Модел. и анализ информ. систем. Т. **0**, № **0** (2000) 0–4

УДК 517.51+514.17

Fractal and Computational Geometry for Generalizing Cartographic Objects

, Oleg Musin, Alexey Ukhalov, Olga Yakimova, Vladislav Alexeev, Victoriya Bogaevskaya, Andrey Gorohov, Margarita

P.G. Demidov Yaroslavl State University

e-mail: polya@uniyar.ac.ru

получена 2012

Ключевые слова: картографическая генерализация, сегментация, кривизна, фрактальная размерность

Предлагается новый алгоритм генерализации линейных картографических объектов. Основным новшеством алгоритма является автоматическая сегментация – разбиение ломаной на участки с одинаковыми свойствами. Сегментация позволяет подобрать параметры сглаживания индивидуально для каждого участка, за счет чего существенно повышается качество результата.

We consider the problem of *generalization* of linear cartographic objects as defined in [1]. Given a possibly closed curve specified by a linear or cyclic sequence of points(its vertices) in the plane, the goal is to draw the curve to scale, maintaining the important features and suppressing the unimportant ones. The curve is assumed to represent a geographic object, such as the border of a state, the contour of a continent, or the flow of a river.

It is obvious that on relatively even sections of the curve, we can delete many points without loss of important information, while on relatively curved sections, we need to retain a larger number of points. The difficulty lies in formalizing the intuitive notion of importance that controls the simplification of the curve.

Our algorithm consists of the following five steps:

Step 1: **Uniform parametrization.** Sample points uniformly along the curve to transform the representation to edges with uniform length.

Step 2: **Segmentation.** Decompose the curve into segments with roughly uniform curvature properties.

Step 3: **Assessment of fractal dimension.** Calculate the fractal dimension of each segment.

Step 4: **Simplification.** Coarsen the representation of each segment using the Douglas-Peucker algorithm [3] controlled by the locally accessed fractal dimension.

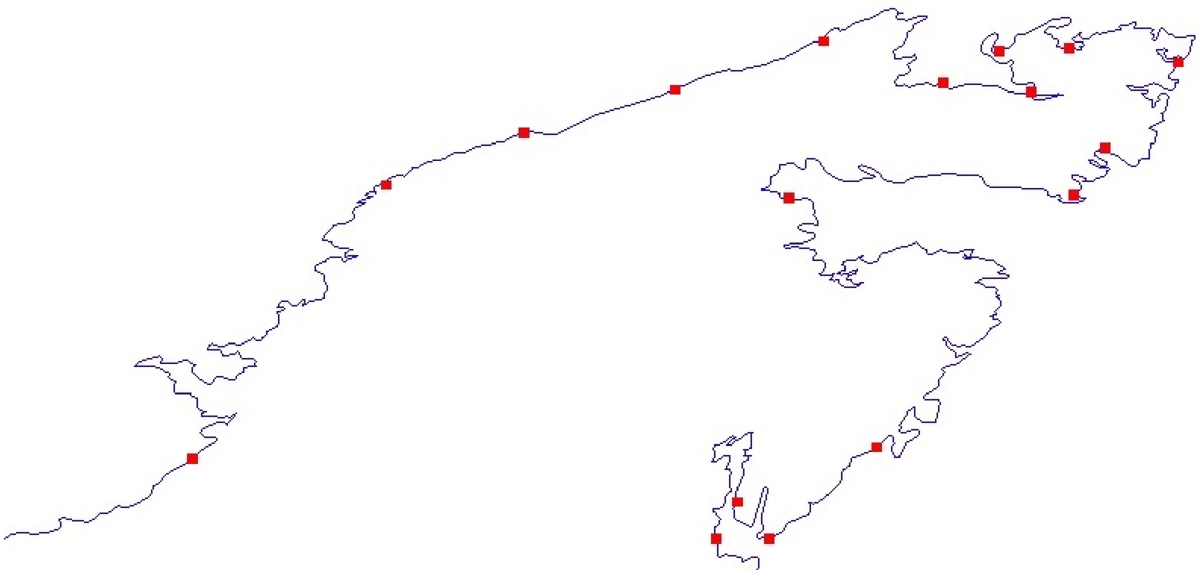

Рис. 1. Example of a segmentation. The endpoints of the segments are marked by square red dots.

Step 5: **Smoothing.** Form a smoother result by approximating the sections with B-spline curves.

The idea of using fractal dimension for cartographic generalization is one of the popular modern approaches [4]. Earlier it was discussed by one of the authors in the book [1]. The main new idea of the algorithm is the automatic segmentation of the curve into pieces of roughly uniform curvature. We do this in a bottom-up fashion, merging edges into progressively longer sections. We describe the details of this process in six steps, which implement Step 2 of overall algorithm. Recall that Step 1 has already been completed so that all edges of the curve have roughly the same length.

1. Choosing a positive integer n, we decompose the curve into section of $n-1$ edges each. Writing $P_i \in \mathbf{R}^2$ for the i-th point of a section, we get the section as the curve $L = (P_1, P_2 \ldots, P_n)$.

2. At each interior point P_i of L, we compute the absolute turning angle as the supplement of the angle between the two edges meeting at P_i, that is:

$$\gamma_i = \arccos\langle \frac{P_{i+1} - P_i}{\| P_{i+1} - P_i \|}, \frac{P_i - P_{i-1}}{\| P_i - P_{i-1} \|} \rangle \quad (1)$$

$$= \arccos \frac{(x_{i+1} - x_i)(x_i - x_{i-1}) + (y_{i+1} - y_i)(y_i - y_{i-1})}{\rho^2}, \quad (2)$$

where we write $P_i = (x_i, y_i)$ and assume that both edges have length ρ.

[1] The work was supported by a grant 11.G34.31.0053 from the government of the Russian Federation.

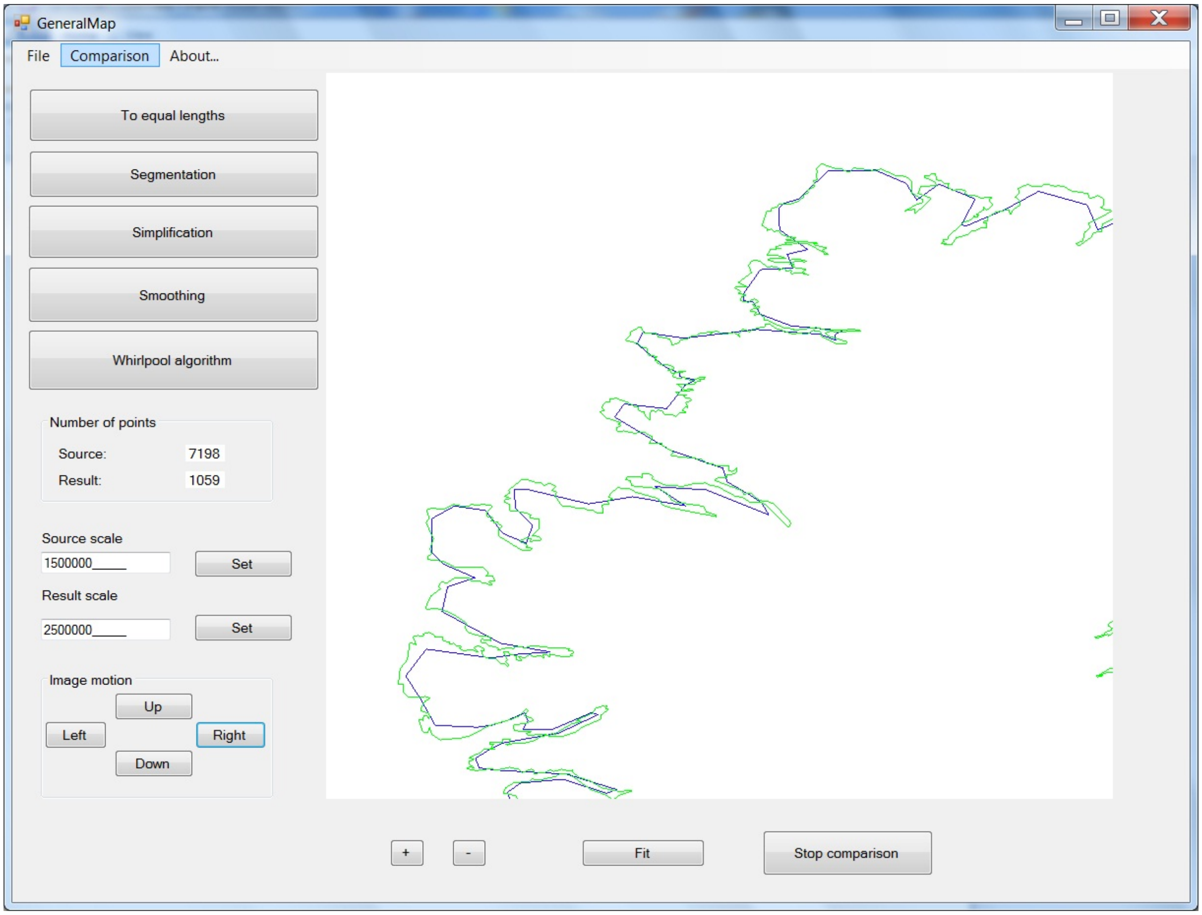

Рис. 2. Result of generalization with our software. The green curve is the given input, and the blue line shows the result of our generalization.

3. Modeling the curvature of P_i by the turning angle, we compute the total curvature by taking the sum over all interior vertices :

$$\gamma(L) = \sum_{i=2}^{n-1} \gamma_i.$$

   Normalizing the result, we call $\gamma(L)/(2\pi)$ the *total variation* of the section.

4. An interior point P_i is a local minimum of the curvature function if $\gamma_{i-1} > \gamma_i < \gamma_{i+1}$, and it is a local maximum if $\gamma_{i-1} < \gamma_i > \gamma_{i+1}$, where we set $\gamma_0 = \gamma_n = 0$ at the ends. Counting the local minima and maxima, we write $E(L)$ for their number.

5. Using a positive constant, c, we combine the total variation and the number of local extrema to get $M(L) = \gamma(L)/\pi + cE(L)$, which we refer to as *characteristic* of the section L.

6. We merge the sections with minimum difference in characteristics, M, iterating as

long as the number of sections is greater than same given number, N_{seg}, and the number of points per section is less than some other given number, N_{vert}.

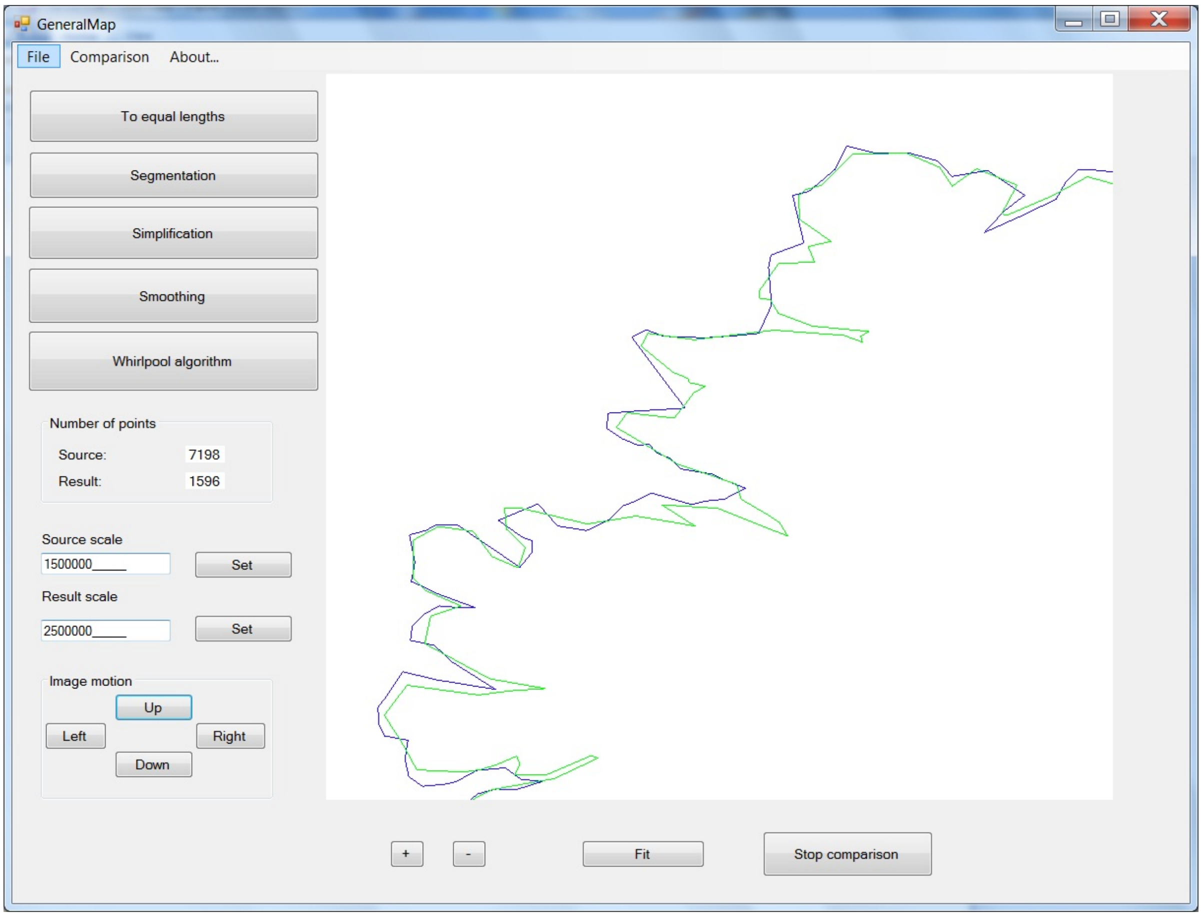

Рис. 3. Comparison of our result (green line) with result of the Whirlpool algorithm (blue line).

We refer to the sections produced at the end of the six steps as the segments, which define the segmentation of step 2 of the overall algorithm.

Figure 1 shows the example of segmentation produced by this algorithm. The computer program used to produce this result requires the original scale and final scale as input parameters. Other parameters, such as the lengths of the initial sections, n, the maximum number of vertices per segment, N_{vert}, the minimum number of segments, N_{seg}, and the weight of the number of locally extreme vertices, c, can be set as default values or chosen by the user. Figure 2 shows the result of our algorithm for a curve given by 7,198 points. Changing the scale from 1:15 to 1:25, and using default setting for the other parameters ($n = 5$, $N_{vert} = 198$, $N_{seg} = 133$, $c = 0.5$), the algorithm reduces the number of points from 7,198 to 1,059. For comparison, Figure 3 shows the result of our algorithm together with that of the Whirlpool algorithm [2], which retains 1,596 of the points. We see that our algorithm maintains the shape of the curve more accurately while retaining a smaller number of points.

Список литературы

1. A. M. Berlyant, O. R. Musin and T. V. Sobchuk. Cartographic generalization and fractal theory. Moscow, 1998.
2. E. Fritsch. Use of Whirlpool algorithm for ADBS data generalization. ADBS meeting, 1999.
3. D. H. Douglas and T. K. Peucker Algorithms for the reduction of the number of points required to represent a digitized line or its caricature. *Canadian Cartographer* **10** (1973), 112-122.
4. Bin Jiang, Xintao Liu and Tao Jia. Scaling of geographic space as a universal rule for mapping or cartographic generalization. 2011. http://arxiv.org/abs/1102.1561v1

Fractal and Computational Geometry for Generalizing Cartographic Objects

Keywords: cartographic generalization, segmentation, curvature, fractal dimension,

We present an algorithm for simplifying linear cartographic objects and results obtained with a computer program implementing this algorithm.

Сведения об авторе: Edelsbrunner Herbert, IST Austria (Institute of Science and Technology Austria), Klosterneuburg, Austria, and Departments of Computer Science and Mathematics, Duke University, Durham, North Carolina, USA
Musin Oleg Rustumovich, The University of Texas at Brownsville, Texas, USA
Ukhalov Alexey Ur'evich, P.G. Demidov Yaroslavl State University, associate professor
Yakimova Olga Pavlovna, P.G. Demidov Yaroslavl State University, associate professor
Alexeev Vladislav Vladimirovich, P.G. Demidov Yaroslavl State University, student
Bogaevskaya Victoriya Grigor'evna, P.G. Demidov Yaroslavl State University, student
Gorohov Andrey Aleksandrovich, P.G. Demidov Yaroslavl State University, student
Preobrazhenskaya Margarita Mikhailovna, P.G. Demidov Yaroslavl State University, student