

Лабораторная работа 3. Задание 1

Используя `std::thread`, `std::jthread` или асинхронные задания, реализовать

многопоточную версию программы умножения матрицы на вектор с

параллельной инициализацией массивов

Ускорение в зависимости от количества потоков

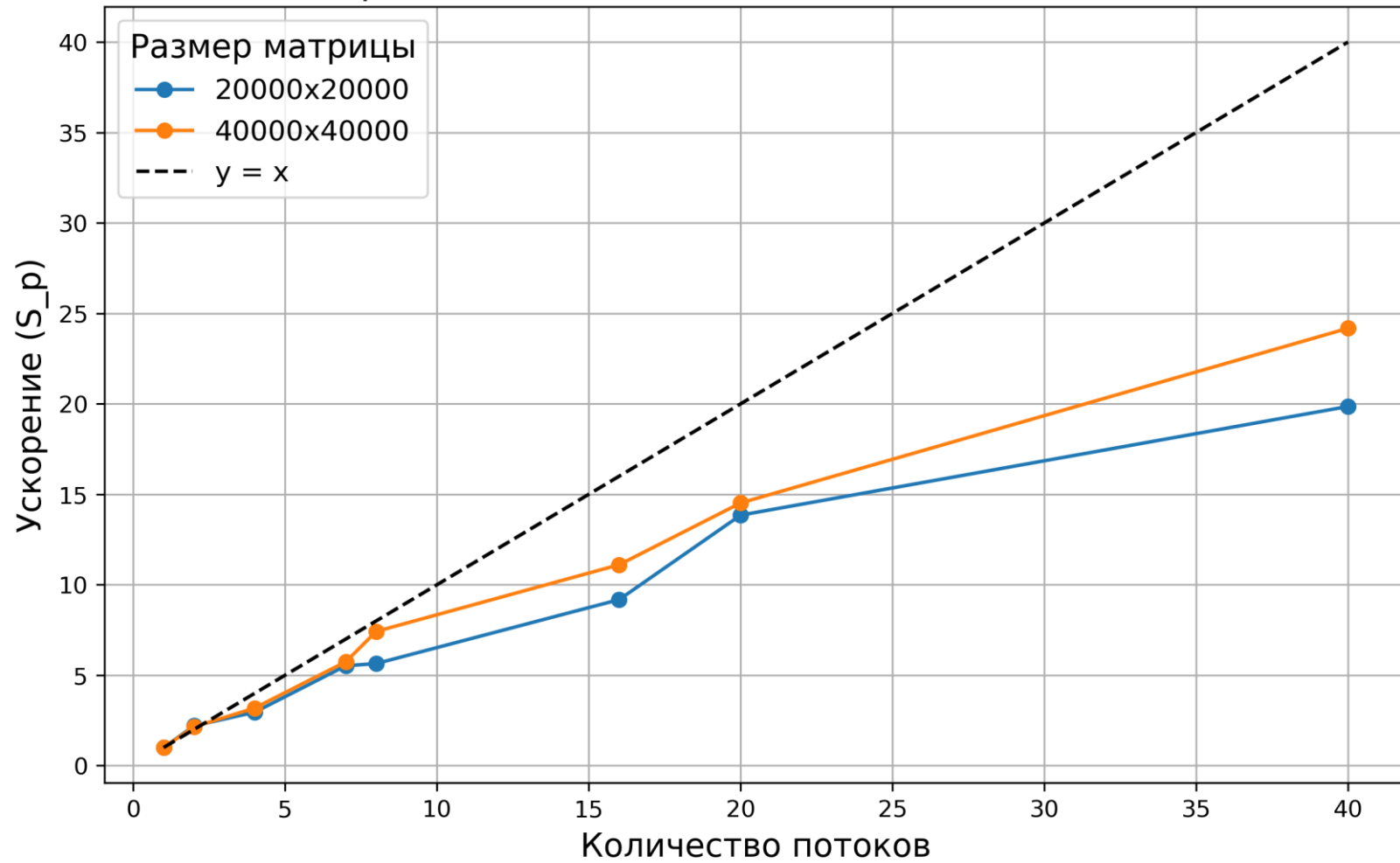


Таблица времени выполнения и ускорения

Размер матрицы / Потоки	1	2	4	7	8	16	20	40
20000 (T _p)	6.97	3.17	2.36	1.26	1.24	0.76	0.50	0.35
	1.00	2.20	2.95	5.52	5.64	9.18	13.85	19.85
40000 (T _p)	29.59	13.79	9.29	5.15	3.99	2.66	2.04	1.22
	1.00	2.15	3.19	5.74	7.42	11.11	14.52	24.18

- Для обоих размеров матриц наблюдается значительное снижение времени выполнения с увеличением количества потоков. При малом количестве потоков (1–4), ускорение растет линейно или почти линейно. При большом количестве потоков (16–40), ускорение продолжает расти, но темпы роста замедляются (скорее всего из-за эффектов параллельной обработки). Матрица размером 40000x40000 показывает более высокое ускорение по сравнению с матрицей 20000x20000 при одинаковом количестве потоков. Это связано с тем, что большие матрицы лучше используют преимущества параллелизма, так как объем вычислений на поток больше.