

Постановка задачи

- 1) Реализовать последовательный алгоритм перемножения матриц.
- 2) Реализовать программу параллельного перемножения матриц методом Кэннона с помощью MPI.
- 3) Провести ряд тестов. Сравнить ускорение параллельного и не параллельного алгоритма.

Одной из основных операций над матрицами является их перемножение. Эта операция используется практически везде: от компьютерной графики до расчета поверхностей самолетов.

Цель данной работы - реализация параллельного и последовательного алгоритмов, вычисляющих произведение квадратных матриц $n \times n$. В результате работы программы, мы должны получить еще одну матрицу порядка $n \times n$, являющуюся произведением двух исходных, а так же время работы каждого алгоритма.

Требуется реализовать последовательный и параллельный (Кэннона) алгоритмы. Реализацию параллельного алгоритма провести средствами MPI. Кроме того, необходимо провести сравнения времени работы последовательного и параллельного алгоритмов.

Описание алгоритма и метод решения

Наиболее простым методом решения поставленной задачи является последовательная схема перемножения матриц. Он представляется тремя вложенными циклами. Трудоемкость такого алгоритма $O(n^3)$.

Алгоритм Кэннона

В первую очередь создается некоторое число P исполняющих процессов. Это число должно быть полным квадратом. Процессы организуются в виртуальную декартову топологию. Исходные матрицы должны иметь размерность, кратную \sqrt{P} . Матрицы разбиваются на равное количество квадратных блоков. Блоки исходных матриц распределяются по исполняющим процессам. Затем для каждой строки i решетки подзадач блоки матрицы A сдвигаются на $(i-1)$ позиций влево, для каждого столбца j решетки подзадач блоки матрицы B сдвигаются на $(j-1)$ позиций вверх. Далее проводится \sqrt{P} итераций, во время которых сначала происходит перемножение блоков по методу трех вложенных циклов, и произведение складывается с текущим значением результирующего блока. Затем выполняется циклический сдвиг блоков матрицы A вдоль строк решетки и циклический сдвиг блоков матрицы B вверх по столбцам виртуальной решетки. После выполнения всех итераций результирующая матрица собирается из полученных на каждом процессе результирующих блоков.

Демонстрация

