Conforming Morse-Smale Complexes

Attila Gyulassy, David Günther, Joshua A. Levine, *Member, IEEE*, Julien Tierny, and Valerio Pascucci, *Member, IEEE*

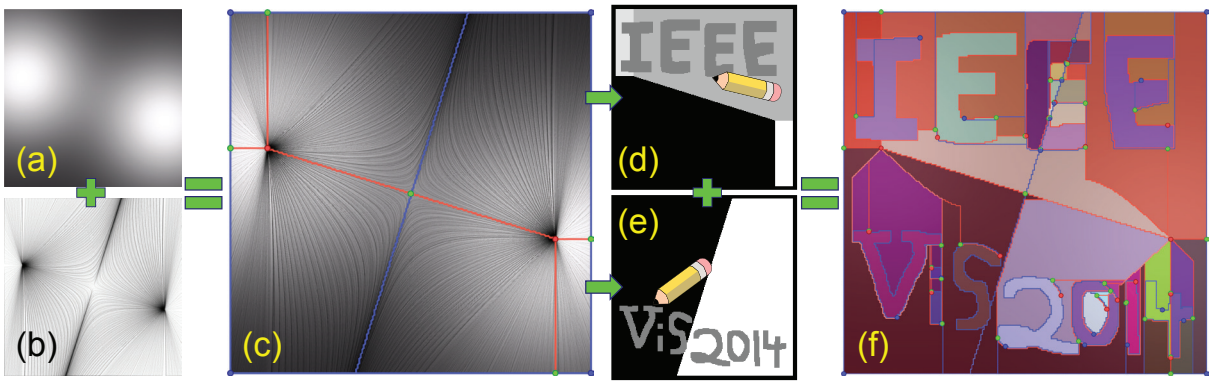


Fig. 1. We present a new approach for computing Morse-Smale (MS) complexes. (b) Streamlines are integrated on the gradient of a scalar function (a) to generate an origin/destination map. (c) This map is used to compute an MS complex that conforms to the numerically-computed geometry. (d,e) The ascending/descending manifolds of the MS complex form one such origin/destination map, but our algorithm allows for arbitrary maps that enable a user to edit the MS complex. (f) Using the edited maps and using them as input for another MS complex computation, we can construct MS complexes that conform with any user-specified map.

Abstract— Morse-Smale (MS) complexes have been gaining popularity as a tool for feature-driven data analysis and visualization. However, the quality of their geometric embedding and the sole dependence on the input scalar field data can limit their applicability when expressing application-dependent features. In this paper we introduce a new combinatorial technique to compute an MS complex that conforms to both an input scalar field and an additional, prior segmentation of the domain. The segmentation constrains the MS complex computation guaranteeing that boundaries in the segmentation are captured as separatrices of the MS complex. We demonstrate the utility and versatility of our approach with two applications. First, we use streamline integration to determine numerically computed basins/mountains and use the resulting segmentation as an input to our algorithm. This strategy enables the incorporation of prior flow path knowledge, effectively resulting in an MS complex that is *as geometrically accurate* as the employed numerical integration. Our second use case is motivated by the observation that often the data itself does not explicitly contain features known to be present by a domain expert. We introduce edit operations for MS complexes so that a user can directly modify their features while maintaining all the advantages of a robust topology-based representation.

Index Terms—Computational Topology, Morse-Smale Complex, Data Analysis

1 Introduction

Topology-based techniques for visualization and analysis of scalar functions have demonstrated their usefulness due to the robustness induced by their combinatorial nature as well as their ability to both extract features of interest and analyze them at different scales. One family of topology-based techniques uses the Morse-Smale (MS) complex. An MS complex of a scalar function f defined on a domain D partitions the domain according to the flow behavior of the gradient of f. Recent advances simplifying the construction and increasing the accuracy of MS complexes have enabled their use in a wide variety of scientific domains. For example, the MS complex was used in characterizing bubble formation in a Rayleigh-Taylor instability [31], extracting the core structure of a porous material [24], understanding the filamentary structure of the cosmic web [45], and more recently in characterizing dissipation structures in turbulent flow [22].

A key component of MS complex computation algorithms is the construction of a discrete representation of the gradient of f, which governs both the geometrical properties of the output MS complex and its structural connectivity. While recent approaches improve the geometric accuracy of computed MS complexes, they produce improved geometry either at the expense of a costly, inherently serial computation, or only converge in a probabilistic sense as the sampling is refined [21,37]. A direct numerical solution has also been elusive, since numerical techniques tend to face problems when dealing with degenerate conditions, such as flat regions or multi-saddles, and also face topological challenges, such as non-intersection of streamlines.

Using the MS complex in the analysis of imaged scientific data poses additional challenges. The MS complex is an impartial arbiter of features: only if they exist in the scalar function will they be computed and extracted for analysis. However, errors in the imaging process often cause segmentation artifacts in the domain, such as in the cases of multiple objects overlapping in their two-dimensional projection, insufficient dye penetration, shadows, or diffraction. Indeed, often such data is preprocessed in an analysis pipeline to resolve many of these problems, but even preprocessing may introduce further issues. By carefully selecting parameters for both preprocessing and MS complex simplification/filtering, one may attain very high rates of precision and recall. In such cases, bridging the gap to 100% accuracy may be impossible as valid features may not be sufficiently separated from

- *Attila Gyulassy is with SCI Institute, University of Utah. E-mail: jediati@sci.utah.edu.*
- *David Günther is with Institut Mines-Télécom, Télécom ParisTech, CNRS LTCI, Paris, France. E-mail: gunther@telecom-paristech.fr.*
- *Joshua A. Levine is with School of Computing, Visual Computing Division, Clemson University, Clemson, SC, USA. Email: levinej@clemson.edu*
- *Julien Tierny is with Institut Mines-Télécom, Télécom ParisTech, CNRS LTCI, and CNRS LIP6, UPMC, Sorbonne Universites, Paris, France. E-mail: tierny@telecom-paristech.fr.*
- *Valerio Pascucci is with SCI Institute, University of Utah. E-mail: pascucci@sci.utah.edu*

Manuscript received 31 Mar. 2014; accepted 1 Aug. 2014. Date of publication 11 Aug. 2014; date of current version 9 Nov. 2014. For information on obtaining reprints of this article, please send e-mail to: tvcg@computer.org.
Digital Object Identifier 10.1109/TVCG.2014.2346434

non-features in the parameter space.

The two problems, computing geometrically accurate MS complexes and correcting for deficiencies in the analysis of imaged data, at first seem unrelated. Herein we argue that they both actually share a common underlying model. In each case, the quality of the MS complex is determined by how well it conforms to some *a priori* domain-specific knowledge of a segmentation of the image. For instance, the geometric accuracy of an MS complex can be defined by how well the MS cells match a segmentation of the domain into regions labeled by which pair of limit sets are approached by continuous flow paths that follow the gradient. In the case of analysis of imaged data, a domain expert's knowledge and understanding of confounding factors allows him/her to distinguish regions of the domain belonging to distinct features. Such features can be similarly described by a (user-driven) segmentation (i.e. "*that* region must be distinct from *this* part of the domain"). Thus in this work, to address both problems we propose a technique to faithfully respect such knowledge in the computation of the MS complex. Furthermore, we demonstrate the utility of our approach first in computing MS complexes with accurate geometry, and second, in enabling editing of existing MS complexes.

Contributions:

- We introduce an algorithm that constructs discrete gradient fields such that they conform to any user-supplied labeling of the domain.
- To illustrate the utility of our approach, we first show that our algorithm enables construction of MS complexes that agree geometrically with numerically-computed streamlines both in two and three dimensions.
- To further demonstrate the versatility of our approach, we show that this same algorithm enables user editing of an existing MS complex to correct for data idiosyncrasies.

2 Related Work

It has only recently become possible to use MS complexes in the analysis of large, complex data, as the concepts have evolved from intuitive descriptions for terrains to theorems and algorithms for the sampled datasets of modern computational science. Terrain decompositions relating to gradient flow properties were first discussed by Cayley [9] and Maxwell [33]. These descriptions were formally defined in terms of the Morse-Smale (MS) complex by Smale [42, 43]. The first robust algorithm for computing the MS complex for two-dimensional, piecewise-linear (PL) data was presented by Edelsbrunner et al. [13]. The geometric accuracy of this technique was improved by Bremer et al. [6] with an approach that subdivided triangles to follow the numerically computed PL gradient. While a similar approach for volumetric PL data was proposed [12], no practical implementation of the algorithm has been achieved.

Forman's discrete Morse theory [15] fully brought the concepts of gradient flow to the discrete domains. This gave rise to several algorithms to construct a discrete gradient field and extract the MS complex using a steepest-descent technique [19, 23, 30, 32, 38, 40, 41]. The main shortcoming of these approaches is the poor geometric reconstruction of gradient flow features, with results strongly biased in the grid directions. To mitigate the biasing problem of steepest-descent, a randomized approach was independently introduced by Reininghaus et al. [37], for two-dimensional simplicial complexes, and by Gyulassy et al. [21], for volumetric meshes. This technique was shown to produce a better geometric reconstruction, and proven to converge to the smooth flow features under mesh refinement [21]. Sousbie et al. [46] introduced a technique that integrated the probabilities of a randomized selection to produce watershed transforms with accurate geometry. Such a construction was extended to discrete gradient fields by Gyulassy et al. [21], achieving the highest geometric quality for computed MS complexes. The stochastic constructions [21, 46] share similarities as well as limitations, the primary of which is their inherent serial nature, relying on a global sort, that leads to a high computational cost.

One of the major advantages of using a topology-based approach for analysis is the ability to represent the function at multiple topological scales. Edelsbrunner et al. [13] introduced critical point cancellation for two-dimensional MS complexes. Bremer et al. [6] extended this approach to a full multi-resolution hierarchy with independent cancellations. Gyulassy et al. [26] introduced three-dimensional cancellation operations on the algebraic structure of an MS complex, and showed how to compute the geometric embedding of features at any level of a cancellation sequence [25]. In his seminal work on discrete Morse theory, Forman [15] introduced a generic cancellation as a reversal of a V-path. Günther et al. [18] showed the differences between this approach and the algebraic one for three-dimensional MS complexes. While cancellation operations are typically ordered by the difference in function values of the critical points being canceled, alternative orderings have proven useful in analysis, such as restricting cancellations that cross user defined thresholds [24], or computing alternative weights based on separatrix persistence [20, 51].

Various components of the MS complex have been used in a wide variety of scientific domains to extract features. For example, Laney et al. [31] use the descending manifolds from maxima to identify the evolution of bubbles in Rayleigh-Taylor instabilities. The same kind of structures were used by Bremer et al. [8] to identify flame surfaces in turbulent combustion simulations, and Kasten et al. [29] to study vortical structures. The arcs and nodes of the MS complex were used by Gyulassy et al. [24] used to study the core structure of porous media, and by Sousbie [45] to study filamentary structure in cosmology simulations. Recently, Gyulassy et al. [22] re-defined dissipation elements used in the study of turbulence structures as crystals of the MS complex, and illustrated their use in understanding of the stability of subsequent analysis.

Beyond the simplification of topological abstractions (contour trees, Reeb graphs, Morse-Smale complexes), several approaches have been proposed to reconstruct a scalar field from the input which reflects the result of the abstract topological simplification. In particular these techniques aim at reconstructing a scalar field as close as possible from the input for data fitting purpose, such that only a specific set of critical points is maintained. While this problem can be cast as a numerical optimization problem [7, 16, 34, 35, 50], several combinatorial techniques have been proposed for piecewise-linear scalar fields [1, 14, 44, 49], filtrations [4] or discrete Morse functions [5]. All of these works only addressed two-dimensional functions. For three-dimensional functions, this problem has been recently shown to be NP-hard [3], which challenges the design of a practical algorithm. In contrast to existing scalar field reconstruction techniques, our approach does not aim at enforcing the cancellation or maintenance of critical points but rather of separatrices.

3 Background

Scalar-valued volumetric data is most often available as discrete samples at the vertices of an underlying mesh. The adaptation of Morse theory from smooth scalar functions to these discrete domains has focused on maintaining the combinatorial properties of the theory. We first present some basic definitions from smooth Morse theory, and then present the discrete analogue.

3.1 Morse Functions and the MS Complex

Let f be a real-valued smooth map $f : \mathbb{M} \to \mathbb{R}$ defined over a compact d-manifold $\mathbb{M}$. A point $p \in \mathbb{M}$ is *critical* when $|\nabla f(p)| = 0$, i.e. the gradient is zero, and is non-degenerate when its Hessian (matrix of second partial derivatives) is non-singular. The function f is a *Morse function* if all its critical points are non-degenerate and no two critical points have the same function value. In this case the *Morse Lemma* states that there exists local coordinates around p such that f has the following *standard form*: $f_p = \pm x_1^2 \pm x_2^2 \cdots \pm x_d^2$. The number of minus signs in this equation gives the *index* of critical point p. In three-dimensional functions, minima are index 0, 1-saddles are index 1, 2-saddles are index 2, and maxima are index 3.

An integral line in f is a path in $\mathbb{M}$ whose tangent vector agrees with the gradient of f at each point along the path. The integral line passing through a point p is the solution to $\frac{\partial}{\partial t} L(t) = \nabla f(L(t)), \forall t \in \mathbb{R}$, with initial value $L(0) = p$. Each integral line has an origin and destination point that asymptotically approaches the critical points of f. *Ascending* and *descending* manifolds are obtained as clusters of integral lines having common origin and destination respectively. An index i critical point has an i-dimensional descending manifold and

a $(d-i)$-dimensional ascending manifold. A Morse function f is a *Morse-Smale function* if the ascending and descending manifolds of its critical points only intersect transversally. The descending manifolds of a Morse-Smale function form a cell complex that partitions $\mathbb{M}$; this partition is called the *Morse* complex. Similarly, the ascending manifolds also partition $\mathbb{M}$ in a cell complex.

The intersection of transversal ascending and descending manifolds of f defines the *Morse-Smale (MS) complex*. The MS complex encodes the neighborhood relation of critical points by means of unique integral lines of ∇f, called *separatrices*. For instance, for three-dimensional MS complexes, at each 1-saddle two separatrices terminate connecting the 1-saddle to at most two minima while at a 2-saddle two separatrices emanate connecting the 2-saddle to at most two maxima. Depending on the intersection of the ascending and descending manifolds of a 1- and 2-saddle, respectively, multiple integral lines may exist connecting this saddle-saddle pair.

3.2 Discrete Morse Theory

Discrete Morse theory is at the heart of current techniques for efficiently computing Morse-Smale complexes. We provide a brief overview with basic definitions from Forman [15], and we refer the reader to this introductory work for an intuitive description. An i-cell is a topological space that is homeomorphic to a Euclidean i-ball $B^i = \{x \in \mathbb{E}^i : |x| \leq 1\}$, with $\mathbb{E}^i$ denoting i-dimensional Euclidean space. For cells α and β, $\alpha < \beta$ means that α is a *face* of β and β is a *co-face* of α, i.e. the vertices of α are a proper subset of the vertices of β. If $\dim(\alpha) = \dim(\beta) - 1$ we say α is a *facet* of β and β is a *co-facet* of α, and we denote this $\alpha \dot{<} \beta$. When necessary to clarify the discussion, we may denote the dimension of a i-cell α with $\alpha^{(i)}$. The *star* of a cell α, denoted $St(\alpha)$, is the set of co-faces of α. The *lower star* of α, denoted $St^-(\alpha)$ is subset of $St(\alpha)$ where each element has lower function value. The *link* of α is the closure of the star, minus the star itself, $Lk(\alpha) = \overline{St(\alpha)} - St(\alpha)$. Similarly, the *lower link* of α is $Lk^-(\alpha) = \overline{St^-(\alpha)} - St^-(\alpha)$.

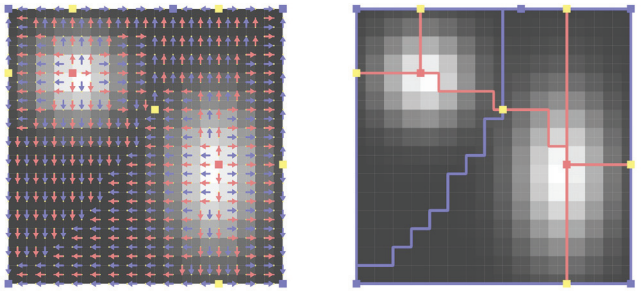
Let K be a regular complex whose geometric realization is a mesh representation of $\mathbb{M}$. The *barycenter* $B : K \to \mathbb{M}$ of an i-cell $\alpha \in K$, $B(\alpha)$, is the average of its vertices. A function $F : K \to \mathbb{R}$ that assigns scalar values to every cell of K is a *discrete Morse function* if for every $\alpha^{(i)} \in K$, its number of co-facets $|\{\beta^{(i+1)} \dot{>} \alpha | F(\beta) \leq F(\alpha)\}| \leq 1$, and its number of facets $|\{\gamma^{(i-1)} \dot{<} \alpha | F(\gamma) \geq F(\alpha)\}| \leq 1$. A cell $\alpha^{(i)}$ is called critical if its number of co-facets $|\{\beta^{(i+1)} \dot{>} \alpha | F(\beta) \leq F(\alpha)\}| = 0$ and its number of facets $|\{\gamma^{(i-1)} \dot{<} \alpha | F(\gamma) \geq F(\alpha)\}| = 0$. A critical cell $\alpha^{(i)}$ has index of criticality equal i.

A *vector* in the discrete sense is a pairing of cells $\langle \alpha^{(i)}, \beta^{(i+1)} \rangle$, where $\alpha \dot{<} \beta$. We say that an arrow points from $\alpha^{(i)}$ to $\beta^{(i+1)}$. The *direction* of the arrow relates the combinatorial notion of the pairing to the geometric interpretation of the flow, and is given by $B(\beta^{(i+1)}) - B(\alpha^{(i)})$. Intuitively, this vector simulates a direction of flow, see Fig. 2(a). A *discrete vector field* V on K is a collection of pairs $\langle \alpha_j^{(i)}, \beta_j^{(i+1)} \rangle$ of cells of K such that each cell is in at most one pair of V. All unpaired cells are critical cells. In Fig. 2(a), the minima, saddles, and maxima are shown as big blue, yellow, and red squares, respectively. Given a discrete vector field V on K, a *V-path* is a sequence of cells

$$\alpha_0^{(i)}, \beta_0^{(i+1)}, \alpha_1^{(i)}, \beta_1^{(i+1)}, \alpha_2^{(i)}, \ldots, \beta_r^{(i+1)}, \alpha_{r+1}^{(i)}$$

such that for each j = 0,..., r, the pair $\langle \alpha_j^{(i)}, \beta_j^{(i+1)} \rangle \in V$, and $\alpha_j^{(i)}$ and $\alpha_{j+1}^{(i)}$ are both facets of $\beta_j^{(i+1)}$. A discrete vector field in which each V-path has disjoint start and end-points is a *discrete gradient field*, denoted G, of a discrete Morse function. Note that this condition implies that a discrete gradient field G does not contain any loops. When constructing a discrete gradient field, we say that G is *valid* if this condition is met.

The discrete gradient field G is the combinatorial analogue of ∇f, enabling the notions introduced in Section 3.1 to transfer to the discrete setting. Since a V-path is the discrete equivalent of a continuous integral line, we also call it a *discrete integral line*. Thus, *discrete ascending* and *descending manifolds* are given by the collection of discrete integral lines having the same critical cell as origin and destination, respectively. Similar to the continuous setting, the discrete ascending and descending manifolds form a partition of $\mathbb{M}$. Their intersection defines the *discrete MS complex*. The separatrices of the discrete MS complex are discrete integral lines connecting critical cells. An illustration of a discrete MS complex is given in Fig. 2(b).



(a) Discrete gradient field. (b) Discrete MS complex.

Fig. 2. Discrete gradient field of a simple synthetic function. Blue arrows illustrate the pairing of 0-cells (vertices) with 1-cells (edges) while red arrows show the pairing of 1-cells (edges) with 2-cells (faces). Critical i-cells are shown as blue ($d = 0$), yellow ($d = 1$), and red ($d = 2$) squares. The discrete gradient field defines the MS complex – combinatorial separatrices (blue and red lines) connecting the critical cells.

3.3 Simplification

A function f is simplified by repeated cancellation of pairs of critical points connected by a single arc in the MS complex. The local change in the MS complex indicates a smoothing of the gradient vector field and hence of the function f. Forman [15] showed how a cancellation could be achieved in a discrete gradient field by reversing the gradient path between two critical cells. A full characterization of cancellation operations in terms of how they affect the connectivity of the complex and the geometry of the ascending/descending manifolds was introduced in two dimensions by Edelsbrunner et al. [13], and in three dimensions by Gyulassy et al. [23, 26]. These cancellation operations act solely on the combinatorial structure of the complex: each cancellation removes a pair of critical points, reconnects arcs of the complex, and merges their ascending and descending manifolds with their neighbors geometry. Repeated application of cancellations in order of *persistence*, the absolute difference in function value of the canceled critical points, results in a hierarchy of MS complexes and a multi-resolution representation of features.

4 Conforming MS Complex

Our approach to compute geometrically accurate MS complexes and to enable user editing relies on constructing a discrete gradient vector field that mirrors a user-supplied labeling of cells of a mesh. To define the concept of a conforming MS complex, we first introduce conforming discrete gradient vector fields. Given a regular complex K whose geometric realization is a mesh representation of $\mathbb{M}$, let $f : K^0 \to \mathbb{R}$ be a scalar function mapping vertices to values. Furthermore, let $I : K \to S$ be a surjective map from cells of K to an index set $S \subset \mathbb{Z}$.

Conforming Discrete Gradient: We define a discrete gradient vector field V as *conforming to a map I* if and only if for each discrete vector $\langle \alpha^i, \beta^{i+1} \rangle \in V$, $I(\alpha^i) = I(\beta^{i+1})$, meaning both the head and tail of a discrete vector have the same label in I.

We remark that the space of discrete gradient vector fields is the same as the space of conforming discrete gradient vector fields. This is easily seen, since (1) every conforming discrete gradient vector field is also a discrete gradient vector field, and (2) any discrete gradient vector field V is also a conforming discrete vector field under the map $I(\alpha) = 1$. Since the MS complex Γ is uniquely determined by a discrete gradient vector field V, we say Γ *conforms* to a map I if and only if V also conforms I.

4.1 Algorithm

Computing discrete gradient vector fields so that they conform to any arbitrary map I involves only a slight variation of previous techniques. In particular, the added restriction that a gradient vector $\langle \alpha, \beta \rangle$ can only be created if $I(\alpha) = I(\beta)$ by definition creates a conforming discrete gradient. Note that any valid discrete gradient V can be converted to a conforming one by simply making each cell α and β critical when $\langle \alpha, \beta \rangle$ and $I(\alpha) \neq I(\beta)$. However, such a trivial modification does not represent how the flow behaves when restricted to the boundary of a labeled section. In the following algorithm, we create a discrete gradient vector field that conforms to I but nevertheless avoids creating critical cells where a discrete gradient vector $\langle \alpha, \beta \rangle$ can be created where $F(\beta) \leq F(\alpha)$ and $I(\alpha) = I(\beta)$, i.e., there exists *some* gradient flow restricted to the label $I(\alpha)$.

Our algorithm is inspired by Robins et al. [38]: first create a discrete gradient vector for a vertex-edge pair and then perform simple homotopic expansions in the lower star. The main difference is that we only consider for pairing those cells that also share the same label. The Robins algorithm has a strict ordering for performing homotopic expansions in the lower star, and straightforward restriction of pairing based on labels would create many extra critical cells. We instead take a very pragmatic approach to reducing the number of spurious critical cells produced by reordering the homotopy expansion to perform any possible pairing in the lower star that is a simple homotopy expansion (the inverse of homotopic collapses [10]) yet also conforms to the labeling.

In the following, we denote a cell that has been identified as critical by pairing it with itself, for example, $\langle \alpha, \alpha \rangle$. Furthermore, a cell is *assigned* if and only if it has been identified as critical or paired in a discrete gradient vector. The function $\#UCF(\gamma)$, (Number of Unassigned Conforming Facets) counts the number of facets $\beta \prec \gamma$ of a cell restricted to the lower star $St^-(\alpha)$ such that β has not been assigned and $I(\beta) = I(\gamma)$.

Algorithm 1 ConformingGradient(K, F, I)

```
1:  V = {}
2:  for α ∈ K^0 do
3:      S = {β ∈ St^-(α) | I(α) = I(β), α ≺ β}
4:      if S = ∅ then
5:          V = V ∪ ⟨α, α⟩
6:      else
7:          V = V ∪ ⟨α, β⟩ where β ∈ S is in direction of steepest descent
8:      for i ∈ [1, ..., d] do
9:          while ∃ unassigned β^i ∈ St^-(α) do
10:             while S^{i+1} = {γ^{i+1} ∈ St^-(α) | #UCF(γ) = 1} ≠ ∅ do
11:                 V = V ∪ ⟨β^i, γ^{i+1}⟩, β^i is the unassigned conforming
                    facet of γ^{i+1} ∈ S^{i+1}
12:             if ∃ unassigned β^i ∈ St^-(α) then
13:                 V = V ∪ ⟨β^i, β^i⟩, β^i is unassigned
14: return V
```

The algorithm processes each vertex independently, first creating a vertex-edge vector in the direction of steepest descent (Lines 3-7), restricted to the set of edges sharing the same segmentation label as the vertex. If no pairing for the vertex is possible, it is made critical (Line 5). Next, simple homotopy type expansions are performed in order of increasing dimension (Lines 8-13), again restricting possible candidates for pairing to those sharing the same segmentation label. For each dimension i, while there exists unassigned i-cells in the lower star of α, simple homotopy expansions of an unassigned i-cell with unassigned $i+1$ cells are attempted (Lines 10-11), marking the i-cell critical (Line 13) when such an expansion is not possible. The test to check if there exist unassigned i-cells in the lower star of α can be implemented by placing the i-cells in $St^-(\alpha)$ in a list, whose size is typically bounded by a small constant. The output is guaranteed to produce a discrete gradient vector field, since all pairings are restricted to the lower star of a vertex and a homotopy expansion is only performed when all faces of the i-cell have previously been assigned, and the $i+1$-cell has only one unassigned face. These two conditions along with the fact that every cell of the domain is either paired or marked critical ensure that all V-paths produced are monotonically decreasing and V is acyclic, hence a discrete gradient vector field (just as in [38]). Algorithm 1 can be applied to every vertex in the domain in an embarrassingly parallel manner.

This same kind of modification is also possible for the steepest descent gradient constructions by Lewiner [32] and Gyulassy et al. [23], the randomized algorithm by Reininghaus [37], and also the geometrically accurate algorithm by Gyulassy et al. [21]. Just as in our *ConformingGradient*() algorithm, the modifications to these include checking the segmentation label when assigning the first vertex-edge gradient vector, and subsequently checking the label when assigning the higher dimensional vectors during simple homotopy expansion. Furthermore, slight reorderings may be necessary to remove spurious critical points.

Once the discrete gradient vector field has been computed, any one of the algorithms for traversal of the discrete gradient field [17, 23, 38, 41] can be used to compute the conforming MS complex.

5 MS Complexes that Conform to Numerically-Computed Geometry

The first application of our conforming algorithm is to compute a discrete gradient vector field and associated MS complex that agrees with a numerically integrated segmentation. Historically, discrete techniques have been used to compute the MS complex because sampled data contains degeneracies and special cases that confound a purely numeric approach, such as flat regions, non-transversal intersections, and multi-saddles. We are able to leverage both the geometric accuracy of numeric integration as well as the robustness of combinatorially computed discrete gradient fields by using the numerics to construct a label map I that encodes the geometry of the data. By conforming to I, the MS complex produced can then benefit from the numerically computed geometry.

This process begins by numerically integrating two sets of streamlines. The first set are integrated in the positive gradient direction from every vertex of K, and the second are integrated in the negative gradient direction from the center of every d-cell in K. Each vertex and d-cell is labeled according to the extremum at which its streamline terminates. These streamlines allow us to construct a *termination map*, $\omega : K^{0,d} \rightarrow S \subset \mathbb{Z}$, that stores a labeling of the destination of each vertex and d-cell.

We then use this labeling to construct a map ∂ that records whether each cell (of any dimension) belongs to the boundary of an ascending or descending d-manifold. This map is the input for our *ConformingGradient*() algorithm to generate a discrete gradient vector field such that discrete flow (in the form of V-paths) does not cross any of the boundaries of the initial segmentation.

5.1 Numeric Integration to Compute the Termination Map

Given an input scalar field, we use two numerical steps to build the termination map ω. First, we approximate the gradient field given an input scalar field. We use central differences of order $O(h^4)$ in our experiments, accounting for boundaries where sampling is insufficient [39]. We project gradient vectors at points on the boundary that also exit the domain so that their vectors lie on the boundary. This computation produces a gradient field sampled at each vector.

Given this sampled field, we use an adaptively time-stepped Runge-Kutta scheme of order $O(h^2)$ to integrate streamlines. Gradient vectors are linearly interpolated for positions lying off-vertex. In our experiments, each streamline is integrated a user-specified maximum number k steps using this scheme, with early termination for a streamline whose position has converged to a likely extremum, detected by checking that the current streamline has travelled a distance of less than a ε threshold in voxel units (in our experiments, typical parameters for k were anywhere from 100 to 10000 and $\varepsilon = 10^{-12}$).

This technique is meant to serve only as an approximation to the *limit sets* of the gradient field [2]. Indeed, nothing inherent in the procedure is intended to guarantee that all extrema are identified or that each streamline has actually arrived at its limit. In practice, our goal is to find a "best guess" termination position (at the voxel resolution) for each streamline, so a relatively generous error tolerance is provided. The benefit of this approach is simplicity (per-streamline integration

could easily be parallelized e.g. [36]). In fact the combinatorial algorithm need not have a complete geometrical realization of the flow paths, only their termination locations. A user can adjust the error tolerance and iteration limit as their computational budget permits. The output is simply a map that records (a) for each vertex α^0, at which vertex the streamline starting at α^0 going in the negative gradient direction terminates (within a 1/2 grid cell of the vertex) , and (b) for each d-cell α^d, at which d-cell the streamline starting at α^d going in the positive gradient direction terminates.

5.2 Boundary Map

Gyulassy et al. [21] constructed geometrically accurate MS complexes by carefully selecting discrete gradient vectors such that they do not cross ascending/descending manifold boundaries. Our construction is similar. Instead of integrating probabilities to obtain the likely destination of integral paths, the map ω provides this information directly. In particular, we use ω to identify those cells in K that are on the boundaries of ascending/descending d-manifolds. We can extend ω to construct a new label map over all simplices that encodes the appropriate boundary information. In this case, $\partial : K \to \{0,1,2,3\}$ and simply records whether a cell is the boundary of ascending d-manifolds, descending d-manifolds, both, or neither.

$$\partial(\alpha) = \begin{cases} 0 & \text{if } \alpha \text{ is not a boundary cell} \\ 1 & \text{if } \alpha \text{ is a boundary between ascending } d\text{-manifolds} \\ 2 & \text{if } \alpha \text{ is a boundary between descending } d\text{-manifolds} \\ 3 & \text{if } \alpha \text{ is a boundary between both} \end{cases}$$

To compute I given ω, for each cell α we count the number of distinct labels in ω of the vertices and d-cells that are faces and cofaces of α, respectively.

Algorithm 2 BoundaryMap ∂(K, ω)

```
∂ = {}
for α ∈ K do
    if |{ j ∈ ω(γ^0), γ^0 < α }| > 1 then
        ∂[α] := 1
    else
        ∂[α] := 0
    if |{ j ∈ ω(β^d), α < β^d }| > 1 then
        ∂[α] := ∂[α] + 2
return ∂
```

Given the input boundary map ∂, the *ConformingGradient*() algorithm produces a discrete gradient field which benefits from the geometry of the numerical integration. Fig. 3 illustrates this construction: streamlines are integrated in the scalar function (a) to obtain the ascending and descending termination map (b, c), which is used to generate the boundary map (d). The resulting discrete gradient vector field is used to construct the MS complex with accurate geometry (e).

5.3 Restricted Simplification

Although the conforming MS complex agrees with the map ω, it may contain critical points where ω prevented pairing in a decreasing direction, such as shown in figure 4 (e-g). One cause of this effect is that gradient estimated with higher order methods may cause flow from a *higher valued* cell to cross into a *lower valued* cell when integrating in the *positive* gradient direction. In other words, while a specific streamline is monotonic in a higher-order interpolation of the function, the piecewise constant values of the cells in the discrete gradient representation may not be. In the cases where such a scenario occurs, the *ConformingGradient*() algorithm inserts critical points to ensure that all V-paths are monotonic. Other sources of disagreement between the termination map ω and the discrete gradient produced occur where the numerically integrated streamlines violate properties of a Morse function, such as not crossing, or not terminating at an inflection point.

Although critical point cancellations in the order of increasing difference of function value typically remove the spurious critical points, precisely which critical points are kept and which are removed is not well controlled. This matters because the separatrices of the critical points that are *not* also boundaries in the boundary map ∂ are not guaranteed to be geometrically accurate. Fig. 3(e) illustrates such a scenario. Here, the finest resolution MS complex contains separatrices that are not also present in the boundary map, and these have poor geometric embedding. Our solution is pragmatic: first cancel critical point pairs that have identical labels in ∂ before canceling those with different labels. This effectively removes spurious critical points that exist in each connected region of the segmentation. Fig. 3(f) shows a simplified version of (e), and (g) highlights that *every* remaining separatrix in (f) corresponds to a boundary in the boundary map (c).

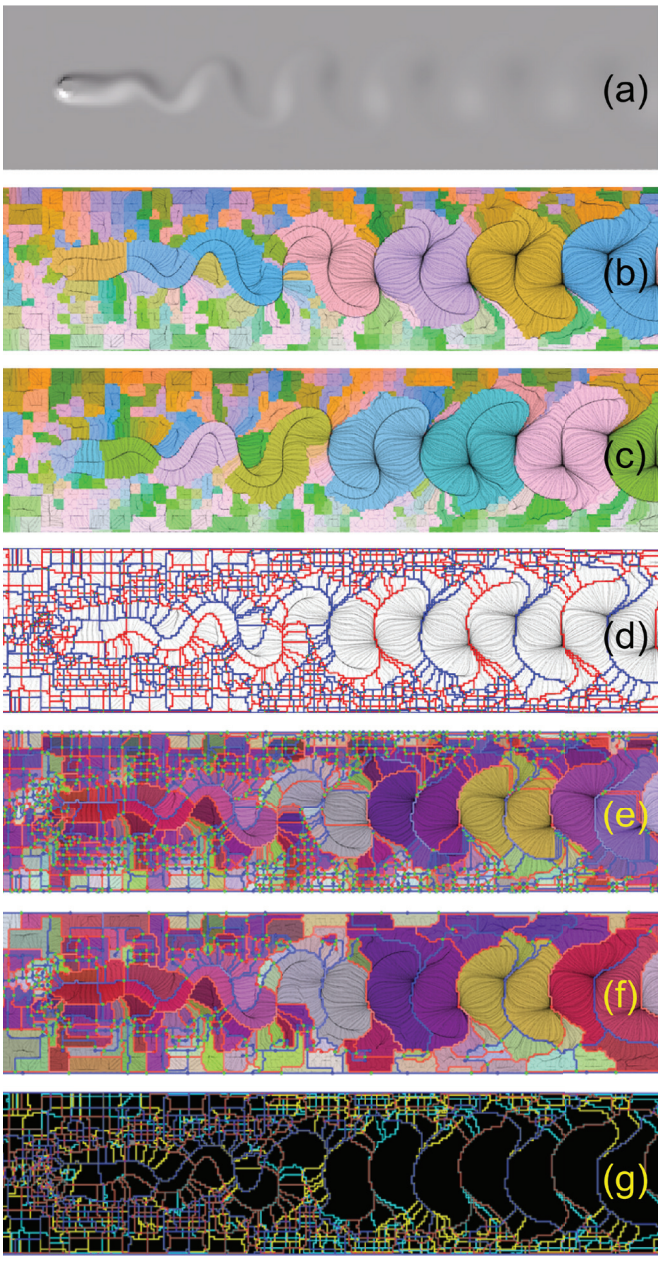


Fig. 3. Vorticity of a flow around a cylinder signed by the rotation direction (a) is shown. The termination map showing sources (b) and sinks (c) computed using streamline integration. The termination map generates a boundary map (d), where red lines indicate ascending boundaries, blue lines indicate descending boundaries, and green points indicating both. The full resolution MS complex is computed (e) using our *ConformingGradient*() algoithm, and simplified (f) using restricted simplification. The arcs of the simplified complex (blue, red) are a subset of the boundary map (cyan, yellow) (g).

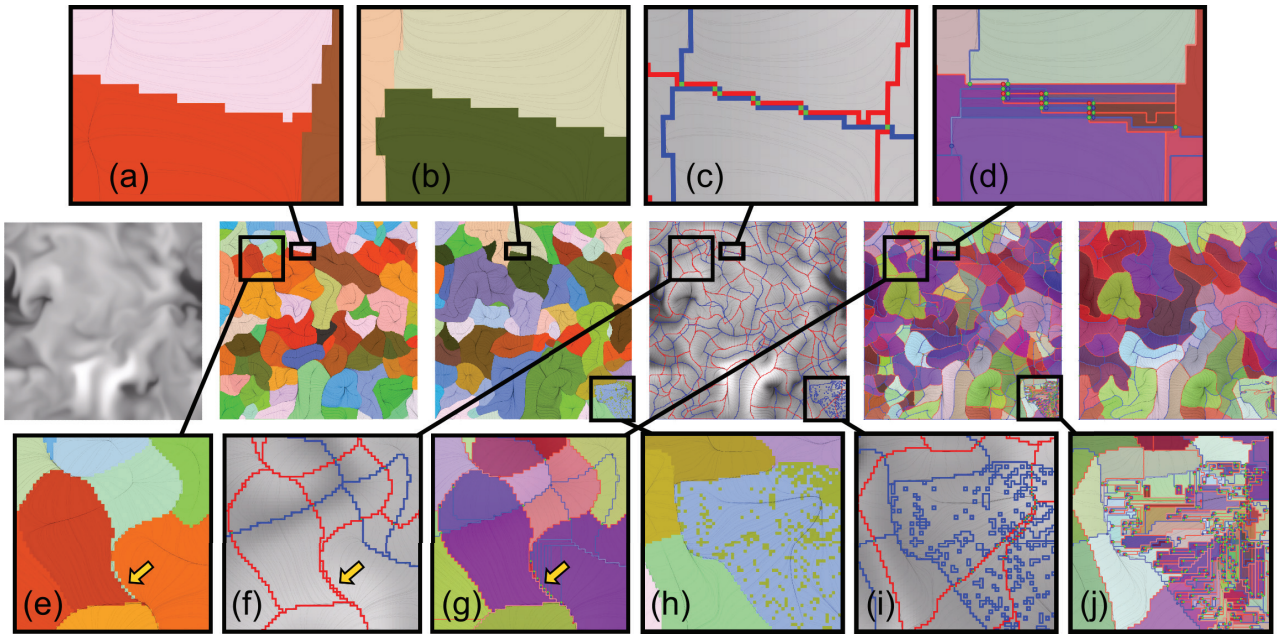


Fig. 4. The conforming MS complex is computed for an HCCI combustion example. The ascending (a) and descending (b) segmentations of the numerically computed termination map can intersect non-transversally. The boundary map (c) enforces transversality by restricting which cells can be ascending and descending boundary. The added constraints generate spurious critical points in the MS complex (d). Furthermore, the numerically computed termination map can represent flow at a finer scale than the mesh resolution, which results in isolated regions in the termination map (e). These isolated regions are generate boundaries (f) and extra critical points in the MS complex (g). Numerically computed streamlines can cross, such as near a maximum that is located exactly on the boundary of two cells, generating spurious regions (h-j).

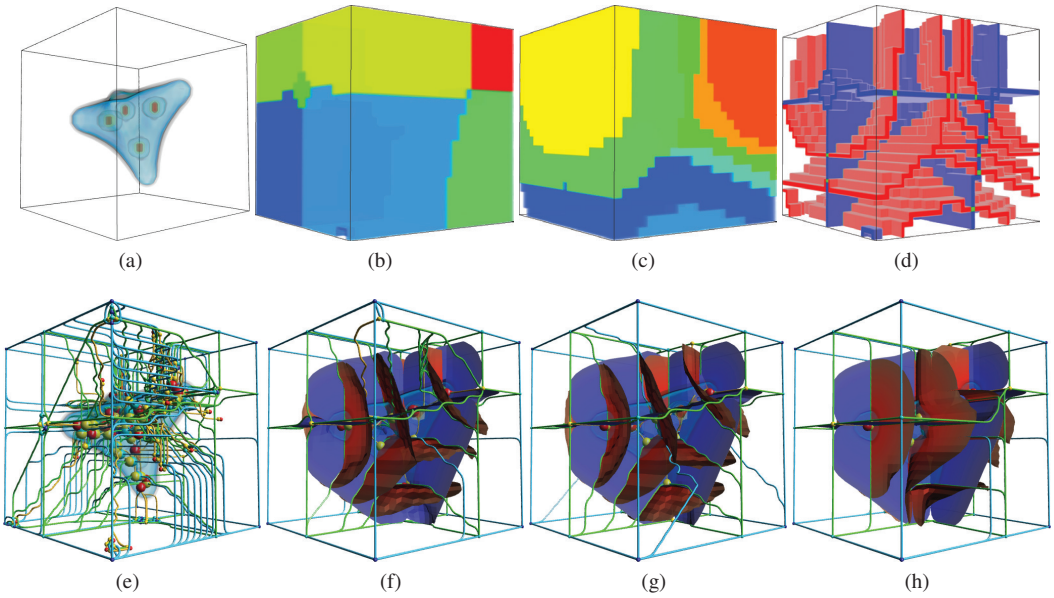


Fig. 5. Three-dimensional MS complexes that conform to numerically-computed geometry are computed the same as their two-dimensional counterparts. The C4H4 molecule $(24 \times 24 \times 24)$ is shown (a), and the ascending (b) and descending (c) components of its termination map computed using streamline integration. The boundary map (d) shows ascending boundaries (blue), descending boundaries (red), and both (green). The finest resolution MS complex (e) is simplified (f), and surfaces separating ascending and descending 3-manifolds are displayed. The accurate geometry [21] algorithm (g) produces similar results, while the geometric embedding of the steepest descent algorithm [38] (h) is more aligned with the grid directions.

5.4 Examples/Discussion

In addition to the two-dimensional flow around a cylinder, Fig. 4 shows the same sequence of operations for a slice of a homogeneous charge compression ignition (HCCI) simulation. This example notably illustrates the conditions where the streamline integration results in a segmentation that is not consistent with a discrete MS complex. For instance, (a-d) shows a non-transversal intersection in the numerically computed ascending and descending 1-manifolds, (e-g) highlights a case where the mesh resolution is not sufficient to resolve narrow features, and (h-j) illustrates a region where the limit of the streamline integration occurs on the boundary between two pixels resulting in two labels in an otherwise homogeneous region. One simple solution to avoid such spurious regions could be to modify the map ω to cluster nearby extrema, although one could envision more robust solutions

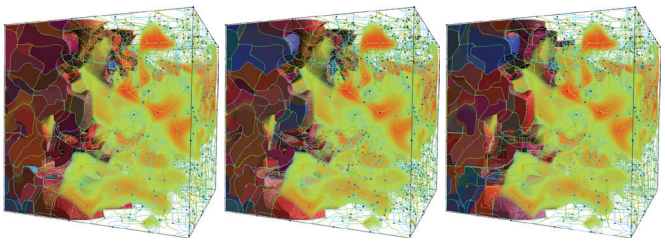


Fig. 6. We compare the conforming (a) to the accurate geometry [21](b) and steepest descent [38] (c) algorithms applied to a three-dimensional HCCI combustion simulation. The streamline integration can be deployed in parallel, resulting in faster time-to-solution than the accurate geometry, while producing a better reconstruction than the steepest descent approach.

combining combinatorial detection and merging of extremal points. Alternatively, one could detect such configurations in the computed MS complex, and perform cancellations to remove those regions.

Fig. 5 shows the same pipeline working in three dimensions. Similar to the two-dimensional case, streamlines are integrated to generate termination maps, which is used to create the boundary map. Just as in the two-dimensional examples, the additional restrictions on the gradient vector pairing creates spurious critical points (Fig. 5(e)). The separatrices of the MS complex are visualized after simplification (f). We compare this result against the accurate geometry algorithm (g) by Gyulassy et al. [21] and the steepest descent algorithm (h) from Robins et al. [38]. The geometric quality of the numerically computed two-dimensional separatrices are similar to those of the accurate geometry algorithm. However, the embedding of the descending 1-manifolds from 1-saddles and ascending 1-manifolds from 2-saddles is more similar to the steepest descent. Indeed, only the 1-saddle-2-saddle lines are guaranteed to be geometrically accurate, as the other lines are entirely *within* sheets of the boundary map, and therefore one degree of freedom of their geometric embedding is controlled by steepest descent. Fig. 6(a-c) shows the same comparison for a $280 \times 280 \times 280$ HCCI combustion simulation. This example illustrates the performance and quality tradeoffs of our numeric approach, having lower quality reconstruction of minimum-1-saddle arcs and maximum-2-saddle arcs, than the accurate geometry approach [21], while the parallel deployment resulted in a faster computation. For the experimental set-up used to produce Fig. 6 the numerically accurate technique parallelized on eight cores completed in 5 minutes (2 for integration + 3 for constrained gradient computation), as opposed to 45 minutes for the geometrically accurate algorithm.

6 Editable MS complexes

Our *ConformingGradient*() algorithm for computing conforming MS complexes can also be used to enable editing of MS complexes. Editing can be used by domain experts to correct errors when doing feature extraction. For instance, in bio-medical imaging, overlapping morphological structures, insufficient dye penetration, shadows, and light diffraction often contribute to poor feature representation during analysis. In this case, *the image data* contains insufficient or incorrect information for an accurate MS complex based feature identification. However, a domain scientist may understand the deficiency and have the interpretation skills necessary to understand what the segmentation *should* be. Often, long pipelines of image processing filters are used to prepare image data for semi-automated segmentation, such as deconvolution, noise removal, color/contrast correction, thresholding, and smoothing. Furthermore, the MS complex itself may be simplified and filtered to extract the desired features. Each stage of this pipeline may depend on several parameters, each one affecting the accuracy and precision of the resulting feature identification, when compared to the gold standard of a domain expert manually identifying features. While most features can be extracted with such an approach, there may still remain problematic cases. In such an instance, we allow the user to edit the segmentation iteratively as part of the analysis pipeline.

Constructing MS complexes that conform to numerical geometry relied on having origin/destination maps that are generated by integrating streamlines. One can use the same construction for building an MS complex conforming to any origin/destination map. Once an MS complex has been computed on a d-dimensional domain, its ascending and descending d-manifolds define an origin/destination map, such that reconstructing the MS complex using that map as a constraint results in the same MS complex. However, it is also possible to *edit* the maps, effectively editing the reconstructed MS complex. The combinatorial nature of the algorithm guarantees that the result is a valid MS complex that conforms to the new edits. In this section, we restrict the discussion to two-dimensional images.

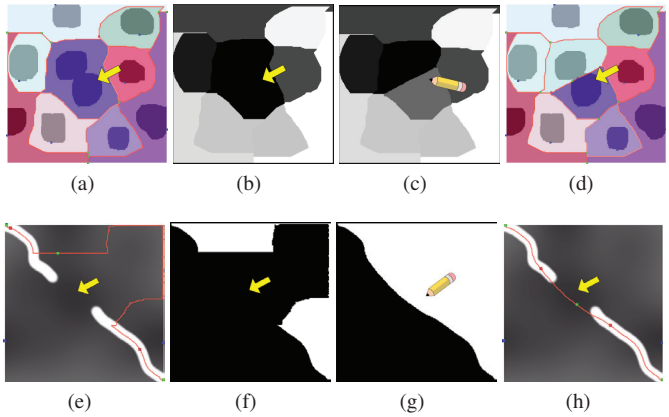


Fig. 7. The MS complex is computed for a simple example showing dark blobs on a light background (a). Two overlapping features are identified as part of the same basin. This MS complex generates a termination map (b), that can be edited by a user (c) to split this region. The edited map is used to generate a new MS complex (d) with the false negative corrected. Similarly, a light ridge-like structure (e) is disconnected, resulting in a ridge reconstruction with poor geometric embedding. The termination map (f) is again edited (g) to reconstruct the desired embedding (h).

6.1 The Identity Maps

Let V be a discrete gradient vector field computed with any algorithm on the mesh K and function f. There are many maps I such that $\mathit{ConformingGradient}(K, f, I)$ produces the exact same discrete gradient vector field V. Two trivial examples are $I(\alpha) = 0$ for any cell α, and the map taking each distinct gradient vector and assigning both cells a their own label. Such maps are called *identity* maps. Our approach for editing the MS complex is to generate an identity-like map from the MS complex, allow the user to modify that map, and then recompute the MS complex with the conforming algorithm.

For our purpose, we use the existing discrete gradient vector field to produce the same termination map ω introduced in Section 5.1. Instead of recording the termination of numerically computed streamlines, we construct the map based on the origin/destination of V-paths in V. Let $M_n = \{\alpha_0, \alpha_1, ..., \alpha_k\}$ be the set of critical vertices, i.e., the minima. For each minimum α_j, we compute the set of vertices belonging to its ascending d-manifold, and assign those vertices the label j. Symmetrically, we compute the descending manifolds of critical d-cells, i.e. the maxima, and assign all d-cells the same label. Given an MS complex Γ constructed from a discrete gradient field V, the termination map constructed from its ascending/descending manifolds is denoted ω_Γ. The boundary map ∂_Γ is constructed from ω_Γ using Algorithm 2 (Section 5.2). For most practical cases in two dimensions, ∂_Γ acts as an identity map, with exceptions occurring at topological strangulations. In practice, this tends to generate spurious low-persistence critical points that are nevertheless removed in subsequent simplification.

6.2 Editing the MS complex

Given an MS complex Γ_i, we produce the map $\omega_{\Gamma i}$, and allow a user to modify it directly to ω_{Γ_i}'. After each edit, the boundary map ∂_{Γ_i}' is recomputed using algorithm 2. The the MS complex Γ_{i+1} is computed

from the gradient field produced by $ConformingGradient(K, f, \partial'_{\Gamma i})$. If persistence simplification/filtering was used to generate Γ_i, that same level of simplification is performed on Γ_{i+1}. Practically, the difference between Γ_i and Γ_{i+1} is restricted to the region of $\omega_{\Gamma i}$ modified by the user. Note that user edits may force changes to the MS complex in non-trivial ways, such as splitting regions and changing the connectivity. For edits such as moving the boundaries of regions, critical points are necessarily added when there is no monotonic path that can be found on the boundary.

An additional simplification rule must be observed to ensure that the edits are not immediately removed as low-persistence features: a critical point of Γ_{i+1} that is a boundary of $\omega'_{\Gamma i}$ but not of $\omega_{\Gamma i}$ can only be canceled with another similarly labeled critical point. Fig. 7 illustrates this process for two simple examples. In the first (a-d), overlapping features cause a false negative to appear, where the finest resolution MS complex image insufficiently segments the domain. The user modifies the termination map $\omega_{\Gamma i}$, and the complex is regenerated with the false negative corrected. In the next example (e-h), the geometric embedding of an arc of the MS complex is fixed by modifying the spatial extents of a label in $\omega'_{\Gamma i}$.

In our examples, editing was accomplished with standard image editing software. In particular, a drawing tool was used to color the map with the grayscale value of the desired label. In the case where an existing feature is split, a new grayscale value was selected for the drawing tool, the separating line was drawn, and the new feature flood-filled with the grayscale value.

6.3 A Histological Example

Applying MS complexes for the automated segmentation of nuclei in phenotypic analysis of histological sections is an active area of research [47]. We apply a straightforward approach, identifying nuclei as simplified basins in a processed image. However, the basins identified often may not sufficiently segment the domain to separate all the nuclei. Our goal is to remove all false negatives using a semi-automated approach. As Fig. 8 illustrates, we apply grayscale conversion, thresholding, and smoothing to the initial image data, and compute basins of the MS complex simplified to some persistence threshold.

This pipeline results in some false negatives where two or more nuclei overlap, as compared to a labeling by a domain expert [52]. The computed MS complex segmentation is used to generate a termination map ω_Γ, then ω_Γ is edited to match the input of a domain expert, the boundary ∂'_Γ is computed from ω'_Γ, and finally the complex Γ' is recomputed and simplified, with the new features present. While the MS complex that results contains no false negatives, it still contains false positives. However, as it is still a valid MS complex, further filtering/simplification is possible to account for such errors. Having every nuclei in its own basin makes it later possible to identify which basins contain nuclei and which do not, for instance, by applying standard filtering/simplification pipelines.

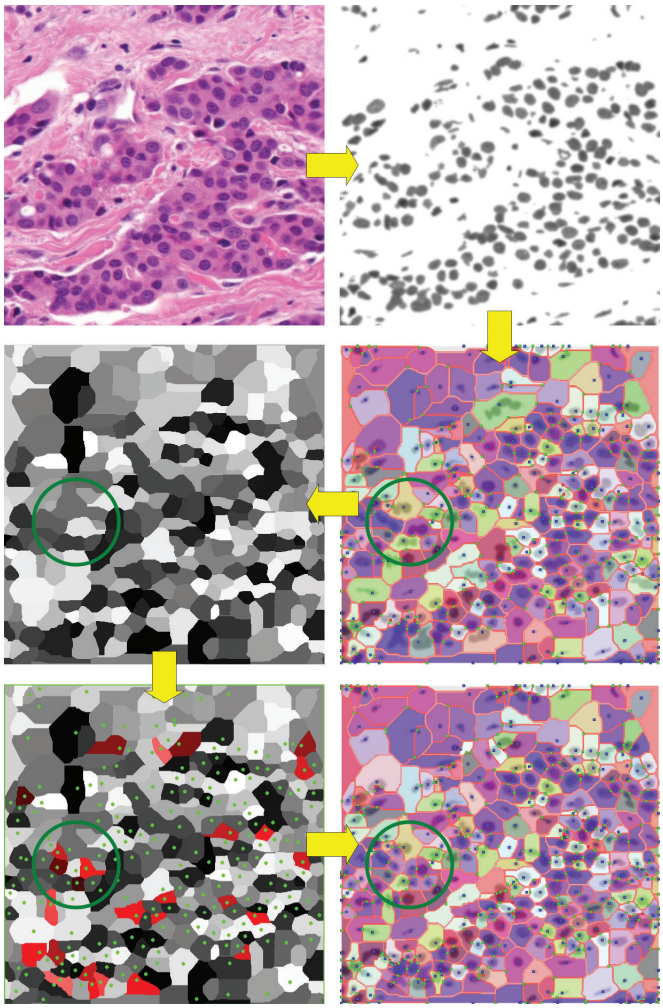


Fig. 8. A section of tissue is imaged in a histology study. We apply grayscale conversion, thresholding, and smoothing to generate an input for an initial MS complex computation. The termination map is edited to match a domain expert's labeling (green dots, bottom left), and a complex is regenerated that has no false negatives. The green circle highlights a region that is corrected through editing.

7 Conclusions/Future Work

We introduced a new algorithm to construct MS complexes that conform with a user-supplied segmentation. For the first time, we are able to compute MS complexes that agree with a numerical streamline-integration based segmentation of the domain. While many challenges remain to develop an efficient parallel technique to perform the streamline integration, a straightforward implementation using OpenMP already results in greater scalability than the inherently serial previous approach. An advantage of this technique is its high level of modularity: any integration scheme can be used to generate the termination map. We plan to investigate approximation methods, such as flow compression, and parallelization strategies, such as GPU deployment, of the streamline tracing. Since the termination map is a sampled approach to the segmentation, our approach enables the investigation of subvoxel accurate computation of MS complexes.

Our approach additionally enables editing of the MS complex in an analysis pipeline. An investigation of the breadth of interactions that must be supported and user interface requirements to make such an approach a viable component of the analyst's toolbox is future work. Additionally, we will explore techniques to restrict the (re-)computation necessary for a local region of influence around the modified segmentation, to enable interactive updates of the MS complex. Furthermore, editing may introduce non-local effects both in the feature generation and simplification process, and fully exploring these in a robust manner would be a valuable future contribution.

Furthermore, this approach offers new tools to investigate simplification of three-dimensional MS complexes. As shown by Joswig and Pfetsch [28], an optimal pairwise simplification is an NP-complete problem. However, the restriction of saddle-saddle connections to a one-dimensional structure in the boundary map may provide insight on what sequence of operations are needed to remove multiply-connected pairs that obstruct simplification.

Finally, another application area for future work involves robust mesh generation. Topological tools have started to emerge as vehicle for surface meshing [11, 27]. We view conforming MS complexes as a first step towards utilizing MS complexes for volumetric mesh generation. User edits with a rich segmentation interface could drive the repair of mesh artifacts interactively, similar to techniques that employ the Reeb graph for interactive quadrangulation [48].

Acknowledgments

This work is supported in part by BNSF CISE ACI-0904631, NSG IIS-1045032, NSF EFT ACI-0906379,DOE/NEUP 120341, DOE/Codesign P01180734, DOE/SciDAC DESC0007446, CCMSC DE-NA0002375, NSF IIS-1314757, NSF IIS-1314896, and by the RTRA Digiteo through the unTopoVis project (2012-063D).

REFERENCES

[1] P. Agarwal, L. Arge, and K. Yi. I/O-efficient batched union-find and its applications to terrain analysis. In *ACM Symp. on Comp. Geom.*, pages 167–176, 2006.

[2] D. Asimov. Notes on the topology of vector fields and flows. Technical Report RNR-93-003, NASA Ames Research Center, 1993.

[3] D. Attali, U. Bauer, O. Devillers, M. Glisse, and A. Lieutier. Homological reconstruction and simplification in R3. In *ACM Symp. on Comp. Geom.*, pages 117–126, 2013.

[4] D. Attali, M. Glisse, S. Hornus, F. Lazarus, and D. Morozov. Persistence-sensitive simplification of functions on surfaces in linear time. In *TopoInVis Workshop*, 2009.

[5] U. Bauer, C. Lange, and M. Wardetzky. Optimal topological simplification of discrete functions on surfaces. *Discrete and Computational Geometry*, pages 347–377, 2012.

[6] P.-T. Bremer, H. Edelsbrunner, B. Hamann, and V. Pascucci. A topological hierarchy for functions on triangulated surfaces. *IEEE Transactions on Visualization and Computer Graphics*, 10(4):385–396, 2004.

[7] P.-T. Bremer, H. Edelsbrunner, B. Hamann, and V. Pascucci. A topological hierarchy for functions on triangulated surfaces. *IEEE Trans. on Vis. and Comp. Graph.*, 10:385–396, 2004.

[8] P.-T. Bremer, G. Weber, V. Pascucci, M. Day, and J. Bell. Analyzing and tracking burning structures in lean premixed hydrogen flames. *IEEE Trans. on Visualization and Computer Graphics*, 16(2):248–260, 2010.

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

[10] M. Cohen. *A course in simple-homotopy theory*. Springer-Verlag, 1973.

[11] S. Dong, P.-T. Bremer, M. Garland, V. Pascucci, and J. Hart. Spectral surface quadrangulation. *ACM Trans. Graph. (Proc. of ACM SIGGRAPH)*, 25(3):1057–1066, 2006.

[12] H. Edelsbrunner, J. Harer, V. Natarajan, and V. Pascucci. Morse-Smale complexes for piecewise linear 3-manifolds. In *Proceedings of the 19th Symposium on Computational Geometry*, pages 361–370, 2003.

[13] H. Edelsbrunner, J. Harer, and A. Zomorodian. Hierarchical Morse-Smale complexes for piecewise linear 2-manifolds. *Discrete Computational Geometry*, 30:87–107, 2003.

[14] H. Edelsbrunner, D. Morozov, and V. Pascucci. Persistence-sensitive simplification of functions on 2-manifolds. In *ACM Symp. on Comp. Geom.*, pages 127–134, 2006.

[15] R. Forman. A user's guide to discrete Morse theory. In *Séminare Lotharinen de Combinatore*, volume 48, 2002.

[16] Y. Gingold and D. Zorin. Controlled-topology filtering. *Computer-Aided Design*, pages 676–684, 2006.

[17] D. Günther, J. Reininghaus, S. Prohaska, T. Weinkauf, and H.-C. Hege. Efficient computation of a hierarchy of discrete 3d gradient vector fields. In *Proceedings TopoInVis*, Zrich, Switzerland, April 2011.

[18] D. Günther, J. Reininghaus, H.-P. Seidel, and T. Weinkauf. Notes on the simplification of the Morse-Smale complex. In *TopoInVis*, March 2013.

[19] D. Günther, J. Reininghaus, H. Wagner, and I. Hotz. Efficient computation of 3D MorseSmale complexes and persistent homology using discrete Morse theory. *The Visual Computer*, 28:959–969, 2012.

[20] D. Günther, H.-P. Seidel, and T. Weinkauf. Extraction of dominant extremal structures in volumetric data using separatrix persistence. *Computer Graphics Forum*, 31(8):2554–2566, Dec. 2012.

[21] A. Gyulassy, P. Bremer, and V. Pascucci. Computing Morse-Smale complexes with accurate geometry. *IEEE Transactions on Visualization and Computer Graphics*, 18:2014 – 2022, 12 2012.

[22] A. Gyulassy, P.-T. Bremer, R. Grout, H. Kolla, J. Chen, and V. Pascucci. Stability of dissipation elements: A case study in combustion. *accepted for publication: IEEE EuroVis 2014*, June 2014.

[23] A. Gyulassy, P.-T. Bremer, V. Pascucci, and B. Hamann. A practical approach to Morse-Smale complex computation: Scalability and generality. *IEEE Transactions on Visualization and Computer Graphics*, 14(6):1619–1626, 2008.

[24] A. Gyulassy, M. Duchaineau, V. Natarajan, V. Pascucci, E.Bringa, A. Higginbotham, and B. Hamann. Topologically clean distance fields. *IEEE Transactions on Computer Graphics and Visualization*, 13(6):1432–1439, 2007.

[25] A. Gyulassy, N. Kotava, M. Kim, C. D. Hansen, H. Hagen, and V. Pascucci. Direct feature visualization using Morse-Smale complexes. *IEEE Transactions on Visualization and Computer Graphics*, 18(9):1549–1562, 2012.

[26] A. Gyulassy, V. Natarajan, V. Pascucci, P.-T. Bremer, and B. Hamann. Topology-based simplification for feature extraction from 3D scalar fields. *IEEE Transactions on Computer Graphics and Visualization*, 12(4):474–484, 2006.

[27] J. Huang, M. Zhang, J. Ma, X. Liu, L. Kobbelt, and H. Bao. Spectral quadrangulation with orientation and alignment control. *ACM Trans. Graph. (Proc. of ACM SIGGRAPH Asia)*, 27(5):1–9, 2008.

[28] M. Joswig and M. E. Pfetsch. Computing optimal Morse matchings. *SIAM Journal on Discrete Mathematics*, 20(1):11–25, 2006.

[29] J. Kasten, J. Reininghaus, I. Hotz, and H.-C. Hege. Two-dimensional time-dependent vortex regions based on the acceleration magnitude. *IEEE Transactions on Visualization and Computer Graphics*, 17(12):2080–2087, 2011.

[30] H. King, K. Knudson, and M. Neza. Generating discrete Morse functions from point data. *Experimental Mathematics*, 14(4):435–444, 2005.

[31] D. Laney, P.-T. Bremer, A. Mascarenhas, P. Miller, and V. Pascucci. Understanding the structure of the turbulent mixing layer in hydrodynamic instabilities. *IEEE Transactions Visualization and Computer Graphics*, 12(5):1052–1060, 2006.

[32] T. Lewiner. Constructing discrete Morse functions. Master's thesis, Department of Mathematics, PUC-Rio, 2002.

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

[34] X. Ni, M. Garland, and J. Hart. Fair Morse functions for extracting the topological structure of a surface mesh. *ACM Trans. on Graph. (Proc. of ACM SIGGRAPH)*, 23:613–622, 2004.

[35] G. Patanè and B. Falcidieno. Computing smooth approximations of scalar functions with constraints. *Computers and Graphics*, 33:399–413, 2009.

[36] T. Peterka, R. B. Ross, B. Nouanesengsy, T.-Y. Lee, H.-W. Shen, W. Kendall, and J. Huang. A study of parallel particle tracing for steady-state and time-varying flow fields. In *International Symposium on Parallel and Distributed Processing*, pages 580–591. IEEE, 2011.

[37] J. Reininghaus, D. Günther, I. Hotz, T. Weinkauf, and H.-P. Seidel. Combinatorial gradient fields for 2d images with empirically convergent separatrices. *Arxiv*, 1(1208.6523), 2012.

[38] V. Robins, P. Wood, and A. Sheppard. Theory and algorithms for constructing discrete Morse complexes from grayscale digital images. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 33(8):1646–1658, 2011.

[39] P. Sablonniére. Gradient approximation on uniform meshes by finite differences and cubic spline interpolation. In E. R. Hancock, R. R. Martin, and M. A. Sabin, editors, *Mathematics of Surfaces XIII*, volume 5654 of *Lecture Notes in Computer Science*, pages 322–334. Springer Berlin Heidelberg, 2009.

[40] N. Shivashankar and V. Natarajan. Parallel computation of 3d Morse-Smale complexes. *Computer Graphics Forum*, 31(3pt1):965–974, 2012.

[41] N. Shivashankar, M. Senthilnathan, and V. Natarajan. Parallel computation of 2d Morse-Smale complexes. *IEEE Transactions on Visualization and Computer Graphics*, 18(10):1757–1770, 2012.

[42] S. Smale. Generalized Poincaré's conjecture in dimensions greater than four. *Ann. of Math.*, 74:391–406, 1961.

[43] S. Smale. On gradient dynamical systems. *Ann. of Math.*, 74:199–206, 1961.

[44] P. Soille. Optimal removal of spurious pits in digital elevation models. *Water Resources Research*, 40, 2004.

[45] T. Sousbie. The persistent cosmic web and its filamentary structure - I. theory and implementation. *Monthly Notices of the Royal Astronomical Society*, 414(1):350–383, 6 2011.

[46] T. Sousbie, S. Colombi, and C. Pichon. The fully connected n-dimensional skeleton: probing the evolution of the cosmic web. *Monthly Notices of the Royal Astronomical Society*, 393(2):457–477, 2009.

[47] F. Sun, K. C. Cheng, and Q. Du. Modeling and segmentation of nuclei based on morse-smale complex. Submitted, 2013.

[48] J. Tierny, J. D. II, L. G. Nonato, V. Pascucci, and C. T. Silva. Interactive quadrangulation with Reeb atlases and connectivity textures. *IEEE Trans. on Vis. and Comp. Graph.*, 18(10):1650–1663, 2012.

[49] J. Tierny and V. Pascucci. Generalized topological simplification of scalar fields on surfaces. *IEEE Trans. on Vis. and Comp. Graph. (Proc. of IEEE VIS)*, 2012.

[50] T. Weinkauf, Y. Gingold, and O. Sorkine. Topology-based smoothing of 2D scalar fields with C^1-continuity. *Comp. Graph. Forum (Proc. of EuroVis)*, 29:1221–1230, 2010.

[51] T. Weinkauf and D. Günther. Separatrix persistence: Extraction of salient edges on surfaces using topological methods. In *Proceedings of the Symposium on Geometry Processing*, SGP '09, pages 1519–1528, Aire-la-Ville, Switzerland, Switzerland, 2009. Eurographics Association.

[52] S. Wienert, D. Heim, K. Saeger, A. Stenzinger, M. Beil, P. Hufnagl, M. Dietel, C. Denkert, and F. Klauschen. Detection and segmentation of cell nuclei in virtual microscopy images: A minimum-model approach, 2012.