

При помощи OMP (omp tasks) была реализована программа осуществляющая параллельное выполнение алгоритма FFT (Radix-2 DIT<sup>1</sup>). Корректность программы проверяется приложенным скриптом.

В последовательной версии базовый вариант этого алгоритма предполагает  $M = \log_2 N$  стадий, на каждой из которых вычисляется  $\frac{N}{2}$  бабочек. Каждая бабочка требует 1 комплексное умножение и 2 комплексных сложения. В результате выполняется  $\frac{N}{2} \log_2 N$  комплексных умножений и  $N \log_2 N$  комплексных сложений. В итоге имеем асимптотическую сложность  $\mathcal{O}(N \log N)$ .

Результаты моделирования представлены на рисунке 1. На нём изображены ускорение ( $\mathcal{S}$ ) и эффективность ( $\mathcal{E}$ ) параллельной версии алгоритма на нескольких процессах  $p$ . Стоит отметить, что как ускорение, так и эффективность увеличиваются при росте  $N$ , так как время вычисления задач, начинает сильно преобладать над временем распределения задач.

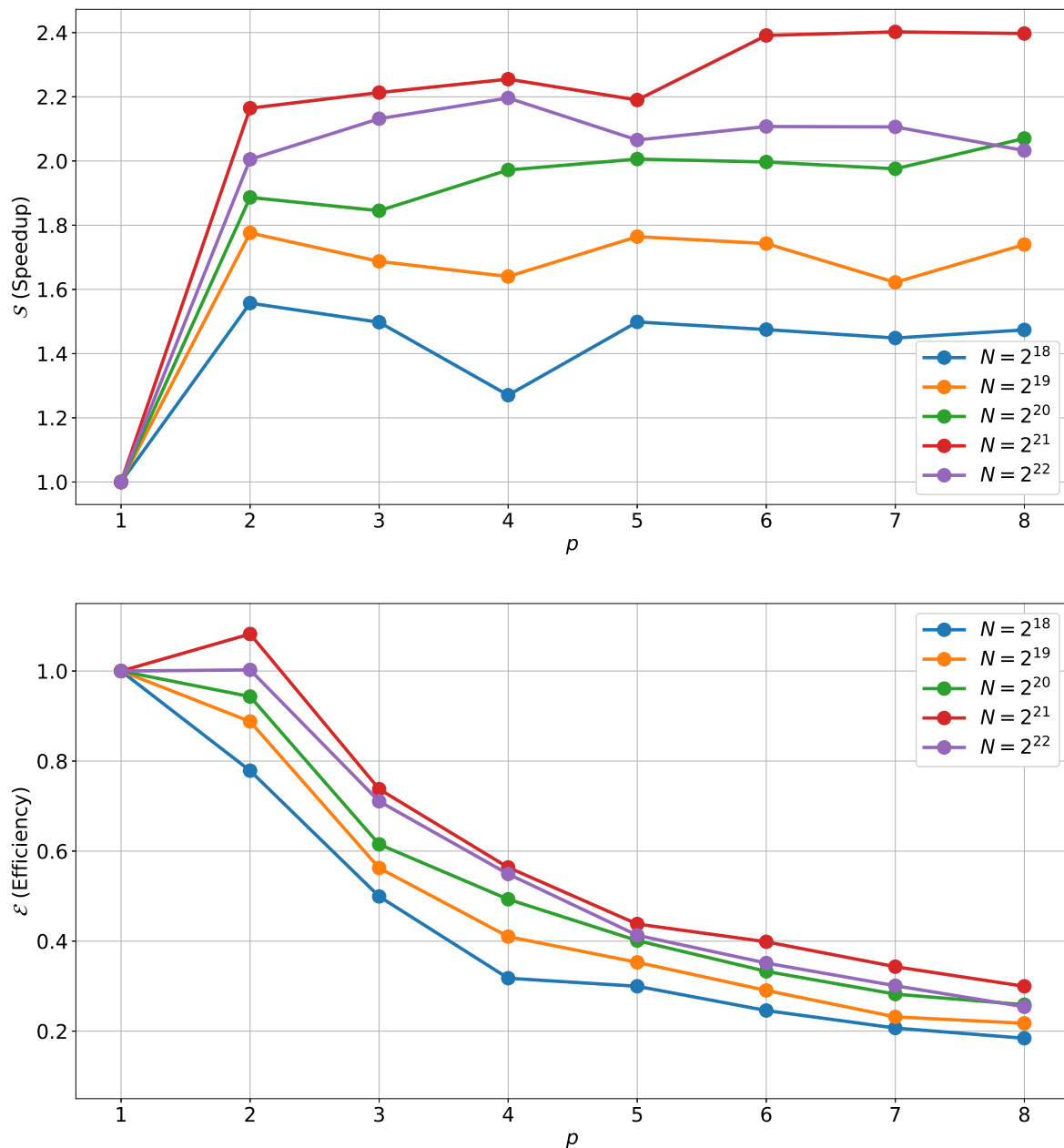


Рис. 1. Результаты моделирования.

<sup>1</sup>[https://en.wikipedia.org/wiki/Coolley%E2%80%93Tukey\\_FFT\\_algorithm](https://en.wikipedia.org/wiki/Coolley%E2%80%93Tukey_FFT_algorithm)