Building and Generalizing Morphological Representations for 2D and 3D Scalar Fields

Lidija Ćomić\textsuperscript{1}, Leila De Floriani\textsuperscript{2} and Federico Iuricich\textsuperscript{2}

\textsuperscript{1}Faculty of Engineering, Novi Sad, Serbia
\textsuperscript{2}Department of Computer Science, Genova, Italy

Abstract

Ascending and descending Morse complexes, defined by the critical points and integral lines of a scalar field \( f \) defined on a manifold domain \( D \), induce a subdivision of \( D \) into regions of uniform gradient flow, and thus provide a compact description of the morphology of \( f \) on \( D \). Here, we propose a dimension independent representation for the ascending and descending Morse complexes, and a data structure which assumes a discrete representation of the field as a simplicial mesh, that we call the incidence-based data structure. We present algorithms for building such data structure for 2D and 3D scalar fields, which make use of a watershed approach to compute the cells of the Morse decompositions. We describe generalization operators for Morse complexes in arbitrary dimensions, we discuss their effect and present results of our implementation of their 2D and 3D instances both on the Morse complexes and on the incidence-based data structure.

Categories and Subject Descriptors (according to ACM CCS): I.3.3 [Computer Graphics]: Computational Geometry and Object Modeling—Object Representations

Keywords: Morphology, Terrain Models, Scalar Fields, Morse Complexes

1. Introduction

Representing morphological information extracted from discrete scalar fields is a relevant issue in several application domains, such as terrain modeling, volume data analysis and visualization, and time-varying 3D scalar fields. Morse theory offers a natural and intuitive way of analyzing the structure of a scalar field as well as of compactly representing the scalar field through a decomposition of its domain \( D \) into meaningful regions associated with the critical points of the field. The ascending and the descending Morse complexes are defined by considering the integral lines emanating from, or converging to the critical points of \( f \), while the Morse-Smale complex describes the subdivision of \( D \) into parts characterized by a uniform flow of the gradient between two critical points of \( f \). Computation of an approximation of the Morse and Morse-Smale complexes has been extensively studied in the literature in the 2D case, and recently algorithms have been proposed in 3D. The discrete watershed transform is one of the most popular methods used in image segmentation for 2D and 3D images and has been applied to regular Digital Elevation Models (DEMs). Here, we extend the watershed approach by simulated immersion [VS91] to compute the ascending and descending Morse complexes for simplicial meshes, focusing on 2D triangle meshes (forming Triangulated Irregular Networks (TINs)) and 3D tetrahedral meshes, discretizing the domain of a 3D scalar field. The approach, however, can be extended to higher dimensions in a straightforward way and our implementation is already dimension independent.

We represent the ascending and descending Morse complexes in arbitrary dimensions as a graph, called incidence graph, in which the nodes represent the cells of the Morse complexes in a dual fashion and the arcs their mutual incidence relations. We show how, in the discrete case, incidence graph can be effectively combined with a representation of the simplicial decomposition of the underlying domain \( D \). This representation, that we call an incidence-based representation of the Morse complexes, is based on encoding the incidence relations of the cells of the two complexes, and exploits the duality between the ascending and descending

submitted to Eurographics Italian Chapter Conference (2010)
2. Morse Theory and Morse Complexes

Morse theory studies the relationship between the topology of a manifold $M$ and the critical points of a scalar (real-valued) function defined on the manifold (for more details on Morse theory, see [Mat02, Mil63]). Recall that a closed $n$-manifold is a topological space in which every point has a neighborhood homeomorphic to the space $\mathbb{R}^n$. Let $f$ be a $C^2$ real-valued function defined over a closed compact $n$-manifold $M$. A point $p$ is a critical point of $f$ if and only if the gradient $\nabla f = \left( \frac{\partial f}{\partial x_1}, \ldots, \frac{\partial f}{\partial x_n} \right)$ (in some local coordinate system around $p$) of $f$ vanishes at $p$. Function $f$ is a Morse function if all its critical points are non-degenerate (i.e., the Hessian matrix $\text{Hess}_pf$ of the second derivatives of $f$ at $p$ is non-singular). The number $i$ of negative eigenvalues of $\text{Hess}_pf$ is called the index of critical point $p$, and $p$ is called an i-saddle. A 0-saddle, or an n-saddle, is also called a minimum, or a maximum, respectively. An integral line of $f$ is a maximal path which is everywhere tangent to the gradient of $f$. Each integral line originates at a critical point of $f$, called its origin, and converges to a critical point of $f$, called its destination.

Integral lines that converge to (originate at) a critical point $p$ of index $i$ form an $i$-cell, which is called a descending (ascending) cell, or manifold, of $p$. The descending and ascending cells decompose $M$ into descending (stable) and ascending (unstable) Morse complexes, denoted as $\Gamma_d$ and $\Gamma_u$, respectively, see Figure 1 (a) and (b) for a 2D example. We will denote as $p$ the descending $i$-cell of an $i$-saddle $p$. Morse function $f$ is called a Morse-Smale function if the descending and the ascending manifolds intersect transversally. If $f$ is a Morse-Smale function, then complexes $\Gamma_u$ and $\Gamma_d$ are dual to each other.

3. Related Work

In this Section, we review related work on morphological representation of scalar fields based on Morse or Morse-Smale complexes. We concentrate on the computation and the simplification of Morse and Morse-Smale complexes.

Several algorithms have been proposed in the literature for decomposing the domain of a 2D scalar field $f$ into an approximation of a Morse, or a Morse-Smale, complex. Recently, some algorithms in higher dimensions have been proposed. For a review of the work in this area, see [BFF*08].

The extraction of critical points of a scalar field $f$ defined on a simplicial mesh has been investigated in 2D [Ban70, NGH04], and in 3D [GP00, WSHH02, WSH03, TTF04, EHN03], as a basis for computing Morse and Morse-Smale complexes. Algorithms for decomposing the domain $D$ of $f$ into an approximation of a Morse, or of a Morse-Smale complex in 2D can be classified as boundary-
based [BS98, BEHP04, EHZ01, Pas04, TIKU95], or region-based [CCL03, DDM03, MDD'07]. In [EHNP03], an algorithm for extracting the Morse-Smale complex from a tetrahedral mesh is proposed. The algorithm, while interesting from a theoretical point of view, exhibits a large computational overhead, as discussed in [GNPH07].

Discrete methods rooted in the discrete Morse theory proposed by Forman [For98] are computationally more efficient. In [DDM03], a dimension-independent approach based on region growing has been proposed which implements the discrete gradient approach and computes the descending and the ascending Morse complexes. In [MM09], this algorithm has been specialized and implemented in 3D to compute segmentations of a 3D scalar field based on an extension of the discrete approximation to Gaussian curvature to tetrahedralized shapes. In [GNPH07], a region growing method, inspired by the watershed approach, has been proposed to compute the Morse-Smale complex. In [GBHP08], a Forman gradient vector field \( V \) is defined, and an approximation of the Morse-Smale complex is computed by tracing the integral lines defined by \( V \).

One of the major issues that arise when computing a representation of a scalar field as a Morse or as a Morse-Smale complex is the over-segmentation due to the presence of noise in the data sets. Simplification algorithms have been developed in order to eliminate less significant features from the Morse-Smale complex. Simplification is achieved by applying an operator called cancellation, defined in Morse theory [Mat02]. It cancels pairs of critical points of \( f \), in the order usually determined by the notion of persistence (absolute difference in function values between the paired critical points) [EHZ01]. In 2D Morse-Smale complexes, cancellation operator has been investigated in [BEHP04, EHZ01, TIKU95, Wol04]. Cancellation operator on Morse-Smale complexes of a 3D scalar field has been investigated in [GNPH07]. Unfortunately, the application of such operators to 1-saddles and 2-saddles increases the number of cells in the Morse-Smale complex.

4. Computing Morse Complexes through a Watershed Approach

In this Section, we recall the definition of the watershed transform. We review in greater detail the watershed algorithm by simulated immersion, introduced in [VS91]. We describe how we have extended this algorithm from images to simplicial meshes.

The watershed transform has been introduced for segmentation of gray-scale images. Several definitions exist in the discrete case [Mey94, VS91]. The watershed transform has also been defined for \( C^2 \)-differentiable functions over a connected domain \( D \) for which the critical points are isolated, and, thus, for Morse functions. Catchment basins and watershed lines are basic notions in the watershed transform.

They can both be defined in terms of topographic distance [Mey94, RM00]. The topographic distance is defined in such a way that it ensures that the path which minimizes the topographic distance between two points \( p \) and \( q \) in \( D \) is the path of steepest slope, if it exists. In other words, if \( p \) and \( q \) are two points in \( D \) and if there is an integral line which reaches both \( p \) and \( q \), then the topographic distance between these two points is equal to the difference in elevation between them. Otherwise, if such an integral line does not exist, the topographic distance between \( p \) and \( q \) is strictly greater than the difference in elevation between \( p \) and \( q \). The catchment basin \( CB(m_i) \) of a minimum \( m_i \) is defined as the set of points which are closer (in the sense of topographic distance) to \( m_i \) than to any other minimum. The watershed \( WS(f) \) of \( f \) is defined as the set of points in \( D \) which do not belong to any catchment basin, i.e., as the complement in \( D \) of the set of catchment basins of the minima of \( f \). When \( f \) is a \( C^2 \)-differentiable Morse function, then the catchment basins of the minima of \( f \) are the closure of the 2-cells in the ascending Morse complex of \( f \), and the set of watershed lines forms a subset of ridge lines, which connect saddles to maxima. Each catchment basin is bounded by a sequence of saddles, ridge lines and maxima.

In the discrete case, all watershed algorithms start by first extracting the minima of the discrete field \( f \) discretized on a regular grid and then they assign the points of the grid to catchment basins related to the minima. Points that are not assigned to any catchment basin belong to watershed lines. A similar procedure can be applied to the same image elevation function \( -f \), starting from the maxima. By computing the overlay of the two segmentations an approximation of the Morse-Smale complex is obtained.

Basically two techniques have been developed for computing the watershed transform in the discrete case starting from a 2D image (regular grid), namely watershed methods based on simulated immersion [VS91] and on discrete topographic distance [Mey94]. The method in [Mey94] extends the idea of topographic distance from the continuous to the discrete case, while the concept of simulated immersion is defined only in the discrete case [VS91]. In our work, we have extended all such approaches to triangle meshes rep-
representing Triangulated Irregular Networks (TINs) and we have compared them in [Vit10]. In [Vit10], we have also compared watershed approaches for TINs with both region-based and boundary-based approaches, and we have seen that the watershed approach by simulated immersion gives very good results.

4.1. Watershed by Simulated Immersion

The idea of simulated immersion can be described in an intuitive way. Let us consider a terrain and assume to drill holes in place of local minima. We assume to insert this surface in a pool of water, building dams to prevent water coming from different minima to merge. Then, the watershed of the terrain is described by these dams, and the catchment basins of minima are delineated by the dams. The method, that belongs to this class, uses the concept of skeleton by influence zones [VS91] in order to define catchment basins and watershed lines. Figure 2 (a) shows the catchment basins of two minima and the related watershed lines.

To understand the concept of skeleton by influence zones, we can imagine a set A, and a set \( B \subseteq A \) composed of \( n \) connected components \( B_1, \ldots, B_n \). The skeleton by influence zones is the set \( C \) of points in \( A \) which are equally close (in the sense of geodesic distance) to at least two connected components of \( B \). We recall that the geodesic distance between two points \( p \) and \( q \) in \( A \) is the length of a minimal path which connects \( p \) to \( q \) and stays within \( A \). The influence zone of a component \( B_i \in B \) is the set of points in \( A \) which are closer to \( B_i \) than to any other connected component \( B_j \) of \( B \). Note that the skeleton by influence zones \( C \) of \( B \) within \( A \) is the complement of the union of influence zones of \( B_i \) within \( A \) (see Figure 2 (b) and (c)).

The method in [VS91] recursively extracts catchment basins and watershed lines, starting from the minimal value of the elevation function \( f \) and going up. At each level of recursion, new minima are found, or already created catchment basins are expanded. The expansion process continues until, at a given level \( h \), a potential catchment basin \( CB_h \) (related to level \( h \)) contains at least two catchment basins (for example \( CB_{h-1}, CB_{j-1} \)) already present at level \( h - 1 \).

This is the case in which the definition of skeleton by influence zones comes up: \( CB_h \) is partitioned into three elements, the two influence zones of \( CB_{h-1} \) and \( CB_{j-1} \) and the set of points in \( CB_h \) equally distant from \( CB_{h-1} \) and \( CB_{j-1} \) (skeleton by influence zones). The influence zones of \( CB_{h-1} \) and \( CB_{j-1} \) will be part of the final set of catchment basins in the output of the algorithm. The process stops when the maximal level is reached. The watershed is defined as the complement of the set of catchment basins.

We have extended the watershed-by-simulated-immersion algorithm to simplicial meshes in arbitrary dimensions. The vertices of the simplicial mesh \( \Sigma \) are sorted in increasing order with respect to the values of the scalar field \( f \). In the second phase, the vertices of \( \Sigma \) are processed level by level in increasing order of elevation values. For each minimum, a catchment basin is formed iteratively through a breadth-first traversal of the graph which forms the 1-skeleton of the simplicial mesh \( \Sigma \). We first label each vertex at level \( h \) with a neutral label. Then, for each vertex \( p \), we examine its adjacent vertices in the mesh and, if they all belong to the same catchment basin \( \beta_m \), or some of them are watershed points, then we mark \( p \) as belonging to \( \beta_m \). If they belong to two or more catchment basins, then \( p \) is marked as a watershed point. Vertices that are not connected to any previously processed vertex are new minima and get a new label corresponding to a new catchment basin.

Finally, each maximal simplex (a \( d \)-simplex if we consider a \( d \)-dimensional simplicial mesh) is assigned to a basin based on the labels of its vertices. If all vertices of a \( d \)-simplex \( \sigma \), that are not watershed points, belong to the same basin \( \beta_m \), then we assign \( \sigma \) to \( \beta_m \), otherwise if the vertices belong to different basins \( \sigma \) is assigned to the nearest one.

Figure 3 illustrates segmentations obtained from a synthetic terrain built by sampling a function which is a combination of two planes and 64 gaussian surfaces and from real data in the 2D case. Figure 4 illustrates the results in the 3D case. We have also compared this approach in 3D with the region-based algorithm in [DDM03, MM09] using different metrics (extended from the ones used from TINs in [Vit10]) and we have obtained more promising results with the watershed approach described here.
5. A Dual Incidence-Based Representation for Morse Complexes

In this Section, we discuss a dual representation for the ascending and the descending Morse complexes \( \Gamma_u \) and \( \Gamma_d \), that we call the incidence-based representation. The underlying idea is that we can represent both the ascending and the descending complex as a graph by considering the boundary and co-boundary relations of the cells in the two complexes. In the discrete case, we consider a representation for the simplicial mesh which generalizes an indexed data structure commonly used for triangle and tetrahedral meshes, and we relate the two representations into the incidence-based data structure.

We have developed a data structure for manifold simplicial meshes which encodes the 0-simplices (vertices) and \( d \)-simplices explicitly plus the following relations. For every \( n \)-simplex \( \sigma \), it encodes the \( n + 1 \) vertices of \( \sigma \) and the \( n + 1 \) \( n \)-simplices which share an \((n - 1)\)-simplex with \( \sigma \). For every 0-simplex, we also encode one \( n \)-simplex incident in it. We store all 0-simplices (vertices) of \( \Sigma \) in a list of size \( |\Sigma_0| \), where \( |\Sigma_0| \) is the number of 0-simplices of \( \Sigma \). We store the \( n \)-simplices of \( \Sigma \) in a list of size \( |\Sigma_d| \), where \( |\Sigma_d| \) is the number of \( n \)-simplices of \( \Sigma \). Note that each vertex \( v \) of an \( n \)-simplex \( \sigma \) defines a unique \((n - 1)\)-face \( \gamma \) of \( \sigma \) which does not contain \( v \).

Recall that there is a one-to-one correspondence between \( i \)-saddles \( p \) and \( i \)-cells \( p \) in the descending complex \( \Gamma_d \), and dual \((n - i)\)-cells in the ascending complexes \( \Gamma_u \), \( 0 \leq i \leq n \). We exploit this duality to define a representation which encodes both the ascending and the descending complexes at the same time, as an incidence graph [Ede87]. The incidence graph encodes the cells of a complex as nodes, and a subset of the boundary and co-boundary relations between cells as arcs. The incidence graph associated with an \( n \)-dimensional descending Morse complex \( \Gamma_d \) (and with an ascending Morse complex \( \Gamma_u \)) is a graph \( G = (N, A) \), in which

1. the set of nodes \( N \) is partitioned into \( n + 1 \) subsets \( N_0, N_1, \ldots, N_a \), such that there is a one-to-one correspondence between nodes in \( N_i \) (which we will call \( i \)-nodes) and the \( i \)-cells of \( \Gamma_d \) (and thus the \((n - i)\)-cells of \( \Gamma_u \)),
2. there is an arc joining an \( i \)-node \( p \) with an \((i + 1)\)-node \( q \) if and only if the corresponding cells \( p \) and \( q \) differ in dimension by one, and \( p \) is on the boundary of \( q \) in \( \Gamma_d \) (\( q \) is on the boundary of \( p \) in \( \Gamma_u \)),
3. each arc connecting an \( i \)-node \( p \) to an \((i + 1)\)-node \( q \) is labeled by the number of times \( i \)-cell \( p \) (corresponding to \( i \)-node \( p \) in \( \Gamma_d \)) is incident to \((i + 1)\)-cell \( q \) (corresponding to \((i + 1)\)-node \( q \) in \( \Gamma_u \)).

Attributes are attached to the nodes of the incidence graph, containing information about geometry, and function values, while arcs have no associated (geometric) attributes. The incidence graph provides also a combinatorial representation of the 1-skeleton of a Morse-Smale complex. Figure 1 (c) shows a portion of the incidence graph encoding the connectivity of the descending Morse complex in Figure 1 (a), and of the ascending Morse complex in Figure 1 (b).

We have designed and implemented a data structure based on the incidence graph by associating with the nodes representing minima the list of the \( n \)-simplices forming the descending cells associated with the minima and with the nodes representing the maxima the list of the \( n \)-simplices forming the descending cells associated with the maxima. We call this data structure an incidence-based representation.

In the incidence-based representation, the incidence graph \( G = (N, A) \) is encoded as three arrays of nodes (one for minima, one for maxima and one for saddles) plus an array of arcs. Each element of the array of the nodes encoding the minima encodes a minimum \( p \) and contains; the coordinates of \( p \), the list of the \( n \)-simplices forming the corresponding ascending \( n \)-cell plus a list of pointers to the arcs incident in \( p \). The array of the nodes corresponding to the maxima has exactly the same information as the array of the minima, but this time each element contains the list of the \( n \)-simplices forming the corresponding descending cell. Each element of the array of the saddles contains the lists of all saddles with the same index \( i \), and for each of them, the coordinates of the corresponding saddle and the lists of the arcs incident in it. More precisely, for a saddle \( s \) of index \( i \), there are two lists of

\[ (a)(b) \] (c) (d)

Figure 4: Segmentations obtained by watershed algorithem extended to the 3D case. (a) Ascending and (b) descending Morse complexes obtained from the synthetic function \( v = \sin(x) + \sin(y) + \sin(z) \) and (c) ascending and (d) descending Morse complexes obtained from real data.
arcs, those joining $s$ to nodes of index $i + 1$ and those joining $s$ to nodes of index $i - 1$.

Arcs are also stored in an array of lists. The $j$-th element of the array contains a list of arcs connecting nodes corresponding to saddles of index $j$ to nodes corresponding to saddles of index $j + 1$. Each element of any of such lists corresponds to an arc $a$ and contains the indexes of the two nodes in which $a$ is incident plus an integer indicating how many times the nodes are incident to each other.

The resulting data structure is completely dimension independent. We have compared the 3D instance of the incidence-based representation with the data structure proposed in [GNP’06] for encoding 3D Morse-Smale complexes. This latter encodes the critical points (together with their geometric location) and, for each critical point $p$, the sets of all 3-simplexes and of all 2-simplexes, forming the (descending or ascending) 3-cell and 2-cell associated with $p$. Moreover, all 1-simplexes of $\Sigma$, which are the edges in the Morse-Smale complex, are maintained. In the 3D instance of the incidence-based representation, we encode only the 3-simplexes defining the 3D cells in the ascending and descending 3D cells associated with the minima and maxima, while the geometry of the edges in the Morse-Smale complex needs to be computed by the boundaries of such 3D cells. Thus, the incidence-based data structure is definitely more compact.

6. Building an Incidence-Based Representation

We have developed algorithms for constructing the incidence graph in 2D and 3D starting from the decomposition of the simplicial complex $\Sigma$ into regions associated with minima and maxima, and the information on the geometry of maxima and minima.

The input of the algorithms consists of the simplicial mesh $\Sigma$ over which a scalar function $f$ is defined, encoded in the data structure described in the previous section. The $n$-cells of the descending Morse complex $\Gamma_a^d$ and of the ascending Morse complex $\Gamma_a^u$ are expressed as collection of $n$-simplexes of $\Sigma$, and they are labeled by the indexes of vertices of $\Sigma$ which are minima and maxima of $f$.

In the preprocessing step, for each descending region in $\Gamma_a^d$, a maximum node is created and inserted in the array of maximum nodes, and the same is done for each ascending region in $\Gamma_a^u$, in which case a minimum node is created. The index of the corresponding vertex in $\Sigma$ is also stored for each extremum. Then, for each $n$-simplex $\sigma$ of $\Sigma$, we add the index of $\sigma$ to the maximum node which represents the region in $\Gamma_a^d$ containing $\sigma$ and to the minimum node which represents the region in $\Gamma_a^u$ containing $\sigma$.

The preprocessing step is common to both the 2D and 3D algorithms, while the other steps are dimension specific and are described in the following two subsections.

6.1. Construction of the Incidence Graph in 2D

In the 2D case, after the preprocessing step, we perform two steps: (i) creation of the nodes corresponding to saddles, and (ii) creation of the arcs of the incidence graph.

To create the saddle nodes, we need to generate the 1-cells either of the ascending or of the descending complex. Each 1-cell is a chain of edges of the triangle mesh. We work on the ascending complex $\Gamma_a^u$. We initialize a queue $Q$ of triangles with an arbitrary triangle, and we label all triangles as non-visited. We repeat the following process while $Q$ is not empty:

- extract the first triangle $t$ from $Q$;
- for each triangle $t_i$ adjacent to $t$,
  - insert $t_i$ in $Q$
  - if $t$ and $t_i$ have not been visited, let $m_1$ and $m_2$ be the nodes representing ascending regions containing $t$ and $t_i$, respectively
  - if $m_1$ is different from $m_2$, check if there is a node $s$ representing the saddle separating the ascending regions. If there is such a node, add edge $e$ common to triangles $t$ and $t_i$ to it. Otherwise, create a new node $s$ and add it to edge $e_i$, and a reference to adjacent nodes $m_1$ and $m_2$.
- if one of the triangles adjacent to $t$ is missing
  - consider node $m_1$ modeling the ascending region containing $t$;
  - create a node which will model the adjacency between $m_1$ and the node modeling the boundary
  - otherwise, if such a node exists, add to it a reference to the edge of $t$ which is on the boundary
- mark $t$ as visited

At this point, all the 1-cells which form the boundaries between 2-cells in $\Gamma_a^u$ have been found. We examine the endpoints of the 1-cells. A 1-cell with more than two end-points is subdivided into 1-cells, creating new saddle nodes corresponding to the new 1-cells. Each saddle node is connected to two nodes corresponding to maxima, and to the 2-cells separated by the corresponding 1-cell. If the 1-cell is on the boundary, and there is a minimum on 1-cell, that minimum is cancelled in order to maintain the duality in the incidence graph.

Next, we create the arcs between saddle nodes and nodes corresponding to maxima

- if a 1-cell has two different end-points corresponding to maxima, the corresponding nodes are connected in the incidence graph;
- if one of the end-points of a 1-cell belongs to the boundary, then an arc is created between a virtual maximum and the node corresponding to the other end-point of the 1-cell
- if the end-points of a 1-cell do not correspond to maxima, we check for each of the end-points if all triangles incident
in it belong to the same descending 2-cell. If this is the case, we create a maximum at the end-point. Otherwise, we delete the saddle node, since we regard it as an error of segmentation algorithm.

- If a 1-cell has no end-points, it circumscribes one of the 2-cells in $\Gamma_a$. In this case, we add a dummy maximum on the 1-cell, thus creating a loop, to maintain topological consistency.

If there is some maximum $p$ not connected to any saddle, then that maximum must be inside some 2-cell in $\Gamma_a$. In this case, a 1-saddle is created by looking at the 2-cells corresponding to $p$ and at its adjacent 2-cells in $\Gamma_d$.

6.2. Construction of the Incidence Graph in 3D

The construction of the incidence graph requires, after the preprocessing, other three steps, namely, (i) generation of the nodes corresponding to 1-saddles and 2-saddles, (ii) generation of the arcs between 1-saddles and minima and 2-saddles and maxima and (iii) generation of the arcs joining 1- and 2-saddles.

The first two steps directly generalize the 2D algorithm. Nodes corresponding to 1-saddles and 2-saddles are constructed in a similar way as we construct saddle nodes in the 2D case. 1-saddles are generated by considering the triangulated surfaces separating 3-cells in the ascending Morse complex (recall that 3-cells correspond to minima), while 2-saddles are generated by considering the triangulated surfaces separating 3-cells in the descending Morse complex (which correspond to maxima). This is simply a generalization of the algorithm we have seen before in the 2D case: the difference is that here we consider tetrahedra instead of triangles, and that we look for triangles separating 3-cells of the Morse complexes instead of edges.

Again, the algorithm for connecting 1-saddle nodes to minimum nodes and 2-saddle nodes to maximum nodes is a simple extension of the algorithm for computing the arcs between saddle and extrema in the 2D case.

The third step consists of generating the arcs connecting the nodes corresponding to 1-saddles to those corresponding to 2-saddles. We work first on the ascending complex $\Gamma_a$. For each 2-cell $s_1$ in $\Gamma_a$ (which corresponds to a 1-saddle), we consider the set $M_1$ of maxima connected to $s_1$, which correspond to the vertices of 2-cell $s_1$. Then, if there is more than one maximum in $M_1$, and the 2-cell $s_1$ is not on the boundary of the domain, we check, for each pair of maxima $m_1$ and $m_2$ in $M_1$, if there exists in the descending complex $\Gamma_d$ a 2-cell $s_2$ (i.e., a 2-saddle) between the 3-cells corresponding to $m_1$ and $m_2$. If $s_2$ exists, then we connect in the incidence graph the two nodes corresponding to 1-saddle $s_1$ and 2-saddle $s_2$.

Otherwise, if the 2-cell $s_1$ is on the boundary, we consider the minimum $p$ associated with the only 3-cell bounded by $s_1$ and the set of edges in the tetrahedral mesh $\Sigma$ incident into vertex $p$. For each of such edges $e$, we consider the set of tetrahedra $T$ incident in $e$, and for each tetrahedron $t$ in $T$, we find the 3-cell in the descending complex $\Gamma_d$ containing $t$. We select only the edges $e'$ incident in $p$ such that $T'$ contains tetrahedra belonging to different 3-cells. We denote the set of such 3-cells $C'$. We replace node $s_1$ in the graph with nodes $q$ corresponding to the selected edges. Nodes $q$ are connected to the same minimum node to which $s_1$ was connected. For each 1-saddle node $q$, we consider all pairs of maxima corresponding to the 3-cells in $C'$, and, for each of such pairs $m_1$ and $m_2$, we connect node $q$ in the graph with the node corresponding to the 2-saddle $s'$ connected to both nodes $m_1$ and $m_2$.

At this point, there may remain some maxima on the boundary of an ascending 2-cell which are not detected by the previous procedure. Thus, we scan the 2-cells in the descending complex corresponding to 2-saddles, and for each 2-cell corresponding to a 2-saddle not connected to any 1-saddle we repeat the procedure used for boundary 2-cells corresponding to 1-saddles. In this way we can process the boundary without introducing artificial critical points (like virtual extrema) and we maintain the duality of the complex and ensure the correctness of the incidence graphs as their common representation.

7. Simplification Operators

In 2D, simplification operators merge an extremum and a saddle into another extremum. Specifically, a removal merges a saddle $q$ and a maximum $p$ connected to it through an integral line into a unique other maximum $p'$ different from $p$ and connected to $q$. Dually, a contraction in 2D merges a saddle $q$ and a minimum $p$ connected to $q$ into a unique other minimum $p'$ different from $p$ and connected to $q$. A removal merges a (2D) descending cell of maximum $p$ into the descending cell of maximum $p'$ by removing the (1D) descending cell of saddle $q$. It also merges a (0D) ascending cell of maximum $p$ into ascending cell of maximum $p'$ by contracting a (1D) ascending cell of saddle $q$. Thus, it corresponds to a region merging on the descending complex and to an edge collapse on the ascending complex. Dually, a contraction merges a (2D) ascending cell of minimum $p$ into ascending cell of minimum $p'$ by removing the (1D) ascending cell of saddle $q$, and it merges a (0D) descending cell of minimum $p$ into descending cell of minimum $p'$ by contracting a (1D) descending cell of saddle $q$. The two operators in 2D have been well studied in GIS, but generally they are considered on the surface network, i.e., the 1-skeleton of the Morse-Smale complex, and thus they correspond to edge collapses on such network.

We have generalized the removal and contraction operators to arbitrary dimensions [CD09], and we have shown that they behave similarly as the operators in 2D. It can be shown that such operators form a minimal basis of operators for simplifying a Morse function in arbitrary dimensions.
Unlike the operators for the 3D case defined in \[\text{GNPH07}\], such operators never increase the number of cells in the complexes. Also the operators in \[\text{GNPH07}\] can be expressed as macro-operators in terms of our operators.

We review here their dimension-independent definition. The first operator, that we call a removal of index \(i\), \(1 \leq i \leq n - 1\), removes an \(i\)-saddle \(q\) and an \((i + 1)\)-saddle \(p\) provided that \(q\) is connected by a unique integral line either to (i) an \((i + 1)\)-saddle \(p\), and exactly one other \((i + 1)\)-saddle \(p'\) different from \(p\), or to (ii) exactly one \((i + 1)\)-saddle \(p\). In the first case, a removal of \(q\) and \(p\) is denoted as \(\text{rem}(p,q,p')\), and in the second case as \(\text{rem}(p,q,\emptyset)\). The second operator, that we call a contraction of index \(i\), \(1 \leq i \leq n - 1\), removes an \(i\)-saddle \(q\) and an \((i - 1)\)-saddle \(p\) provided that \(q\) is connected by a unique integral line to (i) an \((i - 1)\)-saddle \(p\), and exactly one other \((i - 1)\)-saddle \(p'\) different from \(p\), or to exactly one \((i - 1)\)-saddle \(p\). In the first case, a contraction of \(q\) and \(p\) is denoted as \(\text{con}(p,q,p')\), and in the second case as \(\text{con}(p,q,\emptyset)\). For the sake of simplicity, we consider here only removals and contractions of the first kind.

### 7.1. Simplification on the Morse complexes

The removal and contraction operators have a dual effect on the descending and the ascending Morse complexes. The effect of a contraction of index \(i\) on \(\Gamma_d\) (\(\Gamma_a\)) is the same as the effect of a removal of index \(n - i\) on \(\Gamma_a\) (\(\Gamma_d\)). For the sake of brevity, we describe the effect of the two operators on descending Morse complexes only. The effect of a removal \(\text{rem}(p,q,p')\) on the descending Morse complex \(\Gamma_i\) is as follows: \(i\)-cell \(q\) is deleted and \((i + 1)\)-cell \(p\) is merged into \((i + 1)\)-cell \(p'\). A contraction \(\text{con}(p,q,p')\) on the descending Morse complex \(\Gamma_d\) deletes \(i\)-cell \(q\) and merges \((i - 1)\)-cell \(p\) into \((i - 1)\)-cell \(p'\). \(i\)-cell \(q\) is contracted, and each \(i\)-cell in the co-boundary of \(p\) is extended to include a copy of \(i\)-cell \(q\), i.e., each \(i\)-cell in the co-boundary of \(p\) is, after contraction, the union of itself with \(i\)-cell \(q\).

In 2D, there are exactly one removal and exactly one contraction operator (both of index 1). A removal of index 1 removes a 1-cell (saddle) \(q\), and merges the two 2-cells (maxima) which shared \(q\). A contraction of index 1 contracts a 1-cell (saddle) \(q\) and collapses the two 0-cells (minima) which bounded \(q\). Both operators involve an extremum and a saddle.

In 3D, there are two removal and two contraction operators. A removal of index 2 involves a 2-saddle \(q\) and a maximum \(p\). In the descending complex, it removes a 2-cell \(q\), and merges 3-cell \(p\) into a unique 3-cell \(p'\) incident in \(q\) and different from \(p\), as illustrated in Figure 5 (a). A removal of index 1 does not involve an extremum, but it involves a 1-saddle \(q\) and a 2-saddle \(p\). It is defined only if 1-cell \(q\) is incident to exactly two different 2-cells \(p\) and \(p'\). It removes 1-cell \(q\) and merges 2-cell \(p\) into 2-cell \(p'\), as illustrated in Figure 5 (b). An example of the effect of a contraction of index 1 of a minimum \(p\) and 1-saddle \(q\) on a 3D descending Morse complexes is illustrated in Figure 6 (a). 1-cell \(q\) is contracted, and 0-cell \(p\) is collapsed into 0-cell \(p'\). A contraction of index 2 involves a 1-saddle \(p\) and 2-saddle \(q\). It is defined only if 2-cell \(q\) is bounded by exactly two different 1-cells \(p\) and \(p'\). It contracts 2-cell \(q\), and collapses 1-cell \(p\) into 1-cell \(p'\), as illustrated in Figure 6 (b).

\[\text{rem}(p,q,p')\text{ of index 1 (a), and of index 2 (b).}\]

### 7.2. Simplification on the Incidence Graph

We describe the effect of the simplification operators on the incidence-based representation of the dual Morse complexes. For brevity, we will consider only a removal \(\text{rem}(p,q,p')\) of index \(i\), \(1 \leq i \leq n - 1\).

Let \(G = (N,A)\) be the incidence graph representing both the descending and the ascending Morse complexes \(\Gamma_d\) and \(\Gamma_a\) before a removal \(\text{rem}(p,q,p')\). Then we have that

- \(i\)-node \(q\) is connected through an arc in \(A\) to exactly two different \((i + 1)\)-nodes \(p\) and \(p'\), such that the label of arcs \((q,p)\) and \((q,p')\) is 1, and to an arbitrary number of \((i - 1)\)-nodes from a set \(Z = \{z_h, h = 1, \ldots, h_{\max}\}\);
- node \(p\) is connected to an arbitrary number of \(i\)-nodes from a set \(R = \{r_j, j = 1, \ldots, r_{\max}: r_j \neq q\}\), and to an arbitrary number of \((i + 2)\)-nodes from a set \(S = \{s_k, k = 1, \ldots, s_{\max}\}\);
- node \(p'\) is connected to an arbitrary number of \(i\)-nodes from a set \(C = \{c_l, l = 1, \ldots, c_{\max}: c_l \neq q\}\), and to an arbitrary number of \((i + 2)\)-nodes from a set \(D = \{d_{m}, m = 1, \ldots, d_{\max}\}\).

The conditions above translate the feasibility conditions
of a removal operator on the Morse complexes. For example, before the removal $\text{rem}(p,q,p')$, illustrated in Figure 7 (b), 1-node $q$ is connected to exactly two different 2-nodes $p$ and $p'$ (corresponding to 2-saddles), and to two 0-nodes in $Z = \{z_1,z_2\}$ (corresponding to minima), which are not shown in the Figure. 2-node $p$ is connected to 1-nodes in $R = \{r_1,r_2,r_3\}$ and 2-node $p'$ is connected to 1-nodes in $C = \{c_1,c_2,c_3\}$. Nodes $p$ and $p'$ are connected to exactly the same 3-nodes in $S = D = \{s_1,s_2\}$, which are not shown in the Figure.

As an effect of a removal $\text{rem}(p,q,p')$ on $G$, nodes $p$ and $q$ are deleted, as well as arc $(p,q)$ joining $p$ and $q$, and all the arcs incident to $q$. All the arcs incident to $p$, with the exception of arc $(p,q)$, become incident in $p'$. Note that the effect of a contraction on $G$ is exactly the same as the effect of a removal, except for the fact that in a removal $q$ is an i-node, and $p$ and $p'$ are $(i+1)$-nodes, while in a contraction $q$ is an i-node and $p$ and $p'$ are $(i-1)$-nodes.

In the example in Figure 7, after the removal of 1-node $q$ and 2-node $p$, nodes $q$ and $p$ are deleted from the incidence graph ($N' = N \setminus \{q,p\}$), arcs connecting $q$ to $p$ and $p'$, and arcs connecting $q$ to 0-nodes in $Z = \{z_1,z_2\}$ (not illustrated in the Figure) are deleted, as are arcs connecting $p$ to 3-nodes in $S = \{s_1,s_2\}$ (not illustrated in the Figure). Arcs connecting $p$ to 1-nodes in $R = \{r_1,r_2,r_3\}$ are replaced by arcs connecting $p'$ to 1-nodes in $R$.

The effect on the incidence-based representation, that is on the combination of the incidence graph with the underlying simplicial decomposition of the domain, is restricted to the incidence graph when a simplification does not involve an extremum. When we perform a removal $\text{rem}(p,q,p')$ of index $n-1$, then the partition of the $n$-simplices of the underlying mesh into descending cells of maxima is updated by merging the set of $n$-simplices forming the descending cell of $p$ into set of $n$-simplices forming the descending cell of $p'$. Dually, a contraction $\text{con}(p,q,p')$ of index 1 merges $n$-simplices of the ascending cell of $p$ with $n$-simplices of the ascending cell of $p'$.

We have implemented the simplification operators on the incidence-based data structure in a completely dimension-independent way, since the incidence graph is dimension independent. We have experimented with Morse complexes for 2D and 3D scalar fields. Simplifications are performed guided by the persistence criterion. Recall that persistence is defined as the absolute difference in function values between the critical points which are collapsed [EHZ01].

An example of the application of simplification operators to 2D and 3D Morse complexes is shown in Figure 8.

8. Concluding Remarks

We have presented a compact and dimension independent representation, the incidence-based data structure, for both the ascending and descending Morse complexes of a scalar field $f$ based on exploiting the duality of the two complexes. We have proposed an algorithm for computing the incidence graph for Morse complexes of 2D and 3D scalar fields based on a watershed algorithm that we have developed for constructing the maximal cells of the descending and ascending complexes. We have defined dimension independent simplification operators on the Morse complexes and on the incidence graph and we have shown results of our implementation on 2D and 3D scalar fields.

The objective of our research is developing a software tool for the morphological analysis of a 3D scalar field at different levels of abstraction based on the multi-resolution Morse complexes. We have so-far developed the tool up to
the simplification operators. We have designed a compact data structure for encoding the multi-resolution Morse complexes, not reported here for brevity, and we are going to implement such data structures, the refinement operators, and the selective refinement queries.

9. Acknowledgments

This work has been partially supported by the MIUR-FIRB project SHALOM under contract number RBIN04HWR8, and by the National Science Foundation under grant CCF-0541032.

References


Lidija Čomić & Leila De Floriani & Federico Iuricich / Building and Generalizing Morphological Representations for 2D and 3D Scalar Fields
