

МОСКОВСКИЙ ФИЗИКО-ТЕХНИЧЕСКИЙ
ИНСТИТУТ

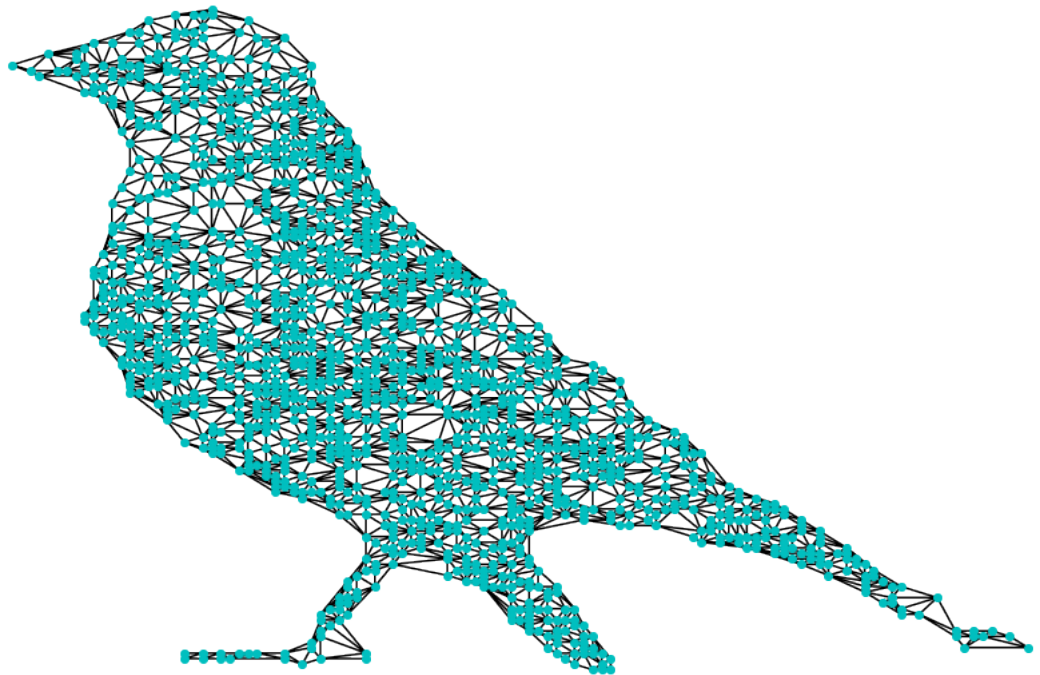
«АЛГОРИТМЫ ВЫЧИСЛИТЕЛЬНОЙ ГЕОМЕТРИИ»

ЛАБОРАТОРНАЯ РАБОТА №2

Построение и анализ триангуляции Делоне

Студент:
Алексей Гришанов
гр. М05-004

Преподаватель:
Леонид Моисеевич
Местецкий



1 Постановка задачи

Нужно оценить периметр и площадь фигуры на основе аппроксимации границы многоугольником. Предлагается разработать программу аппроксимации фигуры на основе триангуляции Делоне заданного множества точек.

Программа должна обеспечить:

- Ввод заданного массива точек, входящих в множество S .
- Вывод периметра и площади фигуры.
- Визуализацию исходного множества точек и полученной оболочки.

2 Описание данных

На плоскости задано облако из n точек, изображающих некоторую фигуру («рыбы» и «птицы»), примеры представлены на рисунке.



Исходные данные задаются в текстовом файле. Птицам соответствуют файлы «п_1.txt» - «п_52.txt», рыбам — «р_1.txt» - «р_53.txt».

Первая запись в файле — число точек, далее координаты точек.

3 Описание метода решения

1. Построим триангуляцию Делоне. Для этого воспользуемся реализацией из пакета `scipy.spatial`.

Самая затратная по времени часть здесь, это построение выпуклой оболочки, асимптотика данного этапа $O(n \log n)$.

2. Удалим «длинные» ребра, чья длина превышает заданный порог. В качестве порога была выбрана длина $\sqrt{50}$ (в реализации не извлекался корень из квадратов длин сторон, квадраты сторон > 50 удалялись). При удалении стороны удалялся весь треугольник, её содержащий.

В триангуляции $O(n)$ треугольников \Rightarrow этот этап выполняется за линейное по числу точек время.

3. Построим оболочку-огibaющую оставшегося подграфа в виде многоугольника. Заметим, что в оболочке лежат те и только те стороны, которые принадлежат лишь одному треугольнику. Пройдя по всем сторонам, оставим только те, которые удовлетворяют данному условию.

Этот этап также выполняется за линейное время.

4. Вычислим периметр и площадь полученного многоугольника. Периметр равен сумме длин отрезков из оболочки, площадь — сумме площадей треугольников, оставшихся после удаления длинных ребёр.

Здесь требуется также линейное время.

Итоговая асимптотика алгоритма $O(n \log n)$, по самому трудоёмкому первому этапу.

4 Описание программной реализации

Алгоритм реализован на python. Использовалась библиотека `scipy` для построения триангуляции Делоне и библиотека `matplotlib` для отрисовки графиков.

Для запуска можно использовать либо команду

```
lab2_grishanov.py
```

и в предложенном диалоге ввести название файла для запуска, либо прописать название файла в самой команде через

```
lab2_grishanov.py --filename "CHOOSSEN_FILENAME"
```

Для сохранения финальных изображений есть дополнительный аргумент

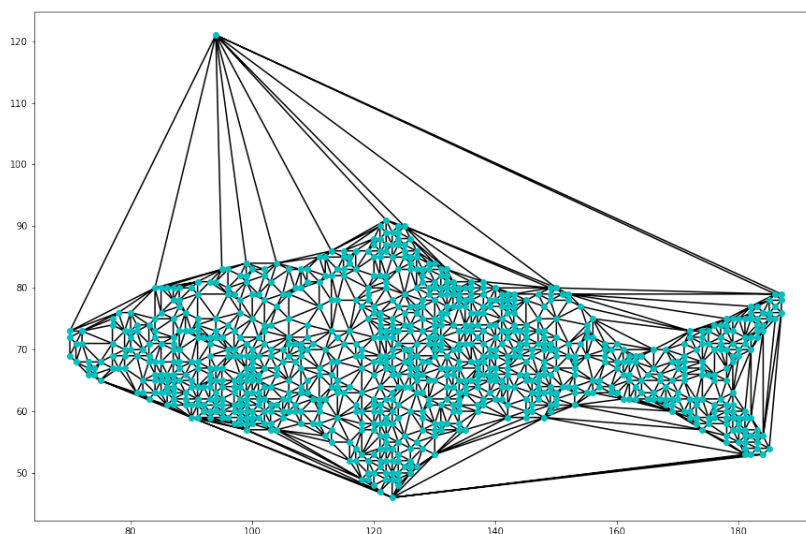
```
lab2_grishanov.py --save_fig
```

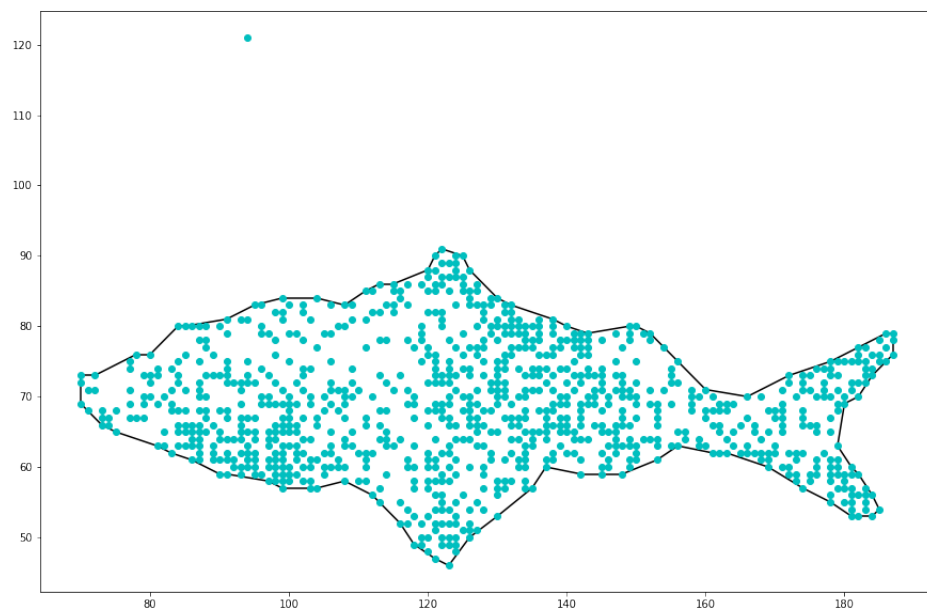
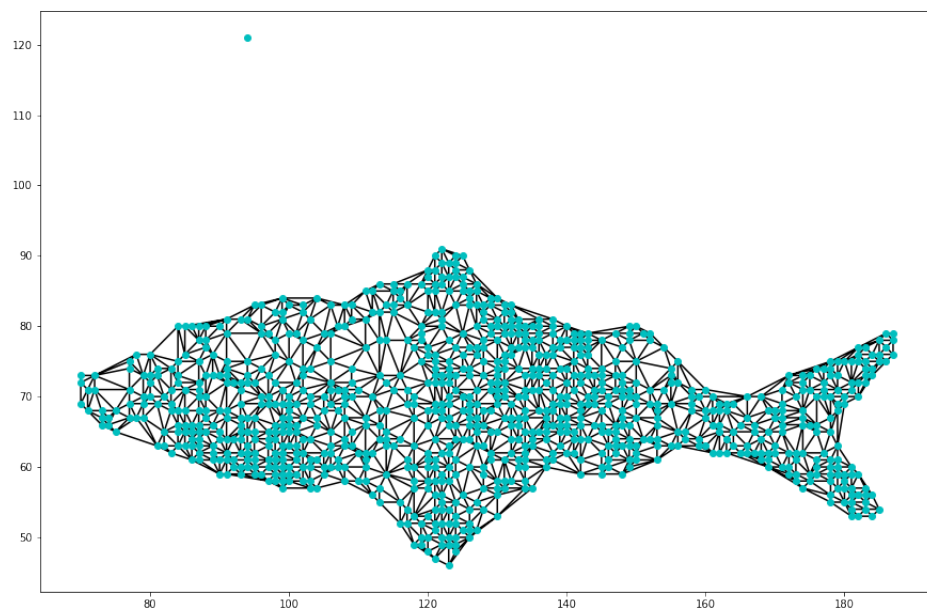
Также доступен `.ipynb` файл с примером работы алгоритма.

5 Эксперименты

Практически для всех предложенных наборов точек алгоритм строит оптимальные аппроксимации.

5.1 Пример («p_1.txt»)

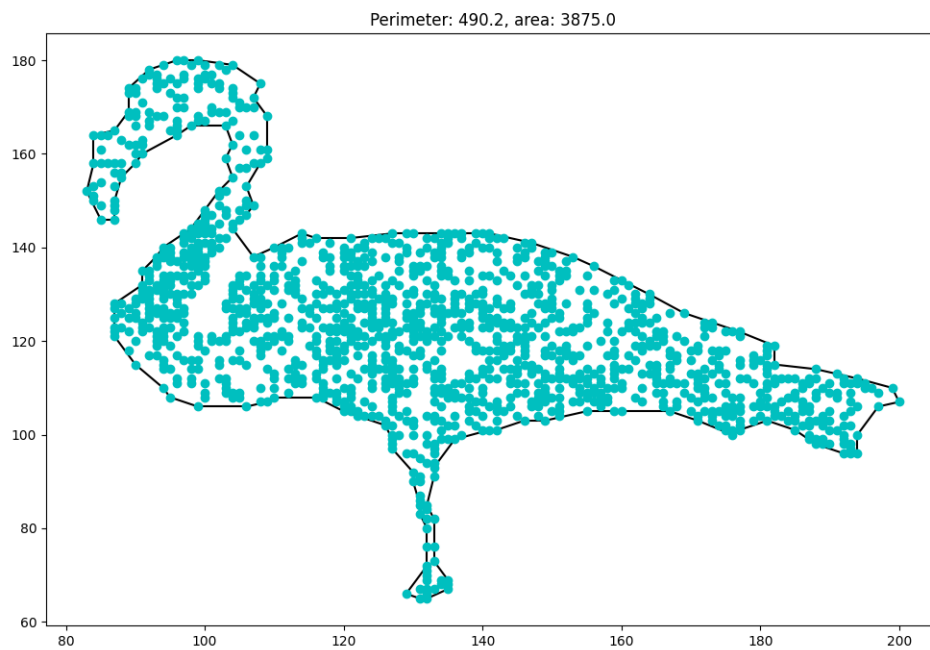




Видно что алгоритм прекрасно отработал, даже несмотря на «выброс» - точку, сильно обособленную от остальных.

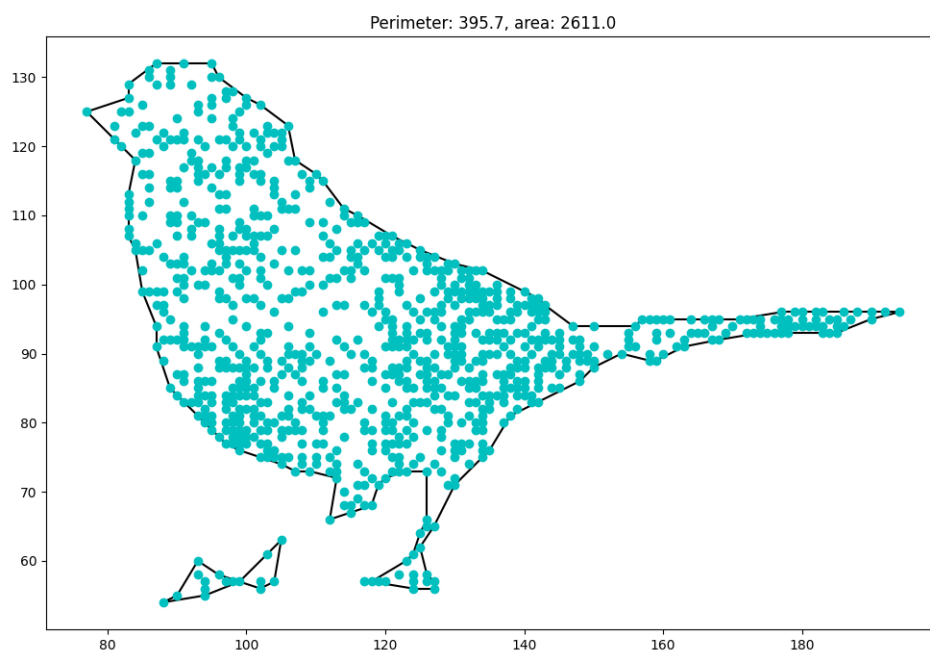
5.2 Потенциальные источники ошибок

Выявлены 2 типа неопимальных решений.



Во

первом случае алгоритм не удаляет лишнее ребро и, как следствие, увеличивает контур фигуры, её периметр и площадь.



Во

втором — алгоритм удаляет нужные рёбра.

Заметим, что даже в таких случаях итоговые величины периметра и площади не так сильно отличаются от истинных.

6 Выводы

Разработан алгоритм с асимптотикой $O(n \log n)$, который по набору точек строит аппроксимирующий контур фигуры, вычисляет ее площадь и периметр.

Качество алгоритма напрямую зависит от качества данных. При плотном расположении точек, без пропусков точек в узкой части и U-образных участков, практически невозможны отклонения от идеальных результатов.

Ни в одном из примеров предложенной выборки не наблюдается сильных погрешностей. Это позволяет сказать, что данный алгоритм успешно решает поставленную задачу,