Paper accepted for the 2nd International Conference on Interoperating Geographic Information Systems, 10-12 March 1999, Zurich, Switzerland

Table of Contents

Semantic and Geometric Aspects of Integrating Road Networks 1
Harry Uitermark (Dutch Cadastre), Anton Vogels (Dutch Cadastre), Peter van Oosterom (Dutch Cadastre)

Semantic and Geometric Aspects of Integrating Road Networks

Harry Uitermark, Anton Vogels and Peter van Oosterom

Dutch Cadastre, P.O. Box 9046, 7300 GH Apeldoorn, The Netherlands
Phone: +31 55 528 5163, Fax: +31 55 355 7931.
Email: {uitermark,vogels,oosterom}@kadaster.nl

Abstract. A prototype of a Geographic Database Integrator is under investigation and development. One of the long term goals of the Geographic Database Integrator is to reduce the need for operator intervention in update operations between objects in different databases. This paper focuses on the research related to road network elements from two independently surveyed and maintained topographic databases, one at large scale and one at mid-scale. Central to the issue of update propagation is certainty of equivalence of different road object representations. Therefore, precise definitions of road segment and road junction are important. In both the large and mid-scale geographic data sets, the roads are area features. Although the whole may be considered as one linear road network. In order to find the junctions and road segments, the constrained Delaunay triangulation is applied. Using these well defined elements, a strategy for finding equivalent or corresponding road network elements has been developed. This forms the basis for processing and propagating updates in the road network from one database to the other database.

1 Introduction: Scope, Context and Related Work

Geographic Database Integration is the process of establishing links between corresponding objects in different, heterogeneous and autonomously produced databases of a certain region [15]. The purpose of geographic database integration, in general, is to share geo-information between different sources. Sharing geo-information is a communication process. In communication the semantics or meaning of data is important and touches at the heart of interoperability. In this paper geographic database integration is being studied in the context of *update propagation*, that is the reuse of updates from one geographic database to another geographic database [20]. Geographic database integration gets more and more attention nowadays since the digitizing of traditional map series has ended. In these map series corresponding objects were only linked implicitly by a common reference system, the national grid [21]. In order to make these links more explicit, geo-science researchers and computer scientists have developed various strategies. In computer science *schema integration* has been the dominant methodology for database integration; see for example [11]. This approach has been extended for geographic databases; see [6] for an overview and see [2] for a fine example. Geo-scientists on the other hand have adopted methods from communication theory like *relational matching* [21] and, in our case, from the field of AI [14]. In our approach the construction and use of an *ontology* for geographic databases [9, 18] makes it possible to inspect the result of the geographic database integration process for *inconsistencies* [13].

But despite these advances in geographic database integration methodologies, there is still a problem that can not be solved by these methodologies alone, that is the demarcation of *homologous* entities, suitable for update propagation especially in the case of road networks; see Section 2. The remainder of this paper is organized as follows. Section 3 introduces the semantics of road networks, specifically the definitions of road segments and road junctions. In Section 4 it is demonstrated how these road segments and road junctions are demarcated by a constrained Delaunay triangulation algorithm. Also the relationship with the road center lines (skeleton) is discussed in this section. A six step update propagation method for road elements is given in Section 5. Note that a road element is a road segment or a road junction. Finally, descriptions of the conclusions and future work end this paper.

Fig. 1. The mid-scale topographic database TOP10vector.

2 The Demarcation of Homologous Entities

In previous research the propagation of updates in building objects (e.g. houses, building blocks, garages, annexes, etc.) has been studied [20]. Building objects share the property that they are demarcated quite naturally. In contrast, road elements are sometimes demarcated in a haphazard way; see Fig. 1 and Fig. 2.

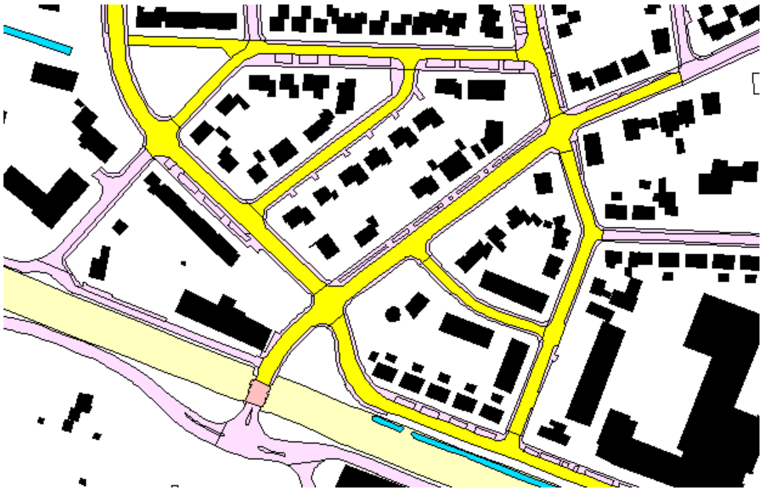

Fig. 2. The large-scale topographic database GBKN.

Here road elements from the mid-scale topographic database correspond to several different road elements from the large-scale topographic database. This n:m correspondence relationship is not suitable for pin-pointing an update from one database to another. So it is necessary to demarcate entities in such a way that 1:1 (or 1:n or n:1) correspondence relationships can be established; that means the demarcation of homologous entities.

3 Semantic Aspects of Road Networks

In general people might observe road networks, as a collection of line segments and nodes (junctions at point locations). In small scale geographic data sets roads are also represented as line features, but in large and mid-scale geographic data sets, the roads and junctions are represented as area features. In this research the GBKN (large scale base map, scale 1:1,000; see Fig. 2) and the TOP10vector (mid-scale map, scale 1:10,000; see Fig. 1) are used. In both geographic data sets roads are represented as area features. Although the roads are area features, it is also useful to think about the linear network topology, because the road network is a complex whole. It is not possible to look at one piece of the road network and forget about the other parts. A single change could affect multiple related road segments and road junctions.

As explained in the previous section, it is difficult to use complex road polygons for update propagation. Therefore, the road network is split in multiple elements. Before this can be done it is important to define these road elements in an unambiguous manner. The road element definition of the *Nationaal Wegenbestand* (NWB) [4] is used, which adheres to the European CEN standard Geographic Data Files (GDF) [5]. The definitions cover road segments and road junctions. A road segment is the reference unit on which users can put attribute information, road junctions are nodes which connect the segments. The NWB sees road segments as line features and road junctions as points. In the large and mid-scale geographic data sets the road segments and road junctions are area features, still it is tried to adhere the NWB as much as possible. The NWB declares that roads have to be divided into segments and junctions if three or more roads come together. Roads also have to be divided at places where the street name changes, or the maintenance is done by a different organization, or at the border of a village or a municipality.

This research focuses on the geometric aspect and the roads are just divided at the junctions. The road network now consist of two types of area features: road segments and road junctions. The NWB has detailed criteria for defining the type of a junction. When two T-junctions are close together, you have to treat them as one junction when one of the extended boundaries of a road lies between the boundaries of the road on the other side of the junction; see Fig. 3. Otherwise you get two independent T-junctions; see Fig. 4.

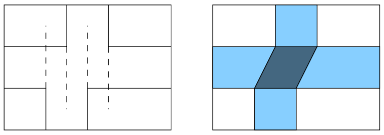

Fig. 3. These 'T-junctions' result in one road junction area and four road segments.

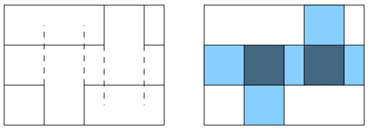

Fig. 4. These T-junctions result in two road junction areas and five road segments.

It is not possible to use GBKN information directly to update the TOP10-vector. Generalization and aggregation play a role in converting GBKN updates into TOP10vector updates; see Figs 1 and 2. In the GBKN speed ramps and small roundabouts are represented as small area features. The same is true for sidewalks and parking strips. None of these objects are represented in the TOP10vector. Updates in these small objects may not have any influence alone, but several small updates together could create an update which might have enough relevance to be propagated.

4 Geometric Aspects of Road Networks

The method used to find the junctions in the road network is based on triangulating the road area. This triangulation is used to compute a skeleton of the road. The nodes in the road skeleton define the location of the junctions and the edges of the surrounding triangle are used to separate the road network in road segment areas and road junction areas. The *constrained Delaunay triangulation* (CDT) algorithm described in [17] is used to compute the triangulation. A CDT over a planar set of n vertices together with a set of non-intersecting edges has the properties that all specified vertices and edges can also be found in the output and the result is as close as possible to the unconstrained Delaunay triangulation [8]. That is the circumcircle of any triangle has no vertex inside unless the vertex is at the other side of a constraining edge. In the case of the road network the set of separate input vertices is empty and the set of edges consists of the road boundary edges.

4.1 Constrained Delaunay triangulation

The applied algorithm runs in $O(n \log n)$ time, which is asymptotically optimal [10]. The algorithm is based on the concept of two other algorithms. The first algorithm is the unconstrained Delaunay triangulation (UDT) algorithm of Lee and Schachter [7] and the second algorithm is the CDT algorithm of Chew [1]. More details about the algorithm and its implementation can be found in [16].

In general the input of the CDT algorithm is a graph $G = (V, E)$ in which V is a set of n vertices (separate points and the end points of the input edges) and E is a set of edges, the so called *G-edges*. Two different kinds of edges appear in a CDT: G-edges, already present in the graph, and *D-edges*, created by the CDT algorithm. If the graph has no G-edges then the CDT and the UDT (unconstrained Delaunay triangulation) are the same. The applied algorithm is based on the *divide-and-conquer* paradigm. The graph can be thought of to be contained in an enclosing rectangle (the domain). This rectangle is subdivided into n separate vertical strips in such a way that each strip contains exactly one region (a part of the strip) which in turn contains exactly one vertex. After dividing the graph into n initial strips, adjacent strips are pasted together in pairs to form new strips. During this pasting new regions are formed of existing regions for which the combined CDTs are calculated. This pasting of adjacent strips is repeated following the divide-and-conquer paradigm until eventually exactly one big strip, consisting of exactly one big region, is left for which the CDT is calculated.

4.2 Interpreting triangles of the road network

The method used to derive the skeleton from the CDT is based on Wilschut *et al.* [22]. In the triangulation four different types of triangles cover the road area based on the number of G-edges in the boundary of the triangle: 0-triangle, 1-triangle, 2-triangle, and 3-triangle. The 3-triangle is an exception and does only occur when there is a non-connected road area with triangular shape in the input data set. A junction can be found by a triangle which has only D-edges and no G-edges, that is a 0-triangle; see the light triangles in Figs 5 and 6. A T-junction is a single 0-triangle with no neighbor 0-triangles. A normal crossing (4-way junction) is defined by 2 adjacent 0-triangles; e.g. the junctions at the bottom center in Fig. 6. In general a n-way junction is defined by (n-2) adjacent 0-triangles.

The 1-triangles form the building blocks of connecting road segments. Finally, the 2-triangles define the end points of the road network, that is, the dead-end

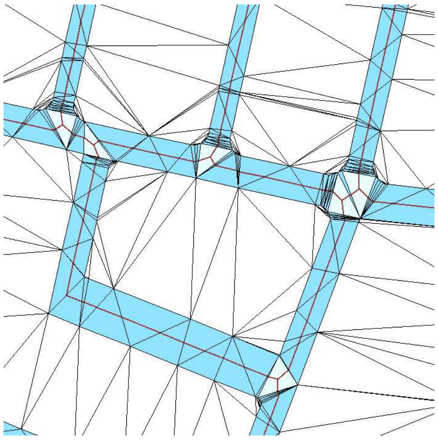

Fig. 5. Triangulated road network

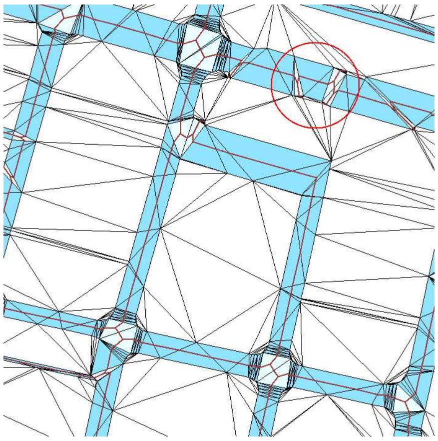

Fig. 6. Another road network

streets. Also, a small lump in the boundary of a road segment may result in 2-triangle and therefore in a small dead-end street; see the top right road segment in Fig. 6. A solution for this 'problem' is to (virtually) remove the 2-triangle smaller than a certain threshold area adjacent to a 0-triangle. In such a case, also the original 0-triangle becomes a (virtual) 1-triangle, that is, part of a connecting road segment and not a junction. Further, due to the fine distribution of vertices on the road boundary, the two 0-triangles defining a 4-way junction may not be topologically adjacent. In this situation another approach is needed to couple the two 0-triangles for one 4-way junction: if the distance between the center points of the two 0-triangles is less than a certain threshold, then the 0-triangles can be coupled. Note that this is also true for n-way junctions in general.

The separation (demarcation) between road segments and road junctions is defined by the edges of 0-triangles and leaving out the shared edges of 0-triangles in case of n-way junction with $n > 3$. A last point of attention is the location of the junction (point) and the separation edges. In case of the T-junction, there may not be a close vertex at the road boundary at the other side of the road; see Fig. 7: in the top horizontal road, the junction is about 30 meters to the east of the actual location of the junction. This may be solved by adding additional (intermediate) vertices within long road boundary edges; whenever they are longer than a certain maximum length (e.g. the average width of a road, about 15 meters); see Fig. 8. Note that adding additional vertices will increase the computing resources (memory and time) during the triangulation and subsequent processes. First applying line generalization may reduce the required computing resources during the triangulation. It has also the advantages that it removes some virtual 2-triangles and that close, but no direct neighbor 0-triangles, may become direct neighbor 0-triangles (beneficial for finding 4- and higher way road junctions); see Fig. 9.

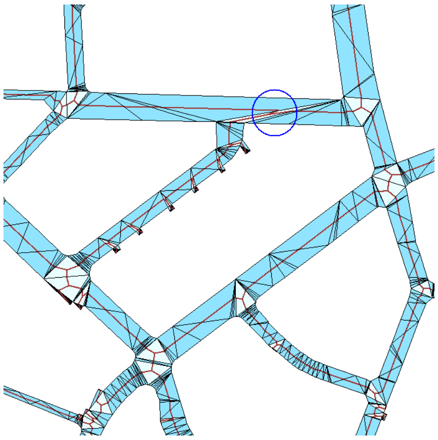

Fig. 7. Road junction with displaced 0-triangle (center node).

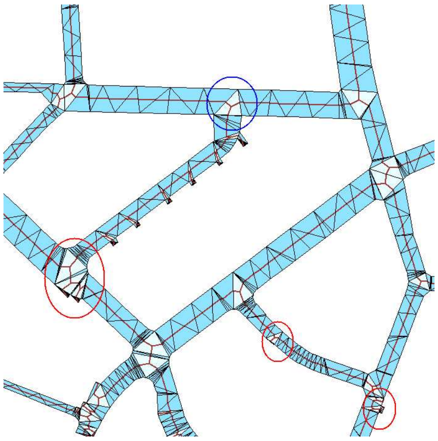

Fig. 8. Add more intermediate points: better 0-triangles.

4.3 Road center lines

Once the 0-, 1-, and 2-triangles are obtained it is not only possible to find the road junctions and dead-end roads, but it is also relatively easy to find a skeleton of the road. That is, the corresponding linear network based on the road center lines. The construction of the skeleton is based on following the middle of the internal edges of the 1-triangles. In a 0-triangle the center of the triangle is connected to the middle of all three edges. In a 2-triangle the middle of the D-edge is connected to the common point between the two G-edges, that is, the end-point of the road. Finally, this method needs some post processing in order to remove the 'dip' in a straight road in T-junction and also to make one center point of 4-way junction instead of two connected center points of T-junctions.

A general method for solving these problems is described in detail by Gao and Minami [3]. Their method is based on looking at the *trend-lines* of the parts of the center lines within a certain radius around the node (or average of a group of nodes within radius distance of each other). A pair of trend-lines with nearly the same angle is replaced by a straight line connecting the two corresponding center lines. In case there are more pairs of trend-lines with nearly the same angle, then the intersection(s) of the corresponding straight line connections is (are) computed. The other center lines are connected through their trend-line to the straight line. The location of the junction node is the (average) intersection point, to which all center lines are connected.

The method, described in this paper, to obtain center lines is a vector based approach. It assumes a topologically correct input of the road boundaries. In case the vector data is inaccurate, a raster based approach may be more appropriate. A description of this method is given by Thomas [12], in which the raster is represented by a compact run-length encoded binary image.

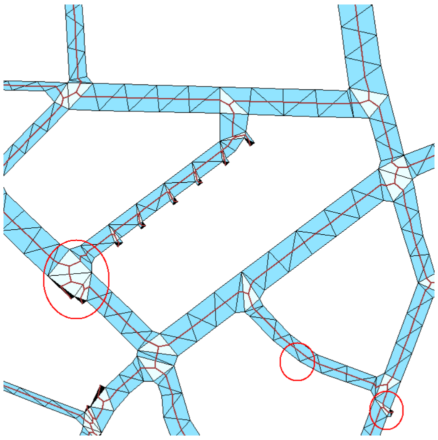

Fig. 9. Apply line generalization: less fake 0-triangles

5 Update Propagation

As explained before road update propagation is different from building update propagation. The most important difference is that buildings are usually unattached to other objects, whereas road segments are connected to road junctions and other road segments; it is a complex whole even after the road polygons are divided into road segments and road junctions. To reduce the complexity and find the 1:1 (or 1:n or n:1) correspondences in order to propagate the relevant updates we propose the following six steps:

- *Step 1: Synchronization of GBKN and TOP10vector to the same moment in time.* In general every 6 to 12 months updates in the field are measured and used to update the GBKN. On the other hand the TOP10vector is updated every four years. Before updates can be propagated to the other database, the two databases have to be synchronized. Synchronization means rolling back the GBKN in time until its date is the same as the TOP10vector date. This is possible because every object in the GBKN database has two time stamps: Tmin and Tmax[19]. Tmin is the date an object has been added to the database. Tmax is the date an object has been replaced by one or more other objects. These 'old' objects remain in the database, but are not 'valid'. If you bring back the database in time, the old objects become valid and represent the desired moment in the past.
- *Step 2: Create road segments and road junctions areas in the GBKN and the TOP10vector.* This step is based on the constrained Delaunay triangulation and is explained in the previous section.
- *Step 3: Find corresponding road segments and the road junctions areas between GBKN and TOP10vector.* The correspondences between road elements is found by computing the overlap between the elements from both the

GBKN and TOP10vector databases; the method is explained in [20] and is similar to finding correspondences between building updates.

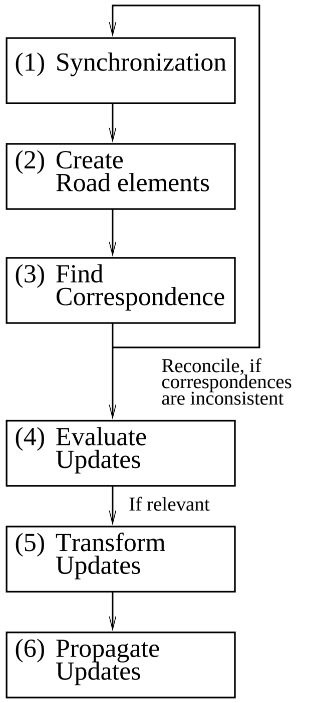

Fig. 10. Overview of the update propagation steps

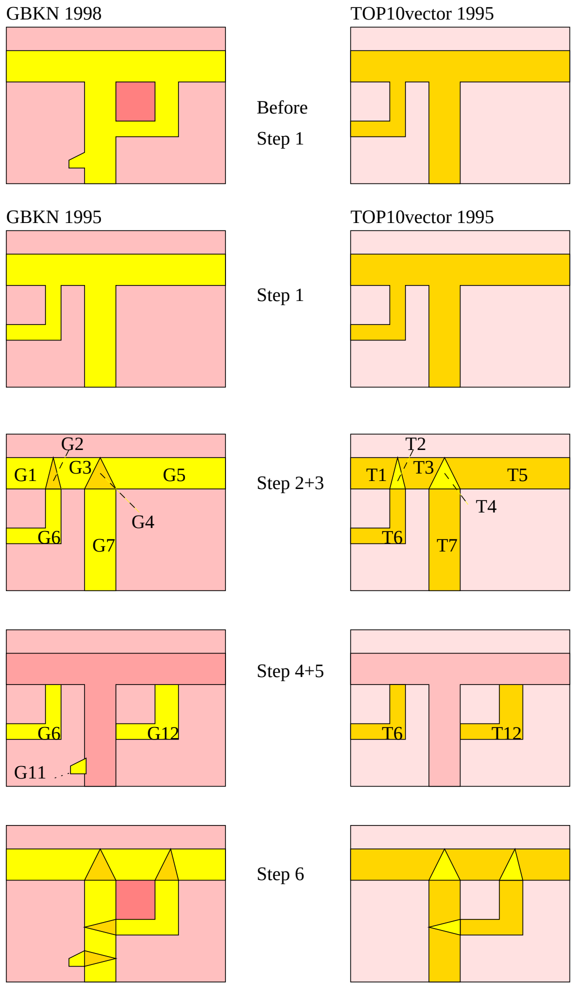

Fig. 11. Changes in the road network.

Before it is possible to propagate updates, line feature updates (as a result of a change in the road boundary edges) have to be transformed into area feature updates. This has to be done because surveyors in the field will always collect (point and) line measurements. Attention has to be paid to which lines have been used for creating road objects, which areas are effected by deleted, changed or new lines. We also have to find an answer to the questions when and how GBKN road updates should and could be propagated into the TOP10vector?

- *Step 4: Decide if an update is important enough to be propagated to TOP10-vector.* Determine whether the update affects the TOP10vector objects while respecting generalization and aggregation rules and relevance of an object for the TOP10vector. For example, in Fig.11 the new road element G12 is relevant, but the new road element G11 is not relevant because it is too small.
- *Step 5: Transform GBKN updates into TOP10vectors updates.* The object definitions are not the same. For example, in the TOP10vector a ditch belongs to the road, but this is not the case in the GBKN. Before changing an update it is helpful to analyze the relationship between the objects in both geographic databases. We analyze correspondences at geometric, attribute and semantic levels to understand if and how database integration could be possible. Making it possible to find a mechanism to change GBKN updates into TOP10vector updates. For example, in Fig.11 the GBKN road element G12 is first generalized and its classification is adjusted, and it then becomes TOP10vector road element T12.
- *Step 6: Propagate the updates into the TOP10vector.* An update has changed an original object. If that update has to be propagated to TOP10vector, it has to be propagated to the TOP10vector object, which corresponds with the GBKN object, which has been updated. It is not just possible to propagate the new object and remove the old version just like propagation buildings. If there is a new road, it is also necessary to connect the new road to the existing road segments, road junctions and to other nearby areas. That is, it has to be fitted into the TOP10vector topology structure. For example, in Fig.11 the GBKN road element G12 is connected to road elements G5 and G7, in the same manner T12 has to be connected to T5 and T7 in the TOP10vector.

Fig.11 shows an example area with changes in the road network. In this figure only the roads are shown. One old road disappears, one new road appears.

6 Conclusion and Future Work

In this paper the importance of well defined road network elements is argued for the purpose of geographic database integration in general and update propagation in specific. It was shown that the constrained Delaunay triangulation gives a good basis for the demarcation of road segments and road junctions. However, a few refinements are required such as removing small lump (or very small dead end street) and grouping neighbor or very close T-junctions to one n-way junction.

Future work consists of experimenting with the update propagation steps described in the previous section with real data sets. Further, investigation into other 'linear' feature types, such as railroads and water ways, is planned. The question is whether the same method for defining objects is valid for these feature types. Finally, we have to consider update propagation, and geographic database integration in general, not only on a 'feature by feature' basis, but in a more integrated way. Very often the different feature types are embedded in the same planar topology structure and do heavily influence each other.

References

1. L. Paul Chew. Constrained delaunay triangulations. *ACM*, ?:215–222, 1987.
2. T. Devogele, C. Parent, and S. Spaccapietra. On spatial database integration. *International Journal of Geographical Information Systems*, 12:335–352, 1998.
3. Peng Gao and Michael M. Minami. Raster-to-vector conversion: A trend line intersection approach to junction enhancement. In *Auto-Carto 13*, pages 297–303, April 1997.
4. L. Heres and P. Plomp J. den Hartog-Sprockel. NWB-kernmodel en extenties. Technical Report LAV2_007, RWS/AVV, 1997.
5. L. Heres et al. Geographic data files (GDF). Technical report, CEN/TC 287, Brussels, 1995.
6. R. Laurini. Spatial multi-database topological continuity and indexing: a step towards seamless gis data interoperability. *International Journal of Geographical Information Systems*, 12:373–402, 1998.
7. D. T. Lee and B. J. Schachter. Two algorithms for constructing a delaunay triangulation. *International Journal of Computer and Information Sciences*, 9(3):219–242, 1980.
8. Franco P. Preparata and Michael Ian Shamos. *Computational Geometry.* Springer-Verlag, New York, 1985.
9. Ravi. Geo-information terrain model. A Dutch standard for: Terms, definitions and general rules for the classification and coding of objects related to the earths surface (NEN-3610). Technical report, Ravi Netherlands Council for Geographic Information, Amersfoort, 1995.
10. M. I. Shamos and D. Hoey. Closest-point problems. In *Proceedings of the 16th Annual Symposium on the Foundation of Computer Science*, pages 151–162, October 1975.
11. S. Spaccapietra, C. Parent, and Y. Dupont. Model independent assertions for integration of heterogeneous schemas. *VLDB Journal*, 1:81–126, 1992.
12. Federico Thomas. Generating street center-lines from inaccurate vector city maps. *Cartography and Geographic Information Systems*, 25(4):221–230, 1998.
13. H. T. Uitermark. *Ontology-based map integration.* PhD thesis, Wageningen Agricultural University, 1999. In preparation.
14. H. T. Uitermark, P. J. M. van Oosterom, N. J. I. Mars, and M. Molenaar. Propagating updates: finding corresponding objects in a multi-source environment. In T. K. Poiker and N. Chrisman, editors, *Proceedings of the 8th International Symposium on Spatial Data Handling, Vancouver, Canada, 1998*, pages 580–591, 1998.
15. Harry T. Uitermark. The integration of geographic databases. Realising geodata interoperability through the hypermap metaphor and a mediator architecture. In *Second Joint European Conference & Exhibition on Geographical Information (JEC-GI'96), Barcelona, Spain*, pages 92–95, 1996.
16. Joost van Bemmelen and Wilko Quak. Master's thesis: Cross country movement planning. Technical Report FEL-93-S205, TNO Physics and Electronics Laboratory, August 1993.

17. Joost van Bemmelen, Wilko Quak, Marcel van Hekken, and Peter van Oosterom. Vector vs. raster-based algorithms for cross country movement planning. In *Auto-Carto 11*, pages 304–317, October 1993.
18. P. E. van der Vet and N. J. I. Mars. Bottom-up construction of ontologies. *IEEE Transactions on Knowledge and Data Engineering*, 10:513–526, 1998.
19. Peter van Oosterom. Maintaining consistent topology including historical data in a large spatial database. In *Auto-Carto 13*, pages 327–336, April 1997.
20. F. A. van Wijngaarden, J. D. van Putten, P. J. M. van Oosterom, and H. T. Uitermark. Map integration. Update propagation in a multi-source environment. In R. Laurini, editor, *5th ACM Workshop on Advances in Geographic Information Systems ACM-GIS'97, Las Vegas, Nevada, USA*, pages 71–76, 1997.
21. V. Walter and D. Fritsch. Linking objects of different spatial data sets by integration and aggregation. *GeoInformatica*, 5:1–23, 1997.
22. Annita N. Wilschut, Roelof van Zwol, Niek Brasa, and Jan Flokstra. Road collapse in magnum. In R. Laurini, editor, *6th ACM Workshop on Advances in Geographic Information Systems ACM-GIS'98, Washington DC, USA, 6-7 november 1998*, 1998.