Санкт-Петербургский государственный университет
Кафедра системного программирования

Экспериментальное исследование алгоритмов для регулярных запросов

Виктория Владимировна Островская, группа 23.Б11-мм

Преподаватель: С.В.Григорьев, доцент кафедры системного программирования

Санкт-Петербург
2025

Постановка задачи

Целью является исследование производительности алгоритмов RPQ при разных типах представления разреженных матриц и размерах стартовых множеств

Задачи:

- Реализовать экспериментальную установку
- Провести серию замеров на тестовых графах
- Определить оптимальные представления матриц для каждого алгоритма
- Оценить зависимость времени выполнения от размера стартового множества

Исследовательские вопросы

- RQ1. При каком представлении разреженных матриц и векторов алгоритм показывают наилучшее время выполнения?
- RQ2. При каком размере стартового множества становится выгоднее по времени выполнить алгоритм для всех пар вершин с последующим выбором нужных вершин, чем непосредственное вычисление достижимости для заданного множества стартовых вершин?

Методология эксперимента

- Алгоритмы
  - tensor_based_rpq
  - ms_bfs_based_rpq
- Типы матриц
  - csr, csc, coo, lil
- Графы
  - travel (131 вершин, 277 рёбер)
  - wine (733 вершин, 1839 рёбер)
  - funding (778 вершин, 1086 рёбер)
- Оборудование
  - MacBook Air M1, 8 ядер, 8 GB RAM, macOS Sequoia 15.5

Выбор графов

- Небольшой размер графов $\rightarrow$ 25 запусков за разумное время (11 ч)
- travel — самый маленький, для отладки
- wine и funding — одинаковое число вершин, разная плотность рёбер $\rightarrow$ проверка на плотных и разреженных данных
- как минимум 4 метки, соответствуют выбранным регулярным выражениям

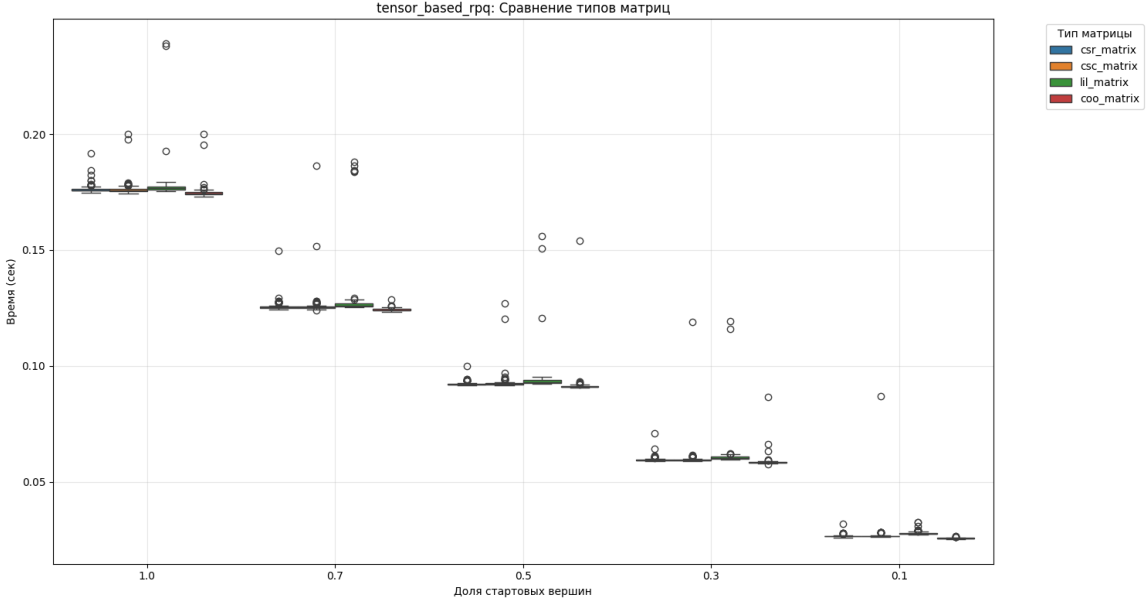
План эксперимента

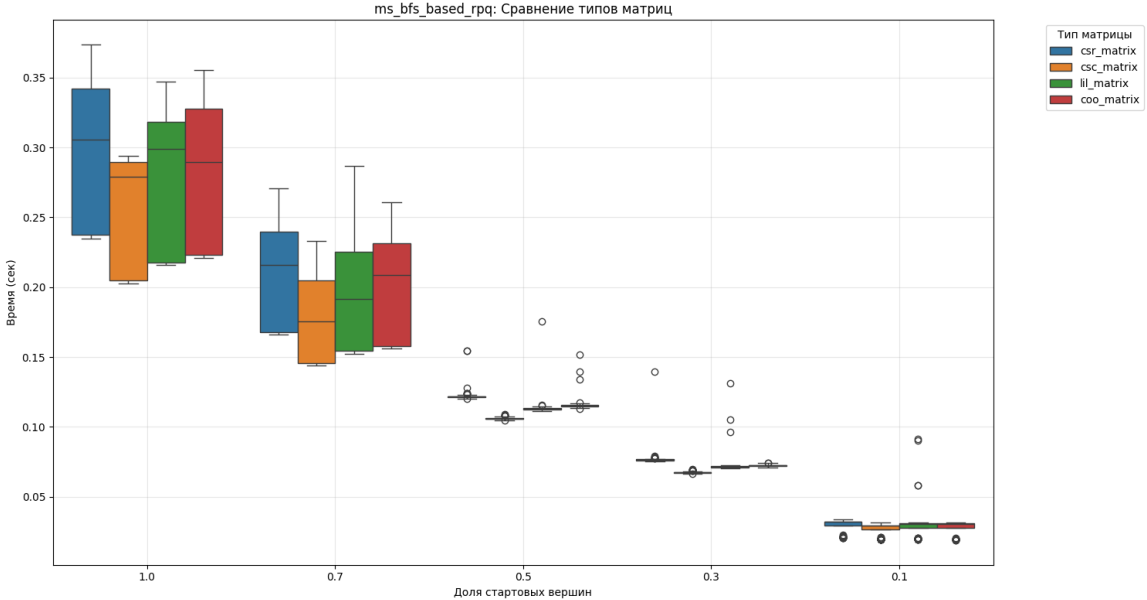
- Регулярные выражения:
  - (a | b)* c d+
  - a (b | c)* d+
  - ((a | b)+ c)* d
  - ((c+) | a) b d*
- Конфигурация запуска:
  - 5 долей стартового множества (1.0, 0.7, 0.5, 0.3, 0.1)
  - Тип матрицы
  - Регулярное выражение
  - Алгоритм
  - 25 запусков для каждой конфигурации

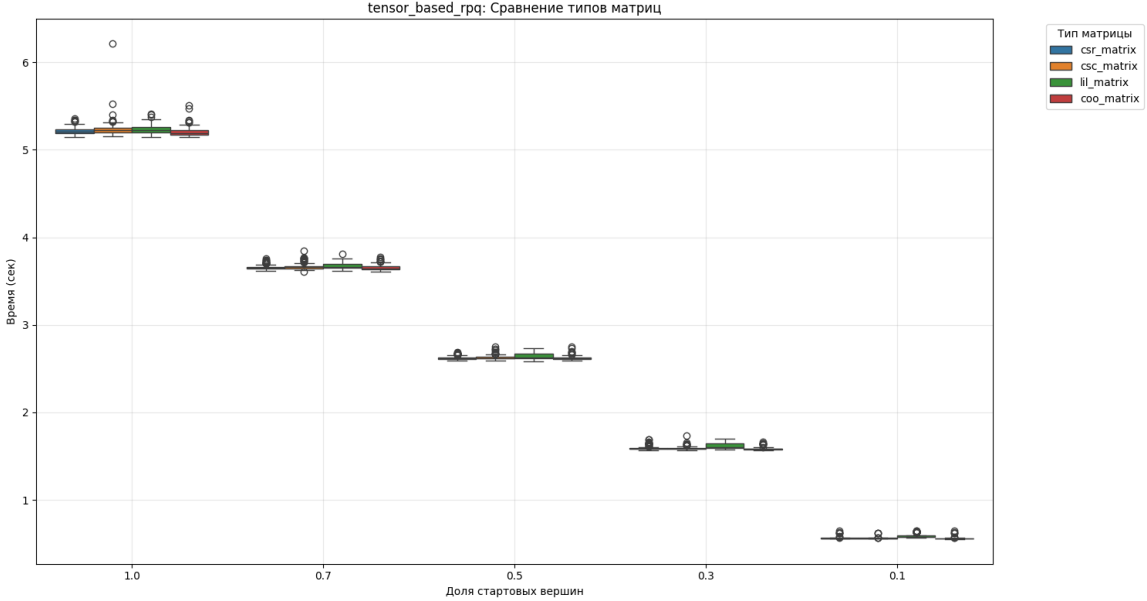
Результаты по RQ1

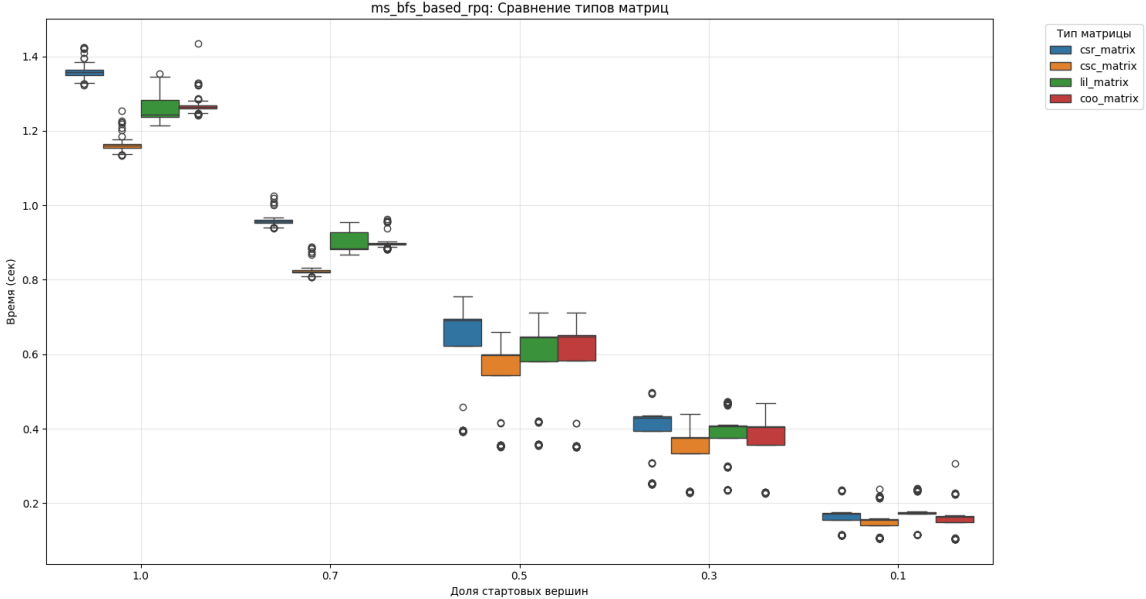
- tensor_based_rpq
  - В 87% сценариев лучшая производительность у coo_matrix
  - Максимальный выигрыш 7.84%
- ms_bfs_based_rpq
  - Всегда лучшая производительность у csc_matrix
  - Выигрыш 9–14%

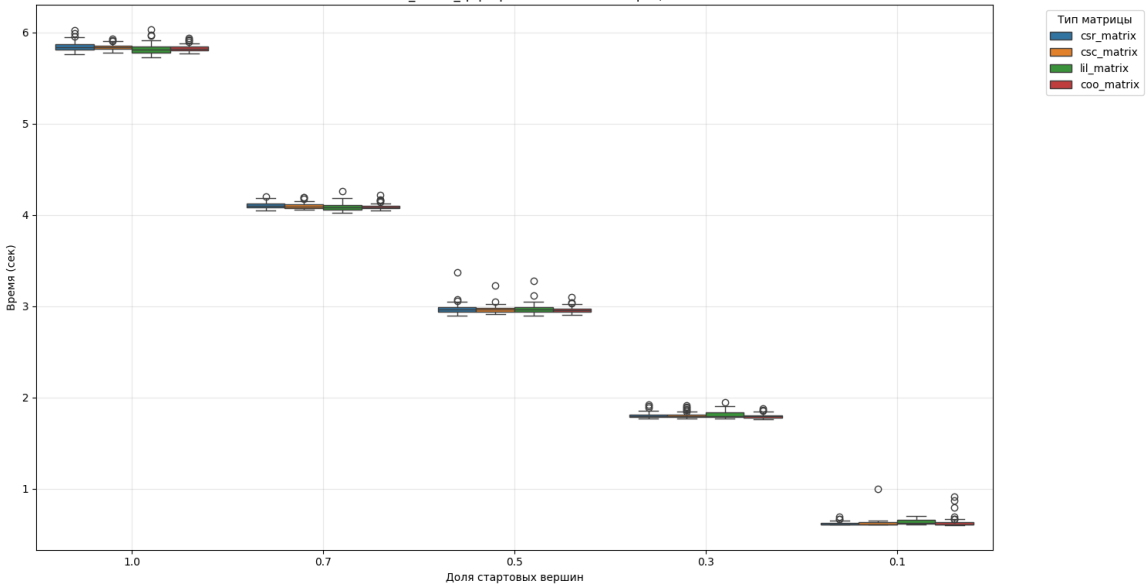
Далее будут представлены графики для графов travel, wine, funding соответственно (по 2 графика на каждый: для алгоритма tensor_based и ms_bfs_based)

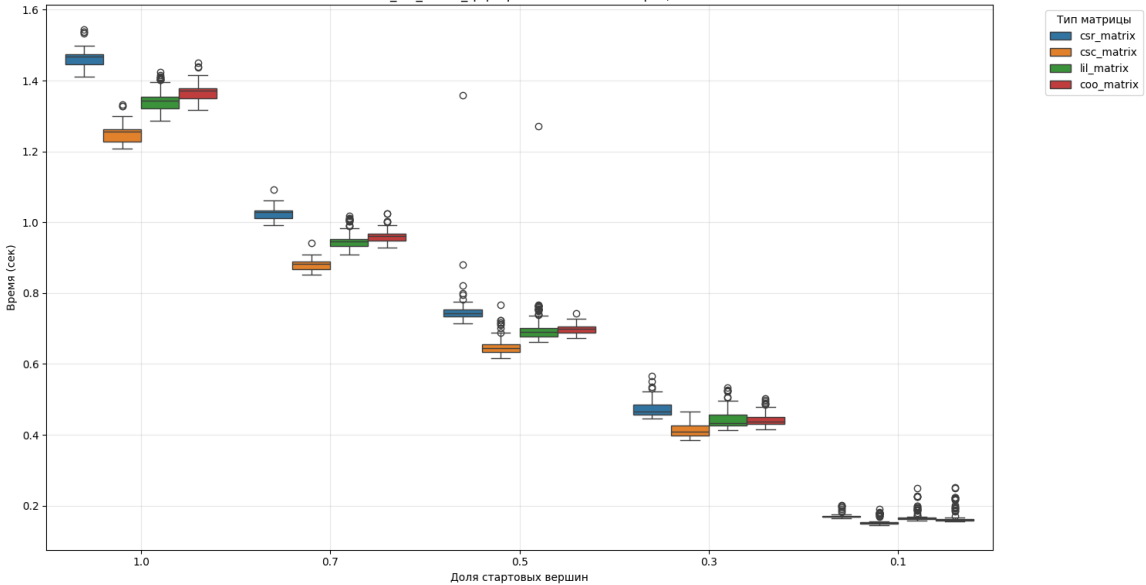

tensor_based_rpq: Сравнение типов матриц
Тип матрицы
csr_matrix
csc_matrix
lil_matrix
coo_matrix
Время (сек)
0.20
0.15
0.10
0.05
1.0
0.7
0.5
0.3
0.1
Доля стартовых вершин


ms_bfs_based_rpq: Сравнение типов матриц
Тип матрицы
csr_matrix
csc_matrix
lil_matrix
coo_matrix
Время (сек)
0.35
0.30
0.25
0.20
0.15
0.10
0.05
1.0
0.7
0.5
0.3
0.1
Доля стартовых вершин


tensor_based_rpq: Сравнение типов матриц
Тип матрицы
csr_matrix
csc_matrix
lil_matrix
coo_matrix
Время (сек)
1
2
3
4
5
6
1.0
0.7
0.5
0.3
0.1
Доля стартовых вершин


ms_bfs_based_rpq: Сравнение типов матриц
Тип матрицы
csr_matrix
csc_matrix
lil_matrix
coo_matrix
Время (сек)
1.4
1.2
1.0
0.8
0.6
0.4
0.2
1.0
0.7
0.5
0.3
0.1
Доля стартовых вершин

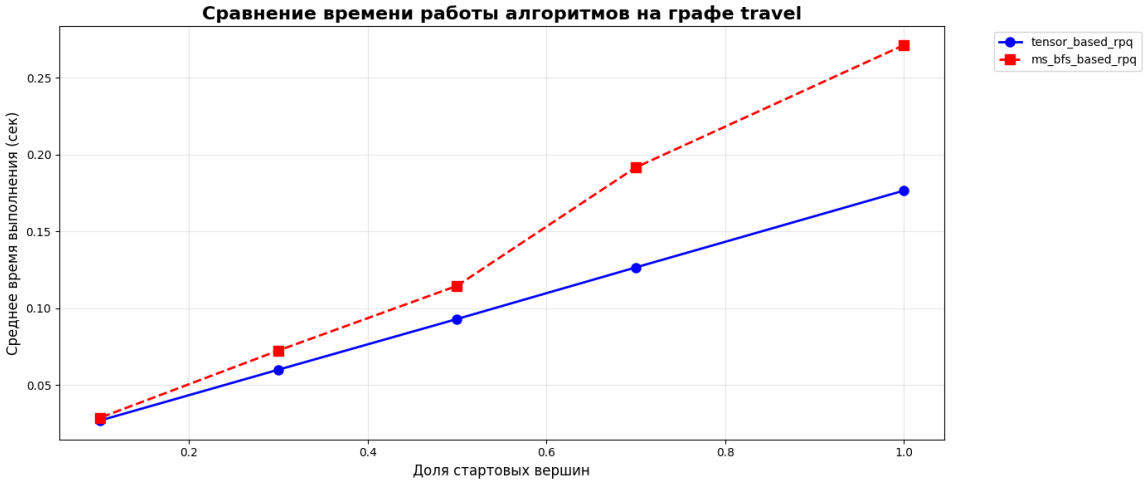

tensor_based_rpq: Сравнение типов матриц
Тип матрицы
csr_matrix
csc_matrix
lil_matrix
coo_matrix
Время (сек)
1
2
3
4
5
6
1.0
0.7
0.5
0.3
0.1
Доля стартовых вершин

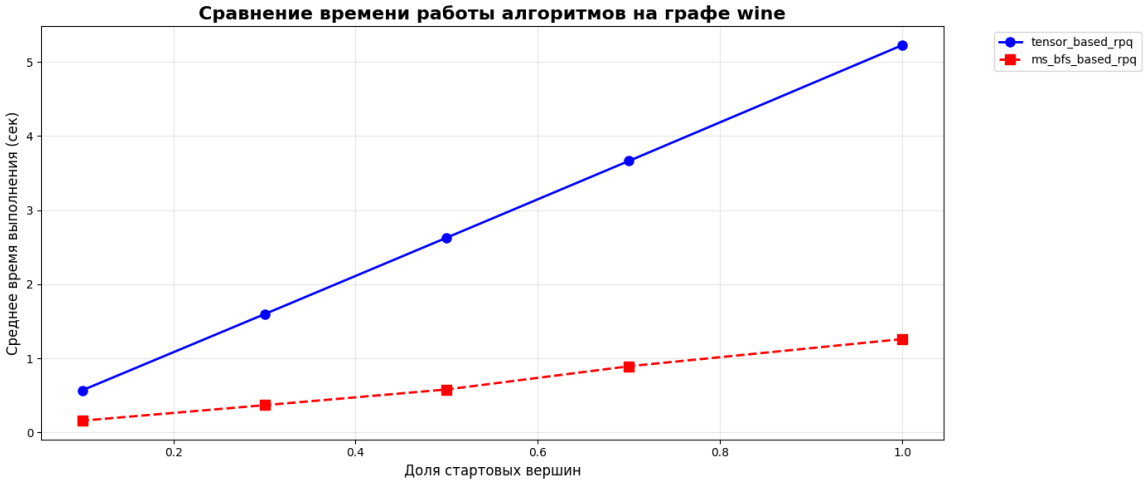

ms_bfs_based_rpq: Сравнение типов матриц
Тип матрицы
csr_matrix
csc_matrix
lil_matrix
coo_matrix
Время (сек)
1.6
1.4
1.2
1.0
0.8
0.6
0.4
0.2
1.0
0.7
0.5
0.3
0.1
Доля стартовых вершин

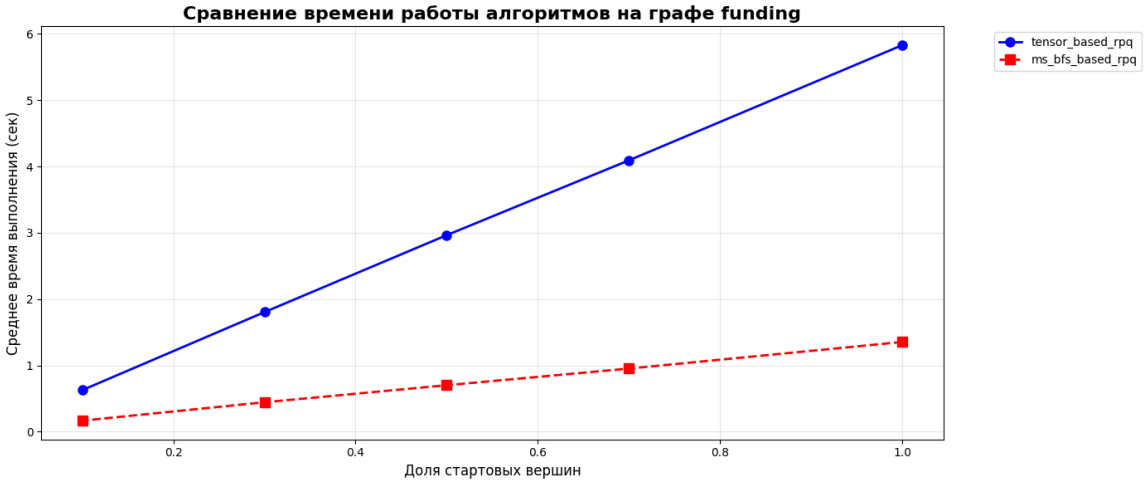
Результаты по RQ2

- Время работы растёт с увеличением доли стартовых вершин
- Алгоритмы не выигрывают при переходе на вычисление для всех пар вершин
- Прямая стратегия для подмножеств всегда быстрее

Далее будут представлены графики для графов travel, wine, funding


Сравнение времени работы алгоритмов на графе travel
tensor_based_rpq
ms_bfs_based_rpq
Среднее время выполнения (сек)
0.05
0.10
0.15
0.20
0.25
0.2
0.4
0.6
0.8
1.0
Доля стартовых вершин


Сравнение времени работы алгоритмов на графе wine
Среднее время выполнения (сек)
Доля стартовых вершин
tensor_based_rpq
ms_bfs_based_rpq
0
1
2
3
4
5
0.2
0.4
0.6
0.8
1.0


Сравнение времени работы алгоритмов на графе funding
Среднее время выполнения (сек)
Доля стартовых вершин
tensor_based_rpq
ms_bfs_based_rpq
0
1
2
3
4
5
6
0.2
0.4
0.6
0.8
1.0

Угрозы валидности

- Эксперименты запускались только на одной машине
- Реализация для каждого типа матриц потенциально может быть эффективнее, нужны дополнительные эксперименты

Ссылка на pull request: https://github.com/vicitori/formal-lang-course/pull/6