

Maximizing Adaptivity in Hierarchical Topological Models

Peer-Timo Bremer
Dept. of Computer Science,
University of Illinois,
Urbana-Champaign
ptbremer@acm.org

Valerio Pascucci
Center for Applied Scientific
Computing
Lawrence Livermore National
Laboratory
pascucci@llnl.gov

Bernd Hamann
Institute for Data Analysis and
Visualization
Dept. of Computer Science
University of California, Davis
hamann@cs.ucdavis.edu

ABSTRACT

We present an approach to hierarchically encode the topology of functions over triangulated surfaces. The topology of a function is described by its Morse-Smale complex, a well known structure in computational topology. Following concepts of Morse theory, a Morse-Smale complex (and therefore a function's topology) can be simplified by successively canceling pairs of critical points. We demonstrate how cancellations can be effectively encoded to produce a highly adaptive topology-based multi-resolution representation of a given function. Contrary to the approach of [4] we avoid encoding the complete complex in a traditional mesh hierarchy. Instead, the information is split into a new structure we call a *cancellation forest* and a traditional dependency graph. The combination of this new structure with a traditional mesh hierarchy proves to be significantly more flexible than the one previously reported [4]. In particular, we can create hierarchies that are guaranteed to be of logarithmic height.

1. INTRODUCTION

Topology-based methods used for visualization and analysis of scientific data are becoming increasingly popular. Their main advantage lies in the capability to provide a concise description of the overall structure of a scientific data set. Subtle features can easily be missed when using “traditional” visualization methods like volume rendering or contouring, unless “correct” transfer functions and isovalues are chosen. On the other hand, the presence of a large number of small features creates a “noisy visualization,” in which larger features can be overlooked. By visualizing topology directly, one can guarantee that no feature is missed. Furthermore, one can use sound mathematical principles to simplify a topological structure. The topology of functions is also often used for feature detection and segmentation (e.g., in surface segmentation based on curvature).

However, for topology-based data analysis one needs flexible, hierarchical models able to adaptively remove noise or features not relevant for a particular segmentation. In practice, the simplification/refinement should be fast (possibly interactive) and highly adaptive in order to be useful for a large variety of situations. Requiring interactivity inadvertently leads to the use of hierarchical encodings rather than simplification schemes. Hierarchical models

often reduce the adaptivity of a representation to gain the ability to perform incremental changes for varying queries.

We address the need for adaptive topology-based data exploration by improving significantly the topological hierarchy described in [4]. By storing the hierarchy in two different but connected data structures, we show how one can remove many of the dependencies in the original hierarchy, making the structure simpler, more compact, and more adaptive than the original one.

1.1 Related Work

The topological structure of a scalar field can be described partially by its contour tree [21, 5, 22], which describes the relations between the connected components of its level sets. This structure provides a user with a compact representation of the topology [1] and can be used to accelerate the computation of isosurfaces [31]. However, the contour tree provides only limited information about the embedding of the level sets and therefore remains somewhat abstract. Morse theory [20, 19], on the other hand, provides methods to analyze the complete topology of a function over a manifold as well as its embedding. Early approaches for the bivariate case are discussed in [6, 18, 23]. More recently, the Morse-Smale complex was introduced by Edelsbrunner et al. [10, 9] as a description of the topology of scalar-valued functions over two- and three-dimensional manifolds. Applications of this theory vary from implicit geometry modeling [27] to shape description [15]. Related concepts are also used in flow visualization. Helman and Hesselink [14] showed how to find and classify critical points in flow fields and proposed a structure similar to the Morse-Smale complex for vector fields. Later, methods to analyze and simplify this complex were proposed by de Leeuw and van Liere [8] and Tricoche et al. [29, 30].

A large amount of research has been devoted to the construction of multi-resolution representations for meshes. Even a partial overview of the related work is beyond the scope of this paper. Conceptually, the hierarchy for the reduced Morse-Smale complex is related to concepts from [25, 13, 26, 16, 17], as it is created by successive vertex removal. For more references and a discussion on a general theory of multi-resolution modeling we refer the reader to [12]. Nevertheless, the complete Morse-Smale complex incorporates many topological constraints, and its structure (and multi-resolution encoding) is quite different from those of traditional meshes. The first multi-resolution encoding of a Morse-Smale complex we are aware of was proposed by Pfaltz [24], which has been improved and extended by Edelsbrunner et al. [10] and Bremer et al. [3, 4]. The more recent hierarchical structures are based on the concept of *persistence* [11], which relates the difference in function value of critical point pairs to the importance of a topological feature.

We first review necessary concepts from Morse theory and the construction of a Morse-Smale complex (Section 2). In Section 3, we describe the cancellation forest, and we discuss the resulting

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, to publish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.

Copyright 200X ACM X-XXXXX-XX-X/XX/XX ...\$5.00.

hierarchy in Section 4. We conclude with results and possibilities for future research (Section 5).

2. MORSE-SMALE COMPLEX

We base our algorithms on intuitions derived from the study of smooth functions. We review key aspects from Morse theory [20, 19] for smooth functions and discuss how these can be used in the piecewise linear case.

2.1 Morse Theory

Throughout this paper \mathbb{M} denotes a smooth 2-manifold. Given a smooth function $f : \mathbb{M} \rightarrow \mathbb{R}$, a point $a \in \mathbb{M}$ is called *critical* when its *gradient* $\nabla f(a) = (\delta f / \delta x, \delta f / \delta y)$ vanishes; it is called *regular* otherwise. For two-manifolds, (non-degenerate) critical points are maxima (f decreases in all directions), minima (f increases in all directions), or saddles (f switches between decreasing and increasing four times around the point). Using a local coordinate frame at a , we compute the *Hessian* H of f , which is the matrix of second partial derivatives. If H is non-singular we can construct a local coordinate system such that f has the form $f(x, y) = f(a) \pm x^2 \pm y^2$ in a neighborhood of a . 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.

At any regular point, the gradient (vector) is non-zero, and when we follow the gradient we trace out an *integral line*, which starts at a critical point and ends at a critical point, while technically not containing either of them. Since f is smooth, two integral lines are either disjoint or the same. The *descending manifold* $D(a)$ 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 . The collection of descending 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* $A(a)$ of a as the union of a and all integral lines that start at a . When neglecting certain degenerate cases, see [10], we can overlay these two complexes and obtain what we call the *Morse-Smale complex*, or *MS complex*, of f . Its *vertices* are the minima, maxima, and saddles of f . Its *edges* are integral lines between saddles and extrema that bound four-sided *faces*. An example is shown in Figure 1.

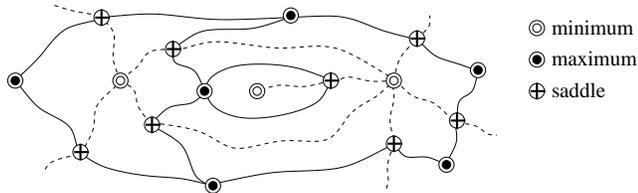


Figure 1: Morse-Smale complex: descending one-manifolds shown as dotted and ascending one-manifolds as solid lines.

Using the insight gained from smooth Morse theory when applied to piecewise linear functions, we follow the concepts described in [3]. We identify and classify critical points based on their local neighborhood, see [2, 10]. If all neighboring vertices of a point v have function values below that of v , we call it a maximum; if all are above v , then we call it a minimum etc., see Figure 2. In general, there can exist saddles with multiplicity greater than one that we split into simple ones, as shown on the far right in Figure 2.

2.2 Cancellations

To simplify an MS complex we remove connected pairs of critical points. The two possible cases are a maximum-saddle or a

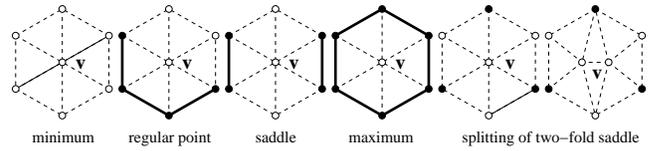


Figure 2: Classification of a vertex v based on relative height of its neighbors, where light vertices/edges mark higher-value neighbors and solid vertices/edges lower-value neighbors.

minimum-saddle pair. Since all saddles have valence four (have four incident edges) each maximum(minimum)-saddle pair uniquely defines another maximum(minimum) connected to the same saddle. We refer to such a triple as a *pass*.

Following [10] and [4] we use *cancellations* of passes to simplify an MS complex, see Figure 3.

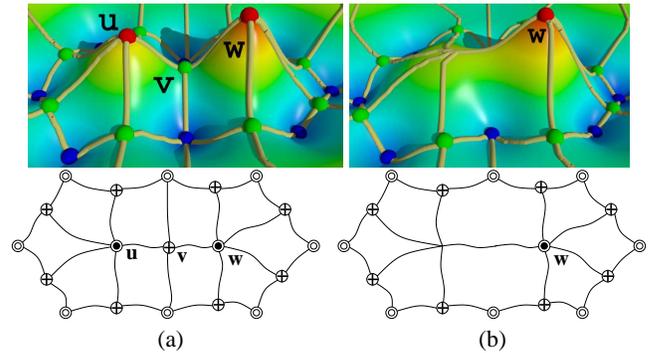


Figure 3: MS complex before (a) and after (b) the cancellation of the pair (u, v) .

Definition 1. Let (u, v, w) be a pass of the MS complex M such that $|f(u) - f(v)| < |f(w) - f(v)|$. The **cancellation** $C_{u,v}$ concatenates the edges $u - v - w$ to edges ending at u and removes the remaining edges incident to v (merging the adjacent faces).

The importance of a pass is measured by the difference in function value (or height) between its first two vertices

Definition 2. Let (u, v, w) be a pass, then the difference in function value $p = |f(u) - f(v)|$ is called the **variation** of (u, v, w) .

2.3 Construction

The MS complex is constructed by successively computing its edges, starting from the saddles, see [3]. We compute two lines of steepest ascent and two lines of steepest descent connecting each saddle to two maxima and two minima. Two ascending (descending) edges can merge; while ascending edges must remain separate from descending edges. Once two edges have been merged they never split. Following these rules, we are guaranteed to produce a non-degenerate MS complex. A more detailed analysis can be found in [4]. Having computed all edges, we partition the surface into regions by growing each region from saddles without ever crossing an edge.

3. SIMPLIFICATION

To create an efficient and flexible hierarchical encoding of an MS complex one must be able to store and manipulated sequences of cancellations. In this section we introduce a structure we call a *cancellation forest* that encodes a sequence of cancellations and provides a simple way to implement a bottom-up simplification of an MS complex.

3.1 Cancellation Forests

First we define a cancellation forest and discuss some of its properties

Definition 3. Given an MS complex Q and a sequence T of cancellations of passes $(u_0, v_0, w_0), \dots, (u_n, v_n, w_n)$ the **cancellation forest** corresponding to T is defined as a set of trees with the u_i 's and w_i 's as nodes and an arc for each v_i between its two incident extrema of the same index as u_i (in Q). Each **cancellation tree** contains exactly one node which is not removed by T which is called its **root**, see Figure 4.

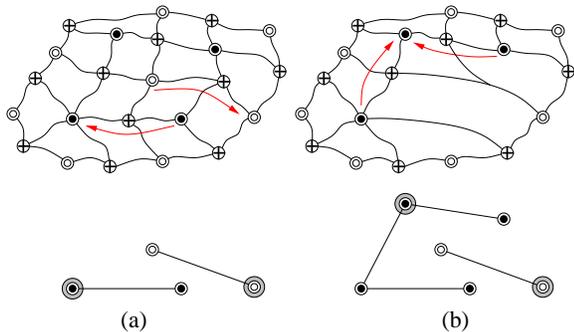


Figure 4: Constructing a cancellation forest: An MS complex with cancellations (indicated by arrows) is shown on the top, the corresponding cancellation forest on the bottom (roots marked in grey).

Cancellation trees have several useful properties:

1. A cancellation tree consists of either only minima or only maxima;
2. Since $u_i \neq w_i$ a tree cannot have a cycle;
3. Each arc in a tree uniquely identifies a cancellation or alternatively a saddle and two incident edges of the MS complex.

3.2 Bottom-up Simplification

Given the cancellation forest CF of a sequence of cancellations T it is straight forward to construct the corresponding simplified MS complex M' . The algorithm is illustrated in Figure 5. The connection between arcs of the forest and edges of the MS complex provides a natural way to draw a forest embedded into its MS complex. Once all cancellations of T are performed all edges in M' have either not been changed or have been concatenated with edges corresponding to arcs of the forest. In the original MS complex M the second type of edges were connected to extrema that are leaf's of CF . It is easy to see that edges of M' result from concatenating the original edge with those edges encountered on the way from the leafs of CF to its roots. Therefore, a bottom-up simplification algorithm for an MS complex M can be implemented as follows:

1. Construct a cancellation forest by choosing a sequence of appropriate cancellations;
2. For each leaf in the forest compute a sequence of edges (of M) that connects the leaf to its root; and
3. For each edge e of M either
 - a) do nothing, if its extremum has not been canceled; or
 - b) concatenate e with the sequence of edges that connects its original extremum with the root of its tree.

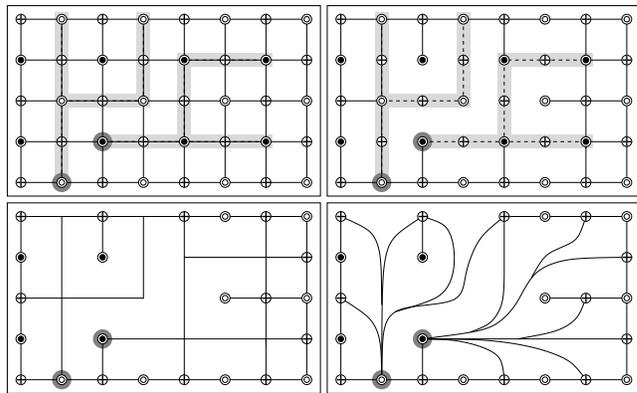


Figure 5: Bottom-up simplification using a cancellation forest. (Upper-left) Original MS complex with the edges corresponding to arcs highlighted. (Upper-right) MS complex after removing all edges that will be removed during cancellations. (Lower-left) Resulting simplified complex created by concatenating edges. (Bottom-right) The resulting MS complex draw with “smoothed” edges to highlight the structure.

4. HIERARCHY

Bottom-up simplification as described above by definition always starts from the original MS complex and reconstructs each simplified complex from scratch. This is impractical for large complexes and likely to be slow even for medium sized ones. Instead, one should aim for an iterative construction in which each simplification is adapted from the one before. In this section we describe the data structure we use to store an MS complex hierarchically, how this structure is create, and how it is used at run-time.

4.1 Data Structure

The data structure to hierarchically encode an MS complex consists of two interconnected parts: One particular cancellation forest and a dependency graph of arcs of the forest.

Rather than trying got encode all possible cancellation forests of an MS complex M we focus on only one forest which is created based on different metrics depending on the application, see below. This *master forest* is stored as a doubly connected tree meaning each node contains a list of parents and a list of children. However, the master forest can be adapted by activating and deactivating arcs which creates a set of *sub-forests*. As discussed in Section 3, each sub-forest represents a particular simplification of M .

Changing one simplification, one sub-forest, into another requires to deactivate some arcs of the master forest and to activate others. This corresponds to performing some anti-cancellations (deactivations) and cancellations (activations) in M . The second part of the data structure is to store arcs (representing cancellations) of the master forest in a *dependency graph*. The dependency graph encodes a partial order among arcs in which each arc a has a set of predecessors that must be active before a can be active and a set of successors that can only be active if a is active. As usual, we store the partial order between arcs as a directed graph.

4.2 Construction

First, we describe how the data structure is build given a specific sequence of cancellations T . Second, we discuss different objectives in how to choose T and explain the algorithm used for all examples in this paper.

The construction mostly follows standard concepts, see, for ex-

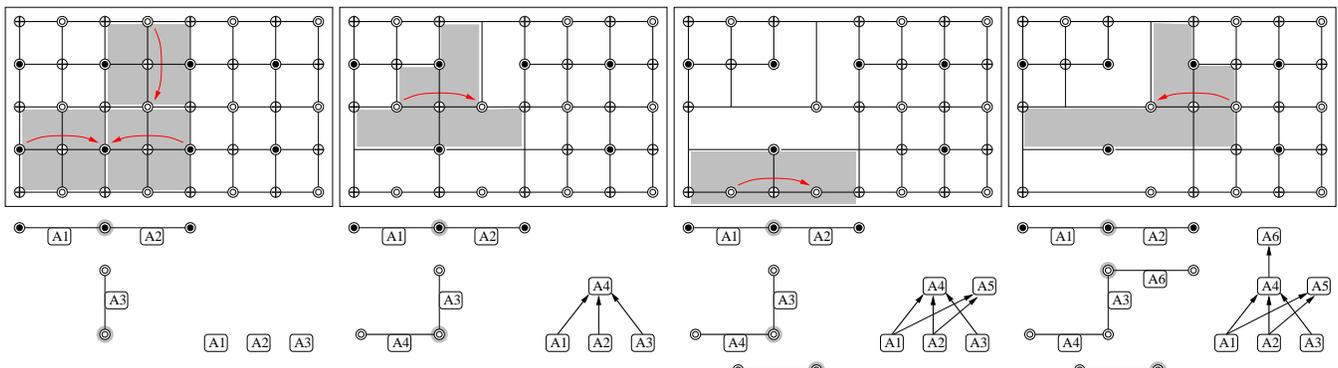


Figure 6: Hierarchy construction: The successively simplified MS complex is shown on top with cancellations indicated by arrows and regions of interferences shaded. On the bottom the resulting cancellation forest is shown on the left and the dependency graph of arcs on the right.

ample, [12]. The *region of interference*, RoI , of cancellation $C_{u,v}$ is defined as all cells incident to v . Then for every cancellation C_{u_i,v_i} of T we:

1. Create a new arc a_i in the master forest;
2. For all C_{u_j,v_j} whose RoI intersects the cells incident to v_i make a_j a predecessor of a_i and a_i a successor of a_j ; and
3. Cancel (u_i, v_i) as discussed in Section 2.2

Finally, we use path compression [28] to reduce the number of arcs in the dependency graph. In practice, most of the path compression is done implicitly during step 2. by removing trivial shortcuts in the dependency graph. An example is shown in Figure 6.

To choose a particular T one must analyze the effects of different sequences. The set of cancellations (independent of the order) defines the master forest. Only extrema connected by an arc can be canceled. The order among the cancellations determines the dependency graph. Because, arcs can only be activated if all their predecessors are activated, the order among cancellations is locally (within the $RoIs$) preserved for all possible cancellations. There exist two main objectives: Passes with small variation should be canceled first. This ensures that the master forest allows the most unimportant passes to be canceled. Second, the order should create a “shallow” dependency graph meaning short distances from any leaf (node with no predecessor) to its root(s). It is well known that a shallow dependency graph translates into a large number of different configurations. This translates into a large number of possible simplifications encoded in the hierarchy.

The first objective can be optimized by always choosing the cancellation with minimal global variation. However, it is possible, that the $RoIs$ of each successive pair of cancellation overlap in which case the dependency graph degenerates into a line with only a linear number of possible configurations. The second objective is optimized by choosing maximal sets of independent cancellations (non-intersection $RoIs$). Nevertheless, cancellations with small variation can become dependent on ones with large variation which is undesirable in most applications. We choose a compromise, by creating large “badges” of independent cancellation constructed by successively adding cancellations with minimal variation among the set of still independent cancellations. Furthermore, we restrict the maximal variation in each badge to be no more than twice that of the previous badge.

4.3 Analysis

Before a cancellation the region of interference consists of at most four cells, which merge into at most two cells. Therefore, each

arc can have at most four predecessors and two successors. If we do not restrict the range of variation in a badge of cancellations it is clear that each badge contains about $1/4$ of the remaining critical points in the MS complex and the hierarchy is of logarithmic height. When restricting badges as suggested above this can no longer be guaranteed. However, we found no difference in adaptivity when restricting the badges. However, for maximal independent sets the overall quality of the approximation decreased notably due cancellations with large variation being predecessors to ones with small variation.

4.4 Run-Time Traversal

As common geometric hierarchies our topological hierarchy can be used with a large variety of refinement criteria. For example, one might refine based on global variation, view-point, and/or function value. The refinement criterion determines a set of arcs that must be active, which if necessary activate all their predecessors. The current simplification is then modify by deactivating (anti-canceling) arcs that no longer need to be active and activating (canceling) those that should be active. Cancellations are implemented as defined in Section 2.2. However, for each cancellation we store which saddle has been removed as well as the (unique) indices of the cells involved before and after the cancellation. During an anti-cancellation, we replace two cells by four and introduce the appropriate saddle all of which is based on the information stored explicitly. However, we also must choose which extremum to introduce into the MS complex. This information is stored implicitly in the master forest. An anti-cancellation means the deactivation of an arc which splits one cancellation tree into two. By definition, each tree is represented in the MS complex by its root, which is the highest(lowest) maximum(minimum). At each anti-cancellation we therefore search for the roots in the trees on both sides of the arc and use their respective roots as vertices. Potentially, this search is linear in the number of nodes of the tree. Since in practice trees seem to have a very low branching factor, see Figure 7, and can be split at any arc more elaborate data structures are unlikely to improve the search time. In any case, the trees are rather small and searching for roots takes only a small fraction of overall computation cost.

It is interesting to note that the search for roots symbolizes the trade off between the higher flexibility of the hierarchy presented here and the constant time take for each cancellation of the method proposed in [4]. In the original hierarchy of [4], the extrema were stored explicitly with each cancellation. In order to maintain a consistent MS complex the RoI of each cancellation must then be in-

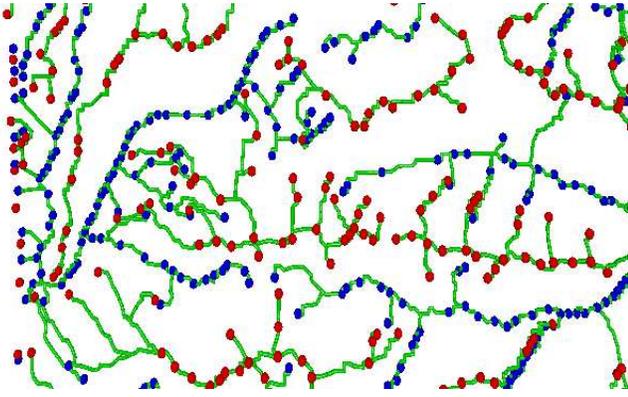


Figure 7: Typical extrema trees. Maxima are shown in red, minima in blue, and arcs in green.

creased to contain all cell incident to either u , v , or w . As a result the dependency graph becomes significantly “higher” and the hierarchy is far less adaptable, see Section 5.

5. RESULTS

To compare the new hierarchy with the one proposed in [4] we have applied both strategies to a 1201×1201 single-byte integer value terrain data set of the Grand Canyon. Figure 10 shows a rendering (a) and the initial MS complex (b) of the Grand Canyon data set with 11620 critical points. We assess quality via a fly-over, comparing the adaptivity of the original hierarchy with the one using the cancellation forest. A view-frustum is defined, where the topology is refined to the highest resolution. Outside the given view-frustum only dependent topology is used. Figure 11 shows two frames of the fly-over for two distinct stages of the fly-over path. An animation showing the complete fly-over can be found at <http://www.pascucci.org>

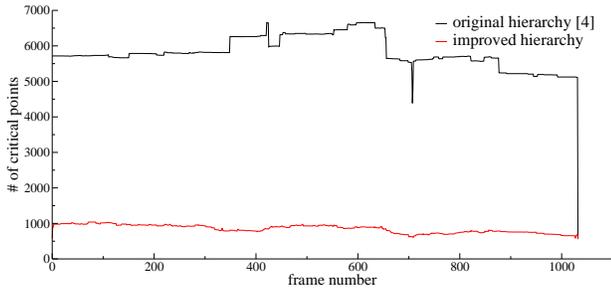


Figure 8: Number of critical points used during a fly-over (Grand Canyon data set).

Figure 8 shows the number of critical points in the adaptive MS complex during the fly-over for both methods used for hierarchy construction. The hierarchy using the cancellation forest is clearly superior to the original encoding. One explanation for the large difference in quality is the presence of high-valency extrema in the MS complex. Often, data sets (especially terrains) are biased to contain significantly more maxima than minima (or the reverse), which consequently results in some extrema of the MS complex with high-valency values. Using the original large region of interference, the hierarchy around a high-valency extremum degenerates into a linear sequence.

The adaptive refinement and display of topology is useful in many areas. Figure 9 shows the oil pressure of an underground oil

reservoir. (Oil is extracted by pressing water into the reservoir at some sites and pumping oil at others. As more water is forced into the reservoir it becomes increasingly saturated with water, and at some point oil production ceases to be effective.) The figure shows an isosurface of water saturation, pseudo-colored by oil pressure. The linear color map used in Figure 9(c) provides little structural information. However, the seven oil extraction sites are clearly visible as local minima in the simplified MS complex.

Figure 12(a) shows a rendering of the Yakima data set using 1201×1201 single-byte integer values, (b), and (c) show the corresponding MS complex with 17691 critical points and the same complex refined to preserve only features below a function value of 0.14 (with function values scaled to $[0, 1]$) using 8063 critical points. The density of the MS complex shows how the region around the canyons remains highly refined.

Figure 13 shows the Mixing Fluid data set. The surface is an isosurface representing the boundary between two mixing fluids extracted from one time-step of a turbulent mixing simulation. The data has been generated by the Miranda code a higher order hydrodynamics code for computing fluid instabilities and turbulent mixing at the Lawrence Livermore National Laboratory [7]. In particular, scientists are interested in “bubbles” formed during the mixing process and their automatic segmentation. Using the z -coordinate as Morse function on the surface bubbles are described by the descending manifolds of maxima as shown in Figure 13. Nevertheless, the segmentation of Figure 13(a) is not optimal as some bubbles have multiple maxima and there exist many superfluous maxima cause by noise in the data set. Using a uniform simplification of the MS complex one can remove most of these artifacts and create a much cleaner segmentation, as shown in Figure 13(b). Figure 13(c) and (d) show a non-uniform refinement of the same data which concentrates only on data within the grey focus sphere. As in the Grand Canyon fly-over, the hierarchy using the cancellation forest proves to be far more adaptive than the original one of [4].

6. CONCLUSIONS

We have improved our original results discussed in [4] significantly in several different ways, moving towards the practical application of topology for data visualization and analysis. Using the cancellation forest, the hierarchy is smaller, more adaptable, and supports the use of larger, more complicated MS complexes. Furthermore, cancellation trees are easy to implement and to maintain during refinement. Overall, we have presented simple and highly flexible methods for the topological data analysis. We are currently working on extending all algorithms to volumetric and time-dependent data and are collaborating with different scientists applying and/or adapting the concepts to different problems.

Acknowledgments

This work was performed under the auspices of the US Department of Energy by University of California Lawrence Livermore National Laboratory under contract No. W-7405-Eng-48. B. Hamann is supported by National Science Foundation under contract ACI 9624034 (CAREER Award), through the Large Scientific and Software Data Set Visualization (LSS-DSV) program under contract ACI 9982251, through the National Partnership for Advanced Computational Infrastructure (NPACI), and by a large Information Technology Research (ITR) grant. We thank the members of the Data Science thrust from the Center for Applied Scientific Computing at Lawrence Livermore National Laboratory, and the Visualization and Computer Graphics Research Group at the University of California, Davis.

7. REFERENCES

- [1] C. L. Bajaj, V. Pascucci, and D. R. Schikore. Visualization of scalar topology for structural enhancement. In D. Ebert, H. Hagen, and H. Rushmeier, editors, *Proc. IEEE Visualization '98*, pages 51–58, Los Alamitos California, 1998. IEEE, IEEE Computer Society Press.
- [2] T. F. Banchoff. Critical points and curvature for embedded polyhedral surfaces. *American Mathematical Monthly*, 77(5):457–485, May 1970.
- [3] P.-T. Bremer, H. Edelsbrunner, B. Hamann, and V. Pascucci. A multi-resolution data structure for two-dimensional Morse-Smale functions. In G. Turk, J. J. van Wijk, and R. Moorhead, editors, *Proc. IEEE Visualization '03*, pages 139–146, Los Alamitos California, 2003. IEEE, IEEE Computer Society Press.
- [4] P.-T. Bremer, H. Edelsbrunner, B. Hamann, and V. Pascucci. A topological hierarchy for functions on triangulated surfaces. *IEEE Trans. on Visualization and Computer Graphics*, 10(4):385–396, 2004.
- [5] H. Carr, J. Snoeyink, and U. Axen. Computing contour trees in all dimensions. *Comput. Geom. Theory Appl.*, 24(3):75–94, 2003.
- [6] A. Cayley. On contour and slope lines. *The London, Edinburgh and Dublin Philosophical Magazine and Journal of Science*, XVIII:264–268, 1859.
- [7] A.W. Cook, W. Cabot, and P.L. Miller. The mixing transition in rayleigh-taylor instability. *J. Fluid Mech.*, to appear, 2004.
- [8] W. de Leeuw and R. van Liere. Collapsing flow topology using area metrics. In *Proc. IEEE Visualization '99*, pages 349–354. IEEE Computer Society Press, 1999.
- [9] H. Edelsbrunner, J. Harer, V. Natarajan, and V. Pascucci. Morse-Smale complexes for piecewise linear 3-manifolds. In *Proc. 19th Sympos. Comput. Geom.*, pages 361–370, 2003.
- [10] H. Edelsbrunner, J. Harer, and A. Zomorodian. Hierarchical Morse-Smale complexes for piecewise linear 2-manifolds. *Discrete Comput. Geom.*, 30:87–107, 2003.
- [11] H. Edelsbrunner, D. Letscher, and A. Zomorodian. Topological persistence and simplification. *Discrete Comput. Geom.*, 28:511–533, 2002.
- [12] L. De Floriani, E. Puppo, and P. Magillo. A formal approach to multiresolution modeling. In W. Straßer, R. Klein, and R. Rau, editors, *Theory and Practice of Geometric Modeling*. Springer Verlag, 1996.
- [13] B. Hamann. A data reduction scheme for triangulated surfaces. *Computer Aided Geometric Design*, 11(2):197–214, 1994.
- [14] J. L. Helman and L. Hesselink. Visualizing vector field topology in fluid flows. *IEEE Computer Graphics and Applications*, 11(3):36–46, May/June 1991.
- [15] M. Hilaga, Y. Shinagawa, T. Kohmura, and T. L. Kunii. Topology matching for fully automatic similarity estimation of 3d shapes. In E. Fiume, editor, *Proceedings of ACM SIGGRAPH 2001*, pages 203–212, New York, NY, USA, 2001. ACM.
- [16] R. Klein, G. Liebich, and W. Straßer. Mesh reduction with error control. In R. Yagel and G. M. Nielson, editors, *Proc. IEEE Visualization '96*, pages 311–318, Los Alamitos California, Oct. 1996. IEEE, IEEE Computer Society Press.
- [17] L. Kobbelt, S. Campagna, and H.-P. Seidel. A general framework for mesh decimation. In *Proc. of Graphics Interface 2001*, pages 43–50, June 1998.
- [18] J. C. Maxwell. On hills and dales. *The London, Edinburgh and Dublin Philosophical Magazine and Journal of Science*, XL:421–427, 1870.
- [19] J. Milnor. *Morse Theory*. Princeton University Press, New Jersey, 1963.
- [20] M. Morse. Relations between the critical points of a real functions of n independent variables. *Transactions of the American Mathematical Society*, 27:345–396, July 1925.
- [21] S. P. Morse. A mathematical model of the analysis on contour-line data. *Journal of the Association for Computing Machinery*, 15(2):205–220, Apr. 1968.
- [22] V. Pascucci and K. Cole-McLaughlin. Efficient computation of the topology of level sets. In M. Gross, K. I. Joy, and R. J. Moorhead, editors, *Proc. IEEE Visualization '02*, pages 187–194, Los Alamitos California, 2002. IEEE, IEEE Computer Society Press.
- [23] J. Pfaltz. Surface networks. *Geographical Analysis*, 8:77–93, 1976.
- [24] J. Pfaltz. 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, 1979.
- [25] W. J. Schröder, J. Zarge, and W. E. Lorensen. Decimation of triangle meshes. *Computer Graphics (Proc. SIGGRAPH '92)*, 26(2):65–70, July 1992.
- [26] M. Soucy, G. Godin, and M. Rioux. A texture-mapping approach for the compression of colored 3d triangulations. *The Visual Computer*, 12:503–514, 1996.
- [27] B. T. Stander and J. C. Hart. Guaranteeing the topology of implicit surface polygonization for interactive modeling. In *Proc. of ACM SIGGRAPH 1997*, volume 31, pages 279–286, New York, NY, USA, Aug. 1997. ACM, ACM Press / ACM SIGGRAPH.
- [28] R. E. Tarjan. Applications of path compression on balanced trees. *J. ACM*, 26(4):690–715, 1979.
- [29] X. Tricoche, G. Scheuermann, and H. Hagen. A topology simplification method for 2d vector fields. In *Proc. IEEE Visualization '00*, pages 359–366, Los Alamitos California, 2000. IEEE, IEEE Computer Society Press.
- [30] X. Tricoche, G. Scheuermann, and H. Hagen. Continuous topology simplification of planar vector fields. In *Proc. IEEE Visualization '01*, pages 159–166, Piscataway, NJ, Oct. 2001. IEEE, IEEE Computer Society Press.
- [31] M. J. van Kreveld, R. van Oostrum, C. L. Bajaj, V. Pascucci, and D. Schikore. Contour trees and small seed sets for isosurface traversal. In *Symposium on Computational Geometry*, pages 212–220, 1997.

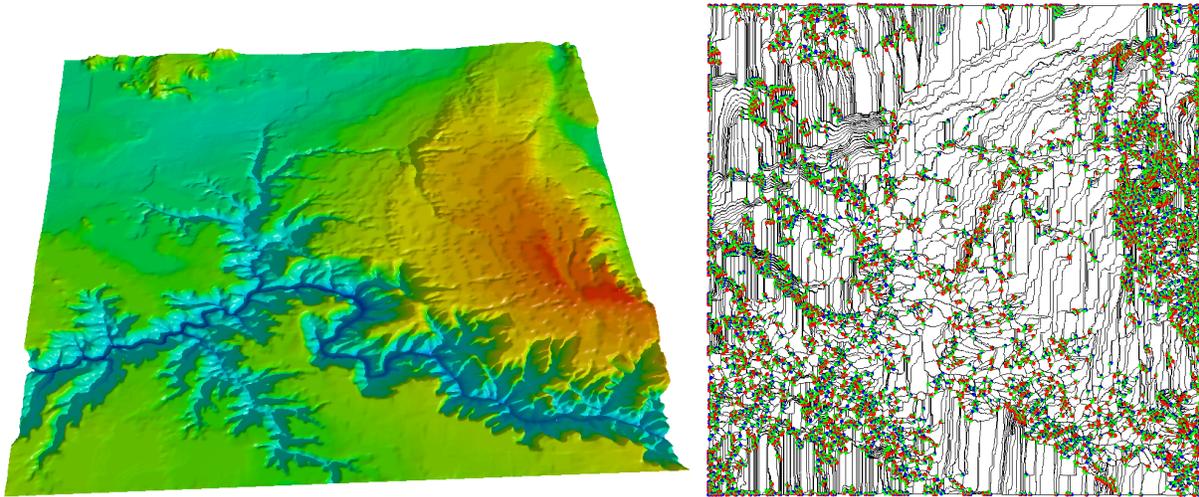


Figure 10: (a) Rendering of Grand Canyon data set; (b) original MS complex of (a) using 11620 critical points (minima shown in blue, maxima in red, and saddles in green.)

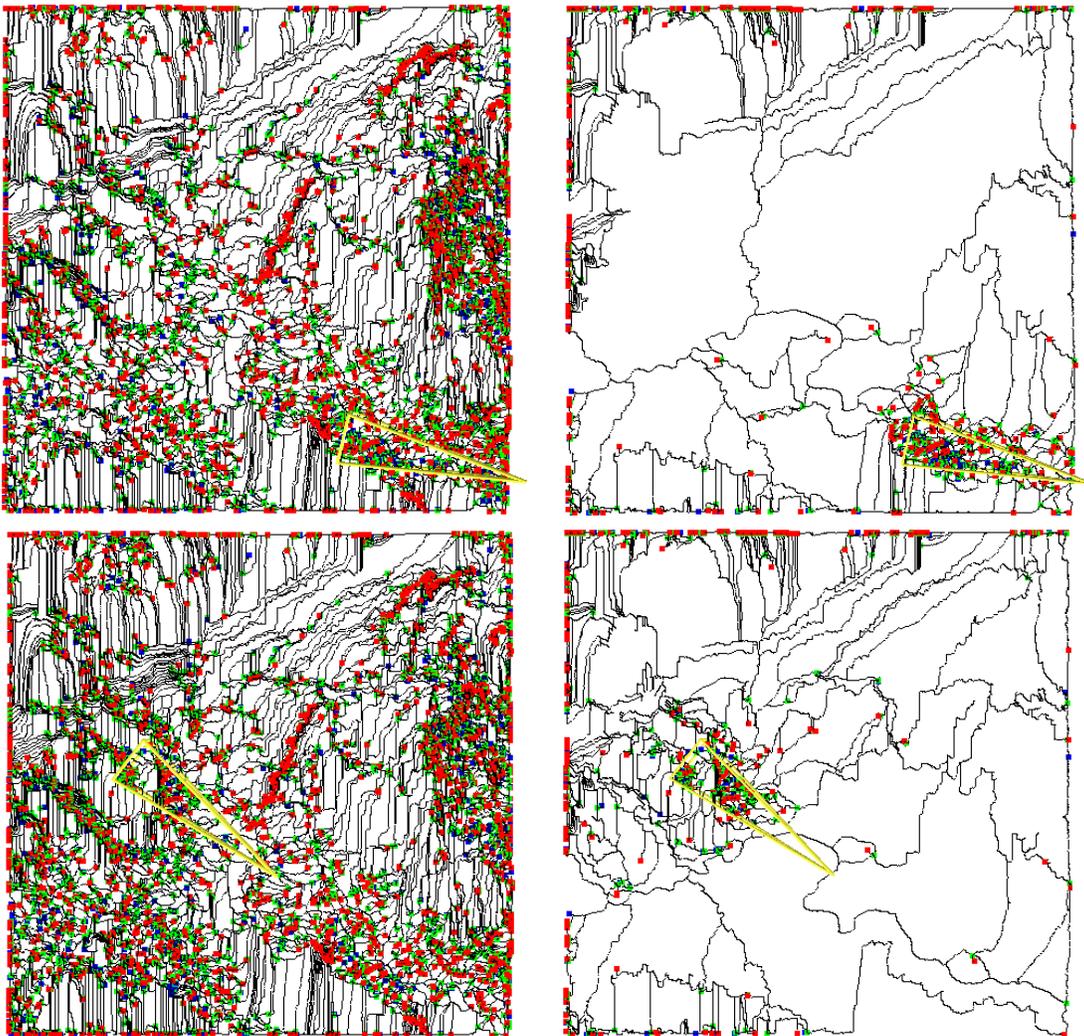


Figure 11: To frames of the fly-over of the Grand Canyon data set. (Left) Using the original hierarchy; (right) using the cancellation forest.

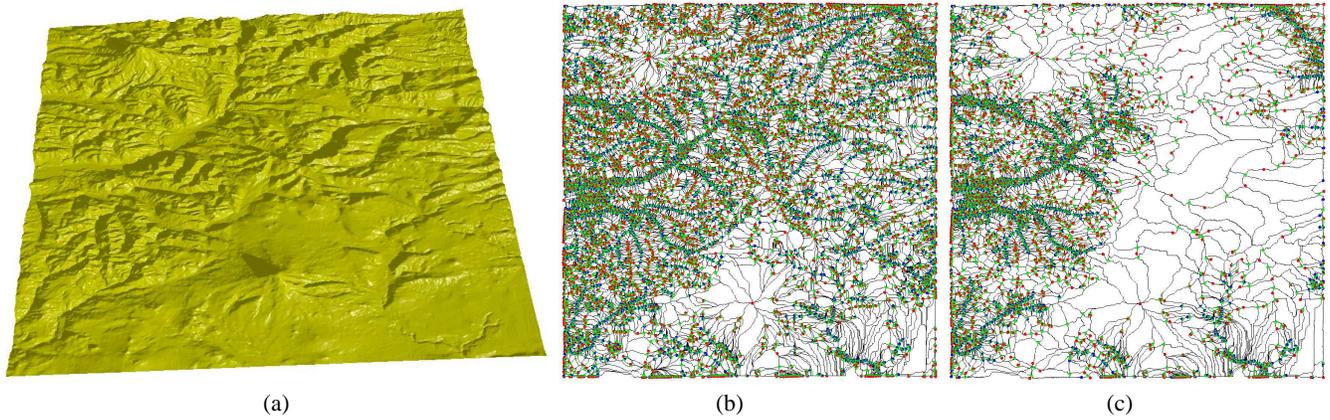


Figure 12: (a) Rendering of the Yakima data set; (b) original MS complex of (a) (17691 critical points); (c) adaptively refined MS complex, where only features below function value of 0.14 are preserved (8063 critical points).

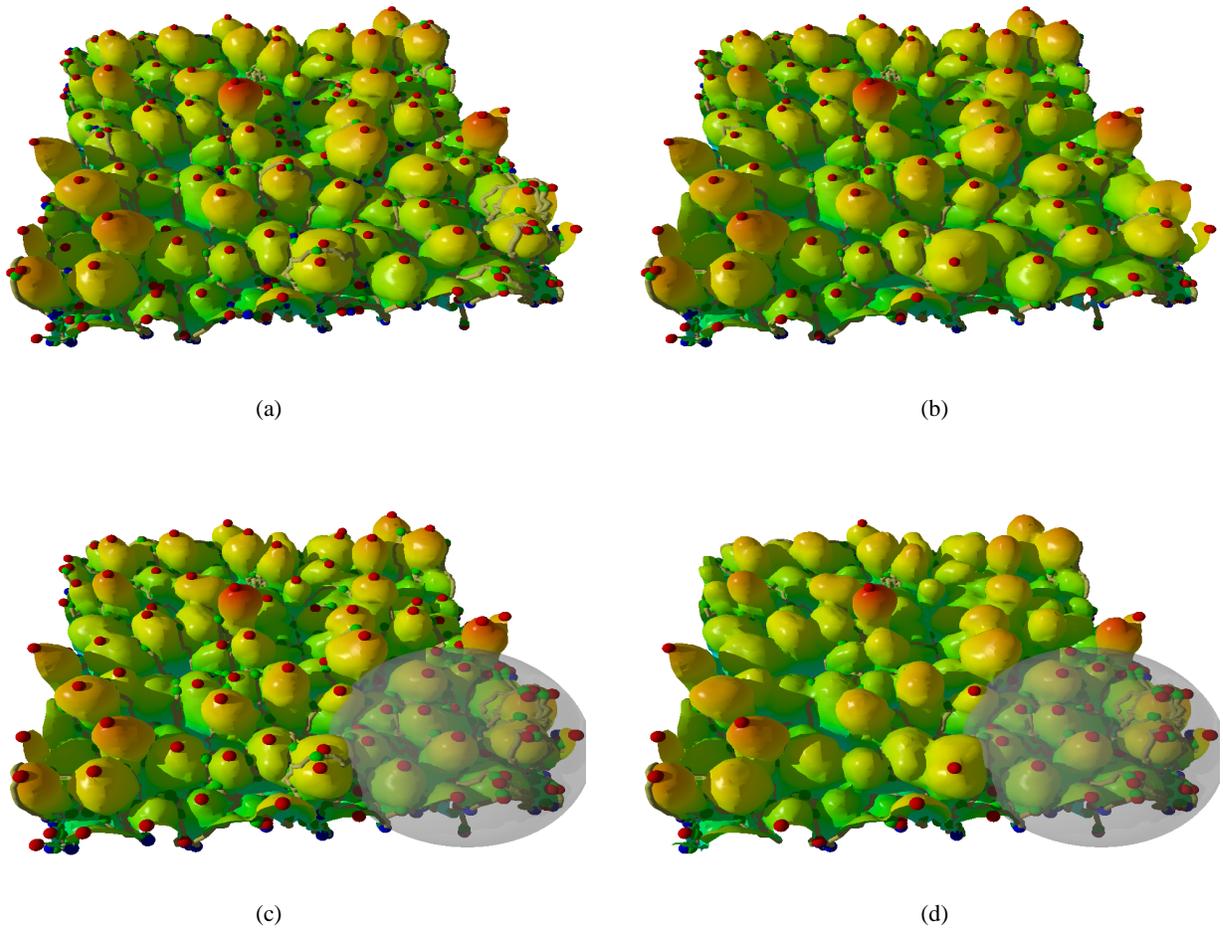


Figure 13: Color-mapped rendering of the Mixing Fluids data set including critical points and descending arcs. (a) Highest resolution with 252 maxima; (b) simplified data with 140 maxima showing one maximum for each "bubble." The bottom row shows the same data set locally refined to only preserve critical points within the focus sphere shown in grey. (c) Using the hierarchy of [4] with 155 maxima; (b) using the cancellation forest with 80 maxima.

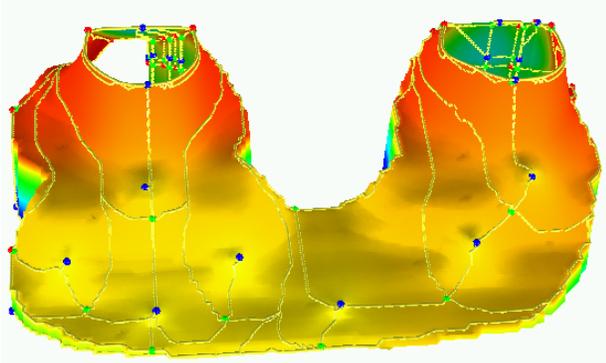


Figure 9: Pseudo-colored rendering and simplified MS complex of oil-pressure data set.