Московский Государственный
Университет

Паросочетания в плоских графах

Выполнил: Курцев Д.В.
Группа: 517

Факультет Вычислительной математики и кибернетики
Кафедра Математических методов прогнозирования

Декабрь 2023

1 Постановка задачи

На плоскости заданы два множества точек: B (черные) и W (белые) по n точек в каждом. Никакие три точки не лежат на одной прямой.

Правильным паросочетанием называется множество из n пар (b, w), $b \in B$, $w \in W$ такое, что отрезки прямых, соответствующие этим парам, не пересекаются.

1) Доказать, что всегда существует прямая, проходящая через черную и белую точки, для которой количество черных точек, попавших в одну из полуплоскостей относительно этой прямой, равно количеству белых точек в этой же полуплоскости. Опишите, как найти эту прямую за время $O(n \log n)$.

2) Разработать алгоритм, позволяющий в течение времени $O(n^2 \log n)$ построить правильное паросочетание. Необязательный критерий – предпочтительным является решение, минимизирующее среднюю длину пути между точками в парах.

3) Разработать и реализовать программу вычисления и визуализации правильного паросочетания для заданных множеств черных и белых точек. Программа должна обеспечить:

- ввод заданного массива точек, входящих в множества B и W;
- вывод картины паросочетания;
- подсчет и вывод средней длины пути между белыми и черными точками в парах;
- оценку реального времени работы программы по каждому эксперименту.

Исходные данные задаются в текстовом файле. Первая запись – число точек, далее в каждой записи одна точка: номер точки, координаты и цвет – черный 1, белый – 0.

2 Доказательство алгоритма

Возьмём левую нижнюю точку. Без ограничения общности будем считать, что она чёрная. Поместим начало координат в эту точку. Отсортируем остальные точки по увеличению полярного угла. Далее будем поочерёдно идти по ним и считать количество чёрных и белых пройденных точек. Если мы указываем на белую точку и счётчики совпадают, значит мы нашли такую прямую.

Докажем, что такое всегда произойдёт. Если в нулевой момент мы указываем на белую точку, то данное разбиение найдено. Если нет, то заводим счётчики, как было сказано раннее. Обозначим их b_i и w_j - чёрные и белые соответственно. Заметим, что $0 = w_1 < b_1 = 1$. Так же $b_{2n-1} \leq w_{2n-1} = n-1$ так как $n-1$ максимально возможно количество точек, которое может остаться. Если в последний $(2n-1)$ момент мы указываем на белую точку, то $b_{2n-1} = w_{2n-1} = n-1$ и разбиение найдено. Если на чёрную, то $b_{2n-1} < w_{2n-1} = n-1$. Значит существовал раннее такой момент k, где $b_k = w_k$ и мы указываем на белую точку. Если предположить, что такого момента не было, то тогда для $\forall j \in \{0, 2n-2\}$ было бы верно: $w_j \leq b_j$ и $w_{j+1} < b_{j+1}$, что противоречит шагу $2n-1$.

Чтд

3 Реализация алгоритма

Выполнение данной задачи было разделено на несколько функций.

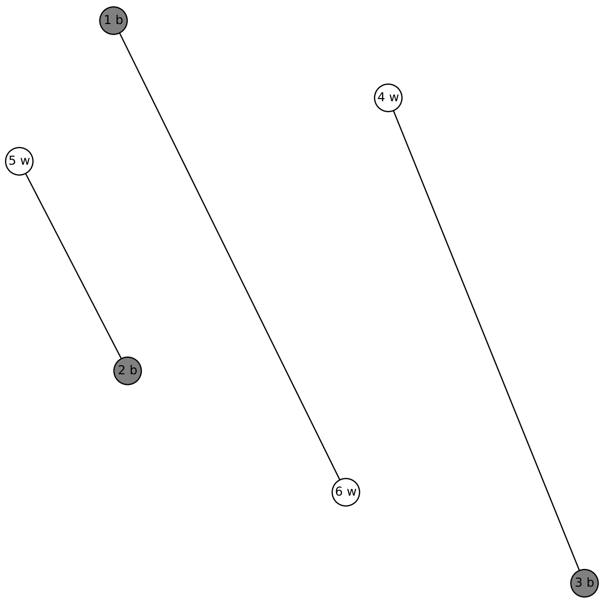
Функция $get\_down\_left\_point$ ищет нижнюю левую точку. Для этого применялась сортировка по координате y - $O(n \log n)$.

Функция get_edge строит ребро по вышеописанному алгоритму. Проход по всем точкам занимает $O(n)$ времени

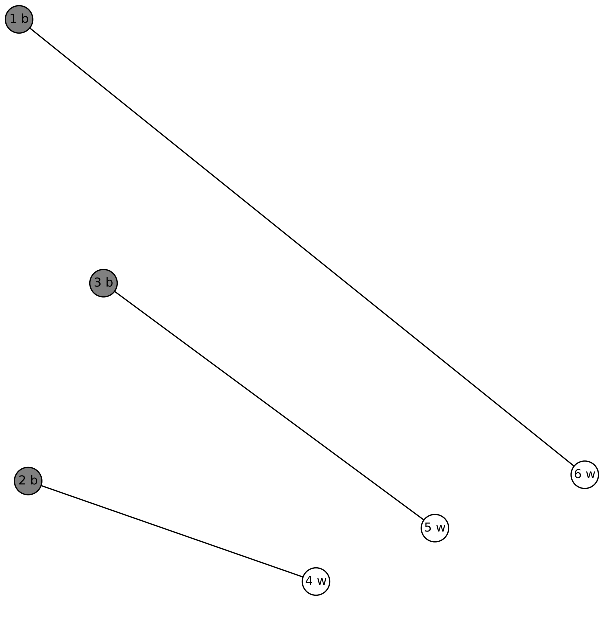
И функция $make\_pairs$ рекурсивно строит пары через функцию get_edge. Ребро разделяющее плоскость добавляется в пары и далее алгоритм точно так же применяется к оставшимся полуплоскостям.

4 Эксперименты

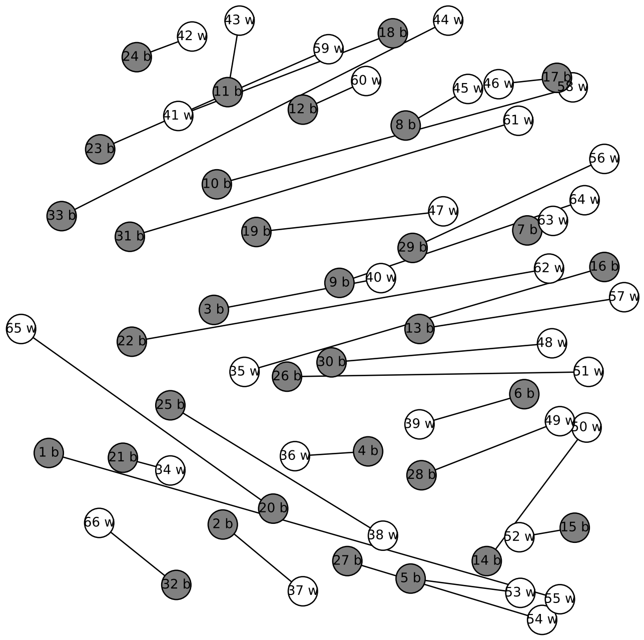
3.txt



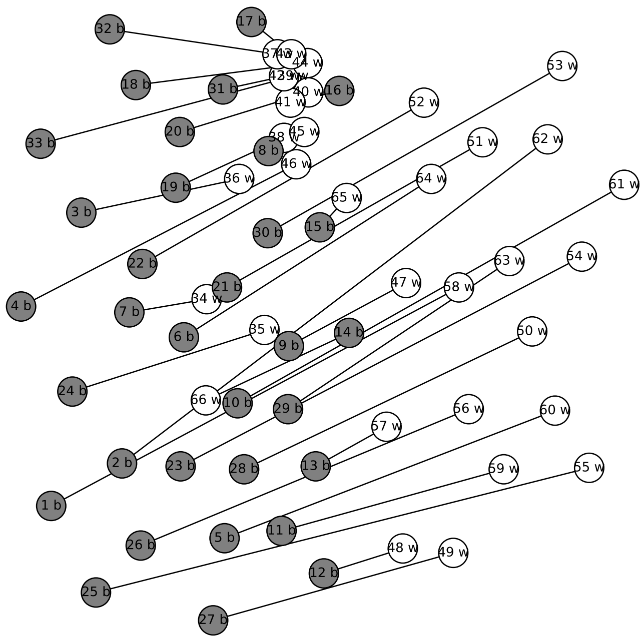
3-2.txt



33-1.txt



33-2.txt



5 Выводы

В данной работе был доказан алгоритм о разбиении плоскости и реализован другой на основе доказанного для парасочетаний точек разного цвета. Продемонстрирована его работа на нескольких экспериментах. Видно, что алгоритм быстро и верно соединил точки рёбрами без пересечений.