

Санкт-Петербургский национальный исследовательский университет  
информационных технологий, механики и оптики

Факультет информационных технологий и программирования  
Кафедра компьютерных технологий

## Разработка гибридного алгоритма недоминирующей сортировки

Маркина Маргарита Анатольевна  
Группа М3438

Научный руководитель: к.т.н. доцент кафедры КТ  
М. В. Буздалов

## Предметная область

- Недоминирующая сортировка.
- Многокритериальная задача оптимизации.
- Гибритизация алгоритмов.
- Оценка времени работы алгоритмов.

## Введение

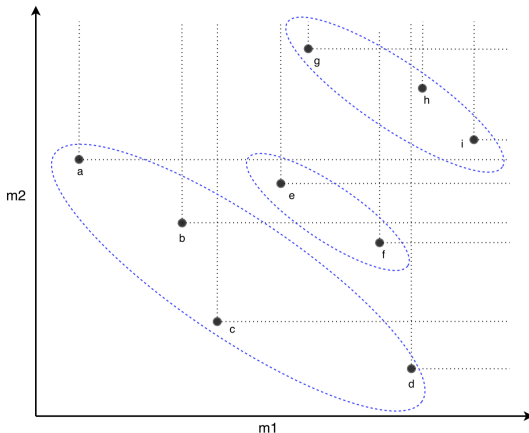
- Точка  $A = (a_1, \dots, a_M)$  доминирует точку  $B = (b_1, \dots, b_M)$ , когда  $\forall 1 \leq i \leq M : a_i \leq b_i$  и  $\exists j : a_j < b_j$ .
- Недоминирующая сортировка множества точек  $S$  в  $M$ -мерном пространстве — это процедура, назначающая всем точкам из  $S$  ранг.
- Все точки, которые не доминируются ни одной точкой из  $S$ , имеют ранг 0.
- Точка имеет ранг  $i + 1$ , если максимальный ранг среди доминирующих её точек равен  $i$ .

# Решаемая проблема

## Введение

На рисунке 3 фронта:

$\{a, b, c, d\}$  имеет ранг 0,  $\{e, f\}$  - ранг 1,  $\{g, h, i\}$  - ранг 2.



## Актуальность

- Многокритериальные эволюционные алгоритмы.
- Задача минимизации.

## Цель исследования

- Выбрать наиболее подходящие алгоритмы.
- Выявить преимущества каждого алгоритма.
- Научиться по входным данным выбирать стратегию.
- Сделать гибридный алгоритм.

## Fast + BOS

- Fast Version of the Generalized Algorithm.
- Best Order Sort.

## Fast

- Fast Version of the Generalized Algorithm.
  - Разделяй и властвуй по  $N$  и  $M$ .
  - На каждом этапе делим на 3 множества по  $k_i$  критерию текущее множество точек.
  - Если все  $k_i$  в одном из подмножеств равны между собой, переходим к  $k_{i-1}$ .
  - Запускаемся рекурсивно.



## BOS

- Best Order Sort.
  - $M$  отсортированных списков,  $i$  список отсортирован по  $i$  критерию.
  - Далее определяем ранг начиная с наиболее подходящих элементов.
  - Во время определения ранга используем уже обработанные точки.

## Асимптотика

- Fast  $O(N \log^{M-1} N)$ .
- BOS  $O(MN \log N + MN^2)$ .
  - лучшем случае – за  $\Theta(MN \log M)$
  - худшем случае – за  $\Theta(MN^2)$

## Гибритизация

- По входным данным подбирать стратегию сортировки
- В момент рекурсивного запуска мы можем переключиться на BOS.
- Оценка входных данных должна быть очень быстрая, так как она принимается неоднократно.

## Эксперименты

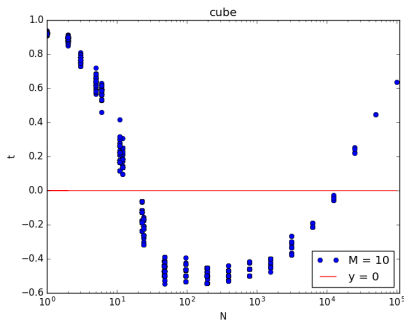
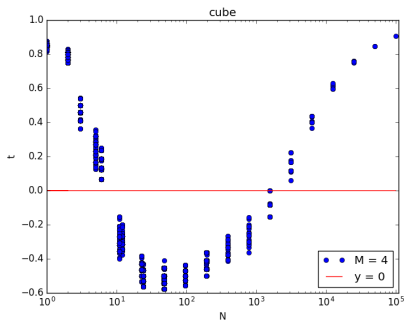
Рассмотреть влияние входных данных на время работы алгоритмов

- Рандомные точки в гиперкубе.
- Точки одного ранга.

# Практические результаты

## Рандомные точки в гиперкубе

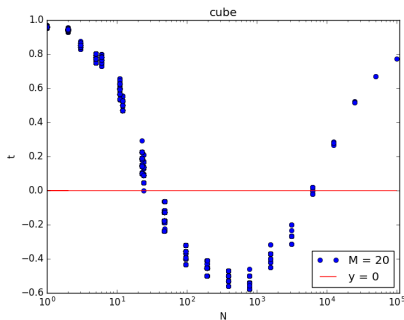
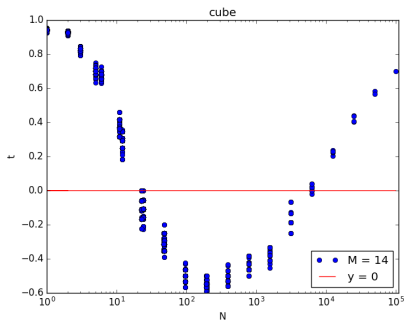
Алгоритм запускался для  $N = 100\,000$  раз при  
 $M \in \{4, 6, 8, 10, 12, 14, 16, 18, 20\}$



# Практические результаты

## Рандомные точки в гиперкубе

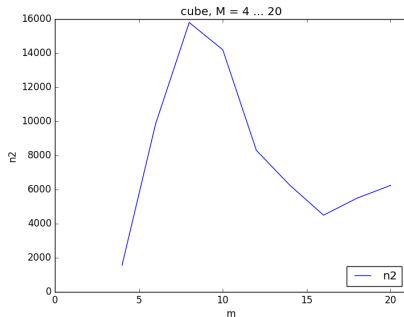
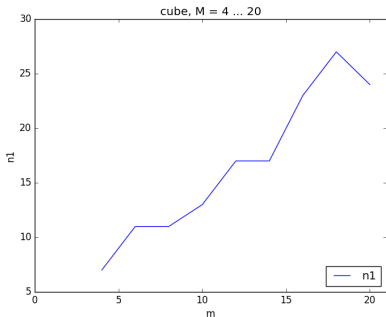
Алгоритм запускался для  $N = 100\,000$  раз при  
 $M \in \{4, 6, 8, 10, 12, 14, 16, 18, 20\}$



# Практические результаты

## Рандомные точки в гиперкубе

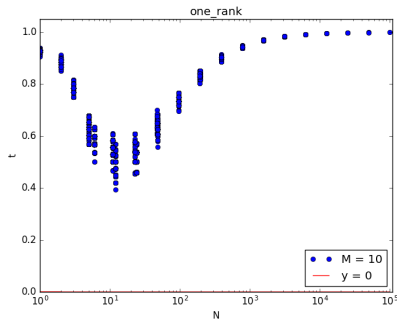
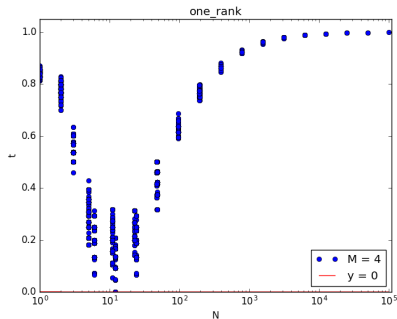
Левая и правая границы



# Практические результаты

## Точки одного ранга

Алгоритм запускался для  $N = 100\,000$  раз при  
 $M \in \{4, 6, 8, 10, 12, 14, 16, 18, 20\}$

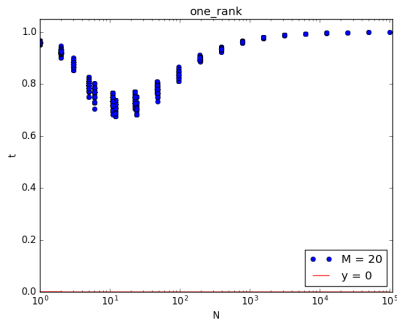
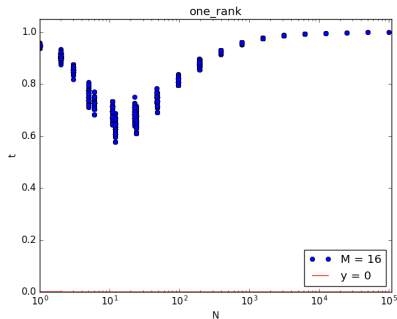




# Практические результаты

## Точки одного ранга

Алгоритм запускался для  $N = 100\,000$  раз при  
 $M \in \{4, 6, 8, 10, 12, 14, 16, 18, 20\}$



# Практические результаты

Выводы

TODO

# Дальнейшие действия

TODO

Спасибо за внимание!

TODO