Report

Generating topologically correct schematic maps

Author(s):
Avelar, Silvania; Müller, Matthias

Publication Date:
2000

Permanent Link:
https://doi.org/10.3929/ethz-a-006653901

Generating Topologically Correct Schematic Maps

Silvania Avelar, Matthias Müller
Department of Computer Science
Swiss Federal Institute of Technology
8092 Zurich, Switzerland
{avelar,muellerm}@inf.ethz.ch

Abstract

This paper studies the creation of schematic maps from traditional vector-based, cartographic information. An algorithm is proposed to modify positions of lines in the original input map with the goal of producing as output a schematic map that meets certain geometric and aesthetic criteria. Special emphasis is placed here on preserving topological structure of features during this transformation. The known, existing methods for preserving topology during map transformation generally involve computing several constrained Delaunay triangulations [Mol89, Gol94, JBW95, Rua95]. The algorithm proposed here computes a transformation which preserves topological relations among linear features using simple geometric operations and tests.

Keywords. Schematic maps, digital cartography, topology, computational geometry

1 Introduction

Schematic maps are linear cartograms designed to convey only essential features of network routes [Mon96]. They indicate important topological information on transportation or utility maps such as connectivity and stops, while preserving only a general sense of the original geometry [Dor96]. As long as essential topological information is preserved, the length and shape of routes in schematic maps need not be faithful to reality. This cartographic flexibility is possible because geometric accuracy is considered to be less important than linkages, adjacency, and relative position [Mon96, Bra98]. Because schematic maps do not have the mass of details usually presented in conventional topographic maps, they present the essential information more clearly. Figures 1 and 2 illustrate the topological nature of schematic maps.

Generating schematic maps *on demand* can be seen as a geometric constraint problem that should be solved algorithmically, in interactive time. In general, to create schematic maps automatically, first geometric line detail should be removed via smoothing or filtering. Then the new, straightened line network should be adjusted to improve legibility of important information. This line displacement follows a set of constraints defined by some common sense geometric and aesthetic criteria for the schematic map. Direction and distance are only approximately preserved for recognition purposes, while topological information of the road network is kept. The difficult aspect of generating schematic maps automatically is to reposition the network routes of the input map to locations which, besides fulfilling the set of constraints, also preserve the map topology [Elr91].

By preservation of map topology we mean the following three properties:

(i) no absence of line crossings that were present in the input map;

(ii) no line crossings that were not present in the input map;

(iii) cyclic order of outgoing connections around any node agrees with the ordering of connections in the input map.

In this paper we propose a technique to preserve the map topology during the transformation process. The technique is based on simple geometric operations and tests. To the best of our knowledge, there does not seem to be such a public, generally acknowledged solution for preserving topological relations while moving points.

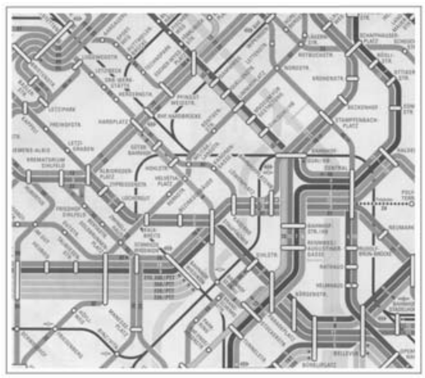

Figure 1: Example of a public transportation map of Zurich, Switzerland.

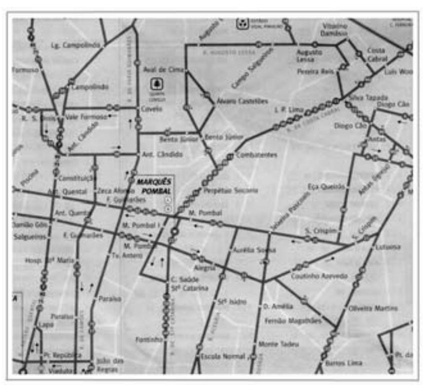

Figure 2: Example of a bus map of Porto, Portugal.

Some suggestions and theory concerning the automation of schematic maps can be found in the literature. Morrison [Mor96] studies public transport maps of various cities and proposes rules to govern the mapping method based on characteristics of transport services. Elroi describes his experience in generating schematic maps using GIS [Elr91]. Our contribution here is to produce schematic maps on demand while taking into account geometric and aesthetic constraints in such a way that map topology is preserved.

2 Characteristics of schematic maps

Routes on schematic maps are usually drawn as straight lines with cartographic microcrenulations removed. Direction of lines usually varies only via fixed, stylized angles — 45 and 90 degrees or 30, 60 and 90 degrees are common, though some schematic maps have only simplified lines with arbitrary, but few directions. Network lines with overlapping routes are separated by a minimum legibility distance. This distance can either be zero or a constant chosen for the map. Sometimes adjacent schematized lines have smooth, artistic, circular arcs around bends — these preserve the graphic proximity for the greatest length possible (see for example Figure 1, top left corner, where the lines curve between the two white line connectors).

We also observe that the straight network lines are generally not too long. A small number of breaks or changes in the line direction can be incorporated to add a sense of the original geometry, and to provide a better visualization of all routes together. In some schematic maps, all lines are presented in one color and the route of a particular network is traced by number only. In other maps, contrasting colors differentiate the various transportation lines.

Differing geometric and aesthetic criteria can be used to design a schematic map. Although it is easy to find differences in style (e.g. compare Figure 1 and 2), common goals are graphic simplicity, the retention of network information content and presentation legibility.

3 Generating schematic maps

The purposeful deformation of a geographic map is a common operation in cartography. Two types of map deformations can be considered: one for bringing two maps together, such as combining a satellite image and a geographic map via rubber-sheeting or minor geometric adjustment, and the other, to highlight quantities other than geographic distance and area, such as in cartograms [DEG98]. Our schematic maps can be thought of as the first type of deformation, but we use techniques for transforming lines similar to [HK98, Dor96] for constructing continuous area cartograms.

We assume the network lines to be schematized are provided as input to our algorithm to begin the automatic generation of schematic maps. We chose to create straight lines which can be horizontal, vertical, or at angles of 45 degrees. For readability and aesthetic considerations, we have set a user-defined maximum length for a straight line and a minimum distance between the lines. The cartographic data structure contains points and polylines.

As a preprocessing step to remove line crenulations, we perform point and line filtering (simplification) to reduce the number of visualization points that constitute the detailed map. The Douglas–Peucker algorithm [DP73] was used for this study; this algorithm is quick to implement and flexible, with a user definable threshold that controls the amount of simplification. After the simplification, long straight lines are cut according to the user-defined maximum length.

The next step is to find a new location for each point of each pre-simplified line according to the other constraints of the schematic map. We used a dynamic iterative algorithm driven by the map constraints [Mue99, HK98]. The network conformation is improved iteratively in order to satisfy all map constraints. The user decides when the schematic network reaches an acceptable appearance to stop the process. To change the actual conformation, the algorithm visits all points $p_1 ... p_n$ of the simplified map in turn. If the current point p violates one or more constraints, a better location $p\prime$ for p is computed. For each constraint $c^i \in \{c^1, \cdots, c^m\}$ that affects p, $p^{(i)}$ is the location nearest to p that satisfies c^i. The new location $p\prime$ is chosen to be the arithmetic mean of all $p^{(i)}$s. Instead of computing a simple mean of the displacement vectors, it would also be feasible to evaluate a weighted average of the vectors if certain roads are more important than others.

For the constraint of stylized angles, say c^j, the new location $p^{(j)}$ is chosen among the locations that fit the line segment in one of the allowed schematic directions. The nearest distance Δp determines the new location for p (see pq in Figure 3a). The constraints evaluation is carried out for all line segments in which p exists. The final location $p\prime$ for p will be obtained by the arithmetic mean of all required displacement vectors for all of its lines segments (in Figure 3b three constraints are considered).

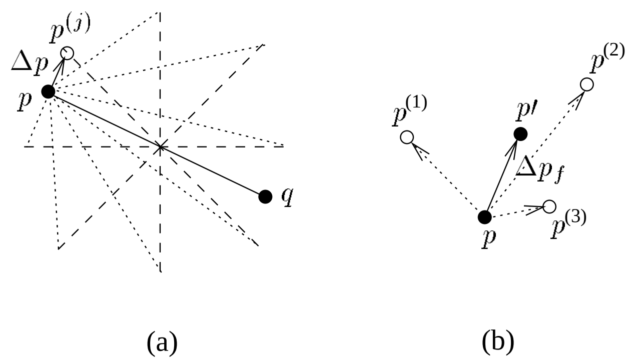

Figure 3: (a) Fitting a line segment $\overline{pq}$ to the nearest schematic direction and (b) finding the new location $p\prime$ for p.

We then check the minimum distance constraint. The final found location $p\prime$ should keep the specified minimum distance to the other linear features of the map. Adjustments in the arithmetic mean of the final displacement Δp_f vector are performed when necessary. The line segments may be stretched or shortened to reach the new locations, which we refer to as schematic locations. However, the topology of the map has to be preserved during this modification process.

4 Preserving the map topology

The previous section described how to compute a better location $p\prime$ (better according to the aesthetic and geometric constraints) for a given point p. Before displacing the point from its original position p to the new location $p\prime$, a test must be performed to detect situations that can lead to a change in the map topology, that is, to detect if one of the properties of Section 1 might be violated. We recall them:

(i) no absence of line crossings that were present in the input map;

(ii) no line crossings that were not present in the input map;

(iii) cyclic order of outgoing connections around any node agrees with the ordering of connections in the input map.

In this section we describe such a test and also how to adjust, if necessary, the new location $p\prime$ to avoid change in topology. We explain the test first and prove correctness later. The test will involve all line segments which have p as an endpoint. It will be sufficient to consider a single such segment, say pq, to describe the test. Consider the triangle $T = T(pp\prime q)$ to perform the test. We have to find out whether there is any line segment of the map crossed by the boundary edge $\overline{pp\prime}$ of the triangle T. We also have to check whether the triangle T contains inside it any point [Ave97].

If the topology would change, the move of p must be smaller than $p\prime$ to avoid the intersection. It is important whether a point moves at all. If a point moves, this will affect other points in the polyline when the algorithm tries to find a better location for them.

We distinguish between the following cases:

(1) *There is no point inside the triangle T and no line segment crossing edge $\overline{pp\prime}$.*
In this case map topology will not be changed. The move of p to $p\prime$ is allowed (see Figure 4).

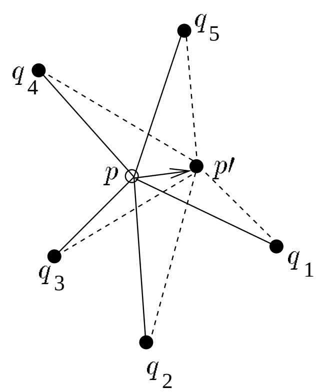

Figure 4: Point p can be moved to $p\prime$.

(2) *There is at least one line segment intersecting edge $\overline{pp\prime}$.*
In this case map topology will change (see Figures 5a and 5b). A new location for $p\prime$ has to be obtained. The new $p\prime$ is taken to be the nearest intersection point, say u, to p plus or minus the pre-defined minimum distance d, measured along $\overline{pp\prime}$ (see Figure 6a). The paramater d is an aesthetic constraint for the map.

(3) *There is at least one point v inside the triangle T.*
In this case map topology might change (see Figures 5c and 5d). To calculate the new location for $p\prime$, define a straight line l through v and q and calculate the intersection point u of l and the edge $\overline{pp\prime}$ of T. Take $p\prime$ to be the nearest intersection point u to p plus or minus a pre-defined minimum distance d (see Figure 6b).

The cases (2) and (3), as well more cases of each one, can occur together. In such situations, the new location for $p\prime$ will also be calculated taking the nearest intersection point to p.

Note that, after the above checks have been performed, the newly adjusted location $p\prime$ will be such that case (1) holds for all line segments $\overline{pq}$. In other words, the situation in Figure 4 holds.

Lemma 4.1 *If case (1) holds for all line segments $\overline{pq}$, then moving the point with location p to the new location $p\prime$ will preserve properties (i), (ii) and (iii).*

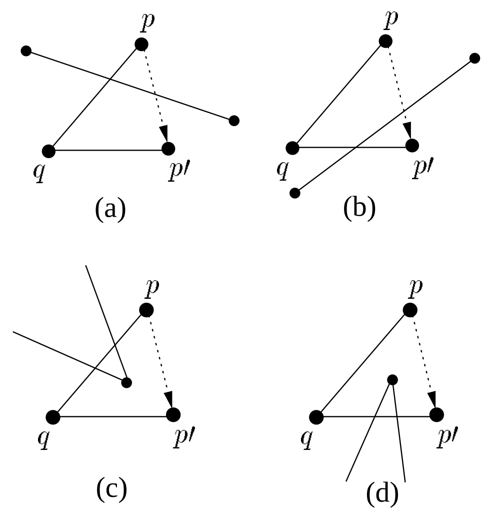

Figure 5: Illustration of possible cases for which topology changes.

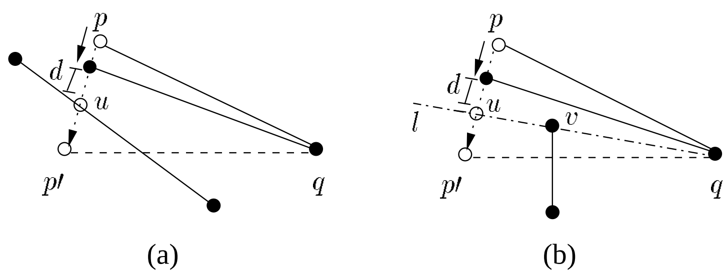

Figure 6: Topology checking.

Proof. We consider each property in turn.

(p1) Since only point p is moved, we need only consider crossings with line segments of the form $\overline{pq}$. Assume there is a line segment $\overline{ab}$ which crosses such a segment $\overline{pq}$. Because, by assumption, neither point a nor b lies inside triangle $T(pp'q)$, segment $\overline{ab}$ must cross also $\overline{p'q}$. See Figure 7.

(p2) Note that triangle $T(pp'q)$ coincides with triangle $T(p'pq)$ and line segment $\overline{pp'}$ coincides with segment $\overline{p'p}$. The result now follows from property (i) and from symmetry by considering the transformation of the point with new location p' back to the old location p.

(p3) For nodes of type p see Figure 4. Now consider nodes of type q. Suppose that $\{\overline{qp}, \overline{qp_1}, \overline{qp_2}, \ldots, \overline{qp_k}\}$ is the set, in cyclic order, of all outgoing connections from node q. See Figure 8. Provided that angle γ is less than θ, the cyclic order will be preserved; but this must hold since otherwise condition (1) would not be satisfied. This ends the proof.

We remark that although condition (1) is sufficient to preserve map topology, it is not necessary. Consider Figure 9. Since there are no line crossings in pq, neither in $p'q$, so the topology is preserved. We consider this a positive feature of the triangle test, since we want, whenever possible, to avoid changes in the "sidedness" relations among geographic elements, as represented

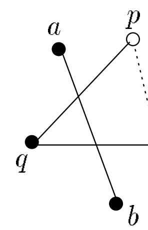

Figure 7: Segments cross preserving topology.

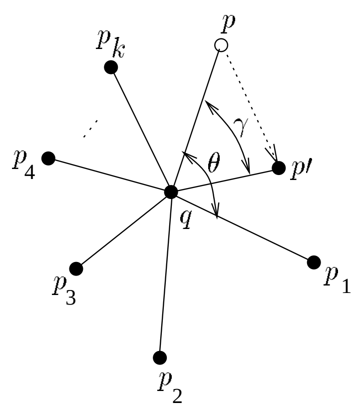

Figure 8: Cyclic order around node is preserved.

by pq and v. The test of point inside triangle can also detect some of such changes among geographic features, as shown in Figure 9 (p is at one side of the lake and $p\prime$ at the other side). Extensions of the algorithm could check the neighbourhood along $pp\prime$ out of the triangle.

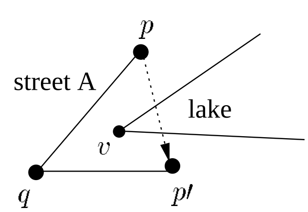

Figure 9: In this case topology is preserved, but not "sidedness" relations.

5 Implementation

We next present the algorithm as developed so far:

Procedure SchematicMap (map, maxSegment, tolerance);
Input: N points of the conventional map, the maximum line segment, the simplification tolerance

```
Output: n points for the schematic map
begin
  simplify lines with Douglas–Peucker algorithm using given tolerance;
  cut long line segments according to maxSegment;
  repeat
    for all points p1..pn of the map do
      test constraints and calculate a new schematic location p' for pi;
      if there is any segment collision with pi p' then
        calculate new location for p';
      for each point q that makes a line segment with pi do
        if there is any point inside triangle pi p' q then
          calculate new location for p';
      end for;
      change location of pi to p';
    end for;
  until reached desired appearance;
end;
```

As stated, the search starts with the original configuration. The structure is iteratively improved by performing optimization steps based on different types of moves in the inner loop.

We do not need to test all line segments and points in the map to look for geometric intersections of segments and points inside triangle. Only adjacent regions of the point being analysed are considered. The nodes and segments are stored in a uniform grid that divides the plane into non-overlapping regions. Line segments are distributed among the regions where their center point lies. The grid spacing is chosen in such a way that potentially intersecting segments can only be found in either the same or two neighbour cells. Spatial partioning algorithms like this [ASB+94] are sufficient because we do not move points too far and the line segments are at maximum the size of a cell.

6 Results and Conclusion

We have presented in this paper a new method for preserving topological relations among linear features while generating schematic maps. The method could also be applied when schematic maps are generated using grid fitting algorithms [Elr91].

The main ideas of the schematic map generator were implemented with encouraging first results. Figure 10 shows an example of the results obtained with our prototype using a database containing all streets of Zurich. For this study, map labels and cartographic features were not included.

The Porto bus map in Figure 2 has less aesthetic treatment and relatively less geometric simplification than the public transportation map of Zurich in Figure 1. For this reason, we hoped to achieve, in early stages, a map of comparable quality to Figure 2. Preliminary inspection of our results indicates our schematic maps have a similar design appearance and feel to Figure 2. Our primary focus here was to preserve map topology, but there are also other features that are desirable to be preserved in the resulting map, such as side relations among geographic features and the graphical continuity of main streets.

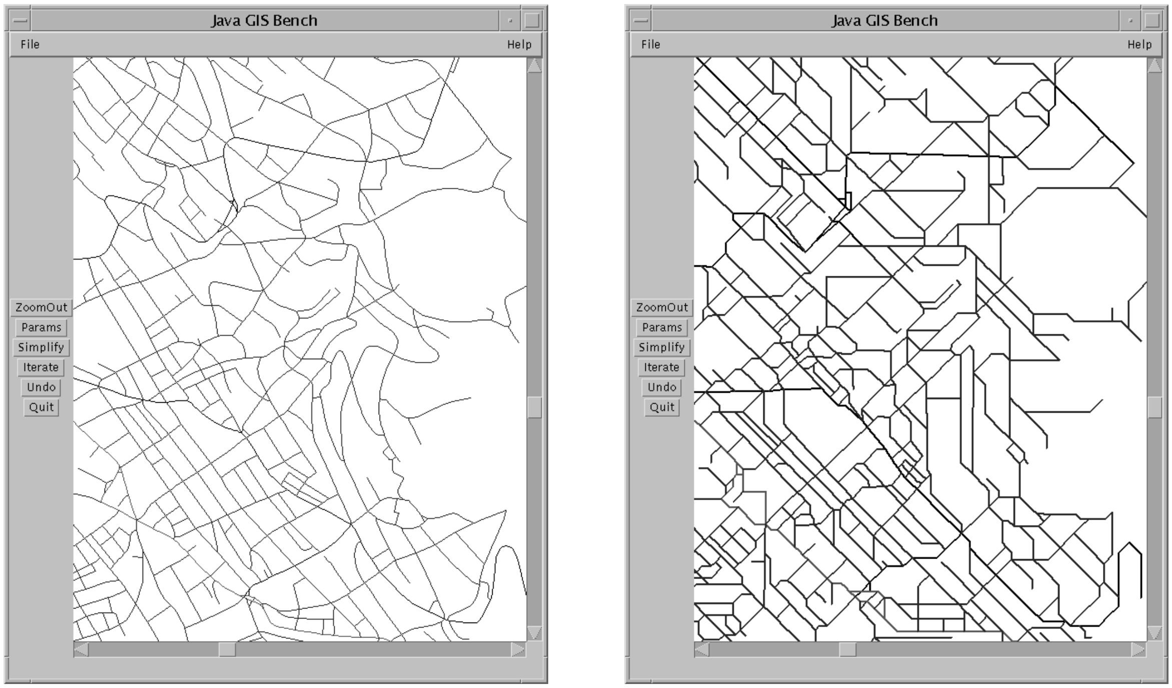

Figure 10: Example of streets in a standard map and a derived schematic map.

For the line simplification, we used the classical Douglas–Peucker algorithm. However, there could be some topological and self-intersection errors introduced by using this algorithm. Variants of the Douglas–Peucker algorithm which take topology into account [Saa99] or other line simplication methods [dBvKS95] could be tried instead.

Further research on the presented topic is still necessary. We hope to continue this work by studying and defining other formulation of aesthetic and geometric constraints to generate automatic schematic maps and determining a measure of how good a resulting map is. We plan also to extend the prototype to add cartographic features to the visualization.

Acknowledgments

The authors of this paper are grateful to Ambros Marzetta and Matthieu Barrault for helpful comments. Special thanks go to Arne Storjohann and Frank Brazile for valuable suggestions on proof-reading this paper.

References

[ASB+94] D. Andrews, J. Snoeyink, J. Boritz, T. Chan, G. Denham, J. Harrison, and C. Zhu. Further Comparison of Algorithms for Geometric Intersection Problems. In *6th Int. Symposium on Spatial Data Handling*, pages 1–12, September 1994.

[Ave97] Silvania Avelar. The problem of contour in the generation of digital topographic maps. In *Auto-Carto 13*, pages 397–403, Seattle, 1997. ACSM/ASPRS.

[Bra98] Francisco Brazile. A generalization machine design that incorporates quality assessment. In *8th Int. Symp. on Spatial Data Handling*, pages 349–360, Vancouver, 1998.

[dBvKS95] Mark de Berg, Marc van Kreveld, and Stefan Schirra. A new approach to subdivision simplification. In *Auto-Carto 12*, pages 79–88, Charlotte, 1995. ACSM/ASPRS.

[DEG98] Tamal K. Dey, Herbert Edelsbrunner, and Sumanta Guha. Computational topology. In B. Chazelle, J.E. Goodman, and R. Pollack, editors, *Advances in Discrete and Computational Geometry*, Providence, 1998. AMS.

[Dor96] Daniel Dorling. Area cartograms: their use and creation. *Concepts and Techniques in Modern Geography (CATMOG)*, 59, 1996.

[DP73] David H. Douglas and Tom K. Peucker. Algorithms for the reduction of the number of points required to represent a line or its caricature. *The Canadian Cartographer*, 10(2):112–122, 1973.

[Elr91] Daniel Elroi. Schematic view of networks - why not have it all? In *Gis for Transportation Symposium*, Orlando, 1991.

[Gol94] Christopher Gold. Three approaches to automated topology and how computational geometry helps. In T.C. Waugh and R. G. Healey, editors, *Advances in GIS Research I*, pages 145–158, London, 1994. Taylor and Francis.

[HK98] Donald H. House and Christopher J. Kocmoud. Continuous cartogram construction. In *9th IEEE Visualization Conference*, pages 197–204, North Carolina, October 1998.

[JBW95] Christopher B. Jones, Geraint L. Bundy, and J. Mark Ware. Map generalization with a triangulated data structure. *Cartography and Geographic Information Science*, 22(4):317–331, 1995.

[Mol89] Martien Molenaar. Single valued vector maps - a concept in geographic information systems. *Geo–Informationssysteme*, 2(1):18–26, 1989.

[Mon96] Mark Monmonier. *How to lie with Maps.* The University of Chicago Press, 1996.

[Mor96] Alastair Morrison. Public transport maps in western european cities. *The Cartographic Journal*, 33(2):93–110, September 1996.

[Mue99] Matthias Mueller. *The Structure of Dense Polymer Systems, Geometry, Algorithm and Software.* PhD thesis, Department of Computer Science, ETH–Zurich, 1999.

[Rua95] Anne Ruas. Multiple paradigms for automating map generalization: Geometry, topology, hierarchical partitioning and local triangulation. In *Auto-Carto 12*, pages 69–78, Charlotte, 1995. ACSM/ASPRS.

[Saa99] Alan Saalfeld. Topologically consistent line simplification with the douglas-peucker algorithm. *Cartography and Geographic Information Science*, 26(1):7–18, 1999.