

Алгоритм картографической генерализации, сохраняющий топологию*

В. В. АЛЕКСЕЕВ

Ярославский государственный университет им. П. Г. Демидова
e-mail: thekindbeetle@gmail.com

В. Г. БОГАЕВСКАЯ

Ярославский государственный университет им. П. Г. Демидова
e-mail: victory36@mail.ru

М. М. ПРЕОБРАЖЕНСКАЯ

Ярославский государственный университет им. П. Г. Демидова
e-mail: rita-uvarova@yandex.ru

А. Ю. УХАЛОВ

Ярославский государственный университет им. П. Г. Демидова
e-mail: alex-uhalov@yandex.ru

Х. ЭДЕЛЬСБРУННЕР

Институт науки и технологии Австрии
e-mail: edels@ist.ac.at

О. П. ЯКИМОВА

Ярославский государственный университет им. П. Г. Демидова
e-mail: olga_pavl02@mail.ru

УДК 517.38

Ключевые слова: картографическая генерализация, карта, изменение масштаба, сохранение топологии, теория графов, стягивание рёбер.

Аннотация

В работе предлагается алгоритм генерализации картографических объектов, который может быть использован для представления карты в различных масштабах.

Abstract

V. V. Alexeev, V. G. Bogaevskaia, M. M. Preobrazhenskaya, A. Yu. Ukhlov, H. Edelsbrunner, O. P. Yakimova, An algorithm for cartographic generalization that preserves global topology, Fundamentalnaya i prikladnaya matematika, vol. 18 (2013), no. 2, pp. 5—12.

We propose an algorithm for the generalization of cartographic objects that can be used to represent maps on different scales.

*Работа выполнена при поддержке гранта Правительства РФ по постановлению № 220 «О мерах по привлечению ведущих учёных в российские образовательные учреждения высшего профессионального образования» по договору № 11.G34.31.0053.

1. Введение

Процесс генерализации играет важную роль при подготовке карт и других геометрических данных. *Генерализация* — это обобщение изображаемых на карте объектов в соответствии с назначением карты, масштабом отображения и особенностями территории (см. [1]). В течение долгого времени генерализация считалась субъективным процессом, требующим участия квалифицированного специалиста. В связи с развитием вычислительной техники и средств автоматизации возникла необходимость формализации этого процесса. Особенно актуальной эта задача стала при распространении систем автоматической навигации и электронных карт, доступных через интернет. Невозможно хранить на сервере данные для отображения сразу во всех масштабах, которые могут быть затребованы пользователем. Естественным оказывается хранить данные, необходимые для представления в самом подробном масштабе, и осуществлять масштабирование на лету. Одной из трудностей, возникающих при автоматическом масштабировании, является избыточность данных.

Например, изображение 1 км^2 местности в масштабе $1 : 1\,000$ занимает 1 м^2 карты, в масштабе $1 : 100\,000$ — 1 см^2 , в масштабе $1 : 1\,000\,000$ — 1 мм^2 . Изобразить местность во всех этих масштабах с одинаковой подробностью и насыщенностью невозможно. Неизбежно исключение деталей и менее значимых объектов, возрастающее по мере уменьшения масштаба. Но воздействие масштаба определяется не только ограничением места: на карте мелкого масштаба, охватывающей значительное пространство, детали теряют значение и, если их сохранить, затруднится восприятие основного содержания. Кроме того, передача лишних точек создаёт дополнительную нагрузку на сети передачи данных.

В процессе генерализации картографических данных необходимо сохранять взаимное расположение объектов. Простое усреднение данных (например, при уменьшении масштаба в 10 раз оставляется каждая сотая точка из набора исходных данных) может привести к тому, что дорога, проходящая по берегу реки, реку пересечёт, хотя моста в этом месте нет. В данной работе предлагается новый алгоритм для выполнения автоматической генерализации картографических данных, сохраняющий их топологию.

Исходными данными для предлагаемого алгоритма является набор планарных графов — *слоёв* картографических данных. Каждый слой содержит информацию о некотором виде картографических данных: границах, реках, дорогах и т. д. На рис. 1 показан пример набора исходных данных.

Наша цель — упростить данные путём уменьшения числа вершин графа с сохранением глобальной топологии системы, т. е. положения объектов относительно друг друга. Результатом такого упрощения является представление исходной карты в заданном масштабе.

Для генерализации данных без изменения топологии слои объединяются в специальную структуру данных и упрощаются с помощью стягивания рёбер. Ребра стягиваются в порядке, обеспечивающем минимизацию геометрической

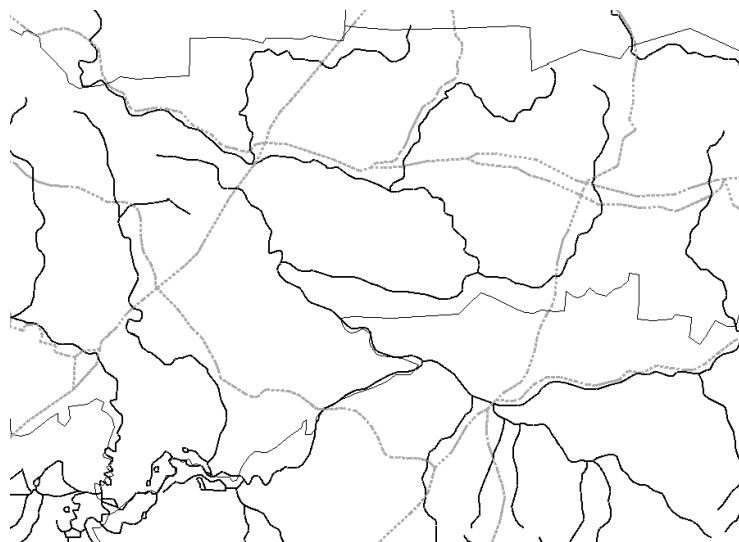


Рис. 1. Пример исходных данных. Разные слои данных показаны линиями разного типа

ошибки. Положение новых вершин, появляющихся при стягивании рёбер, выбирается из соображений минимизации ошибки. Для сохранения топологии перед стягиванием ребра выполняется проверка специального условия.

Алгоритм был реализован. Ниже приводятся результаты сравнения нашего алгоритма с другими алгоритмами картографической генерализации на некотором наборе данных.

2. Основные идеи алгоритма

Основной операцией предлагаемого алгоритма упрощения карты является стягивание ребра AB в точку. Ребро стягивается в точку X , заменяющую вершины A и B во всех рёбрах, инцидентных каждой из этих вершин. Положение точки X , заменяющей A и B , выбирается в соответствии с методом, изложенным в [4].

В соответствии с идеями работы [4], для каждой вершины графа вычисляется матрица квадратик, которая строится по следующему алгоритму. Пусть N — количество рёбер, инцидентных вершине A . Для каждого такого ребра рассмотрим уравнение прямой, на которой лежит i -е ребро: $a_i x + b_i y + c_i = 0$, где $a_i^2 + b_i^2 = 1$ ($i = 1, \dots, N$). Рассмотрим сумму квадратов расстояний от i -й прямой до точки с координатами (x, y) :

$$F(x, y) = \sum_{i=1}^N (a_i x + b_i y + c_i)^2 = q_{11} x^2 + 2q_{12} xy + q_{22} y^2 + 2q_1 x + 2q_2 y + q_0.$$

Построенная таким образом квадратичная форма называется *ошибкой* вершины A , а матрица этой квадратичной формы

$$Q_A = \begin{pmatrix} q_{11} & q_{12} & q_1 \\ q_{12} & q_{22} & q_2 \\ q_1 & q_2 & q_0 \end{pmatrix}$$

называется *матрицей квадрик* вершины A . Ребру AB поставим в соответствие матрицу $Q_{AB} = Q_A + Q_B$, где Q_A, Q_B — матрицы квадрик вершин A и B соответственно.

При стягивании ребра AB положение новой точки $X = (x, y, 1)^T$ определяется как решение задачи минимизации квадратичной формы с матрицей Q_{AB} , т. е. задачи

$$XQ_{AB}X^T \rightarrow \min.$$

Это минимальное значение квадратичной формы будем считать *ценой* стягивания ребра AB . Решение задачи минимизации квадратичной формы сводится к отысканию решения системы двух линейных уравнений с двумя неизвестными. В случае, если решение системы не определено однозначно, мы выбираем в качестве новой вершины X ту из вершин A, B , на которой квадратичная форма $XQ_{AB}X^T$ принимает наименьшее значение. Ценой стягивания в этом случае будем считать соответствующее значение квадратичной формы.

Стягивание ребра может изменить топологию карты. В таких случаях мы запрещаем стягивание. Более точно, мы ограничиваемся только теми стягиваниями рёбер, которые переводят карту в топологически эквивалентную: стягивание каждого ребра соответствует отображению плоскости в себя, являющемуся гомеоморфизмом.

Общие условия, позволяющие выяснить, сохраняет ли топологию стягивание ребра, приводятся в [2, 6, 7]. Для случая планарного графа эти условия формулируются очень просто: стягивание ребра AB сохраняет топологию тогда и только тогда, когда

- а) хотя бы один из концов отрезка принадлежит в точности двум рёбрам (рис. 2);
- б) вершины A и B не имеют общих соседей.

3. Алгоритм

Предлагаемый алгоритм состоит из пяти шагов. Наиболее важным является шаг 3, в ходе которого производится упрощение путём стягивания рёбер.

1. *Объединение слоёв.* Все слои данных объединяются в специальную структуру данных. Это позволяет нам упрощать все слои одновременно, сохраняя их глобальную топологию, а не только топологию одного слоя. Данные об исходной принадлежности каждого объекта к некоторому слою сохраняются.

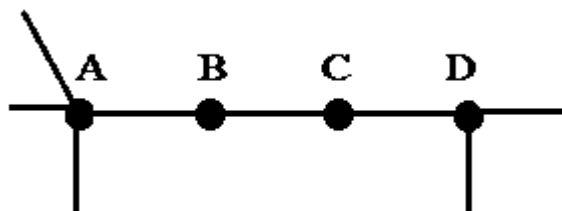


Рис. 2. Условие стягивания рёбер. Каждое из рёбер AB , BC , CD может быть стянуто, но после того как стянуто два из них, оставшееся ребро будет иметь концы с вершинами степени 4 слева и 3 справа. Следовательно, стягивание этого оставшегося ребра уже приведёт к изменению топологии

2. *Сортировка.* Для каждого ребра вычисляется цена стягивания — оценка геометрической ошибки, возникающей при стягивании данного ребра (см. [4]). Рёбра включаются в список, упорядоченный по возрастанию цены стягивания. В этом порядке рёбра будут стягиваться на следующем шаге.
3. *Упрощение.* Данные упрощаются путём стягивания рёбер в порядке возрастания цены. Упрощение прекращается после того, как число вершин графа станет меньше в процентном отношении, чем некоторая заданная величина. Возможны и другие условия остановки работы алгоритма.
4. *Сглаживание.* Полученные ломаные линии сглаживаются с помощью B -сплайнов для улучшения внешнего вида.
5. *Разделение на слои.* Упрощённая таким образом структура данных может быть разделена на слои, соответствующие слоям исходных данных.

Работа алгоритма проиллюстрирована на рис. 3, где приводится исходный слой и его упрощённая версия. Заметим, что все вершины степени 3 сохранены.

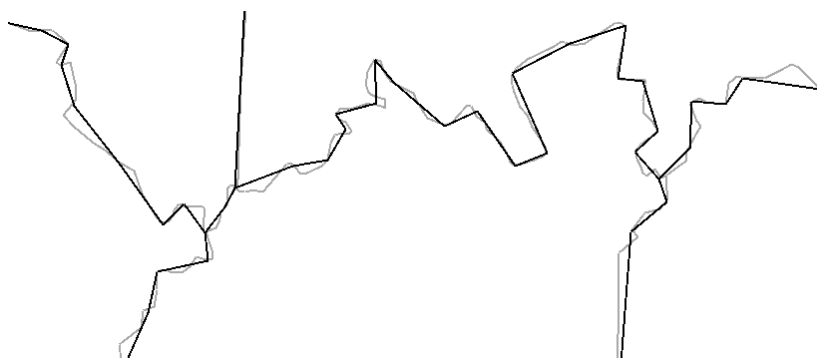


Рис. 3. Небольшой фрагмент данных, упрощённых с помощью нашего алгоритма. Серая линия соответствует исходному объекту, чёрная — упрощённому

4. Численные эксперименты

Алгоритм был реализован в виде компьютерной программы на языке C#.

Было проведено сравнение работы предлагаемого алгоритма с другими алгоритмами картографической генерализации: вихревым алгоритмом [3] и алгоритмом Ли—Опеншоу [5].

Исходные версии вихревого алгоритма и алгоритма Ли—Опеншоу предназначены для работы с ломаными линиями (графами, содержащими лишь вершины степени 1 и 2). Мы адаптировали эти алгоритмы для работы с графами: граф разбивается на набор ломаных, концевыми вершинами которых служат вершины со степенями, большими или равными 3.

Вихревой алгоритм работает следующим образом.

1. Для каждой вершины ломаной строится круг выбранного радиуса ε с центром в этой вершине.
2. Пересекающиеся круги рассматриваются как *кластеры*. Вершины, являющиеся центрами одного кластера, удаляются и заменяются на новую вершину, положение которой определяется как центр тяжести всех вершин, относящихся к данному кластеру.
3. В полученном таким образом графе удаляются циклы.

Алгоритм картографической генерализации линейных объектов Ли—Опеншоу состоит в следующем.

1. На линию накладывается регулярная сетка, шаг которой выбирается так, чтобы при требуемом масштабе одна клетка соответствовала минимальному видимому элементу изображения.
2. Отмечаются точки пересечения исходной ломаной с линиями сетки: точки, в которых линия входит в клетку и в которых она выходит из неё.
3. Середина отрезка, соединяющего точки входа и выхода линии, используется для представления ломаной в данной клетке.

В качестве меры для оценки точности аппроксимации мы используем метрику, которая может рассматриваться как обобщённое хаусдорфово расстояние. Метрика определяется следующим образом. Пусть G_i — граф, V_i и n_i — множество вершин и число вершин графа G_i , E_i — множество рёбер графа G_i . Пусть $d(X, e)$ — евклидово расстояние от точки X до ребра e . Введём обозначение $d(X, E_i) = \min_{e \in E_i} d(X, e)$. Положим

$$D_{1,2} = \max \left\{ \frac{\sum_{Y \in V_2} d(Y, E_1)}{n_2}, \frac{\sum_{X \in V_1} d(X, E_2)}{n_1} \right\}.$$

Это определение включает элементы хаусдорфова расстояния между множествами на плоскости.

Название объекта	исходное количество точек	Алгоритм, сохраняющий топологию		"Вихревой алгоритм"		Алгоритм Ли—Опеншоу	
		количество точек после упрощения	обобщенное хаусдорфово расстояние	количество точек после упрощения	обобщенное хаусдорфово расстояние	количество точек после упрощения	обобщенное хаусдорфово расстояние
Реки Ярославской, Костромской и Вологодской областей	52550	10510	837	10623	1439	10693	1081,92
Границы административных образований трех областей	14267	4855	854	4843	1389	4903	884
Границы административных образований трех областей и все водные объекты	82177	15460	969	15037	1401	15479	1177
Границы административных образований трех областей и дорожная сеть	32292	11948	773	11889	986	11942	672
Дорожная сеть и границы областей	19270	8325	780	8327	971	8401	1014

Рис. 4. Сравнение алгоритма, сохраняющего топологию, алгоритма Ли—Опеншоу и вихревого алгоритма. В таблице приводятся расстояния между исходным и упрощённым графом после того, как число вершин снизилось до некоторой заданной величины

Мы сравнили работу трёх алгоритмов на некотором наборе исходных данных, полученном при оцифровке карты масштаба 1 : 1 000 000. Результаты сравнения приводятся на рис. 4 и 5. Наши эксперименты дают основания предполагать, что предложенный в данной статье алгоритм работает лучше, чем другие рассмотренные алгоритмы картографической генерализации.

Выражаем благодарность студентам математического факультета ЯрГУ им. П. Г. Демидова А. А. Горохову и В. Н. Князеву за участие в работе по написанию программ и подготовке данных для тестирования алгоритмов.

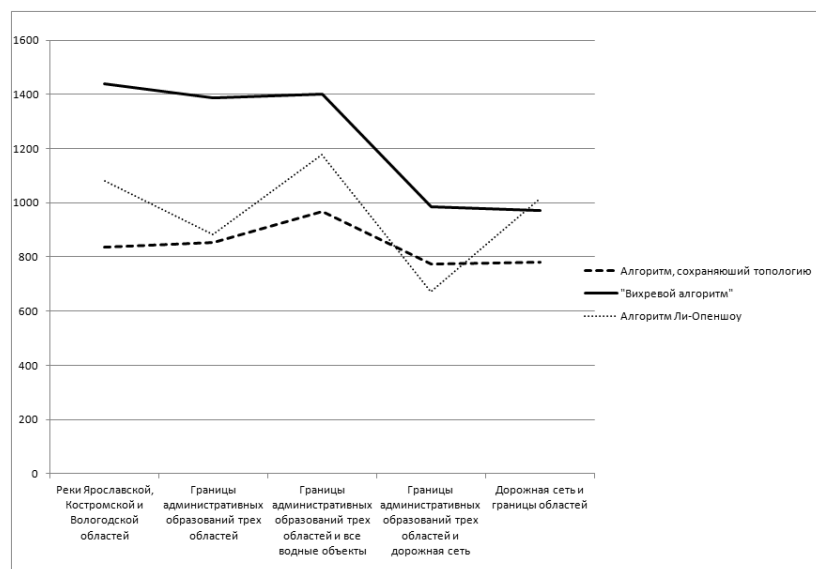


Рис. 5. Графики, иллюстрирующие численные результаты, приведённые в таблице на рис. 4: сравнение предлагаемого в нашей работе алгоритма, вихревого алгоритма и алгоритма Ли—Опеншоу

Литература

- [1] Берлянт А. М., Мусин О. Р., Собчук Т. В. Картографическая генерализация и теория фракталов. — М., 1988.
- [2] Dey T. K., Edelsbrunner H., Guha S., Nekhayev D. V. Topology preserving edge contraction // Publ. Inst. Math. (Beograd). (N. S.) — 1999. — Vol. 66. — P. 23—45.
- [3] Fritsch E. Use of Whirlpool algorithm for ADBS data generalization. — 1999. — ODBS meeting.
- [4] Garland M., Heckbert P. S. Surface simplification using quadric error metrics // SIGGRAPH '97, Proc. 24th Ann. Conf. Comput. Graphics. — New York: Addison-Wesley, 1997. — P. 209—216.
- [5] Li Z., Openshaw S. Algorithms for automated line generalization based on a natural principle of objective generalization // Internat. J. Geographical Inform. Systems. — 1992. — Vol. 6, no. 5. — P. 373—389.
- [6] Thomas D. M., Natarajan V., Bonneau G.-P. Link conditions for simplifying meshes with embedded structures // IEEE Trans. Vis. Comput. Graphics. — 2011. — Vol. 17. — P. 1007—1019.
- [7] Vivodtzev F., Bonneau G.-P., Le Texier P. Topology preserving simplification of 2D non-manifold meshes with embedded structures // Visual Computer. — 2005. — Vol. 21. — P. 679—688.