

Правительство Российской Федерации
Федеральное государственное автономное образовательное учреждение
высшего профессионального образования
**"Национальный исследовательский университет
Высшая школа экономики"**
Департамент прикладной математики, бакалавр

Высокопроизводительные вычисления

ДОМАШНЯЯ РАБОТА №1:

Параллельное умножение матриц (DGEMM)

Выполнил:

Колодин Матвей Алексеевич

Преподаватель:

Тарарушкин Евгений Викторович

Москва, 2025

Содержание

1	График 1. Время выполнения (мс) vs Количество процессов (P)	3
1.1	Наблюдения	3
2	График 2: GFLOPS vs Количество процессов	4
2.1	Наблюдения	4
3	Заключение	5
3.1	Сильная масштабируемость	5
3.2	Итог	6

1 График 1. Время выполнения (мс) vs Количество процессов (P)

1. Ось X — число процессов $P \in \{1, 2, 4, 8, 16, 24\}$,
2. Ось Y — время выполнения (логарифмическая шкала).
3. Три линии — размеры матриц $N = 500, 1000, 1500$.

1.1 Наблюдения

1. Для всех N кривая идёт вниз \rightarrow время уменьшается при увеличении числа процессов.
2. Для малых N (500) эффект слабее — накладные расходы на синхронизацию становятся заметнее.
3. Для больших N (1500) почти идеальная линейная зависимость до 16 процессов, потом рост замедляется (ограничение пропускной способности памяти).

При увеличении числа потоков наблюдается ожидаемое уменьшение времени выполнения. При $N=1500$ масштабируемость близка к идеальной до 16 процессов. После 16 наблюдается насыщение, вызванное конкуренцией потоков за общие ресурсы памяти и шины. Для меньших матриц ($N=500, 1000$) параллельные накладные расходы становятся доминирующими, что снижает эффективность.

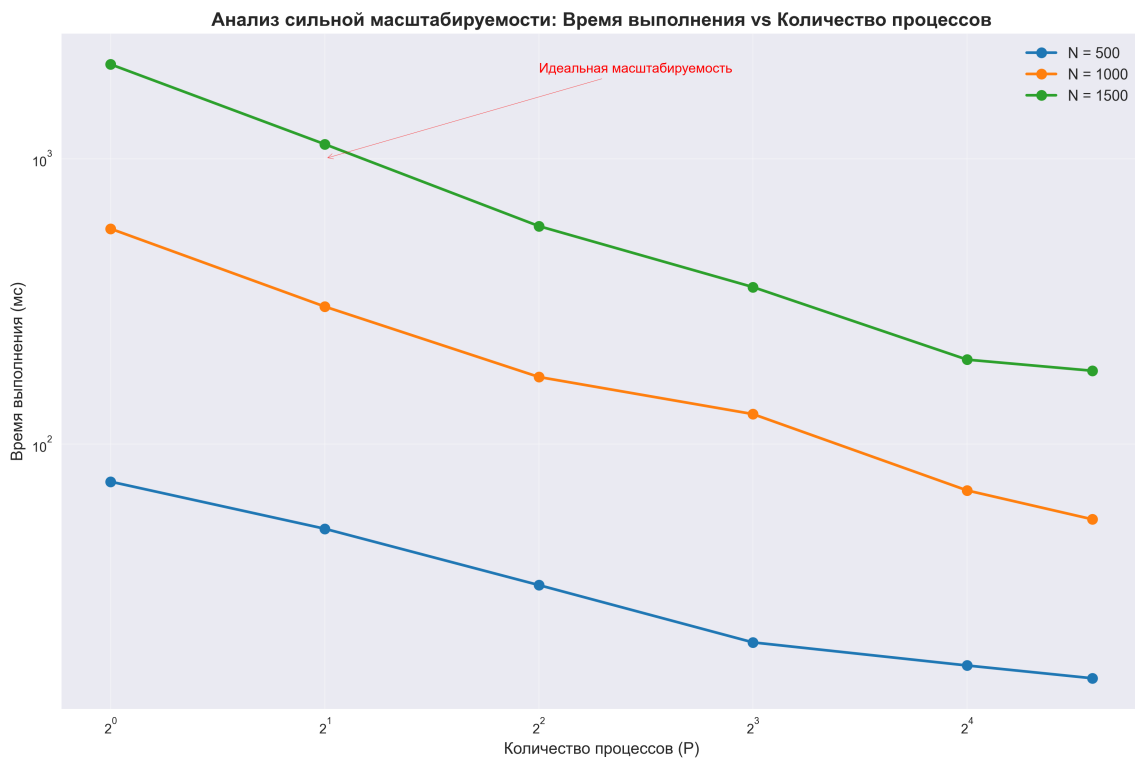


Рис. 1: Результаты расчета по времени выполнения (мс) и связи с количеством процессов (P)

2 График 2: GFLOPS vs Количество процессов

Показывает реальную производительность (кол-во операций с плавающей точкой в секунду).

2.1 Наблюдения

1. Для $N=1500$ производительность растёт почти линейно до 16 процессов и достигает ≈ 35 GFLOPS.
2. Для $N=500$ рост слабее — вычислений мало, и значительная часть времени уходит на синхронизацию потоков.

GFLOPS увеличивается с ростом числа процессов, что подтверждает наличие сильной масштабируемости. Для крупных матриц наблюдается почти линейный рост производительности, что говорит о хорошем утилизации вычислительных ресурсов OpenMP. Для малых матриц рост ограничен накладными расходами, характерными для параллельных вычислений.

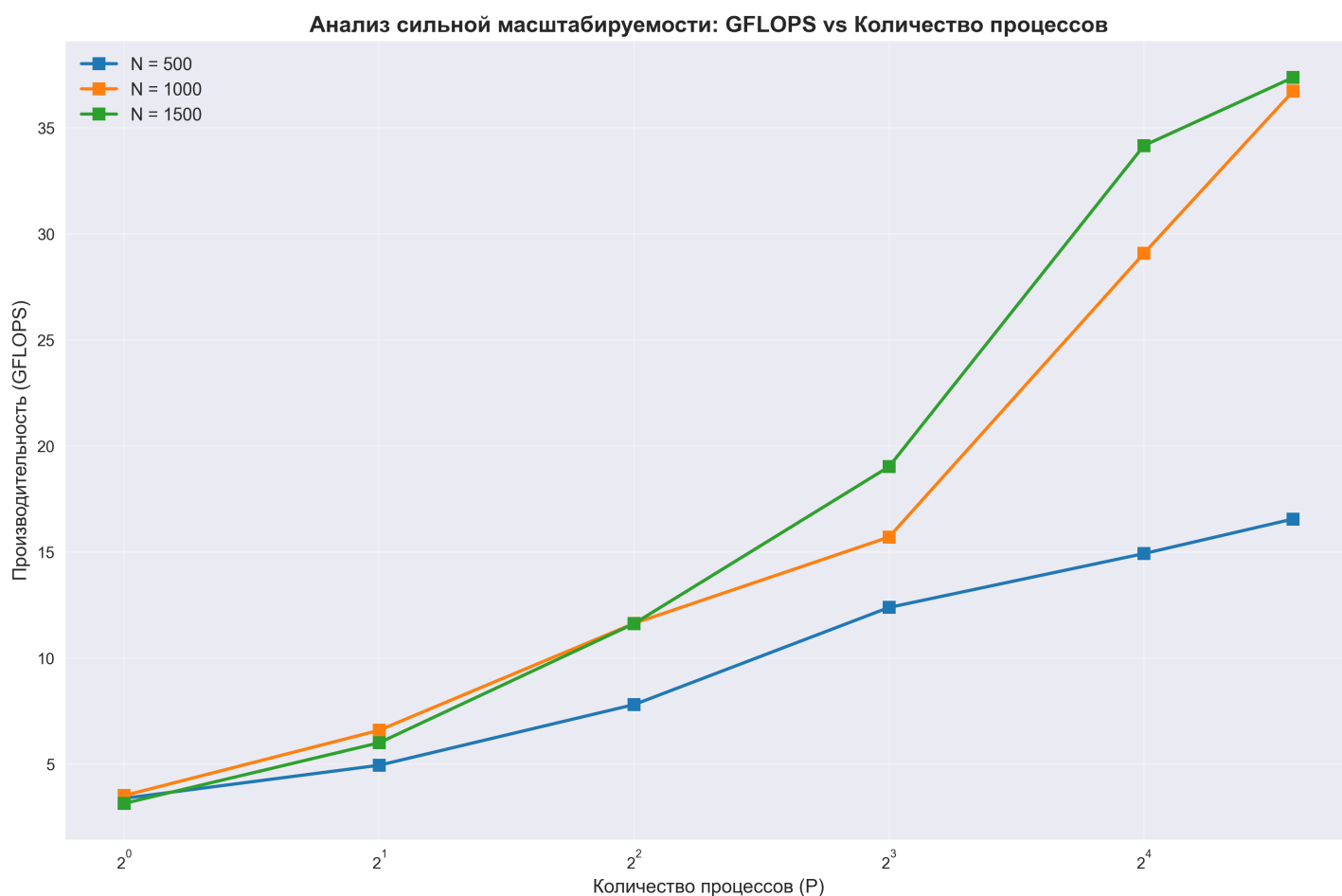


Рис. 2: Результаты расчета по GFLOPS и связи с количеством процессов (P)

3 Заключение

3.1 Сильная масштабируемость

Сильная масштабируемость означает: при фиксированном размере задачи (N константа) — время выполнения уменьшается с ростом числа процессов.

Идеальная масштабируемость - $T(P) = T(1)/P$.

Если построить линии для разных N и провести пунктиром идеальный случай — можно увидеть, что реальные кривые близки к идеальным до определённого порога (обычно 8–16 потоков).

Для размеров матриц $N=1000$ и $N=1500$ наблюдается почти идеальная сильная масштабируемость до 16 потоков.

При дальнейшем увеличении числа потоков прирост производительности снижается, что связано с ограничением пропускной способности памяти и ростом накладных расходов OpenMP :)

См. следующую страницу для совмещённого анализа масштабируемости

3.2 Итог

Проведён анализ сильной масштабируемости DGEMM на суперкомпьютере Харизма для размеров матриц $N = 500, 1000, 1500$ и числа потоков $P = 1 \dots 24$. Результаты показали, что при увеличении числа потоков время выполнения уменьшается, а производительность (GFLOPS) растёт.

Для крупных матриц наблюдается почти идеальная масштабируемость до 16 потоков, после чего рост замедляется из-за накладных расходов синхронизации и ограничений подсистемы памяти.

Для малых матриц масштабируемость слабее из-за снижения отношения вычислений к обмену данными.

В целом реализация демонстрирует хорошую сильную масштабируемость и эффективное использование многопроцессорных ресурсов OpenMP.

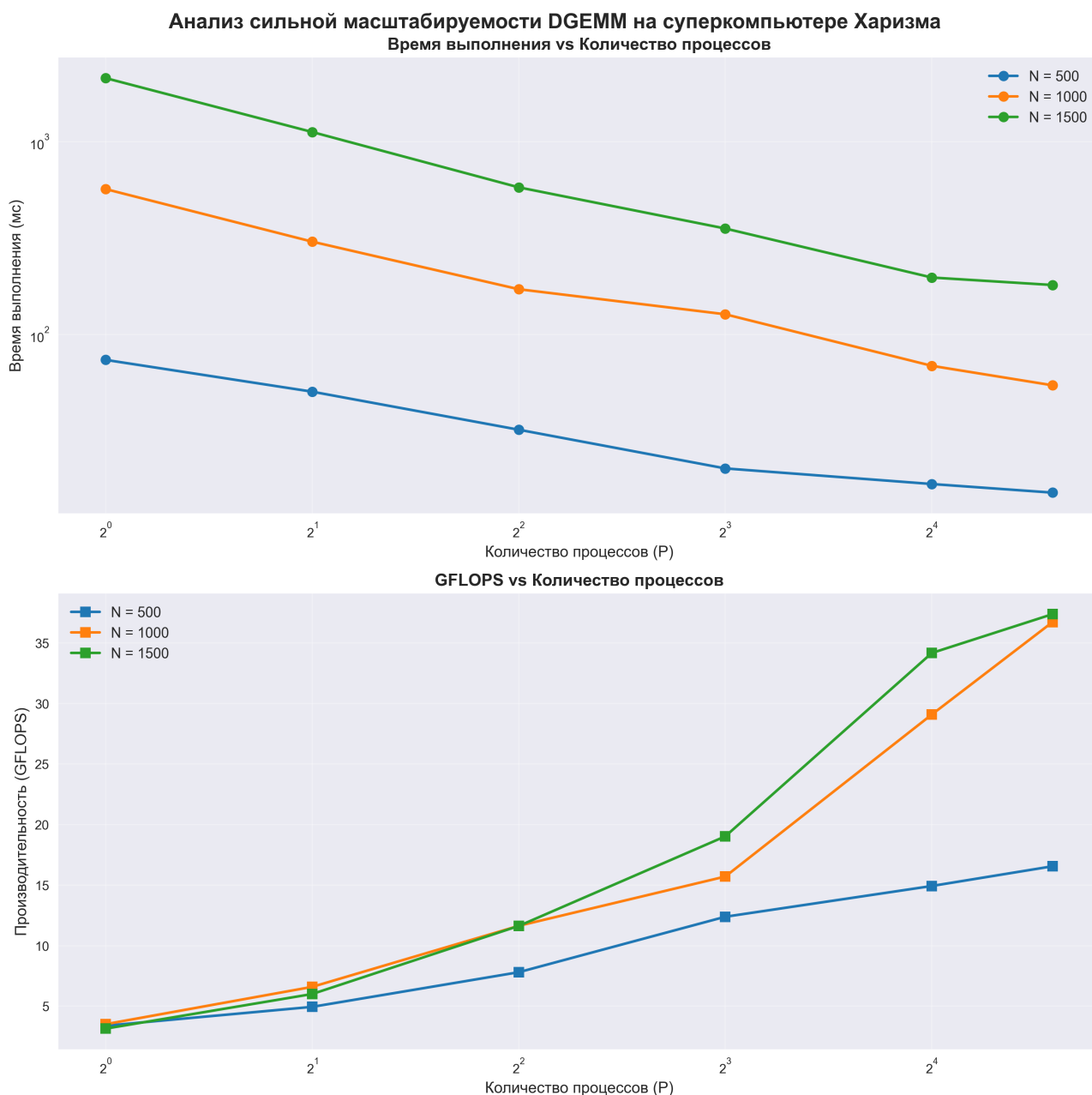


Рис. 3: Совмещённый анализ сильной масштабируемости: зависимость времени выполнения и производительности (GFLOPS) от количества процессов (P) для различных размеров матриц