

# Generating Realistic Terrains with Higher-Order Delaunay Triangulations

*Thierry de Kok*

*Marc van Kreveld*

*Maarten Löffler*

institute of information and computing sciences, utrecht university

technical report UU-CS-2005-020

[www.cs.uu.nl](http://www.cs.uu.nl)

# Generating Realistic Terrains with Higher-Order Delaunay Triangulations

Thierry de Kok      Marc van Kreveld      Maarten Löffler

Institute of Information and Computing Sciences,  
Utrecht University, the Netherlands  
`takok,marc,mloffler@cs.uu.nl`

## Abstract

For hydrologic applications, terrain models should have few local minima, and drainage lines should coincide with edges. We show that triangulating a set of points with elevations such that the number of local minima of the resulting terrain is minimized is NP-hard for degenerate point sets. The same result applies when there are no degeneracies for higher-order Delaunay triangulations. Two heuristics are presented to reduce the number of local minima for higher-order Delaunay triangulations, which start out with the Delaunay triangulation. We give efficient algorithms for their implementation, and test on real-world data how well they perform. We also study another desirable drainage characteristic, few valley components, and how to obtain it for higher-order Delaunay triangulations. This gives rise to a third heuristic. Tables and visualizations show how the heuristics perform for the drainage characteristics on real-world data.

## 1 Introduction

A fundamental geometric structure in computational geometry is the triangulation. It is a partitioning of a point set or region of the plane into triangles. A triangulation of a point set  $P$  partitions the convex hull of  $P$  into triangles whose vertices are exactly the points of  $P$ . The most common triangulation of a set of points is the Delaunay triangulation. It has the property that for every triangle, the circumcircle through its vertices does not contain any points inside. It maximizes the smallest angle used over all possible triangulations of the point set.

If the points all have an associated attribute value like elevation, then a triangulation defines a piecewise linear interpolant. Due to the angle property, triangles in a Delaunay triangulation are generally well-shaped and are suitable for spatial interpolation. When using triangulations for terrain modeling, however, one should realize that terrains are formed by natural processes. This implies that there are linear depressions (valleys) formed by water flow, and very few local minima occur [16, 18]. Local minima can be caused by erroneous triangulation: an edge may stretch from one side of a valley to the opposite side. Such an edge is an artificial dam, and upstream from the dam in the valley, a local minimum appears. See Figure 1 (left). It is often an artifact of the triangulation. Therefore, minimizing local minima is an optimization criterion for terrain modeling. In extension, the contiguity of valley lines is also a natural phenomenon. Valleys do not start and stop halfway a mountain slope, but the Delaunay triangulation may contain such artifacts. Hence, a second optimization criterion is minimizing the number of valley line components.

Terrain modeling in GIS is used for morphological processes like drainage and erosion, or for hazard analysis like avalanches and landslides. Local minima and undesirable valley lines can influence the computation of these processes, leading to unreliable outcomes. This motivates our study for the construction of realistic terrains by avoiding local minima and artifact valleys.

An alternative to deal with local minima is flooding. Here a local minimum and its surroundings are elevated until a height at which a local minimum does not appear anymore (pit filling) [10, 11, 12]. Such methods help to create terrains with better drainage characteristics, or to define a drainage basin hierarchy, but it is clear that other artifacts are introduced at the same time. Furthermore, this approach does not respect the given input elevations.

Returning to planar triangulations, there are many different ways in which one can define the quality of a triangulation of a set of points. A criterion that is always important for triangulations is the nice shape of the triangles. This can be formalized in several ways [3, 4]. In this paper, nice shape is formalized by *higher-order Delaunay triangulations* [8]. They provide a class of triangulations that are all reasonably well-shaped, depending on a parameter  $k$ .

**Definition 1** *A triangle in a point set  $P$  is order- $k$  if its circumcircle contains at most  $k$  points of  $P$ . A triangulation of a set  $P$  of points is an order- $k$  Delaunay triangulation if every triangle of the triangulation is order- $k$  (see Figure 1 (middle)).*

So a Delaunay triangulation is an order-0 Delaunay triangulation. For any positive integer  $k$ , there can be many different order- $k$  Delaunay triangulations. The higher  $k$ , the more freedom to eliminate artifacts like local minima, but the worse the shape of the triangles can become.

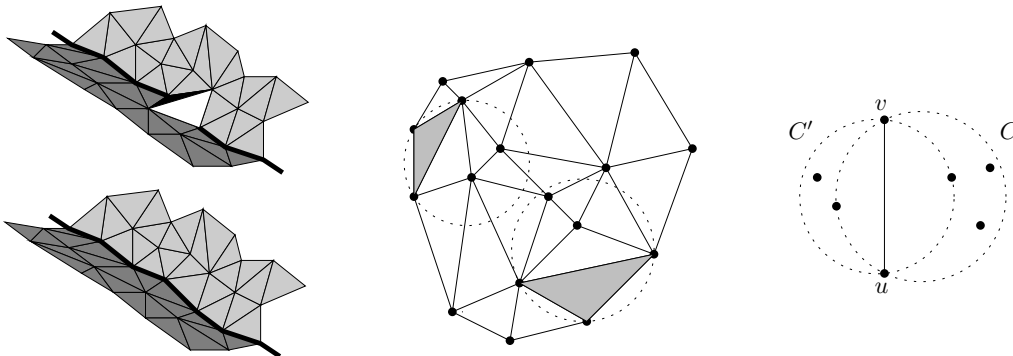


Figure 1: Left, an artificial dam and local minimum in a terrain. Middle, an order-2 Delaunay triangulation, with two triangles and their circumcircles, showing the order. Right, the useful order of the edge  $\overline{uv}$  is 3, the maximum of the number of points inside  $C$  or  $C'$ .

This paper discusses triangulations of a point set  $P$  of which elevations are given for terrain modeling. In Section 2 we concentrate on minimizing local minima. We show that over all possible triangulations, minimizing local minima is NP-hard. This result relies heavily on degenerate point sets. For order- $k$  Delaunay triangulations, NP-hardness can also be shown for non-degenerate point sets, for  $k = \Omega(n^\epsilon)$  and  $k \leq c \cdot n$ , for some  $0 < \epsilon < 1$  and some  $0 < c < 1$ . (For  $k = 1$ , an  $O(n \log n)$  time algorithm that minimizes local minima was given in [8].) Then we discuss two heuristics for reducing local minima in order- $k$  Delaunay triangulations, the flip and hull heuristics, and their efficiency. The latter was introduced before in [8]; here we give a more efficient algorithm. Then we compare the two heuristics experimentally on various terrains. We only examine orders 0 up to 8; higher orders are less interesting in practice since the interpolation quality may be less good, and skinny triangles may cause artifacts in visualization. The experiments address the following questions:

- Does the concept of higher-order Delaunay triangulations help to reduce the number of local minima on real terrains?
- In particular, do the flip and hull heuristics accomplish this? How does this depend on the order of the Delaunay triangulation?

- How do these heuristics compare to each other?

In Section 3 we extend our study to deal with valley line components as well. It appears that the removal of local minima actually can create artifact valley lines, especially for the flip heuristic. Two solutions are presented for this problem. We give a method to remove isolated valley edges, and to extend valley line components so that they join with other valley components. In short, we try to reduce the number of valley line components, leading to the valley heuristic. Again we complement our methods by experiments on various terrains. The results are given in tables and by visualization. The experiments address the following questions:

- Does the reduction of local minima by the flip or hull heuristics have influence on the number of valley edge components?
- Does the valley heuristic achieve a reduction in the number of valley edge components? Can it be applied best to the Delaunay triangulation or the outcome of the flip or hull heuristic for minimizing local minima?

Section 4 gives the conclusions and lists several directions for further research.

## 2 Minimizing the number of local minima

The next subsection shows NP-hardness of minimizing local minima in two settings. Subsections 2.2 and 2.3 present the flip and hull heuristics and contain a running time analysis. Both heuristics can be implemented in  $O(nk^2 + nk \log n)$  time, where  $n$  is the number of points and  $k$  is the order of the higher-order Delaunay triangulation. For the hull heuristic this is an improvement over the  $O(nk^3 + nk \log n)$  time bound of Gudmundsson et al. [8].

### 2.1 NP-hardness of minimizing local minima

For a set  $P$  of  $n$  points in the plane, it is easy to compute a triangulation that minimizes the number of local minima if there are no degeneracies (no three points on a line). Assume  $p$  is the lowest point. Connect every  $q \in P \setminus \{p\}$  with  $p$  to create a star network with  $p$  as the center. Complete this set of edges to a triangulation in any way. Since every point but  $p$  has a lower neighbor, no point but  $p$  can be a local minimum. Hence, this triangulation is one that minimizes the number of local minima. When degeneracies are present, minimizing the number of local minima is NP-hard.

**Theorem 1** *Let  $P$  be a set of  $n$  points in the plane, and assume that the points have elevations. It is NP-hard to triangulate  $P$  with the objective to minimize the number of local minima of the polyhedral terrain.*

**Proof:** By reduction from maximum size non-intersecting subset in a set of line segments [2]. Let  $S$  be any set of  $n$  line segments in the plane, and assume all  $2n$  endpoints are disjoint (this can easily be enforced by extending segments slightly). Let  $P$  be the set of the  $2n$  endpoints. Let  $\gamma$  be the smallest distance between two points in  $P$ . For every point  $p \in P$ , let  $C(p)$  be a circle centered at  $p$  with radius  $\gamma/3$ . If  $p'$  is the point in  $P$  such that  $\overline{pp'}$  is a segment of  $S$ , then for every  $q \in P \setminus \{p, p'\}$ , place a point at the intersection of  $\overline{pq}$  and  $C(p)$ , see Figure 2, left. We call these points *shields*, because they prevent  $p$  and  $q$  from being connected by a line segment in any triangulation. Let  $H$  be the set of shields.

We assign elevations as follows. For every segment in  $S$ , one endpoint is assigned elevation 1 and the other is assigned elevation 2. Every shield in  $H$  is assigned elevation 3. By the choice of shields, every point with elevation 1 is a local minimum, and no point with elevation 3 can be a local minimum. A point with elevation 2 is a local minimum if and only if the segment from  $S$  that connects  $p'$  to the other endpoint  $p$  is not in the triangulation. Hence, the maximum non-intersecting subset of  $S$  corresponds one-to-one with the points with elevation 2 that are local

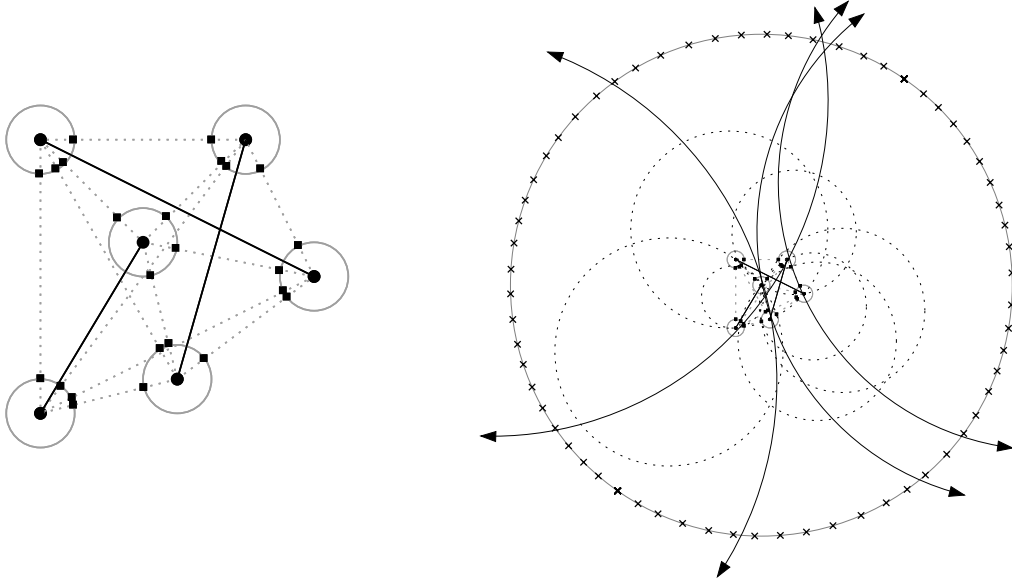


Figure 2: Left, construction for the NP-hardness proof. Square points are in  $H$ , circular points are in  $P$ . Right, NP-hardness for higher-order Delaunay triangulations. Crosses are points at least  $4r$  from the points in  $P \cup H$ .

minimum. Since the number of shields is quadratic, NP-hardness follows directly.  $\square$

Based on the construction in the proof above, we can show NP-hardness of minimizing the number of local minima for higher-order Delaunay triangulations even when no degeneracies exist.

We define the *useful order* of an edge as the lowest order of a triangulation that includes this edge. In [8] it was shown that the useful order is actually defined by the number of points in one of two circles, see Figure 1 (right). Let  $\overline{uv}$  be the edge whose useful order we wish to determine, and assume without loss of generality that  $\overline{uv}$  is vertical. Let  $C$  (and  $C'$ ) be the circle that passes through  $u$  and  $v$ , with leftmost (resp. rightmost) center, and which does not contain in its interior any point left (resp. right) of the line through  $u$  and  $v$ . If  $k$  is the maximum of the number of points inside  $C$  and  $C'$ , then the useful order of edge  $\overline{uv}$  is  $k$ .

**Corollary 1** *Let  $P$  be a set of  $n$  points in the plane such that no three points of  $P$  lie on a line, and assume that the points have elevations. For any  $0 < \epsilon < 1$  and some  $0 < c < 1$ , it is NP-hard to compute an order- $k$  Delaunay triangulation that minimizes the number of local minima of the polyhedral terrain for  $n^\epsilon \leq k \leq c \cdot n$ .*

**Proof:** Start out with the proof of the theorem above;  $|P| = 2n$  and  $|H| = 2n(2n - 1)$ . Let  $r$  be the radius of the largest (finite) circle that passes through any three points of  $P \cup H$  of the construction. Note that  $r$  is at least half of the diameter of  $P \cup H$ . Let  $\delta > 0$  be a value chosen very small, such that if three non-collinear points from  $P \cup H$  move over a distance  $\delta$ , then their circle has radius at most  $2r$  (such circles are shown dashed in Figure 2, right). Both  $r$  and  $\delta$  can be computed in cubic time. Choose any point of  $P$  to be center of a large circle  $C$  of radius  $5r$ , and place  $128n^2$  points evenly spaced on it (crosses in Figure 2, right). Due to the diameter of  $P \cup H$  and the upper bound of  $2r$  on the radius of circles defined by three non-collinear points (possibly displaced slightly), none of these circles intersects  $C$ .

For every shield  $h \in H$ , displace it over a very small distance  $d$  where  $0 < d < \delta$ , and such that the radius of the circle through  $h$  and the two points of  $P$  for which  $h$  is a shield has radius at least  $6r$ ; some of these circles are shown partially in Figure 2, right. Such a circle contains an arc of  $C$  of length more than  $r$ , and hence it contains at least  $\frac{1}{10\pi} 128n^2 > 4n^2$  of the points placed

on it. This makes sure that the triangulation edge for which  $h$  was a shield, is still not possible in an order- $4n^2$  Delaunay triangulation: the useful order of the triangulation edge is too high. By construction, other edges between points in  $P$  and  $H$  are possible because the relevant circles cannot intersect  $C$ . The extra points on  $C$  get elevation 3 as well, and the problem of minimizing the number of local minima in order- $4n^2$  Delaunay triangulations is again the same as maximizing the size of a non-intersecting subset of  $S$ . Displacements of shields of  $H$  can be done to remove all degeneracies while creating no new ones. The points in  $C$  can also be placed without creating degeneracies. The total number of points in the construction is  $O(n^2)$ . This gives the proof for  $k = c \cdot n$  for some  $0 < c < 1$ . For smaller values of  $k$  we simply place many more points on  $C$ . As long as  $k = \Omega(n^\epsilon)$  the construction is polynomial.  $\square$

## 2.2 The flip heuristic

Given a value of  $k$ , the flip heuristic repeatedly tests whether the diagonal of a convex quadrilateral in the triangulation can be flipped. It will be flipped if two conditions hold simultaneously: (i) The two new triangles are order- $k$  Delaunay triangles. (ii) The new edge connects the lowest point of the four to the opposite point. A flip does not necessarily remove a local minimum, but cannot create one, and it can make possible that a later flip removes a local minimum.

Our algorithm to perform the flips starts with the Delaunay triangulation and  $k' = 1$ , then does all flips possible to obtain an order- $k'$  Delaunay triangulation, then increments  $k'$  and repeats. This continues until  $k' = k$ .

We first analyze the maximum number of flips possible, and then we discuss the efficiency of the heuristic.

**Lemma 1** *The flip heuristic terminates after at most  $O(n^2)$  flips.*

**Proof:** Normalize the heights of the vertices to be integers in the range  $1, \dots, n$ . Observe that this does not influence the flipping criterion. Consider the function  $F(T)$  for a triangulation  $T$ :

$$F(T) = \sum_{\overline{uv} \in T} \min(u, v).$$

Any flip decreases  $F(\cdot)$  with at least one, and  $F(\cdot)$  is at most  $O(n^2)$  to begin with.  $\square$

**Lemma 2** *If an edge  $\overline{ab}$  is in the triangulation, then the flip heuristic will never have an edge  $\overline{cd}$  later with  $\min(c, d) \geq \min(a, b)$  that intersects  $\overline{ab}$ .*

**Proof:** Assume without loss of generality that  $a < b$  and  $c < d$ . Assume further that  $a \leq c$ , edge  $\overline{ab}$  is in the triangulation  $T$  at some moment, and that  $\overline{cd}$  is the first edge flipped in a triangulation  $T'$  that exists after  $T$  that violates the property of the lemma. In  $T'$ ,  $\overline{ab}$  is not an edge, and  $c$  and  $d$  must be in a convex quadrilateral where  $c$  is the lowest point of the four. The other two points,  $f$  and  $g$ , cannot be  $a$ , because  $a$  is lower by assumption. Possibly,  $f$  or  $g$  is the same as  $b$ . The quadrilateral has edges  $\overline{cf}, \overline{cg}, \overline{df}, \overline{dg}$ , and at least one of these intersects  $\overline{ab}$ . This contradicts the assumption that  $\overline{cd}$  is the first edge violating the property.  $\square$

An immediate consequence of the lemma above is that an edge that is flipped out of the triangulation cannot reappear. There are at most  $O(nk)$  pairs of points in a point set of  $n$  points that give order- $k$  Delaunay edges [8]. Therefore, we conclude:

**Lemma 3** *The flip heuristic to reduce the number of local minima performs at most  $O(nk)$  flips.*

To implement the flip heuristic efficiently, we maintain the set of all convex quadrilaterals in the current triangulation, with the order of the two triangles that would be created if the diagonal

were flipped. The order of a triangle is the number of points in the circumcircle of the vertices of the triangle. Whenever a flip is done, we update the set of convex quadrilaterals. At most four are deleted and at most four new ones are created by the flip. We can find the order of the incident triangles by circular range counting queries. Since we are only interested in the count if the number of points in the circle is at most  $k$ , we implement circular range counting queries by point location in the order- $(k + 1)$  Voronoi diagram [15], taking  $O(\log n + k)$  time per query after  $O(nk \log n)$  preprocessing time. We conclude:

**Theorem 2** *The flip heuristic to reduce the number of local minima in order- $k$  Delaunay triangulations on  $n$  points takes  $O(nk^2 + nk \log n)$  time.*

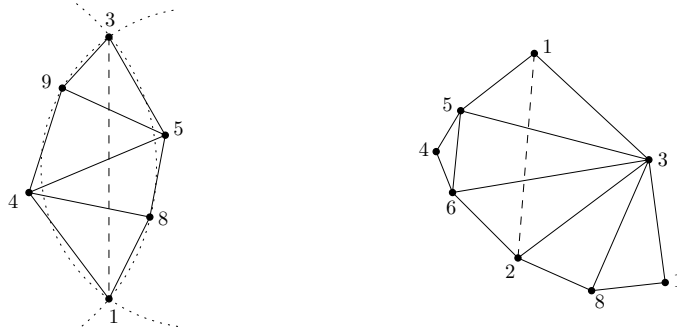


Figure 3: Left, the flip heuristic does not always give the optimum, even for order-2. Right, the hull heuristic does not always give the optimum, even for order-2. Numbers are elevations.

Figure 3 (left) shows an example where the flip heuristic does not compute the optimal solution even for order-2 Delaunay triangulations. We use elevations as names. The edge 5, 9 may not be flipped because  $\triangle 3, 4, 9$  is not order-2, and for the same reason the edge 4, 8 may not be flipped. Edge 4, 5 is not flipped due to the heuristic. Therefore, edge 1, 3 will not be inserted, although it is useful order-2.

The possibilities of flipping in higher-order Delaunay graphs were also considered in [1]. However, their definition of the order of a Delaunay edge is not its *useful* order, and the order of triangles is not considered. Therefore, their results do not apply to higher-order Delaunay triangulations in the definition of [8, 9] used here.

### 2.3 The hull heuristic

The second heuristic for reducing the number of local minima is the hull heuristic. It was described by Gudmundsson et al. [8], and has an approximation factor of  $\Theta(k^2)$  of the optimum. The hull heuristic adds a useful order- $k$  Delaunay edge  $e$  if it reduces the number of local minima. This edge may intersect several Delaunay edges, which are removed; the two holes in the triangulation that appear are retriangulated with the constrained Delaunay triangulation [5] in  $O(k \log k)$  time. These two polygonal holes are called the hull of this higher-order Delaunay edge  $e$ . The boundary of the hull consists of Delaunay edges only. No other higher-order Delaunay edges will be used that intersects this hull. This is needed to guarantee that the final triangulation is order- $k$ . It is known that two useful order- $k$  Delaunay edges used together can give an order- $(2k - 2)$  Delaunay triangulation [9], which is higher than allowed. Here we give a slightly different implementation than in [8]. It is more efficient for larger values of  $k$ .

Assume that a point set  $P$  and an order value  $k$  are given. We first compute the Delaunay triangulation  $T$  of  $P$ , and then compute the set  $E$  of all useful order- $k$  Delaunay edges, as in [8], in  $O(nk \log n + nk^2)$  time. There are  $O(nk)$  edges in  $E$ , and for each we have the lowest order  $k' \leq k$  for which it is a useful order- $k'$  Delaunay edge.

	0	1	2	3	4	5	6	7	8
Calif. Hot Springs	47/47	43/43	33/31	29/26	25/20	24/19	23/18	21/18	18/16
Wren Peak	45/45	37/37	31/31	27/27	24/22	23/21	21/20	19/20	19/20
Quinn Peak	53/53	44/44	36/36	31/29	26/25	24/23	23/21	21/20	20/19
Sphinx Lakes	33/33	27/27	22/22	20/19	19/18	17/16	15/12	12/9	11/9
Split Mountain	24/24	17/17	14/14	9/9	9/9	9/9	8/8	7/8	6/7

Table 1: Results of the flip/hull heuristic for orders 0–8.

Next we determine the subset  $P' \subseteq P$  of points that are a local minimum in the Delaunay triangulation. Then we determine the subset  $E' \subseteq E$  of edges that connect a point of  $P'$  to a lower point. These steps trivially take  $O(nk)$  time.

Sort the edges of  $E'$  by non-decreasing order. For every edge  $e \in E'$ , traverse  $T$  to determine the edges of  $T$  that intersect  $e$ . If any one of them is not a Delaunay edge or is a marked Delaunay edge, then we stop and continue with the next edge of  $E'$ . Otherwise, we remove all intersected Delaunay edges and mark all Delaunay edges of the polygonal hole that appears. Then we insert  $e$  and retriangulate the hull, the two polygons to the two sides of  $e$ , using the Delaunay triangulation constrained to the polygons. We also mark these edges. Finally, we remove edges from  $E'$ : If the inserted edge  $e$  made that a point  $p \in P$  is no longer a local minimum, then we remove all other edges from  $E'$  where  $p$  is the highest endpoint.

Due to the marking of edges, no edge  $e \in E'$  will be inserted if it intersects the hull of a previously inserted edge of  $E'$ . Every edge of  $E'$  that is not used in the final triangulation is treated in  $O(\log n + k)$  time, and every edge of  $E'$  that is used in the final triangulation is treated in  $O(\log n + k \log k)$  time. We conclude:

**Theorem 3** *The hull heuristic to reduce the number of local minima in order- $k$  Delaunay triangulations on  $n$  points takes  $O(nk^2 + nk \log n)$  time.*

The hull heuristic may also not give the optimum. Figure 3 (right) shows an example where an edge 1, 2 shown dashed may be inserted first, because it is a useful order-2 Delaunay edge and there are no useful order-1 Delaunay edges in the set. Edge 3, 4 is useful order-2 Delaunay but it cannot be inserted anymore, since it intersects a previously inserted edge (and hence its hull). So the point with elevation 4 will stay a local minimum. However, if 2 were connected to the other point 1, also a useful order-2 Delaunay edge, then the edge 3, 4 could have been used, and the resulting triangulation would have one fewer local minimum.

## 2.4 Experiments

Table 1 shows the number of local minima obtained after applying the flip and hull heuristics to five different terrains. The terrains roughly have 1800 vertices. The vertices were chosen by random sampling 1% of the points from elevation grids. Vertices that have the same elevation may cause flat edges and triangles, in which case a whole triangle can be a local minimum. We simulated simplicity by treating a vertex  $(x, y)$  with height  $z$  as a vertex with height  $(z, x, y)$  interpreted lexicographically, where  $x$  and  $y$  are the lesser significant components in the lexicographic order. This takes care of most potential problems. Furthermore, local minima that appear on the terrain boundary were not counted as local minima in the table.

The values in the table show that higher-order Delaunay triangulations indeed can give significantly fewer local minima than the standard Delaunay triangulation (order-0). This effect is already clear at low orders, indicating that many local minima of Delaunay triangulations may be caused by having chosen the wrong edges for the terrain (interpolation).

The difference in local minima between the flip and hull heuristics shows that the hull heuristic usually is better, but there are some exceptions.

To test how good the results are, we also tested how many local minima of each terrain cannot be removed simply because there is no useful order- $k$  Delaunay edge to a lower point. It turned



out that the hull heuristic found an optimal order- $k$  Delaunay triangulation in all cases except for five, where one local minimum too many remained. In four of these cases the flip heuristic found an optimal order- $k$  Delaunay triangulation. In one case (Wren, order-6) it is not clear; a triangulation with 19 local minima may exist.

### 3 Minimizing the number of valley edge components

In a triangulation representing a terrain, there are three types of edges: ridge or diffluent edges, normal or transfluent edges, and valley or confluent edges [7, 13, 19]. These edges are used to delineate drainage basins and other hydrological characteristics of terrains. Flow on terrains is usually assumed to take the direction of steepest descent. This is a common assumption used in drainage network modeling [17, 19]. Assuming no degeneracies (including horizontal triangles), every point on the terrain has a unique direction of steepest descent, except local minima. Hence, a flow path downward in the terrain can be defined for any point. The direction of steepest descent at a vertex can be over an edge or over the interior of a triangle. Flow on a triangle is always normal to the contour lines on that triangle.

Ridge edges are edges that do not receive flow from any point on the terrain. The incident triangles drain in a direction away from this edge. Valley edges are edges that receive flow from (part of) both incident triangles; they would be ridge edges if the terrain were upside down. Normal edges receive flow from (part of) one incident triangle and drain to the other. Valley edges can be used to define the drainage network, and ridge edges can be used for the drainage basins, also called catchment areas [13]. Many more results on drainage in GIS exist; it is beyond the scope of this paper to review it further.

Just like local minima in a triangulation for a terrain are often artifacts, so are isolated valley edges, and sequences of valley edges that do not end in a local minimum. In the latter case, flow would continue over the middle of a triangle, which usually does not correspond to the situation in real terrains. If channeled water carves a valley-like shape in a terrain, then the valley does not suddenly stop, because the water will stay channeled. This is true unless the surface material changes, or the terrain becomes nearly flat [14]. Besides being unrealistic, isolated valley edges may influence the shape of drainage basins [13]. It is also known that allowing drainage through the interior of triangles, according to steepest descent, can cause drainage networks of complexity up to  $\Theta(n^3)$  for triangulations with  $n$  triangles [6].

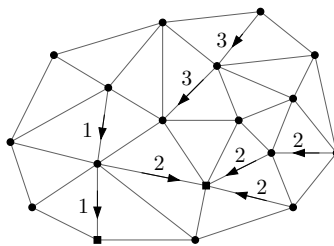


Figure 4: Example of a terrain with three valley edge components.

We define a *valley (edge) component* to be a maximal set of valley edges such that flow from all of these valley edges reaches the lowest vertex incident to these valley edges. A valley component necessarily is a rooted tree with a single target that may be a local minimum. Figure 4 shows an example of a terrain with three valley components; the valley edges are shown by the direction of flow, numbers indicate the identity of each component, and squares show local minima. In this example, the direction of steepest descent from the vertex where components 1 and 2 touch is over the edge labeled 1 to the boundary. Component 3 ends in a vertex that is not a local minimum; flow proceeds over a triangle.

By the discussion above, the drainage quality of a terrain is determined by:

- the number of local minima, and
- the number of valley edge components that do not end in a local minimum.

The sum of these two numbers immediately gives the total number of valley edge components. We will attempt to reduce this number with a new heuristic called the valley heuristic. But first we analyze how many valley edges and valley components appear in the results of the flip and hull heuristics.

### 3.1 Consequences of the flip and hull heuristics on the valleys

The flip and hull heuristics can remove local minima of triangulations, and therefore they seem more realistic as terrains. However, the heuristics may create valley edges, including isolated ones. This is especially true for the flip heuristic. We examined the triangulations obtained from the experiments of Subsection 2.4 and analyzed the number of valley edges and valley components.

	Quinn	flip-8	hull-8	Wren Peak	flip-8	hull-8
Edges	5210	5210	5210	5185	5185	5185
Valley edges	753	862	684	799	860	761
Local minima	53 (12)	20 (9)	19 (6)	45 (17)	19 (12)	19 (17)
Valley components	240	289	224	259	270	247
Not min. ending	173 (8)	246 (15)	191 (8)	185 (18)	218 (23)	199 (18)
	Sphinx	flip-8	hull-8	Hot Springs	flip-8	hull-8
Edges	5179	5179	5179	5234	5234	5234
Valley edges	675	830	627	853	964	807
Local minima	33 (16)	11 (11)	9 (16)	47 (20)	18 (16)	16 (20)
Valley components	261	313	244	249	256	231
Not min. ending	213 (5)	285 (9)	218 (7)	179 (11)	210 (17)	190 (13)

Table 2: Statistics for four terrains. For each terrain, counts for the Delaunay triangulation are given, and for the outcome of the flip and hull heuristics for order 8. The last row gives the number of valley edge components that do not end in a local minimum. Numbers between brackets are the additional numbers for the terrain boundary.

Table 2 shows statistics on four terrains (the fifth terrain we tested has similar values). To define valley edges and flow when vertices have the same elevation, we treated elevation as a lexicographic number  $(z, x, y)$  as before. Local minima on the boundary are not counted, but their number is shown separately in brackets. The same is true for valley components that end on the boundary of the terrain, but not in a local minimum. Note that local minima on the boundary may not have any valley component ending in it. We can see that the flip heuristic has increased the number of valley edges considerably, whereas the hull heuristic decreased this number. The same is true for the number of valley components. Another observation from the table is that there are many valley edge components. The average size is in all cases between 2 and 4 edges. This shows the need for further processing of the triangulation, or for a different heuristic.

### 3.2 The valley heuristic

We apply two methods to improve the drainage quality of the terrain. Firstly, isolated valley edges can sometimes be removed by a single flip or useful order- $k$  edge insertion, reducing the number of valley components. Secondly, if a valley edge has a lower endpoint whose direction of steepest descent goes over a triangle, then the valley component can sometimes be extended downhill and possibly be connected to another valley component, which reduces the number of valley components. We observe:

**Observation 1** (i) For a convex quadrilateral in a terrain, at most one diagonal is a valley edge.  
(ii) If in a triangle in a terrain, two edges are valley edges, then their common vertex is the lowest

vertex of that triangle. (iii) A flip in a convex quadrilateral affects at most five edges (for being valley or not).

To remove an isolated valley edge, five candidate flips can take care of this: the valley edge itself, and the four other edges incident to the two triangles incident to the valley edge. A flip can potentially remove one isolated valley edge but create another one at the same time; such a flip is not useful and termination of the heuristic would not be guaranteed. Any flip changes the flow situation at four vertices. There are many choices possible when to allow the flip and when not. We choose to flip only if the flow situation of the four vertices of the convex quadrilateral does not change, except for the removal of the isolated valley edge, and the two new triangles are order- $k$  Delaunay. In particular, a flip may not create new valley edges. It is undesirable to change any valley component in an unintended way. Algorithmically, identifying isolated valley edges and flipping them, if possible, can be done efficiently. There will be at most  $O(n)$  flips. The most expensive test is to decide if the two new triangles are order- $k$ . By computing the order- $(k+1)$  Voronoi diagram, storing with every cell the  $(k+1)$ -th closest point, and preprocessing the diagram for planar point location, we can decide this in  $O(\log n)$  time after  $O(nk \log n)$  preprocessing time [15].

To extend a valley component downward, we take its lowest endpoint  $v$  and change the triangulation locally to create the situation that  $v$  has a direction of steepest descent over a valley edge to a lower vertex. We do this regardless of the situation that  $v$  is a local minimum, or  $v$  has its direction of steepest descent over the interior of some triangle. We only do this with flips. For every triangle incident to  $v$ , we test if it is part of a convex quadrilateral  $vpwq$ , and if so, we test if the flip of  $\overline{pq}$  to  $\overline{vw}$  yields two order- $k$  Delaunay triangles,  $\overline{vw}$  is a valley edge,  $w$  is lowest of  $vpwq$ , the steepest descent from  $v$  is to  $w$ , and no valley components are interrupted at  $p$  or  $q$ . Possibly, more than one triangle incident to  $v$  satisfies the requirements. We only do one flip, namely the one giving the steepest valley edge  $\overline{vw}$ .

Throughout the algorithm, any vertex can be lowest point of a valley component at most twice. First as the lower endpoint of an isolated valley edge, and once more by extending valley components. Once a vertex is in a valley component with more than two edges, it will stay in such a component. Hence, there will only be  $O(n)$  flips. Again using point location in the order- $(k+1)$  Voronoi diagram, we conclude:

**Theorem 4** *The valley heuristic to reduce the number of valley components in order- $k$  Delaunay triangulations on  $n$  points takes  $O(nk \log n)$  time.*

### 3.3 Experiments

The two ways of reducing the number of valley components were applied to the Delaunay triangulation, and to the outcomes of the flip and hull heuristics. We show the results in Table 3 for order 8 only. In fact, the table shows the outcome of applying the valley heuristic (with order 8) to all outcomes of Table 2.

We observe that the valley heuristic succeeds in reducing the number of valley components considerably in all cases. The reduction is between 20% and 40% for all triangulations. There is no significant difference in reduction between the three types of triangulations. The valley heuristic by itself also reduces the number of local minima, as can be expected. The number of valley components is lowest when applying the valley heuristic to the outcome of the hull heuristic, and sometimes when applying the valley heuristic directly to the Delaunay triangulation. Further examination of the outcome shows that the largest reduction in the number of valley components comes from the removal of isolated valley edges. This accounts for roughly 60% to 80% of the reduction.

We also evaluated the number of valley components as a function of the order for the flip, hull, and valley heuristics. The result is visualized in Figure 5 for the data sets Sphinx and Wren Peak; the graphs for the other data sets are similar. For the valley heuristic, the major reductions occur up to order 5.

	Quinn	flip-8 +v	hull-8 +v	Wren Peak	flip-8 +v	hull-8 +v
Edges	5210	5210	5210	5185	5185	5185
Valley edges	686	762	641	743	798	712
Local minima	35 (12)	20 (10)	19 (12)	31 (16)	19 (12)	19 (16)
Valley components	147	189	144	167	208	169
Not min. ending	102 (5)	148 (13)	115 (5)	112 (16)	161 (19)	126 (16)

	Sphinx	flip-8 +v	hull-8 +v	Hot Springs	flip-8 +v	hull-8 +v
Edges	5179	5179	5179	5234	5234	5234
Valley edges	597	729	565	790	895	759
Local minima	20 (16)	11 (11)	9 (16)	28 (19)	18 (16)	16 (19)
Valley components	157	212	155	169	187	161
Not min. ending	125 (5)	191 (4)	133 (6)	118 (13)	148 (11)	123 (12)

Table 3: Statistics for four terrains when applying the valley heuristic (order 8) to the Delaunay triangulation and the order-8 outcome of the flip and hull heuristics.

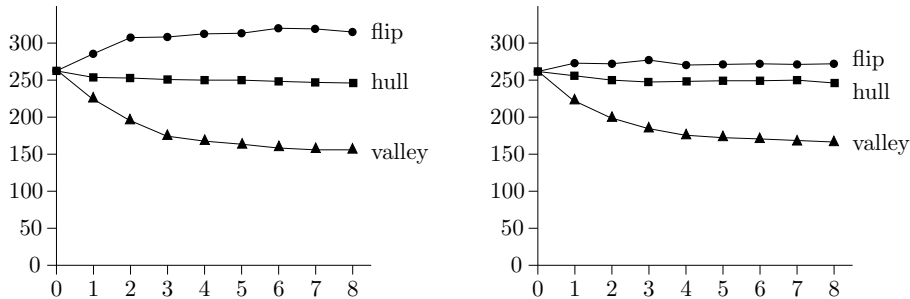


Figure 5: The number of valley edge components as a function of the order, for the Sphinx and Wren Peak data sets.

The visualization of one terrain and five outcomes of the heuristics is given in Figures 6 and 7 for the Sphinx and Wren Peak data sets. We again observe that the hull heuristic is better than the flip heuristic, but also that the valley heuristic helps to improve the visual quality of the valley edge components. A closer inspection shows that the hull heuristic with the valley heuristic applied to its outcome is generally best, but there are situations visible where the valley heuristic by itself is better than when preceded by the hull heuristic.

## 4 Conclusions and further research

We examined the computation of triangulations for realistic terrains using higher-order Delaunay triangulations. The realistic aspect is motivated by hydrologic and other flow applications, perhaps the most important reason for terrain modelling in GIS. Realistic terrains have few local minima and few valley edge components.

Theoretically, we showed that triangulating with the minimum number of local minima is NP-hard due to alignment of points. For order- $k$  Delaunay triangulations we obtain the same result in non-degenerate cases for  $k = \Omega(n^\epsilon)$  and  $k \leq c \cdot n$ . The case of constant orders but at least 2 remains open.

We presented two heuristics (one new, one old) to remove local minima, analyzed their efficiency, and implemented them. It turns out that higher-order Delaunay triangulations exist with considerably fewer local minima for low orders already, and the hull heuristic is better at computing them. We tested orders 0 up to 8. Often we obtain an optimal order- $k$  Delaunay triangulation. Also, the hull heuristic creates fewer valley edges and valley edge components than the flip heuristic. This answers the first four experimental research questions from the introduction.

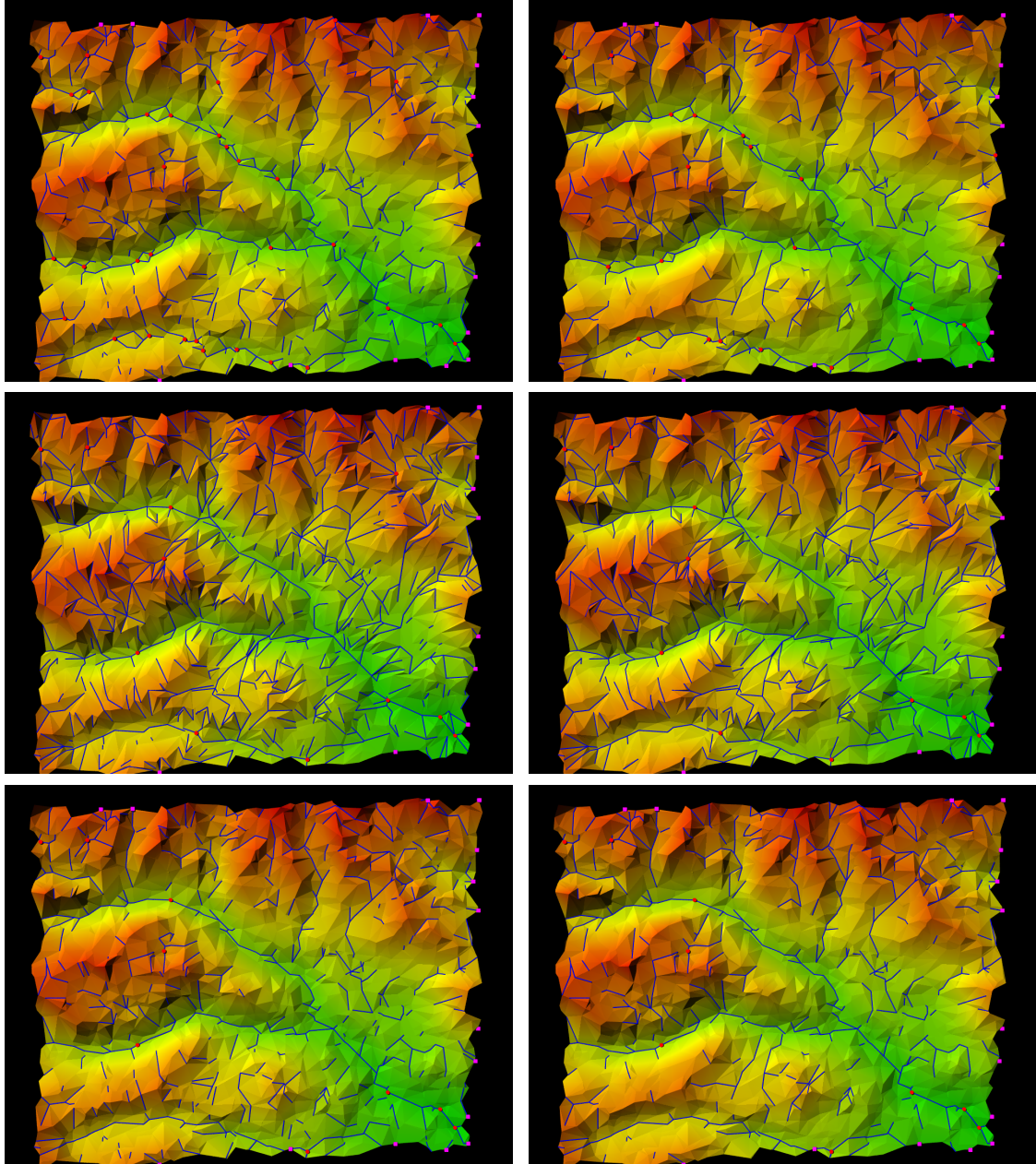


Figure 6: Visualization of the valley edges and local minima after applying the valley heuristic for the Sphinx data set. Left column: Delaunay triangulation, outcome of flip-8, outcome of hull-8. Right column: same, but after valley heuristic.

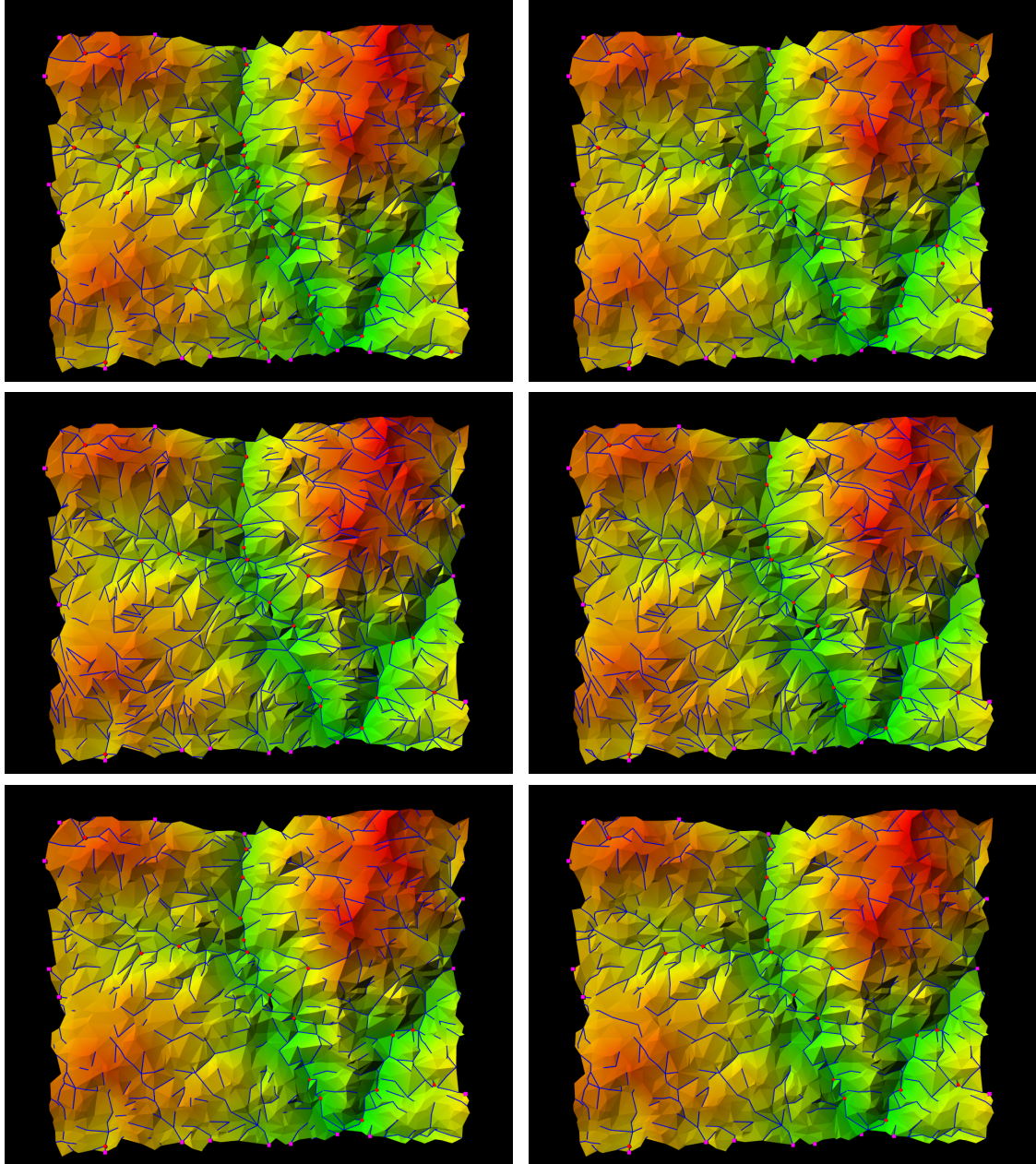


Figure 7: Visualization of the valley edges and local minima after applying the valley heuristic for the Wren Peak data set. Left column: Delaunay triangulation, outcome of flip-8, outcome of hull-8. Right column: same, but after valley heuristic.

We also presented the valley heuristic to reduce the number of valley edge components. The valley heuristic removes isolated valley edges and extends valley components down, hopefully to connect them to other valley components. The valley heuristic can be applied to any triangulation. The experiments and images suggest that the valley heuristic applied to the outcome of the hull heuristic gives the best results, largely answering the fifth experimental research question. We have not tested whether the reduction in local minima or valley components obtained by the heuristics is dependent on the sampling density, or level of detail, of the terrain. This is left for future work.

It is possible to devise a valley heuristic that inserts useful order- $k$  edges, similar to the hull heuristic, but now to reduce the number of valley components. Furthermore, it is possible to integrate minimizing local minima and reducing valley components in one heuristic. We leave this for future work.

It would also be interesting to extend the research to computing high quality drainage basins, or a good basin hierarchy. However, it is not clear how this quality should be defined, nor how it should be combined with local minima and valley components used in this paper. Finally, other criteria than higher-order Delaunay triangulations can be used to guarantee a good shape of the triangles. Again we leave this as future research.

## References

- [1] M. Abellanas, P. Bose, J. García, F. Hurtado, M. Nicolàs, and P.A. Ramos. On properties of higher-order Delaunay graphs with applications. In *Abstracts of the 21st Europ. Workshop on Comput. Geom.*, pages 119–122, 2005.
- [2] P.K. Agarwal and N.H. Mustafa. Independent set of intersection graphs of convex objects in 2D. In *Proc. SWAT 2004*, number 3111 in LNCS, pages 127–137, Berlin, 2004. Springer.
- [3] M. Bern, H. Edelsbrunner, D. Eppstein, S. Mitchell, and T. S. Tan. Edge insertion for optimal triangulations. *Discrete Comput. Geom.*, 10(1):47–65, 1993.
- [4] M. Bern and D. Eppstein. Mesh generation and optimal triangulation. In D.-Z. Du and F. K. Hwang, editors, *Computing in Euclidean Geometry*, volume 4 of *Lecture Notes Series on Computing*, pages 47–123. World Scientific, Singapore, 2nd edition, 1995.
- [5] L. P. Chew. Constrained Delaunay triangulations. *Algorithmica*, 4:97–108, 1989.
- [6] M. de Berg, P. Bose, K. Dobrint, M. van Kreveld, M. Overmars, M. de Groot, T. Roos, J. Snoeyink, and S. Yu. The complexity of rivers in triangulated terrains. In *Proc. 8th Canad. Conf. Comput. Geom.*, pages 325–330, 1996.
- [7] A.U. Frank, B. Palmer, and V.B. Robinson. Formal methods for the accurate definition of some fundamental terms in physical geography. In *Proc. 2nd Int. Symp. on Spatial Data Handling*, pages 583–599, 1986.
- [8] J. Gudmundsson, M. Hammar, and M. van Kreveld. Higher order Delaunay triangulations. *Comput. Geom. Theory Appl.*, 23:85–98, 2002.
- [9] J. Gudmundsson, H. Haverkort, and M. van Kreveld. Constrained higher order Delaunay triangulations. *Comput. Geom. Theory Appl.*, 30:271–277, 2005.
- [10] M.F. Hutchinson. Calculation of hydrologically sound digital elevation models. In *Proc. 3th Int. Symp. on Spatial Data Handling*, pages 117–133, 1988.
- [11] S.K. Jenson and C.M. Trautwein. Methods and applications in surface depression analysis. In *Proc. Auto-Carto 8*, pages 137–144, 1987.
- [12] Y. Liu and J. Snoeyink. Flooding triangulated terrain. In P.F. Fisher, editor, *Developments in Spatial Data Handling, proc. 11th Int. Sympos.*, pages 137–148, Berlin, 2004. Springer.
- [13] M. McAllister and J. Snoeyink. Extracting consistent watersheds from digital river and elevation data. In *Proc. ASPRS/ACSM Annu. Conf.*, 1999.
- [14] C. Mitchell. *Terrain Evaluation*. Longman, Harlow, 2nd edition, 1991.
- [15] E.A. Ramos. On range reporting, ray shooting and  $k$ -level construction. In *Proc. 15th Annu. ACM Symp. on Computational Geometry*, pages 390–399, 1999.

- [16] B. Schneider. Geomorphologically sound reconstruction of digital terrain surfaces from contours. In T.K. Poiker and N. Chrisman, editors, *Proc. 8th Int. Symp. on Spatial Data Handling*, pages 657–667, 1998.
- [17] D.M. Theobald and M.F. Goodchild. Artifacts of TIN-based surface flow modelling. In *Proc. GIS/LIS*, pages 955–964, 1990.
- [18] G.E. Tucker, S.T. Lancaster, N.M. Gasparini, R.L. Bras, and S.M. Rybarczyk. An object-oriented framework for distributed hydrologic and geomorphic modeling using triangulated irregular networks. *Computers and Geosciences*, 27:959–973, 2001.
- [19] S. Yu, M. van Kreveld, and J. Snoeyink. Drainage queries in TINs: from local to global and back again. In M.J. Kraak and M. Molenaar, editors, *Advances in GIS research II: Proc. of the 7th Int. Symp. on Spatial Data Handling*, pages 829–842, 1997.