LAWRENCE
LIVERMORE
NATIONAL
LABORATORY

UCRL-JC-154014

A Multi-resolution Data Structure for Two-dimensional Morse Functions

P.-T. Bremer, H. Edelsbrunner, B. Hamann, and V. Pascucci

July 30, 2003

IEEE Visualization, Seattle, Washington, October 19-24, 2003

A Multi-resolution Data Structure for Two-dimensional Morse Functions

P.-T. Bremer*‡ H. Edelsbrunner† B. Hamann* V. Pascucci‡

* Visualization and Graphics Research Group
Center for Image Processing and Integrated Computing
Department of Computer Science
University of California, Davis, CA 95616-8562

† Department of Computer Science
Duke University
Box 90129
Durham, NC 27708

Center for Applied Scientific Computing
Lawrence Livermore National Laboratory
Box 808, L-560
Livermore, CA 94551

ABSTRACT

The efficient construction of simplified models is a central problem in the field of visualization. We combine topological and geometric methods to construct a multi-resolution data structure for functions over two-dimensional domains. Starting with the Morse-Smale complex we build a hierarchy by progressively canceling critical points in pairs. The data structure supports mesh traversal operations similar to traditional multi-resolution representations.

Keywords: Critical point theory, Morse-Smale complexes, terrains, simplification, multi-resolution data structure.

1 INTRODUCTION

This paper describes a multi-resolution data structure representing a continuous function over a two-dimensional domain. An example of such data is a terrain over a piece of the plane or over a sphere (e.g. the Earth). The distinguishing feature of this data structure is the fusion of topological and geometric measurements driving its construction.

Motivation Scientific data typically consists of measurements over a geometric domain or space. We think of it as a discrete sample of a continuous function over the space. In this paper we are interested in the case in which the space is a compact 2-manifold. Examples are the sphere and the torus, and either one of them can be obtained by compactifying a simply connected open region of the plane.

A multi-resolution representation is crucial in the efficient and preferably interactive exploration of scientific data. The traditional approach to constructing such a representation is based on progressive data simplification driven by a numerical measurement of the error. Alternatively, we may drive the simplification process with measurements of the topological features in the data. We refer to the former as the *geometric* and the latter as the *topological approach* to multi-resolution representations. The latter approach is appropriate if the topological features and their spatial relationships are essential to understand the phenomena under investigation. An example is water flow over a terrain, which is influenced by possibly subtle slopes. Small but critical changes in the landscape may result in catastrophic changes in water flow and accumulation. There are applications beyond the analysis of measured data. For example, we may artificially create a continuous function over a surface (e.g. describing a car-body) and use that function to guide the segmentation of the surface into patches.

Related work The topological analysis of scientific data has been a long standing research focus. Morse theory related methods were already developed in the late 19th century [Cayley 1859; Maxwell 1870] and later even hierarchical representations were proposed [Pfaltz 1976; Pfaltz 1979]. However, most of this research was lost and has been rediscovered only recently. Most modern research in the area of multi-resolution structures is geometry and many techniques have been developed during the last decade. The most successful algorithms developed in that era are based on edge contraction as the fundamental simplifying operation [Hoppe 1996; Popovic and Hoppe 1997] and accumulated square distances to plane constraints as the error measure [Garland and Heckbert 1997; Lindstrom and Turk 1998]. This work focussed on triangulated surfaces embedded in three-dimensional Euclidean space, which we denote as $\mathbb{R}^3$. We find a similar focus in the successive attempts to include the capability to change the topological type of the surface [El-Sana and Varshney 1998; He et al. 1996]. If we interpret the surface as the zero-set of a continuous function over $\mathbb{R}^3$ we may interpret such operations as smoothing or simplifying this function. This point of view is taken in a sequence of recent papers on the topic [Gerstner and Pajarola 2000; Guskov and Wood 2001; Ju et al. 2002], but the simplification is limited to a small neighborhood of the zero set. In this paper we widen the focus to the simplification of the entire function, which is equivalent to removing spurious topological features from all level sets simultaneously. To obtain a mathematical formulation of this process, we interpret the critical points of the function as the culprits responsible for topological features that appear in the level sets [Bajaj and Schikore 1998; Fomenko and Kunii 1997]. While sweeping through the level sets we see that critical points indeed start and end such features, and we may use the length of the interval over which a feature exists as a measure of its importance [Edelsbrunner et al. 2002]. We refer to this measure as the *persistence* of the two critical points delimiting the interval. In this view, the Morse-Smale complex of the function domain occupies a central position. Its construction and simplification is studied for 2-manifolds in [Edelsbrunner et al. 2002] and for 3-manifolds in [Edelsbrunner et al. 2003].

Results In this paper we follow the approach taken in [Edelsbrunner et al. 2002], with some crucial differences and extensions. Given a piecewise linear continuous function over a triangulated 2-manifold, we

1. construct a decomposition of the 2-manifold into monotonic quadrangular regions by connecting critical points with lines of steepest descent;
2. simplify the decomposition by performing a sequence of cancellations ordered by persistence; and
3. turn the simplification process into a hierarchical multi-resolution data structure whose levels correspond to simplified versions of the function.

The first two steps have been discussed in [Edelsbrunner et al. 2002], but the third step is new. Nevertheless, we have original contributions in all three steps and in the application of the data structure to concrete scientific problems. These contributions are

(i) a modification of the algorithm of [Edelsbrunner et al. 2002] that constructs the Morse-Smale complex without the use of handle slides;

(ii) the simplification of the complex by simultaneous application of independent cancellations;

(iii) a numerical algorithm to construct geometric realizations of the monotonic patches in the graph of the simplified function;

(iv) a low depth multi-resolution data structure combining the simplified versions of the function into a single hierarchy;

(v) an algorithm for traversing the data structure that combines different levels of the hierarchy to construct adaptive simplifications of the function; and

(vi) the application of our software to various scientific data sets and the visualization of the results.

The hallmark of our method is the fusion of the geometric and topological approaches to multi-resolution representations. The entire process is controlled by topological considerations, and the geometric method is used to realize monotonic paths and patches. The latter plays a crucial but sub-ordinate role in the overall algorithm.

2 BACKGROUND

We design essentially combinatorial algorithms based on intuitions provided by investigations of smooth maps. In this section, we describe the necessary background, first in Morse theory [Matsumoto 2002] and second in combinatorial topology [Alxeandrov 1998].

Morse functions. Throughout this paper, $\mathbb{M}$ denotes a compact 2-manifold without boundary and $f : \mathbb{M} \to \mathbb{R}$ denotes a real-valued smooth function over $\mathbb{M}$. Assuming a local coordinate system at a point $a \in \mathbb{M}$, we take two partial derivatives, $\frac{\partial f}{\partial x_1}(a)$ and $\frac{\partial f}{\partial x_2}(a)$. The point a is *critical* if both partial derivatives are zero and *regular* otherwise. Examples of critical points are maxima (f decreases in all directions), minima (f increases in all directions), and saddles (f switches between decreasing and increasing four times around the point).

Using again the local coordinates at a, we compute the *Hessian*, which is the matrix of second partial derivatives $\frac{\partial^2 f}{\partial x_i \partial x_j}(a)$, for $1 \leq i, j \leq 2$. A critical point is *non-degenerate* if the Hessian is non-singular, which is a property that is independent of the coordinate system. According to the Morse Lemma, it is possible to construct a local coordinate system such that f takes the form $f(x_1, x_2) = f(a) \pm x_1^2 \pm x_2^2$ in a neighborhood of a non-degenerate critical point. The number of minus signs is the *index* of a and distinguishes the different types of critical points: minima have index 0, saddles have index 1, and maxima have index 2. Technically, f is a *Morse function* if all its critical points are non-degenerate and have pairwise different function values. Much of the technical challenge in our work is rooted in the need to simulate these conditions for functions that do not satisfy them in a literal sense.

Morse-Smale complexes. Assuming an orthonormal local coordinate system, the *gradient* at a point a of the manifold is $\nabla f(a) = [\frac{\partial f}{\partial x_1}(a), \frac{\partial f}{\partial x_2}(a)]^T$. The set of gradients forms a smooth vector field on $\mathbb{M}$, with zeroes at the critical points of f. At any regular point we have a non-zero gradient vector, and if we follow that vector we trace out an *integral line*, which starts at a critical point and ends at a critical point while not containing either of them. Since integral lines ascend monotonically, the two endpoints cannot be the same. Because f is smooth, two integral lines are either disjoint or the same. The set of integral lines covers the entire manifold, except for the critical points.

The *descending manifold* of a critical point a is the set of points that flow toward a. More formally, it is the point a union all integral lines that end at a. For example, the descending manifold of a maximum is an open disk, that of a saddle is an open interval, and that of a minimum is the minimum itself. The collection of stable manifolds is a complex, in the sense that the boundary of a cell is the union of lower-dimensional cells. Symmetrically, we define the *ascending manifold* of a as the point a union all integral lines that start at a.

For the next definition, we need an additional non-degeneracy condition, namely that if ascending and descending manifolds intersect then they do that transversally. For example, if an ascending 1-manifold intersects a descending 1-manifold then they cross. Because of the disjointness of integral lines, this implies the crossing is a single point, namely the saddle common to both. Assuming this transversality property, we overlay the two complexes and obtain what we call the *Morse-Smale complex*, or MS complex, of f. Its cells are the connected components of the intersections between ascending and descending manifolds. Its vertices are the vertices of the two overlayed complexes, which are the minima and maxima of f, together with the crossing points of ascending and descending 1-manifolds, which are the saddles of f. Each 1-manifold is split at its saddle, thus contributing two arcs to the Morse-Smale complex. Each saddle is endpoint of four arcs, which alternately ascend and descend around the saddle. Finally, each region has four sides, namely two arcs emanating from a minimum and ending at two saddles and to additional arcs continuing from the saddles to a common maximum.

Piecewise linear functions. Functions are abundant in scientific problems, but they are rarely smooth and mostly known only at a finite set of points spread out over the manifold. It is convenient to assume the function is pairwise different at these points. We assume the points are the vertices of a triangulation K of $\mathbb{M}$, and we extend the function values by piecewise linear interpolation over the edges and triangles of K. The *star* $\mathrm{St}\, u$ of a vertex u consists of all simplices (vertices, edges and triangles) that contain u, and the *link* $\mathrm{Lk}\,(u)$ consists of all faces of simplices in the star that are disjoint from u. Since K triangulates a 2-manifold, the link of every vertex is a topological circle. The *lower star* contains all simplices in the star for which u is the highest vertex, and the *lower link* contains all simplices in the link whose endpoints are lower than u. Note that the lower link is the subset of simplices in the link that are faces of simplices in the lower star. Topologically, the lower link is a subset of a circle. We define what we mean by a critical point of a piecewise linear function based on the lower link. As illustrated in Figure 1, the lower link of a *maximum* is the entire link and that of a *minimum* is empty. In all other cases, the lower link of u consists

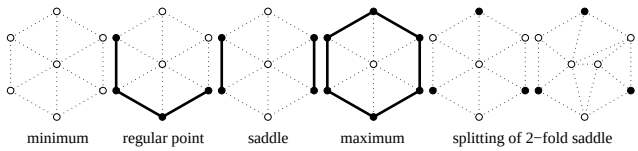


Figure 1: The classification of a vertex based on the relative height of the vertices in its link. The lower link is marked black.

of $k + 1 \geq 1$ connected pieces, each being an arc or possibly a single vertex. The vertex u is *regular* if $k = 0$ and a *k-fold saddle* if $k \geq 1$. As illustrated in Figure 1 for $k = 2$, a k-fold saddle can be split into k simple or 1-fold saddles.

Persistence We need a numerical measure of the importance of a critical point that drives the simplification of an MS complex.

Following the ideas described in [Edelsbrunner et al. 2002] critical points can only be removed in pairs (they cancel each other). The *persistence* of a cancellation is defined as the absolute difference in function value between the two critical points involved. The persistence is also directly related to the geometrical error (error in function space) as will be explained in Section 5.

3 MORSE-SMALE COMPLEX

We introduce an algorithm for computing the MS complex of the function f defined over the triangulation K. In particular, we compute the ascending/descending 1-manifolds (*paths*) of f starting from the saddles, and use them to partition K into quadrangular regions.

Construction of 1-manifolds Starting from each saddle, we construct two lines of steepest ascent and two lines of steepest descent. We do not adopt the original algorithm proposed in [Edelsbrunner et al. 2001] and choose to the follow actual lines of maximal slope instead of following the edges of K. In particular, we split triangles to create new edges in the direction of the maximal gradient.

Additionally, we avoid degenerate cases where the interior of a MS quadrilateral is not connected. This may happen even when following steepest ascent/descent lines because f is not smooth and integral lines can merge. Figure 2(a) shows one such case, where paths merge at *junctions* and split the interior of a MS quadrilateral into two regions. To address this problem we allow a pair of paths to join only if they are both ascending or descending. Figure 2(b) shows how this strategy avoids splitting MS quadrilaterals. If two paths are not allowed to merge we split a triangle such that the structure of the MS complex is preserved but locally we avoid the junction.

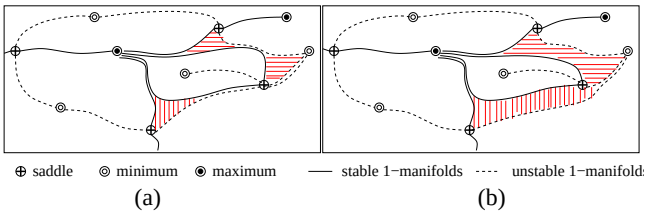


Figure 2: MS complex of a piecewise linear function. Since the gradient is not continuous ascending and descending 1-manifolds can share segments. (a) Complex with no restrictions on sharing segments. The vertically marked region touches only one saddle, the horizontally marked one is split into two components only connected by a 1-manifold. (b) Only 1-manifolds of the same type can share segments. All regions consist of a single connected component and touch both saddles.

After all paths have been computed, we partition K into quadrangular regions forming the cells of the MS complex. Each quadrangle is extracted from K with a simple region-growing strategy that starts from a triangle incident to a saddle and never crosses any 1-manifold.

Diagonals and diamonds The central element of our data structure for the MS complex is a (simple) saddle or, equivalently, the halves of the at most four quadrangles that share the saddle as one of their vertices. To be more specific about the halves recall that in the smooth category each quadrangle consists of integral lines that go from its minimum to its maximum. Any one of these integral lines can be chosen as *diagonal* to decompose the quadrangle into two triangles. The triangles sharing a given saddle form the *diamond* centered at the saddle. As illustrated in Figure 6(a), each diamond is a quadrangle whose vertices alternate between minima and maxima around the boundary. This includes the possibility that two vertices are the same and the boundary of the diamond is glued to itself along two consecutive diagonals.

Robustness We are careful in defining robust algorithms that alway produce consistent results. Especially in degenerate regions, where several vertices may have the same function value, the greedy choices of local steepest ascent/descent may not work consistently. For example, a 1-manifold can be lead into a "dead end" or can force endless splitting of the edges in the triangulation. We address this problem using a technique based on the *simulation of simplicity* [Edelsbrunner and Mücke 1990]. We mark each edge of K in the direction of ascending function value. Vertex indices are used to break ties on flat edges such that the resulting directed graph has no cycles. Now, the vertices of K can be treated as if they were in general position. The search for the steepest path is therefore transformed to a weighted-graph search. When searching for an ascending path only ascending edges or triangles with at least one ascending edge are considered. The function values are only used as preferences when they agree with the edge labels. In this way, our algorithm becomes stable even for highly degenerate data sets as the one shown in Figure 3.

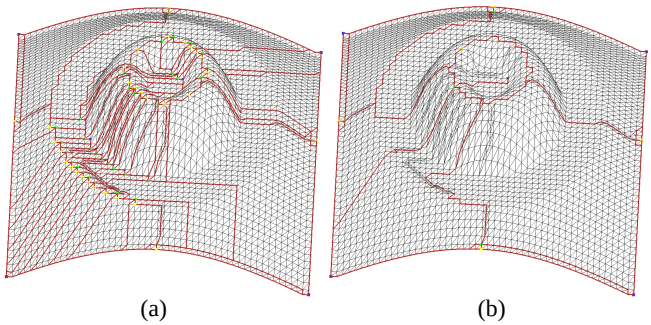


Figure 3: MS complex of degenerate data set. The "volcano" is created by a sin() function that is flat both inside the "crater" and at the foot of the mountain. (a) Originally computed MS complex. A large number of critical points is created by eliminating flat regions using simulation of simplicity. (b) The same complex after removal of "topological noise."

We compute the descending paths starting from the highest saddle and the ascending paths starting from the lowest saddle. Thus, if two paths aim for the same extremum, the one with higher persistence (our measure of importance) is computed first. The boundary of the data set is artificially tagged as a path. The complete algorithm is shown in Figure 4.

4 HIERARCHY

The main objective in this paper is the design of a hierarchical data structure that supports adaptive coarsening and refinement of the data. In this section, we first describe such a data structure and second show how to use it.

Cancellations We use only one atomic operation to simplify the MS complex of a function, namely a *cancellation* that eliminates two critical points. The inverse operation that creates two critical points is referred to as an *anti-cancellation*. In order to cancel two critical points they need to be adjacent in the MS complex. This leaves only two possible combinations: a minimum and a saddle or

```
T = {F, E, V};    //Triangulation, Faces, Edges, Vertices
M = {}; P = {}; C = {};    //quasi-MS complex, Paths, Cells
initializeArrows(T);    //Initializing simulation of simplicity
S = findSaddles(T);
S = splitMultiSaddles(T);
sortByHeight(S);
for (s in S ascending order)
    computeAscendingPath(P);
for (s in S descending order)
    computeDescendingPath(P);
while (f in F not touched) {
    growRegion(f,p0,p1,p2,p3) //pi bounding paths
    createMorseCell(C,p0,p1,p2,3); }
M = connectMorseCells(C) //M contains the final MS complex
```

Figure 4: Algorithm used to create an MS complex.

a saddle and a maximum. The two configurations are symmetric, so we can limit ourselves to discussing to the second case, which is illustrated in Figure 5. Let u be the saddle and v the maximum of

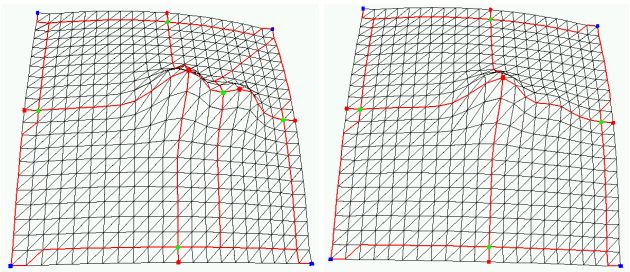


Figure 5: A portion of the graph of a function before (left) and after (right) canceling a maximum and a saddle.

the canceled pair and let w be the other maximum connected to u. We require $w \neq v$ and $f(w) > f(v)$ or else prohibit the cancellation of u and v. We view the cancellation as merging three critical points into one, namely u, v, w into w. The four paths ending at u are removed and the remaining paths ending at v are extended to w. The reason for requiring $f(w) > f(v)$ should be clear. First, it implies that all paths remain monotonic, except the paths extended from v to w, which will be fixed by numerical methods to be explained in Section 5. Second, we do not lose any level sets and only simplify the level sets between $f(u)$ and $f(v)$ by merging the component around v into the component surrounding w. We may think of a cancellation as deleting the two descending paths of u and contracting the two ascending paths of u. Alternatively, we may think of it as gluing the diagonals in pairs, thus zipping up the diamond centered at the saddle to a line, as illustrated in Figure 7.

Node removal We build the multi-resolution data structure from bottom to top. The bottom layer stores the MS complex of the function f, or rather the corresponding decomposition of the 2-manifold into diamonds. Figure 6(b) illustrates this layer by showing each diamond as a node with arcs connecting it to neighboring diamonds.

Each node has degree four, but there can be loops starting and ending at the same node. A cancellation corresponds to removing a node and re-connecting its neighbors. When this node is shared by four different arcs we can connect the neighbors in two different ways. As illustrated in Figure 7, this corresponds to the two different cancellations merging the saddle with the two adjacent maxima or the two adjacent minima. There is only one way to remove a node shared by a loop and two other arcs, namely to delete the loop and connect the two neighbors.

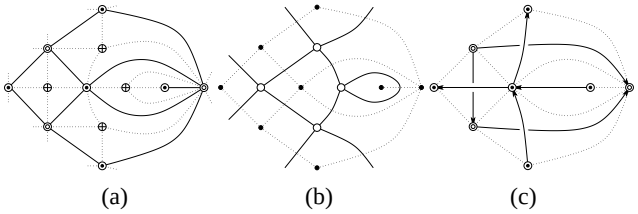


Figure 6: (a) The (dotted) portion of a Morse-Smale complex and the (solid) portion of the corresponding decomposition into diamonds. (b) The (solid) portion of the data structure representing the (dotted) piece of the decomposition into diamonds. (c) The (solid) cancellation dag of the (dotted) decomposition into diamonds.

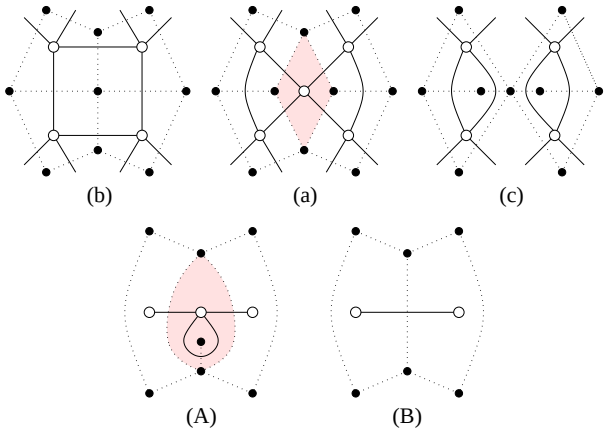


Figure 7: A four-sided diamond (a) can be zipped up in two ways: from top to bottom (b) or from left to right (c). A three-sided diamond (A) can be zipped up in only one way (B).

Hierarchy As mentioned earlier, we build a hierarchy from the MS complex by repeatedly canceling pairs of critical points. For example, we may use the algorithm in [Edelsbrunner et al. 2002] to pair up the critical points as $(s_1, v_1), (s_2, v_2), \ldots, (s_k, v_k)$, with persistence increasing from left to right. Let Q_j be the MS complex obtained after the first j cancellations, for $0 \leq j \leq k$. We get Q_{j+1} by modifying Q_j and leaving enough information behind so we can recover Q_j from Q_{j+1}. The hierarchy is complete when we reach Q_k.

Call each Q_j a *layer* in the hierarchy and represent it by activating its diamonds as well as neighbor and vertex pointers and de-activating all other diamonds and pointers. To ascend in the hierarchy (coarsen the quadrangulation) we de-activate the diamond of s_{j+1} and to descend in the hierarchy (refine the quadrangulation) we activate the diamond of s_{j-1}. Both activating and de-activating a diamond consists of updating a constant number of pointers.

Independent cancellations We generalize the notion of a layer in the hierarchy to permit view-dependent simplifications of the data. The key concept here is the interchangeability of cancellations. We will see shortly that the most severe limitation to interchanging cancellations derives from the assignment of extrema as vertices of the diamonds and from re-drawing the paths ending at these extrema. To understand this limitation, we introduce the *cancellation dag* (directed acyclic graph) whose vertices are the minima and maxima. Figure 6(c) shows an example of such a dag. For each diamond there is a directed edge (a di-edge) from the higher to the lower minimum and another di-edge from the lower to the higher maximum. We have no loops and therefore sometimes only one di-edge in a diamond. The di-edges connecting the maxima in-

crease in height, and similarly the di-edges connecting the minima decrease in height. It follows that the cancellation dag is indeed acyclic. Zipping up a diamond corresponds to contracting one of its di-edges and deleting the other, if any. The end-point of the di-edge remains as a vertex and the start-point disappears. It follows that the diamonds that share the start-point now receive a new vertex. A special case arises when a diamond shares both, start- and end-point. The connecting di-edge would then turn into a loop and is therefore deleted. We now come back to our original objective of exploring view-dependent simplifications. Two cancellations in a (possibly already simplified) MS complex are *independent* if there is no difference between the data structure generated by the two orderings of the operations. Otherwise, the two cancellations are *dependent.* For example the two cancellations zipping up the same diamond are dependent since one pre-empts the other. In general, two cancellations are dependent if their diamonds share a vertex. We choose this definition of dependence because of the paths that need to be re-drawn whenever we zip up a diamond. The reason we are interested in independent cancellations should be obvious: they offer some freedom in how we choose a layer in the hierarchy and thus permit the adaptation of the representation to external constraints, such as the biased view of the data.

Balance The more independent cancellations we can find the more freedom we have in generating layers in the multi-resolution data structure. Ideally, we identify a large independent set and iterate to construct a shallow hierarchy. In the worst case, every pair of cancellations is dependent, which contradicts the existence of a shallow hierarchy. As illustrated in Figure 9(b), such a configuration exists even for the sphere and for any arbitrary number of vertices. The three-sided diamonds are necessary for constructing such examples. Specifically, we can prove that every MS complex without three-sided diamonds has a large independent set of cancellations.

5 Geometric Approximation

After each cancellation we must create or change the local geometry such that it matches the simplified topology. There exist several objectives for this geometry: (1) The approximation must agree with the given topology. (2) The error between the simplified and the original geometry should be small. (3) To achieve a visual pleasing rendering the approximation should be as smooth as possible.

Error bounds For this part of the algorithm we use vertical distance of the function values at vertices as error. The persistence of a cancellation implies a minimal error between original and simplified geometry. A one-dimensional example is shown in Figure 8(a). The persistence p of the middle maximum-minimum pair is defined by their difference in function value. Ignoring special cases, it is easy to see that any monotone approximation has an error of at least $p/2$. The approximation with minimal change to the original function values is shown in red in Figure 8(a). However, the flat section, and the fact that this approximation is only C^0 continuous, will produce visually unpleasing results. A smoother approximation within the same error bound is shown in blue. Nevertheless, without relaxing the error further a flat segment cannot be avoided. Once a larger error bound than $p/2$ is used even smoother approximations are possible as shown in green. It is important to note that one can always directly construct an approximation with minimal error. In the two dimensional case the same error bounds hold but the approximation with minimal changes to the function values is only C^{-1} continuous.

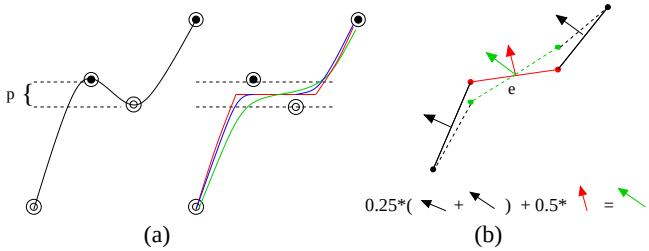


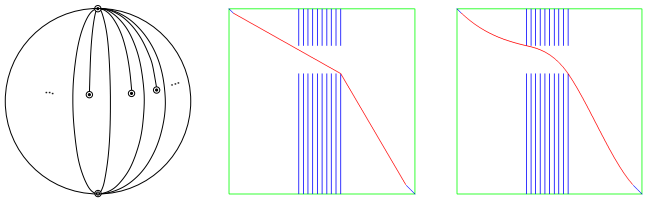
Figure 8: Geometry fitting for 1-manifolds: (a) A one-dimensional cancellation and several monotone approximations. (b) Local averaging scheme used to smooth. The slopes of the neighboring edges are combined with the original slope, and the function values are adjusted accordingly. (For better understanding the edge "normals" are shown.)

Data fitting Since it is guaranteed that beyond a certain error a monotone solution exists to "fill in" the interior geometry of each quadrilateral, the goal is to find an "optimal" approximation (based on some quality measure). A large body of literature exists on the subject of constrained smoothing splines [Greiner 1991; Carlson and Fritsch 1985]. The general problem is to construct the smoothes interpolant to a set of input data while observing some shape constraints (e.g., convexity, monotonicity, and boundary conditions). However, most published work uses weighted errors rather than absolute error bounds. Additionally, the techniques are typically described for tensor product setting, and the definitions of monotonicity for the bivariate case vary and differ from the one we use. To adapt these techniques to fit geometry to given topology will be the subject of future work.

We use a multi-stage iterative approach to fit geometry. It provides a smooth C^1-continuous approximation within a given error bound along paths. In the interior of regions we use a Laplacian smoothing [Taubin 1995] with boundary constraints, which is not guaranteed to observe the error bound. The solution for the paths is also based on iterative smoothing. Rather than averaging function values directly, we average gradients. An example is shown in Figure 8(b). We consider a single edge along a path: First, we compute the gradient along the path for both neighboring edges. We compute a target gradient by averaging the gradient of the current edge with the gradients of the neighboring edges. Second we adjust the function values at both end points such that the resulting edge has the target gradient.

During these modifications we constrain the vertices to lie within the error bound and the complete path to be monotone. Additionally, the gradient at critical points is set to zero. The constraints are included such that initial configurations not subject to these constraints are guaranteed to observe the constraints after a sufficient number of iterations. Two examples of this technique are shown in Figure 9. The procedure performs well in practice, even though it converges slowly. The interiors of regions are smoothed using standard Laplacian smoothing [Taubin 1995] applied to the function values. At each iteration, the function value at a vertex is replaced by the average between its old values and the average of all its neighbors. Since the boundaries are monotone, this approach is guaranteed to converge to a monotone solution.

The complete process can be summarized as follows: Find all paths affected by a cancellation; use the one-dimensional gradient smoothing to geometrically remove the canceled critical points; smooth the old regions until they are monotone; erase the paths and re-compute new paths using the new geometry; as new paths might violate the error bound use one-dimensional gradient smoothing again to force the new paths to comply with the constraints; and, finally, smooth the new regions until all points are regular.



(a) (b)

Figure 9: (a) A MS complex on the sphere with k diamonds and k pairwise dependent cancellations. (b) Constrained one-dimensional smoothing: Shown in blue are constraints on end point derivatives and error bounds. (left) Initial configuration; (right) limit curve subject to blue constraints and monotonicity requirements.

6 REMESHING

We must remesh each quadrangle and create a consistent (crack-free) triangulation across neighboring quadrangles. Starting from triangulated four-side regions with polygons as boundary curves, we create a regular triangulation that approximates the original function values within a user-defined error bound. The goal is to create a triangulation whose connectivity is that of a uniformly subdivided square obtained via application of repeated longest-edge bisection.

Parametrization The initial stage of remeshing creates a parametrization for each quadrangle. We are using mean value coordinates as proposed by Floater [2003] to parametrize each quadrangle over the unit square. Each vertex in K is expressed as a convex sum of its neighbors. The necessary weights can be computed by solving a sparse linear system. We map the four boundary curves by arc-length to the four edges of the unit square. Given a parametrization for the boundary the convex combinations uniquely define a parametrization for the complete quadrangle.

Next, we sample the parameter space on a uniform grid using $2^n + 1$ samples on each boundary curve. We create a new mesh for each quadrangle by mapping the samples back to physical space. The grid resolution is defined by some given error bound. Paths follow locally the maximal gradient direction. Therefore, we define the error on the boundary alone. We sample each boundary until the error bound (Hausdorff-distance) is met and choose the grid resolution to agree with the maximum of the four boundary polygons resolutions.

Avoiding cracks To avoid additional dependencies the rendering of each quadrangle must be independent of the neighbor quadrangles. The global triangulation is crack-free when each polygon bounding a quadrangle (each path of the complex) is rendered without cracks. There are two potential sources of cracks between quadrangles.

Neighboring quadrangles could be sampled at different resolutions. Therefore, two neighbor quadrangles might render the same polygonal boundary using a different number of samples, creating t-junctions in the triangulation. We address this problem by first sampling all paths until they meet the error bound. We force all additional samples to lie on the linear interpolation of its neighboring samples. This approach ensures that existing t-junctions create no visible physical cracks.

The second source of potential cracks are junctions. As described in Section 3, quadrangles can be geometrically partially degenerate as a result of allowing paths to share common edges. By construction, each junction is a corner of the geometrical representation of at least one quadrangle. Each corner is rendered independently of an adaptive rendering scheme (as long as the quadrangle is visible). Hence, each junction must always be rendered. This fact results in two properties: (1) Each junction must be a sample point of the grid, and (2) when defining screen space errors we must ensure that junctions are always rendered. The first problem is addressed by modifying the parametrization along polygons. Rather than sampling a complete path using its arc-length parametrization, we only consider segments between junctions and/or critical points. Each segment is sampled using a standard bisection procedure, first adding a sample at arc-length value 0.5, then two samples at 0.25 and 0.75, etc., until the error bound is met. Each boundary curve consists of one or more such segments. A boundary curve must be sampled using $2^n + 1$ samples, and the sampling of the segments must be observed.

Data layout and rendering Rather than storing a triangulation for each quadrangle explicitly, we use regular grids. This approach allows us to use the methods described in [Linstrom and Pascucci 2002] for rendering purposes. By storing each grid in two interleaved embedded quadtrees we avoid having to store any connectivity information, while maintaining high flexibility during rendering. As described by Lindstrom and Pascucci [Linstrom and Pascucci 2002], this framework can be extended easily to adaptive, view-dependent rendering, as well as efficient view frustum culling and geomorphing. One disadvantage of this data layout is a 33% memory overhead. Additionally, we waste memory by representing partial boundary quadrangles as general quadrangles. Another important aspect is the definition of local error coefficients. As we are working with many smaller grids, rather than a single high resolution one, we must ensure consistency across boundaries. We ensure that samples on grid boundaries are shared. This approach also guarantees that their respective error terms agree. Therefore, a sample along a path is either rendered from both adjoining quadrangles or neither one. Concerning the rendering of junctions, we set their error terms to infinity. (For a more detailed discussion of the different error terms possible and their computation we refer to [Linstrom and Pascucci 2002].)

7 RESULTS

We tested our algorithm on the Puget Sound terrain data set at resolution 1025-by-1025 and elevation values represented with two-byte unsigned integers. We also used the temperature field of a combustion process simulated at resolution 512-by-512 with temperature values represented by single-byte unsigned integer values. All the tests were performed on a 1.8GHz Pentium 4 Linux PC with 1Gb of main memory.

A straight-forward application of our algorithm is the removal of topological noise without smoothing the data. This functionality is not dependent on the hierarchy since can be implemented by pure repeated cancellation of the critical points with lowest persistence. This noise removal stage should always be applied, even if the persistence threshold is set to zero, to remove at least the topology introduced by the symbolic perturbation. The top of Figure 10 shows the effect of this procedure. The original MS complex (on the left) with 2859 critical points is compared with an approximation (on the right) where all topological features with persistence below 0.1% of the temperature range were removed. When it is not necessary to modify the function values (fit the geometry) this operation can be performed in about one second.

Several resolutions of the Puget Sound data set are shown at the middle-bottom of Figure 10. After removal of all topology with persistence below 0.5% of the maximum elevation the approximation has only 4045 critical points of the 49185 in the original topology (too dense to be shown). These points determine the highest-resolution MS complex (middle-left). The middle-right fig-

RMS error and num. triangles for approximations with guaranteed geometric error and persistence

Combustion Process		persistence (absolute function value) / number of critical points				
		0.00100 / 446	0.01000 / 343	0.05000 / 263	0.50000 / 156	/
max error geometry	0.010	0.00173 / 71126	0.00209 / 72346	0.00247 / 69592	0.02058 / 56881	/
	0.020	0.00228 / 38962	0.00260 / 39869	0.00297 / 38578	0.02067 / 31802	/
	0.050	0.00495 / 17023	0.00497 / 17245	0.00533 / 16619	0.02122 / 13711	/

Puget Sound		Persistence (absolute function value) / Number of Critical Points				
		0.00125 / 4045	0.0025 / 2408	0.00500 / 1408	0.0100 / 677	0.02000 / 364
max error geometry	0.005	0.00102 / 81809	0.00125 / 79298	0.00159 / 77503	0.0022 / 71924	0.00294 / 73099
	0.010	0.00122 / 53494	0.00145 / 49680	0.00177 / 46680	0.0023 / 41881	0.00303 / 43574
	0.030	0.00152 / 41288	0.00172 / 36558	0.00203 / 31810	0.0026 / 24378	0.00326 / 23862

Table 1: Approximation errors and number of faces used for different persistence and geometric error bounds.

ure shows an approximation with 2025 critical points and a uniform persistence of about 1.2%. The bottom-left shows the coarsest MS complex at 50% persistence with 289 critical points. On the bottom-right is a view-dependent adaptive refinement based on the purple view frustum, which yield an approximation with 1070 critical points. The full-resolution topology is preserved inside the frustum, while outside only the minimal dependent topology is maintained. Note how the topology can drop quickly from the highest to the lowest resolution while maintaining a consistent mesh.

The pre-processing of the Puget Sound data set took about three and a half hours, mainly due to the slow convergence of the geometric fitting procedures. The traversal of the hierarchy as well as rendering are fully interactive (see animation at http://graphics.cs.ucdavis.edu/ ptb/MorseComplexResults).

Table 1 shows the measured RMS errors between the geometric approximation at several levels of resolution compared to the original data sets. The error measures have been generated with the Metro tool [Cignoni et al. 1998] with disabled vertex sampling as the t-junctions in our meshes seemed to cause numerical problems. Approximations for several persistence bounds are compared to the original data and, for the same persistence, with several adaptive meshes. The error used for the meshing procedure is vertical distance at the vertices.

8 Conclusions

We have described a new topology-based multi-resolution hierarchical data structure for functions over 2-manifolds and demonstrated its use for two-dimensional height fields. The hierarchy allows one to extract geometry adaptively for a given topological error. Due to its robustness in the presence of topological noise and well defined simplification procedures, it is appealing for applications using topological analysis, for example, data segmentation and feature detection and tracking. Future work will be concerned with fitting the complete geometry within a given error bound and the extension to volumetric datasets.

Acknowledgments

This work was performed under the auspices of the U. S. Department of Energy by University of California Lawrence Livermore National Laboratory under contract No. W-7405-Eng-48.

B. Hamann is supported by the National Science Foundation under contract ACI 9624034 (CAREER Award), through the Large Scientific and Software Data Set Visualization (LSSDSV) program under contract ACI 9982251, and through the National Partnership for Advanced Computational Infrastructure (NPACI); the National Institute of Mental Health and the National Science Foundation under contract NIMH 2 P20 MH60975-06A2; the Lawrence Livermore National Laboratory under ASCI ASAP Level-2 Memorandum Agreement B347878 and under Memorandum Agreement B503159; H. Edelsbrunner is partially supported by NSF under grants EIA-99-72879 and CCR-00-86013.

The combustion data set is courtesy of Echekki and Chen [Echekki and Chen 2003] and shows the numerical simulation of the autoignition of a spatially non-homogeneous hydrogen-air mixture.

References

Alxeandrov, P. S. 1998. *Combinatorial Topology*. Dover, New York.

Bajaj, C. L., and Schikore, D. R. 1998. Topology preserving data simplification with error bounds. *Computers and Graphics 22*, 1, 3–12.

Carlson, R., and Fritsch, F. N. 1985. Monotone piecewise bicubic interpolation. *Journal on Numerical Analysis 22* (Apr.), 386–400.

Cayley, A. 1859. On contour and slope lines. *The London, Edinburgh and Dublin Philosophical Magazine and Journal of Science XVIII*, 264–268.

Cignoni, P., Rocchini, C., and Scopigno, R. 1998. Metro: Measuring error on simplified surfaces. *Computer Graphics Forum 17(2)* (June), 167–174.

Echekki, T., and Chen, J. H. 2003. Direct numerical simulation of autoignition in non-homogeneous hydrogen-air mixtures. *Combust. Flame*. to appear.

Edelsbrunner, H., and Mücke, E. P. 1990. Simulation of simplicity: A technique to cope with degenerate cases in geometric algorithms. *ACM Trans. Computer Graphics 9*, 66–104.

Edelsbrunner, H., Harer, J., and Zomorodian, A. 2001. Hierarchical Morse-Smale complexes for piecewise linear 2-manifolds. In *Symp. on Computational Geometry*, ACM Press, New York, NY, USA, ACM, 70–79.

Edelsbrunner, H., Letscher, D., and Zomorodian, A. 2002. Topological persistence and simplification. *Discrete Comput. Geom. 28*, 511–533.

Edelsbrunner, H., Harer, J., Natarajan, V., and Pascucci, V. 2003. Morse-Smale complexes for piecewise linear 3-manifolds. In *Proc. 19th Sympos. Comput. Geom.*, 361–370.

El-Sana, J., and Varshney, A. 1998. Topology simplification for polygonal virtual environments. *IEEE Transactions on Visualization and Computer Graphics 4*, 2, 133–144.

Floater, M. S. 2003. Mean value coordinates. *Computer Aided Geometric Design 20*, 19–27.

Fomenko, A. T., and Kunii, T. L., Eds. 1997. *Topological Modeling for Visualization*. Springer-Verlag.

Garland, M., and Heckbert, P. S. 1997. Surface simplification using quadric error metrics. In *Proceedings of ACM SIGGRPAH 1997*, ACM Press / ACM SIGGRAPH, New York, NY, USA, T. Whitted, Ed., vol. 31, ACM, 209–216.

Gerstner, T., and Pajarola, R. 2000. Topology preserving and controlled topology simplifying multiresolution isosurface extraction. In *Proc. Visualization 2000*, IEEE Computer Society Press, Los Alamitos California, T. Ertl, B. Hamann, and A. Varshney, Eds., IEEE, 259–266.

Greiner, H. 1991. A survey on univariate data interpolation and approximation by splines of given shape. *Math. Comput. Model. 15*, 97–106.

Guskov, I., and Wood, Z. 2001. Topological noise removal. In *Proceedings of Graphics Interface 2001*, B. Watson and J. W. Buchanan, Eds., 19–26.

He, T., Hong, L., Varshney, A., and Wang, S. W. 1996. Controlled topology simplification. *IEEE Trans. on Visualization and Computer Graphics 2*, 2, 171–184.

Hoppe, H. 1996. Progressive meshes. *Computer Graphics (Proc. SIGGRAPH '96) 30*, 4 (Aug.), 99–108.

Ju, T., Losasso, F., Schaefer, S., and Warren, J. 2002. Dual contouring of hermite data. *ACM Trans. on Graphics (Proc. SIGGRAPH '02) 21*, 3, 339–346.

Lindstrom, P., and Turk, G. 1998. Fast and memory efficient polygonal simplification. In *Proc. IEEE Visualization '98*, IEEE Computer Society Press, Los Alamitos California, IEEE, 279–286.

Linstrom, P., and Pascucci, V. 2002. Terrain simplification simplified: A general framework for view-dependent out-of-core visualization. *IEEE Transactions on Visualization and Computer Graphics 8*, 3 (July-September), 239–254.

Matsumoto, Y. 2002. *An Introduction to Morse Theory*. American Mathematical Society.

Maxwell, J. C. 1870. On hills and dales. *The London, Edinburgh and Dublin Philosophical Magazine and Journal of Science XL*, 421–427.

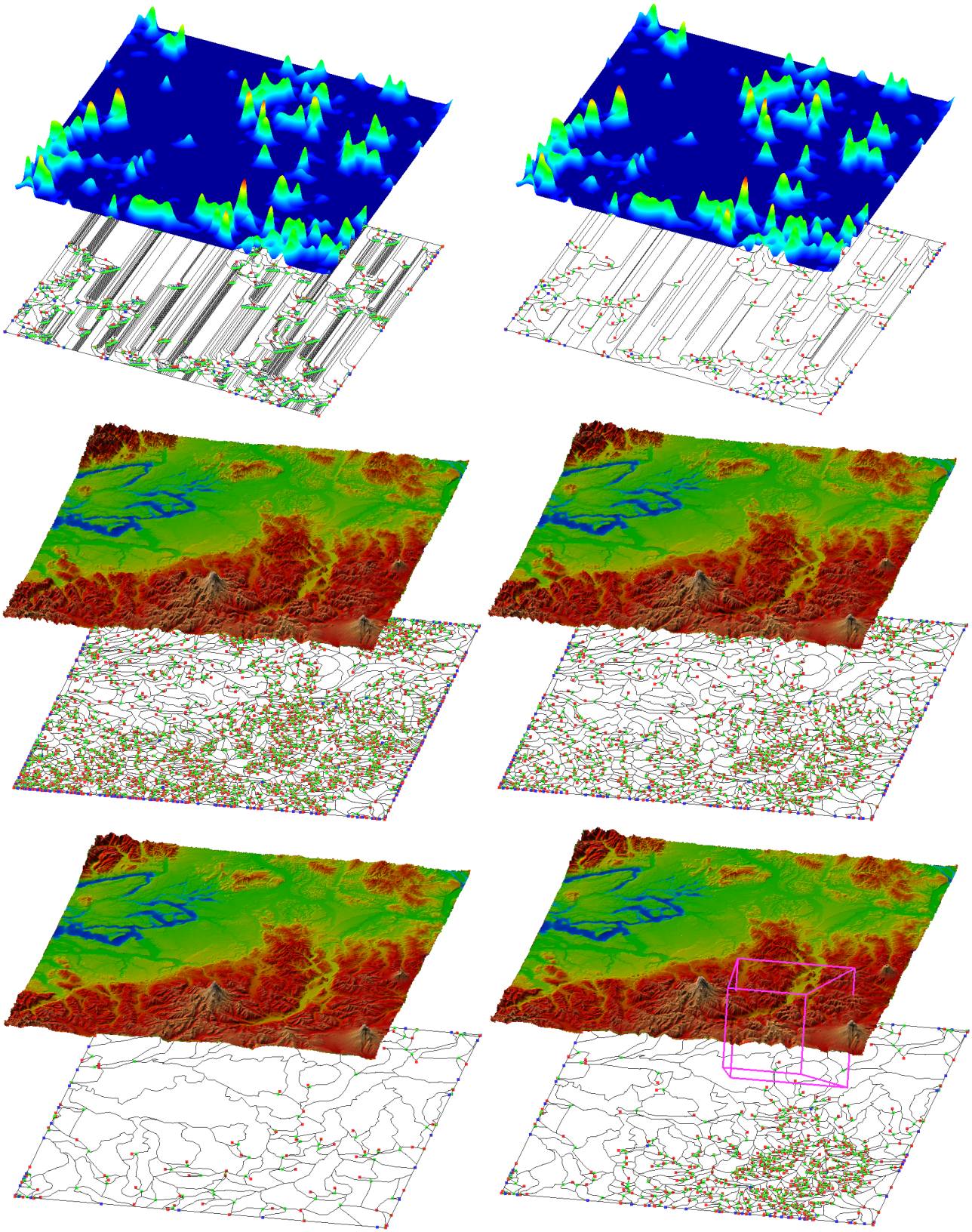


Figure 10: (Top-left) Original MS complex of a data set of a combustion process (2859 critical points(. A textured rendering is shown together with the corresponding MS complex. Maxima are shown in red, minima in blue, saddles in green, and paths in black; (top-right) after removal of all topology with persistence smaller than 0.1% of the temperature range (446 critical points); (middle-left) Puget Sound data set after topological noise removal; (middle-right) the data set at persistence of 1.2%; (bottom-left) only topology with persistence larger than 20% of max height; (bottom-right) view-dependent refinement - current frustum shown in purple;

PFALTZ, J. 1976. Surface networks. *Geographical Analysis 8*, 77–93.

PFALTZ, J. 1979. A graph grammar that describes the set of two-dimensional surface networks. *Graph-Grammars and Their Application to Computer Science and Biology (Lecture Notes in Computer Science no. 73*.

POPOVIC, J., AND HOPPE, H. 1997. Progressive simplicial complexes. In *Proceedings of ACM SIGGRPAH 1997*, ACM Press / ACM SIGGRAPH, New York, NY, USA, T. Whitted, Ed., vol. 31, ACM, 209–216.

TAUBIN, G. 1995. A signal processing approach to fair surface design. *Computer Graphics (Proc. SIGGRAPH' 95)*, 351–358.