

Московский государственный технический университет им. Н.Э. Баумана

Сравнительный анализ методов решения задачи линейного упорядочения

09.06.2025

Научный руководитель: д. т. н., профессор РК6 МГТУ им. Н. Э. Баумана
Божко А. Н.
Студент: Антонов А. С. РК6-41М



Цели работы

Провести сравнительный анализ способов решения задачи линейного упорядочения

Точные методы

?

Эвристики

?

Метаэвристики

Задачи:

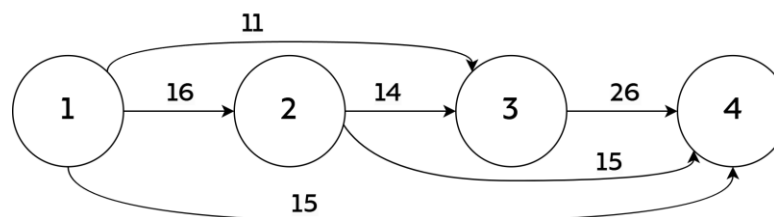
1. Исследовать и систематизировать существующие методы
2. Разработать программный комплекс и реализовать в нём методы
3. Провести анализ на основании результатов экспериментов

Постановка задачи

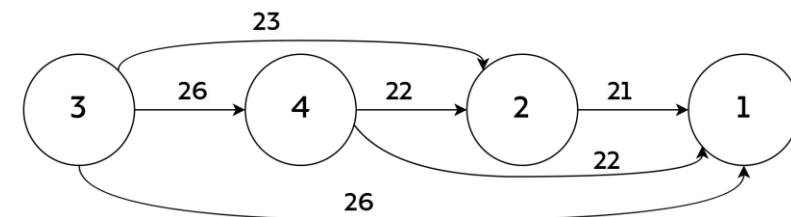
Linear Ordering Problem (LOP)

NP-hard

Графовая форма



Начальное решение (97)



Оптимальное решение (148)

Матричная форма

Initial matrix (Исходная матрица) for 5 nodes. The matrix is symmetric, and the lower triangle is shaded. The values are:

	1	2	3	4	5
1	0	16	11	15	7
2	21	0	14	15	9
3	26	23	0	26	12
4	22	22	11	0	13
5	30	28	25	24	0

Исходная матрица (138)

$$f(\pi) = \sum_{i=1}^n \sum_{j=i+1}^n B_{\pi(i)\pi(j)}$$

Optimal matrix solution (Оптимальное решение) for 5 nodes. The matrix is symmetric, and the lower triangle is shaded. The values are:

	5	3	4	2	1
5	0	25	24	28	30
3	12	0	26	23	26
4	13	11	0	22	22
2	9	14	15	0	21
1	7	11	15	16	0

Оптимальное решение (247)

Приложения

Актуальность LOP в различных областях

- **Социология**
Ранжирование социальных факторов по степени влияния на показатели (уровень образования на доход)
- **Биоинформатика**
Анализ генной экспрессии с целью выявления регуляторных процессов
- **Информационные технологии**
Ранжирование поисковой выдачи, обработка естественного языка
- **Логистика**
Оптимизация маршрутов доставки, где матрица весов отражает время или стоимость перемещения между точками
- **Археология**
определения хронологической последовательности культурных слоёв и артефактов

Жадные методы и алгоритм Беккера

Конструктивные
эвристические методы

Конструктивные

order = [] → [5] → [5, 3] → [5, 3, 4] → [5, 3, 4, 2] → [5, 3, 4, 2, 1]

Жадные

$\max(\Delta f)$

Алгоритм Беккера

$$q_i = \frac{\sum_{k=1}^n c_{ik}}{\sum_{k=1}^n c_{ki}}, i = 1 \dots n$$

	1	2	3	4	5	Sum
1	0	16	11	15	7	49
2	21	0	14	15	9	59
3	26	23	0	26	12	73
4	22	22	11	0	13	68
5	30	28	25	24	0	107

	1	2	3	4	5	Becker
1	0	16	11	15	7	0.5
2	21	0	14	15	9	0.6
3	26	23	0	26	12	1.4
4	22	22	11	0	13	1.6
5	30	28	25	24	0	2.6

Методы локального поиска

Restricted Insert
Neighbourhood

Окрестности

Вставка



Обмен



Реверс



$$\sum_{z=1}^{i-1} (b_{\sigma'_z k} - b_{k \sigma'_z}) < 0 \text{ или } \sum_{z=i+1}^n (b_{\sigma'_z k} - b_{k \sigma'_z}) > 0$$

	1	2	3	4	5
1	0	16	11	15	7
2	21	0	14	15	9
3	26	23	0	26	12
4	22	22	11	0	13
5	30	28	25	24	0

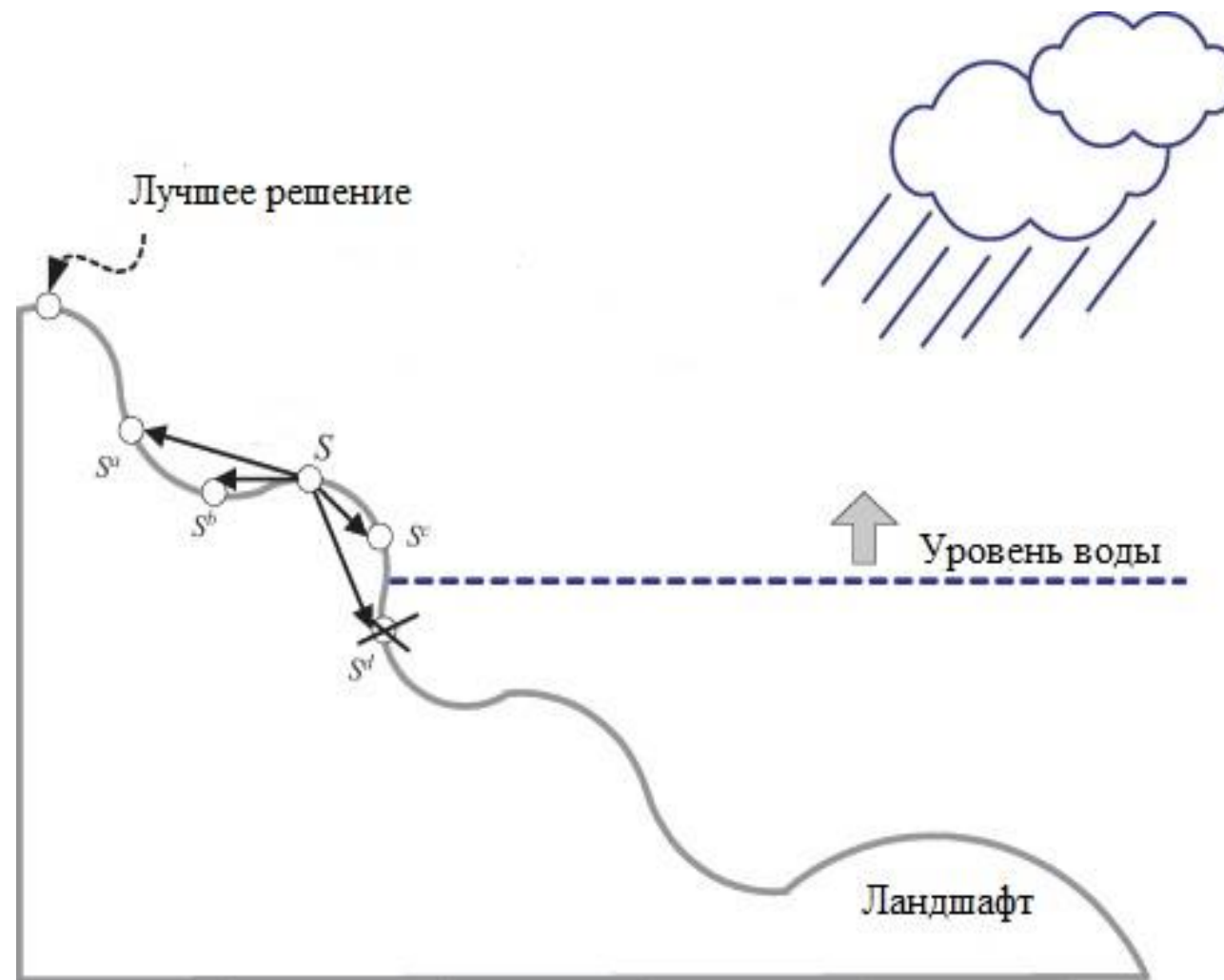
Исходная матрица

	1	2	3	4	5
1	0	0	0	0	1
2	0	0	0	1	1
3	1	1	0	0	0
4	0	1	1	1	1
5	1	0	0	0	0

Матрица ограничений R

Алгоритм великого потопа

Пороговый алгоритм



Параметры:

Начальный уровень
воды

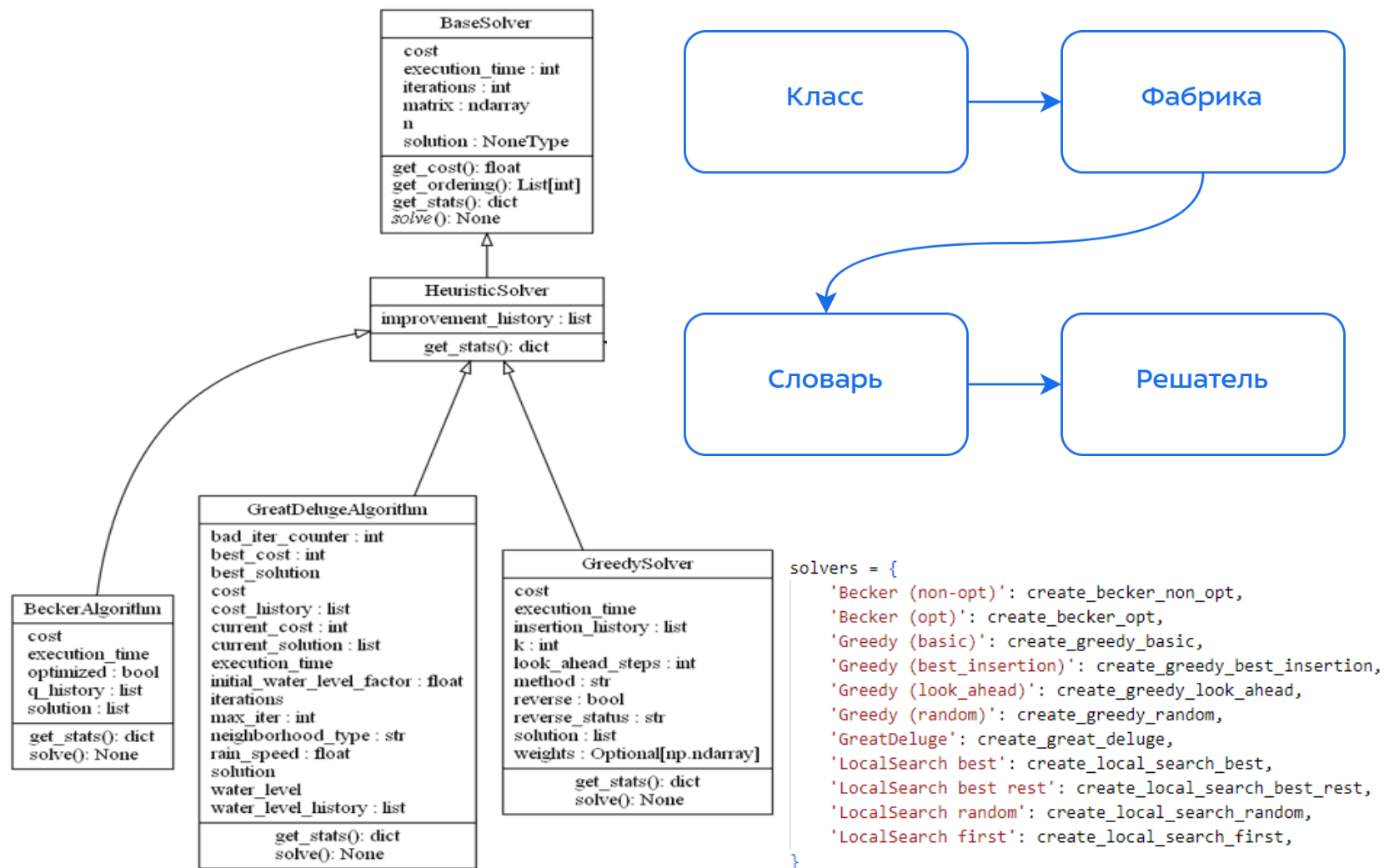
Скорость дождя

Архитектура программного комплекса

Python

Numpy

Pandas



Обзор бенчмарков

Матрицы различных
размерностей и свойств

- **LOLIV** – IO матрицы с экономическими данными, MB матрицы с реальными данными о предпочтениях в голосованиях западногерманского парламента 70-80х годов
- **Случайно сгенерированные матрицы** – матрицы с элементами, равномерно распределенными в заданном диапазоне и матрицы Тайберга

$$c_{ij} = |i - j| * k$$

- **xLOLIV** – матрицы повышенной размерности. Сложные для определённых классов методов

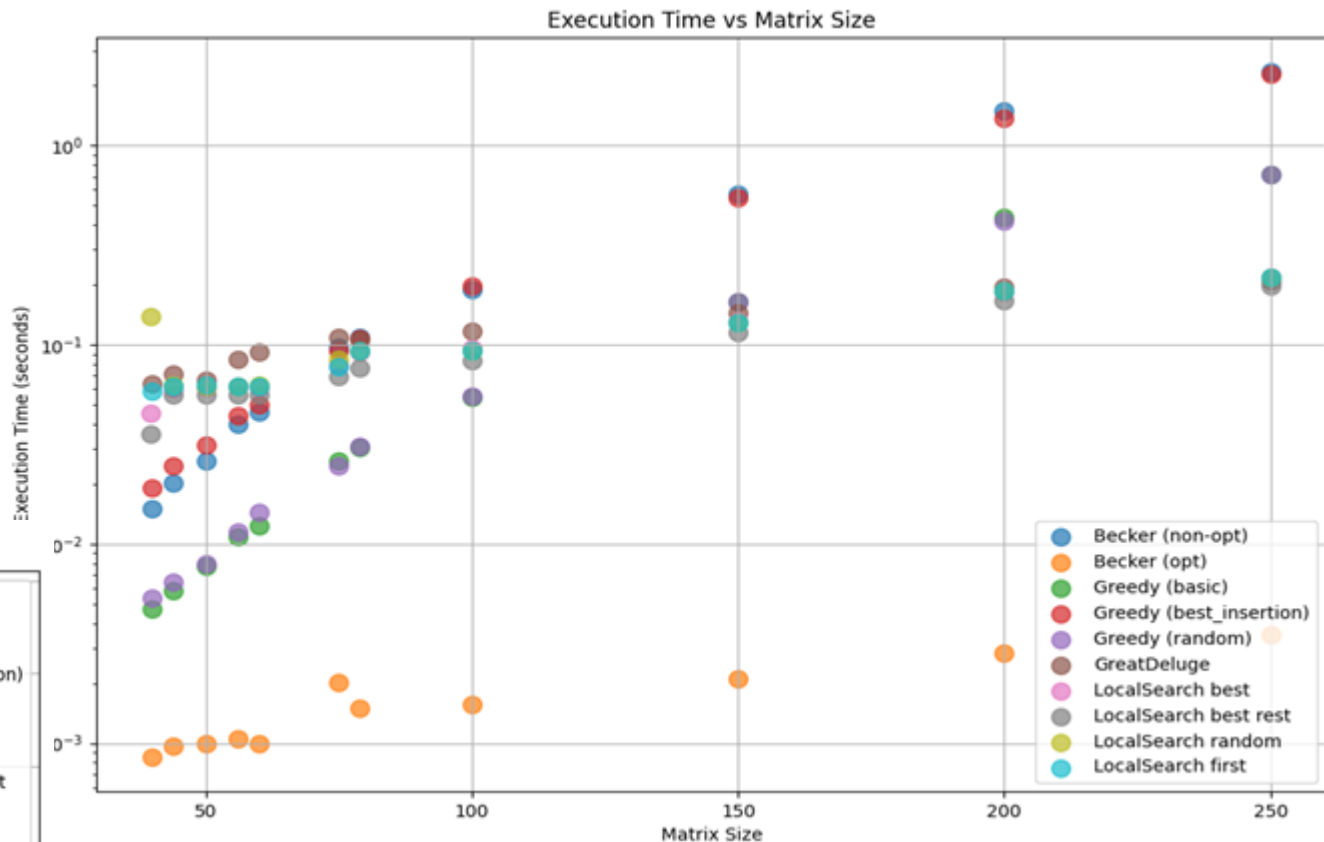
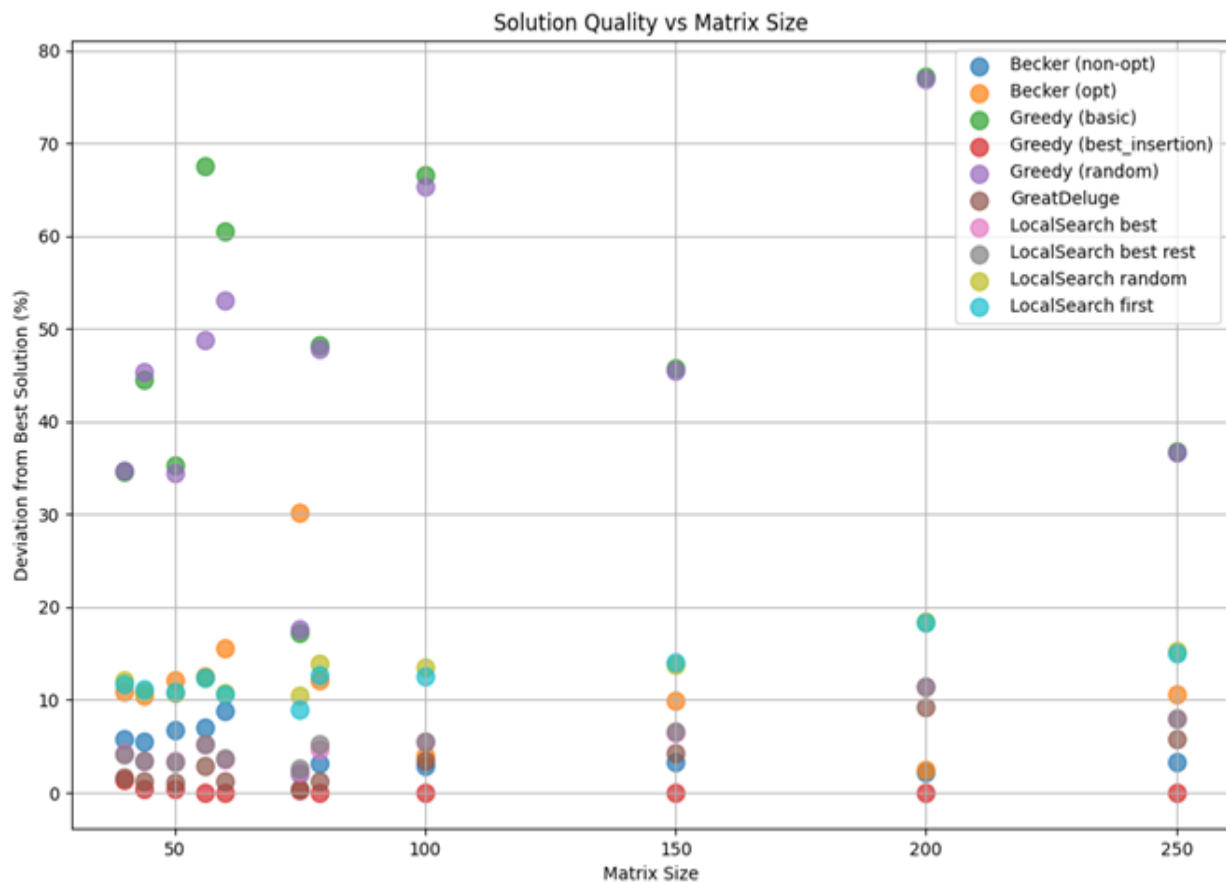
Размерность

от 44 до 60 и 75

от 100 до 500

от 100 до 1000

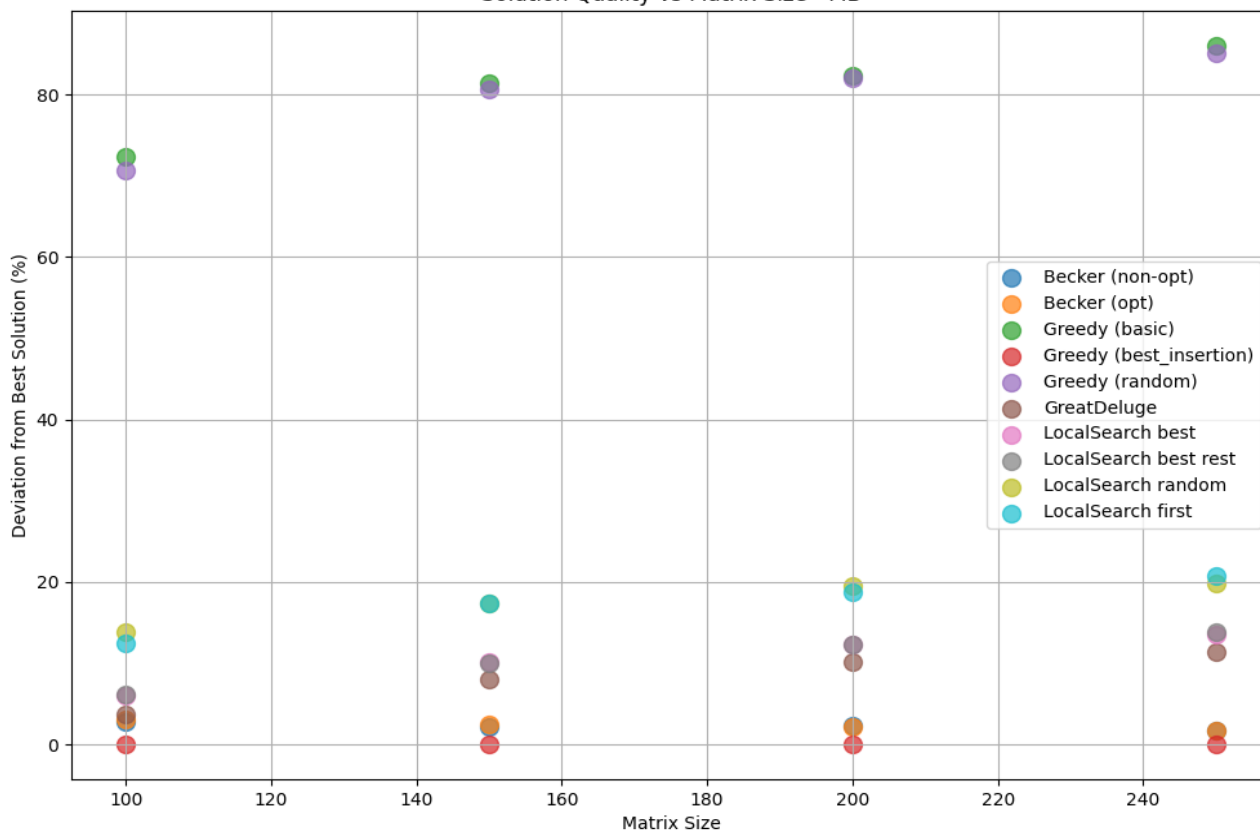
Анализ результатов на обобщённых данных



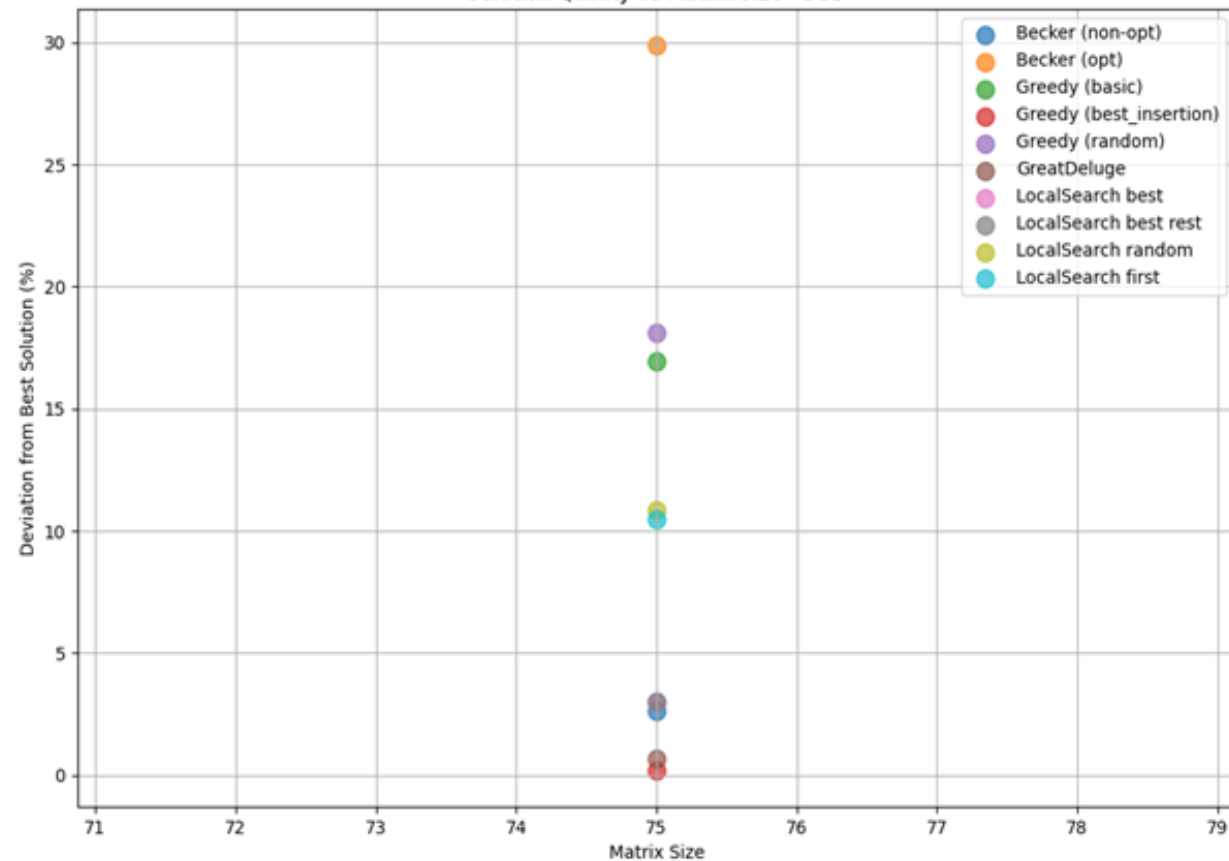
До 100:
Greedy Best Improvement
и Becker
Свыше 100:
LocalSearch и GDA

Анализ результатов в МВ и SGB

Solution Quality vs Matrix Size - MB

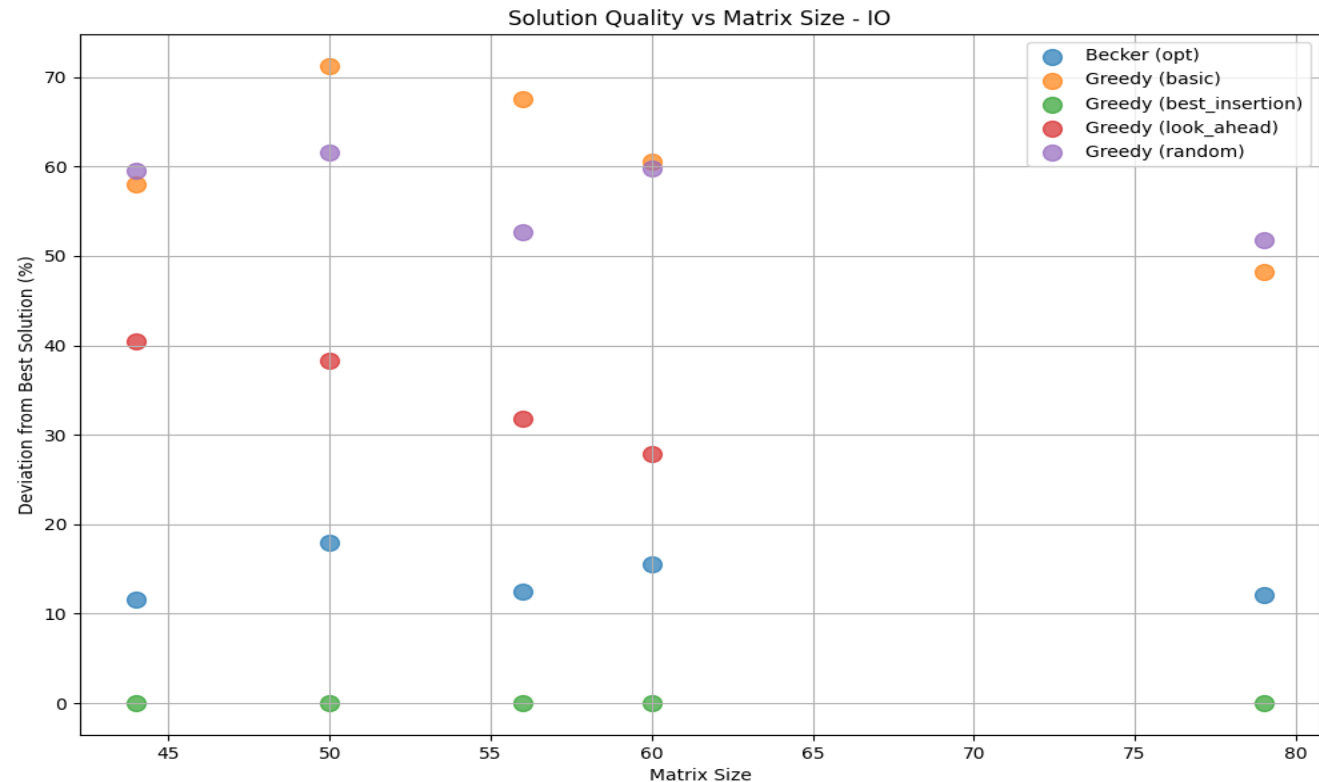
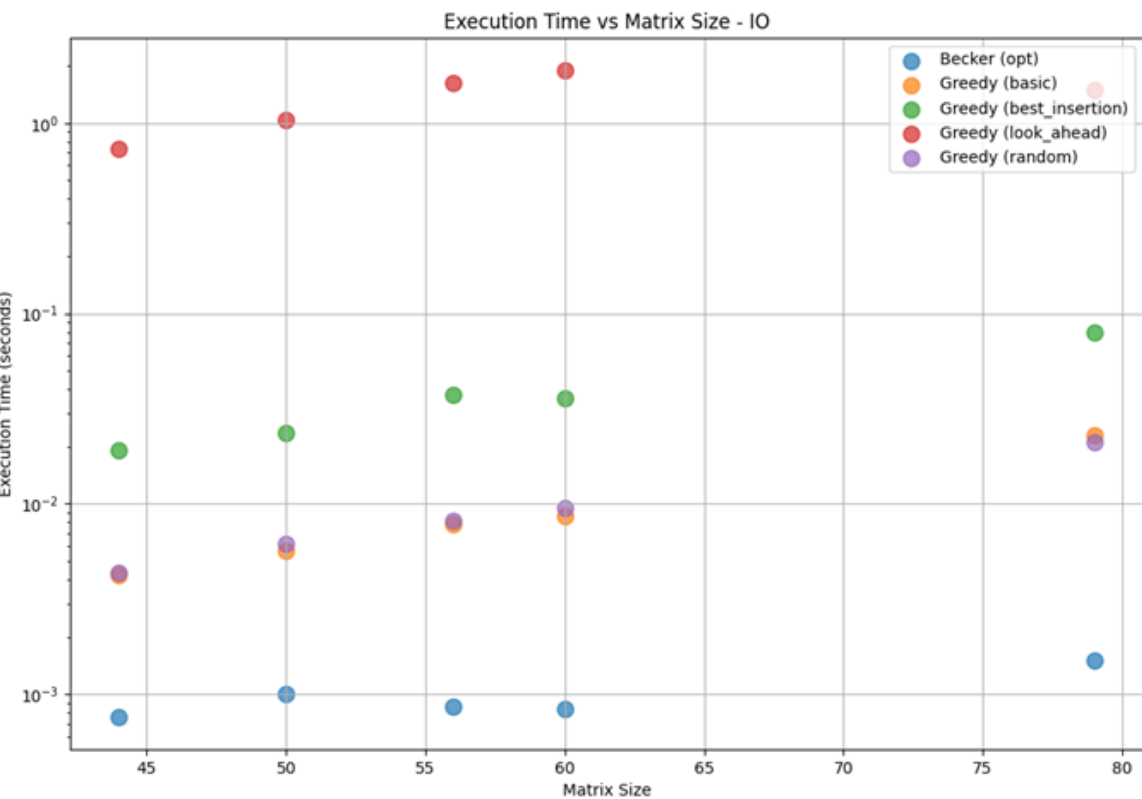


Solution Quality vs Matrix Size - SGB



Becker Opt быстр но
нестабилен

Жадные алгоритмы в датасете IO



- Look Ahead – крайне затратен по времени
- Basic, Random, First – низкая точность

Заключение

1. Проведен сравнительный анализ жадных методов и их модификаций, эвристик Беккера в стандартном и оптимизированном вариантах, локального поиска и метаэвристики великого потока для задачи линейного упорядочения
2. Продемонстрирована способность находить решения, близкие к оптимальному (в пределах 5% от известного оптимума или лучшего найденного), для экземпляров практически любого размера за приемлемое время
3. Лучший результат для задач малых размерностей (<100) показал жадный конструктивный метод наилучшей последовательной вставки
4. Для задач повышенной размерности лидером оказался алгоритм великого потока

Спасибо за внимание!