

*Н. АЗИМБАЕВ¹, С. КУЛЬДИН², А. ПАК⁴, В. КЛИМЕНОВ³, А. ИСОВА³,
И. НЕВМЕРЖИЦКИЙ³, С. ТОКМОЛДИН³*

(¹Московский государственный университет им. М. В. Ломоносова, Москва, РФ,

²Московский физико-технический институт, Москва, РФ,

³Физико-технический институт, Алматы, Республика Казахстан,

⁴Уорикский университет, Ковентри, Великобритания)

РАСПАРАЛЛЕЛИВАНИЕ СИСТЕМЫ ВИЗУАЛИЗАЦИИ И ОБРАБОТКИ ДАННЫХ ЛАЗЕРНОГО ГЕТЕРОДИННОГО ДЕФОРМОГРАФА НА БАЗЕ ТЕХНОЛОГИИ «NVIDIA CUDA»

Аннотация. В работе проводится анализ эффективности использования графических процессоров (GPU) для решения актуальной задачи подсчета спектральных характеристик обработанных данных лазерного гетеродинного деформографа в режиме реального времени. Актуальность задачи связана с широким внедрением графических процессоров в современные вычислительные системы и необходимостью учета особенностей этих процессоров при разработке прикладного программного обеспечения. Одной из возможных перспектив является использование для решения рассматриваемой в работе задачи гибридных кластерных вычислительных систем, имеющих в составе своих узлов как графические, так и универсальные процессоры, неоднородных высокопроизводительных вычислительных систем. Разработка эффективного прикладного обеспечения для таких систем в настоящее время позволит обрабатывать большой объем информации, тем самым увеличить точность производимых вычислений.

Ключевые слова: технология «NVIDIA CUDA», обработка информации, спектральный анализ, лазерный гетеродинный деформограф, параллельные вычисления.

Тірек сөздер: «NVIDIA CUDA» технологиясы, ақпараттарды өңдеу, спектрлік талдау, лазерлі гетеродиндік деформограф, параллельдік есептеу.

Keywords: «NVIDIA CUDA» technology, data mining, spectral analysis, laser heterodyne deformograph, parallel computations.

Началом проведения работ послужило развитие новых высокоточных аппаратных средств геомониторинга в широком частотном диапазоне. Прежний подход включал в себя накопление базы данных сигналов физических величин с последующей Off-line обработкой статистическими методами анализа. Увеличение чувствительности и расширение частотного диапазона аппаратных средств мониторинга привело к увеличению потока данных и вызвало некоторые сложности с накоплением и систематизацией в базу данных. Кроме того с расширением частотного диапазона измерительной аппаратуры прежние методы анализа столкнулись с отсутствием статистических решений для более высоких частот. Поэтому переход к динамическому

анализу в виде спектрального представления сигналов открывает новую область математического анализа и аппаратно-программных решений для широкого круга задач.

В данной работе проводится анализ эффективности использования графических процессоров (GPU) для решения актуальной задачи подсчета спектральных характеристик обработанных данных лазерного гетеродинного деформографа в режиме реального времени.

Актуальность задачи связана с широким внедрением графических процессоров в современные вычислительные системы и необходимостью учета особенностей этих процессоров при разработке прикладного программного обеспечения. Одной из возможных перспектив является использование для решения рассматриваемой в работе задачи гибридных кластерных вычислительных систем, имеющих в составе своих узлов как графические, так и универсальные процессоры, неоднородных высокопроизводительных вычислительных систем. Разработка эффективного программного обеспечения для таких систем в настоящее время позволит обрабатывать большой объем информации, тем самым увеличить точность производимых вычислений.

Алгоритм чтения и обработки данных, получаемых с лазерного гетеродинного деформографа, был реализован в соответствии с их внутренней структурой, представленной во входных двоичных файлах. Исходные данные представляют собой короткие файлы, содержащие информацию с деформографа за каждые 5 минут. При этом частота записи составляет 100 Hz.

Для улучшения производительности исследования программы была реализована возможность сбора всех полученных данных в единую базу («склейка»). Более того, была добавлена возможность сжатия информации и хранения в необходимом формате. При этом каждый файл может быть просмотрен с использованием интерфейса разработанного программного обеспечения. Для оптимизации чтения большого числа маленьких файлов были использованы методы параллельного ввода/вывода (например, MPI-IO). В случае увеличения объемов входных данных и дополнительной нагрузки на систему ввода/вывода возможно также проведение отдельного исследования по нахождению оптимальных вариантов программно-аппаратной реализации ввода/вывода на целевых вычислительных архитектурах.

В работе были рассмотрены возможности распараллеливания для двух основных классов параллельных архитектур:

– SMP (и другие архитектуры на общей памяти)

Так как исходный массив данных используется исключительно для чтения, в случае параллельной обработки на SMP-архитектурах к нему могут осуществлять доступ все потоки без каких-либо существенных потерь на блокировки, синхронизацию и пересылки данных. Кроме того, цикл, вычисляющий i -й элемент массива частных интегральных сумм, не содержит каких-либо сложных зависимостей от смежных элементов, и может успешно распараллеливаться на общей памяти. При этом на каждый процессор будут отводиться вычисления определенного блока данного массива.

– MPP (и другие архитектуры на распределенной памяти)

В данном случае массив исходных данных может копироваться на каждый вычислительный узел кластера, за счет его предварительной широковещательной рассылки (broadcast) и далее использоваться в локальных вычислениях каждого узла без дополнительных пересылок данных. Что касается вычисления элементов массива частных интегральных сумм, то его целесообразно осуществлять на одном узле, а распараллеливание производить по целым строкам матрицы (на каждый вычислительный узел будет приходиться несколько строк матрицы). Распараллеливание внутреннего цикла на MPP архитектуре привело бы к лишним пересылкам данных и скорее всего не принесло бы существенного выигрыша в производительности.

Нетрудно заметить, что для архитектур с общей памятью приемлемо распараллеливание как по внешнему, так и по внутреннему циклу, тогда как для архитектур с распределенной памятью только по внешнему. В результате для параллельного вычисления матрицы спектра сопоставимой по размерам с объемом оперативной памяти рассматриваемых архитектур (до нескольких гига-байт) – вычисления на общей памяти наиболее целесообразны.

Однако в случае очень больших объемов данных, в связи со сказанным выше, наиболее оптимально было бы использование современных гибридных NUMA-архитектур (многопроцессорные системы с распределенной памятью, где каждый узел представляет собой многоядерный процессор на общей памяти). На такой архитектуре возможно совмещение плюсов обоих перечисленных подходов: строки матрицы вычисляются на отдельных вычислительных узлах, тогда как отдельные блоки массива интегральных сумм вычисляются на общей памяти за счет многоядерности узлов системы.

При распараллеливании вычислений на каждой конкретной архитектуре для достижения максимальной производительности будет необходимо эмпирическим путем оценить оптимальные значения следующих параметров:

- Количество строк матрицы приходящихся на один поток вычислений.
- Размер блока массива интегральных сумм, приходящийся на один поток вычислений.
- Нахождение оптимальных значений предлагается осуществить путем написания и прогона специальных профилирующих тестов, позволяющих получить графики производительности алгоритма для разных значений перечисленных параметров.

Методы расчета спектральных характеристик нестационарных сигналов. В случае с дан-ными лазерного гетеродинного деформографа наиболее подходящие базисные вейвлетные функции были получены как производные функции Гаусса, которые имеют наилучшую локализацию как во временной, так и в частотной областях. Для нашей задачи мы рассмотрели два варианта.

Первый-вейвлет вычисляется по первой производной. Форма вейвлета относится к нечетным функциям и, соответственно, спектр вейвлета является мнимым. Уравнение вейвлета с единичной нормой:

$$\varphi(x) = \frac{-1.786}{\sqrt{2}} x e^{-x^2}$$

Второй-вейвлет вычисляется по второй производной. Вейвлет симметричен, спектр вейвлета представлен только действительной частью и хорошо локализован по частоте,

нулевой и первый моменты вейвлета равны нулю. Применяется для анализа сложных сигналов. Уравнение вейвлета с единичной нормой:

$$\varphi(x) = 1. \frac{031}{\sqrt{2}} e^{-x^2} (1 - 2x^2)$$

Анализ подходов к сегментации частотного отрезка. В связи с большими вычислительными затратами по нахождению спектральных характеристик был разработан алгоритм использования фрактальной теории распределения частот. Согласно данному алгоритму сегментация частотного отрезка происходит согласно следующей схеме.

- 1) Разбиение всего частотного отрезка на целое число сегментов одинаковой длины.
- 2) Для каждого сегмента частотного отрезка производится выборка анализируемых частот по нормальному распределению с математическим ожиданием в его центре и дисперсией, зависящей от длины этого сегмента.
- 3) Вычисляется спектр для этого сегмента.

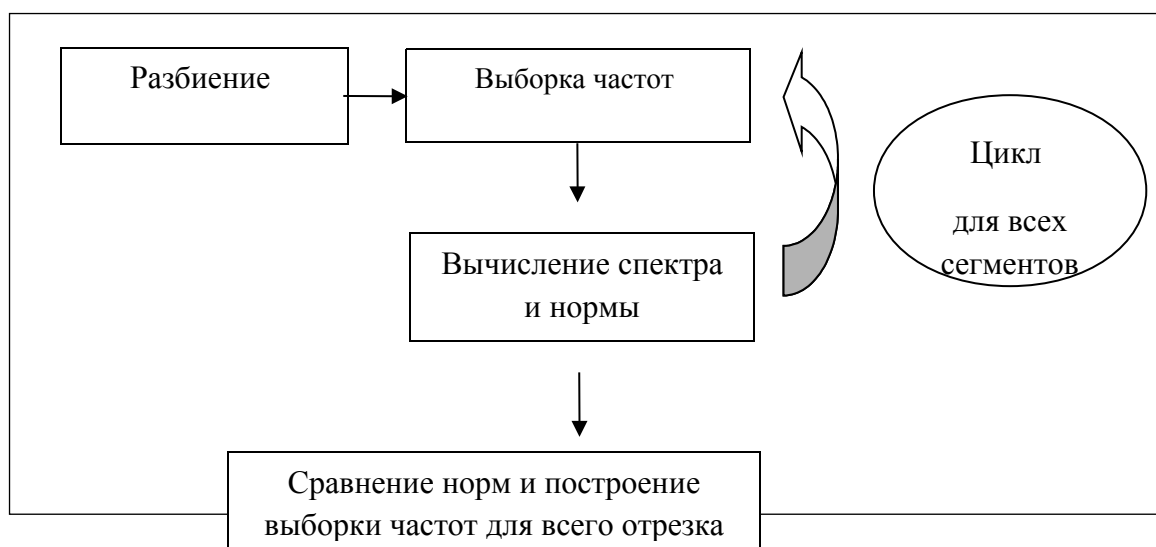


Рисунок 1 – Общая схема разработанного алгоритма

4) Вводим понятие нормы по построенному спектру для каждого сегмента как объем под графиком абсолютного значения спектра, и вычисляем его.

5) Производим анализ полученных значений норм. Далее выбираем сегмент, который имеет наибольшую норму. В результате чего строим выборку анализируемых частот всего отрезка по такому же разбиению, взяв в качестве математического ожидания середину значимого сегмента.

Алгоритм построения спектра. Значение спектра в момент времени b и с частотой a вычисляется по формуле:

$$C(a, b) = |a|^{\frac{1}{2}} \int_{-\infty}^{\infty} s(t) \varphi((t - b) * a) dt$$

где $s(t)$ – исходный сигнал, $|a|^{\frac{1}{2}}$ – нормировочный коэффициент. Диапазон значений b – это весь временной диапазон исходных данных, а частоты a выбираются согласно пункту «Анализ подходов к сегментации частотного отрезка».

В связи с дискретностью данных интеграл считается как частичная сумма с наименьшим возможным временным шагом, в нашем случае это 0,01 (частота лазерного гетеродинного дефор-мографа равна 100 Hz). Можно также заметить, что носитель вейвлетных функций есть отрезок $[-30, 30]$, так как оценка порядка ее значения при $|x| > 30$ равна $1.0e-390$, что приводит к неразличимости ее с нулем для ЭВМ. В связи с этим при использовании первого вейвлета, у которого значение в нуле равно нулю, значения спектра с частотой, превосходящей частоту исходных данных (100 Hz) в 30 раз, будут равны 0. Для второго вейвлета спектр на «высоких» частотах будет повторять исходный сигнал.

Применение GPU как базиса для построения кластерных систем. Параллельные компьютерные системы можно условно классифицировать в соответствии с уровнем, на котором оборудование поддерживает параллелизм [4]:

- многоядерные системы;
- симметричные мультипроцессорные системы;
- распределенные компьютерные системы.

Распределенный компьютер представляет собой компьютерную систему с распределенной памятью, в которой вычислительные устройства объединены в единую систему по сети. Данный вид параллельных компьютерных систем проявляет высокую способность к масштабированию. Основные подклассы распределенных компьютерных систем следующие:

- кластерные системы;
- массивно параллельные компьютеры;
- грид-системы;
- специализированные параллельные компьютеры;
- FPGA-системы;
- GPGPU-системы.

Особый интерес в данное время вызывают системы с архитектурой GPGPU. GPGPU (General Purpose GPU) – системами называют компьютеры, позволяющие производить вычисления общего назначения на видеокартах. Вычисления общего назначения на видеокартах являются современным трендом исследований в области компьютерной инженерии. Они позволяют задействовать мощности современных графических карт для решения задач, которые не ограничиваются областью компьютерной графики.

Графические процессоры являются, по сути, сопроцессорами, значительно оптимизированными под задачи компьютерной графики – в частности, под решение задач линейной алгебры.

На данный момент существует несколько технологий (например, CUDA и OpenCL), позволяющие использовать достаточно большие мощности современных видеокарт без изучения особенностей низкоуровневого программирования данных устройств.

Одним из наибольших преимуществ данных систем является высокая распространенность (практически каждый современный персональный компьютер оснащен видеокартой, поддерживающей вычисления общего назначения) и низкая стоимость единицы вычислительной мощности по сравнению с, например, кластерными системами.

При построении современных вычислительных систем для высокопроизводительной обработки данных перспективным является совмещение возможностей кластерных систем и графических процессоров. Графические процессоры обеспечивают параллельную обработку данных по принципу SIMD с применением большого числа потоков инструкций (100, 1000 и более).

В то же время системы на основе GPU обладают относительно небольшим объемом памяти.

В свою очередь, кластер, состоящий из нескольких узлов, позволяет обеспечить независимую обработку данных с использованием практически неограниченного объема памяти, но с временными задержками при передаче данных с одного узла кластера на другой. Совмещение кластерной архитектуры и графических процессоров в каждом узле кластера позволяет воспользоваться преимуществами каждой архитектуры.

Проблемы построения универсальной кластерной системы на основе GPU. При организации универсальной кластерной системы на основе графических процессоров возникают следующие проблемы:

- обеспечить независимое функционирование каждого узла кластера при решении разных задач одновременно;
- обеспечить гибкое независимое конфигурирование архитектуры кластера для одновременного решения разных задач;
- обеспечить динамическое выделение и миграцию ресурсов системы между узлами кластера.

Указанные выше проблемы возникают в связи с тем, что при построении кластерной системы присутствует естественное ограничение по количеству имеющихся в наличии физических узлов. Организация кластера усложняется тем, что указанные ресурсы в разнообразных сочетаниях необходимы в один момент времени (кластер универсальный и должен позволять решать разные задачи независимо друг от друга).

Фактически необходимо отобразить векторы ресурсов каждой задачи на реальные физические ресурсы кластерной системы. Для выполнения данного условия можно предложить два способа организации каждого узла кластера.

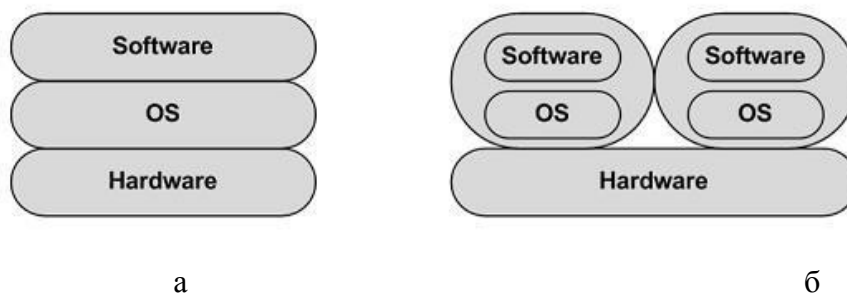


Рисунок 2 – Организация узла кластера

На рисунке 2а показан вариант организации узла, при котором физические ресурсы узла напрямую предоставлены операционной системе и прикладному программному обеспечению. Это самый простой способ, но в этом случае выполнение условия независимости задач становится трудновыполнимым, так как в рамках одной операционной системы практически невозможно организовать изолированный доступ разных задач к одним и тем же физическим ресурсам. При этом прикладное программное обеспечение и задачи могут требовать разные виды операционных систем, что также невыполнимо при такой организации.

На рисунке 2б показан альтернативный вариант организации узла кластера. Каждый узел представляет собой группу контейнеров, каждый из которых имеет доступ к одним и тем же физическим ресурсам, но в каждом контейнере организована изолированная среда для выполнения своей операционной системы и своего набора программного обеспечения. Такое решение называется виртуализацией, а контейнеры – виртуальными машинами.

При выборе платформы для виртуализации необходимо учесть, что виртуальная машина – это прослойка между аппаратным обеспечением и гостевой операционной системой. Поэтому при построении кластера нецелесообразно использовать платформу виртуализации, которая представлена в виде программного продукта, который запускается под управлением хост-операционной системой.

Возможности распараллеливания алгоритма построения спектра в кластерной системе на основе GPU. Разработанный алгоритм может быть очень эффективно распараллелен на архитектурах с использованием GPU ввиду следующих его характеристик:

- 1) При построении спектра по большей части используются эффективно вычисляемые на GPU линейные операции над векторами.
- 2) Вычисление отдельных строк матрицы спектра может быть произведено практически независимо.

3) Строки матрицы могут быть разбиты на блоки, также вычисляемые параллельно. Размер блока может варьироваться для достижения оптимальной производительности на различных конфигурациях аппаратного обеспечения.

4) Требуемые объем оперативной памяти, и производительность кластерной системы растут линейно в зависимости от роста размера матрицы спектра.

Таким образом, для достижения необходимой скорости построения спектра при заданном разрешении достаточно легко рассчитать требования и сконфигурировать вычислительный кластер с использованием GPU, решающий поставленную задачу.

REFERENCES

1 Bridson R., Houriham J., Nordenstam M. Curl-noise for procedural fluid flow. ACM Trans. Graph., 2007. 26(3):46.

2 Bridson R. Muller-Fischer M. Fluid simulation: Siggraph, 2007 course notes. In SIGGRAPH '07: ACM SIGGRAPH 2007 Courses, p. 1-81, New York, NY, USA, 2007. ACM.

3 Robert L. Cook and Tony De Rose. Wavelet noise. ACM Trans. Graph., 24(3):803-811, 2005.

4 Keenan Crane, Ignacio Llamas, and Sarah Tariq. Real-Time Simulation and Rendering of 3D Fluids, volume GPU Gems 3, pages 633-675. Addison-Wesley Professional, 2008.

5 Ronald Fedkiw, Jos Stam, and Henrik Wann Jensen. Visual simulation of smoke. In SIGGRAPH '01: Proceedings of the 28th annual conference on Computer graphics and interactive techniques, pages 15-22, New York, NY, USA, 2001. ACM.

6 Milan Ikits, Joe Kniss, Aaron Lefohn, and Charles Hansen. Volume Rendering Techniques, volume GPU Gems: Programming Techniques, Tips and Tricks for Real-Time Graphics, p. 667-692. Addison-Wesley Professional, 2004.

7 Joe Kniss, Simon Premoze, Charles Hansen, and David Ebert. Interactive translucent volume rendering and procedural modeling. In VIS '02: Proceedings of the conference on Visualization '02, p. 109-116, Washington, DC, USA, 2002. IEEE Computer Society.

Резюме

Н. Азімбаев¹, С. Кульдин², А. Пак⁴, В. Клименов³, А. Исова³,

И. Невмержицкий³, С. Тоқмолдин³

¹М. В. Ломоносов атындағы Мәскеу мемлекеттік университеті, Мәскеу, РФ,

²Мәскеу физика-техникалық университеті, Мәскеу, РФ,

³Физика-техникалық институты, Алматы, Қазақстан Республикасы,

⁴Уорик университеті, Ковентри, Ұлыбритания)

«NVIDIA CUDA» ТЕХНОЛОГИЯСЫ НЕГІЗІНДЕ ЛАЗЕРЛІ ГЕТЕРОДИНДІК
ДЕФОРМОГРАФТЫ ДЕРЕКТЕРІН ВИЗУАЛИЗАЦИЯЛАУ МЕН
ӨНДЕУ ЖҮЙЕСІН ПАРАЛЛЕЛЬСІЗДЕНДІРУ

Берілген жұмыста нақты уақыт режимінде лазерлік гетеродиндік деформографтың өңделген мәлімет-терінің спектрлік сипаттамасын есептеу мәселесін шешу үшін графикалық процессорларды (GPU) қолдану тиімділігіне сараптама жасау жүргізіледі. Жұмыстың маңыздылығы графикалық процессорларды заманауи есептегіш жүйелерге кең қолданысқа енгізу және қолданбалы бағдарламалық қамтамасыздандыруды жасау кезінде бұл процессорлардың ерекшеліктерін ескеру қажеттілігімен байланысты. Жұмыста қарастырылып отырған, өздерінің түйіндерінің құрамында әмбебап процессорлары да бар, графикалық процессорлары да бар гибриді кластерлік есептегіш жүйелер мәселесін шешу үшін біртекті емес жоғары өндіргішті есептегіш жүйелерді қолдану мүмкін перспективалардың бірі болып табылады. Мұндай жүйелер үшін тиімді қолдан-балы қамтамасыздандыруды жасап шығару қазіргі уақытта мәліметтердің үлкен көлемін өндеуге мүмкіндік береді, сонымен қатар жүргізілген есептеулердің дәлдігін арттырады.

Тірек сөздер: «NVIDIA CUDA» технологиясы, ақпараттарды өндеу, спектрлік талдау, лазерлі гетеродин-дік деформограф, параллельдік есептеу.

Summary

N. Azimbaev¹, S. Kuldin², A. Pak⁴, V. Klimenov³, A. Issova³, I. Nevmerjiskii³, S. Tokmoldin³

¹Lomonosov Moscow State University, Moscow, Russia,

²Moscow Institute of Physics and Technology State University, Moscow, Russia,

³Institute of Physics and Technology, Almaty, Republic of Kazakhstan,

⁴University of Warwick, Coventry, Great Britain)

PARALLELIZATION OF LASER HETERODYNE DEFORMOGRAPH'S VISUALIZATION
SYSTEM

AND DATA PROCESSING BASED ON «NVIDIA CUDA» TECHNOLOGY

The paper presents a conceptual analysis of the effectiveness of using graphics processing units (GPU) to solve actual problem of calculating the spectral characteristics of the processed data from the laser heterodyne defor-mograph in real time. The urgency of the problem is related to the widespread introduction of GPUs in modern computing systems and necessity to consider the features of these processors in the application software development process. The key prospects of the problem is to use the hybrid cluster computing systems having a composition of hosts both graphic and CPU. Developing effective application software for such systems currently allows a greater amount of information to process, hence it makes the accuracy of the calculations more precise.

Keywords: «NVIDIA CUDA» technology, data mining, spectral analysis, laser heterodyne deformograph, parallel computations.

Поступила 10.09.2013 г.