JOURNAL OF SPATIAL INFORMATION SCIENCE
Number 22 (2021), pp. 33–59
doi:10.5311/JOSIS.2021.22.681

RESEARCH ARTICLE

# Surface network extraction from high resolution digital terrain models

Eric Guilbert

Department of Geomatics Sciences, Laval University, Québec, Canada

*Received: July 30, 2020; returned: October 28, 2020; revised: December 2, 2020; accepted: February 11, 2021.*

**Abstract:** A surface network is a topological data structure formed by a set of thalwegs and ridges on a digital terrain model. Its computation relies on the detection of saddles on the terrain. Hence, computation methods must guarantee enough saddles are detected but also that no improper conflicts between ridges and thalwegs are created, leading to an inconsistent network. This paper presents a new approach that maximizes the number of saddles and ensures this topological consistency for high-resolution terrain models represented by a raster grid. The grid is triangulated in order to preserve saddles and to facilitate thalweg and ridge computation. It does not require any user parameter and lines remain aligned with triangulation edges, avoiding numerical errors. The method also includes a coherent partitioning of the terrain into hills and dales. A case study shows that the surface network computation can be achieved in reasonable time and hence can be applied to the analysis of large terrain models.

**Keywords:** digital terrain modeling, digital elevation model, surface network, Morse-Smale complex, thalweg, ridge, data structure, geomorphometry

## 1 Introduction

In environment science, geomorphometry is an essential tool for terrain modeling and analysis. One main objective of geomorphometry is to provide a classification of the land surface into significant elements. A seminal work in this direction is [17]. The authors describe a method to detect pits, peaks, passes, ridges, and thalwegs (or ravines) on a terrain model. For that purpose, they rely on the Morse theory which provides a mathematical definition of each element. From there, new methods were developed in two directions. On one

side, terrain segmentation methods looked at segmenting the terrain into morphometric classes [25]. On the other side, methods computed critical elements gathered into a surface network, forming a skeleton of the terrain [21]. While the former provides a segmentation into landform elements used for landform analysis, the latter provides a data structure which was used mainly for terrain simplification and visualization.

A surface network is a topological data structure formed by the arrangement of critical lines (ridges and thalwegs) on a terrain. It provides a compact description of the terrain morphology based on its critical lines [24]. Surface networks are extracted from digital terrain models (DTM), more than often modeled by rasters. A surface network is modeled by a tripartite graph where critical lines are edges connecting critical points (pits, peaks, and saddles). The network obeys certain rules that ensure its topological consistency: saddles are connected to pits by a thalweg and to peaks by a ridge, and ridges and thalwegs cannot intersect except at saddles (Figure 1). The surface network has been mainly used for multiple scale visualization and DTM simplification in order to preserve terrain features [5].

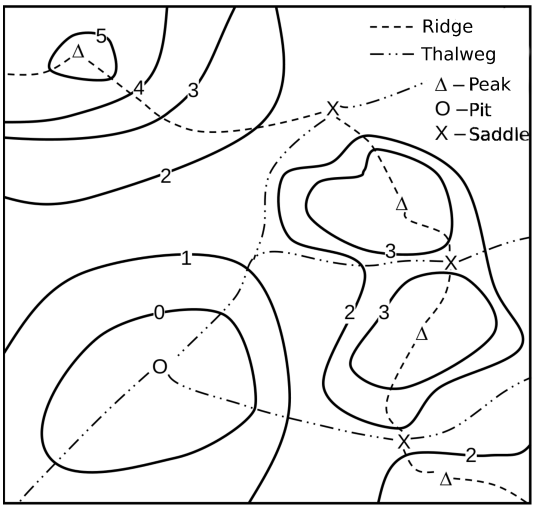


Figure 1: Surface network formed by a set of peaks, pits, and saddles connected by ridges and thalwegs [11].

On raster DTMs, surface networks are usually calculated by detecting saddles and then initiating ridges and thalwegs at each saddle point by following the steepest slope upward or downward to a peak or a pit. These critical lines partition the terrain into subregions in two ways. The set of ridges provides a decomposition into dales where each dale contains a pit. Meanwhile, the set of thalwegs partitions the terrain into hills, each of them containing a peak. This approach directly yields a topological structure and does not require any parameter with user-defined threshold. However, while surface networks are based on the Morse theory for 2D continuous manifolds, they suffer some limitations in practice because the DTM is a discrete representation of a continuous function.

First, surface network computation relies on the detection of saddles. The more saddles there are, the denser the network. This depends on the terrain resolution and its smoothness. Second, the computation of critical lines and points also relies on the gra-

dient computation and requires that no two adjacent pixels have the same elevation so that a consistent steepest slope can always be calculated. This can never be guaranteed on sampled data, leading to singularities. Third, overlaps or crosses between thalwegs and ridges outside saddles may occur leading to an inconsistent network. This phenomenon is also due in part to the discrete representation problem which results in an inability to achieve a proper decomposition in hills and dales. Indeed, these computation methods focused on the extraction of ridges and thalwegs. They proposed some ad hoc techniques to handle inconsistencies on adjacent pixels and on lines but these techniques do not guarantee a proper computation of hills and dales.

Other approaches were also applied on TINs (triangulated irregular networks) where hills and dales are computed first [5]. However, these approaches also require some ad hoc treatments to handle points at the same height. They focused on the terrain segmentation and they may not guarantee the topological correctness of critical points and lines.

Furthermore, the advancement of airborne lidar technologies has allowed for the production of very high resolution DTMs that provide much more detail about the terrain. As such, a much larger set of critical points can be extracted from the DTM, leading to a denser set of critical lines. But this larger set is also more likely to generate inconsistencies that would hinder the construction of a proper network. Since topological errors cannot be corrected post-extraction, the method for extracting the surface network must be robust enough to handle singularities and guarantee the topological correctness of the network.

Hence, in this paper, a computational method that ensures the consistency of the network is proposed. The method is designed for raster DTMs and provides a consistent data structure including critical nodes and lines in addition to a decomposition of the DTM into hills and dales. While some approaches compute critical points and lines at sub-pixel accuracy, critical points and lines in the proposed approach always match raster pixels and edges. In this way, uncertainties due to computational errors are avoided and no numerical parameter is required. Furthermore, when working on high resolution DTMs, the precision obtained by sub-pixel computation is not meaningful in comparison to the DTM accuracy. Specific attention is also applied to the handling of singularities, when critical lines merge or overlap. Constraints are defined to take care of such issues and maintain a correct data structure. The following contributions have been introduced in the proposed method:

- A new saddle computation technique that maximizes the number of saddles identified on the terrain;
- A new critical line computation method where lines match with pixels and remain close to the steepest slope;
- A data structure that guarantees the consistence of the network and includes hills and dales.

The paper first reviews other surface network computation methods (Section 2) and introduces the new approach in Section 3. More specifically, Section 3 presents a technique for extracting saddle points, a method for computing thalwegs and ridges and also discusses the computation of hills and dales. Section 4 presents results on a case study. The final section concludes the work and provides perspectives for the application of the method in terrain analysis.

# 2 Existing approaches for surface network computation

## 2.1 Definition of a surface network

A surface network is a tripartite graph where vertices are peaks, pits, and saddles of the terrain and edges are ridges and thalwegs. Surface networks are grounded in the Morse theory. The terrain is considered as a 2D manifold defined by a scalar function $z = f(x, y)$ over a domain $D \subset \mathbb{R}^2$ where $f$ is a height function. If $f$ is twice differentiable, critical points are points where the gradient is zero: $\vec{\nabla} f(p) = \vec{0}$ and a critical point is called degenerated if its hessian is equal to zero. The function $f$ is a Morse function if none of its critical points is degenerated.

On a Morse function, critical points can be of three kinds:

- a pit: the point $p$ is a local minimum;
- a peak: $p$ is a local maximum;
- a saddle: $f$ has the shape of a hyperbolic paraboloid in the vicinity of $p$.

An integral line is a path on the terrain that follows the gradient direction, i.e. the direction of steepest ascent. This path always ends at a peak. Going in the opposite direction, the steepest descent will always lead to a pit. Hence an integral line starts at a pit and ends at a peak and the terrain can be decomposed into descending or ascending manifolds (Figure 2). The descending (or stable) manifold of a pit is the set of integral lines originating from the same pit. Similarly, the ascending (or unstable) manifold of a peak is the set of integral lines that end at this peak.

The terrain can be partitioned into a complex in which each cell is a descending manifold. Lines delineating each manifold are lines of divergent flow called ridges: integral lines on each side of this line originate from two different pits. Furthermore, following this ridge leads to a peak. Hence, on a terrain, the descending manifold of a pit corresponds to a dale and this manifold is bounded by ridges connecting at peaks (Figure 2, left). Similarly, one can build a complex of ascending manifolds where lines separating two manifolds are thalwegs that connect at pits (Figure 2, right). This manifold can be seen as a hill on the terrain.

The intersection of an ascending manifold around a peak $p$ and a descending manifold around a pit $q$ results in a Morse-Smale cell delineated at its bottom by the pit and the two thalwegs originating from $q$ and at the top by the peak and the two ridges terminating at $p$ (Figure 2 bottom). The two points located at the thalweg-ridge intersections are saddles. Hence, a Morse-Smale cell is always delineated by a quadrangle formed by a sequence of pit – thalweg – saddle – ridge – peak – saddle – thalweg [7] and includes all the integral lines starting at $q$ and ending at $p$. Intuitively, a Morse-Smale cell corresponds to a hillslope of uniform flow. The Morse-Smale complex is the set of all Morse-Smale cells obtained by intersecting all descending and ascending manifolds.

Based on the definitions above, a Morse-Smale complex always satisfies the following properties:

- Ridges and thalwegs intersect only at saddles;
- A ridge always connects a peak and a saddle;
- A thalweg always connects a pit and a saddle;
- At a saddle, the number of ridges is equal to the number of thalwegs;
- When turning around a saddle, edges alternate between ridges and thalwegs;

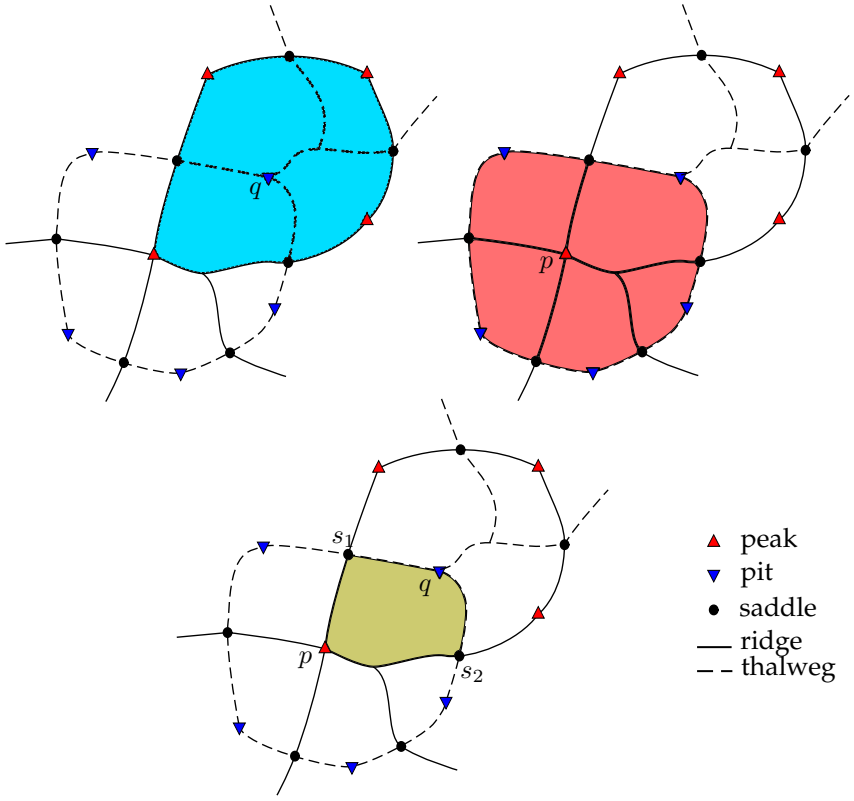


Figure 2: Top left: descending manifold formed by all integral lines originating from pit $q$. Top right: ascending manifold formed by all integral lines ending at peak $p$. Bottom: Morse-Smale cell obtained by the intersection of a descending and an ascending manifold. All integral lines in this cell start at $q$ and end at $p$.

- Each ascending cell contains exactly one peak;
- Each descending cell contains exactly one pit.

Considering $f$ as a closed manifold, topologically equivalent to the surface of the sphere, its descending complex can be seen as a CW-complex where each face is a descending manifold associated with a pit, bounded by edges and vertices. Each vertex is a peak and each edge is a pair of ridges connected by a saddle. As a CW-complex, it satisfies the Euler-Poincaré characteristic such that:

$$\#\text{peak} + \#\text{pit} - \#\text{saddle} = 2 \tag{1}$$

In the case of a terrain, $f$ is usually not closed and a virtual pit, i.e. a pit located below the terrain, can be added to close the surface. Adding this virtual point allows satisfaction of the Euler-Poincaré characteristic and facilitates handling boundary cases.

## 2.2 Computational approaches

Morse-Smale complexes are defined for smooth surfaces. When working on a DTM defined by a raster, the elevation and its gradient are no longer continuous over the whole terrain, bringing singularities in the definition of critical points and lines. The earliest approach for computing a surface network on a grid DTM was presented by Peucker and Douglas [17]. The method was based on a pixel classification approach where each pixel was compared to its neighbor elevations. The approach yields a classified raster and can be used to select characteristic points on a surface, such as [8] but does not return a topological structure. Indeed, this approach led to the development of several methods in geomorphometry where the terrain is segmented into six geomorphometric classes. Classification is done by computing derivatives at each pixel using a local window. Hence, the result depends on the size of the window and on threshold values to discriminate the classes. [25] proposed a method using different window sizes to provide a more robust classification. Further works led to computing a larger number of classes based on the curvature [20] or more recently using geomorphons [12].

Later on, several approaches were developed to compute a Morse-Smale complex [3]. Three approaches have been applied to DTMs and more specifically to TINs. The first approach is the watershed approach. It has been designed for segmentation in image processing. It is based on the idea of the immersion of the terrain [23] or, in the other way round, on simulating rainfall [14]. These methods assign a label to each pixel. A label corresponds to a catchment area or to a watershed separating two catchment areas. One issue is that both methods do not yield the same classification because of the discrete representation [3].

A second approach is based on a growing region approach [5]. Starting from a triangle, a region is built by following the gradient direction. The algorithm is applied a first time to build the ascending manifold and a second time, by following the steepest descent, to build the descending manifold. The Morse-Smale complex is built by intersecting the manifolds. Critical lines are lines separating two manifolds and critical points are nodes connecting critical lines. However, according to [3], saddles are not always located on lines.

The third approach is based on the detection of critical points followed by the detection of critical lines [7, 21] defining the boundaries of Morse-Smale cells. Since critical lines are initiated from saddles, the consistency between lines and saddles is always guaranteed. However, boundary-based approaches can generate dangling lines, which are located inside a manifold, that do not correspond to a Morse-Smale cell boundary.

Both the region-based and the boundary-based approaches have been used for TIN simplification with the objective of preserving the main terrain features. A comparison of these different approaches was conducted in [13] on a synthetic dataset and on a TIN. It showed that both approaches as well as immersion simulation yield similar results and the authors could not conclude that a method was more effective than another.

Computation methods specific to raster DTMs all rely on the boundary-based approach. These methods were not developed only for simplification purpose but also for the extraction of salient features on the terrain, considering that the set of critical lines provides a skeleton of the surface with an explicit topology that can be used for further interpretation and analysis. In comparison with segmentation methods based on geomorphometric classification, surface network computation is parameter-free and provides a set of points and lines connected together in a graph. Segmentation methods compute segments defined as clusters of pixels where peaks or pits are not necessarily defined by a single pixel. Similarly,

ridges and thalwegs do not necessarily correspond to lines connecting to saddles and peaks or pits and are instead regions characterized by their relief.

Two main approaches are considered to handle a raster DTM: the DTM is triangulated to define a piecewise linear surface, in which case, methods designed for TINs can apply, or pixels form quadrangles and the surface is modeled as a piecewise bilinear surface.

## 2.3 Computation on a piecewise linear surface

A method specific to piecewise linear surfaces defined from a raster DTM was introduced by Takahashi et al. [21]. The DTM is first triangulated by inserting a diagonal in each cell. Depending on the triangulation, a point can have between four and eight neighbors.

Pits and peaks are simply defined as points respectively below and above all their neighbors. A saddle is defined by scanning its neighbors and counting the number of times $N$ they alternate from being higher or lower than the point. If they alternate four times ($N = 4$), the point is a simple saddle. If $N = 6$ or $N = 8$, the point is a double or triple saddle. Hence, the multiplicity of a saddle is given by $\mu = \frac{N}{2} - 1$.

In order to satisfy the Euler-Poincaré characteristics, saddles are counted with their multiplicity and a virtual pit is added to the terrain. A virtual pit is a point located below the terrain with an elevation equal to $-\infty$. It closes the terrain to form a manifold topologically equivalent to the surface of a sphere.

The algorithm defines critical lines by starting at saddle points and marching along pixels to the lowest neighbor or the highest neighbor until reaching a pit or a peak. Hence, critical lines are always aligned with edges of the triangulation. The number of ridges and thalwegs at a saddle are always equal and are related to the multiplicity of the saddle.

Properly identifying critical points on the grid requires that two adjacent points are not at the same elevation so that a strictly ascending or descending direction can always be computed. In order to handle flat areas, Takahashi et al. introduced a simple but robust ordering rule: if two points share the same height, they are ordered according to their lexicographical order, i.e. according to their $x$ and $y$ coordinates. In this way, comparisons can be made without consideration for singularities.

While the method is robust, it still suffers limitations that can hinder further analysis or processing. First, the result depends on the diagonalization which directly affects the detection of critical points and construction of critical lines. Figure 3 shows two surface networks computed from the same terrain with two different triangulations. The underlying terrain is a rather flat terrain with small variations. On the left, each pixel cell is divided by a diagonal running from the top left to the bottom right corner. On the right, diagonals are oriented from top right to bottom left. This leads to significant differences in the detection of critical points and lines. Second, two thalwegs or two ridges can merge and cross at places other than saddles (black circles on Figure 3). This leads to an invalid Morse-Smale complex where Morse-Smale cells cannot be defined correctly.

Other approaches that were developed for triangulated irregular networks (TIN) can also apply to triangulated grids [2, 7]. These approaches were developed to simplify a surface while preserving its features. Edelsbrunner et al. [7] computed a complex in several steps. They first compute integral lines by moving along edges of the triangulation. When two integral lines merge, they are not considered to overlap but to be infinitely close. By preserving their relative positions, the authors ensure that the lines never cross. Since inte-

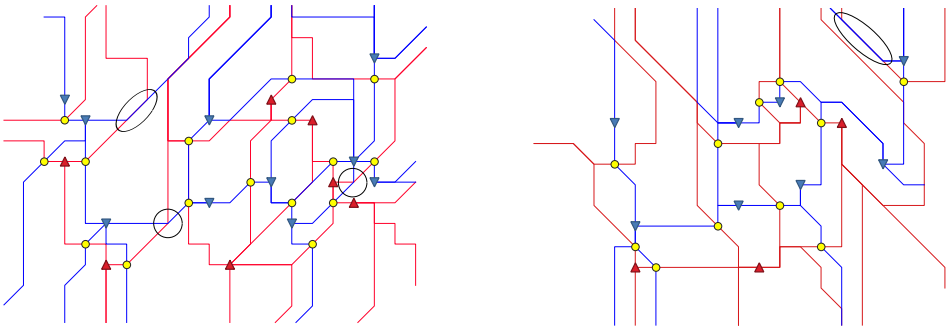

Figure 3: Surface networks obtained from two different triangulations. Saddles in yellow, thalwegs and pits in blue, ridges and peaks in red. Black circles highlight inconsistent crossings between ridges and thalwegs.

gral lines do not follow the gradient, they apply a transformation to reroute the lines along the gradient.

Bremer et al. [2] presented a method where integral lines directly follow the gradient. Paths are recorded by splitting triangles so that critical lines are still aligned with edges. Two ridges or two thalwegs can merge but instead of keeping two overlapping lines, a junction point is introduced and lines are split at the junction. On the opposite, a ridge and a thalweg cannot cross or overlap. In order to avoid any intersection, a new edge is inserted in the triangulation to offer an alternative path. While the previous methods both introduce a virtual pit, Bremer et al. defined the domain boundary as a path with its critical points. In the last two methods, authors also handle vertices at the same height by introducing an ordering rule based on the value of the local gradient.

One common issue to all these methods is that the original grid must be triangulated. However, the result will depend on the orientation of the diagonals. Most importantly, critical lines are initiated at saddles. If some saddles are ignored, some critical lines may be missing. Hence, other methods that do not require a triangulation were developed.

## 2.4 Computation on a piecewise bilinear surface

The second general approach to computing a Morse-Male complex considers each quadrangle formed by four pixels as a bilinear surface [16, 18]. Each pixel has exactly four neighbors. Pits and peaks are also located at pixels but saddles can be located inside a quadrangle. In all cases, saddles are always simple, meaning that exactly two ridges and two thalwegs are initiated from each saddle. These critical lines are aligned with the pixel edges when the saddle is on a pixel and at a 45° angle when the saddle is inside a quadrangle. Gradient paths are computed along edges and through quadrangles. Since the surface is bilinear, they do not follow a straight line across a quadrangle and are computed by a gradient descent.

The approach was first introduced by Schneider [18], who does not add a virtual pit but includes a path along the boundary. This approach does not consider singularities such as pixels at the same elevation or prevent thalweg-ridge intersections at non-saddle points. Conversely, when such intersections occur, pseudo-saddle points are inserted. Indeed, such

intersections usually occur in rather flat areas such as alluvial plains where thalwegs and ridges are not clearly identifiable. While this ensures the topological consistency of the surface network, the Euler-Poincaré characteristic is no longer satisfied. Furthermore, as with previous methods, intersections between critical lines occur because the gradient is not continuous over the whole terrain. Hence inconsistencies should be seen as issues related to the terrain representation as compared to natural terrain features.

A later method was proposed by Norgard and Bremer [16]. They present a full analysis of the gradient in each quadrangle, recording all changes of sign. As such, their solution prevents intersection between ridges and thalwegs within a quadrangle. However, it does not alleviate computational issues and several critical lines can run through a same quadrangle where their position depends on the step of descent (Figure 4 left). This leads to details at a precision much below the resolution of the DTM and may be superfluous in the case of high resolution DTM: it generates many Morse-Smale cells that are not significant. Furthermore, while the method guarantees that thalwegs and ridges cannot cross, two critical lines of the same type may still cross (top left of Figure 4). A smaller descent step may reduce the number of conflicts but cannot completely rule them out since they are caused by the discrete computation of the gradient. While the gradient computed inside grid cells is smooth, it is still discontinuous between two quadrangles so that the staircase effect observed on linear surfaces is still occasionally observed on piecewise bilinear surfaces (Figure 4). Finally, the approach does not take into account the case where a saddle is located on the edge of a quadrangle. While this case does not occur in theory, it still can happen on discrete data, leading, for example, to quadrangles containing two saddles (one inside and one on the border) or an edge containing two saddles (each belonging to a different quadrangle).

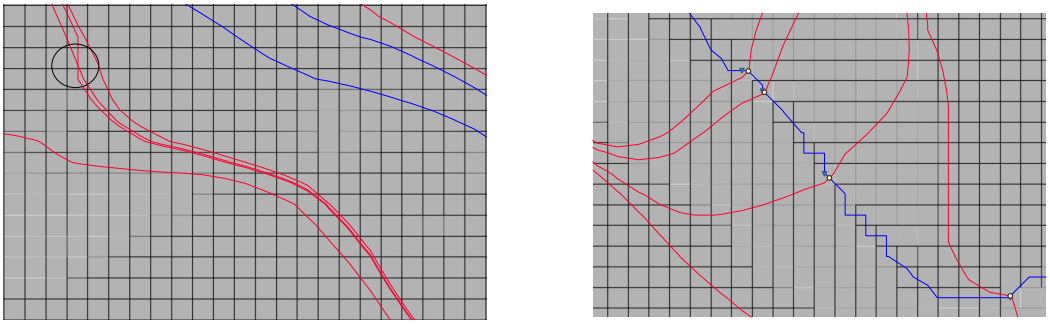

Figure 4: Critical lines computed on a piecewise bilinear surface. Left: several lines run through the same pixel and intersect in the top left corner. Right: staircase effect.

On top of these approaches, Schneider and Wood [19] also introduced a computation method that applies to piecewise biquadratic surfaces. The method also applies to a raster but each biquadratic patch is computed from a $3 \times 3$ (or more) block of pixels forming a conic section. Critical points are classified according to the type of conic section, requiring prescriptive threshold values to discriminate them. Hence, the approach does not detect all critical points but only those that can be identified at the given scale and thresholds.

## 2.5 Discussion

Among these different methods, [21] provides a solution where all critical points are on vertices and lines are aligned with edges of the triangulation. However, the method does not handle inconsistencies between thalwegs and ridges, limiting further analysis on the network. Methods based on a TIN provide a way to guarantee the consistency of the network. The introduction of junctions allows for an easier handling of the structure, avoiding overlapping lines. These methods also provide a more accurate computation of the gradient. However, such approaches require the insertion of new nodes in the TIN. This technique is not applicable if one wants to preserve the grid structure. Splitting triangles can also lead to numerical issues when nodes are close to the grid vertices, creating small triangles.

If the terrain is modeled by a bilinear surface, no triangulation is required. Saddles and lines are no longer restricted to vertices and edges. Working at subpixel precision yields smoother lines. However, inconsistencies between critical lines may still remain. Lines running in parallel at close distance may occur along narrow slopes, such as when several thalwegs run along a valley floor. This leads to difficulties to identify the proper line that marks the valley thalweg. When working on high resolution DTMs such as 1 m DTM built from lidar data, positioning several lines at centimetric precision within a 1 m cell may not increase the accuracy of the network. It can instead bring unwanted detail and yield a more complex structure than one where lines match with grid cell edges. In both bilinear approaches, the authors did not consider the case where vertices share the same elevation. In [16], the authors consider that vertices are all at a different height. This is an assumption that cannot be made when dealing with large, high-resolution DTMs. A useful algorithm should be able to handle such singularities.

Finally, the methods presented above do not address the definition of hills and dales. While hill and dale computation was not presented as an objective of these methods, one benefit of their computation is the validation of the topology. A valid network of ridges and thalwegs should automatically lead to a valid partition of the DTM into hills and dales. Conversely, inconsistencies such as non-valid critical lines intersection will lead to ill-defined hills and dales where, for example, a hill does not contain any peak or a ridge crosses two dales.

Following these observations, in the case of a high-resolution DTM, computation of a surface network is more robust on a triangulated grid. Critical lines should remain aligned with the edges and diagonals of the grid cells, preserving the topological accuracy of the network while maintaining sufficient precision. To achieve critical line alignment, grid coordinates are stored as integer values facilitating computation and comparison for intersection and consistency check. In the wake of these considerations, the method should handle the following issues:

- The number of critical lines depends on the number of saddles but saddle detection depends on the triangulation. The method should aim at maximizing the number of saddles to capture as much information as possible;
- Direction for ascent and descent is computed along edges and diagonals and often does not follow the steepest slope. Lines may deviate from the integral path and a correction may be required.
- Sticking to edges and diagonals is more likely to create overlaps and intersections between thalwegs and ridges. Such cases must be handled and explicit topological relationships must be recorded in the data structure;

- Recording dales and hills in the data structure must also be done with consideration of the above issues.

# 3 Computation of the surface network

## 3.1 Overview

The objective of the proposed method is to extract thalwegs and ridges corresponding to characteristic features of the terrain. While in previous works, thalwegs and ridges were seen as equivalent structures and processed in a similar way, this method considers that thalwegs should be given more importance. Indeed, one of the most important components used in terrain modeling and analysis is the drainage network. Streams on a terrain correspond to areas of higher water accumulation and often fit with the thalweg network, especially in fluvially eroded terrains. Hence, thalwegs present features of interest for hydrologists and geomorphologists [1]. On the opposite, ridges do not have so clear an importance and the definition of a ridge in a surface network does not correspond to the definition in hydrology: in hydrology, ridges correspond to water divides and do not join thalwegs (or streams) at saddles but instead at the confluence of two streams. Ridges also correspond for the geomorphologist to passive areas preserving original terrain elements. Therefore, in this method, thalwegs are extracted first since they are the results of the different processes shaping the relief. Ridges are extracted after and their shape and position will be constrained by existing thalwegs.

As in earlier methods, thalwegs and ridges are initiated at saddle points. In order to detect all possible lines, as many saddles as possible must be detected. On a raster DTM, a pixel can have eight neighbors. Triangulating the raster first always misses some saddles because all neighbors are not considered. However, extracting all saddles based on their eight neighbors can lead to conflicts [21]. In Figure 5 left, all potential saddles with the first segment of their critical lines were computed. In comparison with Figure 3, more saddles are detected but intersections occur between lines coming from adjacent saddles. A different approach is proposed where the number of saddles is maximized by first detecting all saddles based on their eight neighbors and then removing conflicting saddles. The terrain is then triangulated in order to preserve all the remaining saddles.

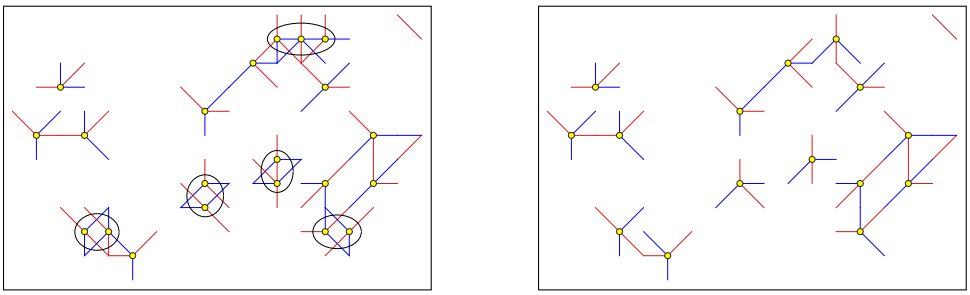

Figure 5: Left: clusters of conflicting saddles. Right: saddles after conflict removal. Ridges in red, thalwegs in blue.

Triangulation is done in order to favor steepest descent in each raster cell by connecting the diagonal with the lowest vertex. Such an approach facilitates the extraction of thalwegs. Before extracting the ridges, diagonals will be flipped to connect with the highest vertex in each cell. Ridge computation is then constrained by thalwegs to prevent crosses. Overall steps of the computation process are developed in the subsequent sections and are summarized in algorithm 1.

**Algorithm 1** Overview of the surface network computation algorithm

1: Compute saddles
2: Triangulate the raster
3: Compute thalwegs
4: Compute hills
5: Adjust triangulation
6: Compute ridges
7: Compute dales

## 3.2 Saddle computation

Saddle computation is done by scanning the whole terrain and for each pixel, comparing it with its eight neighbor pixels. As in previous approaches, the pixel elevation is compared with its neighbors. If the sign of the elevation difference changes four times or more, the pixel is marked as a saddle. Initial segments of each critical line are defined by local steepest slopes. Depending on its multiplicity, a saddle can have between 2 and 4 thalwegs and as many ridges.

At the end of the computation, several conflicts are created. They occur only when two saddles are adjacent and their critical lines cross, usually because the terrain is flat or nearly flat with small height differences. Only diagonal lines can cross. Since a saddle can be in conflict with several other saddles, a correction is done by grouping saddles in clusters (Figure 5 left). Each cluster contains two or more saddles in conflict. A conflict is corrected by reducing the possible number of neighboring pixels of a saddle and hence the possible directions for critical lines (Algorithm 2). Starting with the first conflicting saddle in a cluster, the method removes the pixels where conflicts occur from its list of neighbors. It then checks within the new neighborhood whether the pixel is still a saddle. If it is not, the pixel is removed from the cluster and from the saddle list. If the pixel is still a saddle but there is no more conflict, the saddle is removed only from the cluster and the method can move to the next conflict. The process is repeated until all conflicts are resolved. An example is shown in Figure 5. On the left are the saddles grouped in clusters. On the right is the result obtained after correction. Saddle selection depends on the order in which saddles are assessed for conflicts in each cluster. Between two adjacent saddles, conflicts would occur between the thalwegs of the highest saddle and the ridges of the other. Since the process starts with thalweg computation, lowest saddles are checked first to preserve the thalwegs.

## 3.3 Thalweg computation

Once saddles are detected, the raster is triangulated. As mentioned earlier, triangulation is done by choosing an orientation for the diagonals in each cell. Triangulation around

**Algorithm 2** Saddle selection

```
1: function SELECTSADDLES(S list of candidate saddles)
2:     C ← ∅ empty list of clusters
3:     for each s in S do
4:         if ¬s.visited then
5:             c ← [s] new cluster with one saddle
6:             s.visited ← True
7:             stack ← [s]
8:             while stack do
9:                 t ← stack.pop()
10:                for all saddles around t do                    ▷ D4 connectivity
11:                    saddle.visited ← True
12:                    stack.append(saddle)
13:                    c.append(saddle)
14:                end for
15:            end while
16:            if length(c) > 1 then                ▷ No conflict if only one saddle in cluster
17:                C.append(c)
18:            end if
19:        end if
20:    end for                                      ▷ All clusters have been computed
21:    while C do
22:        c ← C.pop()
23:        while length.c > 1 do
24:            s ← min(c)                                      ▷ Take the lowest saddle
25:            l ←conflict(s, c)        ▷ Critical lines of s in conflict with other saddles in c
26:            while l ≠ ∅ do
27:                Remove directions of l from s neighbors
28:                checkSaddle(s)                  ▷ Recompute saddle with new neighbors
29:                if s is no longer a saddle then
30:                    S.remove(s)
31:                    l = ∅
32:                else
33:                    l ←conflict(s, c)
34:                end if
35:            end while
36:            c.remove(s)
37:        end while
38:    end while
39:    return S
40: end function
```

saddles is done to preserve critical lines. In order to facilitate the flow and limit the number of spurious pits, other diagonals are chosen to connect the lowest pixel [6].

Pits are defined as points lower than all their neighbors. A virtual pit located below the terrain is also added. As in [2], to avoid overlaps between thalwegs, confluences are inserted when a thalweg merges with an existing thalweg. Hence, when two thalwegs merge, the confluence is the end node of the two thalwegs and the starting node of a third thalweg. Figure 6 shows thalwegs running along a slope. The valley marking the bottom of the slope is located on the top right of the image. Similarly, if a thalweg goes through a saddle, it is also interrupted and the saddle becomes the end point of the thalweg.

The steepest descent direction is the direction opposite to the gradient and does not necessarily follow triangle edges. Hence, the descent runs along the closest possible direction to the steepest descent. However, in that case, descent is always done along one of the

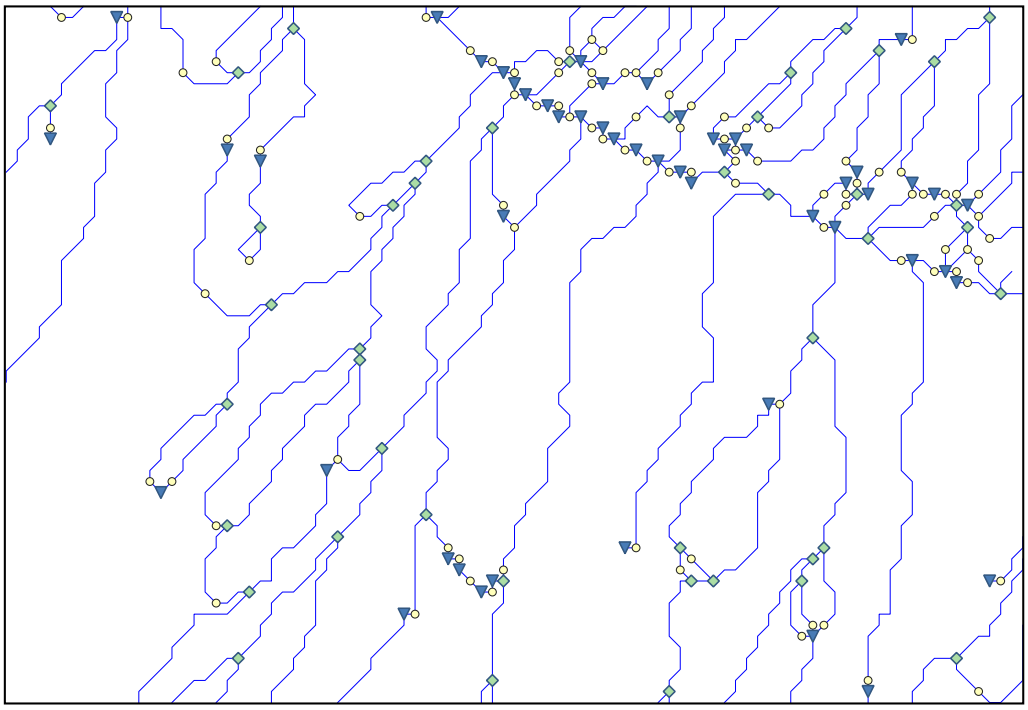

Figure 6: Introduction of confluences in the network. Circles: passes, triangles: pits, diamonds: confluences.

eight possible directions and, especially on long, regular slopes, can lead to a significant shift between the two, as shown in Figure 7. A correction is provided by recording the shift at each step and applying an adjustment when it is larger than one pixel width.

Considering that the path reached a node $p_0$ and the next node from $p_0$ is to be computed, the gradients in each triangle adjacent to $p_0$ are compared and the steepest descent is chosen among them. For a given triangle, its three nodes are noted $p_0$, $p_1$, and $p_2$ of respective elevation $z_0$, $z_1$, and $z_2$. It is considered that both $z_1 < z_0$ and $z_2 < z_0$. If not, the steepest descent is the vector from $p_0$ to the lowest node. In this triangle, a local system is defined such that $p_0 = (0, 0)$, $p_1 = (1, 0)$, and $p_2 = (1, 1)$. The gradient is then given by $\vec{\nabla} z = (z_1 - z_0, z_2 - z_1)$. The path following the steepest descent should intersect $[p_1 p_2]$ at point $p$ of coordinate $(1, p_y)$. Since $\overrightarrow{p_0 p}$ is colinear to $\vec{\nabla} z$,

$$p = \left(1, \frac{z_2 - z_1}{z_1 - z_0}\right) = \left(1, \frac{z_2 - z_0}{z_1 - z_0} - 1\right)$$

The next node in the path will depend on $p_y$. If $p_y \leq \frac{1}{2}$, the path will descend to $p_1$ otherwise to $p_2$. Hence, if $z_2 - z_0 \leq \frac{3}{2}(z_1 - z_0)$, the next node is $p_1$, otherwise it is $p_2$.

When selecting a node $q$, an error given by the distance between $p$ and $q$ is made. Descending on a regular slope, the error accumulates leading to a significant shift in direction. Hence, when the error becomes bigger than 1, meaning that the shift is bigger than

Figure 7: Thalwegs computed along a regular slope: through grid nodes (blue) as in [21], and along the steepest descent (red) as in [16].

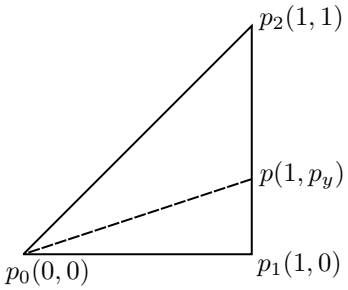


Figure 8: Steepest direction $\overrightarrow{p_0p}$ from $p_0$ to segment $[p_1p_2]$. If $p$ is closer to $p_1$, next thalweg point is $p_1$.

one pixel, a correction is applied. As shown in Figure 9, this produces a path closer to the steepest descent.

## 3.4 Ascending manifold computation

At the end of this stage, each thalweg definition includes a start node (a saddle or a confluence) and an end node (a pit, a confluence, or a saddle). The pit is the virtual pit if the thalweg reaches the border of the domain. Each critical point also includes the list of thalwegs starting and ending at this node. The list of starting thalwegs is always empty at a pit. There is always exactly one thalweg starting at a confluence point and at least two ending at this confluence. In order to facilitate the computation of ascending cells, thalwegs at each critical node and each confluence are arranged in clockwise order, except for the virtual pit. Because the virtual pit is supposedly located on the opposite side of the

Figure 9: Thalwegs computed along a regular slope: along the steepest descent (red) [16] and through grid nodes with correction (blue).

terrain, thalwegs are arranged in counterclockwise order: thalwegs around the virtual pit are seen from below in a clockwise order but counterclockwise when seen from above.

Building an ascending manifold (or hill) is done by starting from a saddle and picking one thalweg and then marching along the thalwegs. When reaching a critical point, the next thalweg is the leftmost thalweg. The process is repeated until reaching back to the first saddle and results in a list of oriented thalwegs. This list includes thalwegs located on the outer boundary of the hill but also some thalwegs forming inner rings and dangling thalwegs ending at a pit. Hence an ascending manifold is defined by three kinds of thalweg: a first list of oriented thalwegs forming the outer boundary, a list of inner rings also defined by oriented thalwegs and a list of dangling thalwegs. In Figure 10, a ring can be seen inside the hill. This ring forms a second hill which is not part of the first hill but delineates another ascending manifold.

## 3.5 Ridge computation

Ridges are computed in the next stage. Since ridges are paths of steepest ascent, the triangulation is modified to facilitate the ascent to the highest node. Diagonals that connect to a saddle or belong to a thalweg path remain fixed while other diagonals are flipped to connect with the highest node in each quadrangle.

In order to prevent any conflict, the position of ridges is constrained by the hills computed previously: each ridge must remain within a hill. Since the initial segment of each ridge was computed during the first stage of the process, the ridge computation process starts by assigning each ridge to a hill. The assignment is done at each saddle by comparing ridges with thalwegs: if a ridge is on the right side of a thalweg then it is located in the hill on the right of this thalweg. Singular cases can occur when a thalweg ends at a

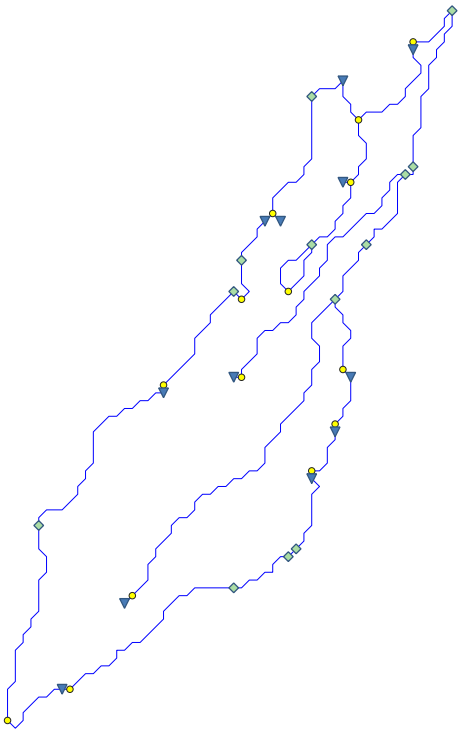

Figure 10: Ascending manifold delineated by thalwegs, including dangling thalwegs and one ring. Triangles, diamonds, and circles mark pits, confluences and saddles connecting thalwegs.

saddle, overlapping a ridge segment. The assignment is done by computing the gradient in triangles adjacent to the ridge. The ridge would then belong to the hill with the highest gradient.

Ridge computation is performed in each hill separately. Since the ridge must remain in its hill, conflicts with thalwegs and merges with other ridges can only occur within the same hill. Hence, ridge computation is performed with the same approach as for thalwegs adding the condition that the next node must always remain inside the ascending manifold and must not cross a dangling thalweg. Ridges may overlap with thalwegs as long as they always remain on the same side. By this way, topological consistency is preserved: as in [7], lines running side by side can be considered to be at an infinitely small distance from each other. Two ridges can even overlap when they are located in two different ascending manifold, overlapping also the thalweg at the boundary, without creating a topological inconsistency. In this case, each ridge is considered on a different side of the thalweg (Figure 11).

Junction nodes are also introduced when two ridges merge in an ascending manifold. The junction can be located on a thalweg but is not connected to it. The junction is considered, like ridges, to be on one side of the thalweg. The junction can even be located on a confluence. In that case, both nodes are maintained in the data structure. They both have the same coordinates but are not topologically related (Figure 12).

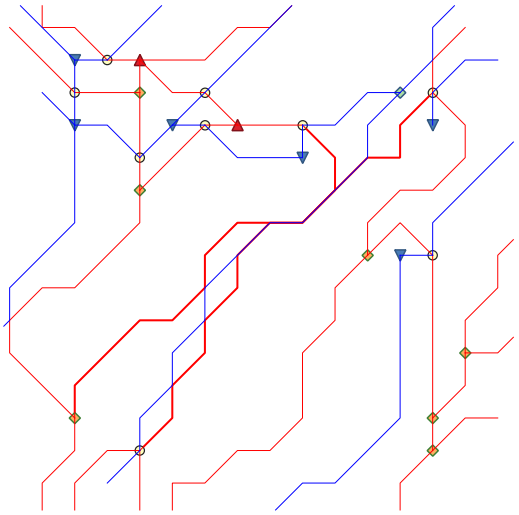

Figure 11: Thick red lines are two ridges in two adjacent ascending manifolds. They are considered to be separated by a thalweg (in blue).

## 3.6 Descending manifold and Morse-Smale cell computation

In a similar way to ascending manifold computation, ridges are organized around each critical point and each junction in a clockwise order. Descending manifolds (or dales) can then be computed by marching along the ridges. A descending manifold is defined by a list of oriented ridges forming the outer boundaries, a list of holes, also formed by oriented ridges, and a list of dangling ridges. While ascending manifolds form proper polygons in the sense that no two edges can cross or overlap, segments of descending manifolds may overlap since they can belong to overlapping ridges. Nonetheless, topological relationships can still be assessed based on the data structure.

Finally, Morse-Smale cells can be computed from the set of critical lines and points. The approach still consists in walking the lines and turning when reaching a node. Here, a path starts from a saddle by choosing a thalweg and walking to the next node. The path is closed when reaching this same saddle from a ridge. When turning around a saddle, the next critical line is the next line in a clockwise order considering both thalwegs and ridges. If a peak or a pit is connected to only one critical line, the same line is taken twice.

Different singularities can occur. First, the cell may not follow the pattern saddle - pit - saddle - peak as mentioned in Section 2. A pit or a peak may not be reached if the line turns before at a confluence or a junction. In some cases, a saddle may connect directly to another saddle. However, the Morse-Smale cell will always be delineated by at least one ridge and one thalweg. It also contains at least two saddles except in cases where the same saddle appears twice.

Similarly to descending manifolds, Morse-Smale cells may not be defined by a regular simple polygon. Especially, when a ridge overlaps a thalweg or when the Morse-Smale cell includes a dangling line, lines will overlap. Figure 12 presents several cases of singular cells. The yellow cell is delineated by a sequence saddle-ridge-saddle-thalweg-pit-thalweg, having neither peak nor junction. The blue cell contains a dangling ridge and is defined by

a sequence saddle-thalweg-confluence-thalweg-saddle-ridge-peak-ridge where the same saddle and the same ridge appear twice. This cell does not include a pit. The pink and the brown cells contain some flat segments where ridges and thalwegs overlap. The pink cell does not contain any peak or pit. The orange cell is a regular cell defined by three ridges and two thalwegs.

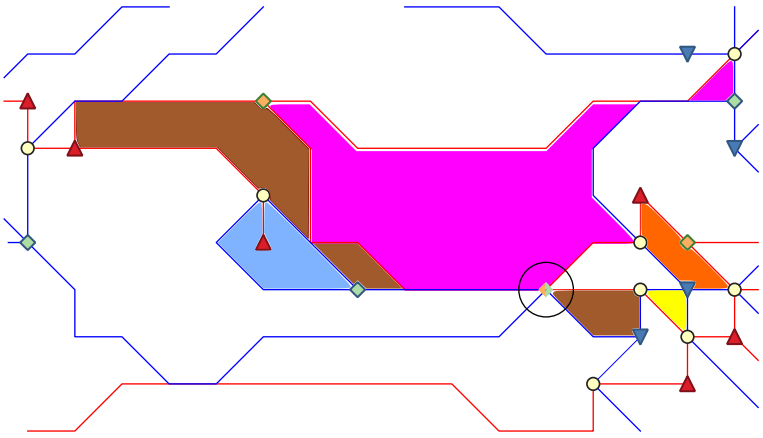

Figure 12: Examples of different configurations of Morse-Smale cells. The diamond in the black circle indicates a junction and a confluence located at the same position

A last singularity is the occurrence of linear cells. An example is shown in Figure 13 where a ridge and a thalweg, that start and end at the same saddles, overlap. This case occurs in flat areas where both saddles are at the same elevation. The Morse-Smale cell is defined by a saddle-ridge-saddle-thalweg sequence and does not include any peak or pit.

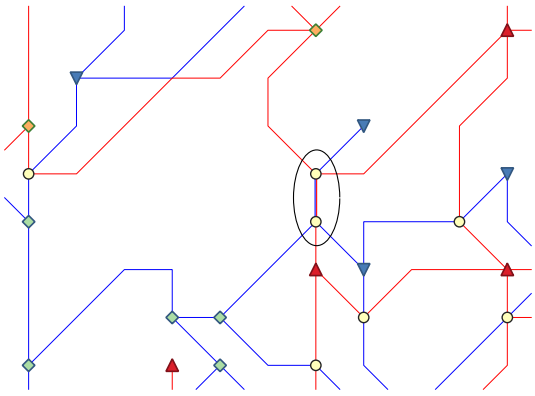

Figure 13: Flat Morse-Smale cell formed by two saddles and overlapping ridge and thalweg.

# 4 Results

## 4.1 Case study

The objective of this section is to assess whether the approach satisfies the different rules presented in section 2. The method was applied to a one-meter resolution raster DTM provided by the ministry of forest, wildlife and parks of Québec province. It represents the catchment area of the "Ruisseau des eaux volées", an area of 105 hectares located in the Montmorency forest, north of Québec city (Figure 14). Elevations vary between 773 m and 978 m. The area includes a regular slope in the south and a higher area in the west and north-west. The outlet of the catchment area is located to the east. The area also includes a forest track visible in the center of the figure.

Figure 14: Hillshade view of the Ruisseau des eaux volées catchment area.

The DTM was built from an airborne lidar point cloud. The vertical accuracy of the point cloud is of 15 centimeters with a density of 4 points per square meter. The raster DTM was interpolated by binning. The area does not include water surfaces.

The method was implemented in Python. The code is openly available at https://github.com/ericguilbert/SurfaceNetwork. Singularities where two neighboring pixels share the same height were treated by imposing a lexicographical order [21]. Since all nodes can be defined by their row and column index in the raster, all grid coordinates are stored as integer values, reducing the storage space and avoiding rounding errors from floating point representation. It also fastens the process since coordinates can easily be indexed in a hash table when looking for critical points or lines.

## 4.2 Critical points

The first step in the process was to compute saddles. The results were compared with two other approaches: one based on [21] and the second on [16]. The first method was applied in [9] on the same DTM. Since the authors wanted to extract only the thalwegs, the triangulation was done by connecting each node to the lowest node in a grid cell. The second method was implemented in Python considering the DTM as a piecewise linear surface. Table 1 summarizes the number of critical points computed with each method. Saddles are counted with their multiplicities and the virtual pit is included in the number of pits.

| | Linear [9] | Piecewise bilinear [16] | This approach |
|---|---|---|---|
| Saddles (with multiplicity) | 15209 | 21671 | 20688 |
| Pits | 4615 | 11275 | 10110 |
| Peaks | 10596 | 10597 | 10580 |

Table 1: Number of critical points computed with three methods.

As shown in Table 1, the method presented in [9] yielded less critical points than this approach. This is the result of the triangulation. In the first approach, the triangulation was performed in order to avoid spurious pits and obtain longer thalwegs that may better match with the streams. This also explains why the first approach gave less than half the number of pits. Furthermore, both methods observe the Euler-Poincaré characteristic.

On the contrary, computation on a bilinear surface provides more critical points. The method always provides the highest possible number of pits and peaks because they are computed on four neighbors while, on a triangulated grid, a node can have up to eight neighbors. The number of pits in the new method is also smaller because the triangulation also tends to limit the number of pits.

Bilinear surface approaches do not satisfy the Euler-Poincaré characteristic. While there are more saddles than in other methods, it does not mach the number of pits and peaks. The reason is due to cases where a saddle is on a grid edge. Norgard and Bremer [16] do not consider singular points but consider that pixels are all at different heights so that saddles cannot be located on the edge or along the diagonals of a grid cell. In practice, it is not possible to enforce such a constraint, especially on high resolution DTMs. Disregarding singularities leads to a smaller number of saddles since only saddles located at grid nodes and inside grid cells are detected. In the present implementation, saddles along edges were also detected but further study is required to check whether a higher multiplicity should be considered, and whether a same saddle should belong to two neighboring cells.

A comparison was done on saddle positions obtained by the bilinear approach and this approach by measuring the distance between saddles. Most of the extra saddles correspond to cases where two bilinear saddles were located within one meter, in two neighboring grid cells or along an edge, with some cases where two saddles were detected along the same edge. Two bilinear saddles were located at more than two meters from any linear saddle. This issue corresponds to cases where neighboring nodes are at the same elevation. On the opposite, 113 linear saddles were located at more than 2 meters from any bilinear saddles. These saddles were located on the boundary of the domain where one neighbour along a diagonal was located outside the domain and considered below any other points.

Distribution of critical points is irregular and follows the terrain morphology. Figure 15 shows saddles over a part of the terrain. Only saddles are shown for clarity but other critical points follow a similar distribution. Regions with higher density of points correspond to more or less flat areas. On the left, points are located in a sort of plateau on the border of the catchment area while in the center, they are located in a lower, marshy area along the main stream. On the opposite, saddles along slopes are sparser. This is due to the fact that saddle detection is sensitive to small variations in elevation. Hence, many variations in flat areas may be seen as noise since the height variation in these places can be below the vertical accuracy. Finally, critical points are also aligned along streams and forest tracks.

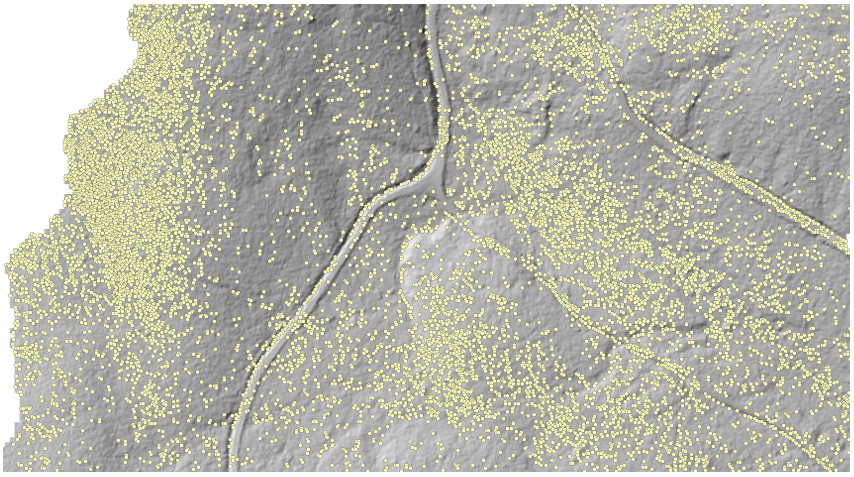

Figure 15: Saddle distribution over the terrain.

Streams are marked by series of pits and saddles while forest tracks are marked by pits and peaks on either side marking the bank of the track (Figure 16).

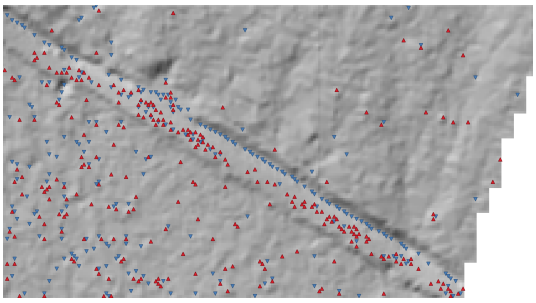

Figure 16: Pits and peaks along a road track.

## 4.3 Critical line and manifold computation

Critical lines are presented in Figure 17 for a total of 53073 thalwegs and 53435 ridges. The numbers of ridges and thalwegs are not equal because of new lines created at each merge. As such, 12066 confluences and 12428 junctions were added into the data structure.

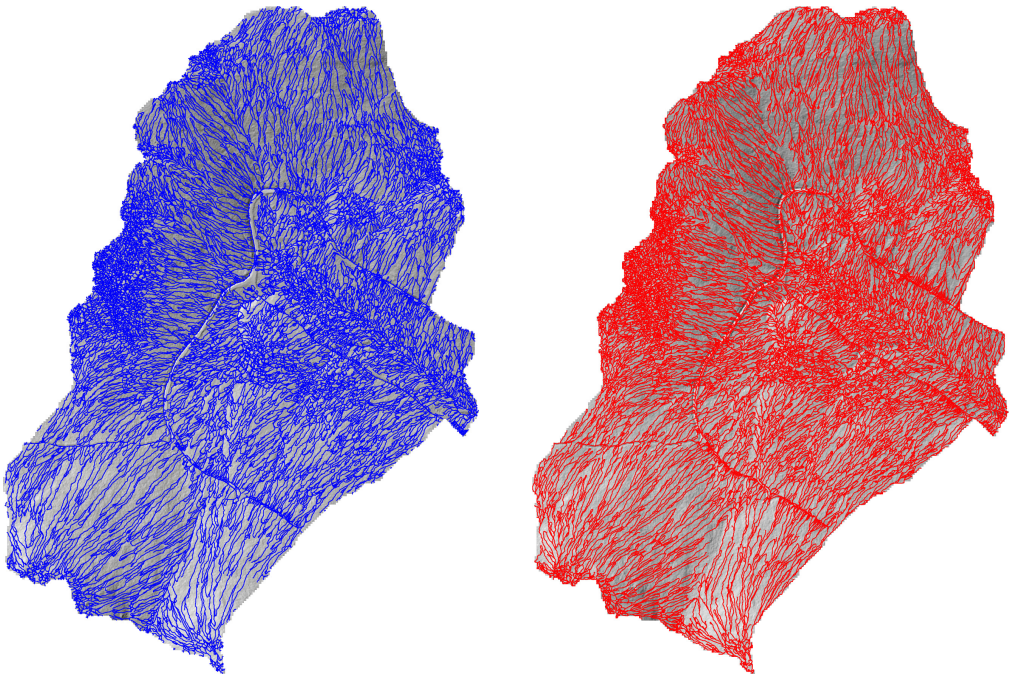

Figure 17: Thalwegs (blue) and ridges (red) computed over the whole terrain.

Results logically follow those obtained with saddles and there are more short lines in flat areas such as in the upper part on the west and along the main stream in the center. A geormorphometric classification approach would yield a different result since the classification depends on threshold values. These flat areas would mainly be identified as plane areas, mainly because the surface network records details that would be considered as noise by the user. However, the number and type of segments can vary with parameter settings.

The south part of the terrain corresponds to a regular slope where most thalwegs are ruts. A segmentation method would classify most of the area as planes and would not extract continuous thalwegs from the top to the bottom of the slope since the ruts do not carve the terrain sufficiently. By initiating thalwegs from saddle points and following the steepest descent, the surface network is able to compute a path and connect the thalwegs to the bottom of the slope.

Consistency of the network was validated on ascending and descending manifolds in two ways. First, each thalweg or ridge must remain in its descending or ascending manifold and all thalwegs (resp. ridges) reaching a pit or a confluence (resp. a peak or a junction) must belong to the same descending (resp. ascending) manifold. Second, each ascending (resp. descending) manifold must contain only one peak (resp. pit). One can note that,

if in the Morse-Smale theory, a Morse-Smale cell always contains a pit and a peak on its boundary, this is not necessarily the case in this approach as shown in Figure 12. However, following a thalweg or a ridge would always lead to the pit or the peak in the dale or hill that contains the Morse-Smale cell.

It took about 1 min 40s to process the whole terrain. Critical point computations are the most demanding parts of the method because they impose a scan of the whole raster. Saddles took 30 seconds to compute while pits and peaks took about 10 seconds each. Indeed, diagonalizations performed after saddle computation and thalweg computation also require a raster scan and took about 10 seconds each. Thalweg and ridge computation took respectively 7 and 19 seconds. The latter is more expensive because of the consistency check at each step which sometimes requires a line in polygon test. Manifold computations took only about one second for each manifold. Their computation is based only on the data structure and computation on the DTM is required only in some singular cases where the relative position of two lines depends on the gradient.

# 5 Conclusion

The paper introduced a new approach for computing a surface network over a raster DTM. In comparison to other methods, a special focus was brought on the robustness of the method in order to ensure the topological consistency of the result. For that purpose, the network node and edge positions are restricted to edges and pixels of the raster. The new method contributed in the following ways:

- The number of saddles is maximized to capture all details on the terrain;
- A correction is brought to line computation to remain closer to the gradient;
- Topological constraints are added to ridge computation to prevent any conflict with thalwegs;
- Ascending and descending manifolds are defined and included in the data structure.

The method guarantees the absence of crossing between ridges and thalwegs and the topological correctness of the structure. Explicit and coherent relationships between lines allow for the robust construction of ascending and descending manifolds.

The surface network provides information about the terrain morphology and includes salient features that summarize the shape and main characteristics of terrain elements. The network can be computed on any kind of terrain model however it relies heavily on the detection of saddles. A large number of saddles is required to capture all features on the terrain exhaustively. Hence this approach is relevant mainly for high-resolution DTM. While many critical lines and saddles may be considered as spurious details, computing saddles on a low-resolution or smoothed DTM may lead to missing some critical lines.

Furthermore, the computation time on large DTMs remains reasonable. Most of it is spent on the computation of critical points and on the triangulation. Further operations such as network simplification and analysis can be performed on the surface network only, without requiring the DTM. These operations are therefore extremely fast since the surface network provides a compact representation of the terrain with an explicit topological structure.

Based on these results, the surface network can be a significant tool for the analysis of large DTMs. As a consequence, future works will focus at making use of the surface

network for the analysis of terrain features. Two directions are considered. First is the construction of the drainage system from the thalweg network by assuming that streams should match with thalwegs and there should be a relationship between ridges and watersheds, although ridges are not exactly equivalent to watersheds [1]. Second is the analysis of landforms. Existing works have proposed that elementary forms on a terrain can be delineated by lines of discontinuity [15] or that landforms can be recognized from their salient features [10]. Critical points and lines can be such salient elements and the surface network provides the necessary relationships for the computation of geomorphometric features. For example, thalwegs and ridges have been used to identify canyons [4] and delineate landslide systems [22]. Since these features may be computed solely from the surface network, not requiring the DTM and as such with less dependence to prescriptive threshold values, it may provide a fast and reliable analysis tool.

## Acknowledgments

The author is supported by a Natural Sciences and Engineering Research Council of Canada Discovery grant. The DTM was provided by the ministry of forest, wildlife and parks of Québec. The following students contributed to earlier versions of the code through their master projects: Sébastien Gerzlak (implementation of the original method by Takahashi et al.), and Apollinaire Valentin and Alexis de Chavagnac (topological constraints on ridge computation).

## References

[1] BRÄNDLI, M. Hierarchical models for the definition and extraction of terrain features. In *Geographic Objects With Indeterminate Boundaries* (1996), Taylor & Francis, pp. 257–270.

[2] BREMER, P.-T., EDELSBRUNNER, H., HAMANN, B., AND PASCUCCI, V. A topological hierarchy for functions on triangulated surfaces. *IEEE Transactions on Visualization and Computer Graphics 10*, 4 (2004), 385–396. doi:10.1109/TVCG.2004.3.

[3] ČOMIĆ, L., DE FLORIANI, L., MAGILLO, P., AND IURICICH, F. *Morphological Modeling of Terrains and Volume Data*. SpringerBriefs in Computer Science. Springer, 2014. doi:10.1007/978-1-4939-2149-2.

[4] CORTÉS MURCIA, A. C., GUILBERT, E., AND MOSTAFAVI, M. A. An object based approach for submarine canyon identification from surface networks. In *Short paper proceedings of the Ninth international conference on GIScience* (2016). doi:10.21433/B31105h6f9b1.

[5] DANOVARO, E., DE FLORIANI, L., MAGILLO, P., MESMOUDI, M. M., AND PUPPO, E. Morphology-driven simplification and multiresolution modeling of terrains. In *The 11th International Symposium on Advances in Geographic Information Systems* (2003), E. Hoel and P. Rigaux, Eds., ACM press, pp. 63–70. doi:10.1145/956676.956685.

[6] DE KOK, T., VAN KREVELD, M., AND LÖFFLER, M. Generating realistic terrains with higher-order Delaunay triangulations. *Computational Geometry 36* (2007), 52–65. doi:10.1016/j.comgeo.2005.09.005.

[7] EDELSBRUNNER, H., HARER, J., AND ZOMORODIAN, A. Hierarchical morse-smale complexes for piecewise linear 2-manifolds. *Discrete & Computational Geometry 30* (2003), 87–107. doi:10.1007/s00454-003-2926-5.

[8] FOWLER, R. J., AND LITTLE, J. J. Automatic extraction of irregular network digital terrain models. *ACM SIGGRAPH Computer Graphics 13*, 2 (1979), 199–207. doi:10.1145/965103.

[9] GUILBERT, E., JUTRAS, S., AND BADARD, T. Thalweg detection for river network cartography in forest from high-resolution lidar data. In *ISPRS - International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences* (2018), vol. XLII-4, pp. 241–247. doi:10.5194/isprs-archives-XLII-4-241-2018.

[10] GUILBERT, E., AND MOULIN, B. Towards a common framework for the identification of landforms on terrain models. *ISPRS International Journal of Geo-information 6*, 1 (2017), 12. doi:10.3390/ijgi6010012.

[11] GUILBERT, E., MOULIN, B., AND CORTÉS MURCIA, A. A conceptual model for the representation of landforms using ontology design patterns. In *ISPRS Annals of Photogrammetry, Remote Sensing and Spatial Information Sciences* (2016), L. Halounova, S. Li, V. Šafář, M. Tomková, P. Rapant, K. Brázdil, W. J. Shi, F. Anton, Y. Liu, A. Stein, T. Cheng, C. Pettit, Q.-Q. Li, M. Sester, M. A. Mostafavi, M. Madden, X. Tong, M. A. Brovelli, K. HaeKyong, H. Kawashima, and A. Coltekin, Eds., vol. III-2, pp. 15–22. doi:10.5194/isprsannals-III-2-15-2016.

[12] JASIEWICZ, J., AND STEPINSKI, T. F. Geomorphons—a pattern recognition approach to classification and mapping of landforms. *Geomorphology 182* (2013), 147–156. doi:10.1016/j.geomorph.2012.11.005.

[13] MAGILLO, P., DANOVARO, E., DE FLORIANI, L., PAPALEO, L., AND VITALI, M. Extracting terrain morphology – a new algorithm and a comparative evaluation. In *Proceedings of the Second International Conference on Computer Graphics Theory and Applications - Volume 2: GRAPP* (2007), INSTICC, SciTePress, pp. 13–20. doi:10.5220/0002076200130020.

[14] MANGAN, A. P., AND WHITAKER, R. T. Partitioning 3D surface meshes using watershed segmentation. *IEEE Transactions on Visualization and Computer Graphics 5*, 4 (1999), 308–321. doi:10.1109/2945.817348.

[15] MINÁR, J., AND EVANS, I. S. Elementary forms for land surface segmentation: the theoretical basis of terrain analysis and geomorphological mapping. *Geomorphology 95* (2008), 236–259. doi:10.1016/j.geomorph.2007.06.003.

[16] NORGARD, G., AND BREMER, P.-T. Robust computation of morse-smale complexes of bilinear functions. *Computer Aided Geometric Design 30* (2013), 577–587. doi:10.1016/j.cagd.2012.03.017.

[17] PEUCKER, T. K., AND DOUGLAS, D. H. Detection of surface-specific points by local parallel processing of discrete terrain elevation data. *Computer graphics and image processing 4* (1975), 375–387. doi:10.1016/0146-664X(75)90005-2.

[18] SCHNEIDER, B. Surface networks: extension of the topology and extraction from bilinear surface patches. In *Proceedings of the 7th international conference on GeoComputation* (2003). doi:10.1.1.89.9824.

[19] SCHNEIDER, B., AND WOOD, J. Construction of metric surface networks from raster-based DEMs. In *Topological data structures for surfaces. An introduction to geographical information science* (2004), S. Rana, Ed., Wiley, pp. 53–70. doi:10.1002/0470020288.ch4.

[20] SHARY, P., SHARAYA, L., AND MITUSOV, A. The problem of scale-specific and scale-free approaches in geo-morphometry. *Geografia Fisica e Dinamica Quaternaria 28* (2005), 81–101.

[21] TAKAHASHI, S., IKEDA, T., SHINAGAWA, Y., KUNII, T. L., AND UEDA, M. Algorithms for extracting correct critical points and constructing topological graphs from discrete geographical elevation data. *Computer Graphics Forum 14* (1995), 181–192. doi:10.1111/j.1467-8659.1995.cgf143_0181.x.

[22] VALIANTE, M. *Integration of object-oriented modelling and morphometric methodologies for the analysis of landslide systems*. unpublished phd thesis, Sapienza Università di Roma, 2020.

[23] VINCENT, L., AND SOILLE, P. Watershed in digital spaces: an efficient algorithm based on immersion simulation. *IEEE Transactions on Pattern Analysis and Machine Intelligence 13*, 6 (1991), 583–598. doi:10.1109/34.87344.

[24] WOLF, G. W. Topographic surfaces and surface networks. In *Topological data structures for surfaces. An introduction to geographical information science* (2004), S. Rana, Ed., Wiley, pp. 15–29. doi:10.1002/0470020288.ch2.

[25] WOOD, J. *The geomorphological characterisation of digital elevation models*. unpublished PhD thesis, University of Leicester, 1996.