

# Параллельный алгоритм умножения матрицы на вектор

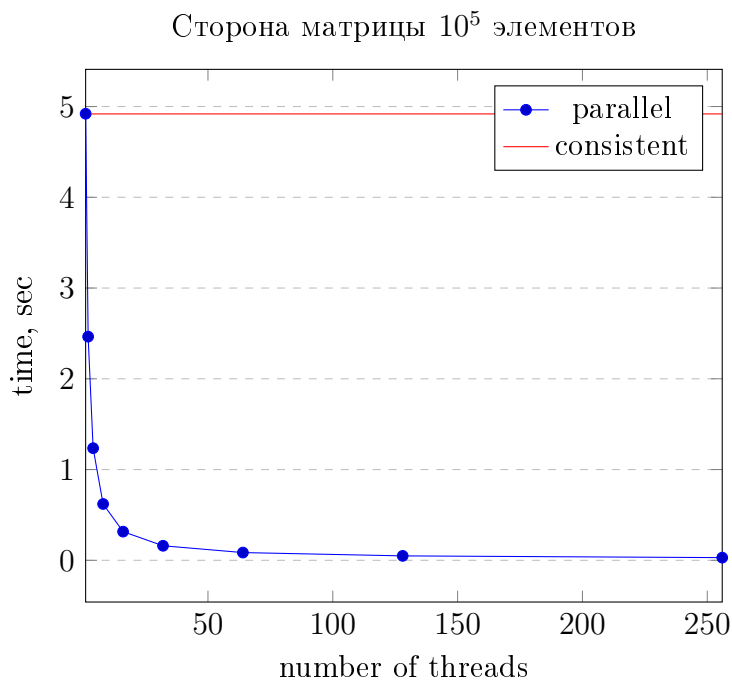
Р.М. Куприй, 323 группа

Входная матрица читается из файла нулевым процессом, разбивается на полосы и распределяется между остальными процессами с помощью операции ScatterV. Копия входного вектора есть у каждого процесса. На каждом процессе вычисляется часть результирующего вектора, который впоследствии собирается воедино через AllgatherV.

Тестирование проводилось на вычислительной системе Bluegene/P для квадратной матрицы со стороной 10000 элементов, и использованием от 1 до 256 процессов.

## 1. Время исполнения

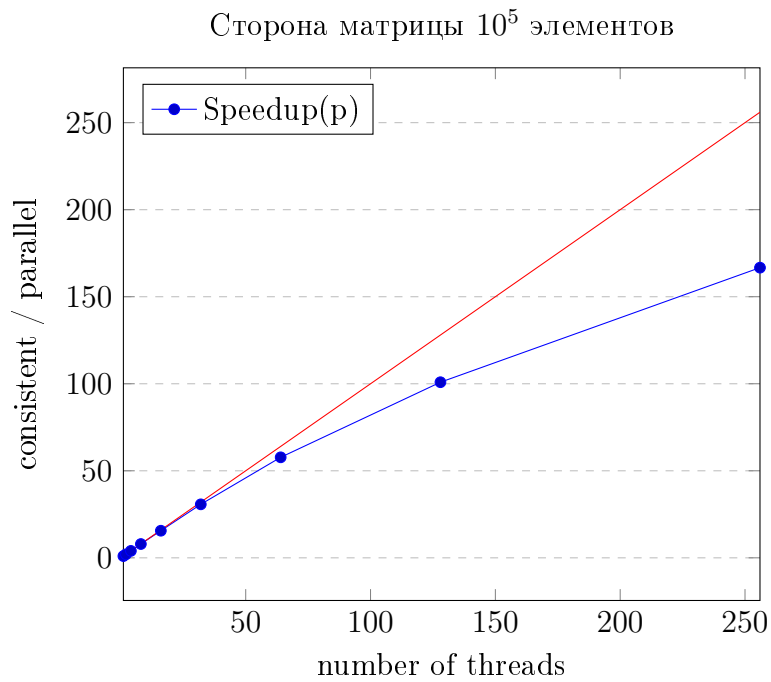
Масштабируемость задачи в зависимости от количества процессов представлена в виде графиков:



По графику видно, что параллельная версия алгоритма сильно сокращает время выполнения операции `dgemv`.

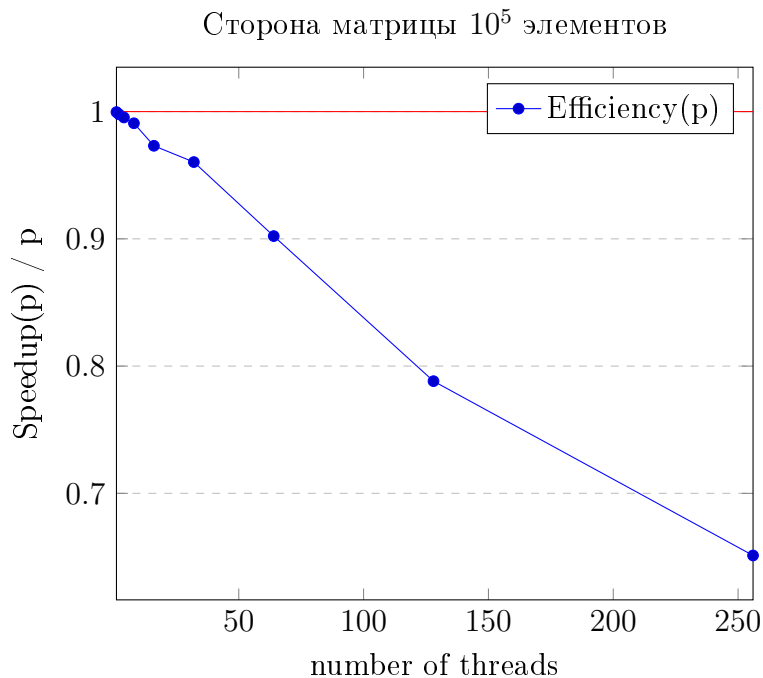
## 2. Ускорение

Соответствующий график ускорения:



По результатам видно, что алгоритм почти линейно масштабируется при использовании до 100 процессов, затем эффективность начинает постепенно снижаться.

### 3. Эффективность



По графику видно, что эффективность алгоритма крайне высока при использовании небольшого числа процессов. При использовании больше 64 процессов эффективность начинает линейно уменьшаться, так влияют дополнительные затраты памяти и времени на коммуникацию между процессами.