Drainage Paths derived from TIN-based Digital Elevation Models

Henrique R. A. Freitas, Sergio Rosim, João R. F. Oliveira, Corina C. Freitas

National Institute for Space Research – Image Processing Division
PO BOX 515 – 12.227-010 – São José dos Campos – SP – Brazil

{henrique,sergio,joao,corina}@dpi.inpe.br

Abstract. *Triangular Irregular Networks efficiently define Digital Elevation Models that represent terrain surfaces and drainage paths can be calculated from these terrain models. This paper describes a method for calculating drainage paths from a triangulated irregular terrain model that was obtained from contour lines and points. Contour lines crossed by triangles edges and flat areas, which prevent path continuity, are removed by edge rotations and by inserting interpolated points into the triangulation, respectively. Drainage paths are connected by processing the triangles with an associated priority. Results achieved are consistent with an available drainage network and with real-world terrain information from a RapidEye image.*

1. Introduction

Digital Elevation Models (DEM) can be defined by Triangular Irregular Networks (TIN) in order to represent terrain models. A TIN is a very efficient terrain model as the density of information can vary from region to region in a way that more points are included where there is more elevation variation while fewer points are necessary in regions of less elevation variation avoiding data redundancy.

The triangulation is calculated from a set of points where each point is defined by its x, y coordinates on the plane and an elevation z. These points contain the main features and characteristics of the terrain and the most common triangulation used is the Delaunay triangulation that maximizes the minimum angle among all triangles thus creating less skinny triangles [De Berg et al. 2008].

In this work, all the points used as input for calculating the triangulation define contour lines and elevation points, and as the original Delaunay triangulation could produce edges crossing these contour lines, it turns out to be necessary to apply a further procedure that removes these intersections in order to modify connections between points that could result in wrong terrain features. This procedure defines a Constrained Delaunay Triangulation [Zhu and Yan 2010] where every contour line is considered as a restriction line.

Besides intersections of triangles edges and contour lines, another problem that may arise when using a TIN as a terrain model is the existence of flat triangles. These triangles define flat areas where it is not possible to determine a flow direction because all three points or vertices of each triangle have the same elevation. This problem is solved by the insertion of new points into the triangulation with interpolated elevation values in order to guarantee that every new triangle created, after a re-triangulation with these new points, has a defined flow direction and drainage paths have no discontinuities.

DEMs are very important in many research areas and they have useful applications especially in Hydrology where drainage patterns calculated from a DEM are essential in the understanding of many hydrologic processes of nature. This paper focuses on a method for calculating drainage paths in a TIN where the flow direction in a triangle is determined by a gradient vector. Results are consistent with real-world terrain information and with an available drainage network indicating that a TIN is an appropriate alternative structure for terrain modeling and hydrologic applications.

The paper is organized as follows. Next section mentions works already developed and the motivation for the theme of drainage paths derived from TIN. Section 3 contains the methodology including a description of the Constrained Delaunay Triangulation, the procedure for removing flat areas and the gradient method for tracing drainage paths. In section 4, drainage paths are compared to an available drainage network of the analyzed region where similarities between them indicate that these drainage paths represent good approximations and are also consistent with water flow patterns. Computational times took by the procedures are also given. Section 5 presents the conclusions as well as suggestions for future work. References are placed at the end.

2. Related Work and Motivation

Some authors investigated and developed techniques to calculate drainage paths directly from TIN terrain models. Many important concepts were described by [Jones et al. 1990] considering the flow direction in each triangle defined by the gradient of the plane that contains it. Another approach was developed by [Silfer et al. 1987] determining how water should be routed across the surface of a TIN distinguishing from two different conditions between TIN facets. More recently, a trickle path procedure by [Tsirogiannis 2011] traces a sequence of edges and vertices determined from intersections between points and terrain features.

The above-mentioned techniques can be added as hydrology-specific functionalities in Geographic Information Systems (GIS) as these systems are able of storing and processing a wide range of georeferenced data. Many GIS applications that process terrain models have limited capabilities when it comes to flow modeling in TIN because they require the design of more robust data structures and algorithms in order to solve problems of computational geometry so that this type of functionality is less developed than for the most common and simple DEM defined by regular grids.

TIN datasets used for terrain modeling and analysis raise many challenges in the development of efficient algorithms that can process and extract useful results from them because their use usually involves complex tasks. Computing flow-related structures on TIN such as drainage paths can present a worst-case complexity of $\Theta(n^3)$ when considering the whole river network with n triangles where this complexity is measured by the number of segments of all paths [Agarwal et al 1996].

This work addresses the problem of automatically calculating drainage paths in a TIN obtained from a dense set of points after removing inconsistencies such as contour lines crossing triangles edges and flat areas that can occur when using triangulated structures as terrain models. The aforementioned works do not define specific procedures for removing flat areas comprised of several flat adjacent triangles branching in different directions and do not specify how drainage paths can be connected in order to make it possible to calculate accumulated flows and drainage networks.

3. Methodology

The set of triangles that defines a TIN is a good approximation to the irregularities inherent to a terrain structure. This structure can be characterized by surface-specific points and lines representing terrain features that are considered as the backbone of the surface [Fowler and Little 1979]. In the present work, a TIN is calculated by a Constrained Delaunay Triangulation algorithm from a set of points that defines contour lines and elevation points.

3.1. Constrained Delaunay Triangulation

There are several different triangulations that can be calculated from the same set of points and a good approximation used for terrain modeling is given by the Delaunay triangulation [De Berg et al. 2008]. The main property of the Delaunay triangulation is that every triangle defines a circle through its three vertices that does not contain any other point of the set inside it. This property is also considered as criteria for calculating the triangulation [Tsai 1993] which indicates that a Delaunay triangulation consists of more equiangular triangles and therefore the minimum angle among all triangles is maximized. Figure 1 shows a Delaunay triangulation calculated from a set of points and its criteria for a circle defined by the three vertices of a triangle.

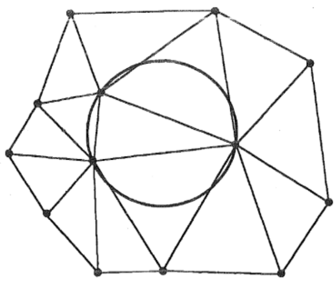

Figure 1. Delaunay triangulation criteria (modified from [Jones et al. 1990])

Many algorithms that calculate the Delaunay triangulation can be found in the literature. Some of them are: Bowyer-Watson [Bowyer 1981, Watson 1981], Incremental [Guibas et al. 1992, De Berg et al. 2008], Divide-and-Conquer [Cignoni 1998], Fortune [Fortune 1987] and Brute Force [O'Rourke 1998]. For the present work, the Incremental algorithm was used because its time complexity is *O(n log n)* [De Berg et al. 2008] where *n* is the number of points. A C++ implementation was developed because the structures and procedures from the source code could be easily modified in the future so that they work with the Terralib library [Câmara et al. 2000].

The algorithm works by initially determining a triangle that contains the set of points all inside it, then inserting each point one at a time, and when a point is inserted into the triangulation, a new triangulation is calculated with possible local changes in the current one. The algorithm also defines a tree structure for storage and search of the triangles and this structure is modified after every insertion of a point. When a triangle contains the inserted point it is divided into new triangles and this triangle division is reflected on the tree structure in a way that old and new triangles are connected by a hierarchy link in the tree. At the end, all Delaunay triangles are leaf nodes of the tree and the initial triangle together with all its incident edges are discarded. The tree height is proportional to $\log(n)$ where n is the number of points, which determines that the search for a triangle that contains some point can be computed in logarithmic time.

It is noteworthy that if the points used as input for calculating the triangulation do not have any kind of specific connection between them, the Delaunay triangulation suffices to define a TIN as a terrain model. However, if the set of points define contour lines, as it is the case in this work, every segment of a contour line must be considered as a restriction line that cannot be crossed by a triangle edge otherwise that intersection would create inconsistencies with the terrain surface. In order to solve this problem, an initial Delaunay triangulation is calculated and a further procedure removes the intersections between contour lines segments and triangles edges. Figure 2 shows triangles edges (dashed lines) that intersect contour lines segments (solid lines) and the resulting triangulation after removing these intersections.

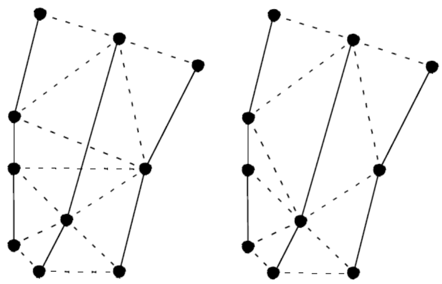

Figure 2. Triangulations before and after removing intersections (taken from [Eastman 2001])

The procedure for removing intersections initially detects for every contour line segment a triangle connected to one segment endpoint such that its edge opposite to the endpoint intersects the segment. This triangle is then processed by a search procedure that verifies each of its adjacent triangles and checks if there is one triangle not processed yet that also intersects the segment. If another triangle is found, this adjacent triangle is tested similarly and the search continues until no more intersections are found, that is, when the other segment endpoint is reached.

All triangles found in the search process are inserted into a queue structure and every pair of triangles in the queue is processed in order to remove all intersections. If an edge of a triangle in the queue intersects the segment (not in a vertex), then its adjacent triangle in the queue also intersects the segment, so their common edge is rotated and the new modified triangles are inserted back into the queue for a further verification. This procedure continues as long as there are intersections between triangles edges and contour lines segments.

3.2 Flat Areas

Flat areas are not common over terrain surfaces, except in plateau areas that consist of relatively flat terrains, but terrain models are prone to such inconsistencies as they are near-to-real approximations. TIN used as terrain models can present flat areas whenever all three vertices of a triangle have the same elevation. This situation must be avoided as the flow direction over a flat area is undefined which turns out to be a major problem in hydrologic computations.

Every flat triangle is removed by first detecting its critical edges that are identified by two cases: a) an edge that connects two non-consecutive points in the same contour line; b) an edge connecting two points in different contour lines but of equal elevation. These two cases are illustrated in figure 3 where solid lines are contour lines and dashed lines are triangles edges with critical edges in red.

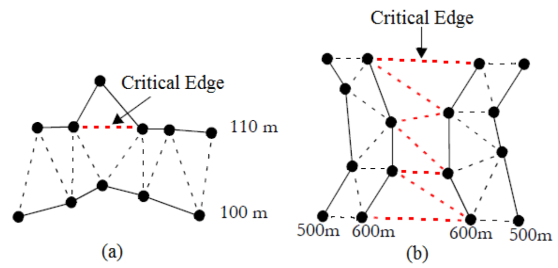

Figure 3. Flat triangles and critical edges (modified from [Eastman 2001])

The procedure for removing flat triangles inserts critical points into the triangulation that are placed exactly in the middle of critical edges assigning to each of these critical points a linearly interpolated elevation value that is calculated after a path of flat triangles is determined by a search process.

Initially, every search process defines a path by starting at corner triangles which are triangles that contain one critical edge and two edges that are non-critical, that is, either contour lines segments or edges connecting points of different elevations. The point that connects the two non-critical edges is defined as the initial point for interpolation. This search process continues always going from the current triangle to one of its adjacent triangles that share a common critical edge, following through the critical edge that contains the closest critical point in relation to the last point considered for interpolation. The search terminates when there are no more adjacent triangles to be visited (and the last point is a critical point) or the current triangle is another corner triangle (in this case, the last point has a defined elevation). All critical points found in the search process have their elevation values linearly interpolated between the elevation values of the initial and final points.

Branches found in the search process (triangles with three critical edges) are processed after the interpolation procedure has assigned an elevation value to every critical point included in the path. The search process repeats once again beginning at every branching triangle found and the same procedure is executed until all critical points have been assigned an interpolated elevation value. Finally, these critical points are inserted into the triangulation and the areas around them are then re-triangulated.

This procedure is illustrated in figure 4 where contour lines segments are dark lines with their endpoints in light green, triangles edges are in red and the critical points in magenta. Flat triangles are in light blue and corner triangles in yellow. In this example, the initial point used for interpolation is circled in red at the top corner triangle and the final point is also circled in red at the bottom. The path from the initial point to the final point following through critical edges is in dark green with branches in cyan.

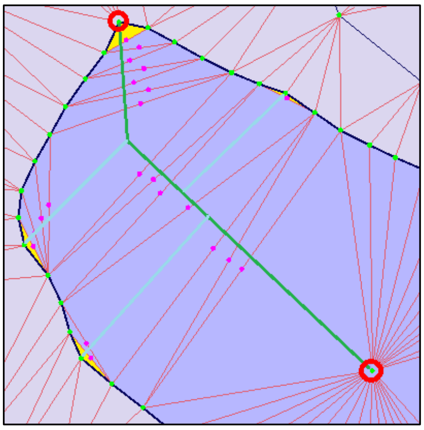

Figure 4. Paths for interpolation of critical points

As mentioned before, a linear interpolation of the critical points is performed considering both initial and final points found in the path. If the final point elevation is not defined (in the case of a critical point) then the elevation variation from the contour line that encloses the flat area in relation to its neighboring contour lines indicates whether the interpolated elevation values to be assigned to every critical point should be increasing (neighboring contour lines values are lower) or decreasing (neighboring contour lines values are higher).

3.3 Drainage Paths

Terrain models represented by TIN consist of several adjacent triangles of different sizes and shapes. Each triangle defines a plane surface that passes through its three vertices and drainage paths can be calculated from any starting point in a triangle following the path of steepest descent given by the plane gradient [Jones et al. 1990]. A plane equation and its coefficients are determined by the equations below, where each (x_i, y_i, z_i) is a triangle vertex with index i = 1,2,3:

$$Ax + By + Cz + D = 0 \tag{1}$$

$$A = y_1(z_2 - z_3) + y_2(z_3 - z_1) + y_3(z_1 - z_2) \tag{2a}$$

$$B = z_1(x_2 - x_3) + z_2(x_3 - x_1) + z_3(x_1 - x_2) \tag{2b}$$

$$C = x_1(y_2 - y_3) + x_2(y_3 - y_1) + x_3(y_1 - y_2) \tag{2c}$$

$$D = -Ax_1 - By_1 - Cz_1 \tag{2d}$$

Writing the plane equation (1) with z as a function of x and y, then calculating the negative gradient of this function by partial derivatives, the direction of steepest descent projected onto the xy plane is defined by equation (4) which determines the flow direction from a point in a triangle.

$$z = f(x, y) = -\left(\frac{A}{C}x + \frac{B}{C}y + \frac{D}{C}\right) \tag{3}$$

$$-\nabla f = -\left(\frac{\partial f}{\partial x}\boldsymbol{i} + \frac{\partial f}{\partial y}\boldsymbol{j}\right) = \frac{A}{C}\boldsymbol{i} + \frac{B}{C}\boldsymbol{j} \tag{4}$$

Every drainage path begins at a starting point in a triangle always following the direction given by the gradient vector of each triangle. When tracing a drainage path, different situations can occur regarding the intersections between gradient vectors and triangles edges. If the gradient vector of a triangle intersects one of its edges and the gradient vector of the adjacent triangle opposite to that edge points back to the first triangle, thus forming a channel edge, then the drainage path should continue along the edge towards the vertex of lowest elevation otherwise the path continues across the adjacent triangle.

When the intersection is exactly in a triangle vertex, then all the edges and triangles incident to that vertex are checked in order to find the lowest elevation point reached from the vertex. Each edge is first verified if it is a channel edge (both gradient vectors of the adjoining triangles by the edge point to each other) and then if the other vertex of the edge has a lower elevation. Triangles are tested by checking if there is an intersection between its gradient vector based at the current vertex and its edge that is opposite to the vertex (the gradient vector lies between the other two edges) and if this intersection has also a lower elevation. After the lowest elevation point has been found, the drainage path continues through an edge to another vertex (in the case of a channel edge) or across a triangle and the process is repeated.

Part of a drainage path can be visualized in figure 5 which contains interpolated elevation values on each plane and the path that every gradient vector follows across triangles and edges beginning at the starting point *a*.

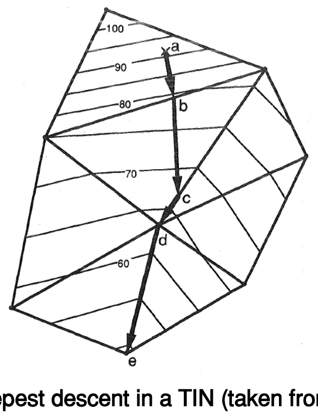

Figure 5. Path of steepest descent in a TIN (taken from [Jones et al. 1990])

The procedure described for constructing drainage paths can be applied by selecting any point as the starting point. In this work, the points selected as starting points are the triangles centroids which approximately represent the elevation of the triangles. Every starting point has its elevation considered as a priority value associated to the point defining the order in which all the points will be processed in the calculation of drainage paths. This approach indicates that it is possible to delineate potential drainage patterns by calculating drainage paths beginning at these starting points ordered from highest to lowest elevations. Another important aspect of this procedure is that when a drainage path is being traced and it reaches a triangle where another path has already been defined, then the current path is connected to the existing path. This procedure terminates after every starting point has been processed and all drainage paths have been connected.

4. Results

All results were obtained from contour lines and elevation points of an area in the city of São José dos Campos – Brazil in a geographic region with bounds 396000.0m-427400.0m West and 7421000.0m-7445000.0m South given in UTM coordinates and SAD69 projection. These UTM coordinates correspond to the geographic coordinates ranging from -46.017 to -45.709 in longitude and from -23.317 to -23.102 in latitude. The entire dataset used as input is from a database named "Cidade Viva" that is updated every 6 months and made publicly available since 2003 by the city's Geoprocessing Service of the Urban Planning Department in a format that is easily imported by a GIS.

As the main focus of this work is to calculate drainage paths from a triangulated terrain model, a TIN was defined by the Constrained Delaunay Triangulation detailed in section 3.1 and the terrain model was calculated from ~20 m *xy* resolution contour lines and elevation points with neighboring contour lines having a 5 m elevation difference represented by approximately 200000 points. Flat areas and drainage paths were processed by the procedures described in sections 3.2 and 3.3.

Figure 6 shows in blue the drainage network available from the previously mentioned database over a RapidEye image of 5 m spatial resolution for part of the total region. This drainage network is considered as the reference drainage for comparison with the drainage paths. The dashed rectangle indicates an area that is shown next in figure 7.

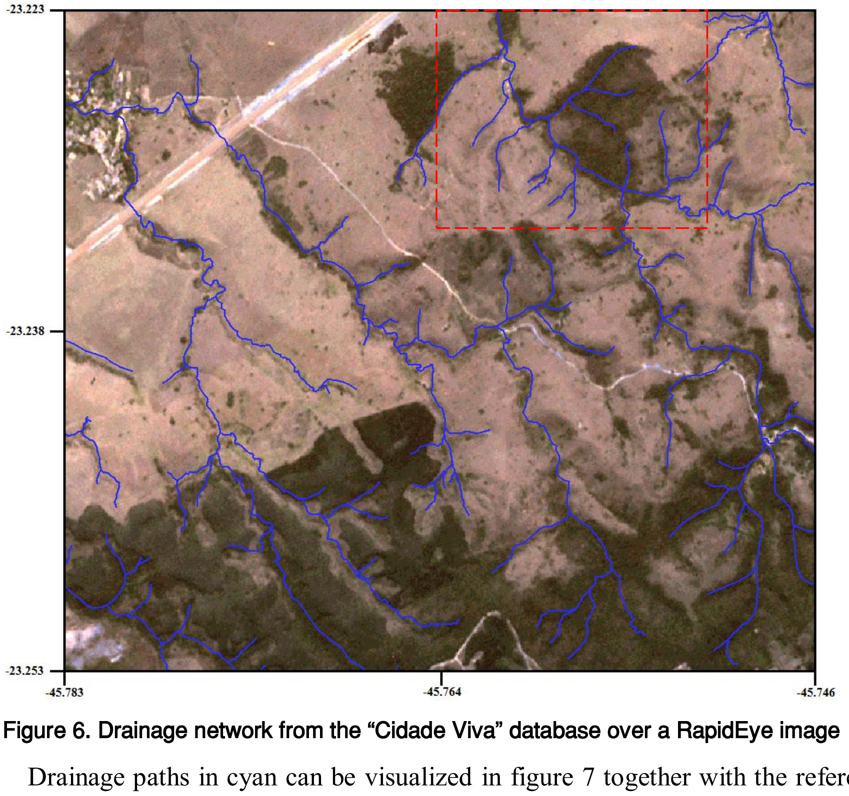

Figure 6. Drainage network from the "Cidade Viva" database over a RapidEye image

Drainage paths in cyan can be visualized in figure 7 together with the reference drainage network for the small region took from the upper-right part of figure 6 that is highlighted by the dashed rectangle. It can be noticed that these drainage paths approximately converge to the drainage network, thus forming drainage patterns very close to the real hydrologic processes governed by the terrain surface.

Discrepancies between the two drainage patterns may be due to the precision of the input data, i.e., the contour lines and elevation points, as it can change the direction of flow from triangle to triangle. Discontinuities in the drainage paths occur by the presence of pits that are located at vertices where flow does not follow through an edge or a triangle because the gradient conditions are not satisfied. Once again, a dashed rectangle indicates another area which is detailed in figure 8 that follows in sequence.

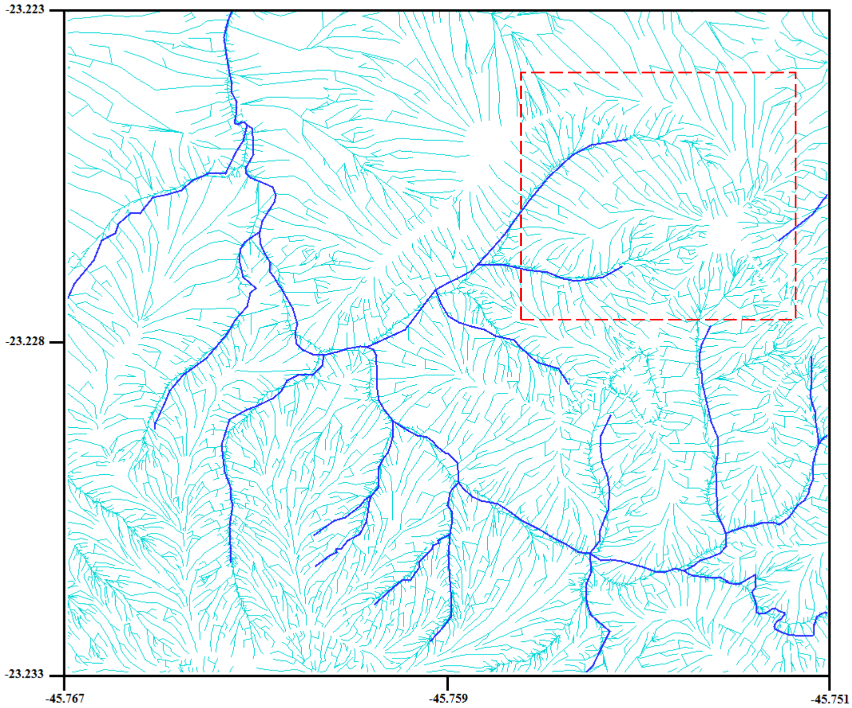

Figure 7. Drainage paths converge to the reference drainage network

For a more precise view of how the drainage paths are distributed across the triangles of the TIN used as terrain model, a closer look at both the drainage paths in cyan and the triangulation in red is given in figure 8 that contains the area bounded by the dashed rectangle of figure 7.

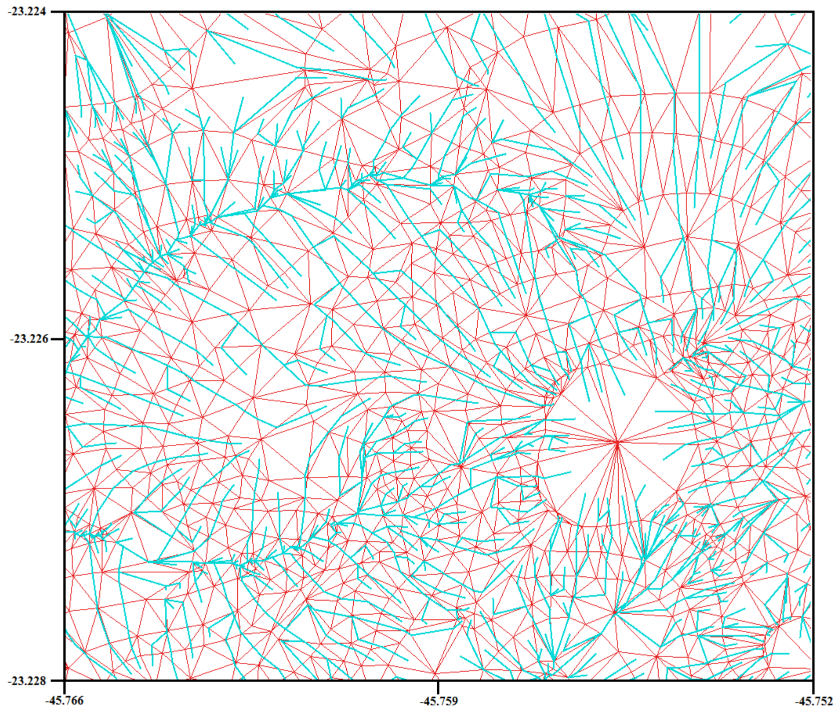

Figure 8. Drainage paths over a TIN

The primary and most significant concern to be considered when analyzing the effectiveness of the methods is the quality in the results obtained after applying all the procedures to the TIN terrain model, i.e., the drainage paths converging to streams of a drainage network, although computational times are also an important aspect related to the complexity of drainage-related structures.

The number of triangles in the final TIN and computational times took by the algorithms described in this work are given in table 1 for different numbers of input points. The total times shown below include the execution times of the procedures for removing the intersections between triangles edges and contour lines, interpolating new elevation values to the critical points in order to remove flat areas, re-triangulating the entire set of points after the addition of these new critical points into the set, calculating the plane gradient and all the drainage paths defined from each triangle. The algorithms were compiled for 64-bit and executed on a PC with Intel Core i7 2.93 GHz CPU and 8 GB of RAM memory.

Table 1. Details on TIN and execution times

Number of points	Number of triangles	Total execution time (s)
50000	148857	1.95
100000	265069	3.33
150000	396958	4.92
200000	512437	6.26

5. Conclusions and Future Work

Triangulated irregular terrain models are structures that can efficiently represent terrain surfaces. These models are calculated from terrain-specific points scattered over a region obtained from a land survey.

The algorithms and procedures developed for processing a TIN have low computational complexities which make this model an attractive alternative to other terrain models. Drainage paths following the streams of the drainage network illustrated in the previous section indicate that these patterns represent good approximations that are consistent to potential surface water flows and can be used in decision-making systems supporting studies of their impacts in hydrologic processes.

In this work, flat areas were removed by a procedure that defines a path of flat triangles and interpolates elevation values of critical points. Branches found in the path are also processed in order to complete paths previously found. The delineation of drainage paths traced by starting at each triangle centroid, ordered by their elevation values and also connected to each other result in very good water flow patterns that are consistent to real-world terrain surfaces.

Next steps to be taken in future works are careful investigations of precise definitions about the concepts of flow accumulation and contributing areas for the delineation of watersheds given by a drainage network. Pit removal must also be considered as the flow directions need to be continuous between all the triangles. Computational times could be improved by a detailed analysis and further optimizations in the algorithms.

The assignment of flow directions obtained from drainage paths to triangles and vertices in flow computation processes can produce important quantifications of water flow distribution that are essential to Hydrology.

References

Agarwal, P., De Berg, M., Bose, P., Dobrint, K., Van Kreveld, M., Overmars, M., De Groot, M., Roos, T., Snoeyink, J., Yu, S. (1996). The complexity of rivers in triangulated terrains. In *8th Canadian Conference on Computational Geometry*, pages 325–330.

Barbalić, D., Omerbegović, V. (1999). "Correction of horizontal areas in TIN terrain modeling–algorithm", http://proceedings.esri.com/library/userconf/proc99/proceed/papers/pap924/p924.htm

Bowyer, A. (1981). Computing Dirichlet tessellations. In *The Computer Journal*, pages 162–166.

Câmara, G., Souza, R. C. M., Pedrosa, B. M., Vinhas, L., Monteiro, A. M. V., Paiva, J. A., Carvalho, M. T., Gattass, M. (2000). TerraLib: technology in support of GIS innovation. In *II Brazilian Symposium on Geoinformatics, GeoInfo2000*, pages 1–8.

Cignoni, P., Montani, C., Scopigno, R. (1998). DeWall: A fast divide & conquer Delaunay triangulation algorithm in E^d. In *Computer-Aided Design*, pages 333–341.

De Berg, M., Cheong, O., Van Kreveld, M. and Overmars, M. (2008). Computational Geometry – Algorithms and Applications, Springer, 3rd edition.

Eastman, J. R. (2001). Idrisi32 Release 2 – Guide to GIS and Image Processing, Volume 2, Clark Labs.

Fortune, S. (1987). A sweepline algorithm for Voronoi diagrams. In *Algorithmica*, pages 153–174.

Fowler, R. J., Little, J. J. (1979). Automatic extraction of irregular network digital terrain models. In *ACM SIGGRAPH Computer Graphics*, pages 199–207.

Felgueiras, C. A., Goodchild, M. F. (1995). An incremental constrained Delaunay triangulation. In *NCGIA Technical Report 95-2*, pages 31–46.

Guibas, L. J., Knuth, D. E., Sharir, M. (1992). Randomized incremental construction of Delaunay and Voronoi diagrams. In *Algorithmica*, pages 381–413.

Jones, N. L., Wright, S. G., Maidment, D. R. (1990). Watershed delineation with triangle-based terrain models. In *Journal of Hydraulic Engineering*, pages 1232–1251.

O'Rourke, J. (1998). Computational Geometry in C, Cambridge University Press, 2nd edition.

Prefeitura Municipal de São José dos Campos. (2003). Base de Dados "Cidade Viva". Departamento de Planejamento Urbano, Serviço de Geoprocessamento, http://www.sjc.sp.gov.br/secretarias/planejamento_urbano/geoprocessamento.aspx (in Portuguese).

Silfer, A. T., Kinn, G. J., Hassett, J. M. (1987). A geographic information system utilizing the triangulated irregular network as a basis for hydrologic modeling. In *8th International Symposium on Computer-Assisted Cartography*, pages 129–136.

Tsai, V. J. D. (1993). Delaunay triangulations in TIN creation: an overview and a linear-time algorithm. In *International Journal of Geographical Information Systems*, pages 501–524.

Tsirogiannis, C. P. (2011). Analysis of flow and visibility on triangulated terrains. PhD Thesis. Eindhoven University of Technology.

Watson, D. F. (1981). Computing the n-dimensional Delaunay tessellation with application to Voronoi polytypes. In *The Computer Journal*, pages 167–172.

Zhu, Y. and Yan, L. (2010). An improved algorithm of constrained Delaunay triangulation based on the diagonal exchange. In *2nd International Conference on Future Computer and Communication*, pages 827–830.