data set courtesy of Stephan Rosswog, Jacobs University, Bremen, Germany.

One iteration takes about 15s. The final average distance between isopoints and the exact isosurface is ≈ 0.14 (the average sample distance: ~ 1, object size: ~ 90). The method is stable to the isosurface's topology change. Larger values of λ lead to disappearance of the central spherical isosurface component due to the high curvature values. High values of $MG(t)$ in the first steps are due to non-smooth behavior of the initial level-set function: the gradient degenerates at the samples close to the spheres' intersection line.

8.4 Real Volume Data

In the next test we extract one of two isosurface components by applying our method to a region of interest of a real-world astrophysical data set. The data set consists of 0.5M samples and represents a White Dwarf passing close to a Black Hole (not represented in data). The isosurface of the star corresponding to $f_{\text{iso}} = 0.001$ consists of two parts (a head and a tail). We extract one of them placed in the core of the White Dwarf. The results are shown in Figure 6. One iteration takes about 12s.

8.5 Method without Reinitialization

We applied the reinitialization-free method (13) to all data sets from previous tests. We found that the propagation of the isosurface is too slow for various parameters combinations we tried. The reason could be that the diffusion-like term may push the isosurface in the direction opposite to its movement. The larger μ, the more obvious is this action. Low μ prevents the suppression of high gradients, which also leads to slow propagation of the surface. In contrast, this is not the case for the method with reinitialization, since we do not change the isosurface location while solving the Eikonal equation (10).

Results of one experiment for the reinitialization-free algorithm are presented in Figure 7. The data set consist of 2.5M samples and represents two White Dwarfs passing close to each other with a mass accretion developed. We extract the isosurface $\Gamma = \{\mathbf{x} : \ f(\mathbf{x}) = 0.055\}$ of one star only. The level-set function was initialized as a sphere. One iteration took $\sim 12s$ computational time. In Figure 8 we compare the evolution of gradient norms (minimal, maximal, and average) with the diffusion term enabled (green curves) and disabled (blue curves). The diffusion term moves the average gradient norm toward the value 1 and therefore helps the surface propagation. However, since the coefficient μ was chosen to be small, the effect is minimal. It is impossible to increase μ, as it leads to unstable computations.

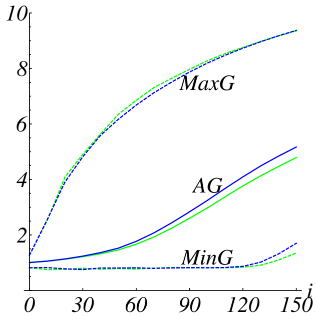

Figure 8 Influence of the diffusion term from Equation (5) on the behavior of the gradient norm. Green lines represent the experiments of Figure 7 with $\mu = 10^{-6}$, while blue lines show the same experiments with $\mu = 0$. The improvement of the gradient norm evolution is minimal, but larger values of μ lead to instable computations.

9 Conclusion and Future Work

We have proposed a novel level-set approach for smooth isosurface extraction that works for both structured and unstructured point-based volume data. A variational approach was used to derive a level-set equation that determines the evolution of the interface towards its steady state. The variational approach provides fast convergence. The action of the main terms of the governing equation at each time step was localized to a narrow neighborhood around the zero level surface of the auxiliary function. This led to a speed-up of our method when compared to algorithms acting on all levels. The locality allowed us to apply our method to subdomains of interest and to extract a subset of isocomponents. Finally, we tested a specific diffusion term in the level-set equation to avoid costly reinitialization.

We compared our algorithm to the state-of-the-art approach proposed by Rosenthal and Linsen. We showed that in their algorithm the efficiency was strongly affected by properties of the underlying scalar field f. In particular, large gradients required small time steps and frequent reinitialization. We removed this issue by a proper choice of the cost functional. We tested the proposed method on synthetic and real-world data sets. Localization of the update reduced the computational time of the isosurface extraction in the White Dwarf data by factor 2 when compared to the method by Rosenthal and Linsen.

In the proposed method, all computations (except those due to the signed distance property) are concentrated in a small neighborhood around the zero-isosurface of the level-set function. It provides a gradual smooth evolution of the surface during computations, but still requires significant computational efforts to maintain the signed distance property. An improvement in the efficiency of the method can be obtained by incorporating a local reinitialization procedure in the algorithm.

There are possible improvements in the algorithm which would increase the accuracy at the cost of efficiency:

- use of a higher-order basis in MLS;
- more accurate estimation of MLS gradients [3]; and
- more accurate time-integration scheme (provided that the spatial approximation is accurate).

To increase the efficiency of the method, one can use individual time steps for samples and smoothing parameters that are adaptive in space and time.

Acknowledgements

This work was supported by the Deutsche Forschungsgemeinschaft (DFG) under project grant LI-1530/6-1.

Appendix

▶ **Theorem 1.** *Let*

$$\varphi^{k+1}(x) = \varphi^k(x) + \Delta t\, \delta_\alpha(\varphi^k(x))\, (\operatorname{sgn}(f(x) - f_{\text{iso}}) - \operatorname{sgn}_\alpha(\varphi^k(x)))$$

for all natural $k \geq 1$ with some $\varphi^0(x)$, $f(x)$, and a constant f_{iso}. Here,

$$\delta_\alpha(x) = \begin{cases} \dfrac{(\alpha - |x|)}{\alpha^2} & |x| < \alpha \\ 0 & \textit{else} \end{cases}, \qquad \operatorname{sgn}_\alpha(x) = \begin{cases} \dfrac{x\,(2\alpha - |x|)}{\alpha^2} & |x| < \alpha \\ \operatorname{sgn}(x) & \textit{else} \end{cases}.$$

Then, $\varphi^k(x) \to \varphi^\infty(x)$ as $k \to \infty$ pointwise for $\Delta t < \alpha^2$, where

$$\varphi^\infty(x) = \begin{cases} \alpha \operatorname{sgn}(f(x) - f_{\text{iso}}) & \textit{for } |\varphi^0(x)| < \alpha \\ \varphi^0(x) & \textit{else} \end{cases}.$$

Moreover, $\varphi^k(x)$ inherits the monotonicity of $\varphi^0(x)$ if $\Delta t < \alpha^2/3$.

Proof. For all x with $|\varphi^0(x)| \geq \alpha$, the main equation reads $\varphi^{k+1}(x) \equiv \varphi^k(x)$. Thus, the statement is trivial. Now, we assume $\operatorname{sgn}(f(x) - f_{\text{iso}}) \equiv 1$ for those x at which $0 \leq \varphi^0(x) < \alpha$. Other cases can be proven by analogy. The simplified equation reads

$$\varphi^{k+1}(x) = \varphi^k(x) + \frac{\Delta t}{\alpha^4}\,(\alpha - \varphi^k(x))\,(\alpha^2 - \varphi^k(x)\,(2\alpha - \varphi^k(x))) = \varphi^k(x) + \frac{\Delta t}{\alpha^4}\,(\alpha - \varphi^k(x))^3.$$

The following estimations for $\Delta t \leq \alpha^2$ and $0 \leq \varphi^k(x) < \alpha$ hold:

$$\varphi^{k+1}(x) - \varphi^k(x) = \frac{\Delta t}{\alpha^4}\,(\alpha - \varphi^k(x))^3 > 0;$$

$$\alpha - \varphi^{k+1}(x) = \alpha - \varphi^k(x) - \frac{\Delta t}{\alpha^4}\,(\alpha - \varphi^k(x))^3 = (\alpha - \varphi^k(x))\left[1 - \frac{\Delta t}{\alpha^4}\,(\alpha - \varphi^k(x))^2\right] \geq 0.$$

This shows that $\{\varphi^k(x)\}$ is a strictly monotonely increasing sequence bounded from above. Clearly, α is the minimal upper bound. Thus, the first part of the theorem is proven.

To show the monotonicity of $\varphi^k(x)$ with respect to x, we observe that for the function $f(x) = x + \Delta t(\alpha - x)^3/\alpha^4$

$$f'(x) = 1 - 3\Delta t(\alpha - x)^2/\alpha^4 \geq 0$$

if $\Delta t \leq \alpha^2/3$ and $0 \leq x < \alpha$. Therefore, $\varphi^{k+1}(x) \geq \varphi^{k+1}(y)$ if $\varphi^k(x) \geq \varphi^k(y)$. ◀

The regularized functions used in our experiments have a support that is two times larger than the support of the functions in the theorem statement. The substitution $\alpha = 2\epsilon$ leads to the estimation $\Delta t \leq 4\epsilon^2/3$, which is close to what one finds numerically.

References

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

2 Nina Amenta and Yong Joo Kil. Defining point-set surfaces. *ACM Trans. Graph.*, 23(3):264–270, 2004.

3 T. Belytschko, Y. Y. Lu, and L. Gu. Element-free galerkin methods. *Int. J. Numer. Meth. Engng*, 37:229–256, 1994.

4 Wolfgang Boehm and Hartmut Prautzsch. *Numerical methods.* VIEWEG, 1993.

5 D. Breen, R. Whitaker, K. Museth, and L. Zhukov. Level set segmentation of biological volume data sets. In *Handbook of Medical Image Analysis, Volume I: Segmentation Part A*, pages 415–478, New York, 2005. Kluwer.

6 T. F. Chan and L. A. Vese. Active contours without edges. *Image Processing, IEEE Transactions on*, 10(2):266–277, 2001.

7 C. S. Co and K. I. Joy. Isosurface Generation for Large-Scale Scattered Data Visualization. In G. Greiner, J. Hornegger, H. Niemann, and M. Stamminger, editors, *Proceedings of Vision, Modeling, and Visualization 2005*, pages 233–240. Akademische Verlagsgesellschaft Aka GmbH, November 16–18 2005.

8 R. A. Gingold and J. J. Monaghan. Smoothed particle hydrodynamics — theory and application to non-spherical stars. *Mon. Not. Roy. Astron. Soc.*, 181:375–389, November 1977.

9 Y. Jung, K. T. Chu, and S. Torquato. A variational level set approach for surface area minimization of triply-periodic surfaces. *J. Comput. Phys.*, 223(2):711–730, 2007.

10 Christian Ledergerber, Gaël Guennebaud, Miriah Meyer, Moritz Bächer, and Hanspeter Pfister. Volume mls ray casting. *IEEE Transactions on Visualization and Computer Graphics*, 14(6):1372–1379, 2008.

11 Aaron E. Lefohn, Joe M. Kniss, Charles D. Hansen, and Ross T. Whitaker. Interactive deformation and visualization of level set surfaces using graphics hardware. In *VIS '03: Proceedings of the 14th IEEE Visualization 2003 (VIS'03)*, page 11, Washington, DC, USA, 2003. IEEE Computer Society.

12 Aaron E. Lefohn, Joe M. Kniss, Charles D. Hansen, and Ross T. Whitaker. A streaming narrow-band algorithm: interactive computation and visualization of level sets. In *SIGGRAPH '05: ACM SIGGRAPH 2005 Courses*, page 243, New York, NY, USA, 2005. ACM.

13 D. Levin. Mesh-independent surface interpolation. *Geometric Modeling for Scientifc Visualization*, 2003.

14 Chunming Li, Chenyang Xu, Changfeng Gui, and M. D. Fox. Level set evolution without re-initialization: a new variational formulation. In *Computer Vision and Pattern Recognition, 2005. CVPR 2005. IEEE Computer Society Conference on*, pages 430–436 vol. 1, 2005.

15 Lars Linsen, Tran Van Long, Paul Rosenthal, and Stephan Rosswog. Surface extraction from multi-field particle volume data using multi-dimensional cluster visualization. *IEEE Transactions on Visualization and Computer Graphics*, 14(6):1483–1490, 2008.

16 L. B. Lucy. A numerical approach to the testing of the fission hypothesis. *The Astronomical Journal*, 82:1013–1024, December 1977.

17 Ken Museth, David E. Breen, Leonid Zhukov, and Ross T. Whitaker. Level-set segmentation from multiple non-uniform volume datasets. *Visualization Conference, IEEE*, 2002.

18 B. Nayroles, G. Touzot, and P. Villon. Generalizing the finite element method: Diffuse approximation and diffuse elements. *Comput. Mech.*, 10:307–318, 1992.

19 Stanley Osher and Ronald Fedkiw. *Level set methods and dynamic implicit surfaces.* Springer, 2003.

20 Sung Park, Lars Linsen, Oliver Kreylos, John D. Owens, and Bernd Hamann. A framework for real-time volume visualization of streaming scattered data. In Marc Stamminger and Joachim Hornegger, editors, *Proceedings of Tenth International Fall Workshop on Vision, Modeling, and Visualization 2005*, pages 225–232,507. DFG Collaborative Research Center, 2005.

21 Sung W. Park, Lars Linsen, Oliver Kreylos, John D. Owens, and Bernd Hamann. Discrete Sibson interpolation. *IEEE Transactions on Visualization and Computer Graphics*, 12(2):243–253, 2006.

22 Danping Peng, Barry Merriman, Stanley Osher, Hongkai Zhao, and Myungjoo Kang. A pde-based fast local level set method. *J. Comput. Phys.*, 155(2):410–438, November 1999.

23 Paul Rosenthal and Lars Linsen. Direct isosurface extraction from scattered volume data. In *Proceedings of Eurographics/IEEE-VGTC Symposium on Visualization*, pages 99–106, 2006.

24 Paul Rosenthal and Lars Linsen. Smooth surface extraction from unstructured point-based volume data using PDEs. *IEEE Transactions on Visualization and Computer Graphics*, 14(6):1531–1546, 2008.

25 Rudiger Westermann, Christopher Johnson, and Thomas Ertl. A level-set method for flow visualization. In *In Proceedings of Viz2000, IEEE Visualization*, pages 147–154, 2000.

26 R.T. Whitaker. Isosurfaces and level-sets. In C.D. Hansen and C.R. Johnson, editors, *The Visualization Handbook*, pages 97–123. Elsevier, 2005.

27 Hong-Kai Zhao, T. Chan, B. Merriman, and S. Osher. A variational level set approach to multiphase motion. *J. Comput. Phys.*, 127(1):179–195, 1996.

28 Hong-Kai Zhao, Stanley Oshery, Barry Merrimany, and Myungjoo Kangy. Implicit and non-parametric shape reconstruction from unorganized points using variational level set method. *Computer Vision and Image Understanding*, 80:295–319, 2000.

29 Matthias Zwicker, Jussi Räsänen, Mario Botsch, Carsten Dachsbacher, and Mark Pauly. Perspective accurate splatting. In *GI '04: Proceedings of Graphics Interface 2004*, pages 247–254, School of Computer Science, University of Waterloo, Waterloo, Ontario, Canada, 2004. Canadian Human-Computer Communications Society.