Automated Organization of Hierarchical Catchments in River Network Based Constrained Delaunay Triangulation

Liu YAOLIN, Martin MOLENAAR, Menno-Jan KRAAK, The Netherlands

Key words: Catchments, Hierarchy, Triangulation Network, Classification of River System.

ABSTRACT

Catchments of river networks are fundamental to the automation of flow-routing management in distributed hydrologic models and for the morphometric evaluation of river network structure.

Catchments ordering also act as a very important factor for the generalization. Each river network has an intrinsic hierarchical structure that can be described by various stream ordering procedures. River network can be schematized as sets of basic unit: links (river segments) and nodes (source nodes, outlet nodes and junction nodes). An algorithm is proposed for automated organization of hierarchical catchments from a database based on classification of river and constrained Delaunay triangulation network in this paper.

The proposed algorithm uses the strategy from highest order link to lowest links to construct the catchments of river networks. At first the river network will be ordered by Horton's classification. Then triangulation network of this ordered river network will be constructed. The river links will be constrained edges in the triangulation network. The hull of this triangulation network will act as the catchments area of the highest order link. The other catchments areas of all other links of each order in the river network will be constructed from higher order to lower order based on relations among links, skeleton lines between links, boundaries of catchments at higher order and the properties of triangles. The triangles in the triangulation network can be classified into several types based on their properties. These different types of triangles play an important role in analysis and building hierarchical catchments of river network. The algorithm has been tested in a test dataset.

CONTACT

Liu Yaolin
Department of Geo-Informatics
International Institute of Geo-Information Science and Earth Observation
7500 AA, Enschede,
THE NETHERLANDS
E-mail: yaolin@itc.nl

Martin Molenaar
Department of Geo-Informatics
International Institute of Geo-Information Science and Earth Observation
7500 AA, Enschede,
THE NETHERLANDS
E-mail: Molenaar@itc.nl

Menno-Jan Kraak
Department of Geo-Informatics
International Institute of Geo-Information Science and Earth Observation
7500 AA, Enschede,
THE NETHERLANDS
E-mail: Kraak@itc.nl

Automated Organization of Hierarchical Catchments in River Network Based Constrained Delaunay Triangulation

Liu YAOLIN, Martin MOLENAAR, Menno-Jan KRAAK, The Netherlands

1. INTRODUCTION

River catchments play an important role in river basin management and traditional map generalization. In generalization, the size of catchments will be one of important factors for selecting the streams from drainage network. Rechardson (1993) discusses the problem of river system generalization based on Strahler classification and Horton classification. The problems of formalization of catchments and catchments generalization based on based on FDS and Strahler classification are discussed in Molenaar and Martinez Casanovas (1996), Martinez Casanovas (1994) and Molenaar (1998). In this section, we mainly concentrate on discussing how to create hierarchical catchments of river network based on the Horton's river classification system and constrained Delaunay triangulation network. The river classifications are introduced briefly.

2. CLASSIFICATION OF RIVER SYSTEM

Each river system has an intrinsic hierarchical structure that can be described by different stream ordering procedures. This hierarchical structure can be utilized as the basis for a feature elimination procedure.

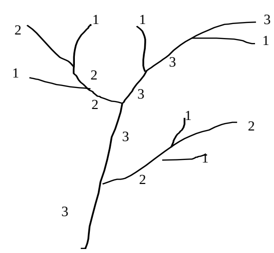


Figure 1 Horton Classification of river system

Stream order is measure of the position of a stream in the hierarchy of tributaries. The criteria for stream order enumeration, though, may be geometric, topologic, or volumetric. Two topologic criteria, magnitude and order, are considered. The magnitude of a given link establishes its position in relation to its directly adjacent links. Thus it is independent of the overall size of the system, of what portion is being enumerated, or of changes that might be made in the system. Order values, on the other hand, describe the link's position in relation to the entire river system. Straler's and Shreve's classifications enumerate stream magnitude which create hierarchic structure river network from the parts to create the whole.

Horton's system of ordering (Horton, 1945) has been shown to be the most useful for establishing a database structure that is amenable to an objective and simple generalization procedure that can be used in computerized data bases where the system may be required to generate different database at a quite different scales. In the system devised by Horton, unbranched fingertip tributaries are always designated as of order 1, tributaries or streams of the 2^{nd} order receive branches or tributaries of the 1^{st} order, but these only; a 3^{rd} order stream must receive one or more tributaries of the 2^{nd} order but may also receive 1^{st} order tributaries. A 4^{th} order stream receives branches of the 3^{rd} and usually also of lower orders, and so on. Using this system the order of the main channels is the highest. Horton assigned order one to the fingertip tributaries and the highest order to the main trunk. He wanted all fingertip tributaries to be of the same number of links. It is shown that in fact the Horton procedure can be used to answer both the questions of "How many" and "which ones". It provides a framework for the establishment of an objective and geographically oriented generalization scheme.

The ordering is only the first step in the quantitative analysis in database generalization.

3. REQUIREMENTS OF CREATING HIERARCHICAL CATCHMENTS OF RIVER

Creating hierarchical catchments area must meet several requirements as follow:

- River system must be complete.

- Catchments area of one order stream include all its branches' catchments areas at next lower order; in other word, each of its branch's catchments area is a part of the catchments area.

- A drainage network consists of a set of drainage links connected by network nodes. Three types of nodes are encountered in a drainage network: the outlet node, upstream tips of the drainage network where drainage links originate (source nodes) and points at which two or more channel links join (junction nodes).

- A stream is considered as basic unit, called as link, which are usually uniquely identified from their sources to their mouths and are fed by several tributaries.

- In order to creating hierarchical catchments, the Horton's classification need to be adjusted.

There will be only one highest order link in a river network according to Horton's classification, but there may be more than one link at each order of the other orders.

4. CONSTRAINED TRANGULATION AND TRIANGLE CLASSIFICATION

A Delaunay triangulation is generally defined as a triangulation W(N, E, T) of a set of points N with the empty circle property, that is, the circumcircle of any of its triangles $t \in T$ does not contain any point $n \in N$ (Preparata and Shamos, 1985). Here E is the set of all the triangle edges in the Delaunay triangulation. The Delaunay triangulation is unique and locally equiangular (Sibson, 1977), hence, it maximizes the minimum angle of its triangles compared to all other triangulations.

A constrained Delaunay triangulation (CDT) W(N, E, T, E_c) is an extension of the standard Delaunay triangulation by allowing pre-described, non-intersection line segments (except at their endpoints) E_c ($\subset E$) to be forced in as part of the triangulation. Note that triangle containing any of such pre-described edges may not be Delaunay triangles. Figure 2 shows examples of constrained and unconstrained Delaunay triangulation.

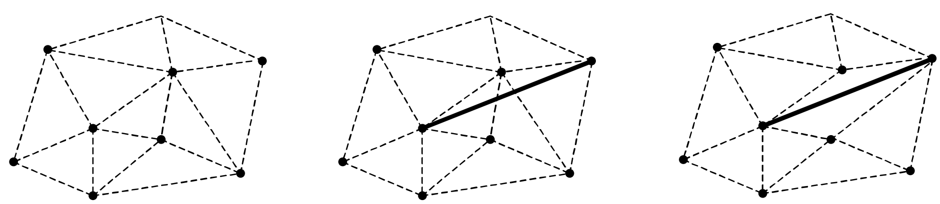


Figure 2 Example of DT and CDT (thick line =constraint)

The triangles in constraint Delaunay triangluation can be classified as four types through observation:

- Triangle having no constrained edge in its three edges, denoted as T1.
- Triangle having only one constrained edge in its three edges, denoted as T2.
- Triangle having two constrained edges in its three edges, denoted as T3.
- Triangle having three constrained edges in its three edges.

5. BUILDING METHODS FOR CATCHMENTS AREA BASED ON CONSTRAINED DELAUNAY TRIANGULATION

The hierarchical catchments of river system are established mainly through analyzing properties of triangulation networks of different orders of river links of river system. The semantic triangles and the extracting boundary lines based on the types of triangles play a key role in building hierarchical catchments.

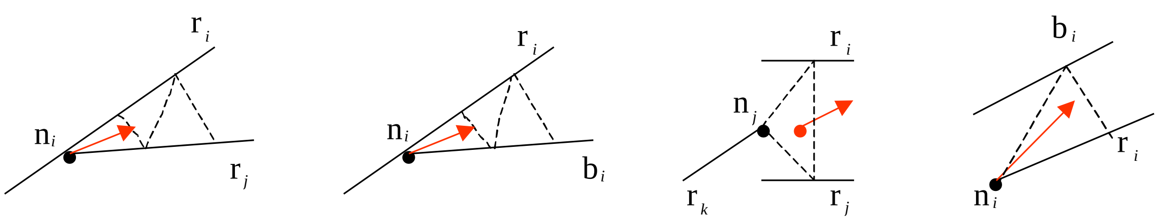
The main steps for automated constructing hierarchical catchments are as follows:

- Organize the river data and classifying the river systems based on Hortons' classification;
- Adjust Hortons's classification result;
- Triangulating ordered river systems;
- Constructing catchments of each order link respectively using the.

The river systems must be classified before constructing hierarchical catchments. For this example, the river systems are classified based on Horton's classification system and the number of the orders of whole river system can be gotten through the classification. The number of river orders will be equal to the number of catchments orders. This classification result must be adjusted using the following the rule, that is: if the order of a link is not the next lower order of its immediate adjacent link except highest order link, then change its order into the next lower order of its adjacent link.

The strategy of creating new catchments area is step by step from highest order link to lowest links. At first a catchments area of highest order link will be created, and then catchments areas of all links of each order are constructed gradually in order downward way until lowest order.

In order to extract boundary of catchments area, the characteristics of triangulation network of river system are analyzed. The types of linking points of boundary of catchments are identified based on types of triangles and source nodes, junction nodes and links of river. The starting points of boundary are shown in Figure 3 the linking middle points of boundary are shown in Figure 4 and the ending points are shown in Figure 4. The different types of linking points of boundary are extracted from these types of triangles.



n_i ----junction node, n_j ----source node, r_i and r_j ----river links, b_i ---boundary

----- linking starting point and direction

Figure 3 Examples of starting point of boundary

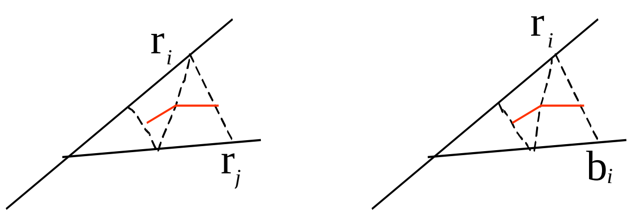


Figure 4 Examples of link middle point of boundary

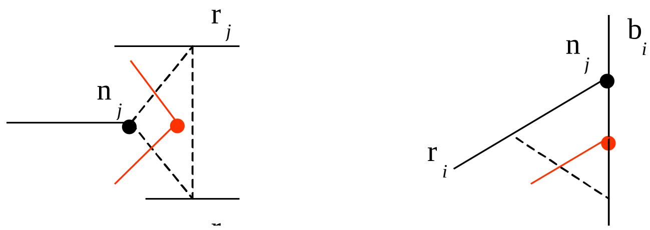
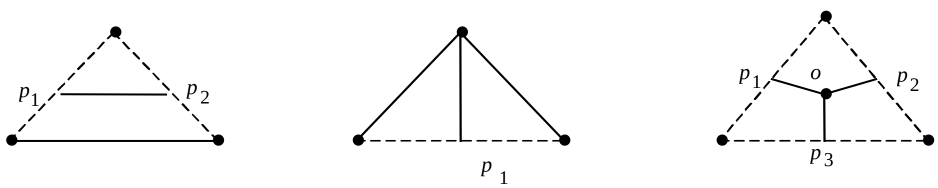


Figure 5 Examples of linking ending point of boundary

There are three different types of linking ways. The process of boundary generation starting with the one of points from Fig 3 and ending with the one of points in Fig.5 are below:

If ($t_i \in T2$) linking midpoints of two non-constrained edge of a triangle as shown in Figure 6(a).
Elseif ($t_i \in T3$) linking midpoint of constrained edge of triangle with opposition point of constrained edge as shown in Figure 6(b).
Else ($t_i \in T1$) linking the midpoint of each edge of a triangle with barycenter of the triangle as shown in Figure 6 (c).



(a) (b) (c)

Figure 6 Example of skeleton linking ways of three different types of triangles

Boundary line can be extracted through connecting three different types of triangles. An example presents in figure 7.

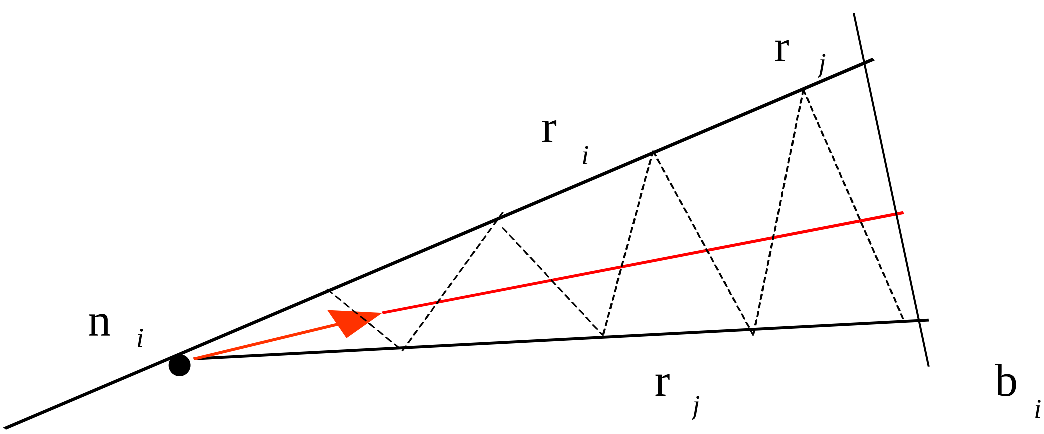


Figure 7 example of connecting boundary of Catchments

6. PROCESS OF CREATING HIERARCHICAL CATCHMENTS OF RIVER NETWORK BASED ON CDT

– Let *thenodeslist* be a node list and used to stores the nodes.
– Let *thecatchmentslist* be an empty catchment boundary list and used to store boundaries of catchments.
– Let *thelinkslist* be a link list and used to store river links.
– Let *maxorder*=number of order of river and *minorde*r=1.
– Get highest order link from *thelinkslist* and all nodes from *thenodeslist*, and form convex, and this convex area may be consider as catchment area of highest order link .
– Let *theorder*=*maxorder*-1.
– From *theorder* to *minorder*, do the following:
    {
    – Get all link objects at the *theorder* level and the link objects on the level directly above *theorder* from *thelinkslist*, and store them in *templinklist*, and get all the nodes of links whose order is lower than theorder from *thenodeslist* and store them in *tempnodeslist*, and get all boundary of catchment area from *thecatchmentslist* and store the *tempcatchmentlist*.
    – Construct constrained delaunay triangulation based on the data of *teplinklist*, *tempnodeslist* and *tempcatchmentlist*.

    – For each link object *linkobject from templinklist*, do the following:
        {
        – For a considered link, use the procedure described in chapter 5 to get all its neighbor nodes neighbor links object and its neighbor boundary of catchments area at next higher or the higher order links;

        – Get a subset of triangles which are relative to the considered link object and its neighbor objects (including source nodes, junction (outlet) nodes and boundary of catchments area) and store then in the list *temptrilist*;

        – Find the triangles which have junction (outlet) node of the considered link object as their vertex and the considered link object as their constrained edges; push the triangles into the stack starttristack, The junction node will be starting point to trace the boundary of catchments area and the triangles will be starting triangles to trace catchments area;

        – If *starttristack is empty,* stop to trace the boundary of catachment area of the considered link object and move to next link object in *templinklist* if there are still link objects at the same level as *theorder* not to be processed; otherwisedo the following:
        {
            – Pop up an object *theobject* from *starttristack,* let *triobject*=*theobject*;

– Add the junction node of the considered link object to the list *tempboundarypointlist*

– For each triangle triobject$\in$ temptrilist, do the following

{

– If the triangle *object triobject* $\in$ tri3

{

– Compute the middle point of non-constrained edge of *triobject;*
– Add the middle point to *tempboundarypointlist*;

}

– If the triangle object *triobject* $\in$ tri2

{

– If one of three vertex of *triobject* is the source node of the considered link or the source node of the next lower /or next higher order link of considered link and the constrained edge is the boundary of catchments at next higher order link, do the following:
– {
  – Add one vertex of the triangle on the constrained edge to the list temp*boundarypointslist.*
  – Connect all points in temp*boundarypointslist* to form line object from junction node to last point through all middle points.
  – Add the line object to the list *tempcatchmentlist* and empty *boundarypointslist*.

  }

– If the vertex of the triangle is the source node of the link whose order is lower than the considered link and the constrained edge of the triangle is the considered link, then check if the corresponding junction node of the source node is on the considered link object . If the result is yes, then do nothing to the triobject;otherwise do the following :

  {
  – compute the middle points coordinates of unconstrained
  – edges of the triangle.
  – add the middle points to *tempboundarypointslist.*

  }

– if the vertex of the triangle is on the adjacent link /or source node of the adjacent link at the same order of the considered link or on the adjacent link at the next higher order and the constrained edge of the triangle is the considered link, then do the following :

  {
  – Compute the middle points coordinates of unconstrained edges of the triangle.

– add the middle points to tempboundarypointslist;
}

– If the triangle object *triobject* $\in$ tri1
{
– If one of three vertexes of *triobject* is outlet node of the lower order neighboring link of the considered link and the other two vertexes of *triobject* are on the two neighboring links at the same order as the considered link respectively; or

– If one of three vertexes of triobject is source node of the considered link and the other two vertexes of triobject are on the two neighboring links at the same order as the considered link respectively, do the following:
{
– Compute the coordinates of the center point of *triobject.*
– Add the coordinates to the list *tempboundarypointslist.*
}
– If one of three vertexes of triobject is source node of the considered link object, and one of the other two vertexes of triobject is on a neighboring link at next higher or the same order and another is on the boundary of the catchments area at the higher order link of the considered link or;

– If one of three vertex of *triobject* is source node of the considered link, and one of the other two vertexes of *triobject* is the source node of a neighboring link at next lower order and another is on the boundary of the catchments area at the higher order link of the considered link or;

– If two of three vertexes of *triobject* are on source node of two next lower order neighboring links of the considered link and another is on the boundary of the catchments area at the higher order link of the considered link or;

– If one of three vertexes of *triobject* is source node of considered link and the two other vertexes are on the different boundary of the catchments at the higher order link of the considered link respective or;

– If one of three vertexes of *triobject* is source node of a neighboring lower order link of the considered link and the other two vertexes are on the neighboring links of the considered link and the considered link respectively; do the following:
{
– Add the vertex of *triobject* on the boundary of catchment toThe list *tempboundarypointslist.*

– Connect all points in *tempboundarypointslist* to form boundary object from junction node to last point through all middle points.
– Add the line object to the list *tempcatchmentlist;* empty *tempboundarypointslist;*
}
If one of the vertexes of *triobject* is on the considered link and the
– other two vertexes are source nodes of two neighboring lower order links of the considered link respectively or;

– If one of the vertexes of *triobject* is on the considered link and the
– other two vertexes are source and outlet node of neighboring lower order link of the considered link respectively or;

– If one of three vertexes of *triobject* is source node of a lower order
– neighboring link of the considered link and the other two vertexes of *triobject are* on two neighboring links of the considered link, then check if the corresponding junction node of the source node is on the considered link. If the result is yes, then do nothing to the *triobject*; otherwise do the following:
{
– Compute the coordinates of the center point of *triobject.*
– Add the coordinates to the list *tempboundarypointslist.*
}
– If three vertexes of *triobject* are source or outlet nodes of lower order links compared to the considered link, then do nothing.
}
}
}

– Add *tempcatchmentlist* to the list *catchmentlist.*
– Empty *templinklist,tempnodeslist tempboundarylist* and *tempcatchmentlist.*
}

– Let theorder=order-1 and repeat the above steps.
}

7.TEST

Triangulating the river system using rivers (links) as constrained edge. The area of the catchments of main river (highest order) will consist of the hull which is be constructed by constrained triangulation network of river system. The area of the catchments of other order links will be constructed by the relation among the given river link, a set of related triangles and boundary line between given river link and its adjacent links or its boundary of previous catchments using the procedure described respectively. The Boundary line can be extracted using the following method.

The experimental data used in the example is simulated data. The purpose of this example test the algorithm and illustrate the process of constructing hierarchical catchments of river systems.

Figure 8 gives test data of river system and Figure 9 is its corresponding order classification system.

Figure 8 Example of river system

Figure 9 Order of the river system

Figure 10 shows the catchments of order 4 of river. Figure 11 gives the catchments of order. Figure 12 presents the catchments of order 2 and Figure 13 shows the catchments of order.

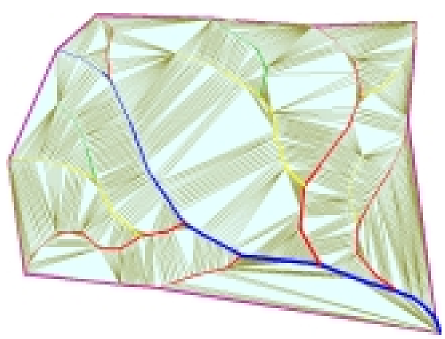


Figure 10 example of catchments of order 4

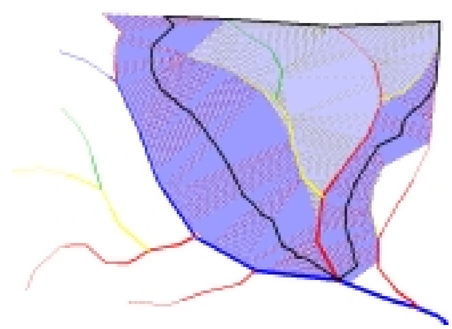
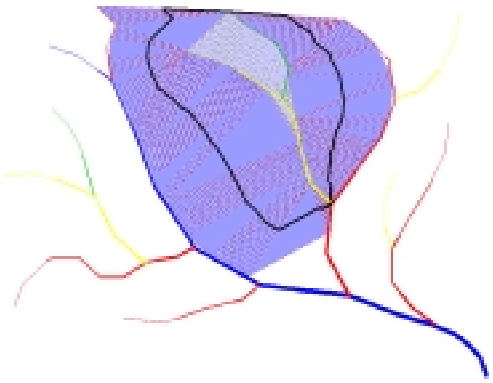


Figure 11 example of catchments of order 3



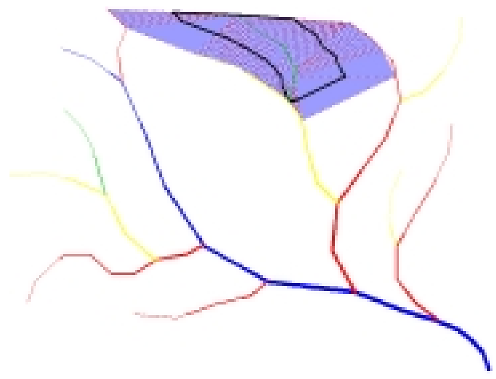


Figure 12 example of catchments of order 2 **Figure 13** example of catchments of order 1

8. CONCLUSION

The algorithm that automatically builds catchments area of river network based on constrained Delaunay triangulation and classification river system. Before building catchments of river system, the river system must be classified based on Horton's classification in order to get the number of levels of catchments area. To create catchments area more effectively and efficiently, this Horton's classification has been adjusted. The characteristics and tapes of triangles in triangulation network of river are analyzed and classified. Characteristic points of boundary of catchments are extracted based on the characteristics and tapes of triangles such as starting points , linking middle points and ending points. These points are used to form boundary of catchments of river. The method of creating catchments area in this study mainly focus on the geometric aspects and does not touch the aspects of river. This will be future research work.

This research work is supported partly financially by ITC, Ministry of Education of P.R. China and National Bureau of Survey and Mapping in China

REFERENCE:

Horton. R.E. (1945) Erosional development of streams and their drainage basins, Hydrophysical approach to quantitative morphology. Bulletin of the Geological Society of Amerca 56, 275-370.

Martinez Casanovas, J.A., 1994, Hydrographic Information Abstraction for Erosion Modleling at a regional level. MSC. Thesis, Department of land surveying and remote sensing. Wageningen: Wageningen Agriculture University.

Molenaar, M and Martinez Casasnovas, G.,A., 1997, A formalism for the structural description of vesctor maps and its use for multi-scale representations: a hydrographic example. Cartographica, 33, 55-63.

Molenaar,M. 1998,An Introduction to the Theory of Spatial Object Modelling for GIS, Taylor & Francis Ltd. London.

Jurgen Garbrecht and Lawence W. Martz 1997, Automated Channel Ordering and Node indexing For Raster Channel Networks. Computer & Geoscience Vol.23, No,9. pp961-966

Rusak Mazur and H.W. Castner, 1990, Horton's ordering scheme and the generalization of river network. The Cartographic Journal Vol. 27, pp.104-112.

Jones, C.B., Bundy. G.L., and Ware,J.M. 1995, Map generalization with a triangulated data structure. Cartographic and Geographic Information systems 22(4) 317-331.

Jones,Chris B. and Ware, M.J., 1998, Nearest neighbor search for linear and polygonal objects with constrained trianglations, 8^{th} International Symposium on Spatial Data Handling, Canada pp. 13-19.

Jones,Chris B. and Ware, M.J., 1998, Nearest neighbor search for linear and polygonal objects with constrained trianglations, 8^{th} International Symposium on Spatial Data Handling, Delft, The Netherlands, pp. 13-19.

Preparata, F., P. and Shamos, M.I. (1985) Computational geometry: an introduction, Springer,New York.

Sibson, R 1977 Locally equiangular triagulations. Comput.J., Vol.21, No.3, pp 243-245.