

Статическая задача в \mathbb{R}^d с $O(\log^{d-1}(n))$

Андрей Козлов

Январь 2015

Постановка задачи

Требуется реализовать дерево запросов принадлежности множества точек данному прямоугольнику.

Также требуется сравнить время работы запроса на дереве со временем работы наивного алгоритма.

Реализация

Реализованы классы `AbstractRangeTree`, `RangeTree2D` и `RangeTreeKD`, параметризуемые типом координаты точки.

Для координат реализован интерфейс `ICoordinate`. Тип координаты должен обладать следующими свойствами:

- существование отношения порядка;
- существование следующего элемента.

```
public interface ICoordinate<C> extends Comparable<C>> extends
    Comparable<ICoordinate<C>> {

    C getValue();

    ICoordinate<C> nextCoordinate();
}
```

В работе приведена реализация целых координат `IntegerCoordinate`. На основе абстрактного класса `AbstractCoordinate` можно реализовать вещественные координаты.

Проверка корректности

Тесты на проверку корректности проводились следующий образом:

1. генерируется прямоугольник `rectangle`;
2. генерируется набор точек `pointsInside` внутри прямоугольника `rectangle`;
3. генерируется набор точек `pointsOutside` снаружи прямоугольника `rectangle`.
4. наборы `pointsInside` и `pointsOutside` смешиваются и по ним строится дерево `tree`;
5. на дереве `tree` выполняется запрос на прямоугольник `rectangle`, ответ сравнивается с набором точек `pointsInside`.

Данные тесты проводились при различных значениях числа точек снаружи и внутри прямоугольника.

Сравнение производительности

В таблице ?? приведены результаты тестирования асимптотик операций запроса. Среднее значение бралось по результатам десяти запусков.

number of points	range query	naive algorithm
10000	1	5
20000	0	0
30000	1	1
40000	0	2
50000	0	3
100000	2	9
200000	5	20
300000	8	32
400000	10	42

Таблица 1: Среднее время выполнения запроса (мс)

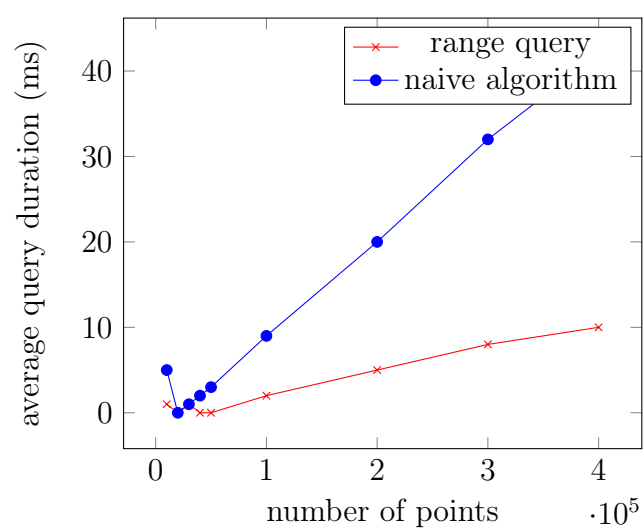


Рис. 1: График зависимости времени выполнения запроса от количества точек n