

Министерство образования и науки РФ
Государственное образовательное учреждение
Высшего профессионального образования
«Волгоградский государственный технический университет»
Кафедра «САПР и ПК»

ОТЧЕТ

по преддипломной практике 2016г.

Студента

Фамилия Чечеткина Имя Ильи

Отчество Александровича

Факультет ФЭВТ курс 2 группа САПР-2.1п

Индивидуальное задание: _____

РУКОВОДИТЕЛЬ

Кафедра САПР и ПК Должность профессор

Фамилия Кравец Имя Алла

Отчество Григорьевна

« » _____ 2016г.

Волгоград 2016

Оглавление

1	Методика тестирования системы кластеризации	2
2	Произведение испытаний разработанной системы	4
3	Результаты полученные в ходе работы	6
4	Структура четвертой главы магистерской диссертации	8
5	Выводы по проделанной работе	9
6	Используемые технологии в работе	10

1 Методика тестирования системы кластеризации

Алгоритмы, предложенные в рамках магистерской работы, были реализованы с использованием языка программирования Python и сервиса построения маршрутов Open Source Routing Machine для расчета расстояния между узлами графа по городским дорогам. Программный код опубликован на хостинге Github <https://github.com/vstu-cad-stuff/clustering/tree/master>.

Для оценки эффективности работы алгоритма и изучения специфики разработанной метрики были проведены эксперименты, в ходе которых менялись выборки данных, количество получаемых кластеров и их начальное местоположение, а так же реализация метода.

Для оценки работы алгоритма на данных, приближенных к реальным, были сгенерированы данные о предпочтениях по перемещению жителей среднего по размерам города с примерным числом жителей около 350 000. В качестве результата было получено 6000 пар точек отправления–назначения (рис. 1.1), или 12000 точек в общей сложности (выборка Main). Эту выборку было решено разбить на 125 кластеров, начальное положение которых было случайным. В последствие это случайное расположение кластеров было взято за основу для тестирования различных версий алгоритма.

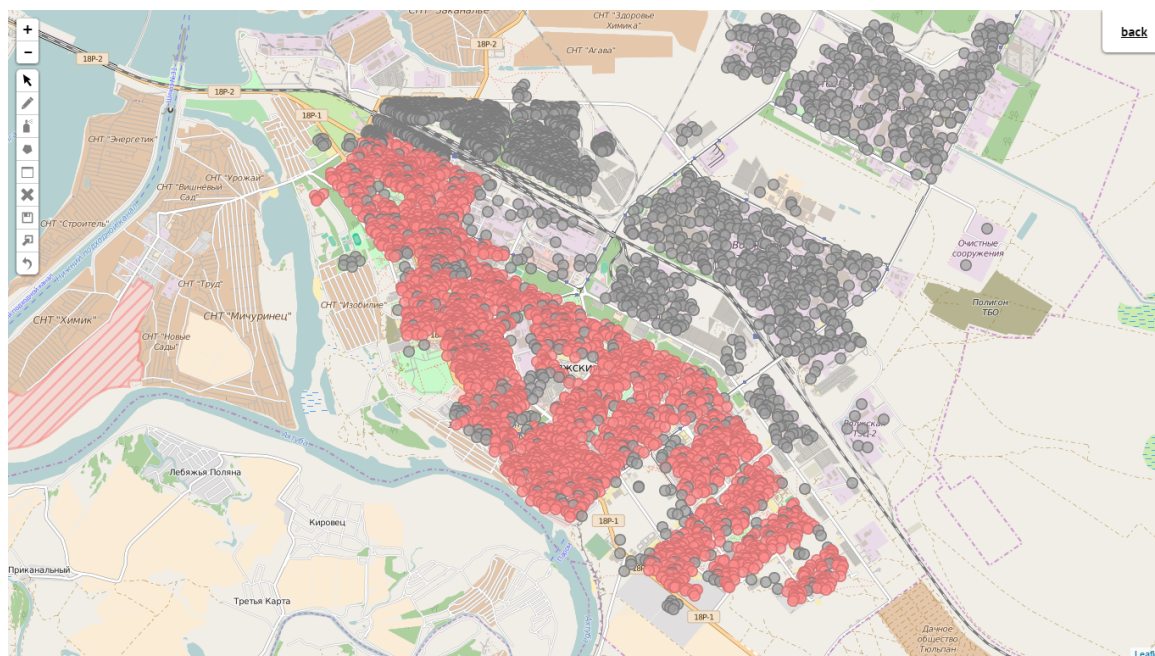


Рис. 1.1: Сгенерированная выборка из 12000 точек. Красным отмечены точки отправления, серым — назначения

Так же были сгенерированы другие выборки для проверки специфичных случаев: обхода препятствий в виде железной дороги (выборка Railway,

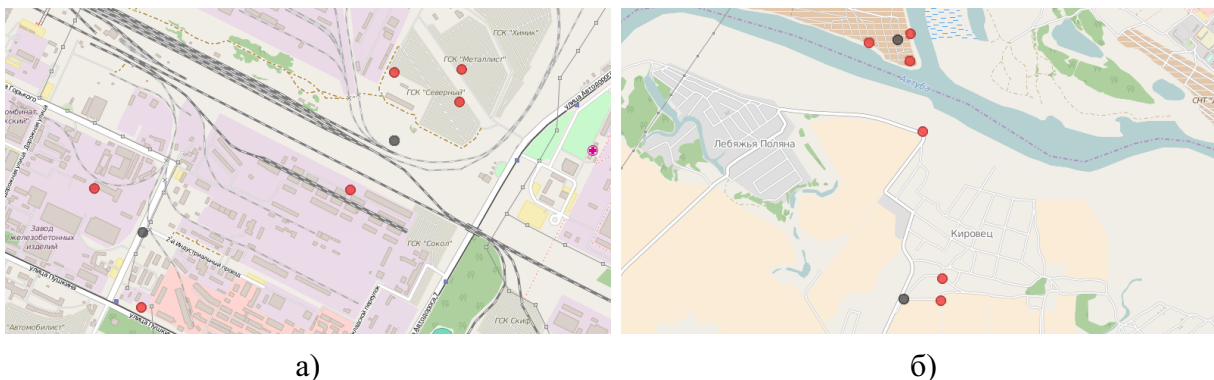


Рис. 1.2: Выборки для проверки обхода препятствий. На рисунке а) между объектами выборки находится железная дорога, на рисунке б) — река. Красными кругами отмечены элементы выборки, черными — начальные центры кластеров

рис. 1.2а) и реки (выборка River, рис. 1.2б). Каждая из них представляет собой набор из шести точек, которые кластеризуются в два кластера. Точки и начальные центры расположены таким образом, что если алгоритм не учитывает препятствия между объектами выборки, то в одном кластере окажутся точки, разделенные препятствием.

Критериями для оценки эффективности разработанных алгоритмов и метрик являются:

- время, затраченное на расчет расстояния между объектами выборки;
- учет препятствий;
- визуальная оценка результатов кластеризации.

2 Производство испытаний разработанной системы

Для тестирования эффективности работы алгоритма была использована еще одна выборка, представленная на рисунке 2.1.

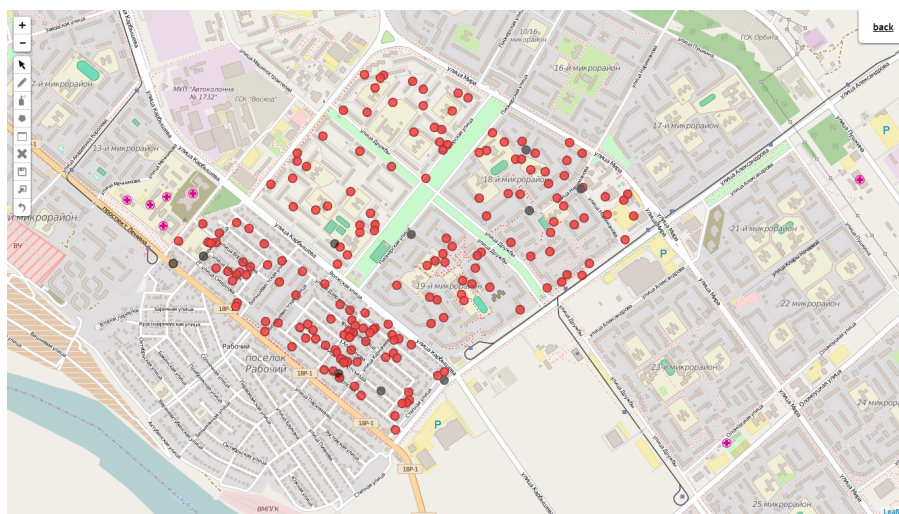


Рис. 2.1: Выборка Common. Красными кругами отмечены элементы выборки, черными — начальные центры кластеров

Таким образом, для тестирования работы алгоритма было использовано 4 выборки:

- выборка Main: $|X| = 12000$ точек, $k = 125$ (рис. 1.1); используется для общего тестирования работы алгоритмов;
- выборка Railway: $|X| = 6$ точек, $k = 2$ (рис. 1.2a); используется для проверки обхода препятствий;
- выборка River: $|X| = 6$ точек, $k = 2$ (рис. 1.2б); используется для проверки обхода препятствий;
- выборка Common: $|X| = 180$ точек, $k = 10$ (рис. 2.1); используется для тестов алгоритмов на скорость.

Чтобы понять эффективность работы алгоритма в зависимости от окружающей среды, были разработаны три альтернативных реализации:

1. Последовательная реализация алгоритма, рассчитывающего расстояние между объектами по прямой, не учитывая препятствия между ними (метрика Surface). Эту реализацию можно рассматривать в качестве базовой.

2. Последовательная реализация с использованием OSRM — это усовершенствованная версия предыдущей стратегии, где расстояние между объектами рассчитывается с использованием движка маршрутизации OSRM по дорожной сети (метрика Route).
3. Параллельная версия с использованием OSRM предполагает возможность в распараллеливании внутреннего цикла алгоритма.

3 Результаты полученные в ходе работы

Результаты кластеризации выборок Railway и River для обеих метрик представлены на рисунках 3.1 и 3.2.

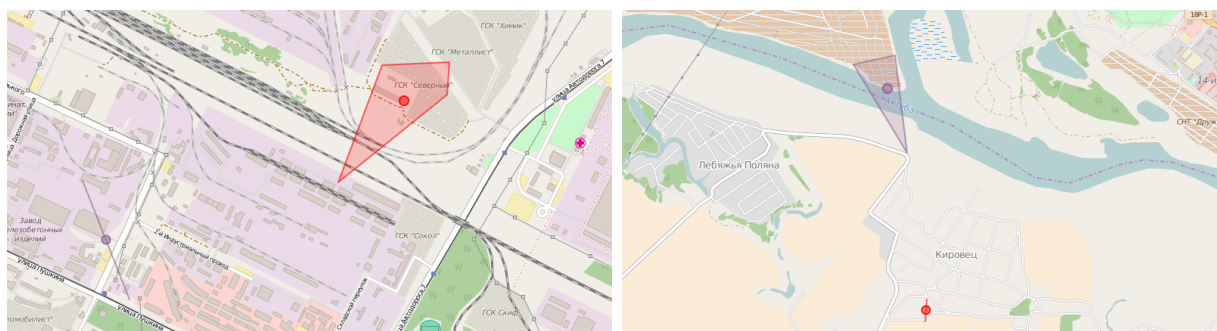


Рис. 3.1: Результаты кластеризации выборок Railway и River с метрикой Surface

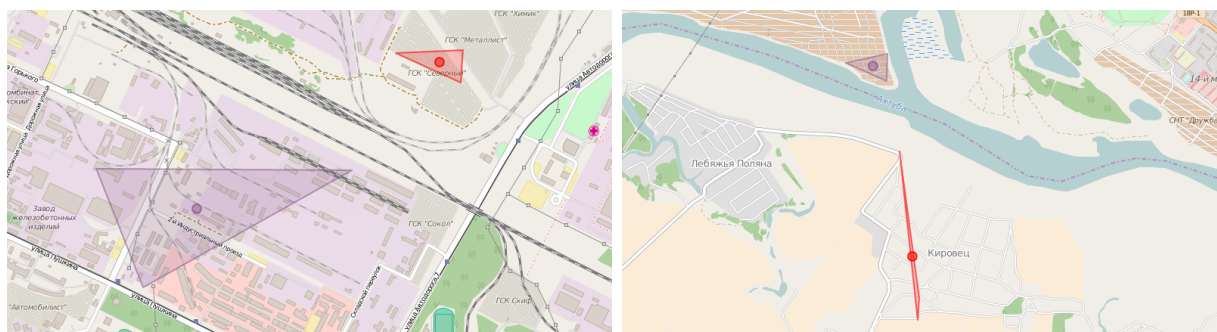


Рис. 3.2: Результаты кластеризации выборок Railway и River с метрикой Route

Результаты обработки выборки Main представлены на рисунках 3.3, 3.4.

В таблице 3.1 приведены средние длительности выполнения одного расчета дистанции между геопространственными объектами для трех реализаций алгоритма: последовательной, параллельной на два потока и параллельной на 4 потока, и для трех метрик: Euclid, Surface и Route. Метрика Euclid является обычной евклидовой метрикой: $\rho(a, b) = \sqrt{(a.x - b.x)^2 + (a.y - b.y)^2}$ и приводится для сравнения. Величины приведены в миллисекундах.

Таблица 3.1: Среднее время выполнения одного расчета расстояния при различных метриках и реализациях алгоритма, мс

Реализация	Euclid	Surface	Route
Последовательная	0,0665	0,709	4,785
Параллельная (2)	0,0659	0,692	4,069
Параллельная (4)	0,0654	0,663	3,596

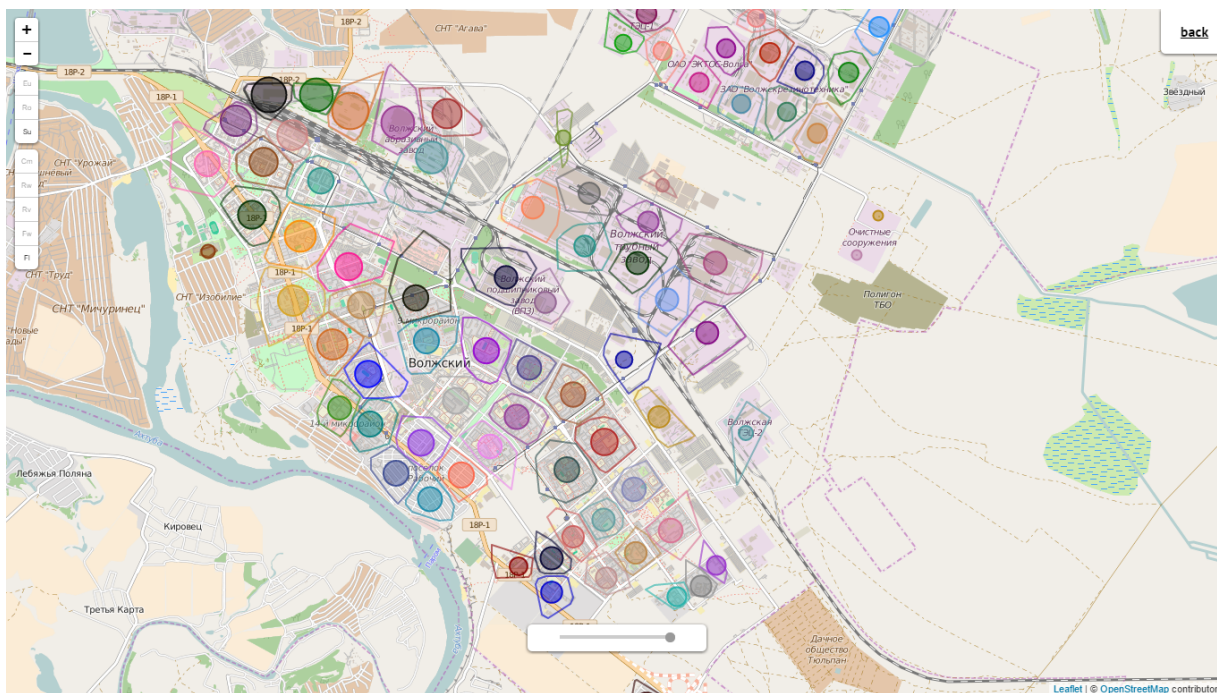


Рис. 3.3: Результаты обработки выборки Main метрикой Surface

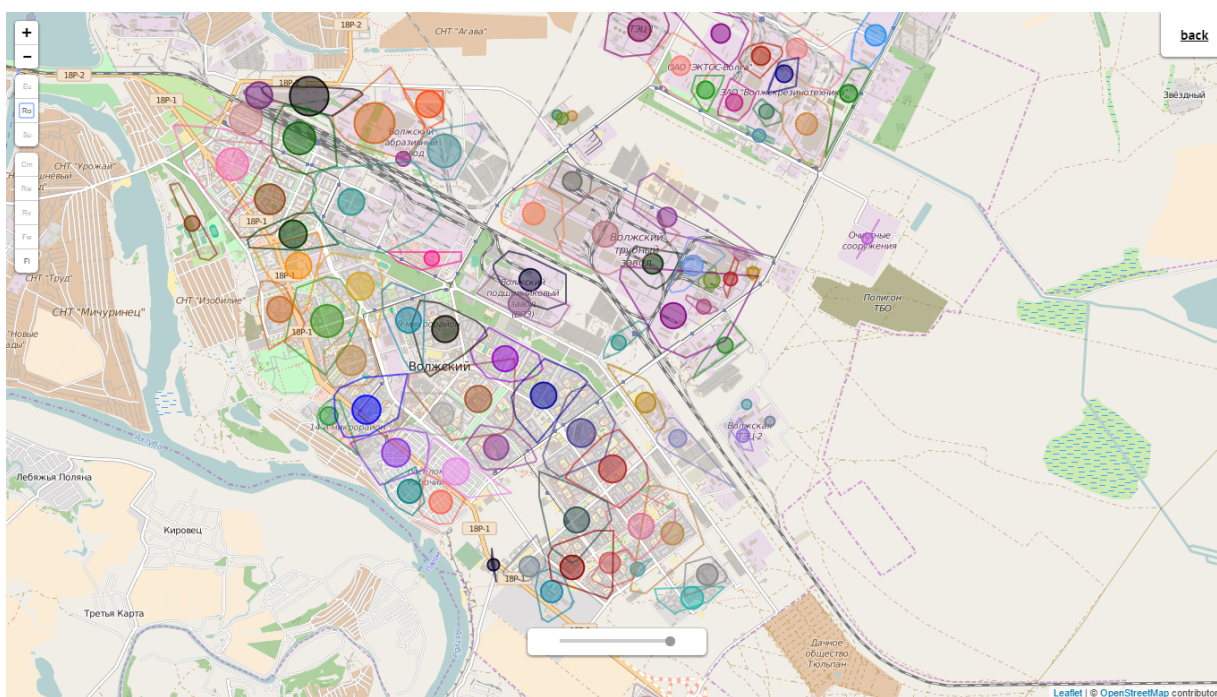


Рис. 3.4: Результаты обработки выборки Main метрикой Route

4 Структура четвертой главы магистерской диссертации

В структуре четвертой главы магистерской диссертации были выделены следующие пункты:

1. Испытание и обоснование эффективности предлагаемых подходов

(a) Проектирование ПО

(b) Методика проведения эксперимента

2. Описание результатов испытания

(a) Проведение эксперимента и описание результатов

i. Последовательная реализация

ii. Последовательная с использованием OSRM

iii. Параллельная с использованием OSRM

iv. Результаты

(b) Обсуждение результатов

(c) Выводы

В разделе «Испытание и обоснование эффективности предлагаемых подходов» описана методика тестирования разработанных алгоритмов и данные используемые для проведения тестирования.

В разделе «Описание результатов испытания» были предложены три альтернативных реализации алгоритма для оценки его эффективности в зависимости от входных данных. В данном разделе производится оценка эффективности альтернативных реализаций и их общее сравнение. Результатом работы являются визуализации работы данных реализаций, представленные на рисунках 3.3 и 3.4, а также данные в таблице 3.1.

5 Выводы по проделанной работе

В ходе преддипломной практики были выполнены следующие задачи:

- разработана методика тестирования системы кластеризации геораспределенных данных;
- произведены испытания разработанной системы на тестовом наборе входных данных;
- произведена оценка разработанной системы и обоснована эффективность используемых алгоритмов.

6 Используемые технологии в работе

Для проектирования и верстки были использованы следующие свободно распространяемые программные продукты:

- Linux Mint – это i686/x86-64 дистрибутив GNU/Linux общего назначения, разрабатываемый на основе Ubuntu/Debian.
<https://www.linuxmint.com/>
- T_EX– система компьютерной верстки, разработанная американским профессором информатики Дональдом Кнутом в целях создания компьютерной типографии. <http://tug.org/>
- L^AT_EX– набор макрорасширений системы компьютерной верстки T_EX.
<http://www.latex-project.org/>
- Atom – свободный открытый текстовый редактор исходных текстов программ. <https://atom.io/>
- Python – высокоуровневый язык программирования общего назначения, ориентированный на повышение производительности разработчика и читаемости кода. <https://www.python.org/>
- Leaflet – Open-Source библиотека, предназначенная для отображения карт на веб-сайтах. <http://leafletjs.com/>
- Git – распределенная система управления версиями файлов. Проект был создан Линусом Торвальдсом для управления разработкой ядра Linux, первая версия выпущена 7 апреля 2005 года.
<http://git-scm.com/>
- GitHub – самый крупный веб-сервис для хостинга IT-проектов и их совместной разработки. Основан на системе контроля версий Git и разработан на Ruby on Rails и Erlang компанией GitHub, Inc.
<https://github.com/>