2. Команда 1

Участники: Рожков Александр, Перевалов Ефим, Нурмухаметов Рафик

BFS Single-source Parent

BFS Single-source Parent - это алгоритм поиска в ширину, который обходит граф от заданной начальной вершины, сохраняя информацию о "родителях" каждой вершины

BFS Multiple-source Parent

Multiple-source Parent BFS - это алгоритм обхода графа, который начинается с нескольких исходных вершин одновременно, а не с одной исходной вершины, как в Single-source Parent BFS

Эксперимент

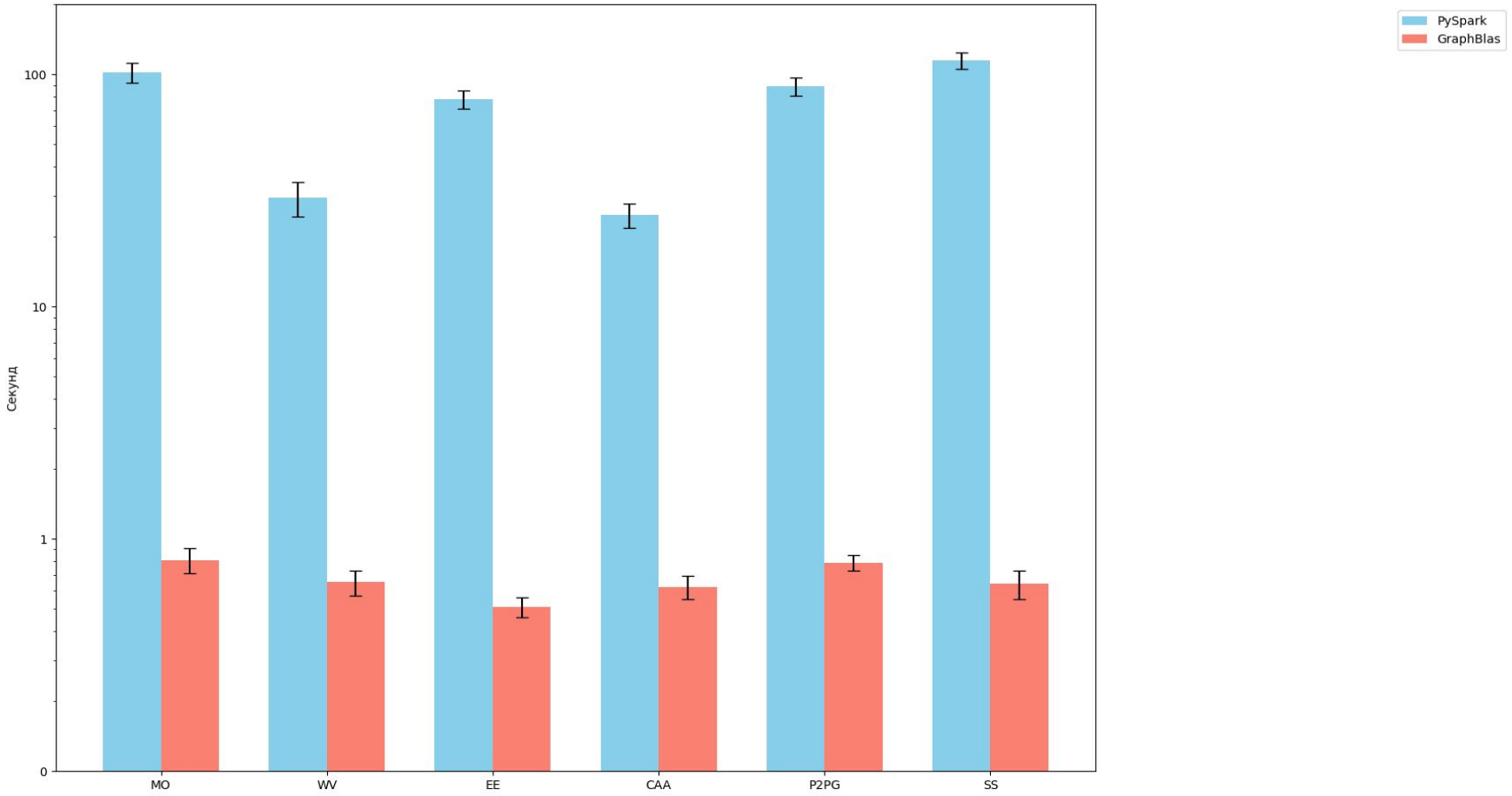
На какой из 2 библиотек (PySpark, GraphBLAS) быстрее будет реализация для алгоритмов BFS Single-source Parent, BFS Multiple-source Parent. Для MS BFS количество стартовые вершины: 2, 4, 8, 16, 32. На каждый вариант по 30 запусков

Граф	Вершин	Ребер
Math_overflow	55 863	858 490
wiki-Voute	7 115	103 689
Email-Enron	36 692	183 831
CA-AstroPh	18 772	396 160
p2p-Gnutella31	62 586	147 892
soc-sign-Slashdot090216	81 871	545 671

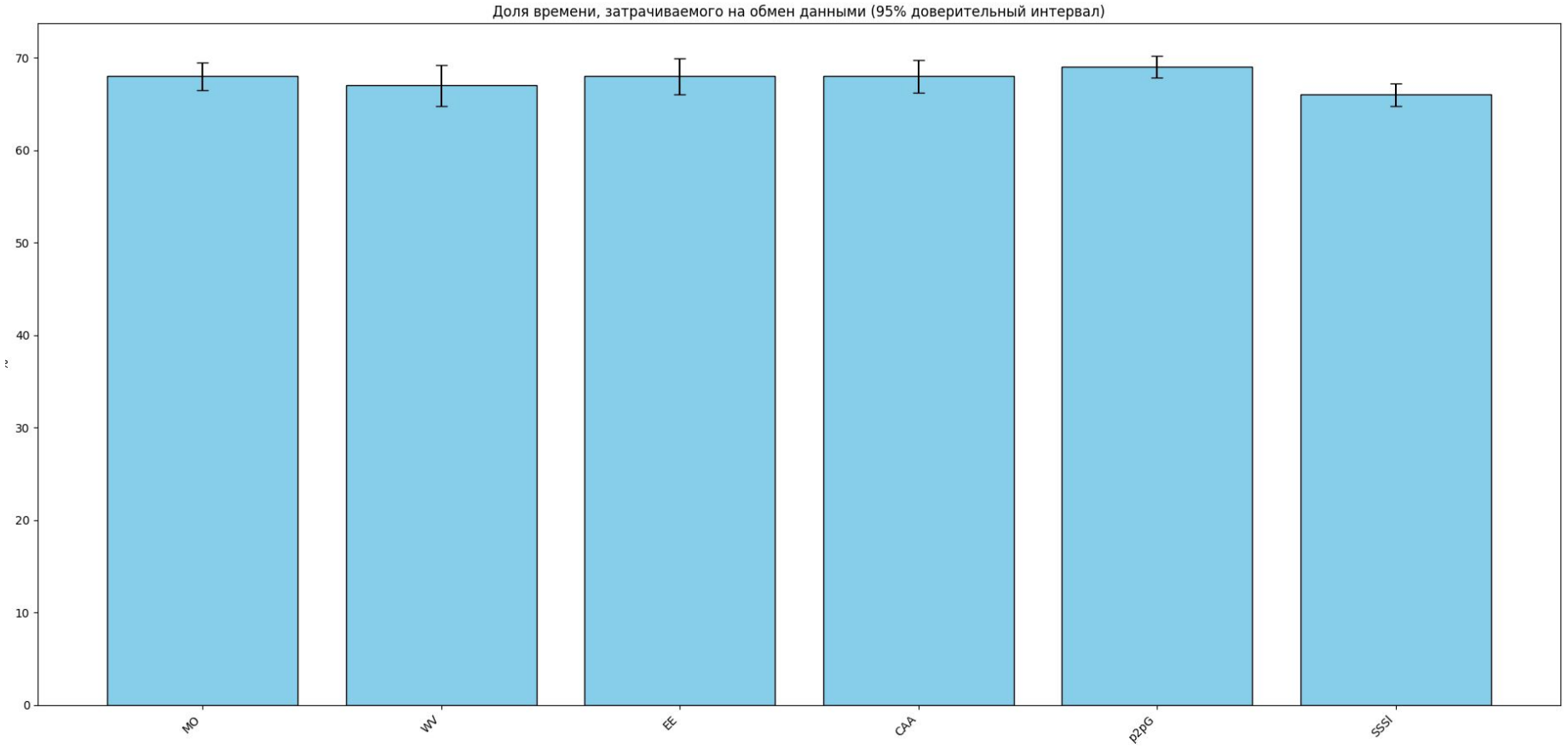
Характеристика машины

- ОС Ubuntu 20.04 LTS
- Intel Core i5-10210U
  - 1.6 GHz
  - 4 ядра
- 16 GB RAM

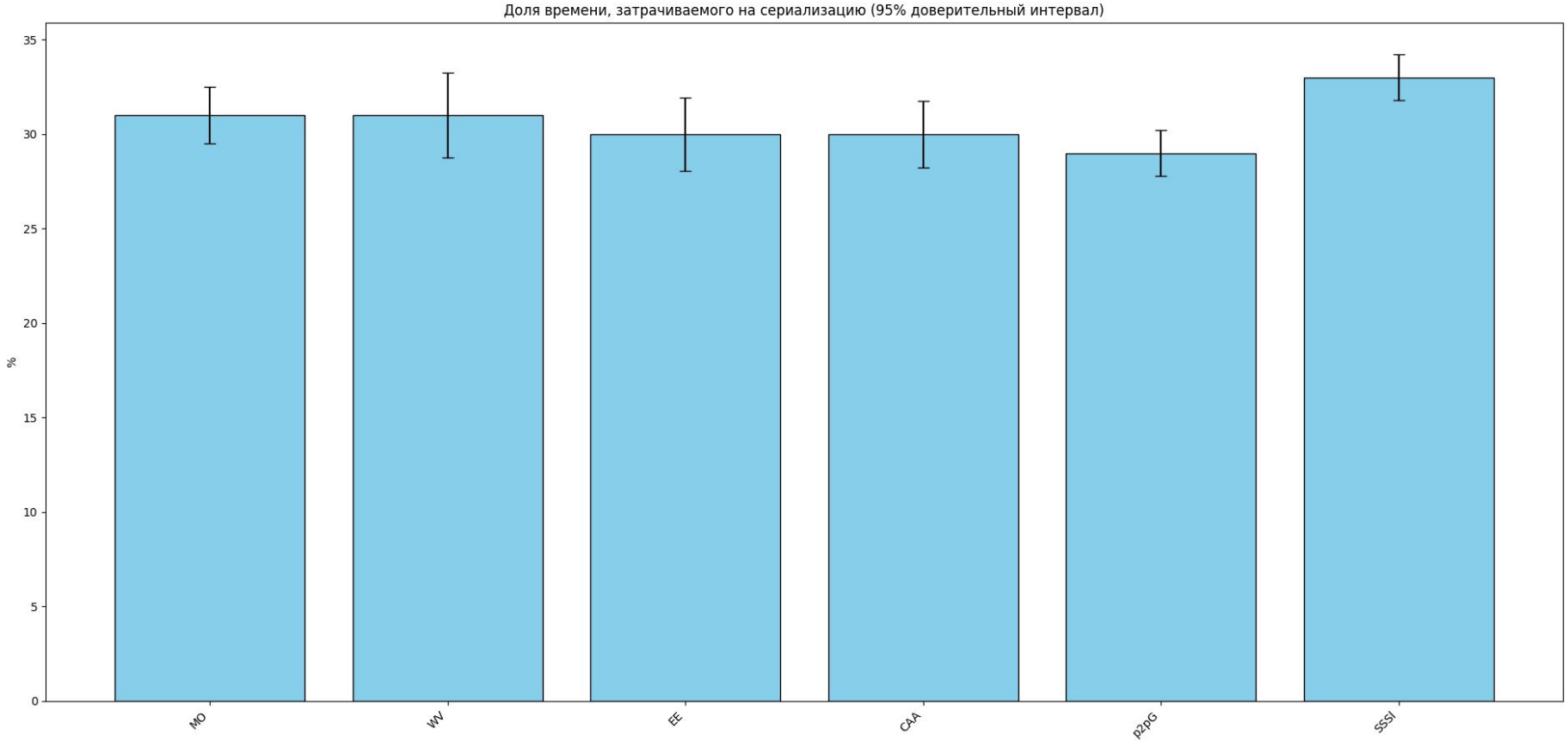
SS-BFS. Сравнение результатов эксперимента



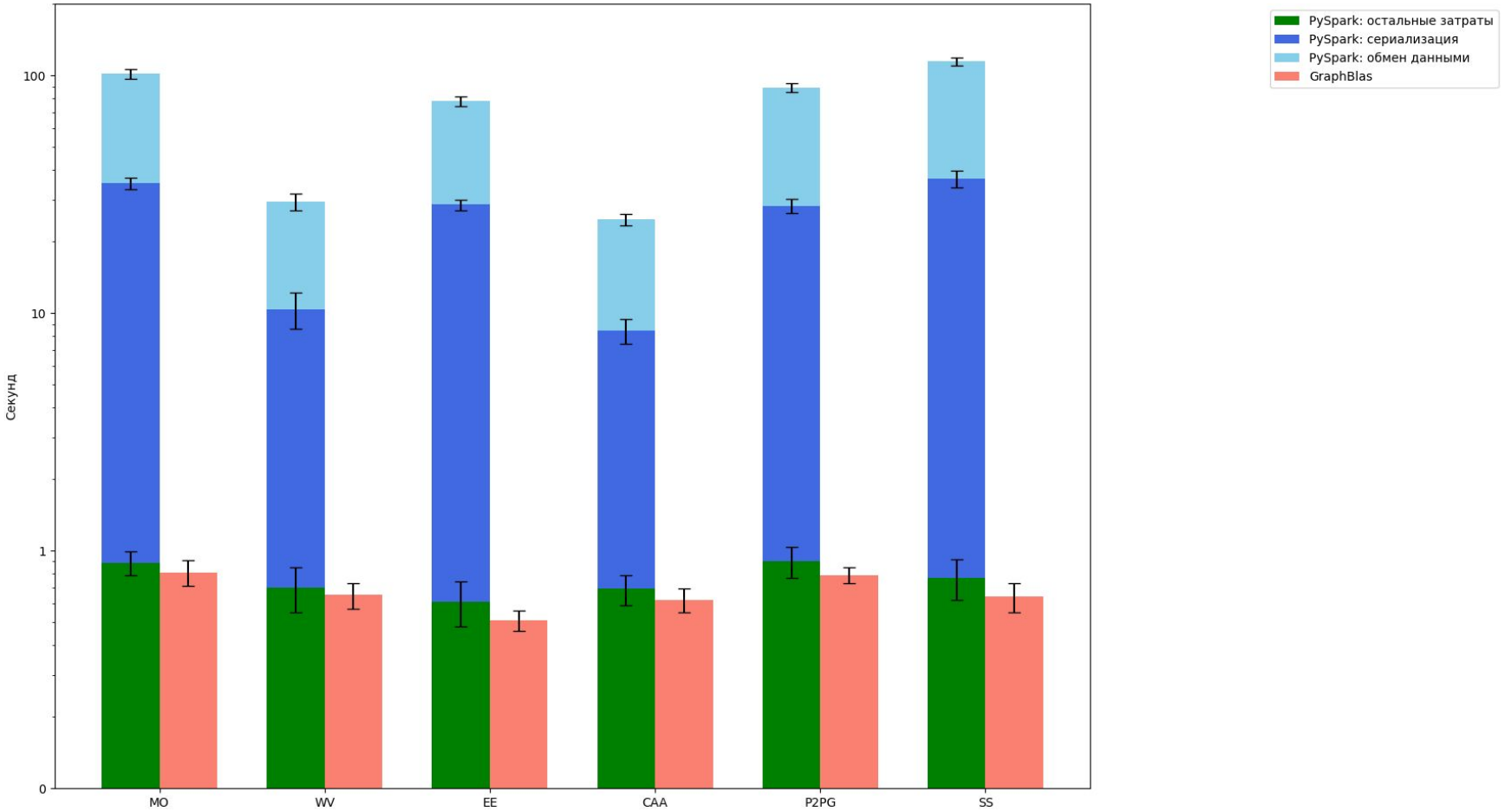
SS-BFS. Результаты профилирования PySpark 1



SS-BFS. Результаты профилирования PySpark 2



SS-BFS. Анализ полученных результатов 1



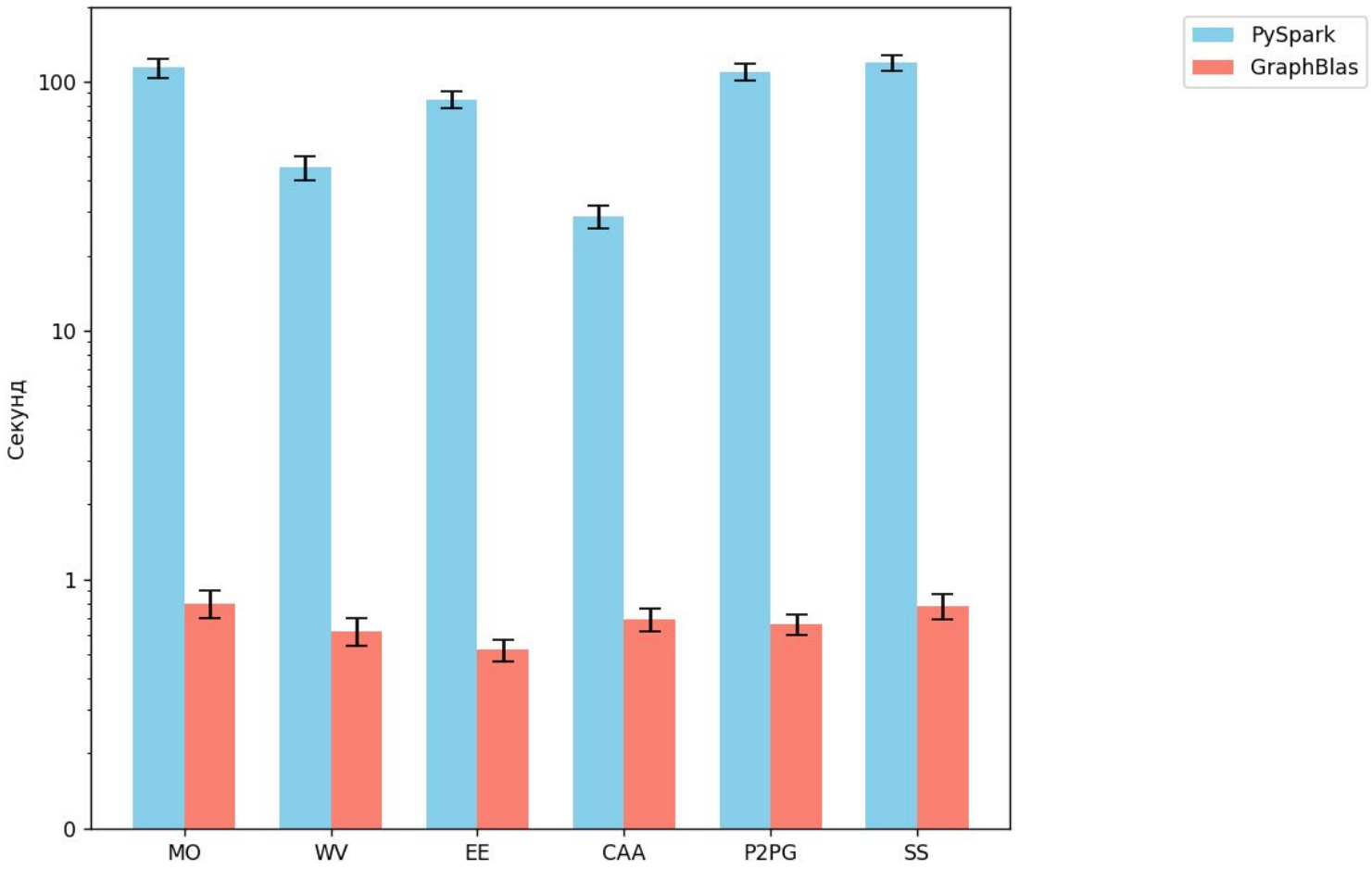
SS-BFS. Анализ полученных результатов 2

Pyspark проигрывает реализации на GraphBlas в среднем в 100 раз

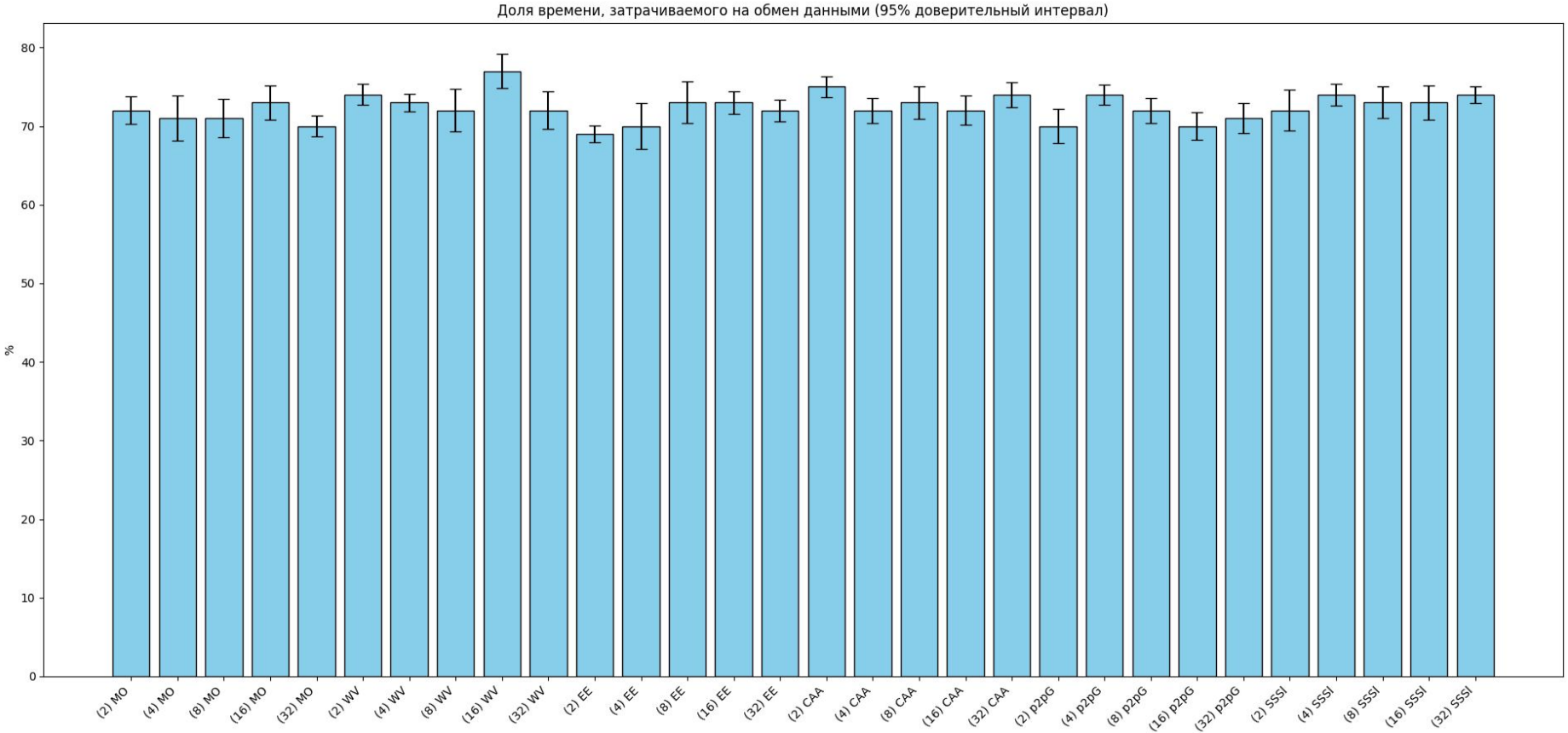
Причины:

- PySpark работает через несколько потоков с затратной передачей данных (обмен данными занимает около 68% рабочего времени)
- Сериализация при проверке новых непосещенных вершин требует пересылки данных между JVM и Python (занимает около 31% рабочего времени)

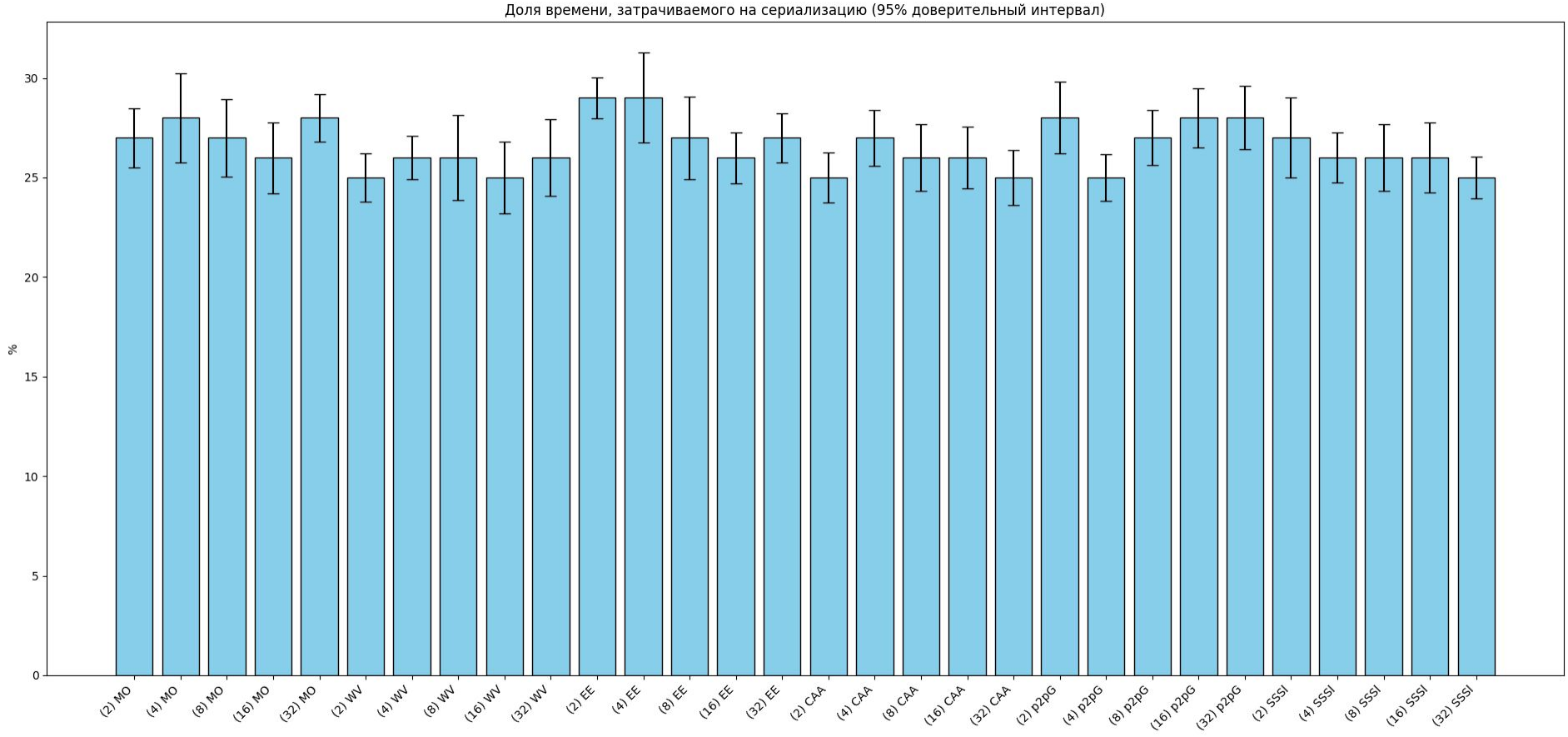
MS-BFS. Сравнение результатов эксперимента



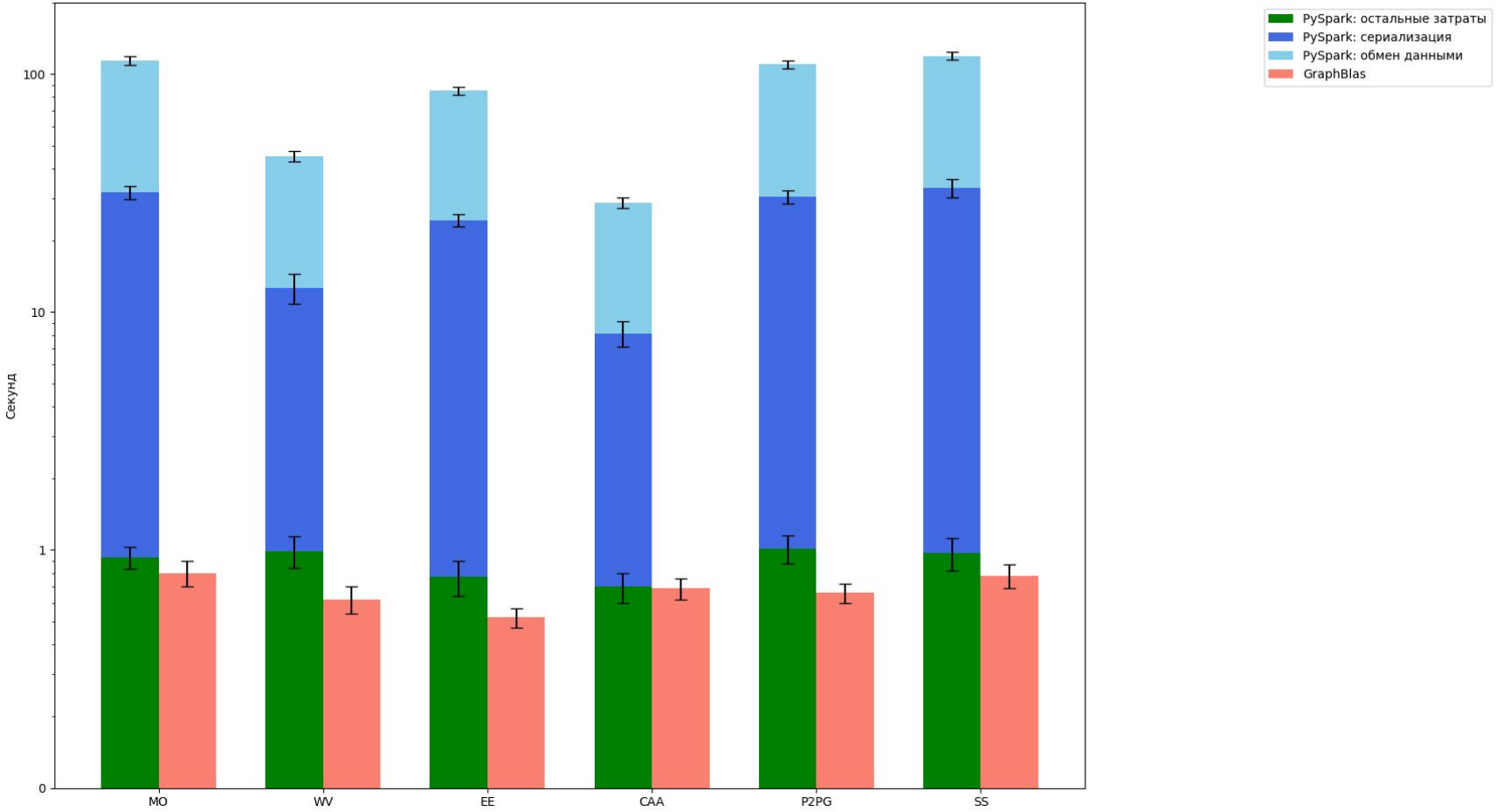
MS-BFS. Результаты профилирования PySpark 1



MS-BFS. Результаты профилирования PySpark 2



MS-BFS. Анализ полученных результатов 1



MS-BFS. Анализ полученных результатов 2

Pyspark проигрывает реализации на GraphBlas в среднем в 100 раз

Причины:

- PySpark работает через несколько потоков с затратной передачей данных (обмен данными занимает около 70% рабочего времени)
- Сериализация при проверке новых непосещенных вершин требует пересылки данных между JVM и Python (занимает около 28% рабочего времени)
- GraphBlas оперирует матричным умножением

Треугольники

Задача: найти количество уникальных треугольников в графе.

Алгоритмы:

- Sandia $\sum_j \sum_i (U^2 * U)$
- Burkhardt $\frac{1}{6} \sum_j \sum_i (A^2 * A)$

SuiteSparse:GraphBLAS - полная реализация стандарта GraphBLAS, использующаяся для работы с графовыми алгоритмами на языке линейной алгебры.

SPLA - открытая библиотека для работы с разреженными матрицами, предоставляющая поддержку GPU.

Эксперимент

Цель: Сравнить реализации на SuiteSparse:GraphBLAS(CPU) и SPLA(CPU/GPU).

Ход эксперимента:

- 10 запусков для каждого графа
- Вычисление среднего значения
- Анализ полученных результатов

Гипотезы:

- SuiteSparse:GraphBLAS быстрее SPLA на CPU
- SPLA с GPU превосходит SuiteSparse:GraphBLAS

Вычислительная машина и по

Вычислительная машина:

- Процессор: 11th Gen Intel(R) Core(TM) i3-1115G4
  - Количество ядер: 2
  - Количество логических ядер: 4
- RAM: 8 GB
- GPU: Mesa Intel® UHD Graphics (TGL GT2)

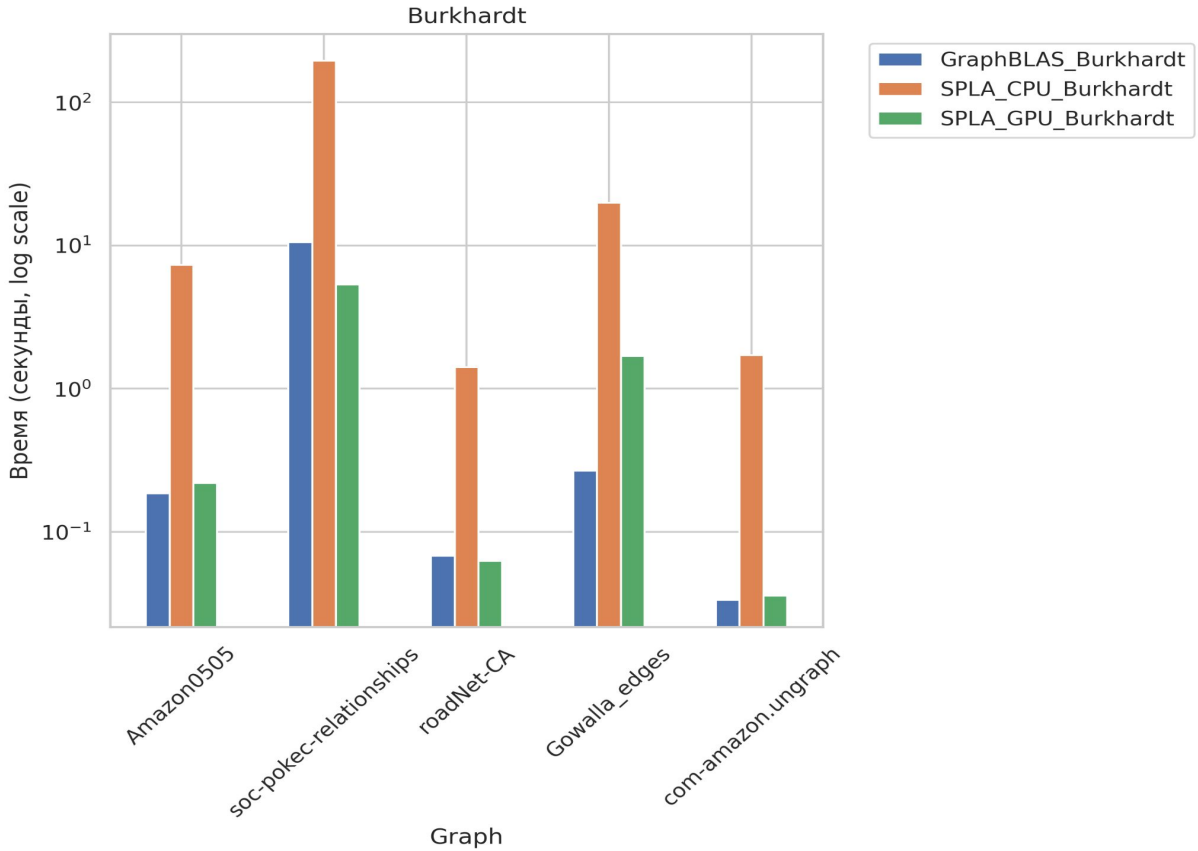
ПО:

- ОС: Ubuntu 22.04.5 LTS
- gcc 11.4.0
- intel openCL 22.14.22890

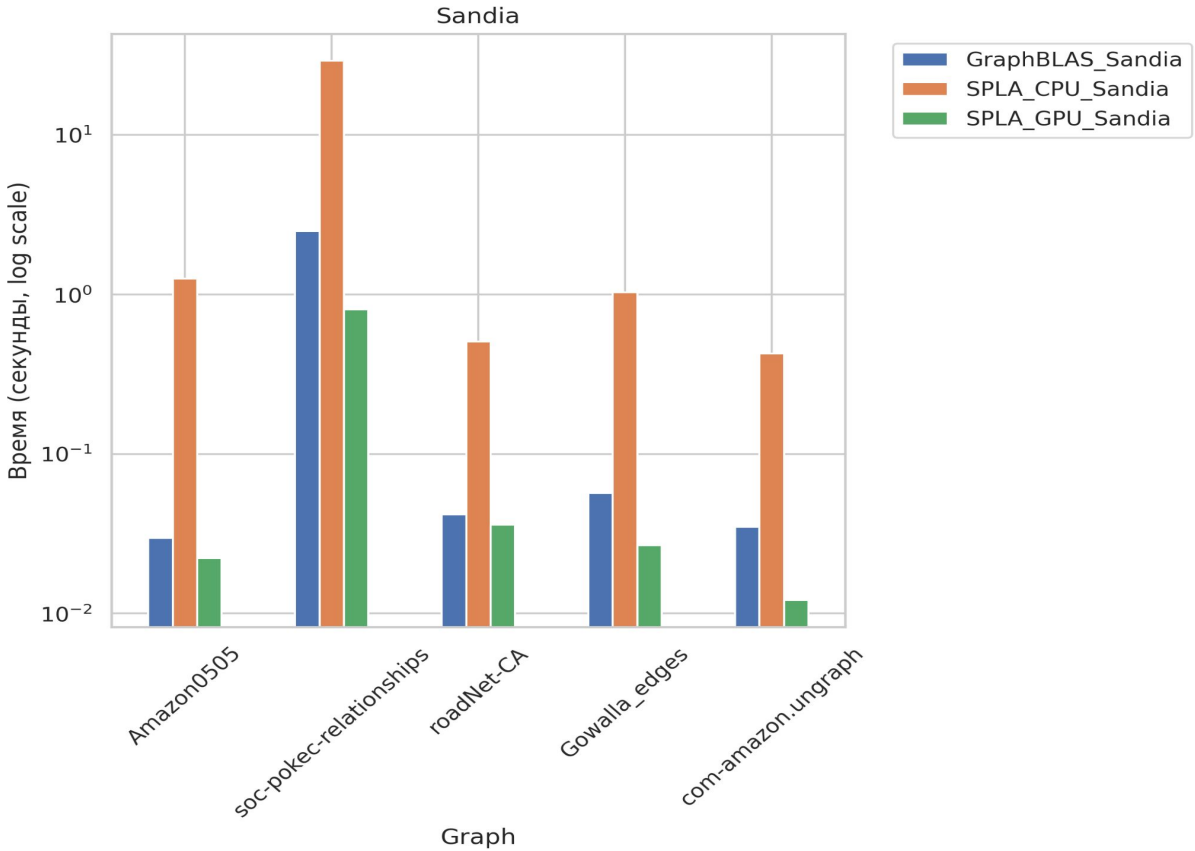
Набор данных

Название	Вершин	Ребер
roadNet-CA	1 965 206	2 776 607
loc-Gowalla	196 591	950 327
amazon0505	410 236	3 356 824
com-Amazon	334 863	925 872
soc-Pokec	1 632 803	30 622 463

Burkhardt


Burkhardt
GraphBLAS_Burkhardt
SPLA_CPU_Burkhardt
SPLA_GPU_Burkhardt
Время (секунды, log scale)
10^2
10^1
10^0
10^{-1}
Amazon0505
soc-pokec-relationships
roadNet-CA
Gowalla_edges
com-amazon.ungraph
Graph

Sandia


Sandia
GraphBLAS_Sandia
SPLA_CPU_Sandia
SPLA_GPU_Sandia
Время (секунды, log scale)
10^{1}
10^{0}
10^{-1}
10^{-2}
Amazon0505
soc-pokec-relationships
roadNet-CA
Gowalla_edges
com-amazon.ungraph
Graph

Анализ полученных результатов

- SuiteSparse:GraphBLAS лучше заточен для вычислений на CPU
- SPLA с GPU превосходит SuiteSparse:GraphBLAS
- С увеличением количества вычислений SPLA требует больше ресурсов