



Применение специализированных вычислительных комплексов позволяет ускорить решение некоторых классов задач за счет аппаратной реализации операций. Однако специализация резко ограничивает область применения таких систем. Напротив, простая в организации архитектура вычислительной системы кластерного типа оказывает влияние на время решения задач. Накладные расходы, связанные с пересылкой сообщений, могут перекрыть эффект от использования множества вычислительных устройств.

Применение кластерных вычислительных систем связано с рядом проблем. Известно, что на время решения задачи в кластере влияет количество используемых процессоров [1]. При проектировании вычислительного кластера для конкретной прикладной задачи требуется предварительный анализ ресурсной базы и реализации алгоритма: необходимо оценить объем вычислительной работы и определить количество вычислительных устройств, при котором время решения будет наименьшим, что позволит максимально эффективно использовать имеющиеся ресурсы.

Определение оптимального количества вычислительных устройств нетривиально, так как время решения задачи зависит от множества факторов: количества оперативной памяти на узлах кластера, производительности дисков и коммуникационной среды и, наконец, программной реализации алгоритма решения задачи [2]. Таким образом, актуальными являются исследования, связанные с разработкой способов преобразования циклических конструкций вычисления рекуррентных последовательностей и циклов с использованием индексных множеств для многопроцессорных систем кластерного типа с учетом количества задействованных вычислительных устройств.

В работе авторы используют методики оптимизации определения минимально достаточного числа необходимых вычислительных ресурсов с помощью разработанных алгоритмов, описанных в работах, где потребовалось находить оценки параметров нелинейных систем с помощью полного метода наименьших квадратов, что обусловило использование больших вычислительных мощностей нескольких компьютеров, ввиду большой вычислительной сложности задачи.

Применение указанных алгоритмов позволило создать эффективные программы решения поставленной задачи, которые работают в реальном времени.

#### Литература

1. Вишневецкий, В.М. Теоретические основы построения компьютерных сетей: учеб. - метод. пособие / В.М. Вишневецкий. - Изд. м.: Техносфера, 2003. - 219 с.
2. Мясин, В.В. Методика определения оптимального количества процессоров кластерной системы : учеб. для вузов / В.В. Мясин. - Изд. Томск, ТПУ, 2007. - 468 с.
3. Максимов, Н.В. Архитектура ЭВМ и вычислительных систем : учеб. для вузов / Н.В. Максимов, Т.Л. Партыка, И.И. Попов. - М.: Инфра - М, 2005.- 512 с./



4. Львов, А.А. Алгоритм оценки параметров математических моделей линейных и нелинейных систем / А.А. Львов, А.А. Северов // Вестник СГТУ, № 4(43), 2009. – С.77-81.

5. Львов, А.А. Анализ моделей метода наименьших квадратов и методов получения оценок / А.А. Львов, М.В. Мусаев // Вестник СГТУ, № 4(43), 2009. – С.137-141.

М.С. Мутагаров

#### ПРОГРАММНЫЙ КОМПЛЕКС ДЛЯ МОДЕЛИРОВАНИЯ ПАРАЛЛЕЛЬНЫХ АЛГОРИТМОВ МАТРИЧНОГО УМНОЖЕНИЯ

(Самарский государственный университет путей сообщения)

Многие алгоритмы управления, идентификации систем, цифровой обработки сигналов широко используют матричные вычисления, одной из основных задач в которых является умножение матриц.

Классический алгоритм матричного умножения является итеративным и ориентирован на последовательное вычисление строк результирующей матрицы  $C$ . Известно [1], если исходные матрицы  $A$  и  $B$  квадратные размера  $n \times n$ , каждый элемент матрицы  $C$  произведения есть скалярное произведение строки и столбца исходных матриц  $A$  и  $B$ , т.е. при умножении квадратных матриц размера  $n \times n$  количество выполненных операций имеет порядок  $O(n^3)$ . Очевидно, что такой большой объем вычислений, которые необходимо производить в системах управления в реальном масштабе времени, существенно ограничивает сложность решаемых такими системами задач.

В настоящее время разработаны и продолжают интенсивно разрабатываться различные параллельные алгоритмы умножения матриц. В параллельных алгоритмах, исходя определения операции матричного умножения, вычисление всех элементов матрицы  $C$  может быть выполнено независимо друг от друга, т.е. используется свойство параллелизма по данным. Поэтому основной принцип организации параллельных вычислений заключается в использовании в качестве базового потока вычисления одного элемента результирующей матрицы  $C$ , т.е. вычисления над элементами одной строки матрицы  $A$  и одного столбца матрицы  $B$ . Очевидно, общее количество потоков получаемых при таком подходе равно  $n^2$  [1], что определяет число требуемых вычислительных устройств.

Актуальной задачей параллельных вычислений является определение показателей эффективности параллелизма: оценка получаемого ускорения вычислений, степень загрузки процессоров и ее балансировка, потери производительности для организации параллелизма, отношение производительности вычислительной системы к ее стоимости и др. Эффективность параллелизма в значительной мере зависит от того, в какой мере параллельный алгоритм учитывает характерные свойства параллельной ВС.



Например, в параллельных алгоритмах достигнутый уровень параллелизма может оказаться избыточным, когда при проведении практических расчетов количество сформированных потоков превышает число имеющихся вычислительных устройств (процессоров и/или ядер). В этом случае целесообразно укрупнение базовых потоков, т.е. объединении в рамках одного потока вычислений всех элементов одной из строк матрицы  $C$ . Тогда общее количества уменьшится до величины  $n$  [1].

Из вышесказанного следует актуальность решения задачи - проведения этапа предварительного моделирования параллельных алгоритмов матричного умножения перед практическим применением в прикладных задачах, решаемых на конкретных вычислительных системах (ВС).

Для решения этой задачи разработан программный комплекс [2] для моделирования параллельных алгоритмов матричного умножения, который позволяет выполнять следующие функции:

- произвести выбор для моделирования из базы алгоритмов того или иного параллельного алгоритма матричного умножения;
- выбрать тот или иной вид и размер матриц для проведения расчетов;
- произвести оценку времени проведения матричного умножения;
- вычислить зависимости времени проведения матричного умножения от размерности и вида матриц, а также от числа используемых ядер;
- получить сравнительные оценки времени проведения матричного умножения для разных видов алгоритмов.

Архитектура программного комплекса изображена на рис.1 и состоит из 9 подсистем, образующих три уровня: презентационный, логики и данных.

Программный комплекс позволяет определять показатели эффективности параллельных алгоритмов и программ матричного умножения, выполняемых на многоядерных ВС, использующих технологию CUDA (Computer Unified Device Architecture). Эта технология компании Nvidia [3] позволяет создавать недорогие высокопроизводительные параллельные ВС для разработки приложений для массивно-параллельных ВС на основе графических процессоров (GPU). Эти процессоры выполняют функцию быстродействующих SIMD-процессоров, архитектура которых (одиночный поток команд и множественный поток данных) удобна для реализации параллельных алгоритмов матричного умножения.

Пользовательский интерфейс программного комплекса для моделирования параллельных алгоритмов матричного умножения представлен на рис.2., а примеры результатов моделирования алгоритмов - на рис.3 и рис.4.

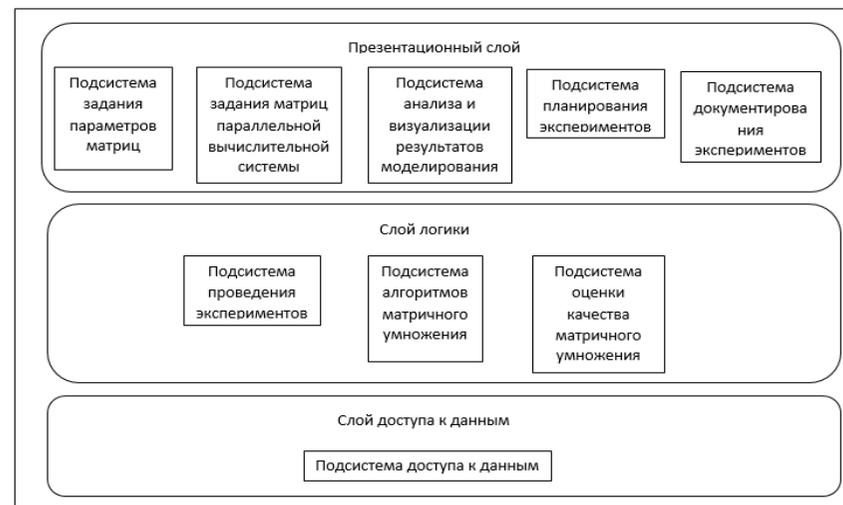


Рис.1. Архитектура программного комплекса для моделирования алгоритмов матричного умножения

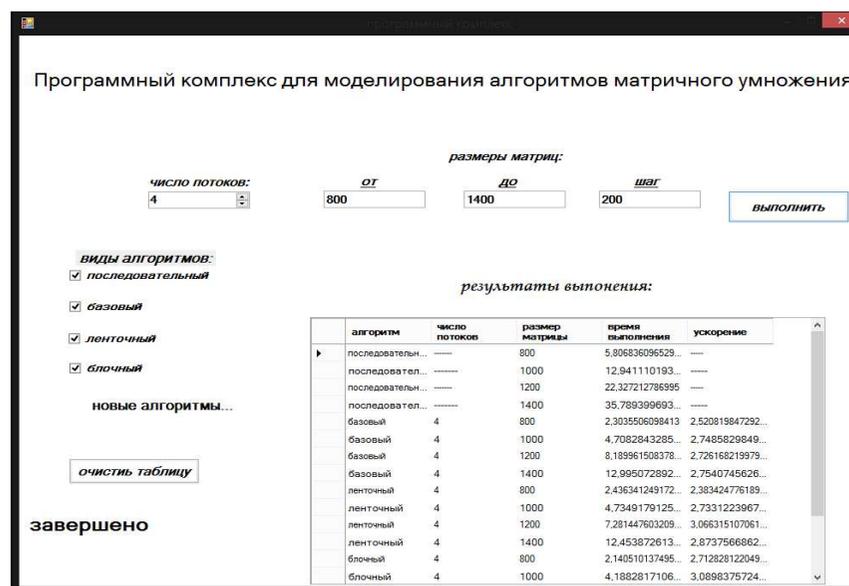


Рис. 2. Пользовательский интерфейс программного комплекса

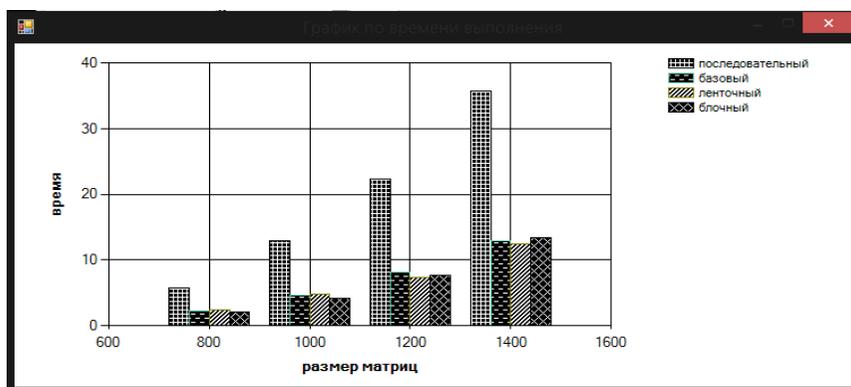


Рис. 3. Сравнительные результаты времени выполнения матричного умножения для различных параллельных алгоритмов

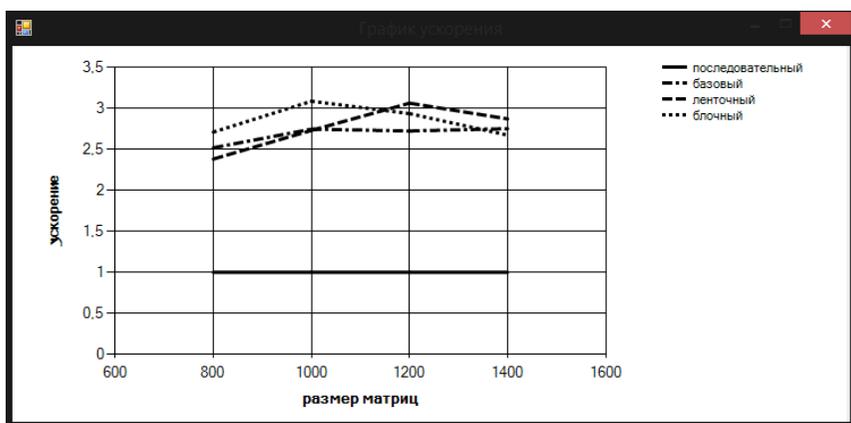


Рис. 4. Сравнительные результаты ускорения выполнения матричного умножения для различных параллельных алгоритмов

Примером практического применения программного комплекса является разработка вычислительной платформы для восстановления и разделения сигналов - определения сигналов источников, недоступных для прямых измерений, по измеренным в доступных точках сигналам приемников [4]. Моделирование позволило разработать программное обеспечение и выбрать оптимальную по соотношению производительность-цена архитектуру ВС для обработки в реальном времени сигналов системы автоматической локомотивной сигнализации [5], обеспечивающей безопасность движения поездов.

### Литература

1. Гергель В.П. Высокопроизводительные вычисления для многоядерных многопроцессорных систем. – Нижний Новгород: НГУ им Н.И Лобачевского, 2010. – 421 с.
2. Мутагаров М.С., Засов В.А. Программа для моделирования параллельных алгоритмов матричного умножения. - Свидетельство о государственной регистрации программы для ЭВМ №2016610466 от 12.01.2016г
3. Боресков А.В., Харламов А.А. Основы работы с технологией CUDA. – М.: ДМК Пресс, 2011.-231с.:ил.
4. Засов В.А., Ромкин М.В. Параллельные вычисления в задаче разделения сигналов в многомерных динамических системах // Труды VI Международной конференции «Параллельные вычисления и задачи управления (РАСО-2012)». - М.: Учреждение Российск. акад. наук Ин-т проблем управления им. В.А. Трапезникова РАН. – 2012. – С.96-102.
5. Засов В.А., Ромкин М.В. Модель нерекурсивного вычислителя для решения задачи разделения сигналов // Аналитические и численные методы моделирования естественно -научных и социальных проблем: сб. статей VIII международной науч.- техн. конф. - Пенза: изд-во ПГУ, 2013. – С.174-179.

Н.В. Мясникова, М.П. Берестень, Б.В. Цыпин, М.Г. Мясникова

### ИСПОЛЬЗОВАНИЕ РАЗЛОЖЕНИЯ НА ЭМПИРИЧЕСКИЕ МОДЫ НА ОСНОВЕ ДИФФЕРЕНЦИРОВАНИЯ И ИНТЕГРИРОВАНИЯ В ИИС

(Пензенский государственный университет)

В последнее время возрастает интерес к разложению на эмпирические моды. Авторы используют такое разложение как предварительную обработку сигналов, позволяющую увеличить отношение сигнал/помеха, упростить алгоритм параметрического анализа (за счет сведения сложной задачи оценивания параметров модели порядка  $p$  к простым задачам оценивания параметров составляющих первого и второго порядков) и при этом существенно сократить время анализа.

Предварительное разложение использовалось авторами для измерения параметров сигналов сложной формы, а также для сжатия и восстановления сигналов в распределенных ИИС [1-5]. При этом рассматривались наиболее известное разложение empirical mode decomposition (EMD) и разложение на основе экстремальной фильтрации (ЭФ) [6-8]. В настоящее время предложены новые методы декомпозиции на основе дифференцирования-интегрирования [9-11]. Эти методы позволяют проводить разложение на эмпирические моды, как в порядке возрастания, так и в порядке убывания частот. Этот фактор важен, так как часто информативными являются именно низкочастотные компоненты.

Предложенный метод основан на подавлении высокочастотных составляющих при интегрировании и на их акцентировании при дифференцировании: