

EXTRACTING TERRAIN MORPHOLOGY

A New Algorithm and a Comparative Evaluation

Paola Magillo, Emanuele Danovaro, Leila De Floriani, Laura Papaleo and Maria Vitali
Department of Information and Computer Science, University of Genova, Via Dodecaneso, 35, 16146 Genova, Italy

Keywords: Terrain models, Triangulated Irregular Networks, morphological structure.

Abstract: We consider the problem of extracting morphology of a terrain represented as a Triangulated Irregular Network (TIN). We propose a new algorithm and compare it with representative algorithms of the main approaches existing in the literature to this problem.

1 INTRODUCTION

Extracting and representing morphological information is a very relevant issue in order to develop automatic tools for gaining and maintaining knowledge of terrain models which are widely used in GIS applications, Virtual Reality and so on.

A terrain model is a scalar field, i.e., a function $f(x,y)$ (usually called height function) defined on a domain D . Often, f is known only at a finite set of sampled points and it is approximated through a discrete digital model: a *Regular Square Grid* (RSG) if the sampled points are regularly spaced, and a *Triangulated Irregular Network* (TIN) if they are irregularly sampled. Both RSGs and TINs provide accurate representations terrains, but they fail in capturing the morphological structure defined by critical points (pits, peaks, passes), and integral lines, like (ridges, valleys). On the contrary, a morphological terrain description is compact and supports a knowledge-based approach to analyze, visualize and understand a terrain dataset, as required, for instance, in visual data mining applications.

In the last decades, there has been a lot of research focusing on extracting critical features (points, lines or regions) from images or terrain data described by an RSG, or a TIN. More recent works in computational geometry concentrate on representing the morphology of terrains through a decomposition of the terrain surface into regions bounded by critical points (minima, maxima, saddle points) and integral lines.

These techniques are rooted in Morse theory and try to simulate the decomposition of a terrain induced by C^2 Morse functions in the discrete case.

In this paper, we propose a new algorithm for extracting morphological information (in the form of the stable and unstable Morse complexes) from a terrain model described by a TIN, which is simple, requires no floating point calculations, and can manage special configurations such as flat triangles and edges. We also present a comprehensive study of analogous existing methods and propose a set of experiments in order to evaluate our approach.

Recall that a TIN basically consists of a triangulation Σ covering the field domain D of the height function f , having its vertices at the sampled points. In a *triangulation*, two nearby triangles can only touch each other by sharing a vertex, or a common edge. On each triangle t in Σ , function f is approximated as a linear interpolant of the height values sampled at the three vertices of t ¹.

In the remainder of this paper, Section 2 introduces some basic background notions; Section 3 discusses related works; Section 4 presents our novel algorithm; Section 5 introduces three representative algorithms that we have implemented for comparison, and Section 6 presents an experimental evaluation of our novel algorithm compared to these three methods. Finally, Section 7 draws some concluding remarks.

¹Note that RSGs can be reduced to TINs by triangulating each square into two triangles.

2 BACKGROUND

Morse theory is a powerful tool to capture the topological structure of a scalar field in the continuum (Smale 1960). Let f be a C^2 real-valued function defined over a domain $D \subseteq \mathbb{R}^2$. A point $p \in \mathbb{R}^2$ is called a *critical point* of f if and only if the *gradient* of f vanishes at p . The function f is a *Morse function* if and only if the Hessian matrix $H_p f$ of the second derivatives of f at a critical point p is non-singular (its determinant is $\neq 0$): basically, if all its critical points are non-degenerate. This implies that the critical points of a Morse function are *isolated*. The number of negative eigenvalues of $H_p f$ is called the *index* of a critical point p . In 2D, a non-degenerate critical point p of a Morse function f can be of three types: a *minimum* (*pit*), a *saddle*, or a *maximum* (*peak*), if p has index 0, 1 or 2, respectively. An *integral line* of a function f is a maximal path which is everywhere tangent to the gradient vector field. It is emanating from a critical point or from the boundary of D , and it reaches another critical point or the boundary of D . An integral line which connects a maximum to a saddle, or a minimum to a saddle, is called a *separatrix line*².

Integral lines that converge to a maximum, a saddle and a minimum form a 2-dimensional (region), 1-dimensional (line) and 0-dimensional (point) cell, respectively, and they are called *unstable manifolds*. Integral lines that originate from a minimum, a saddle and a maximum form a 2-, 1- and 0-dimensional cell, respectively, and they are called *stable manifolds*. The stable (unstable) manifolds are pair-wise disjoint cells and form a complex, since the boundary of every cell is the union of lower-dimensional cells. They are called *stable* and *unstable Morse complexes*, respectively. Figure 1(a) shows a decomposition of the domain of a scalar field into a stable Morse complex.

A Morse function f is a *Morse-Smale function* when the stable and the unstable manifolds intersect only transversally. In two dimensions, this means that the stable and unstable 1-manifolds (lines) cross when they intersect, and the crossing points are saddles.

A *Morse-Smale complex* is the complex defined by the intersection of the stable and unstable Morse complexes for a function f which is a *Morse-Smale function*. The 1-skeleton of a Morse-Smale complex consists of the critical points and the separatrix lines joining them, and it is called a *critical net* (see Figure

²In Geographic Information Systems (GISs), separatrix lines that connect minima to saddles are usually called *ravine*, or *valley lines*, while those that connect saddles to maxima are called *ridge lines*.

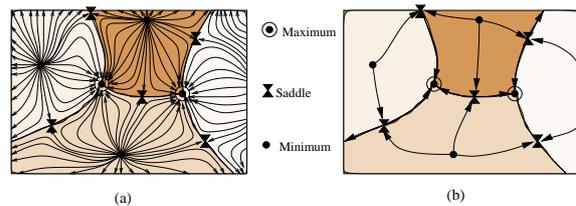


Figure 1: (a) An example of a stable Morse complex (the 2-manifolds correspond to the minima). (b) The Morse-Smale complex. Its 1-skeleton is the critical net.

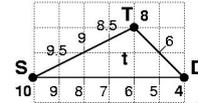


Figure 2: Edge labelled T-D is steeper than edge labelled S-D. Numbers denote vertex heights.

1 (b)).

The *surface network* (Pfaltz 1976; Schneider and Wood 2004) used in Geographic Information Systems (GISs) for morphological terrain modeling, is essentially the critical net.

3 RELATED WORK

Several algorithms have been proposed in the literature for decomposing the domain of a scalar field f (as a terrain model) into an approximation of a Morse complex (or of a Morse-Smale complex). They either fit a C^1 or C^2 surface on a terrain model, or simulate a Morse-Smale complex (a Morse complex) in the discrete case. By assuming that no two adjacent vertices in the TIN have the same height, they ensure that the critical points are isolated, as in the Case of C^2 Morse functions (Edelsbrunner et al. 2001).

A Morse (Morse-Smale) complex can also be defined using the concepts related to watershed transform (Meyer 1994; Vincent and Soille 1991; Roerdink and Meijster 2000; Mangan and Whitaker 1999; Stoev and Strasser 2000). The watershed transform in the C^2 case provides a decomposition of a the domain of a function f into open regions of influence associated to the minima, called *catchment basins*. Catchment basins can be described in terms of topographic distance (Meyer 1994). In the 2D case, if the function f is a Morse function, the catchment basins of the minima are essentially 2-manifolds of the stable Morse complex. Through a change in the sign of the Morse function f , the 2-manifolds (associated to the maxima) of the unstable Morse complex can be extracted.

In order to build a structural representation of a

given scalar field f , all the existing methods extract critical points of f as a first step of the global procedure. The most common approach to compute critical points examines, for each vertex p in the TIN, the neighbors points (sharing with p and edge) and computes the height difference between every point and p . If all differences are positive (p is lower than its neighbors), then p is a *minimum*. If all differences are negative (p is higher than its neighbors), then p is a *maximum*. If the number of sign changes of such difference, while traversing p 's neighbors in cyclic order, is two, then p is a regular, i.e., non-critical point. If the number of sign changes is four, then p is a *saddle*; if it is more than four, then p is a multiple saddle. This technique is used by almost all the algorithms, with the exception of (Bajaj and Shikore 1998).

Existing algorithms for extracting an approximation of a Morse (Morse-Smale) complex can be classified according to: the input they consider (namely RSG or TIN), the output they produce (namely an approximation of a Morse-Smale complex or of a Morse complex) and the algorithmic technique they choose. Here, we have classified them into *boundary-based* or *region-based* techniques (Comic et al. 2005).

Boundary-based techniques basically extract an approximation of the critical net, by computing the critical points and then tracing the integral lines, starting from saddle points (Bajaj et al. 1998; Schneider 2005; Takahashi et al. 1995; Edelsbrunner et al. 2001; Bajaj and Shikore 1998; Bremer et al. 2003; Pascucci 2004). *Region-based techniques* extract a discrete approximation of the stable and unstable Morse complexes, by starting from minima and maxima and letting a region grow until a given condition is reached (Danovaro et al. 2003a; Danovaro et al. 2003b; Meyer 1994; Vincent and Soille 1991; Mangan and Whitaker 1999). We included watershed algorithms in the latter class since they are region-based in nature.

In Section 5 we present our implementations of some representative algorithms of the above techniques. All algorithms, with the exception of the watershed approach, require that the three vertices of a triangle have distinct heights. This is generally achieved, when necessary, by perturbation of the height values.

4 THE STD ALGORITHM

In this section we present our novel algorithm, that we called *STD*, for extracting the 2-manifolds (associated with the minima) of a stable Morse complex for a (Morse) function f defined on a TIN. The algorithm

is region-based in nature since it starts from the minima and lets the 2-manifolds of the Morse complex grow as long as it is possible.

We first describe the algorithm under the assumption that no two vertices of the terrain have the same height. Successively, we relax this assumption and show how to deal with flat triangles, and triangles having one flat edge.

4.1 Basic Version of the Algorithm

The STD algorithm performs three main steps:

1. **Classify** the vertices of each triangle t in the TIN, based on their heights.
2. **Extract** the minima of the function in the TIN.
3. **For each** minimum p , **construct** the stable 2-manifold by iteratively adding triangles to it.

Vertex classification and Extraction of local minima. For each triangle t in the TIN, the highest, middle, and lowest vertex are labeled as *Source* (S), *Through* (T), and *Drain* (D), respectively.

By this STD configuration of the vertices we basically simulate the gradient direction of t in the discrete case. Note that this labelling does not assume any kind of interpolation (linear or higher-order) on triangles or edges of the mesh. Edge labelled S-D is not necessarily the edge of steepest descent. In Figure 2 the steepest descent is at edge labelled T-D.

The local minima identification is very simple: they are found as those vertices labeled *D* in all their incident triangles.

Construction of the stable 2-manifolds. For each minimum p , the stable 2-manifold γ_p associated with p is initialized with all triangles of the TIN which are incident in p . Successively, an iterative phase starts in which, at each step, the algorithm decides if a triangle t , externally adjacent to one edge e of the current perimeter of γ_p , can be added to γ_p . The rationale for this decision takes the following issues into account: (i) the choice must reflect the intuition that water flows from a higher to a lower height, (ii) the choice must be deterministic, i.e., a triangle t cannot be included into different 2-manifolds, depending on the order in which minima are processed.

The algorithm maintains the invariant that, if a triangle t has been included into γ_p , then the edge of t labelled T-D is not on the boundary of γ_p .

4.2 Inclusion of a Triangle

Let e be an edge of the current perimeter of γ_p , and t be the triangle externally adjacent to e . The decision whether to include t into γ_p or not, is based on

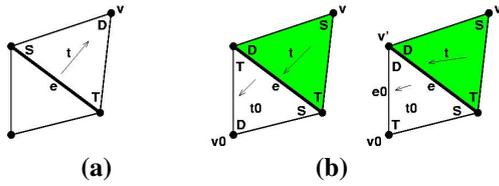


Figure 3: Case 1 (a) and Case 2 (b). Arrows denote water flow. Green triangles are included.

the STD configuration of its vertices. There are three possible cases.

Case 1. If the vertex v of t opposite to e is labelled D in t , then we do not include t into γ_p . See Figure 3 (a). This is according to the intuition that water cannot exit t through e , since it naturally flows towards v . Triangle t will be included when we will reach it from another edge, and Case 2 or 3 will hold.

Case 2. If the vertex v of t opposite to e is labelled S in t , then we include t into γ_p . See Figure 3 (b). Intuitively, water tends to flow across t and reach vertex v' , endpoint of e , which is labelled D in t . The question is whether it will exit t through e (in that case t belongs to γ_p) or through the edge of t labelled S-D. Now, we explain why we have decided that water passes through edge e .

Let t_0 be the triangle belonging to γ_p and adjacent to t along e , and let v_0 be the vertex of t_0 opposite to e . Note that, for the invariant, e cannot be labelled T-D in t_0 (equivalently, v_0 cannot be labelled S).

If e is labelled S-T in t_0 , then water enters t_0 through e , therefore it must exit from t through e .

If e is labelled S-D in t_0 , then water exits t_0 through its edge e_0 labelled T-D (it cannot exit through the other edge, since it is labelled S-T, and it must exit from one edge different from e otherwise t_0 would not have been included in γ_p). Therefore water that flows across t and reaches vertex v (which is labelled D in both t and t_0) turns around v' , enters t_0 , and finally exits t_0 through e_0 .

Note that the invariant is maintained: edge e (labelled T-D in the newly included triangle t) is inside the updated 2-manifold γ_p .

Case 3. If the vertex v of t opposite to e is labelled T in t , then the situation is more complex. Certainly, water flows to vertex v' , endpoint of e , which is labelled D in t . Then, will it exit from t into γ_p through edge e , or will it exit t through its edge e' labelled T-D, towards the 2-manifold existing on the other side?

Starting from t , we explore the maximal fan of triangles having their lowest vertex in v' (i.e., v' is labelled D in all such triangles). Let w be the vertex

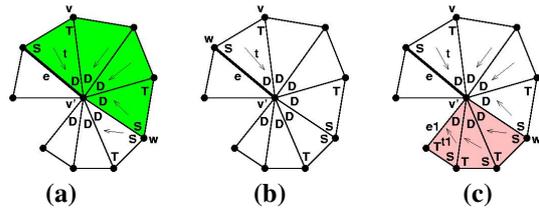


Figure 4: (a) Case 3 with non-empty set of included triangles; green triangles are included. (b) Case 3 with empty the set of included triangles. (c) Inclusion of the remaining triangles of the fan by applying Case 2 from edge e_1 .

of maximum height among the vertices of such triangles. The part of the fan starting from t and going up to edge $v'w$ is included into γ_p . See Figure 4 (a). The other part of the fan will be later included into the 2-manifold existing on the other side. Note that, if w is the same as the vertex labelled S in t , then no triangle is included. See Figure 4 (b).

The invariant is maintained since the edges remaining on the boundary of the updated 2-manifold γ_p are $v'w$, and edges opposite to v' : none of them is labelled T-D. In fact, edges opposite to v' are labelled S-T in the just included triangles, and edge $v'w$ is labelled S-D in both adjacent triangles.

Note that the management of Case 3 does not interfere with Case 2. In fact, the edge e_1 marking the other side of the fan may be labelled T-D in its adjacent triangle t_1 belonging to the fan. In this case, when reached from e_1 , t_1 will be included into the 2-manifold γ_q existing on the other side of e_1 . The triangle adjacent to t_1 along the other edge of t_1 incident in v' may be in the same situation (and thus be included in γ_q as well), and so on. Thus, a whole fan of triangles, starting from t_1 , is included into γ_q . But this fan must end at edge w , because the opposite vertex to $v'w$ is labelled T in the next triangle. Thus, there is no interference between Case 3 applied from edge e , and Case 2 repeatedly applied starting from edge e_1 . See Figure 4 (c).

4.3 Time Complexity

It can be easily shown that every triangle t of the TIN is examined at most three times, one from each edge, before being included into some 2-manifold. Thus, the worst-case time complexity of our algorithm is $O(n)$ where n is the number of TIN vertices. The only non-trivial part in this statement is showing that, in Case 3, a triangle can be in a traversed fan, without being included, at most once during the whole algorithm. The triangles of the fan, which are not included, are those located beyond edge $v'w$. The same fan may be traversed from the opposite side, while

growing another 2-manifold γ_q . Since we will be traversing the same fan in the opposite way, in that situation exactly those triangles, that were not previously included, will be found before edge $v'w$, and will be included into γ_q .

4.4 Management of Special Cases

Now, we explain how the STD algorithm deals with flat triangles, and triangles with a flat edge.

In a preprocessing step, we find edge-connected areas of flat triangles, and vertex-connected networks of flat edges that are not edge- or vertex-incident into a flat triangle. Such areas / networks are candidate to act as 1- or 2-dimensional local minima. Let h be the height of a flat area or network. Let h' be the minimum height of the third vertices of triangles externally adjacent to the perimeter of the flat area, or incident into edges of the network. If $h' > h$ then the flat area / network is treated as a local minimum: its 2-manifold is initialized with all the triangles of the flat area, or with all triangles incident in the flat network, and it is expanded in the same way as other 2-manifolds.

A flat area that is not a local minimum (i.e., $h' < h$) is assigned to the 2-manifold containing the triangle t' , externally adjacent to the flat area, whose third vertex has height h' . If t' is not unique, then we choose the 2-manifold corresponding to the lowest local minimum (if unique), or arbitrarily (otherwise).

During the algorithm, triangles with a flat edge may be examined to test whether they can be included into a growing 2-manifold. For such purpose, Cases 1, 2, and 3 introduce some exceptions when triangle t has a flat edge.

An exception may arise in Case 1, when the opposite vertex v , labelled D, is endpoint of the flat edge of t . In this case, we consider triangle t' which is adjacent to t along its flat edge. If edge e' is higher than the third vertex of t' , we do not include t (no exception). If edge e' is lower than the third vertex of t' , then this is an exception: we construct the fan of triangles incident into the vertex of t which is labelled D, and proceed in the same way as in Case 3.

Another exception arises in Case 2, when the opposite vertex v , labelled S, is endpoint of the flat edge of t . In this case, the two non-flat edges of t , e and e' , are in the same situation. We must decide whether to include t into γ_p from e , or to include t into the 2-manifold that will reach t from edge e' . We construct the fan of triangles incident into the vertex of t which is labelled D, and proceed as in Case 3.

In Case 3, the constructed fan cannot include flat triangles, and cannot include triangles with a flat

edge, when the flat edge belongs to a local minimum network. If we find one of these cases, then we stop extending the fan.

Case 3 takes the edge $v'w$, connecting the center v' of the fan with its upper point w , as the edge where to split the fan and assign its triangles to the 2-manifolds existing on the two sides of the fan. Now, vertex w of maximum height may not be unique. Let w_1, w_2, \dots, w_M ($M > 1$) be the vertices having the maximum height, sorted in counterclockwise order along the fan. We split the fan at edge $v'w_i$ where i is the integer result of division $M/2$.

5 REPRESENTATIVE MORPHOLOGY ALGORITHMS

We have implemented a number of algorithms that we have chosen as representative of the approaches existing in the literature (Section 3).

5.1 A Boundary-based Algorithm

Our implementation of a boundary-based approach, inspired by (Edelsbrunner et al. 2001; Takahashi et al. 1995), extracts the Morse-Smale complex from a TIN by computing the critical net, in two basic steps:

1. **Extract** the critical points and **unfold** multiple saddles.
2. **Compute the 1-cells** of the complex by starting from the saddle points, and tracing two paths of steepest descent and two paths of steepest ascent, which stop at minima and maxima, respectively.

Starting from each (simple) saddle p , the algorithm computes the four lines belonging to the critical net which are incident in p . At each step, the path is extended by adding the edge corresponding to the maximum positive [negative] slope, until a maximum [minimum] is found. In the implementation we present in this paper we refer only to the stable Morse complex: for each saddle we trace *two* lines which follow the maximum positive slope and stop when two maxima are found.

5.2 A Region-based Algorithm

We have presented in (Danovaro et al. 2003a) an algorithm for computing both the stable and unstable Morse complexes for a TIN. The algorithm can be sketched into two main steps:

1. **Extract** minima and maxima.

2. **Compute** the stable (unstable) Morse complex by applying a region-growing procedure. This procedure adds triangles to a 2-manifold iteratively.

For extracting the stable Morse complex, the algorithm computes the *gradient* for each triangle t in M , and the angles between the gradient and the normal vector at each edge of t (pointing outwards from the triangle). The edges of t corresponding to the largest and to the smallest angle are marked as *exit* and *entrance*, respectively.

A 2-manifold γ_p of the stable complex is initialized with the triangles incident in a local minimum p . At a generic step, γ_p is extended by adding a new triangle t sharing an edge e with γ_p , provided that e is an entrance for t and an exit for the triangle t' in γ_p sharing edge e with t . The unstable complex is computed in a completely symmetric way.

5.3 A Watershed Algorithm

We have implemented the watershed algorithm based on simulated immersion (Vincent and Soille 1991). Our implementation is applicable to TINs with flat edges and/or flat triangles and it consists of mainly three macro-steps:

1. **Sort** the vertices in increasing order with respect to the height value.
2. Perform the **flooding step** level by level, starting from the minima: this labels every vertex as belonging to a 2-manifold associated to a minimum.
3. **Assign** triangles to basins based on the labels of their vertices.

The flooding step assigns a distinct label to each minimum m and to the vertices of its associated 2-manifold γ_m . Those vertices, where two 2-manifolds meet are instead labeled as *watershed* vertices. At each iteration, a height value h (initially, the minimum height) is considered. All vertices with the same height h are first given a neutral label. Then those vertices whose neighbors have been labeled during the previous iteration are processed in order to assign the label of a 2-manifold γ_m to them.

To assign the label to a vertex p , we examine the neighbor vertices of p . If they all belong to the same 2-manifold γ_m or are watershed points, then p is marked as belonging to γ_m . If they belong to two or more different 2-manifolds, then p is marked as a watershed point. The same operations is recursively repeated on the neighbors points of the just labeled vertices which have a neutral label (i.e., height = h).

Vertices at height h that are not connected to any previously processed vertex still have the neutral la-

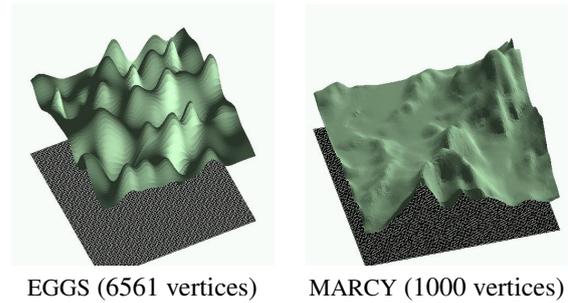


Figure 5: Two of the test TINs.

bel. Such vertices belong to a set of new minima at level h , and get a new label.

Finally, we label each triangle t . If all the vertices of t , that are not watershed points, belong to the same 2-manifold γ_m , then we assign the triangle to γ_m . If two vertices belong to different 2-manifolds, then t is assigned to the 2-manifold related to the vertex with the lowest height.

6 EXPERIMENTS

The goal of this section is to measure the quality of the results of the STD algorithm proposed in this paper, as well as evaluating the degree of uncertainty in morphology computation, i.e., to which extent the current algorithms are able to provide consistent results. We perform different experimental comparisons on both real and synthetic datasets by using our STD algorithm, the boundary-based (BND), the region-based (REG), and the watershed (WTS) algorithm described in Section 5.

Algorithm STD is of course very different from BND; STD and WTS have in common the idea of growing 2-manifolds from local minima; REG is similar in approach, but (i) uses the gradient, and (ii) it builds a 2-manifold in pieces which are then glued together, while STD builds every 2-manifold directly, thanks to the mechanism of fans (Case 3).

We show results using two different terrains: (i) **EGGS**, a synthetic terrain built by sampling a function which is a combination of two planes and 64 gaussian surfaces, and (ii) **MARCY** representing part of a real terrain model provided with the US Geological Survey in which heights have been perturbed in order to remove flat edges.

We have three TINs for EGGS, corresponding to different sampling resolutions (6,561, 25,921, and 103,041 vertices), and three TINs for MARCY, corresponding to approximations of the terrain at different resolutions (1,000, 5,000 and 10,000 vertices). See

Table 1: Triangles (t) and percentage of terrain area (a) assigned to a different 2-manifold in the new STD algorithm and in one of the other three methods.

# triang.		BND	REG	WTS
EGGS				
12,800	t	398	669	71
	a	3.11	5.23	0.55
51,200	t	1934	2,721	62
	a	3.78	5.31	0.12
204,800	t	14,828	14,488	112
	a	7.24	7.07	0.55
MARCY				
1,910	t	107	98	39
	a	3.89	2.95	1.64
9,788	t	554	690	151
	a	4.73	6.10	1.31
19,602	t	1,802	2,066	356
	a	9.20	10.54	1.82

Figure 5. Some images of the computed stable Morse complexes are in Figures 7 and 6.

Table 1 evaluates the difference in the results between our new STD algorithm and the other three. This also provides a measure of the uncertainty of results. In general, the STD algorithm tends to be closer to the watershed method.

Table 2 reports the quantity of TIN surface whose classification results uncertain (i.e., assigned to the 2-manifold of different minima in different algorithms). The various algorithms may disagree in their results up to an extent between 0.5% and 10.5% of the total TIN surface.

It may be surprising that algorithms differ so much in their results: up to 9% of the terrain area may be assigned to four different minima by the four considered approaches. It is also difficult to judge which one is more correct, because a ground truth is only available for C^2 functions, and not for TINs. Indeed, all existing methods only approximate Morse (or Morse-Smale) theory in the discrete case, through simplifications, conventions, and heuristics.

7 CONCLUDING REMARKS

We have proposed a new algorithm for computing the stable (unstable) Morse complex for a TIN terrain model. We performed experiments on both real and synthetic datasets in order to demonstrate the behavior of the STD algorithm with respect to other algorithms, as well as the intrinsic uncertainty of stable manifolds computation at this stage of research. We showed that our STD algorithm behaves quite well

Table 2: Triangles (t) and percentage of terrain area (a) assigned to a unique 2-manifold, or to 2, 3 and 4 different 2-manifolds, by the four algorithms.

# triang.		# of different 2-manifolds			
		1	2	3	4
EGGS					
12,800	t	11,963	42	397	398
	a	93.46	0.33	3.10	3.11
51,200	t	48,221	42	1,003	1,934
	a	94.18	0.08	1.96	3.78
204,800	t	184,608	48	5,316	14,828
	a	90.14	0.02	2.60	7.24
MARCY					
1,910	t	1,744	13	46	107
	a	94.18	0.64	1.31	3.87
9,788	t	8,835	56	343	554
	a	90.26	0.57	3.50	5.66
19,602	t	17,114	149	537	1,802
	a	87.31	0.76	2.74	9.19

for all the test datasets and that it provides intuitively good results. Moreover, our algorithm is very simple, and requires no floating-point calculations since it uses only numerical comparisons.

Morphology algorithms that can be extended to higher dimensions have a special interest from the scientific community. Our STD algorithm is as simple as the boundary-based approach and, unlike it, seem to be more easily extensible to higher dimensions. For instance, in 3D we label the four vertices of each tetrahedron and have four cases to be managed.

Finally, (Danovaro et al. 2003a) present a morphology-based multi-resolution terrain model, to encode different levels of approximation of a Morse-Smale complex. We plan to use the STD algorithm in this context.

ACKNOWLEDGEMENTS

This work has been partially supported by the European Network of Excellence AIM@SHAPE (contract n. 506766), by the National Science Foundation (grant CCF-0541032), by the MIUR-FIRB project SHALOM (contract n. RBIN04HWR8) and by the MIUR-PRIN project on “Multi-resolution modeling of scalar fields and digital shapes”.

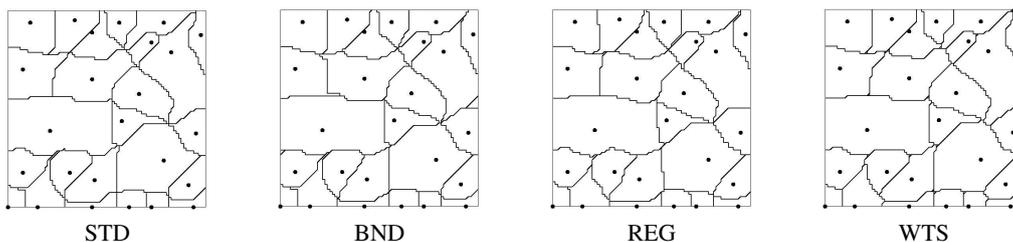


Figure 6: The boundary of the stable Morse complex computed by the four algorithms on the EGGS terrain (6561 vertices).

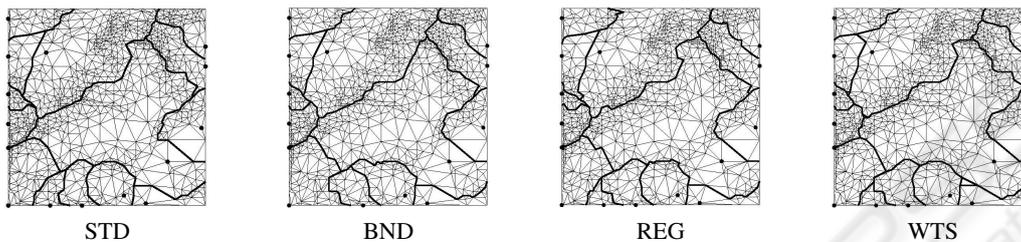


Figure 7: The boundary of the stable Morse complex computed by the four algorithms on the MARCY terrain (1000 vertices).

REFERENCES

- C. L. Bajaj, V. Pascucci, and D. R. Shikore. Visualization of scalar topology for structural enhancement. In *Proceedings IEEE Visualization'98*, pages 51–58. IEEE Computer Society, 1998.
- C. L. Bajaj and D. R. Shikore. Topology preserving data simplification with error bounds. *Computers and Graphics*, 22(1):3–12, 1998.
- P.-T. Bremer, H. Edelsbrunner, B. Hamann, and V. Pascucci. A multi-resolution data structure for two-dimensional Morse functions. In *Proceedings IEEE Visualization 2003*, pages 139–146. IEEE Computer Society, 2003.
- L. Comi, L. De Floriani, and L. Papaleo. Morse-Smale decomposition for modeling terrain knowledge. In *Spatial Information Theory: International Conference*, volume 3693 of *Lecture Notes in Computer Science*, pages 426–444, 2005.
- E. Danovaro, L. De Floriani, P. Magillo, M. M. Mesmoudi, and E. Puppo. Morphology-driven simplification and multi-resolution modeling of terrains. In *Proceedings ACM-GIS 2003 - The 11th International Symposium on Advances in Geographic Information Systems*, pages 63–70. ACM Press, 2003.
- E. Danovaro, L. De Floriani, and M. M. Mesmoudi. Topological analysis and characterization of discrete scalar fields. In *Theoretical Foundations of Computer Vision, Geometry, Morphology, and Computational Imaging*, volume 2616 of *Lecture Notes on Computer Science*, pages 386–402. Springer Verlag, 2003.
- H. Edelsbrunner, J. Harer, and A. Zomorodian. Hierarchical Morse complexes for piecewise linear 2-manifolds. In *Proceedings 17th ACM Symposium on Computational Geometry*, pages 70–79. ACM Press, 2001.
- A. Mangan and R. Whitaker. Partitioning 3D surface meshes using watershed segmentation. *IEEE Transactions on Visualization and Computer Graphics*, 5(4):308–321, 1999.
- F. Meyer. Topographic distance and watershed lines. *Signal Processing*, 38:113–125, 1994.
- V. Pascucci. Topology diagrams of scalar fields in scientific visualization. In *Topological Data Structures for Surfaces*, pages 121–129. John Wiley and Sons Ltd, 2004.
- J. L. Pfaltz. Surface networks. *Geographical Analysis*, 8:77–93, 1976.
- J. Roerdink and A. Meijster. The watershed transform: definitions, algorithms, and parallelization strategies. *Fundamenta Informaticae*, 41:187–228, 2000.
- B. Schneider. Extraction of hierarchical surface networks from bilinear surface patches. *Geographical Analysis*, 37:244–263, 2005.
- B. Schneider and J. Wood. Construction of metric surface networks from raster-based DEMs. In *Topological Data Structures for Surfaces*, pages 53–70. John Wiley and Sons Ltd, 2004.
- S. Smale. Morse Inequalities for a Dynamical System, *Bulletin of American Mathematical Society*, 66: 43–49, 1960.
- S. L. Stoev and W. Strasser. Extracting regions of interest applying a local watershed transformation. In *Proceedings IEEE Visualization'00*, pages 21–28. IEEE Computer Society, 2000.
- S. Takahashi, T. Ikeda, T. L. Kunii, and M. Ueda. Algorithms for extracting correct critical points and constructing topological graphs from discrete geographic elevation data. *Computer Graphics Forum*, 14(3):181–192, 1995.
- L. Vincent and P. Soille. Watershed in digital spaces: an efficient algorithm based on immersion simulation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 13(6):583–598, 1991.