Geometry-Type Change in Model Generalization – A Geometrical or a Topological Problem?

Matthias Bobzien and Dieter Morgenstern
Institute of Cartography and Geoinformation
University of Bonn, Germany
`bobzien@ikg.uni-bonn.de`
`mostern@ikg.uni-bonn.de`

May 31, 2002

Contents

1 Introduction

A digital landscape model (DLM) is a model that represents an alphanumeric description of a landscape. For a DLM one requires a concrete definition of classes of landscape features and specifications of conditions when and where feature objects have to be recorded into the DLM. A DLM contrasts with a digital cartographic model, which presents a pictorial, scale-related description of the landscape [MS84].

Usually a DLM is not scale related. However, due to resource limitations and the need for efficient data analysis, the amount of data has to be reduced. For the area of Germany, the national mapping agencies have built the ATKIS system with its three different levels of detail — DLM 25, DLM 250 and DLM 1000 [AdV89]. A fourth one (DLM 50) is in progress.

The data acquisition for the DLM of lower level detail can be done either by direct acquisition from a generalized map, which requires an enormous additional and manual effort; or via an automated derivation from a DLM of higher level detail. The latter process is called a *model generalization*.

Over the last years, we have developed a conception for automated model generalization as well as an implementation of this conception. A good overview can be found in [MS99], a more detailed account in [Sch02, Bob00].

A model generalization consists of several steps, which concern semantic, topological and geometrical properties of feature objects. We concentrate on that aspect in model generalization which comprises in one step both topology and geometry.

2 Geometry-Type Changes

Each feature object of a DLM belongs to exactly one of three possible *geometry-types*: to area, line or point. For example, a lake would be modelled as an area, a border as a line and a telegraph pole as a point. However, some feature objects can be modelled by two or more geometry types. The choice of modelling these feature objects depends on the level of detail of the corresponding DLM. A river, for example, may be modelled as an area or as a line, a bridge may be modelled as a line or a point, and a building may be modelled as an area or a point.

In these latter cases the task of model generalization is to derive one geometry type from the other. This process is called *geometry-type change* [MS99]. Geometry-type changes usually occur from area to line, from line to point or from area to point. These changes are associated with a loss of geometrical information. Thus the reverse changes (from point to line or area, from line to area) require additional geometrical information which cannot be derived from a DLM of the given level of detail. In a consistent set of DLM, such reverse geometry-type changes should not occur. The first three type changes can be handled by automated model generalization and for this reason are dealt with in this paper.

3 Disadvantages of Purely Geometrical Algorithms

Several publications discuss a number of different approaches for solving the problem of geometry-type changes. A good overview is presented in [Sch02]. For example, a change from area to line can be achieved by calculation of the medial axis; a change from area to point by calculation of the centre point of gravity (centroid). Some algorithms are general approaches, others solutions that suit particular feature classes.

All algorithms that have been found so far are deficient in the following point: They act on isolated objects only. But objects are not isolated: The consistency of a DLM is based on several properties that hold *between* feature objects, and an algorithm calculating a geometry-type change for a DLM has to take into account these properties to preserve consistency. For our purpose the following two properties are important:

- **Topological Relationships:** These include for example *connectivity* between roads, railways and bridges; between a railway station and a railway line; between a power line and a pylon.
- **Area Covering:** Each part of the ground (based on the extent of the DLM) should be assigned to a feature object. Thus there should be no 'hole' in the model.

Neither the topological relationships nor the area covering must be destroyed during the process of generalization, and, in our context, by geometry-type change. Let's have a look on two examples to motivate our perspective on geometry-type changes.

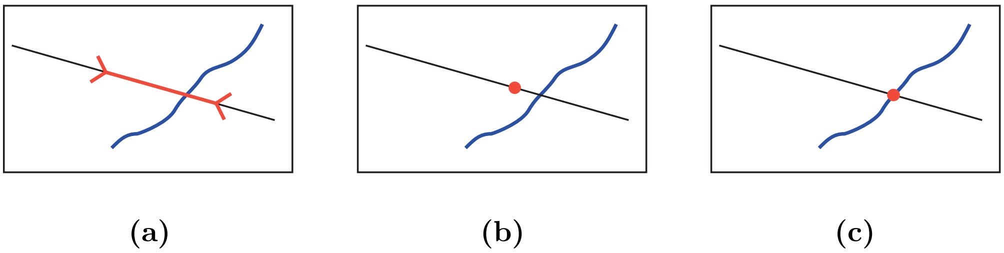


Figure 1: *Geometry-type change of a bridge (red) from line to point.* **(a)** *before generalization* **(b)** *geometry-type change calculated using the centre point* **(c)** *desired result.*

Figure 1 (a) shows a bridge modelled as a line, running across a river. Additionally, a road runs over the bridge. Assume that the bridge has to be reduced to a point. A purely geometrical algorithm, such as calculating the centre point of the line, would lead to a dissatisfying result (b) since the bridge is not located above the river anymore. The desired result is shown in (c).

Figure 2 (a) depicts a railway station modelled as an area. A railway track is leading through the station. Assume that the railway station has to be reduced to a point. A geometrical algorithm for example could calculate the centroid of the area of the station (b). This solution lacks in two points: First, the railway

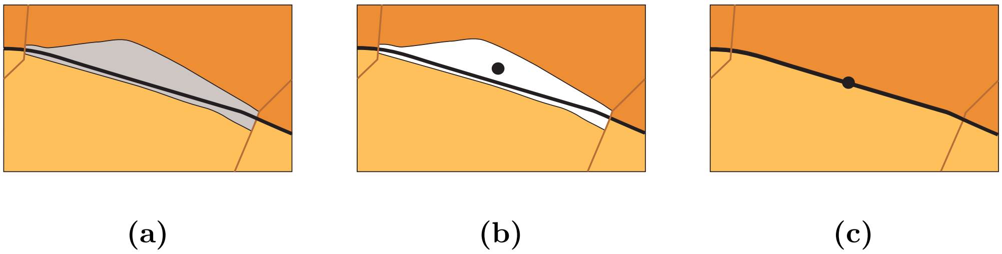


Figure 2: *Geometry-type change of a railway station (grey) from area to point.* **(a)** *before generalization* **(b)** *geometry-type change calculated using the centroid* **(c)** *desired result.*

station is not situated on the railway track which is strange enough, second, the former area of the station results in a hole in the landscape model. The desired result is shown in (c) where the station lies on the track and the surrounding feature objects cover the former area of the station.

To avoid the dissatisfying results of situations as shown above, and to preserve topological relationships and area covering we have developed

- an appropriate *data model* to express the relationships between feature objects within one DLM as well as between two DLMs of various levels of detail, and
- an *algorithm* to preserve the mentioned properties of a DLM.

The next section introduces the data model, the following sections explain the algorithm.

4 Data Model

For efficient analysis of topological relationships we have chosen an object-oriented and *topological data model* [Bob00]. Topology is modelled explicitly to avoid tedious extraction of topological relationships from geometry.

In figure 3 our data model is depicted. The columns represent the different geometry types, the rows specify semantic, topological and geometrical layers. The semantic layer contains the feature objects and their attributive information. The topological layer contains faces, edges and nodes as used in graph theory [Eve79]. An edge is connected to two nodes (end nodes) and to two faces (left and right). A node can be connected to an arbitrary number of edges whereas a face is surrounded by at least two edges. The geometrical layer is positioned below the topological layer. It consists of points and lines. Each line has two end points and is related to exactly one edge. Each point is connected to an arbitrary number of lines and not more than one node.

In model generalization at least two DLMs are involved. Each of them is of the type shown in figure 3. Now we have to express associations between them. The associations obviously should exist on the semantic layer. For each feature object of the target DLM we have to state the corresponding feature object(s) of the source DLM. The question is: Is it possible and useful to

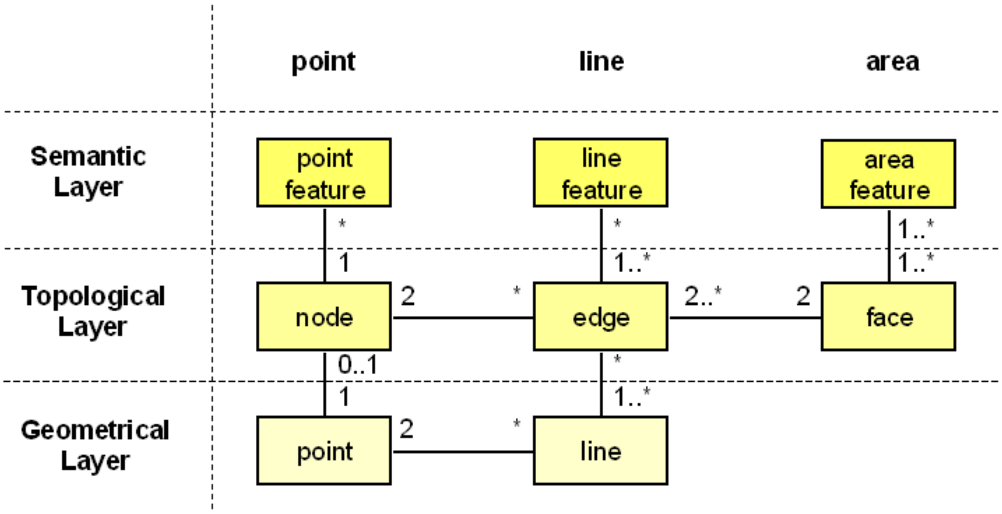


Figure 3: *Topological Data Model [Bob00]. Notation in universal modelling language [RJB99].*

connect the DLMs on the other layers, too? On the one hand the connection is useful (and possible) on the topological layer because relationships between topological elements of two DLMs allow a fast analysis of properties needed for the generalization process. On the other hand it is not useful (and maybe even hard to carry out) to connect the geometrical elements of two DLMs because the geometry of a generalized feature object often has no obvious counterparts in the source DLM, especially if complex generalization algorithms are applied.

The next question is: How can we handle geometry-type changes? The answer is given in figure 4: The modelling of relationships between feature objects of different geometry types is quite obvious. An area object may be generalized to an area, line or point object. A line object may be generalized to a line or point object. A point object may be generalized to a point object only. (Recall the above examples to recognize the bridge and the railway station within this model.)

Relations between topological elements are feasible between a face in the source DLM and a face, edge or node in the target DLM; between an edge in the source DLM and an edge or node in the target DLM; and between two nodes in both DLMs. Taking up the examples again, the edges representing the bridge have to be generalized into the node representing the bridge as a point. The same holds for the railway station.

The next section sketches the algorithm which constructs the above given relationships during geometry-type changes.

5 Topological Shrinking

The key idea of our algorithm is a two-step calculation called *topological shrinking*. The first step is the determination of the resulting topology and geometry depending on topological relations. The second step is to let all topological

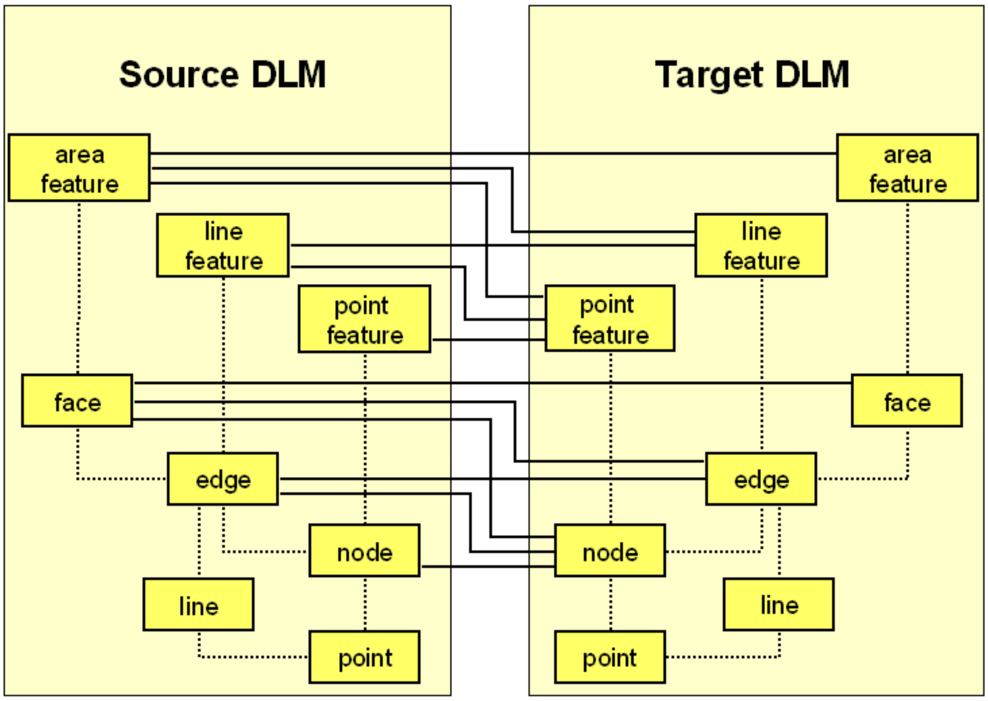


Figure 4: *Data model of two connected DLM with explicit modelling of geometry-type changes. Cardinalities of relations are left out for clarity reasons.*

elements of the source object *shrink* to its desired form, which is the previously determined topology and geometry. This has to be done by a series of elementary merging operations, which are

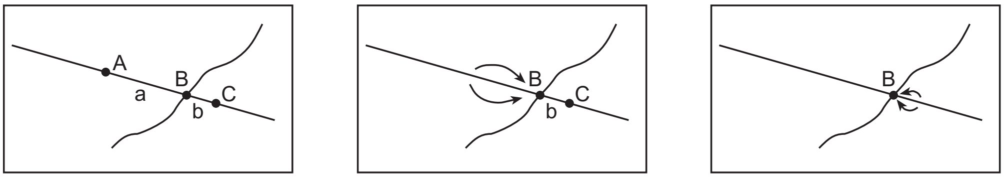
1. merging two nodes that are connected by an edge, and

2. merging two edges that form the boundary of a face.

Operations of type 1 merge two nodes into one of them. The connected edge disappears. One can say that both nodes *and* the edge are being generalized into *one* node. Similarly, operations of type 2 merge two edges into one of them. The bounded face disappears. One can say that both edges *and* the face are being generalized into *one* edge.

Let us illustrate the algorithm by the two examples in section 3. We omit the signatures and concentrate on the topological elements.

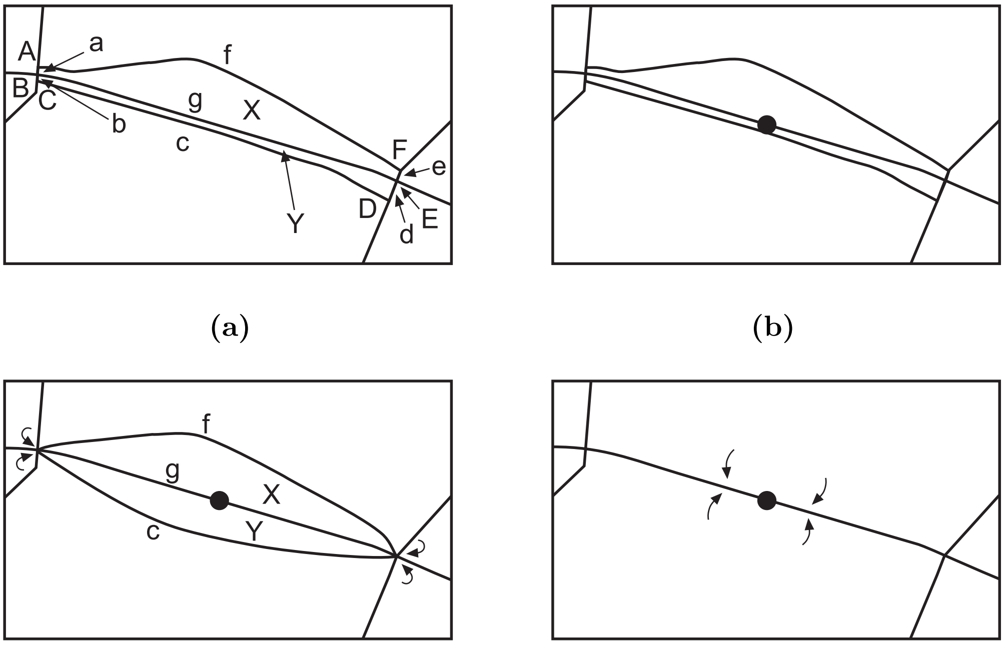
Figure 5 (a) shows the same situation as in figure 1. The bridge, the road and the river are reduced to their topological elements: The nodes are labelled A, B and C and the edges are labelled a and b. The first step of the algorithm is to determine the target topology and geometry. In this case this has to be done by analysis of vertical references: which object is underneath the bridge and which is situated above. The crossing point of the road and the river will be the resulting topology and geometry and hence is node B and its geometry. The second step consists of the shrinking. The remaining nodes have to be merged into B. In this example the order is insignificant. In a first iteration,



(a) **(b)** **(c)**

Figure 5: *Topological shrinking of a line object into a point object.*

let us merge nodes A and B into node B. Edge a disappears, so A, B and a are generalized into B (figure 5 b). In a second iteration, nodes B and C are merged into node B. Edge b disappears and we get the desired result (figure 5 c).



(a) **(b)**

(c) **(d)**

Figure 6: *Topological shrinking of an area object into a point object.*

Figure 6 (a) shows the same situation as in figure 2. The railway station, the railway track and the surrounding feature objects are reduced to their topological elements: nodes A to F, edges a to g and faces X and Y. Again, the first step is to determine the target topology and geometry. In this example the location of the generalized railway station depends on the location of the railway track which is represented by edge g. This edge with end nodes B and E is called the *relevant topology* for this geometry-type change. The resulting geometry has to lie on the relevant topology. A simple geometrical algorithm, the calculation of the centre point of g, suffices our needs. The target position of the station is shown in (b). In the second step we have to shrink the topology. We start with the nodes. On the left side we merge A, B and C into B and on the right side we merge D, E and F into E. Edges a, b, d and e disappear. The result of these four single merging operations is shown in (c). The remaining edges c, f and g have to be merged into g by two merging operations. The faces X and Y disappear during these operations. The result is shown in (d).

These two examples show the principles of the algorithm of topological shrinking. Unfortunately, many feature objects do not fit into simple examples. That's why the general algorithm is more complicated. Especially correct ordering of the merging series requires some additional effort.

The determination of the target topology is heavily dependent on the semantics of the generalized feature objects. This observation has led to a rough classification of geometry-type changes that follows in the next section.

6 Classification of Geometry-Type Changes

The first part of topological shrinking is always the determination of the target topology and, following that, geometry. This determination has to be executed in various ways, depending on the semantics of the feature objects. An extensive analysis of feature objects and possible geometry-type changes has led to a classification into four classes of geometry-type changes. For each of these classes a specific procedure of topology determination had to be implemented. The first two of these four classes have already been introduced by means of the above examples. Examples of the remaining two classes follow below.

6.1 Vertical Reference

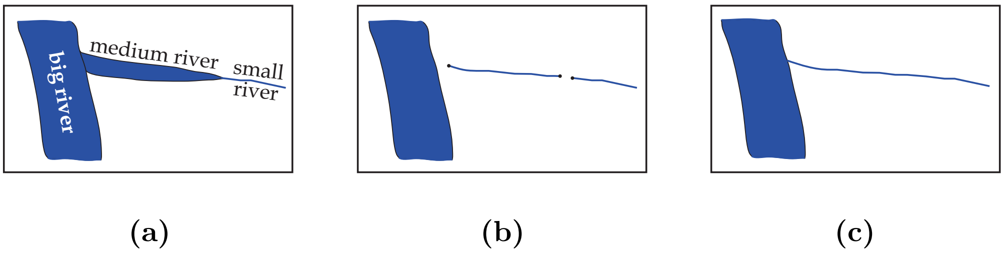
The target topology has to be calculated taking into account whether the feature objects lie above or below the current feature. Usually the target topology will consist of the crossing point of the two feature objects. Apart from bridges (see example) this class also applies to tunnels.

6.2 Inner Topology

Here the target topology depends on a topology that lies within the topology of the current feature. The target topology is either equal to this inner topology or has to be calculated from the corresponding geometry, as in the example of the railway station. The railway track is the *inner topology* of the station (before generalization). Another example for this class might be a small village within which major roads cross. Assuming the village to be generalized from area to point, the target topology would depend on the crossing of the major roads.

6.3 Outer Topology

If a feature is part of a network, the target topology will have to remain connected to that network. See figure 7 for an example: The medium river, modelled as an area, is topologically connected to the big river and to the small river (a). This situation should hold after generalization, too. Assume that the medium river has to be generalized to a line. A purely geometrical algorithm, such as calculation of the medial axis [Chr00], might result in (b). Here the network connections would be lost. The correct calculation has to take the topology of the connected features into account (c). Hence the target topology depends on a topology *external* of the medium river. Similar dependencies occur in road and railroad networks.



(a) (b) (c)

Figure 7: *Outer topology class:* **(a)** *before generalization* **(b)** *geometry-type change not taking into account network connectivity* **(c)** *geometry-type change taking into account network connectivity.*

6.4 Geometry

When a feature is isolated and independent of other feature objects, the target topology can be determined by a purely geometrical calculation. Examples are minings, power plants and oases (changes from area to point). Conventional geometry-type change algorithms can be applied. However, connected features, such as a road leading to a power plant, have to be adapted appropriately. See figure 8 for an example. The power plant, modelled as an area, is connected to a road (a). Geometric generalization of the power plant (here by calculation of the centroid) leads to (b) where the connection is lost. A better solution is presented in (c). The outer topology algorithm (section 6.3) is applied. It causes the position of the power plant to depend on the road (the outer topology). But this solution is still not adequate since the calculated position of the power plant doesn't reflect its original position anymore. This happens, because the dependency is vice versa: The outer topology is dependent on the current object and has to be adapted. This is done automatically by topological shrinking. The result is shown in (d).

7 Conclusion

Digital landscape models have several properties that must not be violated through the process of generalization. Two of these properties are: topological relationships and area covering. The process of geometry-type change may violate both of them. In particular, purely geometrical algorithms do not take the integrity of these properties into account. To model these properties (among others), a topological data model was developed. Central aspects of this data model are the topological elements and their relationships. These relationships hold not only within one DLM but also among two DLM.

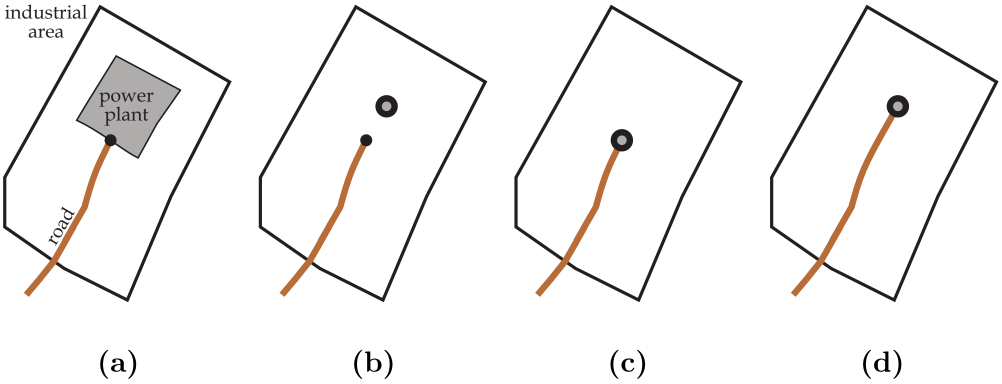


Figure 8: *Geometry class: The outer topology depends on the topology of the generalized object.* **(a)** *Situation before generalization* **(b)** *power plant generalized geometrically, road not adapted* **(c)** *power plant adapted to road, position not suitable* **(d)** *power plant generalized geometrically, road adapted.*

Based on the data model, the method of topological shrinking was developed. It performs geometry-type changes and preserves topological relationships as well as area covering. The algorithm consists of two consecutive steps. The first step is the determination of the target topology and the calculation of its geometry; the second step consists of a series of merging operations. The determination of the target topology is strongly dependent on the semantics of the feature objects. Therefore, four different classes of geometry-type changes were found that have led to four different procedures of topology determination.

Our investigations have shown that geometry-type change is not a problem that is purely geometrical. On the contrary, it is very much dependent on topological relationships. In some cases, this dependency is so significant that geometrical calculations are not even necessary. Hence, in the context of DLM, geometry-type change can rather be seen as more of a topological problem than as a geometrical problem.

References

[AdV89] AdV-Arbeitsgruppe ATKIS. *ATKIS-Gesamtdokumentation.* Arbeitsgemeinschaft der Vermessungsverwaltungen der Länder der Bundesrepublik Deutschland (AdV), Hannover, 1989.

[Bob00] Matthias Bobzien. Implementationsaspekte der Modellgeneralisierung. In *Mitteilungen des Bundesamtes für Kartographie und Geodäsie*, number 17, pages 15–25, Frankfurt am Main, 2000.

[Chr00] Albert H. J. Christensen. Line generalization by waterlining and medial-axis transformation. Successes and issues in an implementation of Perkal's proposal. *The Cartographic Journal*, 37(1):19–28, 2000.

[Eve79] Shimon Even. *Graph Algorithms.* Pitman, London, 1979.

[MS84] Robert B. McMaster and K. Stuart Shea. *Generalization in Digital Cartography.* Association of American Geographers, Washington D.C., 1984.

[MS99] Dieter Morgenstern and Dietrich Schürer. A concept for model generalization of digital landscape models from finer to coarser resolution. In *ICA-Proceedings 1999*, Ottawa, 1999.

[RJB99] James Rumbaugh, Ivar Jacobson, and Grady Booch. *The Unified Modeling Language Reference Manual.* Addison-Wesley, Reading, 1999.

[Sch02] Dietrich Schürer. *Ableitung von digitalen Landschaftsmodellen mit geringerem Strukturierungsgrad durch Modellgeneralisierung.* Dissertation, Institut für Kartographie und Geoinformation, Universität Bonn, Bonn, 2002.