Schematic Road Network Map Progressive Generalization Based on Multiple Constraints

DONG Weihua  GUO Qingsheng  LIU Jiping

Abstract  Multiple constraints for schematic road network map cartographic design are analyzed and summarized. Based on this, a set of quantitative criteria are set up and a new road network generalization method including progressive selection and displacement is proposed. Furthermore, topological checking methods for road networks are researched. Based on these constraints, the points in a road network are classified, and a satisfactory and effective schematic map is designed in a concrete experiment while maintaining topological consistency of the road network between the original and the schematic map.

Keywords  schematic map; spatial cognition; progressive cartographic generalization; graphic simplification

Introduction

As an effective diagrammatic representation based on linear abstractions of networks, schematic maps such as road networks and river networks have become quite a standard way to convey information. In contrast to traditional cartography, schematic network mapping needs new cartographical generalization methods. In graphics and language, schematization is an important method to emphasize certain aspects and to deemphasize others. In cartography, graphic schematization is one aspect of map generalization. The main advantage of schematic maps is that they provide a quick overview of the layout of the network, without showing unnecessary information like the precise shapes of the connections. Schematic drawings of route directions are one of the most common forms of graphic communication. However, this topic has received little attention in traditional mapping and in automated mapping and GIS.

Automated productions of schematic maps have been discussed in several earlier papers[1,2,3], but the topic has reappeared lately in more papers[4,5]. All of these papers make use of some local operator, such that after applying it several times, the map converges to a schematic one. Different criteria based on different measures are used to decide in which order, and to which parts of the map the operator should be applied. Iterative approaches like these have prohibitive costs in terms of time in interactive situations, and in previous papers, no theoretical time bounds or actual computation times were given. A related topic that is relevant to schematization is the rendering of particular routes under queries[6,7]. Although the topic has its own characteristics, both procedures adopt the Douglas-Peucker[8] simplification algorithm for graphic simplification. In this paper, a new road network
generalization method including progressive selection and displacement is proposed, and this effective method is proved by a concrete experiment.

1 Constraints of a schematic road network map

In designing a schematic map, there are many aspects of the mapping situation which may need special attention. Besides the characteristics of the road network, cartographic aspects, such as the amount of simplification of lines, vertices displacement, symbolization and the use of colors are also important. All schematic maps should have graphic simplicity while retaining network information and presentation legibility. Based on the research of J. Mark Ware[9], we propose cartographical criteria of schematic maps as follows: ① The topological-original network and the derived schematic map should be topologically consistent. ② Orientation- if possible, network edges should lie in the horizontal, vertical or diagonal direction. ③ Length- if possible, all network edges should have lengths greater than some minimum length (in order to reduce congestion). ④ Angle- if possible, the angle between a pair of connected edges should be greater than some minimum angle. ⑤ Rotation- an edge’s orientation should remain as close to its starting orientation as possible. ⑥ Clearance- if possible, the distance between disjoint features should be greater than some minimum distance. ⑦ Semantics- road-crossing vertices and turning points should have more important spatial cognitive characteristics than other vertices and should be rendered more scores during the course of simplification and displacement.

Based on these constraints, quantitative constraints shall be set up. In these constraints, a topological constraint and a semantic constraint are directly allowed during the course of generalization. We assume that \( r \) is a predefined minimum pixel length. Other constraints besides topological semantic ones are as follows.

1) Orientation constraint. In a schematic map, based on the user’s defining standard directions including the horizontal, vertical or diagonal direction, all road network edges only choose one of these directions which is most close to their original directions. Supposing the standard orientation \( RR = \{ RR_j, j \in \{1 \sim 8\} \} \) and original orientation is \( OR \), the orientation constraint is quantified by Eq.(1).

\[
\min |RR_j - OR_i|
\]  

(1)

2) Distance constraint. Overlapping among road network lines which often take place in transport networks will not happen in the road network. In this case, we should separate these overlapping lines from the separating distance which, according to the environment, is object resolution.

3) Length constraint. The shortest road network edge must have a greater length than \( r \) and maintain the relative ordering of the roads by length. The length constraint is quantified by Eq.(2) and (3).

\[
L(S) \geq r
\]

(2)

\[
\left| \frac{L(S_i)}{L(S_i')} - \frac{L(S_j)}{L(S_j')} \right| \leq B
\]

(3)

Where \( L(S_i) \) and \( L(S_j) \) is the length of the original road network edge; \( L(S_i') \) and \( L(S_j') \) is the final length of these edges; \( \beta \) is the threshold assumed by the user.

4) Angular constraint. When meeting at one intersection and the angle between each other is small, the direction of overlapped road segments shall be entitled to more detailed standard orientation in \( \{ RR_{i,j} - RR \} \) or \( \{ RR_{i,j} - RR_{i,j} \} \). The angular constraint is quantified by Eq.(4):

\[
OR_i = \begin{cases} 
RR + \frac{RR_{i,j} - RR}{M}, & \text{if } RR_{i,j} \in OJ \\
RR - \frac{RR_{i,j} - RR_{i,j}}{M}, & \text{if } RR_{i,j} \notin OJ 
\end{cases}
\]

(4)

where \( M \) is the amount of the adjusted orientation of the road edges; \( OJ \) is the collection of reoriented road edges; \( RR_{i,j} \) is the adjusted orientation of the road edges.

2 Generalization methods of the schematic road network map

Our techniques are based on the theory that it is more important for users to capture the basic structure of the network than to show accurately physical locations on the map[10]. Furthermore, cognitive psychol-
ogy research shows that an effective route map must clearly communicate all the turning points on the route [11], and that precisely depicting the exact length, angle, and shape of each road is much less important [12].

In order to administer, search these data and make it convenient for shape simplification and displacement, we designed the transport network data structure in Fig. 1. Transport networks are divided into segments by the start points, end points and intersection, which form the original road network $O$ and the simplified road network $G$. We can use the $O$ data to compute the original road segment length $O_{L}$, which is between the remaining vertices, and we compute the conjunction degree of each intersection. Each vertex attribute is the collection $(PT, PD, PL, MS)$, where $PT$ is the vertex type (if the vertex is unremovable, we set $PT=1$, while if the vertex is removable, we set $PT=0$); $PD$ is the symbol of the vertex (if the vertex has been removed, we set $PD=1$, while if the vertex is not removed, we set $PD=0$); $PL$ is the symbol of the vertex which should be removed but is not removed (if the vertex should be removed but is not removed, we set $PL=1$, while we set $PL=0$, if the vertex is an independent point); $MS$ is the symbol of the vertex’s successful displacement (if the vertex or the segment displace successfully, we set $MS=1$, while if the vertex displaces unsuccessfully, we set $MS=0$).

![Fig.1 Road network data structure](image)

### 2.1 Shape simplification

The shape simplification stage reduces the number of segments in each road while leaving the overall shape of the route intact. Shape simplification not only yields a cleaner looking map but also increases the speed and memory efficiency of the subsequent layout stages of the system.

#### 2.1.1 Classification of road network vertices

In order to keep topological consistency between the original network map and the schematic map during the course of shape simplification, we set rules for topology validity checking and divide these vertices into two types—unremovable vertices and removable vertices.

1. **Unremovable vertices**—a vertex is shared by more than two objects or is the only vertex shared by more than one object. These vertices include an intersection vertex shared by more than two line objects and the starting and ending vertex of a line of objects.

2. **Removable vertices**—vertices that can be deleted that usually occur as parts of polygonal curves and do not represent individual geographic entities. In the network, they are vertices of segments and have smaller contributions of kink to the shape of the polygonal curve.

#### 2.1.2 Simplification

Because the schematic map has the characteristic of shape simplification, both M. Agrawala, and S. Avelar adopted the Douglas-Peucker algorithm to simplify the road network, which makes it easy to generate self-intersection and intersection losing, which will change the road network topology. In order to maintain the network topology during the course of shape simplification, we use the algorithm of progressive simplification, while checking if there exist one or more points lying within the triangle $\Delta(v_{i-1}v_{i+1}v_{i})$ spanned by the two segments of the vertex $v$ in question and checking whether or not there exist line segments crossing the edge $v_{i-1}v_{i}$ or $v_{i}v_{i+1}$ of $\Delta(v_{i-1}v_{i+1}v_{i})$.

Based on the semantic constraint, vertices including the start vertex, the end vertex, and the intersection vertex will not be removed. The interior vertex in the collection $P\{p_{1}, p_{2}, \cdots, p_{n}, p_{n+1}\}$ of the road network will be removed according to the graphic relevance measure of the vertex. The contribution of
kinks to the shape of the polygonal curve is calculated by using formulation Eq.(5). The point which has the smallest contribution to the shape of road segments will be removed with each simplification.

\[
K(S_i, S_j) = \frac{\beta(S_i, S_j) \times L(S_i) \times L(S_j)}{L(S_i) + L(S_j)}
\]

where \( K(S_i, S_j) \) is the value of the relevance measure; \( L(S_i) \) and \( L(S_j) \) are the lengths of two consecutive line segments; \( \beta(S_i, S_j) \) is the turn angle at the common vertex of segments. The higher the value of \( K(S_i, S_j) \) is, the larger the contribution of the kink to the shape of the polygonal curve is.

The algorithm of shape simplification is as follows.

\textbf{Input:} Original maps with \( M \) road segments and shape simplification threshold \( \epsilon = 2A \)

\textbf{Output:} Simplified network map with \( M \) road segments;

\textbf{begin}

\textbf{Set} \( i = 1, j = 1 \);

\textbf{for each segment} \( s(i) \in M \) \textbf{do}

\textbf{for each vertex} \( v_j \) of \( s(i) \) do

\textbf{if} \( PD(v_j) \neq 1 \)

Compute the geometric weight \( w_j(v_j) \);
Generate a priority queue \( Q = \{ w_j(v_j) \} \);
Set \( w_j(v_j) = \text{min}(Q) \);
Compute and set \( B = \Delta(v_j, v_j, v_{j+1}) \);
Compute a list \( I \) of intersection points, which are not vertices of a segment;
\( A(v) = \{ v \neq v_j \text{ and } v \text{ is the other vertex of the original road network} \} \);
\textbf{if} \( B \subseteq \epsilon \) and \( v_{j-1}, v_{j+1} \cap I = \phi \) and \( v_j, v_{j+1} \cap I = \phi \) and \( A(v) \cap \Delta(v_j, v_j, v_{j+1}) = \phi \)

\textbf{then}

Remove the vertex \( v_j \);
Set \( PD(v_j) = 1 \) and \( PL(v_j) = 0 \);
\textbf{else} \( B \subseteq \epsilon \) and \( v_{j-1}, v_{j+1} \cap I \neq \phi \) and \( v_j, v_{j+1} \cap I \neq \phi \) and \( A(v) \cap \Delta(v_j, v_j, v_{j+1}) \neq \phi \)

\textbf{then}

Set \( PL(v_j) = 1 \);
Select a vertex with lesser value from;

\textbf{end};

\textbf{end};

\textbf{2.2 Topology checking}

During the course of shape simplification and point displacement, the topology of the road network should be checked and maintained. Before moving a point from its original position \( q \) to the new schematic location \( q' \), we perform a test to detect situations that can lead to a change in the map topology. For this we first create a triangle. The vertices of the triangle are \( q \), \( q' \), and the other endpoint of the line segment being analyzed, \( p \). We have to find out if there is any line segment of the map crossed by the boundary edge \( qq' \) of the triangle \( \Delta(ppo) \). We also have to check if the triangle \( \Delta(ppo) \) contains any point inside it. If the topology would change, point \( q \) displacement must be recomputed. We distinguish the following three cases to check the topology.

\begin{enumerate}
  \item If there is no point inside the triangle \( \Delta(ppo) \) and no line segment crossing the edge \( qq' \) of \( \Delta(ppo) \) the topology will be preserved, so the move of \( q \) to \( q' \) is allowed.
  \item If there is at least one line segment intersecting edge \( qq' \) of \( \Delta(ppo) \) (see Fig.2(a)), map topology will change, then a new location for \( q \) has to be obtained. Because of displacement, \( q \) generates the intersection point, noted as \( u \). Therefore, the new \( q \) will move a distance \( d \) to \( u' \) along line \( qq' \), where \( d \) is the minimum resolution distance of the map (see Fig.2(b)).
  \item If there is at least one point \( v \) inside the triangle (see Fig.3(a)), map topology will also change. To calculate the new location for \( q' \), we define a straight line \( l \) through \( v \) and \( q \) and calculate the intersection point \( u \) of \( l \) and the edge \( qq' \) of \( \Delta(ppo) \). In this case, the new \( q' \) will move a distance \( d \) to \( u' \) along line \( qq' \), where \( d \) is the minimum resolution distance (see Fig.3(b)).
\end{enumerate}
the natural orientation of the road and make the displacement distance the shortest. Using the method of equal length, we tune the road segment orientation to the eight standard directions, which is the collection \( RR = \{0^\circ, 45^\circ, 90^\circ, 135^\circ, 180^\circ, 225^\circ, 270^\circ, 315^\circ\} \) (see in Fig.4), where the positive \( X \) axes represent \( 0^\circ \) and \( 360^\circ \), and the orientation of the road segments is the rotation angular calculated counter-clockwise based on the \( X \) axes.

![Fig.4 Displacement of points of the road network](image)

In order to satisfy the orientation constraint, the orientation and length of every adjusted road segment shall be calculated by using Eqs.(6) and (7).

\[
\begin{align*}
    OR_i &= RR_i, \quad \text{if} \; \min OR_i \neq RR_i \\
    OR_i &= 0^\circ, \quad \text{if} \; RR_i = 360^\circ
\end{align*}
\]

and

\[
\begin{align*}
    OL_i &= \frac{O_i \times r}{\min(OL)}, \quad \text{if} \; \min(OL) < r \\
    OL_i &= OL_i, \quad \text{if} \; \min(OL) \geq r
\end{align*}
\]

Where \( OL \) is the collection of all road segment lengths.

We select intersection points of the road segments, find the road segment collection adjacent to these points and get the adjusted orientation collection \( OJ \).

Firstly, determine whether there are overlapping road segments which have been adjusted. If so, we can say that there is an orientation conflict at this intersection point, and the method for solving the problem is by adopting an Angular constraint. The orientations of overlapping road segments are reoriented during the collection of \( [RR_{j,i}, RR_{j,i+1}] \) or \( [RR_{j}, RR_{j+1}] \) and \( OR_i \)—the length of these segments are recalculated. Next, we grow all roads that are shorter than a pre-defined minimum pixel length, \( \min(OL) \) to be \( r \) pixels long by using Eq.(7).

The algorithm of progressive displacement is as follows.

**Input:** Original maps with \( M \) road segments and shape simplification threshold \( \varepsilon = 2A \)

**Output:** Simplified network map with \( M \) road segments;

**begin**

\[
\begin{align*}
    &\text{Set } i=1, j=1; \\
    &\text{for each segment } s(i) \in M \text{ do} \\
    &\quad \text{for each vertex } v_j \text{ of } s(i) \text{ do} \\
    &\quad\quad \text{if then } PD(v_j) \neq 1 \\
    &\quad\quad\quad \text{Compute the geometric weight } w_j(v_j); \\
    &\quad\quad\quad \text{Generate a priority queue } Q = \{w_j(v_j)\}; \\
    &\quad\quad\quad \text{Set } w_j(v_j) = \min(Q); \\
    &\quad\quad\quad \text{Compute and set } B = \Delta(v_j,v_{j+1}) \\
    &\quad\quad\quad \text{Compute a list } I \text{ of intersection points} \\
    &\quad\quad\quad \text{which are not vertices of a segment;} \\
    &\quad\quad\quad \text{if } B \leq \varepsilon \text{ and } v_{j-1}v_j \cap I = \emptyset \text{ and } v_{j+1}v_j \cap I = \emptyset \\
    &\quad\quad\quad\quad \text{then Remove the vertex } v_j; \\
    &\quad\quad\quad\quad \text{Set } PD(v_j) = 1 \text{ and } PL(v_j) = 0; \\
    &\quad\quad\quad \text{else if } B \leq \varepsilon \text{ and } v_{j-1}v_j \cap I = \emptyset \text{ and } v_{j+1}v_j \cap I = \emptyset \\
    &\quad\quad\quad\quad \text{then } \text{Select a vertex with lesser value from;} \\
    &\quad\quad\quad\quad \text{Set } i++; \\
    &\quad\quad\quad j++; \\
    &\quad\quad \text{end.} \\
    &\quad \text{end.} \\
    &\text{end.} \\
\end{align*}
\]

**3 Results and conclusions**

In order to prove the algorithm’s efficiency, we implement an experiment with Java programming language based on the vector database of 1:4 000 000 road network including seven roads, 111 road segments and 508 vertices (59 intersection vertices, 18 start and end vertices, 431 interior vertices). Results are shown in Fig.5.

Experimental results show that 156 interior vertices of the original road network (Fig.5(a)) are eliminated successfully while keeping the topology. Then, driven by a set of constraints, the vertices of the simplified road network (Fig.5(b)) are displaced and segments are oriented to the horizontal, vertical or diagonal direction. Finally, a desired schematic road network (Fig.5(c)) is generated.

Topology errors are not found in this experiment mainly because of the characteristics of the spatial data. In order to promote the display speed of the
schematic road network, and the computation for resolving the topological conflict is avoided as much as possible. The algorithm is proved feasible and effective by this experiment. However, for the schematic map design of a complicated transport network, more map design aspects should be taken into account, such as overlap of a number of road segments, and false and true intersection points in the road network, such as having an intersection but not having any stations.

![Original road network](image1)

![Graphic simplification result](image2)

![Displacement result](image3)

**Fig.5 Experiment results**

**References**

[1] Elroi D (1988) Designing a network line-map schematization software-enhancement package [C]. 8th Ann ESRI User Conference, San Diego, CA

[2] Elroi D (1988) GIS and schematic maps: a new symbiotic relationship [C]. Proceedings of GIS/LIS’88, San Antonio

[3] Elroi D (1991) Schematic views of networks: why not have it all [C]. Proceedings of the 1991 GIS for Transportation Symposium, Orlando, Florida

[4] Avelar S, Müller M (2000) Generating topologically correct schematic maps [C]. Proceedings of 9th International Symposium on Spatial Data Handling, Beijing

[5] Barbowsky T, Latecki L J, Richter K -F (2000) Schematizing maps: simplification of geographic shape by discrete curve evolution [C]. Proceedings of Spatial Cognition II, Berlin

[6] Avelar S, Huber R (2001) Modeling a public transport network for generation of schematic maps and location queries [C]. Proceedings of 20th International Cartographic Conference, Beijing

[7] Agrawala M, Stolte C (2001) Rendering effective route maps: improving usability through generalization [C]. SIGGRAPH, Los Angeles, California

[8] Douglas D H, Peucker T K (1973) Algorithms for the reduction of the number of points required to represent a digitized line or its caricature [J]. The Canadian Cartographer, 10(2):112-122

[9] Ware J M, Anand S, Taylor G E, et al. (2006) Automated production of schematic maps for mobile applications [J]. Transactions in GIS, 10(1): 25-42

[10] Avelar S (2002) Schematic maps on demand: design, modeling and visualization [D]. Zurich: Swiss Federal Institute of Technology

[11] Denis M (1997) The description of routes: a cognitive approach to the production of spatial discourse [J]. Cahiers de Psychologie Cognitive, 16(4):409-458

[12] Tversky B, Lee P (1999) Pictorial and verbal tools for conveying routes [C]. Proceedings of COSIT, Stade, Germany

[13] Avelar S (1997) The problem of contour in the generation of digital topographic maps[C]. Proceedings of AutoCarto13, Seattle

[14] Berg M de, Kreveld M van, Overmars M, et al. (1997) Computational geometry: algorithms and applications [M]. Berlin: Springer-Verlag