Preprint
UCRL-JC-154014

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

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

This article was submitted to IEEE Visualization

U.S. Department of Energy

Lawrence
Livermore
National
Laboratory

Aug. 1th, 2003

DISCLAIMER

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

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

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

*Center for Image Processing and Integrated Computing
University of California, Davis, CA 95616

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

‡Center for Applied Scientific Computing
Lawrence Livermore National Laboratory, Livermore, CA 94551

Abstract

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 construct a topological hierarchy by progressively canceling critical points in pairs. Concurrently, we create a geometric hierarchy by adapting the geometry to the changes in topology. The data structure supports mesh traversal operations similarly to traditional multi-resolution representations.

CR Categories: I.3.5 [Computer Graphics]: Computational Geometry and Object Modeling—Hierarchy and geometric transformations

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

1 Introduction

The efficient construction of simplified geometric models is a central problem in visualization. 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 planar domain 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 often consists of measurements over a geometric domain or space. We think of it as a discrete sample of a continuous function over the space. We are interested in the case in where the space is a compact 2-manifold. Examples of such data are elevation data and the electrostatic potential on a molecular surface.

A multi-resolution representation is crucial for 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 topological features. 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 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 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 have already been developed in the 19th century [Cayley 1859; Maxwell 1870], long before Morse theory itself was formulated, and hierarchical representations have been proposed [Pfaltz 1976; Pfaltz 1979] without making use of the mathematical framework developed by Marston Morse and others [Morse 1925; Milnor 1963]. However, most of this research was lost and has been rediscovered only recently. Most modern research in the area of multi-resolution structures is geometric 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 focused 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 [He et al. 1996; El-Sana and Varshney 1998]. 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 was 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. We extend the focus to the simplification of an 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 [Fomenko and Kunii 1997; Bajaj and Schikore 1998]. 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. For the special case of two-dimensional height fields this measure was first proposed by Horman [1971] and later adopted by Mark [1977]. We use the more general concept of persistence introduced in [Edelsbrunner et al. 2002]. 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. 2001] and for 3-manifolds in [Edelsbrunner et al. 2003].

Results We follow the approach taken in [Edelsbrunner et al. 2001], 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 the construction of a hierarchical multi-resolution data structure whose levels correspond to simplified versions of the function.

The first two steps are discussed in [Edelsbrunner et al. 2001], but the third step is new. Nevertheless, this paper makes original contributions to 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. 2001] 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 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; and

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

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 describe an essentially combinatorial algorithm based on intuitions provided by investigations of smooth maps. In this section, we describe the necessary background, in Morse theory [Milnor 1963; Matsumoto 2002] and in combinatorial topology [Munkres 1984; Alexandrov 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 compute two partial derivatives and call *a critical* when both 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. A critical point is *non-degenerate* when 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* when all its critical points are non-degenerate and have pairwise different function values. Most of the challenges in our method are rooted in the need to simulate these conditions for functions that do not satisfy them in a literal sense.

Morse-Smale complexes Assuming a Riemannian metric and an orthonormal local coordinate system, the *gradient* at a point a of the manifold is the vector of partial derivatives. The set of gradients forms a smooth vector field on $\mathbb{M}$, with zeroes at the critical points. At any regular point we have a non-zero gradient vector, and when 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 union of a and 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 union of a and all integral lines that start at a.

For the next definition, we need an additional non-degeneracy condition, namely that ascending and descending manifolds that intersect do so transversally. For example, if an ascending 1-manifold intersects a descending 1-manifold then they cross. Due to the disjointness of integral lines, this fact implies that 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 two additional arcs continuing from the saddles to a common maximum. It is generically possible that the two saddles are the same, in which case two of the four arcs merge into one.'

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 that the function has pairwise different values at these points. We assume that 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* of a vertex u consists of all simplices (vertices, edges and triangles) that contain u, and the *link* 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 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.

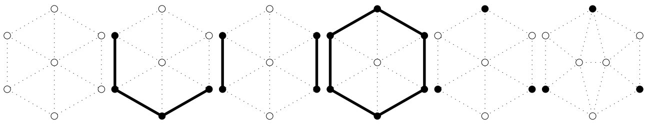


Figure 1: The classification of a vertex based on the relative height of the vertices in its link. The lower link is marked black.

Persistence We need a numerical measure of the importance of critical points that can be used to drive the simplification of a Morse-Smale complex. For this purpose, we pair critical points and use the absolute difference between their function values as importance measure. To construct the pairing, we imagine sweeping the 2-manifold $\mathbb{M}$ in the direction of increasing function value. This view is equivalent to sorting the vertices by function value and incrementally constructing the triangulation K of $\mathbb{M}$ one lower star at a time. The topology of the partial triangulation changes whenever we add a critical vertex, and it remains unchanged whenever we add a regular vertex. Barring some exceptional cases that have to do with the surface type of $\mathbb{M}$, each change either *creates* a component or an annulus or it *destroys* a component (by merging two) or an annulus (by filling the hole). We pair a vertex v that destroys with the vertex u that created what v destroys. The *persistence* of u and of v is the time-lag between the two events: $p = f(v) - f(u)$. An algebraic justification of this definition and a fast algorithm for constructing the pairs can be found in [Edelsbrunner et al. 2002].

3 Morse-Smale complex

We introduce an algorithm for computing the MS complex of a function f defined over a triangulation K. In particular, we compute the ascending and 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 follow actual lines of maximal slope instead of edges of K. In particular, we split triangles to create new edges in the direction of the gradient.

Additionally, we avoid degenerate cases where the interior of a region is not connected. This situation can happen even when following steepest ascent/descent lines since f is not smooth and integral lines can merge. Figure 2(a) shows one such case, where paths merge at *junctions* and disconnect the interior of a region into two. To deal with this problem we allow a pair of paths to join only when they are both ascending or descending. Figure 2(b) shows how this strategy avoids disconnected regions.

After computing all paths, 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 the neighborhood of a simple saddle or, equivalently, the halves of the quadrangles that share the saddle as one of their vertices. To be more specific about the halves, recall that in the smooth case each quadrangle consists of integral lines that emanate from its minimum and end at 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

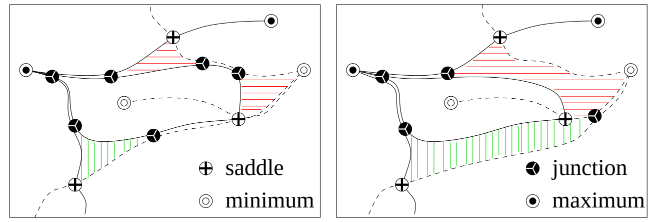
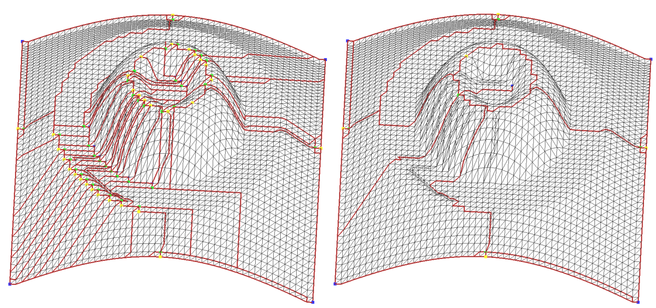


Figure 2: Portion of the MS complex of a piecewise linear function. Since the gradient is not continuous, (solid) ascending and (dotted) descending 1-manifolds can meet in junctions and share segments. (Left) Complex with no restrictions on sharing segments. The green region touches only one saddle, and the red one is disconnected. (Right) Only 1-manifolds of the same type can meet. The interior of each region is connected and touches both saddles.

Figure 6(a), each diamond is a quadrangle whose vertices alternate between minima and maxima around the boundary. It is possible that two vertices are the same and the boundary of the diamond is glued to itself along two consecutive diagonals.

Robustness We carefully design a global algorithm that always produces consistent results. Especially in degenerate areas of $\mathbb{M}$, 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 edges in the triangulation. We address this problem using a technique based on the *simulation of simplicity* [Edelsbrunner and Mücke 1990]. We orient 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. Thus, our algorithm is stable even for highly degenerate data sets as the one shown in Figure 3.



(a) (b)

Figure 3: MS complex of degenerate data set. The volcano 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 removing what we call topological noise.

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

$T = \{F, E, V\}$; //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);
$\forall s \in S$ in ascending order: computeAscendingPath(P);
$\forall s \in S$ in descending order: computeDescendingPath(P);
while (f in F not touched)
    growRegion(f,p_0,p_1,p_2,p_3); //p_i are bounding paths
    createMorseCell(C,p_0,p_1,p_2,p_3);
M = connectMorseCells(C);

Figure 4: Sequence of high-level operations used to create an MS complex.

4 Hierarchy

Our main objective is the design of a hierarchical data structure that supports adaptive coarsening and refinement of the data. In this section, we describe such a data structure and discuss 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 must be adjacent in the MS complex. Only two possible combinations arise: a minimum and a saddle or a saddle and a maximum. The two configurations are symmetric and we can limit the discussion to the second case, which is illustrated in Figure 5. Let u be the saddle and v the maximum of the canceled pair,

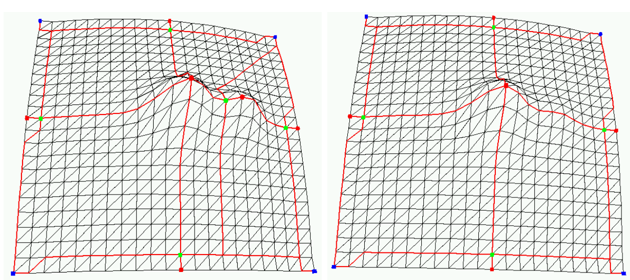


Figure 5: A portion of the graph of a function before (left) and after (right) canceling a maximum and a saddle.

and let w be the other maximum connected to u. We require $w \neq v$ and $f(w) > f(v)$ or otherwise 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 explained in Section 5. Second, we do not eliminate 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.

Node removal We construct 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

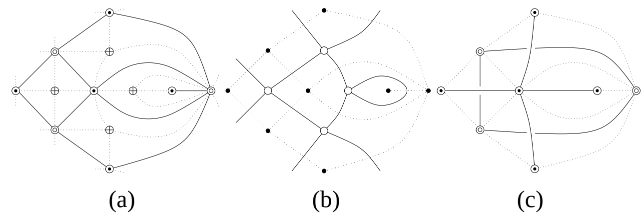


Figure 6: (a) The (dotted) portion of a Morse-Smale complex and the (solid) portion of the corresponding decomposition into diamonds. (b) Portion of the data structure (solid) representing the piece of the decomposition into diamonds (dotted). (c) Cancellation graph (solid) of the decomposition into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

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 operation 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.

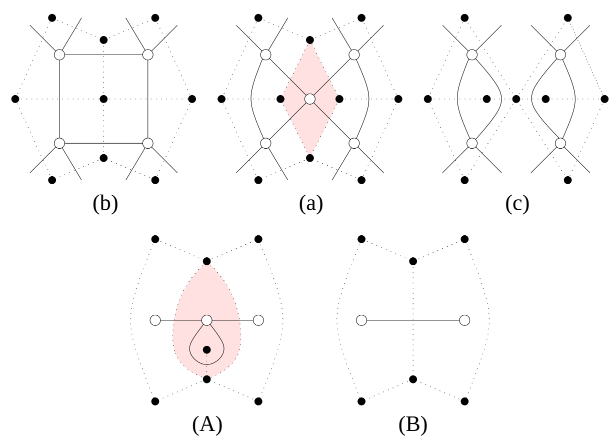


Figure 7: A four-sided diamond (a) can be zipped up in two ways: from top to bottom (b) or from left to right (c). A folded diamond (A) can be zipped up in only one way (B).

To construct the hierarchy by repeated cancellation, we use the algorithm in [Edelsbrunner et al. 2002] to pair critical points as tuples $(s_1, v_1), (s_2, v_2), \ldots, (s_k, v_k)$, 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 obtain Q_{j+1} by modifying Q_j and storing sufficient information so we can recover Q_j from Q_{j+1}. The hierarchy is complete when we reach Q_k. We 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}. Activating and de-activating a diamond requires us to update 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 possibility to interchange cancellations. We will see 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 graph* whose vertices are the minima and maxima. Figure 6(c) shows an example of such a graph. For each diamond there exists an edge connecting the two minima and another edge connecting the two maxima. There are no loops and therefore sometimes only one edge per diamond. Zipping up a diamond corresponds to constricting one of the edges and deleting the other, if it exists. One endpoint of the edge remains as a vertex and the other disappears, implying that the diamonds that share the second endpoint receive a new vertex. A special case arises when a diamond shares both endpoints: the connecting edge that would turn into a loop is deleted.

Two cancellations in a (possibly simplified) MS complex are *interchangeable* when there is no difference between the data structure generated by the two orderings of the operations. For example, the two cancellations zipping up the same diamond are not interchangeable since one preempts the other. In general, two cancellations are interchangeable when their diamonds share no vertex, a condition we refer to as being *independent*. Note that two interchangeable cancellations are not necessarily independent. Even though independence is the more limiting of the two concepts, it offers sufficient flexibility in choosing layers to support the adaptation of the representation to external constraints, such as the biased view of the data.

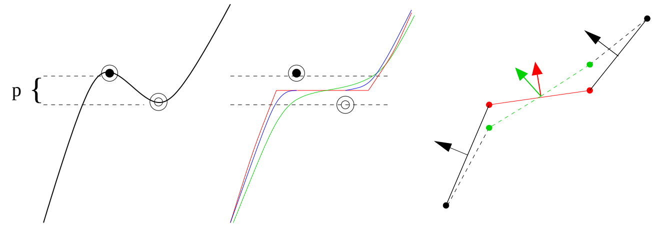
The more independent cancellations we can find the more freedom we have in generating layers in the multi-resolution data structure. Ideally, we would like to identify a large independent set and iterate to construct a shallow hierarchy. However, in the worst case, every pair of cancellations is dependent, which makes the construction of a shallow hierarchy impossible. As illustrated in Figure 9(a), such a configuration exists even for the sphere and for any arbitrary number of vertices. Nonetheless, worst-case situations are unlikely to arise as they require that a large number of folded diamonds. Specifically, it is possible to prove that every MS complex without folded diamonds implies a large independent set of cancellations.

5 Geometric approximation

After each cancellation, we create or change the geometry that locally defines f. We pursue three objectives: the approximation must agree with the given topology, the error should be small, and the approximation should be smooth.

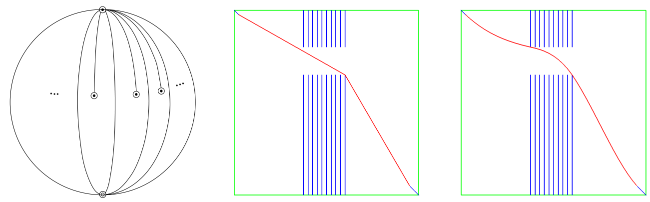
Error bounds We measure the error as the difference between function values at a point. It is convenient to think of the graph of f as the geometry and this difference as the (vertical) distance between the original and the simplified geometry at the location of the point. The persistence of the critical points involved in a cancellation implies a lower bound on the local error. We illustrate this connection for the one-dimensional case in Figure 8(a). Recall that the persistence p of the maximum-minimum pair is the difference in their function values. Any monotonic approximation of the curve between the two critical points has an error of at least $p/2$. We can achieve an error of $p/2$, but only if we accept a flat segment for this portion of the curve, see the blue curve in Figure 8(a). When it is allowed to exceed $p/2$, smoother approximations without flat segments are possible, such as the green curve in the same figure. We note that one can always construct a monotonic approximation that minimizes the error everywhere. However, this approximation is only C^0-continuous for 1-manifolds, as shown by the red curve in Figure 8(a), and it may fail to be continuous for 2-manifolds.

Data fitting We know that monotonic patches exist, provided we are tolerant to errors. Our goal is therefore to find monotonic



(a) (b)

Figure 8: Geometry fitting for 1-manifolds: (a) One-dimensional cancellation and several monotonic approximations. (b) Local averaging used to construct smoothly varying monotonic: Slopes of neighboring edges are combined with the original slope, and the function values are adjusted accordingly. (Edge normals are shown.)



(a) (b)

Figure 9: (a) MS complex on the sphere with pairwise dependent cancellations. (b) One-dimensional smoothing with (blue) error constraints and prescribed endpoint derivatives. (Left) Initial configuration; (right) Constructed solution.

patches that minimize some error measure. A large body of literature deals with the more general topic of shape-constrained approximation [Carlson and Fritsch 1985; Greiner 1991], but due to critical differences (in particular in the notion of monotonicity) we were not able to adapt standard techniques for our purposes. Instead, we use a multi-stage iterative approach to construct the geometry that specifies the simplified representation of f. It provides a smooth C^1-continuous approximation within a specified error bound along the boundaries of the quadrangular patches and a similar approximation but without observing an error bound in the interior of the patches. The paths are constructed iteratively by smoothing the gradients along the edges and post-fitting the function values, as illustrated in Figure 8(b). During each iteration, we first compute the new gradient of an edge as a convex combination of its gradient and the gradients of the adjacent edges. We then adjust the function values at the vertices to realize the new gradients. During an iteration, we maintain the error bound at the vertices and make sure that the completed path is monotonic. In addition, the gradient at the critical points is set to zero. The technique performs well in practice although it converges slowly. Sample results are shown in Figure 9(b). The interior of the quadrangular patches are modified by applying standard Laplacian smoothing to the function values [Taubin 1995]. During each iteration, the value at a vertex is averaged with those of its neighbors. Since the boundaries are monotonic, this procedure converges to a monotonic solution for the patch interior. We now summarize the steps of the geometry fitting process:

1. find all paths affected by a cancellation;
2. use the gradient smoothing to geometrically remove the canceled critical points;
3. smooth the old regions until they are monotonic;
4. erase the paths and re-compute new paths using the new geometry;

5. use one-dimensional gradient smoothing again to force the new paths to comply with the constraints; and
6. smooth the new regions until all points are regular.

The reason for repeating gradient smoothing in Step 5 is that the paths constructed in Step 4 are not guaranteed to satisfy the required error bounds.

6 Remeshing

The geometric approximation of the graph of f requires us to determine a triangular mesh within each quadrangular patch. We discuss the construction of such meshes that are conforming (free of cracks) across shared boundaries.

Parametrization The remeshing of a quadrangular patch requires us to first construct a parametrization over a square. For this purpose, we use mean value coordinates as proposed in [Floater 2003]. We map the four boundary curves by arc length to the four edges of the unit square. To map the interior of the patch (which at this point is represented as a portion of the triangulation K), we use the fact that each vertex can be expressed as a convex combination of its neighbors. The coefficients in this combination may be computed by solving a sparse linear system. Given the parametrization on the boundary, we use these coefficients to map the vertices to the square, thus completing the parametrization.

Next, we sample the unit square on a uniform grid and use its preimage on $\mathbb{M}$ as a new mesh for the patch. The grid resolution is chosen based on a given error bound evaluated along boundary paths which, by construction, follow the direction of maximum change in function value. Specifically, we refine each path until it satisfies the error bound and choose the grid resolution to match the maximum resolution along the four boundary paths.

Crack-free meshes We avoid dependencies between meshes of different patches by allowing T-junctions along shared boundaries. In other words, our representation is not a global triangulation of $\mathbb{M}$ but rather a collection of patches, each triangulated using a regular mesh. We call the collection *crack-free* when the meshes agree geometrically along shared boundaries. One source of potential cracks is the different resolution of neighboring patches. We address this problem by first sampling all paths individually, and when additional samples are required, ensuring that these lie on edges of these paths. In other words, we create T-junctions without physical cracks.

There is a second source of potential cracks that is more difficult to cope with, namely the junctions created by the MS complex algorithm, which are the points at which 1-manifolds merge. As discussed in Section 3, each quadrangle has a connected interior, and we refer to the closure of this open region as a *patch*. Clearly, each junction is the corner of at least one patch. When displaying the data, it is important to render every junction independently of the chosen resolution for the patches. To achieve this goal, we require that each junction is a vertex of the mesh of each patch that touches it. Accordingly, we modify the parametrizations along the paths: We define a *segment* as the connected portion between two junctions or critical points and sample each segment independently.

Data layout and rendering Rather than storing a mesh for each quadrangle explicitly, we use regular grids. This approach allows us to use methods like the one described in [Linstrom and Pascucci 2002] for rendering purposes. By storing each grid in what Lindstrom and Pascucci called interleaved embedded quadtrees we avoid having to store connectivity information, while maintaining high flexibility during rendering. 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. 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 a consistent rendering across boundaries. Since we enforce that samples on grid boundaries are shared their respective error terms agree. Independently of the error term, a patch must always render all its junctions which we guarantee by setting their errors to infinity.

7 Results

We have tested our algorithm on the Puget Sound terrain data set at resolution 1025-by-1025 and elevation values represented by two-byte unsigned integers. We also have used simulation data of the autoignition of a spatially non-homogeneous hydrogen-air mixture, courtesy of Echekki and Chen [2003], at resolution 512-by-512 with temperature values represented by single-byte unsigned integer values. All tests were performed on a 1.8GHz Pentium 4 Linux PC with 1Gb of main memory.

A straightforward application of our algorithm is the removal of topological noise without smoothing the data. This functionality does not dependent on the hierarchy and is implemented by repeated cancellation of critical point pairs with lowest persistence. Our experience suggests that this noise removal step should always be applied, even when one only removes the topological artifacts caused by symbolic perturbation, which can be achieved by setting the persistence threshold during noise removal to zero. The effect of this procedure is illustrated in Figure 10, where it reduces the number of critical points from 2,859 (left) to 446 (right). This reduction is achieved by removing all topological features with persistence less than 0.1% of the temperature range, which was done in about one second.

Figure 11 shows the Puget Sound data set. The original topology has 49,185 critical points and is too dense to be printed. The upper-left picture shows the data with 4,045 critical points obtained after removing the topological noise using a persistence threshold of 0.5% of the elevation range. The upper-right picture shows the approximation of the data with 2,025 critical points obtained by increasing the persistence threshold to 1.2%. The lower-left picture shows a significantly coarser MS complex containing 289 critical points that remain after increasing the persistence threshold to 20%. Finally, the lower-right picture shows a view-dependent image of the data constructed for the purple view frustum. The resolution is preserved inside the frustum yielding a total of 1,070 critical points. Outside the frustum, we have simplified the data to the extent possible. One observes a quick drop in resolution away from the frustum. The pre-processing of the Puget Sound data was done in about three and a half hours, due to the slow convergence of the geometric fitting procedure. The traversal of the hierarchy and rendering are fully interactive.

Table 1 lists numerical measurements that quantify our two experiments. It compares the respective original data set with simplified representations obtained by varying two independent parameters: the persistence threshold that defines the layer in the multi-resolution data structure and the rendering error threshold that controls the resolution of the meshes representing the geometry. The persistence threshold determines the version of the MS complex and therefore also the number of critical points, which we give in square brackets because it is perhaps a more intuitive parameter than the persistence threshold itself. For each chosen pair of parameters, we use the Metro tool [Cignoni et al. 1998] to compute the root-mean-square (RMS) of the vertical error at the vertices. As expected, the RMS error grows and the total size of the meshes shrinks with increasing persistence and rendering error thresholds.

RMS error / # triangles					
Comb. Process		persistence [# critical points]			
		0.1% [446]	1% [343]	5% [263]	50% [156]
rend. error	0.010	0.0017 / 71126	0.0021 / 72346	0.0025 / 69592	0.0206 / 56881
	0.020	0.0023 / 38962	0.0026 / 39869	0.0030 / 38578	0.0207 / 31802
	0.050	0.0050 / 17023	0.0050 / 17245	0.0053 / 16619	0.0212 / 13711
Puget Sound		persistence [# critical points]			
		0.5% [4045]	1% [2408]	5% [1408]	20% [364]
rend. error	0.005	0.0010 / 81809	0.0013 / 79298	0.0016 / 77503	0.0029 / 73099
	0.010	0.0012 / 53494	0.0015 / 49680	0.0018 / 46680	0.0030 / 43574
	0.030	0.0015 / 41288	0.0017 / 36558	0.0020 / 31810	0.0033 / 23862

Table 1: Approximation errors and number of faces used in the representation of the data for different persistence values and different rendering accuracies.

8 Conclusions

We have described a new topology-based multi-resolution 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 its well-defined simplification procedures, the approach is appealing for applications based on topological analysis, for example, data segmentation and feature detection and tracking in medical imaging or simulated flow field data sets. 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. H. Edelsbrunner is partially supported by the National Science Foundation (NFS) under grants EIA-99-72879 and CCR-00-86013. B. Hamann is supported by the NSF under contract ACI 9624034, through the LSSDSV program under contract ACI 9982251, and through the NPACI; the National Institute of Mental Health and the NSF 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.

References

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

BAJAJ, C. L., AND SCHIKORE, D. R. 1998. Topology preserving data simplifi cation 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 simplifi ed 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 simplifi cation. *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 simplifi cation 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 simplifi cation 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. IEEE 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 simplifi cation. *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.

HORMANN, K., 1971. Morphometrie der Erdoberfläche, Schrift. d. Univ. Kiel.

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 effi cient polygonal simplifi cation. In *Proc. IEEE Visualization '98*, IEEE Computer Society Press, Los Alamitos California, IEEE, 279–286.

LINSTROM, P., AND PASCUCCI, V. 2002. Terrain simplifi cation simplifi ed: A general framework for view-dependent out-of-core visualization. *IEEE Transactions on Visualization and Computer Graphics 8*, 3 (July-September), 239–254.

MARK, D. M. 1977. Topological properties of geographic surfaces. In *Proc. Advanced Study Symposium on Topological Data Structures*.

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.

MILNOR, J. 1963. *Morse Theory*. Princeton University Press, New Jersey.

MORSE, M. 1925. Relations between the critical points of a real functions of n independent variables. *Transactions of the American Mathematical Society 27* (July), 345–396.

MUNKRES, J. R. 1984. *Elements of Algebraic Topology*. Addison-Wesley, Redwood City, CA.

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.

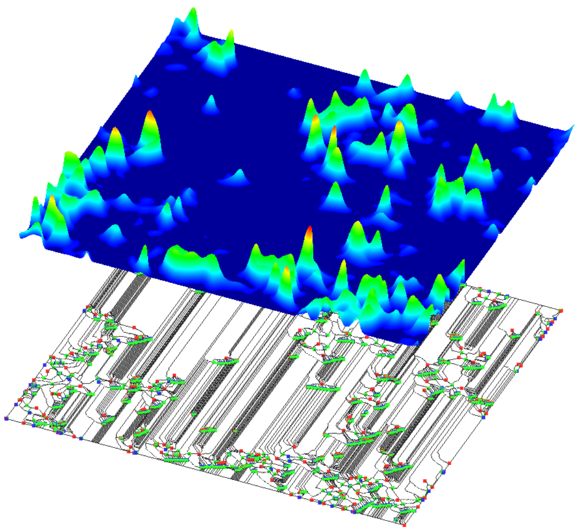


Figure 10: (Left) Original MS complex of the combustion process data with 2,859 critical points. A textured rendering is shown together with the corresponding MS complex in which maxima are red, minima are blue, saddles are green, and paths are black. (Right) Same data with 446 critical points after removing all topological features with persistence less than 0.1% the temperature range.

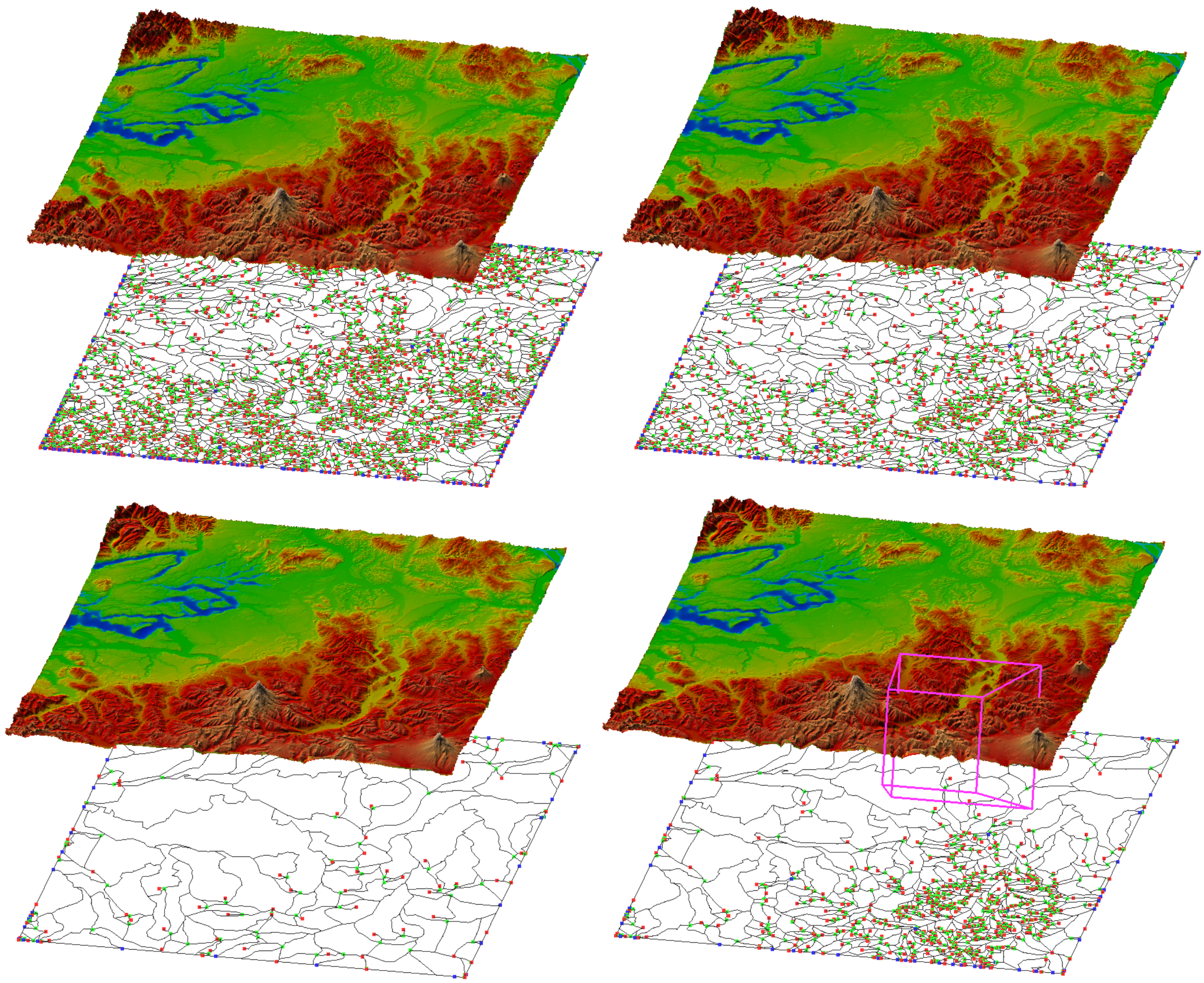


Figure 11: (Upper-left) Puget Sound data after topological noise removal. (Upper-right) Data at persistence of 1.2% of the maximum height. (Lower-left) Data at persistence 20% of the maximum height. (Lower-right) View-dependent refinement (purple: view frustum).

University of California
Lawrence Livermore National Laboratory
Technical Information Department
Livermore, CA 94551