

МИЭТ

Работа допущена к защите
зав. кафедрой

Прокофьев Александр Александрович

«_____» _____ 2018 г.

ВЫПУСКНАЯ РАБОТА БАКАЛАВРА

Тема: **Упрощение топологии объектов на цифровых
географических картах**

Направление: 01.03.04 – Прикладная математика

Выполнил студент гр. МП-40

_____ Березин Александр Андреевич

Научный руководитель,

ст. преподаватель

_____ Назаров Максим Николаевич

Оглавление

1.	Введение	3
2.	Постановка задачи	3
3.	Обзор предметной области	4
3.1.	Существующие решения	4
3.2.	Теоретические сведения	4
4.	Общая методология	4
4.1.	Требование 1	4
4.2.	Требование 2	6
4.3.	Требование 3	7
5.	Реализация алгоритма	8
6.	Заключение	8
Список литературы		10

1. Введение

В работе описывается решение задачи преобразования многосвязных геометрических областей в односвязные в контексте кодирования географических карт. Полученный алгоритм работает в общем случае и может использоваться в других предметных областях.

Речь, конечно, идет о схематических картах, а не о спутниковых фотографиях. Такие карты кодируются в векторном формате. Но конкретная форма представления данных может отличаться. По условию поставленной задачи имеются две системы с различными представлениями. Одна из них поддерживает в составе карты объекты любой топологии, другая – только односвязные. Требуется разработать алгоритм для преобразования данных между этими системами.

2. Постановка задачи

В качестве входных данных имеется географическая карта, закодированная по стандарту Geography Markup Language (GML). Этот стандарт предназначен для кодирования областей различной природы – к примеру, лесных массивов, водоемов или островов – в виде вложенных многоугольников при помощи XML-грамматики [ссылка].

Требуется преобразовать входные данные таким образом, чтобы одновременно выполнялись следующие условия:

1. Отсутствуют вложенные области, т.е. все области односвязны.
2. Число точек, задающих границу каждой области, не превышает заданной константы N .
3. Количество добавочных соединений и их длины минимальны.

При этом создание новых вершин многоугольников запрещено. Возможность работы в реальном времени не требуется, поэтому оптимизация алгоритма в данной работе не затрагивается.

Понятно, что поддержка многосвязных карт автоматически означает и поддержку односвязных. Но обратное неверно. Для обратного преобразования нужно разрезать многосвязную карту, сделав ее односвязной.

3. Обзор предметной области

3.1. Существующие решения

3.2. Теоретические сведения

4. Общая методология

4.1. Требование 1

Формулировка требования: «Отсутствуют вложенные области, т.е. все области односвязны».

Многосвязная область состоит из внешней границы и произвольного числа внутренних границ, заданных многоугольниками. Задача снижения порядка связности области сводится к выбору соединений точек различных границ таким образом, чтобы все они оказались связаны. В простейшем случае с одной внутренней границей подойдет любой отрезок, один из концов которого принадлежит множеству вершин внешней границы, а другой — внутренней, и который не имеет с границами других общих точек. В случае трехсвязной области (с двумя внутренними границами) можно также соединить внутренние границы, а затем одну из них — со внешней.

Чтобы найти требуемые соединения, преобразуем объектную модель карты следующим образом. Поставим в соответствие каждому многоугольнику, включая внешнюю границу, вершину графа и сделаем полученный граф пол-

ным. В полученной графовой модели ребра соответствуют соединениям между многоугольниками. В качестве весов присвоим им длины кратчайших соединений между соответствующими многоугольниками. Далее, нужно исключить из модели недопустимые соединения — такие, которые имеют более двух общих точек с многоугольниками. Для этого будем проверять соединение на пересечение со всеми сторонами многоугольников в рассматриваемой области. Наконец, к полученному графу применим алгоритм Прима [линк] для выделения минимального остовного дерева.

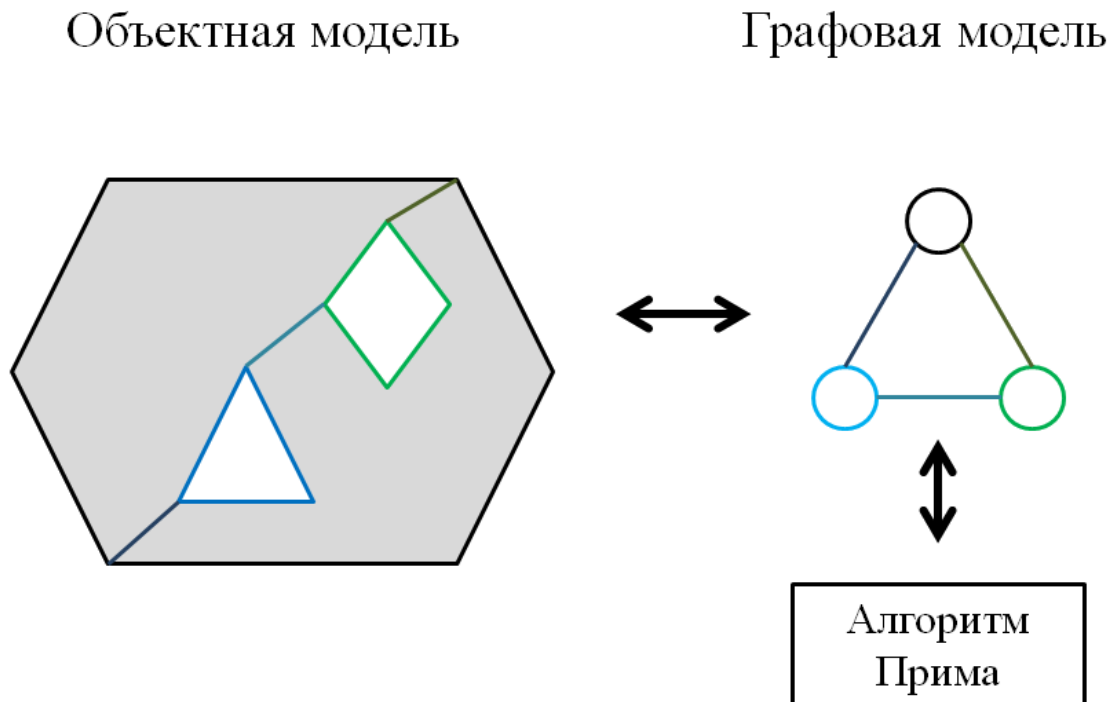


Рис. 1. Использование алгоритма Прима для удовлетворения требованию 1

Восстановление объектной модели из графовой происходит следующим образом. На внешней границе произвольно выбирается начальная точка. Если с ней не связано ни одно из соединений в графовой модели, происходит переход к следующей вершине в том же многоугольнике; в противном случае эта вершина

сохраняется в памяти, а следующей выбирается точка, находящаяся на другом конце найденного соединения, для которой та же процедура повторяется рекурсивно. Когда процесс возвращается в исходную точку, он возвращается по тому же соединению к предыдущему многоугольнику. Алгоритм завершает работу, когда возвращаться становится некуда. В результате вместо многосвязной области получается односвязная, в которой соединения становятся вырожденными туннелями.

4.2. Требование 2

Формулировка требования: «Число точек, задающих границу каждой области, не превышает заданной константы N ». На данном этапе все области уже односвязны.

Рассмотрим вначале область, задаваемую выпуклым многоугольником с более чем N вершинами. Из определения выпуклости очевидно, что ее можно разделить в произвольном отношении, т.к. любая линия, соединяющая несмежные точки, будет лежать внутри области. С учетом требования 3 следует выбрать кратчайшую линию, разделяющую многоугольник таким образом, чтобы в одной из полученных частей было ровно N вершин. Если во второй части их больше N , повторяем ту же процедуру для нее рекурсивно.

Отметим, что при разбиении выпуклого многоугольника получаются также выпуклые многоугольники, поскольку все вершины полученных многоугольников лежат по одну и ту же сторону от новой грани.

Рассмотрим теперь невыпуклые многоугольники. Согласно [1], любой многоугольник можно разбить на смежные треугольники. Очевидно, эти треугольники можно объединять по смежным сторонам, получая таким образом разбивку на части с произвольным количеством вершин. Следовательно, существует и такая разбивка, при которой одна из частей имеет ровно N вершин. Значит, описанный алгоритм можно распространить и на невыпуклые области.

4.3. Требование 3

Формулировка требования: «Количество добавочных соединений и их длины минимальны». В процессе разработки к этому условию было добавлено важное уточнение: если минимизация обоих параметров одновременно невозможна или невыполнима за полиномиальное время, приоритет имеет количество соединений.

Докажем, что после выполнения описанных выше шагов алгоритма это условие уже выполнено.

Вначале рассмотрим добавочные соединения, образовавшиеся в процессе упрощения топологии. Вспомним, что эти соединения представляют собой остовное дерево в графе всевозможных соединений, причем за вес ребра в этом графе принимается длина соединения.

По свойству дерева [2], количество его ребер фиксировано: $B - P = 1$, где B — число вершин, P — число рёбер графа. Следовательно, удаление любого из оставшихся соединений приведет к тому, что объект карты перестанет быть односвязным. Значит, условие минимальности количества дополнительных соединений выполняется.

Для выбора минимальных соединений из графа всех возможных соединений использовался алгоритм Прима. Из его корректности [3] следует, что выбранное остовное дерево графа является минимальным. Значит, условие минимальности длин дополнительных соединений выполняется. Таким образом, требование 3 удовлетворено.

Теперь рассмотрим добавочные соединения, образовавшиеся в процессе уменьшения числа вершин.

Пусть дан многоугольник с V вершинами. Тогда описанный выше алгоритм разобьет его на M частей, причем $(M - 1) * N < V + 2 * M \leq M * N$ (слагаемое $2M$ добавляется за счет того, что вершины, через которые проводится разделяющая линия, дублируются в результирующих многоугольниках).

Очевидно, что при уменьшении M на единицу количество вершин как минимум одного полученного после разбиения многоугольника будет превышать M ; следовательно, условие минимальности количества дополнительных соединений выполняется.

Одновременная минимизация добавочных соединений и их длины теоретически возможна, но также требует полиномиального времени выполнения. С учетом примечания к требованию 3, решено было ограничиться результатами разработанного алгоритма, который дает оптимальное или близкое к оптимальному решение в большинстве практических случаев.

5. Реализация алгоритма

6. Заключение

В рамках данной работы была разработана программа для программно-технического комплекса (ПТК) Автоматизированной системы диспетчерского управления (АСДУ) электрическими сетями PSIcontrol по оптимизации геоинформационных полигонных (замкнутых) структур данных на основе математической обработки входных XML-файлов по следующим критериям:

1. оптимальное перекрытие островных структур на случай их наличия в глобальной структуре без взаимных пересечений с использованием вырожденных туннелей;
2. оптимальное разрезание полигонов, содержащих значительное количество точек, на более мелкие полигоны, количество точек которых не превышает N .

Результаты работы позволили снять ряд существенных ограничений по внедрению ГИС-системы ПТК АСДУ PSIcontrol, что позволило существенно

(на два-три порядка) сократить время предварительной обработки геоинформационных подложек для ПТК.

Результаты работы внедрены в практическую деятельность Департамента энергетики ООО «ПСИ» и используются во всех текущих проектах компании по созданию и модернизации АСДУ на предприятиях российской электроэнергетики.

Список литературы

1. Chapter 3: Polygon Triangulation / М. de Berg [и др.] // Computational Geometry (2nd revised ed.) — Springer-Verlag, 2000. — С. 45—61. — ISBN 3-540-65620-0.
2. Олейник Т. Деревья // Основы дискретной математики: теория и практика. — МИЭТ, 2010.
3. Prim R. C. Shortest connection networks And some generalizations // Bell System Technical Journal. — 1957. — 36 (6). — С. 1389—1401. — DOI: [10.1002/j.1538-7305.1957.tb01515.x](https://doi.org/10.1002/j.1538-7305.1957.tb01515.x).