

Министерство образования и науки Российской Федерации
ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ОБРАЗОВАТЕЛЬНОЕ
УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ
«САРАТОВСКИЙ НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ
ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ ИМЕНИ Н.Г. ЧЕРНЫШЕВСКОГО»

Кафедра информатики и программирования

**ПРИМЕНЕНИЕ ГРАФИЧЕСКИХ УСКОРИТЕЛЕЙ NVIDIA ДЛЯ
РЕШЕНИЯ ЗАДАЧ ВЫЧИСЛИТЕЛЬНОЙ МАТЕМАТИКИ
АВТОРЕФЕРАТ БАКАЛАВРСКОЙ РАБОТЫ**

студента 4 курса 441 группы
направления 02.03.03 Математическое обеспечение и администрирование
информационных систем
факультета компьютерных наук и информационных технологий
Болдырева Дмитрия Александровича

Научный руководитель

доцент, к. ф.-м. н

К. П. Вахлаева

Зав. кафедрой

к.ф.-м.н.

М.В. Огнева

Саратов 2018

ВВЕДЕНИЕ

Актуальность темы. На протяжении последнего десятилетия графические ускорители стремительно наращивали производительность, чтобы удовлетворить непрерывно растущие запросы разработчиков графических приложений и компьютерных игр. Кроме того, за последние несколько лет изменились некоторые фундаментальные принципы разработки графической аппаратуры, в результате чего она стала более программируемой, чем когда-либо ранее. Сегодня графический ускоритель – это гибко программируемый массивно-параллельный процессор для обработки чисел в формате с плавающей точкой, возможности которого могут быть востребованы для решения целого ряда вычислительно трудоемких задач.

Одной из таких задач является задача N тел. Задача N тел, т.е. описание возможных движений N материальных точек, притягивающихся по закону Ньютона, является одной из основных фундаментальных проблем небесной механики и динамики. Роль ее в развитии математики и естествознания невозможно переоценить. За триста лет в этой проблеме получено немало результатов первостепенной важности; еще больше идей и результатов в смежных областях науки обязаны своим происхождением задаче N тел.

Как известно, гравитационная задача N тел не имеет явного аналитического решения уже при количестве тел, равных трём. Но с появлением компьютерной техники появилась реальная возможность изучать свойства систем гравитирующих тел путём численного решения системы уравнений движения. При этом возникла потребность ускорения моделирования задачи, так как при большом объёме входных данных скорость вычисления уменьшается в несколько раз.

Вышесказанное определило **цель бакалаврской работы**: применение графических ускорителей NVidia для эффективного решения задач, характеризующихся большой вычислительной сложностью, на примере расчета эволюции поведения системы из N тел.

Поставленная цель определила следующие *задачи*:

1. изучить математическую модель гравитационной задачи N тел и методы ее численного решения;
2. ознакомиться с основными принципами технологии NVidia CUDA;
3. реализовать параллельное решение гравитационной задачи N тел различными методами на центральном и графическом процессорах;
4. выполнить сравнение эффективности решения задачи для различных архитектур.

Практическая значимость бакалаврской работы. В ходе выполнения практической части бакалаврской работы были разработаны параллельный метод решения гравитационной задачи N тел с использованием технологии Nvidia CUDA и два последовательных метода, для того, чтобы показать, что Nvidia CUDA может использоваться для методов, не предназначенных для распараллеливания, и значительно ускорить вычисления.

Структура и объём работы. Бакалаврская работа состоит из введения, 4 разделов, заключения, списка использованных источников и 8 приложений. Общий объём работы – 86 страниц, из них 52 страницы – основное содержание, включая 20 рисунков и 6 таблиц, цифровой носитель в качестве приложения, список использованных источников информации – 21 наименование.

КРАТКОЕ СОДЕРЖАНИЕ РАБОТЫ

Первый раздел «Математическая формулировка задачи N тел» посвящён описанию математической формулировке задачи моделирования динамики системы N точечных масс, взаимодействующих по закону тяготения Ньютона. Раздел делится на подразделы «задача» и «решение».

В подразделе «задача» описывается постановка гравитационной задачи N тел.

В подразделе «решение» описывается нахождение системы уравнений N гравитирующих тел и формула нахождения ускорения, которое тело приобретает под действием силы гравитации в данной системе.

Второй раздел «Методы решения для задачи N тел» посвящён рассмотрению численных методов для решения гравитационной задачи N тел, описанию и решению проблем, связанных с численными методами, а также описанию алгоритмов для решения гравитационной задачи N тел. Раздел содержит несколько подразделов.

Подраздел «Простейший алгоритм» включает в себя описание метода «всех пар» моделирования задачи N тел, его особенностей и примеры более сложных схем, позволяющих сократить объем вычислений.

Подраздел «Алгоритм Барнса-Хата» содержит описание алгоритма, его вычислительной сложности, а также в данном подразделе описаны основные проблемы, которые встречаются при распараллеливании. Описание алгоритма в этом подразделе так же делится на два подраздела: «построение дерева Барнса-Хата» и «вычисление силы, действующей на тело».

В подразделе «построение дерева Барнса-Хата» описывается рекурсивный алгоритм для построения дерева Барнса-Хата, а в подразделе «вычисление силы, действующей на тело» описывается рекурсивный алгоритм для вычисления силы, действующей на тело.

Третий раздел «Технология NVIDIA CUDA» посвящён описанию преимуществ и особенностей архитектуры параллельных вычислений Nvidia CUDA, которая использует графические процессоры.

Раздел содержит 4 подраздела.

Подраздел «Состав NVIDIA CUDA» состоит из описания API и библиотек, которые содержит CUDA, описания уровней программно-аппаратного стека и библиотек.

Подраздел «Модель программирования CUDA» знакомит с представлением GPU в роли массивно-параллельного сопроцессора к CPU, с различиями между CPU и GPU, а также с особенностями работы потоков на GPU.

В подразделе «Типы памяти и скорость доступа» описываются особенности различных типов памяти в CUDA.

Подраздел «Расширения языка C для работы с CUDA» включает в себя описание особенностей языка CUDA C, который является расширенным языком C. Особенности являются спецификаторы функций и переменных, директивы вызова ядра и добавленные переменные.

Четвёртый раздел «Реализация разных методов для решения гравитационной задачи N тел с помощью Nvidia CUDA» посвящён реализации решения гравитационной задачи N тел различными методами на центральном и графическом процессорах и сравнение эффективности решения задачи для различных архитектур.

Раздел состоит из четырёх подразделов.

Подраздел «Последовательный метод всех пар» содержит в себе описание реализации последовательной версии алгоритма «всех пар» для решения гравитационной задачи N тел на центральном процессоре.

Подраздел «Последовательный метод Барнса-Хата» содержит в себе описание реализации последовательной версии алгоритма Барнса-Хата для решения гравитационной задачи N тел на центральном процессоре.

В подразделе «Параллельный вариант метода Барнса-Хата, реализованный с помощью технологии Nvidia CUDA» описывается реализация параллельной версии алгоритма Барнса-Хата на графическом процессоре при помощи технологии Nvidia CUDA, а также описываются применённые оптимизации для ускорения работы алгоритма. Моделирование системы N гравитирующих тел на основе алгоритма Барнса-Хата легче рассмотреть, если разбить его на этапы, поэтому подраздел разбивается на подраздел глобальных оптимизаций и подразделы описывающих этапы алгоритма.

Подраздел «Глобальные оптимизации» включает в себя описание оптимизаций, которые применимы для всех этапов на GPU.

Подраздел «Этап 1. Вычисление границ вокруг всех тел» включает в себя описание этапа, в котором вычисляется ограничивающий прямоугольник вокруг всех тел, и оптимизаций, применённых в его реализации.

Подраздел «Этап 2. Построение октодерева Барнса-Хата» включает в себя описание этапа, в котором реализуется итеративный алгоритм построения дерева, который иерархически подразделяет ячейку, полученную на этапе 1, пока не будет не более одного тела на самой внутренней ячейке, и описание оптимизаций, применённых в его реализации.

Подраздел «Этап 3. Вычисление центров тяжести» включает в себя описание этапа, в котором функция обходит несбалансированное октодерево с конца, чтобы вычислить центр тяжести каждой ячейки и суммарную массу всех содержащихся тел, и описание оптимизаций, применённых в его реализации.

Подраздел «Этап 4. Сортировка» включает в себя описание этапа, в котором вычисляется ограничивающий прямоугольник вокруг всех тел, и описание оптимизаций, применённых в его реализации.

Подраздел «Этап 5. Вычисление сил, действующих на каждое тело» включает в себя описание этапа, в котором вычисляются силы, действующие на каждое тело,

с помощью обхода октодерева и описание оптимизаций, применённых в его реализации.

Подраздел «Этап 6. Обновление положений и скоростей тел» включает в себя описание этапа, в котором обновляется скорость и положение каждого тела на основе вычисленной силы.

В подразделе «Результаты вычислительных экспериментов» описываются входные данные для реализованных решений, проводятся эксперименты для сравнения эффективности трёх реализованных методов, приводятся примеры работы приложений, реализующих три сравниваемых метода, и анализируются полученные результаты.

ЗАКЛЮЧЕНИЕ

В данной работе представлено несколько методов решения гравитационной задачи N тел. Самым эффективным из представленных, является параллельный вариант метода Барнса-Хата для решения гравитационной задачи N тел с использованием технологии Nvidia CUDA, который показывает, что графические процессоры могут применяться и для алгоритмов, которые не предназначены для распараллеливания, однако это требует многих оптимизаций.

В частности, чтобы ускорить вычисления, пришлось перепроектировать основную структуру данных, преобразовать весь рекурсивный код в итеративный код, явно управлять кешем (разделяемой памятью) и распараллеливать реализацию внутри блоков потоков. Также увеличение производительности произошло благодаря использованию некоторых уникальных архитектурных особенностей графических процессоров.

СПИСОК ИСПОЛЬЗОВАННОЙ ЛИТЕРАТУРЫ

1. General-Purpose Computation Using Graphics Hardware [Электронный ресурс]. URL: <http://www.gpgpu.org> (дата обращения: 12.03.2018)
2. Lars Nyland, Mark Harris, Jan Prins Fast N-Body Simulation with CUDA [Электронный ресурс]. URL: http://developer.download.nvidia.com/assets/cuda/files/nbody_gems3_ch31.pdf (дата обращения: 08.05.2018)
3. Elsen E. et al. N-Body Simulations on GPUs. // ACM/IEEE conference on Supercomputing 2006.
4. А.В. Адинец Анализ эффективности решения задачи N тел на различных вычислительных архитектурах [Электронный ресурс]. URL: <http://www.ict.edu.ru/vconf/files/11859.pdf> (дата обращения: 02.05.2018)
5. Д.К. Боголепов, В.Е. Турлапов. Вычисления общего назначения на графических процессорах с использованием шейдерных языков. [Электронный ресурс]. URL: <http://www.ict.edu.ru/vconf/files/11899.pdf> (дата обращения: 08.05.2018)
6. Параллельные методы решения гравитационной задачи n тел [Электронный ресурс]. URL: http://edu.mmcs.sfedu.ru/pluginfile.php/5634/mod_resource/content/3/OMPMPPI_Gravit2016.pdf (дата обращения: 02.05.2018)
7. Aarseth, S. 2003. Gravitational N-Body Simulations. Cambridge University Press.
8. Makino J, Aarseth S. J. On a Hermite integrator with Ahmad-Cohen scheme for gravitational many-body problems // PASJ: Publications of the Astronomical Society of Japan, 1992, T. 44, pp. 141-151.
9. Barnes J., Hut P. A hierarchical $O(N \log N)$ force-calculation algorithm. 04.10.1986 г., Nature, T. 324, pp. 446 - 449.

10. Michele Trenti, Piet Hut. Gravitational N-body Simulations. [Электронный ресурс]. URL: <https://arxiv.org/pdf/0806.3950.pdf> (дата обращения: 08.05.2018)
11. Tsuyoshi Hamada. etc, A novel multiple-walk parallel algorithm for the Barnes–Hut treecode on GPUs – towards cost effective, high performance N-body simulation, 24 (2009) 21–31, SpringerLink.
12. Tom Ventimiglia, Kevin Wayne Barnes-Hut Galaxy Simulator [Электронный ресурс]. URL: <http://www.cs.princeton.edu/courses/archive/fall03/cs126/assignments/barnes-hut.html> (дата обращения: 20.05.2018)
13. Tom Ventimiglia, Kevin Wayne The Barnes-Hut Algorithm [Электронный ресурс]. URL: <http://arborjs.org/docs/barnes-hut> (дата обращения: 20.05.2018)
14. Параллельные вычисления с CUDA [Электронный ресурс]. URL: <http://www.nvidia.ru/object/cuda-parallel-computing-ru.html> (дата обращения 23.04.2018).
15. Денисенко М.В., Муняев В.О., Сатанин А.М. Применение распределенных вычислительных систем и технологии CUDA для моделирования физических процессов. Нижегородский госуниверситет, 2014. – 85 с.
16. Андреев А.Е., Шаповалов О.В. Технология Nvidia CUDA (Л/Р) 2009 [Электронный ресурс]. URL: <http://fevt.ru/load/cuda/47-1-0-162> (дата обращения 23.04.2018).
17. Боресков А.В., Харламов А.А. Основы работы с технологией CUDA. М.: ДМК Пресс, 2010. 232 с.
18. Nvidia CUDA Programming Guide v1.1. [http://developer.download.nvidia.com/compute/cuda/1_1/NVIDIA_CUDA_Programming_Guide_1.1.pdf]
19. GPU computing gems / editor Wen-mei W. Hwu. P. cm Includes bibliographical references.

20. J. Barnes, A modified tree code: don't laugh; it runs, *J. Comput. Phys.* 87 (1) (1990) 161–170
21. H.C. Plummer, On the problem of distribution in globular star clusters, *Mon. Not. R. Astron. Soc.* 71 (1911) 460–470