

Учебный курс "Вычисления на многопроцессорных компьютерах"

Лабораторный практикум

Лабораторная работа №4:

Параллелизация алгоритмов умножения матрицы на вектор и перемножения матриц (для матриц общего вида)

Цель работы - освоить и с помощью технологии MPI реализовать алгоритмы распараллеливания операции умножения матрицы на вектор и перемножения матриц большой размерности общего вида (плотных матриц) при использовании различных способов декомпозиции и обмена данными.

Задание к работе:

Разработать программу, параллелизирующую средствами MPI операции умножения матрицы на вектор и матричного умножения. Исследовать эффективность параллелизации.

Варианты заданий

Часть 1. Параллелизация операции умножения матрицы на вектор

1. Параллельный алгоритм, основанный на одномерной (ленточной) схеме разбиения матрицы на горизонтальные полосы с непрерывным распределением строк между процессами.
2. Параллельный алгоритм, основанный на одномерной (ленточной) схеме разбиения матрицы на вертикальные полосы с непрерывным распределением столбцов между процессами.
3. Параллельный алгоритм, основанный на одномерной (ленточной) схеме разбиения матрицы на горизонтальные полосы с чередующимся (циклическим) распределением строк между процессами.
4. Параллельный алгоритм, основанный на одномерной (ленточной) схеме разбиения матрицы на вертикальные полосы с чередующимся (циклическим) распределением столбцов между процессами.
5. Параллельный алгоритм, основанный на двумерной (блочной) схеме разбиения матрицы и предполагающий циклическое распределение блоков между процессами.

Часть 2. Параллелизация операции перемножения матриц

1. Параллельный алгоритм, основанный на ленточной схеме разбиения матриц, при которой первая из перемножаемых матриц разбивается на горизонтальные полосы, а вторая - на вертикальные полосы согласованной ширины. Предполагается непрерывное распределение строк (и соответствующих столбцов) между процессами. Ширина полосы (количество строк и столбцов в ней) определяется числом процессов, запускаемых для выполнения программы.
2. Параллельный алгоритм, основанный на ленточной схеме разбиения матриц на горизонтальные полосы с непрерывным согласованным распределением строк между процессами. Ширина полосы (количество строк в ней) определяется числом процессов, запускаемых для выполнения программы.
3. Параллельный алгоритм, основанный на ленточной схеме разбиения матриц на вертикальные полосы с непрерывным согласованным распределением столбцов между процессами. Ширина полосы (количество столбцов в ней) определяется числом процессов, запускаемых для выполнения программы.
4. Параллельный алгоритм Фокса (Fox's algorithm) при блочной схеме разбиения матриц и циклическом распределении блоков между процессами.
5. Параллельный алгоритм Кэннона (Cannon's algorithm) при блочной схеме разбиения матриц и циклическом распределении блоков между процессами.
6. Параллельный алгоритм Страссена (Strassen's algorithm) при блочной схеме разбиения матриц и циклическом распределении блоков между процессами.
7. Параллельный алгоритм SUMMA (Scalable Universal Matrix Multiply).

Замечания

Все инициализирующие операции, а также операции ввода-вывода целесообразно выполнять в одном процессе (например, нулевом). Для рассылки информации о размерностях массивов (векторов и матриц) следует использовать соответствующие процедуры MPI (например, MPI_Bcast).

Структура отчета по лабораторной работе

По результатам выполнения лабораторной работы необходимо подготовить развернутый отчет, который должен включать следующие разделы:

- название лабораторной работы и текст задания к ней
- формулировка параллельных алгоритмов для соответствующих операций (способ декомпозиции, организация обменов данными)
- исходные коды разработанных программ с содержательными комментариями к реализующим параллельные алгоритмы фрагментам программ и использованным процедурам MPI
- результаты верификации программ на тестовых векторах и матрицах (малой размерности)
- показатели эффективности распараллеливания программ при работе с матрицами большой размерности
- общие выводы по лабораторной работе

Учебные материалы:

[Лекция "Параллельные методы умножения матрицы на вектор"](#)

[Лекция "Параллельные алгоритмы матричного умножения"](#)

[Описание параллельного алгоритма Фокса матричного умножения](#)

[Описание параллельного алгоритма Кэннона матричного умножения](#)

[Описание алгоритма Страссена матричного умножения \(параллельный алгоритм\)](#)

[Описание параллельного алгоритма SUMMA матричного умножения](#)

Книги и учебники:

[Гергель В.П., Фурсов В.А. Лекции по параллельным вычислениям](#) (стр. 119-140, параллельные алгоритмы умножения матрицы на вектор и перемножения матриц)

[Беликов Д.А., Говязов И.В. и др. Высокопроизводительные вычисления на кластерах](#) (стр. 82-95, параллельные алгоритмы перемножения матриц)

[Баканов В.М., Осипов Д.В. Введение в практику разработки параллельных программ в стандарте MPI](#) (стр. 50-58, параллельный алгоритм перемножения матриц)

[Букатов А.А., Дацюк В.Н., Жегуло А.И. Программирование многопроцессорных вычислительных систем](#) (Часть 2 - подробное описание стандарта MPI, стр. 156-159, параллельные алгоритмы перемножения матриц)

[Гришагин В.А., Свистунов А.Н. Параллельное программирование на основе MPI](#) (стр. 65-74 - параллельный алгоритм Фокса перемножения матриц)

[Богачев К.Ю. Основы параллельного программирования](#) (с. 261-272, параллельный алгоритм умножения матрицы на вектор)

[Шпаковский Г.И., Серикова Н.В. Программирование для многопроцессорных систем в стандарте MPI](#) (Раздел 2 - подробное описание стандарта MPI, стр. 172-185, параллельные алгоритмы перемножения матриц)

[MPI Tutorial \(примеры программ на Fortran\)](#)

[Ортега Дж. Введение в параллельные и векторные методы решения линейных систем](#) (стр. 50-74, параллельные алгоритмы умножения матрицы на вектор и перемножения матриц)

[Shonkwiler R.W., Lefton L. An Introduction to Parallel and Vector Scientific Computation \(Cambridge Texts in Applied Mathematics\)](#) (стр. 115-120, параллельные алгоритмы умножения матрицы на вектор и перемножения матриц)

[MPI: The Complete Reference](#) (руководство по MPI с примерами)

Примеры программ:

- [Программа инициализации матриц с помощью генератора случайных чисел](#)
- [Программа умножения матрицы на вектор \(последовательный код\)](#)
- [Программа перемножения матриц \(последовательный код\)](#)
- [Программа умножения матрицы на вектор \(параллельный OpenMP-код\)](#)

- [Программа перемножения матриц \(параллельный OpenMP-код\)](#)
 - [Примеры программ из учебника Антонова А.С. "Технологии параллельного программирования MPI и OpenMP"](#)
-