

## Методы анализа и настройки гибридных алгоритмов недоминирующей сортировки

Маркина М. А., Буздалов М. В.

VII Конгресс молодых ученых  
18 апреля 2018 г.

## Введение

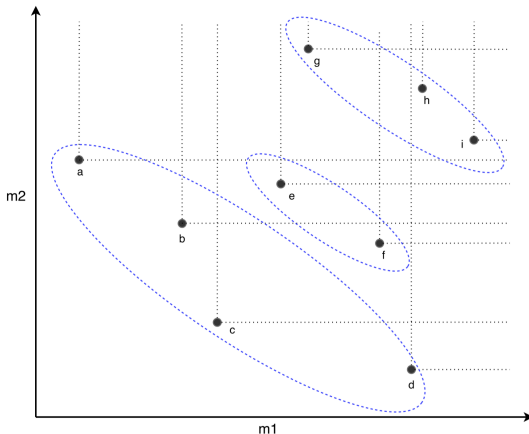
- Точка  $A = (a_1, \dots, a_M)$  доминирует точку  $B = (b_1, \dots, b_M)$ , когда  $\forall 1 \leq i \leq M : a_i \leq b_i$  и  $\exists j : a_j < b_j$ .
- *Недоминирующая сортировка* множества точек  $S$  в  $M$ -мерном пространстве — это процедура, назначающая всем точкам из  $S$  *ранг*.
- Все точки, 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.
- Алгоритм востребован в промышленных задачах.

## Цель исследования

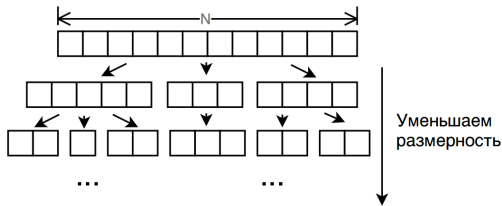
- Выбрать наиболее подходящие алгоритмы.
- Приспособить их для гибридизации.
- Разработать гибридный алгоритм.
- Настроить параметры гибридизации.

## Divide and Conquer + ENS-NDT

- **Divide and Conquer.** M. Buzdalov, A. Shalyto. A Provably Asymptotically Fast Version of the Generalized Jensen Algorithm for Non-Dominated Sorting. (2014)
- **ENS-NDT.** P. Gustavsson, A. Syberfeldt. A new algorithm using the non-dominated tree to improve non-dominated sorting. (2017)

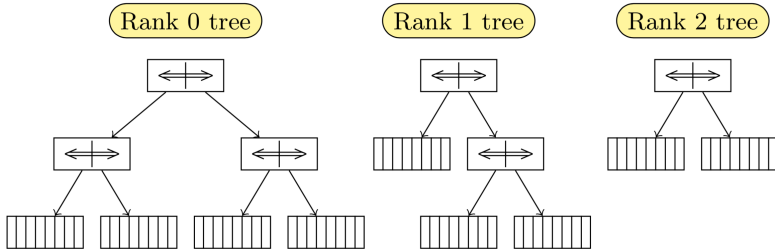
## Divide and Conquer

- 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.



## ENS-NDT

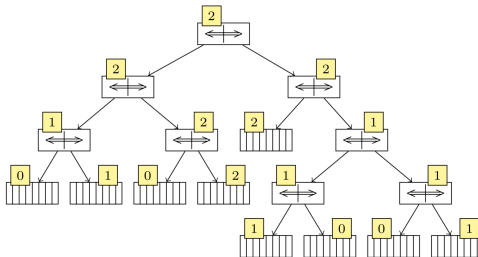
- Для каждого ранга поддерживается отдельное дерево.
- Перебираем точки в лексикографическом порядке:
  - Определяем ранг по текущему набору деревьев.
  - Добавляем точку в соответствующее рангу дерево.





## ENS-NDT-ONE

- Адаптация алгоритма ENS-NDT
  - Структура будет состоять из единственного дерева.
  - В узлах дерева будет содержаться максимальный ранг на поддереве.
    - Оптимизация на этапе определения ранга точки.



## Асимптотика

- Divide and Conquer  $O(N \log^{M-1} N)$ .
- ENS-NDT
  - $O(N^{1.43})$  для случайно сгенерированных точек в гиперкубе.
  - В худшем случае  $O(MN^2)$ .
- ENS-NDT-ONE
  - $O(MN^{1.58})$  для случайно сгенерированных точек в гиперкубе.
  - В худшем случае  $O(MN^2)$ .

## Гибридизация

- Запускаем алгоритм Divide and Conquer, согласно некоторым правилам переключаемся на алгоритм ENS-NDT-ONE.
- Моменты смены алгоритма:
  - HelperA
    - Входные данные: множество точек  $S$  с предварительными рангами.
    - Результат выполнения: множество точек  $S$  с обновленными рангами.
  - HelperB
    - Входные данные: множество точек  $L$  с окончательными рангами и  $R$  с предварительными рангами.
    - Результат выполнения: множество точек  $R$  с обновленными рангами по множеству  $L$ .

## Настройка параметров гибридного алгоритма

- Параметры гибридного алгоритма определяют максимальные размеры множеств точек для каждой размерности, при котором алгоритм Divide and Conquer переключается на алгоритм ENS-NDT-ONE.
- Параметры основаны на экспериментальных исследованиях скорости работы и являются константами.

## Обоснование эффективности гибридного алгоритма

- Экспериментально получено, что для эффективной работы гибридного алгоритма размер точек в момент переключения на алгоритм ENS-NDT-ONE не должен превышать  $10^4 - 2 \cdot 10^4$ .
- Таким образом, худший случай для алгоритма ENS-NDT-ONE с асимптотикой  $O(MN^2)$  незначительно влияет на асимптотику гибридного алгоритма.

## Сравнение скоростей

$N$	$M$	D&C		ENS-NDT		ENS-NDT-ONE		Hybrid	
		cube	plane	cube	plane	cube	plane	cube	plane
$10^6$	3	2.82	1.60	5.25	1.61	4.25	1.65	2.63	1.50
$10^6$	5	45.2	33.0	26.3	5.22	18.2	5.82	17.2	12.8
$10^6$	7	191.5	120.2	55.4	19.4	46.1	18.9	26.8	20.1
$10^6$	10	478.8	228.6	84.8	48.1	104.8	55.0	41.0	33.0
$10^6$	15	587.9	337.5	135.4	76.3	206.8	85.4	64.5	46.0

## Выводы

- Гибридный алгоритм быстрее обоих родительских алгоритмов.
- Нам неизвестны публикации результатов сортировки множеств точек размером  $10^6$  большой размерности с приемлемым временем выполнения.
- Алгоритм адаптирован для многопоточного выполнения, ускорение составляет до 1.8 на двух потоках и до трех раз на восьми потоках.
- По результатам этой работы была подготовлена для публикации статья на конференцию на PPSN 2018, и мы ждем рецензий.

Спасибо за внимание!