МОСКОВСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ

имени М.В. Ломоносова

Биологический факультет МГУ

Москва, Ленинские горы, д.1, строение 12

24-25 мая 2010 года

Научно-практическая конференция

«Вычисления с использованием графических процессоров в молекулярной биологии и биоинформатике»

ТЕЗИСЫ ДОКЛАДОВ

Москва 2010

МОСКОВСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ

имени М.В. ЛОМОНОСОВА

БИОЛОГИЧЕСКИЙ ФАКУЛЬТЕТ

Москва, Ленинские горы, д.1, строение 12

24-25 мая 2010 года

Научно-практическая конференция

«Вычисления с использованием графических процессоров в молекулярной биологии и биоинформатике»

Организационный комитет

Со-председатели:
А.Р. Хохлов — проректор МГУ, академик РАН;
М.П. Кирпичников — декан Биологического факультета МГУ, академик РАН; А.В. Старовойтов — директор ФГНУ ЦИТИС
Ученый секретарь: А.В. Лисица
Члены оргкомитета: А.И. Арчаков, В.В. Воеводин, Р.З. Сагдеев, К.В. Шайтан, А.В.Чертович

I – ТЕХНОЛОГИЯ И ВЫЧИСЛИТЕЛЬНЫЕ МЕТОДЫ

Тенденции развития архитектуры масштабируемых суперЭВМ (1982-2010г.г.)

Галушкин А.И.

ФГНУ Центр информационных технологий и систем органов исполнительной власти (ФГНУ ЦИТиС), Международный центр по информатике и электронике (ИнтерЭВМ), neurocomputer@yandex.ru

В отличие от классических одно- и мало процессорных ЭВМ в начале 80х годов прошлого столетия возникло новое направление вычислительной техники – транспьютерные системы. Это были первые представители масштабируемых вычислительных систем, в которых производительность монотонно зависела от стоимости, а система в зависимости от выполняемых функций и стоимости могла конфигурироваться пользователем. В период с 1988 по 1995 годы нашим коллективом было разработано около 10 типов

масштабируемых высокопроизводительных вычислительных транспьютерных систем, поставлено промышленным и научным организациям около 150 систем.

В их числе были 64-процессорные кластерные системы высокой производительности разработки 1992 г. Однако короткий период эксплуатации кластерных конфигураций, представляющих собой множество транспьютеров, соединенных коммутационной системой, показал их слабую эффективность при решении значительного класса сложных задач. Именно тогда началась разработка так называемых масштабируемых вычислительных систем с двухслойной архитектурой, где первый слой обработки представляет собой сеть транспьютеров, как процессоров с фон-неймановской архитектурой, а второй слой представлял собой сеть специализированных процессоров (типа JMSA100, JMSA110), берущих на себя основную нагрузку при решении сложных задач.

В целом в этом случае реализовывалась архитектура MSIMD (множественная архитектура: один поток команд – много потоков данных). Однако уровень технологии микроэлектроники (1,5-1,2 мк) начала 90х годов не позволял реализовывать второй слой обработки как универсальный. Были реализованы специализированные процессоры в виде одномерного и двумерного Z-фильтров с перестраиваемыми в процессе решения задач коэффициентами.

Транспьютерные системы нашли широкое применение во всем мире. В период с 1991 по 1996 гг. прилагались значительные усилия по формированию рынка супервычислений в России. Для классических кластерных транспьютерных систем и систем обработки сигналов и изображений этот рынок был весьма узок. Одновременно в период с 1988 по 2005 гг. была предпринята активная попытка формирования в России рынка нейросетевых технологий. Эта попытка фактически также закончилась неудачей. Основной причиной этого мы считаем объективное отсутствие в России в этот период необходимости в решении сложных задач, связанной с отсутствием развития высоких технологий в промышленности. Именно по этим причинам после активного участия в реализации транспьютерных систем в течение почти 10 лет (1996-2005 гг.) наш коллектив не проявлял интереса к разработкам классических кластерных суперЭВМ.

Начиная с 2005 г. развитие технологии микроэлектроники и связанное с этим резкое увеличение числа элементов в кристалле позволило создать образцы графических процессоров, которые в процессе своего развития и развития сопровождающего программного обеспечения стали поддерживать вычислительные операции, позволяющие решать более широкий круг задач. При этом наряду с развитием многоядерных процессоров с малым (несколько десятков) числом ядер были разработаны процессоры с несколькими сотнями более простых ядер, аналогично тому, как это было сделано в процессорах JMSA100 и JMSA110.

В период с 2005 по 2007 гг. наш коллектив одним из первых в России реализовал 16-узловой супервычислительный комплекс на базе процессоров NVIDIA и Pentium, а также провел оценку отношения производительности к стоимости при решении отдельных задач. С нашей точки зрения, с учетом опыта формирования рынка транспьютерных систем, нейросетевых технологий и классических кластерных суперЭВМ в России рынок супервычислений в настоящее время нужно формировать производством персональных суперЭВМ с производительностью от 1 до 10 терафлоп с ориентацией на задачи

- нефтегазовой отрасли;

- аэро-, гидро-, газо-, электродинамики;

- обработки больших массивов текстовой информации;

- интегрированных систем видеонаблюдения;

- задач программы «Протеом человека».

Этот круг задач для подобных персональных суперЭВМ будет в ближайшее время резко расширяться.

Литература:

1. Галушкин А.И. Некоторые исторические аспекты развития элементной базы вычислительных систем с массовым параллелизмом (80е-90е годы), «Нейрокомпьютер», №1, 2000 г.
2. Галушкин А.И. Транспьютерные системы – начало становления в России ЭВМ с массовым параллелизмом, «Нейрокомпьютер», №3, 2005 г.
3. Аляутдинов М.А., Галушкин А.И., Казанцев П.А., Остапенко Г.П. Нейрокомпьютеры: от программной реализации к аппаратной, «Горячая линия – ТЕЛЕКОМ», 2008 г.
4. Аляутдинов М.А., Галушкин А.И., Троепольская Г.В. Перспективные средства построения высокопроизводительных масштабируемых вычислительных систем. «Интеллектуальные информационные системы», №1, 2008 г.

UGENE – интегрированные решения на GPU для задач вычислительной биологии.

Ефремов И. Е.

ООО «НЦИТ Унипро», г. Новосибирск

iefremov@unipro.ru

Развитие перспективных вычислительных платформ и архитектур, наряду с экспоненциальным ростом объемов экспериментальных данных молекулярной биологии является катализатором для появления новых, все более мощных, программных средств анализа. При этом на первый план выходит способность программного средства использовать все доступные вычислительные ресурсы: многоядерные процессоры, векторные расширения, графические ускорители. На сегодняшний день, GPU являются одной из важнейших аппаратных платформ, за счет сверхвысокой номинальной производительности.

Современная система анализа генетической информации Unipro UGENE является новой программной платформой для работы молекулярного биолога, при создании которой ставилась цель интеграции наиболее важных вычислительных алгоритмов и их оптимизированных реализаций в единую визуальную среду.

При создании оптимизированных решений для вычислителей специального назначения, таких как графические процессоры, важнейшими условиями для промышленного использования ПО являются стабильность и корректность использования доступных устройств. В процессе работы UGENE автоматически определяет доступное аппаратное обеспечение и его характеристики (как то, объем памяти устройства, количество мультипроцессоров и т. д.) Исходя из этих данных, реализации алгоритмов могут выбирать оптимальные параметры исполнения. UGENE осуществляет разделение доступа к GPU, а также позволяет пользователю выбрать доступные устройства – например,

запретить вычисления на графическом процессоре, который используется операционной системой.

Пилотный оптимизированный для работы на GPU алгоритм, уже несколько месяцев распространяемый в составе UGENE – это локальное выравнивание геномных последовательностей по методу Смита-Ватермана. Код, выполняющийся на бюджетном игровом графическом ускорителе до пяти раз более производителен, чем однопоточная версия для центрального процессора, и до полутора раз – по сравнению с параллельным исполнением на четырехъядерном процессоре.

Типичные примеры решаемых в UGENE задач, перспективных для реализации на GPU – это выравнивание последовательностей (BLAST[1], HMMER[2]), ассемблирование геномов, поиск повторов (Bowtie[3]) . Мощный визуальный интерфейс, включающий в себя возможность конструирования составных схем анализа из доступных алгоритмов, а также кросс-платформенность – дополнительные преимущества UGENE. Продукт распространяется с открытым исходным кодом на условиях GPL.

[1] Altschul S.F., et al. (1990) J Mol Biol. 5;215(3):403-10

[2] Durbin R., et al. (1998). Biological Sequence Analysis: Probabilistic Models of Proteins and Nucleic Acids. Cambridge University Press, 356 p.

[3] Langmead B, Trapnell C, Pop M, Salzberg SL. Ultrafast and memory-efficient alignment of short DNA sequences to the human genome. Genome Biol 10:R25.

Гибридные вычислительные системы и их применение для ысокопроизводительных вычислений в области биологии

Уральский Ю., Джораев А.

компания NVIDIA, г. Москва

До последнего времени ключевым компонентом систем для высокопроизводительных вычислений, включая кластеры, был центральный процессор. Однако несколько лет назад появилась тенденция строить гибридные вычислительные системы, применяя графические процессора (GPU).

Высокая производительность GPU объясняется особенностями его архитектуры. В отличие от центрального процессора, который состоит из нескольких ядер, графический процессор изначально создавался как многоядерная структура, в которой количество ядер измеряется сотнями. Также есть существенная разница и в принципах работы - архитектура CPU предполагает последовательную обработку информации, а GPU исторически предназначался для обработки компьютерной графики, поэтому рассчитан на массивно параллельные вычисления.

Каждая из этих двух архитектур имеет свои достоинства. CPU лучше работает с последовательными задачами, но при большом объеме обрабатываемой информации, и при возможности распараллеливания алгоритмов - очевидное преимущество имеет GPU. Кроме того пиковая производительность современного GPU в разы выше чем у современного CPU.

Собственно параллельная архитектура GPU и их высокая производительность обеспечивают возможность применения гибридных систем на их основе для решения различных классов научных и прикладных вычислительных задач, включая задачи био-информатики, генетики, молекулярной динамики и др.

Главным условием применения таких системы является наличие ПО которое их поддерживает. Компания NVIDIA работает со всеми основными разработчиками прикладного ПО в данных отраслях, оказывая содействие в подготовке программного кода и переносе его на GPU. На сегодняшний день большинство из этих пакетов уже поддерживают GPU.

Источником информации по данной тематике является портал Tesla Bio Workbench, который постоянно обновляется и дополняется новыми материалами, включая и сами приложения.

Моделирование на GPU стохастического клеточного автомата

Смирнов В.Л., Перепелкин Е.Е., Поволоцкий А.М.

НОЦ «Параллельные вычисления», ОИЯИ, г. Дубна

В данной работе с помощью метода Монте-Карло производилось моделирование асимметричного процесса с исключениями. Задачи с клеточными автоматами такого типа требуют больших временных затрат. На практике мы имеем систему с большим числом степеней свободы, которая совершает большое число однотипных операций. Повысить эффективность таких расчетов позволяют методы параллельного программирования. В работе представлены вычисления, выполненные с помощью графических процессоров на базе NVIDIA TESLA C1060. Использование такого подхода существенно ускоряет вычисления, позволяя работать с системами большой размерности. Получена универсальная скейлинговая функция, характеризующая системы, относящиеся к классу универсальности Кардара-Паризи-Жанга.

Использование технологии nVidia CUDA для вычисления меры многочастичной квантовой запутанности

Чернявский А. Ю.

Физико-технологический институт РАН (ФТИАН)

Факультет ВМК МГУ им М.В. Ломоносова

andrey.chernyavskiy@gmail.com

Многие задачи квантовой информатики, особенно связанные с теорией многочастичной квантовой запутанности, представляют собой задачи глобальной многопараметрической оптимизации. В последнее время для эффективного решения таких задач используются генетические или роевые оптимизационные методы, требующие большого количества вычислений целевой функции, но в тоже время поддающиеся легкому распараллеливанию. В связи с этими двумя факторами для подобных вычислений целесообразно использовать современные графические адаптеры.

В работе рассматривается задача вычисления одной меры квантовой запутанности (минимума энтропии измерений) многокубитных квантовых состояний, а именно: для квантового многокубитного состояния

$$|\psi\rangle = \sum_{i_1,i_2,\mathrm{K},i_n=0}^{1} a_{i_1 i_2 \mathrm{K}\, i_n} \, |i_1 i_2 \mathrm{K}\; i_n\rangle$$

вычислить

$$E_{Hmin}(|\psi\rangle) = \min_{U_1,U_2,\mathrm{K},U_n} H(U_1 \otimes U_2 \otimes \mathrm{K} \otimes U_n \,|\psi\rangle),$$

где

$$H(|\psi\rangle) = \sum_{i_1,i_2,\mathrm{K},i_n=0}^{1} \left|a_{i_1,i_2,\mathrm{K},i_n}\right|^2 \ln\left|a_{i_1,i_2,\mathrm{K},i_n}\right|^2,$$

$U_1, U_2, \mathrm{K}\;, U_n$ –

однокубитные преобразования.

На графическом адаптере nVidia GTX 275 удалось получить 14х кратное ускорение относительно четырехядерного процессора Intel Q6600. Невозможность получить максимальное ускорение связана в основном с использованием на CPU более эффективного алгоритма, требующего большего объема памяти, нежели размер разделяемой памяти GPU.

Также рассматриваются другие важные задачи квантовой информатики, требующие высокопроизводительных вычислений и имеющие возможности вычисления на графических адаптерах.

1. A. Yu. Chernyavskiy, “The computable measure of quantum entanglement of multiqubit states”, Mikroelektronika, 2009, Vol. 38, No.3, pp. 217–223
2. A. Y. Chernyavskiy, “Multiparticle Entanglement measure for multipartite pure states and its numerical calculation”, Arxiv preprint arXiv:0905.0201.

Компьютерный дизайн лекарственных веществ.

Н.С.Зефиров, В.А.Палюлин

Химический факультет МГУ имени М.В.Ломоносова

Методы виртуального скрининга, молекулярного моделирования и молекулярной динамики играют важнейшую роль на начальных этапах создания новых лекарственных средств при поиске и оптимизации соединений-лидеров. Комплексное их применение в сочетании с методами исследования количественной связи между структурой соединений и их активностью (QSAR) позволяет значительно сократить временные затраты на отбор

соединений-кандидатов на предклинические испытания. Однако для наиболее успешного применения таких методов требуются значительные компьютерные ресурсы и программное обеспечение, надежно работающее в параллельном режиме на суперкомпьютерах.

В докладе рассматривается роль современных методов молекулярного моделирования, молекулярной динамики, виртуального скрининга и QSAR на примере создания нейропротекторных препаратов. Обсуждается построение молекулярных моделей важнейших биомишеней нейропротекторных препаратов и их комплексов с агонистами, антагонистами, блокаторами ионных каналов и модуляторами. Рассматривается проблема компьютерного исследования влияния структурных модификаций лигандов на их взаимодействие с определенным типом рецепторов и создания веществ, избирательно и эффективно воздействующих только на заданный тип или подтип рецептора.

Приводится анализ необходимых компьютерных ресурсов как для различных режимов виртуального скрининга баз данных, включающих миллионы соединений, так и для моделирования молекулярной динамики белковых молекул-биомишеней с лекарственными веществами.

Аналитические методы и нейросетевые технологии в решении задач по программе «Протеом человека»

Галушкин А.И.

ФГНУ Центр информационных технологий и систем органов исполнительной власти (ФГНУ ЦИТиС), Международный центр по информатике и электронике (ИнтерЭВМ), neurocomputer@yandex.ru

Целью данной работы является создание аналитического ядра информационно-аналитической системы по программе «Протеом человека».

Основными предпосылками роста числа необходимых для решения задач в современных и перспективных информационно-аналитических системах являются следующие:

- резкое увеличение объемов обрабатываемой информации;

- расширение множества типов обрабатываемой информации (параметрическая, текстовая, мультимедийная);

- появление сложных формализуемых и неформализуемых аналитических задач;

- повышение требований к качеству решения аналитических задач.

Проведен анализ более 30 известных на рынке информационно-аналитических систем с целью определения состава решаемых задач и качества их решения. Подобный анализ существующих систем и возможных аналитических задач по программе «Протеом человека» позволит представить структуру аналитического ядра в следующем виде:

Технологии извлечения знаний (data mining) содержит следующие разделы:

- Анализ и прогнозирование временных рядов;

- Классификация или распознавание негативных объектов;

- Кластеризация;

- Заполнение неполных таблиц;

- Задачи оптимизации;

- Построение экспертных систем;

- Поиск ассоциативных связей.

Разработан вариант системы автоматического мониторинга информационных ресурсов сети Интернет и с привлечением экспертов-аналитиков, реализующей следующие технологии:

- технология динамического эволюционного проектирования баз данных;

- Технологии метапоиска информации в сети Интернет на основе нейросетевой экспертной системы;

- Технология сбор текстовых данных из выделенных ресурсов информационного массива Интернет и анализа смысла текста на основе семантических сетей с применением нейросетевых методов.

Практический опыт работ ЦИТиС касается следующих систем обработки мультимедийной информации:

- системы фильтрации изображений;

- системы сегментации изображений;

- системы сжатия изображений;

- системы выделения на изображениях элементов заданной формы;

- системы скелетонизации изображений;

- системы выделения на изображениях движущихся объектов;

- системы оценки качества изображений.

Практическое применение нашли следующие технологии построения моделей:

- сценарные модели;

- модели, описываемые системой дифференциальных уравнений в частных производных;

- модели, описываемые системой дифференциальных уравнений в частных производных;

- нейросетевые модели сосредоточенных и распределенных динамических систем.

В системах, разрабатываемых ЦИТиС, представлены следующие средства визуализации:

- большие составные экраны;

- локальные и составные без очковые средства 3D и стереовизуализации;

- персональные и коллективные средства визуализации с применением специальных очков.

О переработке пакетов программ обработки данных по сборке генома по методике Шотган для портирования на CUDA.

Алексеев Д.Г. Воронков И.

НИИ ФХМ

+79168139913 exappeal@gmail.com

Современные подходы к сборке и аннотации геномов методами Шотган предполагают достаточно мощную вычислительную систему использующую пакет программ для обработки первичных данных и последуюшей сборки контигов и скффолдов генома. На примере реального генома спироплазмы мы рассматриваем действителные сложности возникаюшие при работе с сушествуюшей парадигмой сборки геномов методом Шотган и демонстрируем ограниченность метода при любом варианте рассчетов в силу ограничений самой методики уже для геномов размером несколько млн. п.о. Так же мы предлагаем изменить парадигму от метода поиска точного значения генома к перечислению всех наиболее вероятных вариантов. При таком подходе многократно (до 100 раз) возрастает нагрузка на вычеслительные мощности лаборатории. Оптимальным решением для устранения узкого места - процесса вычисления - в потоке сборки генома, возникающего при таком подходе, является использование параллельных вычислений с помощью графических процессоров. Нами был проанализирован пакет стандартных программ на предмет возможности перевода их на вычисления с использованием графических процессоров. Мы предлагаем нижеследующий набор программ составляющий полный цикл сборки генома:

Phred -- http://www.phrap.org – универсальный инструмент преобразования файлов со спектрами сиквенсов в последовательности и соотнесенные с ними файлы качества (вероятности ошибок при чтении каждого основания).

Lucy -- http://lucy.sourceforge.net/ -- разработан в The Institute for Genomic Research используется для обработки файлов сиквенсов и очистки их от участков плохого качества, и т.н. загрязненных участков (последовательностей экспрессионных векторов использованных при создании биологической библиотеки). Вместе с предыдущей программой используется на каждой стадии проекта – для оперативного контроля получаемой информации.

Tigr -- http://www.jcvi.org/cms/research/software/ -- ассемблер длинных участков генома на основе перекрытий коротких ридов, работает с файлами сиквенсов и файлами качества. Результатом работы являются файлы с контигами – собранными участками генома, а так же информация о потенциальных найденных повторах при сборке.

Mira -- http://www.chevreux.org/projects_mira.html -- альтернативный ассемблер, позволяет делать гибридные сборки – на основе данных от разных типов сиеквенаторов (особенно удачно использует данные сиквенаторов нового поколения – опция не доступная в tigr). Более продуманный интерфейс (в т.ч. графический). Результаты сборки отличаются от tigr.

Bambus -- http://sourceforge.net/apps/mediawiki/amos/index.php?title=Bambus -- инструмент для построения скаффолдов из информации о контигах и парных прочтениях ридов. Скаффолды указывают на последовательность следования друг за другом различных контигов – помогает ориентироваться при необходимости чтения генома с помощью праймеров. Используется на стадии финализации проекта генома.

GPU-алгоритм восстановления поверхности из облака точек

Козлов Д. А., Турлапов В. Е.
Нижегородский Государственный Университет им. Н.И. Лобачевского

Введение

В работе рассматривается задача восстановления поверхности из облака точек, полученных с 3D сканера. В основе исследования лежит алгоритм, приведенный в статье [4]. Алгоритм был существенным образом переработан для эффективного исполнения на архитектурах с массовым параллелизмом, разработаны структуры хранения данных, рассчитанные на такие архитектуры.

Краткий обзор алгоритма

Алгоритм состоит из двух основных стадий. Первая стадия: построение неявной функции $\mathbf{F}: \mathbf{D} \to \mathbb{R}$, где $\mathbf{D} \subset \mathbb{R}^3$ - окрестность поверхности. Функция $\mathbf{F}$ представляет собой оценку расстояния от любой точки из D до поверхности. Вторая стадия: триангуляция поверхности нулевого уровня функции F при помощи алгоритма марширующих кубов [6]. Для построения оценки расстояния мы, следуя [4], ассоциируем с каждой точкой набора $\{x_1, \ldots, x_n\}$ ориентированную касательную плоскость $Tp(x_i)$, которую можно найти при помощи метода главных компонент, примененного к k ближайшим соседям точки x_i. Вычисление функции расстояния осуществляется по формуле: $\mathbf{F(p)} = (\mathbf{p} - \mathbf{o_i})\mathbf{n_i}$, где $\mathbf{o_i}$ – ближайший к $\mathbf{p}$ «центр» касательной плоскости (центроид k ближайших соседей), а $\mathbf{n_i}$– соотвестсвующая ему нормаль.

В параллельном алгоритме для GPU применена ускоряющая структура в виде регулярной сетки, реализованная на GPU следующим образом:

Листинг 1 Параллельное построение регулярной сетки

```
1: BBox = new bbox                    // Шаг 1: Построение bounding box'a
2:Reduce(Q,BBox, UnionOperator)       // Шаг 2: Вычисление идентификаторов точек
```

```
3: Pid = new array [N]
4: for each i = 0 to N-1 in parallel
5:        pid[i] = VoxelId(Q[i]) << 32 + i
6: SortedPid = new array [N]                          // Шаг 3: Сортировка идентификаторов точек
7: Sort(Pid,SortedPid,N)
8: SQ  = new array[N]                                 // Шаг 4: Переупорядочивание точек
9: for each i = 0 to N-1 in parallel
10:       SQ[i] = Q[SortedPid[i]]
11:Marks = new array [ N + 1]
12: for each i = 1 to N-1 in parallel          // Шаг 5: Выделение вокселей сетки
13:       if ( (SortedPid[i]  >> 32) == (SortedPid[i-1]>>32)
14:                 Marks[i] = 1
15:       else Marks[i] = 0
16: Scan(Marks, Marks, +)
17: Voxels = new array [Marks[N]]
18: Compact(Voxels, Marks, SQ)
```

В целях экономии памяти входные данные переупорядочиваются таким образом, чтобы точки, попадающие в один воксель сетки, лежали в видеопамяти последовательно, что позволяет сохранять лишь индекс начальной точки и количество точек, используя константный объем памяти для каждого вокселя. Первичным ключом сортировки является идентификатор вокселя, вычисляемый как индекс в трехмерном массиве, вторичным – индекс точки в начальном массиве. Для сортировки точек используется параллельная поразрядная сортировка (radix sort). Алгоритм построения широко использует шаблоны параллельного программирования, описанные в [3]. Стоит отметить, что в памяти сохраняются только те воксели, которые содержат точки данных, что позволяет значительно снизить использование видеопамяти.

В рамках подзадачи построения функции $F(p)$ реализовано ядро OpenCL, использующее построенную ускоряющую структуру для нахождения ближайшего “центра” касательной плоскости и вычисления оценки расстояния. Алгоритм марширующих кубов так же полностью выполняется на GPU и состоит в расчете значения функции в узлах регулярной сетки и последующей генерации треугольников для каждой ячейки этой сетки на основании шаблона. Алгоритм обрабатывает только те воксели, которые содержат точки данных.

Вычислительные эксперименты

Вычислительные эксперименты проводились при использовании сетки разрешением 256x256x256 на следующей тестовой конфигурации: Intel Core i7 975 Extreme, 6Gb RAM, ATI Radeon HD5870 1Gb RAM, Windows 7 x64, ATI Stream SDK v 2.01

Модель	**Кол-во точек**	**Кол-во обработанных ячеек сетки**	**Время, мсек**			
			Построение сетки	**Рассчет F**	**Триангуляция**	**Общее**
Sphere	**65536**	**52502**	**12**	**6**	**3**	**21**
Sphere	**131072**	**90208**	**26**	**11**	**11**	**48**
Bunny	**353272**	**178714**	**40**	**140**	**20**	**200**

Заключение

Разработан параллельный алгоритм восстановления поверхности из облака точек для GPU, реализована эффективная структура хранения данных, разработаны эффективные

алгоритмы работы с ней, а так же предложена архитектура системы для реализации в рамках технологии гетерогенных вычислений OpenCL. Результаты работы предполагается использовать для построения алгоритмов аппроксимации восстановленной поверхности функциями более высоких порядков. Алгоритм может быть использован для восстановления поверхности в реальном времени.

Список литературы

[1] Alexa, M., Behr, J., Cohen-Or, D., Fleishman, S., Levin, D., And Silva, C. T. 2001. Point set surfaces. In Proceedings of IEEE Visualization'01, 21–28.

[2] Amenta, N., Bern, M., and Kamvysselis, M. 1998. A new Voronoi-based surface reconstruction algorithm. In Proceedings of SIG-GRAPH'98, 415–421.

[3] Harris, M., Sengupta, S., and Owens, J. D. 2007. Parallel prefix sum (scan) with CUDA. In GPU Gems 3, H. Nguyen, Ed. Addison Wesley, Upper Saddle River.

[4] Hoppe, H., Derose, T., Duchamp, T., Mcdonald, J., and Stuetzle, W. 1992. Surface reconstruction from unorganized points. In Proceedings of SIGGRAPH'92, 71–78.

[5] Kazhdan, M., Bolitho, M., and Hoppe, H. 2006. Poisson surface reconstruction. In Proceedings of SGP'06, 61–70

[6] Lorensen, W. E., And Cline, H. E. 1987. Marching cubes: A highresolution 3d surface construction algorithm. In Proceedings of SIG-GRAPH'87, 163–169.

[7] Ohtake, Y., Belyaev, A., Alexa, M., Turk, G., And Seidel, H.-P.2004. Multi-level partition of unity implicits. ACM Trans. Graph. 22, 3, 463–470.

[8] Turk, G., and O'Brien, J. F. 2002. Modelling with implicit surfaces that interpolate. ACM Trans. Graph. 21, 4, 855–873.

[9] Yusov, E., Turlapov, V. 2008. JPEG2000-based compressed multiresolution model for real-time large scale terrain visualization. В кн.: Conference Proceedings of the 18th international Conference on Computer Graphics and Vision "GraphiCon'2007", 2008. С. 164—171

Нейросетевые технологии и их роль в программе «Протеом человека»

Галушкин А.И.

ФГНУ Центр информационных технологий и систем органов исполнительной власти (ФГНУ ЦИТиС), Международный центр по информатике и электронике (ИнтерЭВМ), neurocomputer@yandex.ru

Основными предпосылками применения нейросетевых технологий в программе «Протеом человека» являются следующие:

- резкое увеличение объемов обрабатываемой информации;

- появление сложных формализуемых, слабоформализуемых и неформализуемых задач;

- повышение требований к качеству решения аналитических задач.

Нейросетевые технологии весьма активно используются в геномике и протеомике и синтезе лекарственных соединений уже в течение 20 лет [2,3,4,5]

Необходимо также отметить достаточно широкое применение нейросетевых технологий в разделах вычислительной химии [1]:

- химической информатике (управлении химическими базами данных;

- молекулярном дизайне (поиск химического соединения с заданными свойствами);

- молекулярной механике (поиск наиболее выгодного пространственного расположения атомов в молекуле);

- молекулярной динамике (методы моделирования движения молекул);

- квантовой химии (расчет молекул методами квантовой механики путем решения волнового уравнения Шредингера).

Применение нейросетевых технологий в рассматриваемой области знаний и интеграция результатов привели в настоящее время к формированию нового направления – Computational Neurogenetic [8]. Аналитические исследования в информационно-аналитических системах ведутся в следующих направлениях:

- извлечение знаний;

- обработка больших массивов текстовой информации;

- обработка мультимедийной информации;

- построение моделей.

В докладе представлена роль нейросетевых технологий в решении указанных аналитических задач.

Приведены примеры применения нейросетевых технологий в криптографических задачах по постановке близких к задаче расшифровки генома, в обработке больших массивов текстовой информации и синтезе лекарственных средств.

Описываются основные зарубежные пакеты программ моделирования нейронных сетей для вычислительных систем на базе процессоров NVIDIA. Подобный отечественный пакет представлен отдельным докладом представителя нашей школы.

1. Баскин И.И., Палюлин В.А., Зефиров Н.С. Нейроматематика – будущее вычислительной химии, «Нейрокомпьютер», №3,4 1997 г.
2. Wei Zhang, Ilya Shmulevich (ed) Computational and Statistical Approaches to Genomics, Kluwer Academic Publishers, 1999
3. C.H. Wu, J.W. McLarly, Neural Networks and Genom Informatics, Elsevier, 2000
4. Jure Zupan, Johan Gasteiger Neural Networks in Chemistry and Drug Design, Wiley-VCH, 1999
5. Галушкин А.И. Нейрокомпьютеры в Китае на рубеже тысячелетий, Москва, Издательство «Горячая линия – ТЕЛЕКОМ», 2004 г.
6. Galushkin A.I. Neural Network Theory, Springer, 2007
7. Галушкин А.И. Нейронные сети: основы теории, Москва, Издательство «Горячая линия – Телеком», 2010 г.
8. N. Kasabov, Computational Neurogenetic, IJCNN, 2008.

Платформа создания комплексов нечетких моделей для интеллектуальной поддержки принятия решений с использованием технологий NVIDIA

В.В. Борисов[1], М.М.Зернов[2], А.Г. Забурдаев[3]

[1] vborisov@etna-it.ru, [2] zmmioml@yandex.ru, [3]azaburdaev@etna-it.ru

В первой части доклада представлены основные результаты исследований в области создания нечетких методов и моделей формирования и обработки знаний для интеллектуальной поддержки принятия решений [1–10].

Приведен обзор нечетких моделей и методов формирования и обработки знаний для интеллектуальной поддержки принятия решений (ИППР). Предложена классификация типов знаний в нечетких моделях (НМ) по следующим признакам: предмету знания; характеру неопределенности; способу получения; временным характеристикам. Предложена классификация нечетких моделей, применяемых для ИППР, рассмотрены используемые методы формирования и обработки знаний. Установлено соответствие между типами знаний и комплексами НМ, в соответствии с этой классификацией. Рассмотрены основные задачи ИППР и определены необходимые для их решения типы знаний.

Представлен предлагаемый метод формирования комплекса НМ, ориентированных на решение как простых, так и комплексных задач интеллектуальной поддержки принятия решений. Предложен оригинальный подход к оцениванию эффективности применения НМ в задачах ИППР. Разработан способ обоснования и выбора совокупностей НМ, рационально реализующих простые и комплексные задачи ИППР.

В дополнение к созданному научно-методическому аппарату требуется разработка методов, позволяющих: унифицировать представление НМ, процессы нечетких вычислений, модельные компоненты; формировать базисы нечетких операций в соответствии с задачами ИППР, с использованием комплексов НМ.

Во второй части доклада рассмотрены проблемы разработки комплексов нечетких моделей для ИППР, определены требования к платформе создания комплексов этих моделей (далее – Платформа) и обоснована необходимость реализации платформы с использованием технологий NVIDIA.

В настоящее время существует большое число программных средств, предназначенных для решения задач ИППР, в том числе на основе НМ. Для них характерны следующие проблемы. Создание программных средств ИППР на основе НМ является крайне затратным, а особенности их реализации не позволяют осуществить их оперативную доработку и адаптацию к изменяющимся требованиям. С другой стороны, ошибки, допускаемые на этапах постановки задачи, создания и описания ИППР на основе НМ проявляются только после программной реализации, что приводит к существенному увеличению рисков создания таких решений. Кроме того, программная реализация крайне трудоемка и обусловлена не только сложностью научно-математического аппарата, но и необходимостью настройки на конкретные аппаратные средства, обеспечивающие требуемую производительность.

Платформа должна позволять: *специалисту предметной области* решать типовые задачи ИППР на основе НМ без участия аналитика и программиста; *аналитику* выполнять создание и детальную настройку НМ без участия программиста; *разработчику* уменьшить затраты на реализацию НМ за счет: использования библиотек типовых НМ; абстрагирования от особенностей аппаратной реализации; встроенной поддержки нечетких вычислений.

Платформа представляет совокупность инструментальных и методических средств для разработки и реализации комплексов НМ и включает в себя следующие подсистемы: решения типовых задач ИППР; разработки НМ; настройки НМ; методического обеспечения, позволяющую выполнять постановку задач ИППР и формировать комплекс НМ; реализации нечетких вычислений; распараллеливания вычислений; генерации программного кода НМ.

При реализации комплексов НМ на основе Платформы критическим является обеспечение требуемой производительности. Модели, обеспечивающие хорошее качество принимаемых решений одновременно и более требовательны к ресурсам. Так при решении задач нечеткого моделирования (в т.ч. в реальном масштабе времени) для оценки и достоверного прогноза развития обстановки требуется перебор возможных вариантов

воздействий и изменений. В результате сложность вычислений (и требования к вычислительным ресурсам) растет экспоненциально с ростом глубины прогноза. В результате при заданном времени анализа текущей обстановки, получаем прямую зависимость глубины прогноза от имеющихся вычислительных ресурсов. Характерной особенностью реализации нечетких вычислений при создании и использовании комплексов НМ для ИППР является большой объем массовых матричных и векторных вычислений. Реальной альтернативой при обеспечении требуемой точности и полноты анализируемой информации для повышения производительности при решении этих задач является применение массовых параллельных вычислений.

На сегодняшний день лидерами рынка, обеспечивающими эффективную реализацию массовых параллельных вычислений, являются NVIDIA и ATI (последняя – в составе фирмы AMD). Преимущество технологий NVIDIA заключается в программно-аппаратном решении CUDA, обеспечивающем высокоэффективную реализацию массовых параллельных вычислений.

Параллелизм может быть реализован на уровнях: разделения между процессорами массивов данных, обрабатываемых в рамках одной нечеткой операции с повышенными требованиями к точности; разделения однотипных операций, выполняемых в модели в массовом порядке; естественного параллелизма ветвей вычислительного процесса, а также за счет относительной независимости функционирования отдельных НМ.

Основные публикации авторов по теме доклада

1. Компьютерная поддержка сложных организационно-технических систем / В. В. Борисов, И. А. Бычков, А. В. Дементьев, А. П. Соловьев, А. С. Федулов. – М.: Горячая линия – Телеком, 2002. – 154 с.
2. Борисов В. В., Федулов А. С. Нечеткие оценочные модели сложных систем с учетом согласования неравнозначных целей// Нейрокомпьютеры: разработка, применение, № 5, 2003. – С. 3–12.
3. Борисов В. В., Федулов А. С. Обобщенные нечеткие когнитивные карты// Нейрокомпьютеры: разработка, применение, № 4, 2004. – С. 3–20.
4. Борисов В. В., Круглов В. В., Федулов А. С. Нечеткие модели и сети. – М.: Горячая линия – Телеком, 2007. – 284 с.
5. Зернов М.М. Способ построения нечёткой многокритериальной оценочной модели// Нейрокомпьютеры: Разработка и применение. – №1 – 2007. – С. 40–49.
6. Борисов В. В., Федулов А. С. Способы интеграции нейронных и нечетких сетей // Нейрокомпьютеры: разработка, применение. 2007. № 1. С. 5–11.
7. Борисов В. В., Федулов А. С., Устиненков Е. С. Анализ динамики состояния сложных систем на основе обобщенных нечетких когнитивных карт// Нейрокомпьютеры: разработка, применение. 2007. № 1. С. 17–23.
8. Борисов В. В., Устиненков Е. С. Анализ взаимодействий в сложных системах на основе нечетких когнитивных и игровых моделей // Нейрокомпьютеры: разработка, применение, № 8, 2009. – С. 4–11.
9. Борисов В. В., Белозерский А. Ю. Анализ рисков на основе нечетких байесовых сетей// Нейрокомпьютеры: разработка, применение, № 8, 2009. – С. 23–30.
10. Борисов В. В., Зернов М.М. Реализация ситуационного подхода на основе иерархической ситуационно-событийной сети// Искусственный интеллект и поддержка принятия решений, № 1, 2009. – С. 17–30.

Опыт разработки массивно-параллелльных версий системы data mining PolyAnalyst для TESLA GPU и кластерных архитектур.

Блохин В.,
Аналитик Проект, *vblokhin@ap.ru*
Киселев М.,
Megaputer Intelligence, *mkiselev@megaputer.ru*

Практика внедрения системы data mining PolyAnalyst, реализованной на отдельных (stand alone) 4 - 8 ядерных математических серверах, в крупных корпорациях России и США показывает, что многие типовые операции анализа данных сейчас находятся недалеко от границ практической приемлемости в терминах времени счета и (гораздо реже) необходимого количества оперативной памяти. Причем, если для России эти операции даже в случае самых крупных в своей отрасли компаний еще, как правило, находятся в рамках этих границ, то для американских компаний они уже выходят за них – иногда сильно. Сейчас, очень условно, эту границу можно провести где-то в районе сотен гигабайтов сырых структурированных и естественно-языковых данных (речь идет о данных, непосредственно подвергаемых анализу, без мультимедийных данных). Стало общим местом признавать, что расширять эту границу в наше время возможно только с помощью распараллеливания аналитических и трансформационных алгоритмов. Вопрос только в путях этого распараллеливания. Основных путей 2 – распределенные вычисления на кластерных архитектурах и массивно-параллельные версии алгоритмов для многоядерных ускорителей, встроенных в специализированные математические сервера (хотя возможна и их комбинация – но это уже скорее уникальные сверхдорогие вычислительные комплексы, выходящие за рамки нашей темы). При этом, как уже отмечалось многими авторами, с точки зрения второго пути сейчас трудно найти альтернативу графическим процессорам (GPU) и их потомкам, утерявшим в процессе эволюции разъемы видео-выхода. К числу последних относятся и рассматриваемые в данном докладе платы TESLA производства компании NVIDIA. Выбор того или иного пути в очень большой степени зависит от специфики решаемых задач и коммерческих перспектив создаваемого решения. В данном докладе мы рассматриваем опыт реализации ориентированных на TESLA параллельных версий компонент системы data mining PolyAnalyst (производитель – компания Megaputer Intelligence), осуществляемого компанией Аналитик Проект, в сравнении с опытом создания компанией Megaputer Intelligence кластерной версии системы PolyAnalyst.

Стоимость аппаратуры и обслуживания. С точки зрения этого критерия GPU вне досягаемости. Даже в случае кластерных решений на основе обычных мощных PC, объединенных в локальную сеть, сравнимый уровень распараллеливания достигается ценой большей, как минимум, на 2 порядка. Еще больше этот разрыв делается при учете стоимости поддержки, затрат на электроэнергию и т.д. Правда, эта разница в довольно большой степени сглаживается тем обстоятельством, что стоимость аппаратуры и поддержки в случае решений, предназначенных для очень крупных корпоративных заказчиков, как правило, не имеет решающего конкурентного значения. Таким образом, этот аспект применения GPU наиболее важен с точки зрения внедрения в компаниях, так сказать, второго эшелона, где объем анализируемых данных все еще представляет трудности для непараллельных систем.

Стоимость и время разработки. По этому параметру GPU также оказывается в выгодном положении, особенно если учитывать структуру инвестиций в разработку. В случае реализации кластерной версии первоначально требуется провести дорогостоящую разработку координационно-транспортного уровня поддержки распределенных вычислений на кластере. Хотя существует несколько готовых программных платформ подобного рода (типа MPI или MapReduce), они либо коммерчески недоступны, либо не

удовлетворяют специфическим требованиям решаемых задач анализа данных. В тоже время все необходимое для разработки и внедрения систем, использующих TESLA, на данный момент предоставляется компанией NVIDIA бесплатно. При этом в случае TESLA разработка ведется инкрементально – независимо друг от друга создаются параллельные версии отдельных модулей, которые сразу могут быть включены в пакет поставки. Только при создании самих кодов параллельных версий одних и тех же алгоритмов затраты труда для GPU становятся в несколько раз большими по сравнению с кластерной математикой – сказывается очень необычная для традиционного программирования модель вычислений, множество специфических аппаратных ограничений и пока еще недостаточная поддержка инструментальными средствами разработки.

Уровни распараллеливания. Гибкость и разнообразие схем распараллеливания, конечно же, больше в случае кластерной архитектуры. Там возможно как высокоуровневое распараллеливание (на уровне пользовательских сессий, логически независимых друг от друга операции с данными в рамках одного проекта, и т.д.), так и распараллеливание отдельных базовых вычислительных алгоритмов, в особенности экстенсивных, с линейной зависимостью времени работы от объема данных. Для GPU возможен только последний вариант. Некоторым преимуществом GPU является существенно большая скорость обмена данными, однако, как показывает наша практика, это преимущество редко играет решающую роль. В то же время, даже на уровне алгоритмов диапазон того, что можно распараллелить, у GPU существенно уже. Кроме того, эффект от распараллеливания алгоритмов на GPU как правило гораздо менее предсказуем a priori чем для кластера, а выбор оптимальной схемы распараллеливания существенно более труден.

Гибкость физического и логического конфигурирования, толерантность к отказам. Современные платформы распределенных вычислений на кластерах (например, MapReduce) предусматривают возможность «горячего» изменения физической конфигурации кластера и замены сбойных узлов кластера без остановки вычислений. Кластерная платформа, разрабатываемая компанией Megaputer Intelligence, также обладает этим свойством. В системах, основанных на GPU, это невозможно. Однако, в обоих этих вариантах возможно наращивание вычислительной мощности системы без пересборки ПО.

Достигаемое ускорение. Разумеется, этот параметр сильно определяется используемой аппаратурой, однако характерные значения оказываются сравнимыми в обоих случаях. Для системы, основанной на 4 платах TESLA C1060, максимальное ожидаемое на практике ускорение составляет где-то 300 раз (разумеется, на тех алгоритмах, которые в принципе допускают распараллеливание на TESLA). Для наиболее типичной кластерной конфигурации, включающей сотни узлов, можно ожидать ускорение того же порядка.

Таким образом, на основе нашего опыта мы обрисовали сильные и слабые стороны двух подходов к построению массивно-параллельных систем анализа данных – кластерного и основанного на GPU. Как видно, ни один из них не имеет безусловного преимущества, поэтому наши компании в целях расширения диапазона потенциальных покупателей приняли решение вести разработку в обоих направлениях.

Вопросы построения персональных суперкомпьютерных конфигураций на базе графических процессоров

Аляутдинов М.А.(ИНТЕРЭВМ), Воронков И.М. (ИНТЕРЭВМ)

Современные графические процессоры (GPU) nVidia с унифицированной архитектурой в виде массива однотипных ядерных скалярных процессоров одинаково эффективно используются как для решения задач компьютерной графики так и для неграфических вычислений. К ним относятся :

-G80 и G90 с 16/32/96/128 ядрами и пиковой производительностью до 0;5Tflops;

-GT200 с 96/192/240 ядрами и пиковой производительностью до 1;0 Tflops;

-GT300 с 512 ядрами и пиковой производительностью до 3;0 Tflops.

Это позволяет строить на их базе гибкие масштабируемые супервычислительные комплексы. На базе графических плат с вышеперечисленными чипами nVidia можно строить вычислительные комплексы со следующими конфигурациями:

1.Автономные настольные суперкомпьютеры:

-персональные компьютер с графическими платами акселераторами:

-настольные суперкомпьютеры на базе одного или нескольких nVidia Tesla ;

2.Серверы с супервычислительными возможностями на базе nVidia Tesla

3.Кластерные конфигурации, узлами которых являются вышеназванные рабочие станции с платами акселераторами, настольные суперкомпьютеры и серверы

Новейшие вычислительные модули Nvidia Tesla M2050 и M2070(на базе чипов GT300- «Fermi») обеспечивают прирост производительности вычислений двойной точности более чем в 10 раз по сравнению с четырехядерными процессорами x86. В сравнении с основанными только на CPU системами серверы на основе вычислительного модуля Tesla GPU серии 20 обеспечивают супервычисления при 1/20 потребления электроэнергии и 1/10 стоимости .

В настоящее время коммерчески доступны различные суперкомпьютерные конфигурации на базе GPU. Однако, в ряде случаев у пользователей возникает необходимость создания собственной, в том числе персональной, ЭВМ с супервычислительными возможностями. Для достижения этой цели в настоящее время существуют необходимые предпосылки и возможности.

Для успешной реализации таких проектов необходимо проведение этапа работ по выяснению специфических особенностей и факторов, влияющих на показатели производительности вычислений для принятия решения о целесообразности их использования при реализации на них алгоритмов прикладных задач.

Такие исследования и разработки были проведены авторами в рамках обучения по курсу “Архитектура и программирование потоковых многоядерных процессоров для научных расчетов и параллельных вычислений” при поддержке компании nVidia и центра “Проблем связи, передачи , обработки и защиты информации “ МФТИ.
Работы выполнялись на базе сервера со следующей конфигурацией:

-OS Linux; CPU_HOST Intel Core2 Quad Q6600 2.40GHz CPU, 4 Gb RAM;

-3 x 8800GTS графические карты nVidia ;

- среда программирования nVidia CUDA v.2.

Работа на сервере проходила в режиме удаленного доступа через Интернет.

В качестве тестовой задачи была выбрана задача обработки изображений. Были разработаны алгоритмы и программы распараллеливания обработки изображений как на многоядерном CPU HOST компьютера, так и на трех (multi) GPU платах nVidia.
При реализации тестовой задачи на GPU получено ускорение до 100 и более раз относительно CPU HOST. Было исследованы все компоненты полных временных затрат и проведено сопоставление времени на побочные операции со временем рабочих вычислений на GPU. Это позволяет делать выводы по эффективности и целесообразности распараллеливания прикладной задачи на GPU.

Литература:

Аляутдинов М.А., Воронков И. М. Исследование вопросов построения масштабируемых суперкомпьютерных конфигураций на базе графических процессоров с унифицированной архитектурой. “Информатизация и Связь”., №1 , 2009 г.

Применение графических процессоров для 2D и 3D распознавания лиц.

Коробкова С.В. (компания ВОКОРД), Манолов А.И. (компания ВОКОРД)

paraslonic@gmail.com

Задача распознавания человека с использованием характеристик лица уже в течение нескольких десятилетий остается актуальной. Разрабатываются новые, более точные алгоритмы, при этом растут требования к точности распознавания. В последние годы возрос интерес наряду с 2D распознаванием лиц к распознаванию лиц по форме – 3D распознаванию.
В связи с появлением новых архитектур графических процессоров, позволяющих реализовать сложные параллельные вычисления, а также удобных инструментов программирования для графических процессоров, популярным направлением исследований стало параллельное решение вычислительно сложных задач с использованием графических ускорителей. Данная статья посвящена решению задачи распознавания лиц с использованием графических процессоров для ускорения вычислений в задаче распознавание лиц.

2D распознавание лиц.

Задача распознавания человека по фотографии лица остается актуальной в течение нескольких десятилетий. Алгоритмы 2D распознавания лиц могут применяться для аутентификации, в корпоративных системах доступа и в глобальных системах безопасности.

Работа предлагаемой системы распознавания лица по 2D изображению состоит из 3 этапов:

1. Предобработка изображения – включает в себя выравнивание яркости изображения, нормализация размера, поиск ключевых точек лица, удаление фона и другие необходимые операции.
2. Обработка изображения группой классификаторов, каждый из классификаторов выделяет свой набор входных признаков и осуществляет классификацию.
3. Получение интегрального результата по выходам используемых классификаторов.

Каждый этап обработки тестового изображения может быть реализован как на CPU, так и с использованием графических ускорителей. Для исследования возможности ускорения процесса распознавания был выбран второй этап – классификация изображений. Классификаторы, используемые в системе, разработаны на базисе нейронных сетей. Их естественный внутренний параллелизм позволяет легко перейти на параллельные вычисления. Использование графических ускорителей nVidia GTX260 позволило ускорить данный этап в 12 раз, при этом общее ускорение процесса распознавания составило 2 раза. В дальнейшем планируется исследовать возможности ускорения этапа предобработки и получения интегрального результата.

3D распознавание лиц.

Для реализации распознавания человека по форме его лица требуется разработать систему, позволяющую достаточно быстро получать 3D модель лица данного человека. Для получения такой модели используются 2 или более камер высокого разрешения.

Реконструкция трехмерных объектов по двум и более его снимкам производится с использованием информации о взаимном расположении камер в пространстве (его задают калибровочные матрицы камер) и точках на разных проекциях, которые соответствуют одним и тем же точкам на трёхмерном объекте [5]. Последняя задача и является наиболее вычислительно емкой в процессе реконструкции. Для ее решения мы ищем максимум корреляции между небольшими (порядка 16x16 пикселей) участками изображений, причем поиск проходит в несколько стадий. Вначале поиск соответствия на уменьшенных в три раза изображениях, затем уточнение их на уменьшенных в два раза изображениях, на исходных и, наконец, на полученных интерполяцией увеличенных в два раза изображениях, что позволяет достичь субпиксельной точности.

Задачу поиска соответствий мы можем решать независимо для различных точек изображений. Такая особенность алгоритма позволила значительно ускорить процесс поиска благодаря применению параллельных вычислений на графическом процессоре [6]. Так, реконструкция области 500x500 пикселей с шагом в 4 пикселя для двух камер на процессоре Intel Core2 Q6600 2.4 ГГц занимает 5.1 сек, а на видеокарте NVIDIA GTX260: 0.169 сек. Таким образом, ускорение достигло 30 раз. При реконструкции по снимкам с трех камер и использовании каждого из них как базового, ускорение составило 10 раз (так как вычисление на ядрах процессора также шло параллельно).

Литература.

1. Галушкин А.И. Теория нейронных сетей. М.: Радиотехника, 2000. 416 с.
2. Галушкин А.И. Решение задач в нейросетевом логическом базисе. Нейрокомпьютеры: разработка, применение. № 2. М.: Радиотехника, 2006. с. 49-70.
3. Kyoung-Su Oh, Keeghul Jung GPU implementation of neural networks. Pattern Recognition 37 2004. p. 1311-1314.
4. Ruigang Yang Dense Matrix Algebra on the GPU. Computational and Information Science. Berlin: Springer, 2005. p. 1100-1105.
5. Форсайт Д.А., Джин П. Компьютерное зрение. Современный подход. - М.: "Вильямс", 2004. – 928 с.
6. Joe Stam. Stereo Imaging with CUDA, 2008.

Пакеты программ обработки изображений для вычислительных систем на базе процессоров NVIDIA

Аляутдинов М.А. (ИнтерЭВМ), Коробкова С.В. (компания ВОКОРД), Нгуен В.Х. (ИнтерЭВМ)

Обработка изображений и потокового видео относятся к задачам, требующих интенсивных вычислений, что обусловлено большими объемами обрабатываемой информации, требованиями интерактивности и работы в реальном масштабе времени.

В настоящее время реализации высокопроизводительных вычислений (HPC)- осуществляется на базе многоядерных и многопоточных микропроцессоров с параллельной архитектурой:

- крупно зернистой (multicore) с числом ядер исчисляемых десятками (Intel: Larrabee 24-32 ядра/2009г.;48 ядер-2010г.; Keifer-32 ядра) на базе CPU.

- с мелкозернистой (manycore) с числом ядер исчисляемых сотнями и тысячами на базе графических процессоров(GPU) и FPGA.

Для эффективной реализации обработки изображений и потокового видео на многоядерных процессорах с параллельной архитектурой необходимо использование адекватных инструментальных программных средств.

Среди последних получили признание и применение следующие средства:

1. **INTEL-OpenCV - Open Source Computer Vision -** библиотека алгоритмов компьютерного зрения, обработки изображений и численных алгоритмов общего назначения, реализованная на C/C++ и других языках программирования. Функции этой библиотеки обеспечивают выполнение базовых операций над изображениями, анализ изображений, структурный анализ, трекинг движения объектов, обнаружение объектов и др. OpenCV поддерживают нейросетевые методы распознавания образов. Может использоваться как в академических, так и в коммерческих целях.
2. **OpenVIDIA: Parallel GPU Computer Vision.** OpenVIDIA реализует многие популярные алгоритмы на базе GPU (Convolution, Sobel, RMS, гистограммы, пороговые, построение 3D стерео изображений, операции обработки потокового видео и др.) с использованием средств программирования : OpenGL, Cg и CUDA-C , OpenCL, и Direct Compute API.
3. **GpuCV** - библиотека функций обработки изображений и машинного зрения для GPU. В ней используется интерфейс программирования аналогичный Intel OpenCV-для облегчения переноса существующего приложения OpenCV в среду GPU.
4. **GPU4Vision** - проект исследований и разработок по следующим направлениям: компьютерная графика, медицинское машинное зрение, распознавание объектов, реконструкция объектов, робототехника, виртуальная расширенная реальность(Virtual Reality and Augmented Reality) на базе GPU и nVidia CUDA.
5. **NVIDIA NPP -** библиотека базовых функций для программирования задач обработки изображений и потокового видео (в том числе предобработки, выделение контуров объектов, детекции объектов и др.) на базе графических процессоров в среде программирования nVidia CUDA.

6. **MinGPU** – библиотека программ машинного зрения, которая содержит минимальный необходимый набор функций для переноса существующего кода программ для CPU на GPU для их существенного ускорения.
7. **Sapera Essential** инструментальная среда разработки программ обработки изображений и высокоэффективных систем машинного зрения для многоядерных CPU.
8. **Sapera Essential Nitrous** является набором функций программ обработки изображения, оптимизированных для GPU и CPU процессоров.
9. **DirectX Compute** представляет новый API для вычислений на GPU, работающий с текущей версией архитектуры параллельных вычислений CUDA от NVIDIA в системах Windows VISTA и Windows 7. Благодаря возможностям DirectCompute, разработчики могут использовать всю мощь параллельных вычислений на графических процессорах NVIDIA при разработке приложений для потребительского и профессионального рынков.
10. **FOLKI GPU** – пакет, используемый для быстрого вычисления оптического потока.
11. **GPU VSIPL** – библиотека векторной обработкой изображения на базе CUDA.

Также существует ряд узкоспециализированных пакетов обработки изображений и видеоинформации.

Литература :

1. Open Source Computer Vision Library/ Reference Manual. Intel Corporation. http://www.cs.unc.edu/Research/stc/FAQs/OpenCV/OpenCVReferenceManual.pdf
2. Yannick Allusse, Patrick Horain, Ankit Agarwal and Cindula Saipriyadarshan, "GpuCV: A GPU-accelerated framework for Image Processing and Computer Vision", Advances in Visual Computing, *Proceedings of the 4th International Symposium on Visual Computing (ISVC08)*, December 1-3, 2008, Las Vegas, Nevada, USA, LNCS Volume 5359, Springer-Verlag, p. 430-439 [doi:10.1007/978-3-540-89646-3_42 .
3. OpenVIDIA Parallel GPU Computer Vision. http://openvidia.sourceforge.net/index.php/OpenVIDIA
4. GPU4Vision. http://gpu4vision.icg.tugraz.at/
5. MinGPU: A minimum GPU library for Computer Vision. http://server.cs.ucf.edu/~vision/MinGPU/
6. NVIDIA NPP image video LIB. http://developer.nvidia.com/object/npp_home.html
7. Sapera Essential. http://www.dalsa.com/mv/products/software.aspx
8. Sapera Essential Nitrous .http://www.dalsa.com/mv/products/saperanew.aspx
9. DirectCompute для nVidia. http://www.nvidia.ru/object/directcompute_ru.html

Виртуальные нейросетевые процессоры – метод кроссплатформенного ускорения расчета нейросетевых задач на современных многоядерных процессорах.

Долгополов А.В., Скрибцов П.В.

ООО «ПАВЛИН Технологии»

Сложность применения нейронных сетей в режиме реального времени состоит в сравнительно низкой производительности при реализации их стандартными средствами c++. Нейронные сети хорошо распараллеливаются[1] и в настоящее время для ускорения нейросетевых вычислений могут использоваться различные технологии: Intel MKL, CUDA, ACML.

Технологии распараллеливания - MKL, CUDA

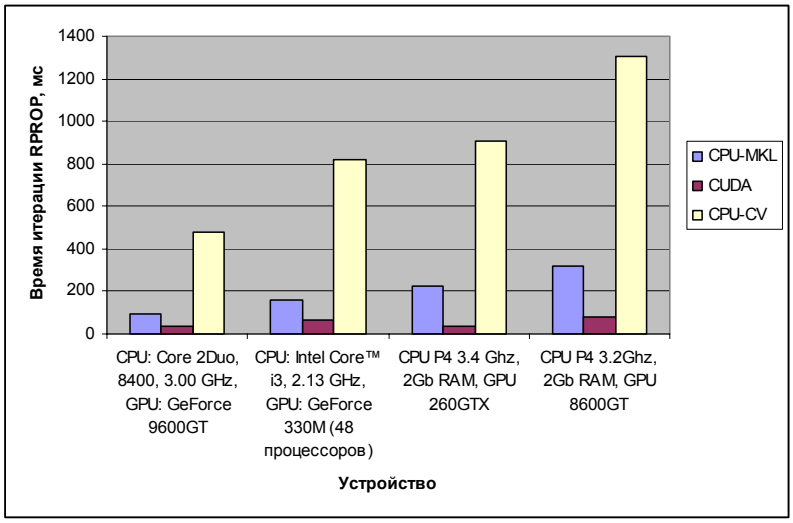
Современные графические процессорные устройства (GPU) являются мощными специализированными вычислительными устройствами, которые могут быть использованы для ускорения решения целого ряда вычислительно сложных задач. Достаточно простым и удобным инструментом для разработки приложений, использующих вычислительные мощности GPU, является технология Nvidia Cuda[6]. Библиотека Intel® Math Kernel Library (Intel® MKL)[7] обеспечивает выполнение высокооптимизированных многопоточных математических операций для научных, инженерных и финансовых приложений. Бесплатной альтернативной необходимого подмножества API библиотеки Intel MKL является открытая библиотека Intel OpenCV[8].

Виртуальный нейросетевой процессор.

Виртуальный нейросетевой процессор оптимизирован для задач с большим объемом данных, и поддерживает в данный момент следующие программно-аппаратные платформы: OpenCV[8], Intel MKL, NVIDIA CUDA. Виртуальный процессор поддерживает многослойные нейронные сети прямого распространения[2] с различными функциями активации. Для всех программано-аппаратных платформ поддерживается два алгоритма обучения: RPROP[4] и QUICKPROP[5], для OpenCV и MKL дополнительно поддерживаются алгоритмы LM(Levenberg Marquardt)[3] и TEL(Target Error Learning)[9]. Для смены программно-аппаратной платформы достаточно изменить текстовый конфигурационный файл и наличие соответствующей библиотеки динамической линковки (DLL), для прикладного ПО не потребуется даже перекомпиляции.

Сравнение производительности программно-аппаратных платформ.

Были произведены эксперименты, с использованием виртуального нейросетевого процессора. Тестирование проводилось для сети конфигурации 100-16-12-1, на обучающей последовательности с числом примеров 100000. Сравнивалось время выполнения одной итерации обучения для алгоритма RPROP. Результаты тестирования для различных конфигураций оборудования представлены ниже.



Из диаграммы видно, что технологии программирования многоядерных процессоров CUDA и MKL позволяют существенно ускорять нейросетевые вычисления, при сохранении переносимости прикладного ПО.

Универсальный алгоритм распознавания объектов на фото и видео с применением аппаратного ускорения вычислений на современных многоядерных процессорах.

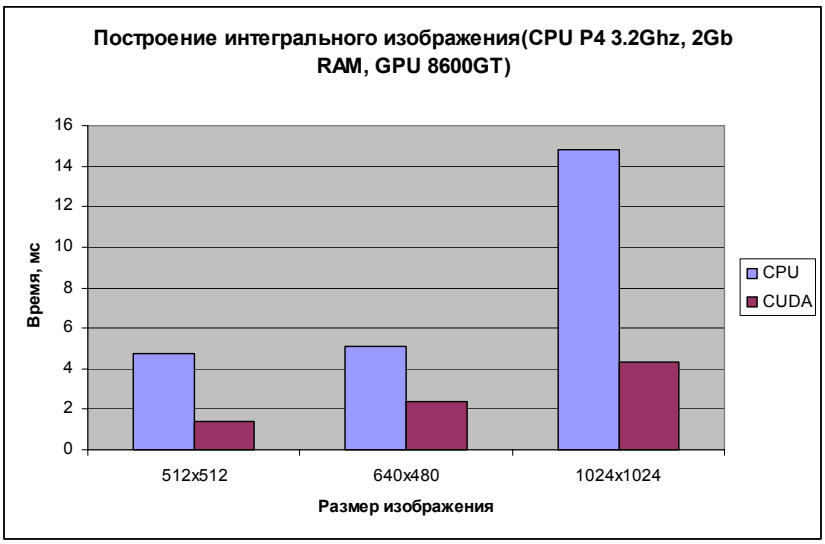
Скрибцов П.В., Долгополов А.В.

ООО «ПАВЛИН Технологии», http://www.pawlin.ru

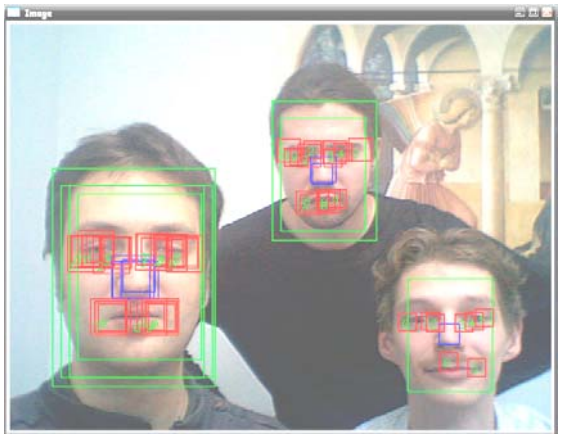
Для решения задачи распознавания объектов на фото и видео с применением аппаратного ускорения вычислений на многоядерных процессорах предлагается универсальный нейросетевой алгоритм с параллелизуемой архитектурой.

Универсальный алгоритм распознавания на фото и видео

В алгоритме можно выделить две фазы: обучения и распознавания. В режиме обучения на вход подаются обучающие примеры объектов. Особенностью алгоритма является использование многокаскадного классификатора. Для каждого каскада формируется своя обучающая матрица признаков и обучается своя нейронная сеть. Формирование признаков и обучение сетей при таком подходе можно распараллелить с помощью графических процессоров. В данном алгоритме наиболее вычислительно емкими являются: формирование интегральных изображений, формирование матриц признаков, расчет выходов нейронной сети. Данные операции можно эффективно распараллеливать с помощью GPU. На графике в качестве примера ускорения одного из этапов оборабоки представлено время построения интегрального изображения для различных размеров изображения на CPU и GPU.



Данный алгоритм был успешно применен задаче детектирования лиц, одной из самых актуальных в системах компьютерного зрения.



В докладе рассмотрен алгоритм распознавания объектов на фото и видео, который использует нейросетевые технологии. Алгоритм хорошо распараллеливается на современных графических (многоядерных) процессорах, и показал свою эффективность в задаче детектирования лиц.

О построении моделей процессов и явлений на вычислительных системах с использованием графических процессоров NVIDIA

Пантюхин Д.В.[1], Аведьян Э.Д.[2]

[1] Международный центр по информатике и электронике (ИнтерЭВМ),
dim_beavis@mail.ru
[2] Центр информационных технологий и систем органов исполнительной власти (ЦИТиС),
avedian@mail.ru

Введение

Центр информационных технологий и систем (ЦИТиС) совместно с Международным центром по информатике и электронике (ИнтерЭВМ) ведут разработку информационно-аналитических систем для различных систем государственного управления.

Одной из важных задач в таких системах является задача построения моделей процессов и явлений. Для построения моделей применяются различные методы, включая:

- регрессионные методы,
- методы составления и решения систем нелинейных обыкновенных дифф. уравнений,
- методы составления и решения систем специальных дифф. уравнений в частных производных,
- методы сценарного анализа,
- нейросетевые методы.

Опыт работ показывает, что зачастую для реализация таких моделей в реальном времени, в силу больших размерностей, вычислительных возможностей обычных персональных компьютеров недостаточно, поэтому требуется применение высокопроизводительных вычислителей. В качестве таких вычислителей выбраны высокопроизводительные вычислительные системы на базе процессоров NVIDIA.

В этом случае можно выделить два направления исследований:

- создание собственных программных моделей на базе процессоров NVIDIA,
- изучение, освоение и использование известных программных продуктов на базе процессоров NVIDIA.

Если, в настоящий момент, собственные программные модели находятся лишь на стадии разработки, то в мире уже существует довольно значительное количество программных продуктов, реализующих те или иные методы. Этим продуктам и посвящен настоящий доклад. Среди всех существующих пакетов прикладных программ моделирования на базе процессоров NVIDIA мы рассмотрим пакеты в следующих областях [1]:

- реализация метода сопряженных градиентов,
- реализация метода Монте-Карло,
- реализация методов максимизации математического ожидания,
- реализация методов решения стохастических дифференциальных уравнений,
- реализация вычислений на процессорах NVIDIA для известных пакетов моделирования.

В докладах наших коллег рассмотрены пакеты программ моделирования на базе процессоров NVIDIA в других областях, таких как:

- нейросетевые технологии,
- решения уравнений математической физики,
- биоинформатика.

Кроме того, существует множество пакетов программ моделирования явлений, напрямую не связанных с проблемами биоинформатики, например в моделировании социально-экономических явлений и финансовых рынков, моделировании транспортных

потоков, расчетах квантовой физике и химии и многих других областях. В данном докладе мы не будем касаться таких пакетов программ, однако, на наш взгляд, изучение заложенных в них методов моделирования и способов реализации этих методов на процессорах NVIDIA может быть полезным при разработке собственных методов моделирования и их реализации.

Реализация метода сопряженных градиентов

Реализация метода сопряженных градиентов представлена пакетом компании Elegant Mathematics [2] "Conjugated Gradient CUDA and CPU solvers for float, double and quad precision" реализованного на языке CUDA с наличием доступного исходного кода. Поддерживает одинарную, двойную, четверную точность вычислений с реализацией как на CPU так и на GPU.

Компанией так же разработаны библиотеки для реализации методов сплайн-аппроксимации, решения линейных уравнений, факторизации и др.

Реализация метода Монте-Карло

Реализация метода Монте-Карло представлена пакетом "Massively Parallel Population-Based Monte Carlo Methods" разработанного в Оксфордском университете и применяется для моделирования пространственно-временных процессов [3]. Пакет реализован на языке CUDA и предоставляется с отрытым кодом. Заявлено ускорение расчетов в 500 раз.

Существуют также реализации метода Монте-Карло в составе пакетов финансовых ("ClusterTech Financial Library in GPU", "SciFinance") и электродинамических ("Monte Carlo eXtreme") расчетов с заявленным ускорением от 30 до 300 раз.

Реализация метода максимизации математического ожидания

Реализация метода максимизации математического ожидания представлена пакетом [4] "CUDA Accelerated Expectation Maximization of Gaussian Mixture Models" с открытым кодом. Также доступна интеграция с пакетом Matlab. Заявлено ускорений до 170 раз.

Реализация методов решения стохастических дифференциальных уравнений

Реализация методов решения стохастических дифференциальных уравнений представлена пакетом университета Силезии (Silesia University) [5] "Stochastic Differential Equations with CUDA". Пакет предоставляется с открытым кодом. Заявлено ускорение в 675 раз.

Реализация вычислений на процессорах NVIDIA для пакетов моделирования

Помимо отдельных пакетов моделирования существуют пакеты программ, позволяющие интегрировать вычисления на процессорах NVIDIA в известные пакеты моделирования, например Matlab, Labview, R, Mathematica.

Для Matlab стоит отметить два пакета программ:

1) "GPUmat" группы GP-you [6] - доступное приложение, позволяющее исполнять программы в синтаксисе Matlab непосредственно на GPU. Вводятся дополнительные функции для создания массивов, операции с которыми будут выполняться на GPU. Вычисления прозрачны для пользователя. На операциях умножения заявлено ускорение до 40 раз.

2) "Jacket for Matlab" [7] - коммерческий продукт фирмы AccelerEyes, позволяет запускать код Matlab на GPU, поддерживает работу с несколькими GPU, и GPU-кластерами. Позволяет также использование GPU для ускорения 3d-визуализации данных, вычислений с двойной точностью, подключение пользовательских CUDA-функций. Заявлено ускорение в среднем до 50 раз.

Также имеется множество библиотек (например "GPULib" [8]) и отдельных функций для Matlab, реализованных на CUDA.

Для других пакетов моделирования также разработаны способы вычислений на GPU: "LabVIEW GPU Computing" для пакета LabVIEW [9], "R+GPU" для пакета R [10], "Mathematica Cuda plug-in" для пакета Mathematica [11].

1. http://www.nvidia.ru/object/cuda_home_new_ru.html
2. http://www.elegant-mathematics.com
3. http://www.oxford-man.ox.ac.uk/gpuss/
4. http://andrewharp.com/gmmcuda
5. http://if.us.edu.pl/cms/index.php?option=com_content&view=article&id=925
6. http://gp-you.org/
7. http://www.accelereyes.com/
8. http://www.txcorp.com/products/GPULib/
9. http://decibel.ni.com/content/docs/DOC-6064
10. http://brainarray.mbni.med.umich.edu/brainarray/rgpgpu/
11. http://hpc.nomad-labs.com/?p=8

О переработке системы TextAnalyst на платформу NVIDIA

Воронков И.М., А.А. Харламов

МФТИ(ГУ) *iliav_scn@mail.ru*,

НПИЦ «Микросистемы» *kharlamov@analyst.ru*

TextAnalyst это библиотека программных компонентов для анализа текстов, основными функциями которой являются следующие:

- приведение различных грамматических форм слов к общей форме;
- формирование частотного словаря понятий текста (слов и словосочетаний), с фильтрацией слов заданного словаря;
- формирование списков слов, связанных в тексте по смыслу с заданными словами;
- численную оценку значимости понятий в тексте и степени их смысловой связанности;
- получение списков предложений текстов, содержащих заданные слова;
- формирование списка наиболее информативных предложений текста.

На основе этих базовых функций TextAnalyst применяется для решения различных задач анализа текстов, среди которых классификация, кластеризация, реферирование, построение семантической сети, ассоциативный поиск и др. Отличительной особенностью TextAnalyst является применение нейросетевых технологий [1] при переходе от частотных характеристик текста к семантической сети, которая является базовой в дальнейшей обработке и представлении информации.

Алгоритмы анализа текстов являются в достаточной степени параллельными, что вполне очевидно и доказывается на практике разработчиками по всему миру получающим впечатляющие результаты распараллеливания на массовых параллельных вычислительных средствах (многоядерных процессорах и в первую очередь графических процессорах с унифицированной архитектурой) [2-5].

Исследования ускорения обработки текстов TextAnalyst успешно проводились в части ускорения нейросетевой обработки [6] и результаты привели к выводу о необходимости полной обработки на графических процессорах вместо реализации ряда алгоритмов. Так как требовательность к вычислительным ресурсам при такой реализации будет максимальной, то это позволит оправдать возникающий в результате расчетов на графических процессорах «оверхед» - операции, выполнение которых не происходит при расчетах на центральном процессоре. Наибольший вклад в такие «лишние» операции будет вносить копирование данных из оперативной памяти в память графического ускорителя, а при обработке текстов значительные объемы памяти (кроме анализируемого текста) занимают применяемые словари. Однако при обработке множества текстов в пакетном режиме можно существенно сократить вклад в «оверхед» от копирования

словарей, достаточно выполнить эту операцию для первого текста и не выполнять очищение выделенной для словарей памяти графического ускорителя до окончания обработки всех текстов.

В настоящее время на базе технологии TextAnalyst разрабатывается инструментарий для реализации персональной библиотеки исследователя с ассоциативной навигацией. Необходимость подобного инструмента продиктована в первую очередь постоянно растущим числом электронных публикаций доступных в сети Интернет, которые необходимо анализировать современным исследователям. Основным преимуществом TextAnalyst для успешного применения в составе такой персональной библиотеки является возможность предоставить результаты в виде тематического дерева основных понятий, а также объединять результаты обработки множества текстов. Организация классов документов на основе списка основных понятий позволяет добавить к удобной навигации по всему множеству текстов автоматическую классификацию новых документов.

1. Харламов А.А. Нейросетевая технология представления и обработки информации (естественное представление знаний). - М.: «Радиотехника», 2006. - 89 с.
2. Lucas Veronese, Alberto F. De Souza, Claudine Badue, Elias Oliveira, Patrick M. Ciarelli, Implementation in C+CUDA of Multi-Label Text Categorizers, GPU Technology Conference, San Jose, CA-USA, 2009.
3. Nadathur Satish, Mark Harris, Michael Garland, Designing Efficient Sorting Algorithms for Manycore GPUs, Proc. 23rd IEEE International Parallel and Distributed Processing Symposium, May 2009.
4. Michael C. Schatz, Cole Trapnell, Fast Exact String Matching on the GPU (2008), http://www.cbcb.umd.edu/software/cmatch/Cmatch.pdf
5. Yongpeng Zhang, Frank Mueller, Xiaohui Cui, Thomas Potok, GPU-Accelerated Text Mining, EPHAM, Seattle, Washington, March 2009.
6. Воронков И.М., Лобов С.А., Использование графических ускорителей для задач текстовой обработки с применением нейросетевых технологий, труды 51-й научной конференции МФТИ - Всероссийская молодёжная научная конференция с международным участием «Современные проблемы фундаментальных и прикладных наук», Долгопрудный, ноябрь 2008, ч. I Радиотехника и кибернетика, с 183-186.

Эффективное перемножение разреженных матриц на графических процессорах.

Аветисян А.И. (Институт системного программирования РАН. Текст тезисов отсутствует)

Сверхплотные блейд-решения на архитектуре NVIDIA Fermi.

Ткачев Д.Ф. (Компания Т-Платформы, г.Москва. Текст тезисов отсутствует)

Моделирование деления клеток.

Железняков А.О. (Физфак МГУ. Текст тезисов отсутствует)

II – GPU В ПОСТГЕНОМНЫХ ТЕХНОЛОГИЯХ

Выявление и анализ функциональных сигналов в регуляторных районах генов микро-РНК млекопитающих.

О.В.Вишневский[1,2], К.В.Гунбин[1], А.В.Бочарников[2], А.А.Романенко[2], Е.В.Березиков[1]

[1] Институт Цитологии и Генетики СО РАН, Новосибирск, Россия

oleg@bionet.nsc.ru

[2] Новосибирский Государственный Университет, Новосибирск, Россия

Целью нашей работы являлось выявление и классификация консервативных мотивов в регуляторных районах генов миРНК, экспрессирующихся в различных тканях млекопитающих. Для этого, с использованием баз данных FANTOM, Ensembl и Ensembl Compara нами сформированы выборки промоторов генов миРНК 10 видов млекопитающих (*Homo sapiens, Pan troglodytes, Pongo pygmaeus, Gorilla gorilla, Macaca mulatta, Bos taurus, Equus caballus, Canis familiaris, Mus musculus, Rattus norvegicus*) в районе [-500; +1] относительно старта транскрипции, экспрессирующихся в мозге, легких и выделительной системе.

Выявление вырожденных олигонуклеотидных мотивов и оценка олигонуклеотидного сходства промоторов производилось с помощью разработанной нами системы ARGO, реализованной с использованием GPU Nvidia Tesla c1060.

Сравнение мотивов, обнаруженных с помощью системы ARGO с известными сайтами связывания транскрипционных факторов (ССТФ) из баз данных TRANSFAC, TRRD и MatBase показало, что большинство из них имеет сходство с ССТФ, участвующими в развития организма и регуляции клеточного цикла. При этом в разных видах организмов и в различных тканях были обнаружены как мотивы, соответствующие повсеместно представленным базовым ССТФ, таким как TATA-бокс и CCAAT, так и мотивы, соответствующие ССТФ ткане-специфичных факторов, например MYT1 и NeuroD в выборке мозг- специфичных промоторов. В то же время оказалось, что значительная часть полученных мотивов не может быть достоверно отнесена ни к одному из известных ССТФ. Видимо, такие мотивы могут соответствовать как сайтам связывания еще не исследованных видоспецифичных транскрипционных факторов, так и некоторым структурным особенностям промоторных районов эукариот, таким как поли-A/поли-T тракты, которые способны приводить к локальному изгибу ДНК и повышению ее легкоплавкости.

Затем мы провели оценку среднего олигонуклеотидного сходства промоторов человека с промоторами всех остальных видов. Для сравнения, с помощью парного выравнивания для тех же выборок последовательностей, была проведена оценка усредненного сходства. Анализ нормированных оценок среднего олигонуклеотидного сходства (H_{Oli}) и усредненного сходства (H_{Align}) показал, что для всех видов и среди всех тканей наибольшими величинами и H_{Oli} и H_{Align} характеризуются промоторы мозга. Причем для приматов эти величины максимальны (0.77 – 0.92). Интересно отметить, что H_{Oli} в выборке промоторов собаки (0.77) значительно превышает значения H_{Oli} у других не-приматов (0.56 – 0.59) и достигает величины, характерной для макаки. Сравнение величин H_{Oli} и H_{Align} показывает, что для филогенетически близких к человеку видов (приматы) H_{Oli} и H_{Align} во всех тканях отличаются не более чем на 2%. Исключение составляет выборка промоторов выделительной системы гориллы, для которой при H_{Align}

сходном с другими гоминидами (0.83) величина H_{Oli} была больше на 8% (0.91), и идентична олигонуклеотидному сходству промоторов мозга. Для промоторов эволюционно удаленных от человека видов млекопитающих (не приматы) характерны относительно низкие значения H_{Align} (0.09 - 0.27), при сохранении достаточно высоких H_{Oli} (0.29 – 0.77), при этом величина отличия $D = H_{Oli} - H_{Align}$ варьировала от 20% до 45%. Можно предположить, что это объясняется сохранением в регуляторных районах тканеспецифичных генов эволюционно удаленных видов сходства только на уровне использования сходных регуляторных сигналов.

Поиск регулярных последовательностей в промоторах из геномов различных групп организмов с использованием критерия серий

Шеленков А.А., Коротков Е.В.

Центр «Биоинженерия» Российской академии наук, Москва, 117312, Россия

В настоящее время проводится широкомасштабный анализ последовательностей различных геномов, в частности, генома человека. Одной из важнейших задач этого анализа является характеристика и определение функций различных генов. Промотор – это участок генома, расположенный вблизи сайта инициации транскрипции и играющий ключевую роль в генетической регуляции. Обнаружение промоторов является важным шагом для проведения аннотации генов. Для того, чтобы разделить участки геномов, содержащие и не содержащие промоторы был использован целый ряд признаков, например CpG островки, TATA-боксы, CAAT-боксы, некоторые характерные сайты связывания факторов транскрипции, матрицы пентамеров, олигонуклеотиды, а также комбинированные подходы. Однако ни один из этих сигналов не описывает все разнообразие промоторов, и каждый признак, полученный на основе изучения промоторных последовательностей, имеет свои ограничения в использовании. В данной работе мы предлагаем новую характеристику символьных последовательностей – регулярность, приводим метод ее обнаружения, основанный на использовании критерия серий, а также применяем этот метод для поиска регулярных последовательностей в промоторах из геномов различных групп организмов. В работе использовались эукариотические промоторы из базы данных EPD, версия 93. Было выбрано 2236 последовательности, представляющих все группы организмов (1). Для проведения расчетов использовалась видеокарта NVIDIA с 64-мя потоковыми процессорами. Распараллеливание вычислений проводилось по входным данным, пакет программ был разработан на языке C. Проведенный поиск регулярных последовательностей в эукариотических промоторах показал, что более 60% из них обладают регулярностью на статистически значимом уровне. В работе обсуждаются возможные биологические функции регулярности и возможность использования данной характеристики для аннотации промоторов.

1. Shelenkov A, Korotkov E. Search of regular sequences in promoters from eukaryotic genomes. Computational Biology and Chemistry. 2009;33:196-204

Информационно-аналитическое сопровождение проекта «Протеом человека» с использованием персональных супер-ЭВМ

Старовойтов А.В., Галушкин А.И., Лисица А.В., Арчаков А.И.
ФГНУ Центр информационных технологий и систем органов исполнительной власти Минобрнауки России, НИИ биомедицинской химии им. В.Н. Ореховича РАМН, Центр «ИнтерЭВМ», технологическая компания «КуБ»
lisitsa.av@gmail.com

Высокая производительность методов исследования протеома вынуждает предусматривать программные средства не только для хранения информации, но и для ее аналитической обработки. Использование графических процессоров позволяет осуществить мониторинг актуальной медико-биологической информации, размещаемой в глобальных банках данных, а также обеспечивает персонифицированный подход к интерпретации накопленных данных.

Программно-технический комплекс на базе супер-ЭВМ регистрирует обращения пользователя к Веб-ресурсам по молекулярной биологии. Например, обращения к электронной библиотеке PubMed фиксируются в журнале в виде ссылок на научные статьи. Каждая публикация транслируется в относящиеся к ней тематические рубрики, анализ которых позволяет автоматизировано выявлять и предлагать пользователю нетривиальные гипотезы в области его профессиональной компетенции.

Связи между рубриками отображаются в виде семантических сетей. Чтобы обеспечить комфортное использование семантического анализа, сеть необходимо обновлять за время загрузки пользователем очередной Веб-страницы. Построение сети связано с одновременным выполнением одинаковых операций для большого числа рубрик, что соответствует классу задач, для решения которых применяется технология параллельных вычислений. Применение в данной ситуации графических процессоров уменьшает время работы программы на несколько порядков, что позволяет взаимодействовать с информационно-аналитической системой в интерактивном режиме.

Комплексные модели регуляции электронных потоков в субклеточных системах.

Рубин А.Б., Красильников П.М., Ризниченко Г.Ю.

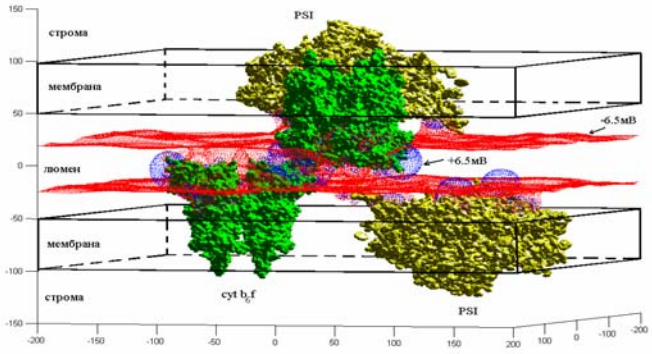
Кафедра биофизики Биологического факультета Московского государственного университета им. М.В.Ломоносова

В связи с проблемами разработки возобновляемых источников энергии на основе фотосинтезирующих организмов возникает задача регуляции направленности потоков электронов и оптимизации процессов преобразования энергии в фотосинтетической мембране. Целенаправленное воздействие (голодание, генные модификации, анаэробные условия) позволяют модифицировать систему первичных фотосинтетических процессов микроводорослей с целью увеличения продуктивности, изменения состава биомассы (увеличения содержания липидов), а также получения молекулярного водорода в качестве биотоплива.

Базовым процессом преобразования энергии при фотосинтезе является перенос электронов, сопряженный с переносом протонов и других ионов, который осуществляется белковыми макромолекулами в сложном интерьере фотосинтетической мембраны. Размеры реакционных объемов здесь сравнимы с размерами белков-переносчиков, в реакциях участвуют десятки-сотни молекул, поэтому традиционный кинетический подход при моделировании таких систем оказывается неэффективным.

На кафедре биофизики Биологического ф-та МГУ совместно с кафедрой компьютерных методов в физике Физического факультета МГУ разрабатывается новый подход прямого многочастичного моделирования процессов в фотосинтетической мембране. Модель представляет собой 3D-сцену, состоящую из люминального пространства, ограниченного двумя мембранами, вне которых находится стромальное пространство. ФС I, ФС II, цитохромный и другие комплексы встроены в мембрану, их форма и расположение соответствуют экспериментальным данным. Перенос электрона внутри комплексов описывается дифференциальными уравнениями для вероятностей состояний комплексов. Движение подвижных переносчиков (пластоцианина в люмене, пластохинона внутри мембраны и ферродоксина в строме) описывается законами броуновской динамики. Изучены детали электростатического взаимодействия отдельных белков-переносчиков: пластоцианина (Pc) и цитохрома f, ферредоксина и ФНР, а также взаимодействия ФС I с Pc на донорной стороне и белками ферредоксином и флаводоксином на акцепторной стороне в растворе и в люмене тилакоида. Белки представлены в виде твердых тел с пространственно фиксированными зарядами. Поступательное и вращательное движение белков являетя результатом их броуновского движения при дополнительном действии электростатических сил. Для пространственного описания электростатического потенциала, генерируемого вокруг белков, использован формализм Пуассона-Больцмана.

Прямое многочастичное моделирование фотосинтетичеких процессов позволяет выявить роль структурной организации в формировании кинетических сигналов, регистрируемых в биофизических экспериментах (флуорометрия, ЭПР и др.), и моделировать процессы в пространстве и времени на субклеточном уровне.



Модельная сцена процессов в фотосинтетической мембране

Экспериментальные методы исследования процессов протекающих в индивидуальных макромолекулах, а также в гетерогенных молекулярных ансамблях, обладают рядом существенных ограничений, как по временному, так и по пространственному разрешению. В то же время, методы молекулярного моделирования лишены таких ограничений и теоретически позволяют проводить исследования молекулярных систем с любым заданным пространственно/временным разрешением. Однако, вычислительная стоимость расчетов молекулярной динамика и квантовой химии, основных методов молекулярного моделирования, быстро возрастает с увеличением размеров исследуемых систем. В связи с этим очень остро встает вопрос увеличения производительности используемых для расчетов ЭВМ, а также снижения их стоимости.

На кафедре биофизики ведутся МД исследования диффузионных процессов в биологических мембранах. В частности исследуется проблема связи между быстрыми тепловыми движениями молекул и усредненными макроскопическими параметрами липидного бислоя. Также ведутся МД исследования изменения энтропии и энтальпии при фазовых переходах в липидных мембранах, и при диссоциации белковых комплексов. Для получения репрезентативных результатов в таких исследованиях необходимы МД расчеты молекулярных ансамблей, состоящих из сотен тысяч атомов. Из-за высокой ресурсоемкости такие расчеты можно осуществить только с использованием высокопроизводительных кластерных вычислительных комплексов. Альтернативой

такому решению могут стать более компактные и дешевые ЭВМ на основе графических карт Tesla. Так, для пакета молекулярно-динамических программ GROMACS, который мы используем в своих расчетах, в настоящее время успешно реализованы алгоритмы вычислений на графических картах, дающие существенный прирост производительности по сравнению с обычными процессорами. В частности, GROMACS на графической карте Tesla C1060 продемонстрировал 3-5-ти кратное преимущество над базовыми алгоритмами (2x Intel Cuad Core E5462) в расчетах электростатических взаимодействий, и 20-ти кратное преимущество при решении задачи формирования списка взаимодействующих атомов молекулярной системы. Эти данные позволяют сделать вывод, что появление стабильно функционирующей версии GROMACS для Tesla GPUs позволит, по крайней мере частично, проводить расчеты на локальных вычислительных станциях, не привлекая сторонние ресурсы.

Расшифровка генома человека: требования к вычислительному обеспечению.
Прохорчук Е.Б.

РНЦ «Курчатовский институт», г. Москва. Текст тезисов отсутствует

3D визуализация изображений, получаемых с электронного микроскопа.
Игнатенко А.В.

ВМК МГУ. Текст тезисов отсутствует

Ш – МОДЕЛИРОВАНИЕ

Исследование моделей жидкокристаллических фаз ДНК с помощью компьютерного моделирования методом Монте-Карло.

В.Н.Блинов, В.Л.Голо, А.А. Киреева, К.В.Шайтан,
Механико-математический факультет МГУ имени М.В. Ломоносова
voislav.golo@gmail.com

Растворы молекул ДНК при определённых условиях (ионный состав, температура, концентрация ДНК) могут образовывать жидкокристаллические фазы, а именно холестерики. Для моделирования формирования структуры применяются отрезки молекул, около 150 пар оснований. Длина этого фрагмента меньше персистентной длины ДНК. В этом случае молекула ДНК обладает достаточной жесткостью, чтобы её прямолинейная конформация существенно не менялась под действием тепловых флуктуаций.

Раствор ДНК рассматривался как ансамбль стержней, который не имеет трансляционной упорядоченности, как обычные кристаллы, но может иметь ориентационную упорядоченность. При этом существенно, что взаимодействие молекул ДНК имеет в значительной степени электростатическую природу, обусловленную наличием зарядов фосфатных групп и дипольных моментов пар оснований. Исследование процесса самоорганизации представляет непростую задачу и в теоретических работах применяются приближённые схемы. В работе рассматривался отрезок молекулы ДНК как

одномерная цепочка, оснащённая зарядами фосфатов и диполями пар оснований. При этом диполи направлены трансверсально к оси цепочки с условием соблюдения геликоидальной симметрии молекулы ДНК, т.е. структуры двойной спирали.

Исследование конфигурации ансамбля молекул с учётом температуры среды даже в таком относительно простом случае является достаточно сложной вычислительной задачей, которая решалась с использованием суперкомпьютеров. Следует отметить, что подробное исследование даже парного взаимодействия молекул ДНК не тривиально, поскольку речь идёт о нескольких десятков диполей, достаточно сложно расположенных в пространстве. Принимая во внимание геликоидальну симметрию молекулы необходимо исследовать структуру минимумов функции многих переменных достаточно общего вида. Стоит отметить, что исследование растворов ДНК и образования в них при определённых условиях жидкокристаллических фаз весьма актуальны для ряда областей биологии и нанобиотехнологии. Интерес к ним, в частности, стимулируется исследованиями структуры хроматина в молекулярной биологии, а также технологическими приложениями для создания компактных чувствительных сенсоров для биологически активных веществ.

Моделирование позволяет определить диапазон параметров системы, при которых формируется холестерическая фаза.

Для проводимых исследований пользовались ресурсы вычислительного кластера с графическими ускорителями кафедры биоинженерии Биологического факультета МГУ и и НИВЦ МГУ, а также суперкомпьютер МГУ «Чебышев». Использование графического ускорителя NVIDIA 1060 позволяло существенно увеличить скорость расчета.

Эффективная компьютерная программа для молекулярно динамического моделирования молекулярных систем на графических процессорах с визуализацией.

Засецкий А.Ю.

Физический факультет МГУ, *azasetsky@gmail.com*

Мы представляем компьютерную программу для молекулярно динамического (МД) моделирования на базе графических процессоров. Межмолекулярные силы вычисляются с помощью модифицированной версии *Fast N-Body* алгоритма [1]. В данной версии мы используем «скоростной» алгоритма Верле, в котором вращения молекул вычисляются с помощью метода кватернионов [2, 3]. Основным преимуществом такого подхода для потоковых процессоров является то, что он не требует итерационных процедур, в отличие от широко используемых итерационных алгоритмов интегрирования, таких «прыжок лягушки» или «скоростной» алгоритм Верле со связями Shake, Rattle и т.д. Интегралы движения, таким образом, могут быть интегрированы параллельно, где каждый отдельный процесс отвечает за обновление положения и ориентации одной молекулы. Используя особенности архитектуры GPU, мы добавили возможность визуализации исследуемых молекулярных систем, в "реальном времени" без какого либо уменьшения эффективности вычислений. Такая визуализация полезна как на стадии дизайна компьютерного эксперимента так и на этапе мониторинга временной эволюции изучаемых молекулярных систем, особенно в случае изучения неравновесных процессов. В текущей версии программы мы добились производительности, которая позволяет получить траектории длиной 28 нс в сутки для системы состоящей из более чем 4 000 атомов на графических процессорах NVIDIA с 240 АЛУ [4]. У нас есть стойкое ощущение, что все еще есть значительные возможности для совершенствования алгоритма и с нетерпением ожидаем выяснить как на предстоящем совещании.

1. R. G. Belleman, J. Bedorf, S. F. Portegies Zwart *New Astronomy* 13 (2008) 103–112

2. M. Hiyama, T. Kinjo, and Shi-aki Hyodo, *Journal of Physical Society of Japan* **77**, 064001 ((2008)
3. I. P. Omelyan, *Computational Physics* **12,** 97 (1998).
4. A.Y. Zasetsky, A.S. Lileev and A.K. Lyashchenko, *Molecular Physics*, 108, 5, 10, 2010, 649–656

Адаптация потенциала погруженного атома для молекулярно динамического моделирования конденсированных веществ на GPU.

Быстрый Р.Г., Казённов А.М., Морозов И.В**.,** Писарев В.В., Стегайлов В.В.

Объединенный институт высоких температур РАН, Москва
Московский физико-технический университет (НИУ), Долгопрудный, Московская обл.
morozov@ihed.ras.ru

Метод молекулярной динамики (МД) широко применяется для моделирования статических и динамических свойств твердых тел и жидкостей. Особое значение этот метод имеет для описания метастабильных состояний и фазовых переходов [1,2], возникающих при ударно-волновом нагружении, лазерной абляции и других интенсивных воздействиях. В настоящее время существуют большое количество программ МД моделирования (LAMMPS, NAMD и др.) позволяющих проводить такие расчеты, некоторые из них адаптированы для выполнения гибридных системах, включающих видеоускорители (GPU).

В настоящей работе обсуждается опыт использования GPU для ускорения МД расчетов гомогенной кристаллизации перегретого расплава алюминия. Для моделирования была использована свободно распространяемая программа HOOMD [3], на основе которой был реализован многочастичный полуэмпирический потенциал погруженного атома (EAM) [4]. Необходимость применения этого потенциала обусловлена тем, что он более точно, чем парные потенциалы, описывает взаимодействие атомов в кристалле, а также в расправе при приближении к границе кристаллизации.

Расчеты проводились на видеоускорителях NVIDIA GeForce 260GTX, GeForce 480GTX, Tesla C1060. Ускорение расчета МД траектории на GPU GF 480GTX по сравнению с CPU Intel Xeon E5520 составило 150-800% в зависимости от числа частиц. Это примерно в два раза меньше, чем ускорение при использовании парного потенциала Леннарда-Джонса.

Кристаллизация в МД модели переохлажденной жидкости носит случайный характер, поэтому для получения достоверных данных о частоте появления зародышей кристаллической фазы необходимо рассчитать множество МД траекторий и построить статистическое распределение по временам жизни метастабильного состояния [2]. Расчет этих траекторий может вестись независимо, что позволяет без потерь распараллелить его на системах с несколькими GPU или на гибридных кластерах.

В результате работы были определены характерные времена жизни переохлажденной жидкой фазы при различных давлениях и температурах, определен коэффициент самодиффузии жидкого алюминия как функция температуры и плотности.

1. Янилкин А.В., Жиляев П.А., Куксин А.Ю., Норман Г.Э., Писарев В.В., Стегайлов В.В. // Вычислительные методы и программирование. 2010. Т. 11. С. 111.
2. Kuksin A.Yu., Morozov I.V., Norman G.E., Stegailov V.V., Valuev I.A. // Molecular Simulation. 2005. V. 31. P. 1005.
3. Anderson J.A., Lorenz C.D., Travesset A. // J. Comput. Phys. 2008. V. 227. P. 5342.

4. Liu X.-Y., Wei Xu., Foiles S.M., Adams J.B. // Appl. Phys. Lett. 1998. V. 72. P. 1578.

Молекулярное моделирование фосфолипидных наночастиц с использованием гетерогенных вычислительных систем

А.Л. Чернобровкин[1*], В.А. Холина[1], Ю.В. Мирошниченко[1,2], А.В. Лисица[1]

[1]Институт биомедицинской химии имени В.Н.Ореховича РАМН, Москва

[2]ООО «КуБ»

* тел. (499)246-3731, chernobrovkin@gmail.com

Включение в состав фосфолипидной транспортной системы (ФЛТС) является перспективным способом повышения биодоступности лекарственных препаратов (ЛП). Ранее было показано существенное улучшение фармакокинетических параметров доксорубицина и индометацина в составе ФЛТС. Однако, размер наночастиц изменялся в достаточно широких пределах, формируя, как мицеллярные, так и липосомальные структуры. В тоже время представляется перспективным, меняя фосфолипидный состав системы, подобрать для произвольно выбранного лекарственного препарата оптимальный набор компонентов, которые бы самопроизвольно образовывали термодинамически устойчивые мицеллы минимально возможного размера с включениями ЛП. Такое конструирование может быть выполнено методами симуляции молекулярной динамики (МД) больших систем с использованием систем распределённых вычислений.

Нами были проведена симуляция молекулярной динамики фосфолипидных мицелл различного размера. Для создания трехмерных моделей мицелл использовался специальный алгоритм, располагающий единичные молекулы фосфатидилхолина (ДПФХ) на поверхности сферы заданного радиуса. Алгоритм позволял контролировать диаметр получаемой мицеллы и диаметр «полости» внутри нее. Для проведения вычислений использовалась программа NAMD, работающая на гетерогенном вычислительном сервере, включающем вычислительный модуль NVIDIA Tesla S1070. Моделирование производилось в полноатомной модели среды, с установленными периодическими граничными условиями, постоянной температуре 309 К и давлении 10^5 Па. Было показано, что мицеллярная структура комплекса молекул ДПФХ является устойчивой при диаметре мицелл от 6 до 12 нм, что соответствует комплексам от 60 до 360 молекул. Однако при увеличении диаметра мицеллы до 15 нм наблюдается существенное изменение структуры фосфолипидной частицы. Было также выполнено моделирование спонтанного мицеллообразования для ДПФХ в водном растворе для различных концентраций фосфолипидных молекул в системе. Был выявлен диапазон концентраций ДПФХ при которых происходит образование мицелл диаметром 6-8 нм.

Аппаратно-программная платформа CUBLIC для моделирования биологических систем.

Емельянов А.А., Терентьев А.Б.

ООО «Научно-исследовательский центр специальных вычислительных технологий»

http://hopcomp.net

В настоящее время всё сильнее проявляется необходимость в высокопроизводительных вычислениях над большими объёмами данных, а следовательно возрастает интерес к средствам, позволяющим реализовать такие вычисления. Одним из

таких средств являются современные графические процессоры, имеющие наилучшее на сегодняшний день соотношение цены, производительности и энергопотребления. Однако сложность разработки соответствующего эффективного программного обеспечения не позволяет этой технологии получить широкое распространение.

Целью и содержанием проекта CUBLIC является создание вычислительной платформы, позволяющей решать вычислительные задачи практически любой сложности без привлечения профессиональных программистов.

Аппаратное обеспечение проекта базируется на графических процессорах Nvidia и ATI. Программная платформа включает в себя три основных компонента: редактор алгоритмов, среда выполнения и система удалённого доступа через Internet. Редактор алгоритмов позволяет описывать любой вычислительный алгоритм как диаграмму, включающую предопределённые вычислительные примитивы (блоки), их свойства и связи. Блоки могут группироваться в модули, которые, в свою очередь, могут быть использованы наравне с блоками на любом уровне вложенности. Процесс низкоуровневого распараллеливания задачи скрыт от пользователя и выполняется автоматически в рамках среды выполнения. Таким образом, для решения прикладной задачи от специалиста-предметника требуется только владение основами математического моделирования.

В системе реализованы следующие виды элементарных блоков: блоки синхронизации, вектор-матричных операций, генерации, элементарных функций, решения СЛАУ, визуализации и блоки специализированных высокоуровневых алгоритмов. На базе вышеперечисленных блоков, реализуются модули, которые можно применять в решении сложных предметных задач в том числе и в области биологии.

Совместно со специалистами, занимающимися изучением биологических процессов головного мозга, с помощью CUBLIC были решены следующие задачи: моделирование внутреннего взаимодействия в нейральных сетях, динамики роста нейральных сетей, межнейронного взаимодействия с учётом периода рефрактерности, длины аксона, пластичности, а также реализованы алгоритмы исследования зависимости параметров нейральной сети от структуры её внутренних связей.

На сегодняшний день программный продукт работает на любых видеокартах Nvidia, поддерживающих технологию CUDA.

Текущая информация о проекте представлена на сайте http://hopcomp.net.

Реализация метода Монте-Карло с использованием графических процессоров для моделирования распространения излучения применительно к задачам биомедицинской диагностики.

М.Ю. Кириллин, И.И. Фикс.

Институт Прикладной Физики, РАН

В последнее время методы оптической биомедицинской диагностики получили широкое развитие, и многие из них переходят из класса экспериментальных в число реально применяющихся в клинической практике. Среди преимуществ методов оптической диагностики и визуализации следует отметить их неинвазивность, безопасность и экономичность (по сравнению с традиционными рентгеновскими методами и магнитно-резонасной томографией), и высокое пространственное разрешение (по сравнению с методами УЗИ). Ряд методов оптической диагностики при применении требуют восстановления трехмерного изображений по результатам регистрации рассеянного объектом излучения, а также интерпретации полученных изображений. Уравнение переноса излучения, описывающее распространение излучения в

рассеивающих средах является довольно сложным, и в общем случае не имеет аналитического решения, что требует применения различных приближений или его решения численными методами.

Одним из эффективных методов численного моделирования распространения излучения является метод Монте-Карло [1]. Применительно к данной задаче он заключается в многократном повторении расчета случайной траектории движения фотона в среде исходя из задаваемых параметров среды (геометрии и оптических свойств) и последующем статистическом анализе полученных данных. Преимуществом данного метода перед различными аналитическими приближениями является возможность расчета интенсивности рассеянного излучения как вблизи источника, так и на больших расстояниях при любой заданной геометрии среды. в то время как аналитические приближения существенно ограничены их областью применения. Единственным недостатком метода Монте-Карло (ММК) является его ресурсозатратность (так, для достижения удовлетворительной точности при моделировании распространения излучения в образце биоткани толщиной около 1 см обычно требуется порядка 10^8 случайных траекторий фотонов). Однако, это ограничение может быть эффективно преодолено с помощью использования графических процессоров, позволяющих осуществлять многопотоковые параллельные вычисления [2, 3].

Поскольку для вычисления случайные траектории фотонов являются независимыми друг от друга и определяются оптическими и геометрическими характеристиками объекта, их расчет может осуществляться параллельно без обмена данными между потоками. Этот факт делает реализацию ММК с применением графических процессоров очень выгодной.

В настоящей работе ММК был реализован для задачи восстановления трехмерного распределения флуорофора внутри биологического объекта в диффузионной флуоресцентной томографии.

Диффузионная флуоресцентная томография (ДФТ) представляет собой современный оптический метод визуализации внутренней структуры оптически неоднородных сред и объектов с использованием флуоресцентных маркеров [4]. Флуоресцентные маркеры различной природы используются для диагностики онкологических заболеваний, исследований молекулярных процессов, типичных для канцерогенеза, исследования метастазирования и ответа на противоопухолевую терапию [5]. Для точного определения местоположения и реальных размеров флуоресцирующей опухоли, находящейся в лабораторном животном на большой глубине (глубже кожного покрова), используют методы ДФТ [6]. В методе ДФТ облучение исследуемого объекта происходит на длине волны возбуждения флуоресцирующих веществ (маркеров), а детектирование сигнала - в спектре их флуоресценции. Для реконструкции распределения флуорофора в ткани в настоящее время используются специальные алгоритмы, учитывающие диффузное распространение света [7,8].

Метод Монте-Карло применительно к ДФТ может применяться для решения следующих задач: расчет распространения света внутри объекта со «сложной» геометрией; проведение численных экспериментов (моделирования результатов эксперимента) для апробации алгоритмов восстановления; восстановление оптических свойств среды (коэффициентов поглощения, рассеяния, фактора анизотропии).

Ниже представлены характерные времена, требуемые для проведения расчета метода Монте-Карло на GPU (Nvidia GeForce 260) с использованием технологии CUDA и CPU (AMD Phenom II x4 920, 2.7 ГГц, однопотоковая реализация).

Количество траекторий	Время счета на GPU, с	Время счета на CPU, с
10^8	138	3932
10^9	1338	39538
10^{10}	13206	—

Таблица 1. Сравнение времени расчета при реализации ММК на GPU и CPU

При использовании технологии CUDA одновременно просчитывается около $2 \cdot 10^6$ различных траекторий фотонов. Каждая нить GPU просчитывает 1 траекторию. Проверка «граничных» условий, таких как: проверка выхождения фотона за пределы объекта, выполнение условий детектирования фотона и пр. осуществляется на CPU.

Таким образом, разработанный алгоритм поможет повысить точность восстановления трехмерного распределения флуорофора внутри биологических объектов. Однако, применения настоящего алгоритма не ограничиваются задачей восстановления в ДФТ, он может быть применен для моделирования сигналов и изображений различных видов оптической биомедицинской диагностики (например, оптических диффузионной и когерентной томографий), а также для планирования терапии различными видами излучения.

1. Wang L., Jacques S. L., Zheng L. “MCML--Monte Carlo modeling of light transport in multi-layered tissues”, *Comput Methods Programs Biomed*, **47**(2), 131-146, (1995).

2. Alerstam E., Svensson T., Andersson-Engels S. Parallel computing with graphics processing units for high-speed Monte Carlo simulation of photon migration, *J Biomed Opt*, **13**(6), 060504 (2008).

3. Q.Fang and D.A. Boas, Monte Carlo simulation of photon migration in 3D turbid media accelerated by graphics processing units, *Optics Express*, Vol. 17, Issue 22, pp. 20178-20190 (2009)

4. S.V. Patwardhan, S.R. Bloch, S. Achilefu, J.P. Culver, “Time-dependent whole-body fluorescence tomography of probe bio-distribution in mice”, *Optics Express*, **13**, 2564-2577 (2005).

5. V. Ntziachristos, E.A. Schellenberger, J. Ripoll, D. Yessayan, E. Graves, A. Bogdanov, L. Josephson, and R. Weissleder “Visualization of antitumor treatment by means of fluorescence molecular tomography with an annexin V–Cy5.5 conjugate”, *PNAS*, **101**, 12294–12299 (2004).

6. V. Ntziachristos, R. Weissleder “Experimental three-dimensional fluorescence reconstruction of diffuse media by use of a normalized Born approximation”, *Optics Letters*, **26**, 893-895 (2001).

7. Ф. Натеррер, *Математические аспекты компьютерной томографии*. М.: Мир, 1990.

8. A. Soubret, V. Ntziachristos “Fluorescence molecular tomography in the presence of background fluorescence”, *Physics in Medicine Biology*, **51**, 3983-4001 (2006).

Применение графических процессоров для ускорения расчетов при решении сложных задач математической физики.

Коробкова С.В. (компания ВОКОРД, svetlana_kor@pisem.net),
Скрибцов П.В. (ООО Павлин Технологии)

Решение задач математической физики актуально во многих областях и имеет множество практических применений, таких как аэродинамики, гидродинамика, электродинамика, моделирование различных экономических и природных процессов и явлений.

Исходные данные во многих задачах математической физики имеют сверхбольшую размерность – порядка 10^9 параметров, что часто делает невозможным или очень длительным процесс расчета на персональном компьютере. Таким образом, можно сделать вывод, что требуются системы и программные средства позволяющие ускорять решение сложных задач математической физики.

В последние несколько лет распространение получила реализация вычислений на графических процессорах, в частности, разработан ряд пакетов для решения определенных часто встречающихся задач. Среди них можно выделить следующие программные пакеты:

1. **OpenCurrent** - решение дифференциальных уравнений в частных производных на регулярной (1D, 2D, 3D) сетке;
2. **nHD** - Решение уравнения Эйлера для идеального сжимаемого газа методом Годунова;
3. **Stochastic Differential Equations with CUDA** - Численное интегрирование стохастических дифференциальных уравнений;
4. **3D Particle Boltzmann Solver** - Решение уравнения Больцмана методом частиц;
5. **CUDA-PSO** - Реализация метода частиц;
6. **Fast and Exact Solution of Total Variation Models on the GPU** - Решение уравнений Эйлера-Лагранжа вариационным методом;
7. **2D FDTD Wave Propagation** - Решение волнового уравнения на 2D сетке;
8. **MIDG** – реализация метода Галеркина для графических процессоров.

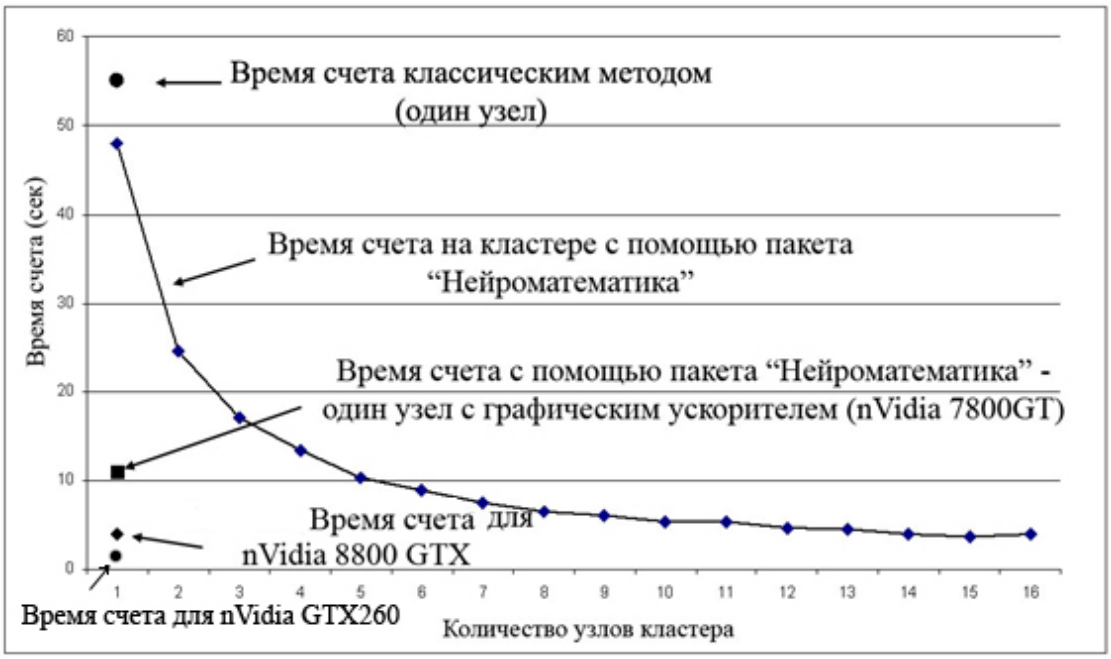
В рамках работы над проектом «Нейроматематика» (пакет программ решения научных задач на базе нейросетевых технологий), были разработаны модули решения задач аэродинамики. Для моделирования аэродинамических процессов в некоторой области была использована клеточная нейронная сеть специальной структуры.

В качестве тестовых задач были выбраны задачи аэродинамики двух видов:

- требующие учета вязкости (описываемые схемой Навье-Стокса) - задача обдува крыла самолета в процессе движения;
- не требующие учета вязкости газа (для таких задач достаточно схемы Эйлера) – задача расчета ветровой обстановки в застроенном регионе (на примере здания Сиднейской оперы)

Для тестирования производительности системы были реализованы 3 варианта аппаратной поддержки выполнения вычислений – CPU-реализация, реализация на кластерной системе с распределенной памятью и GPU-реализация.

Оценки времени, требуемого для нейросетевого решения данной задачи математической физики, представлены на следующем рисунке:



1. Галушкин А.И. Теория нейронных сетей. М.: Радиотехника, 2000. 416 с.
2. Галушкин А.И., Судариков В. А., Шабанов Е. Н. Нейроматематика: методы решения задач на нейрокомпьютерах. Математическое моделирование. М., 1991,- Т. 3, N8, с.101-106.
3. Коллективная монография под редакцией Галушкина А.И. Нейроматематика. М.: Радиотехника, 2002. 448 с.
4. Белоцерковский О.М., Давыдов Ю.М. Метод крупных частиц в газовой динамике. Вычислительный эксперимент. М.: Наука, 1982. 391 с.
5. Харлоу Ф. Численный метод частиц в ячейках для гидродинамики. Вычислительные методы в гидродинамике. М.: Мир, 1967. с.316-342.
6. Alexander Guzhva, Sergey Dolenko, and Igor Persiantsev Multifold Acceleration of Neural Network Computations Using GPU ICANN 2009, Part I, LNCS 5768, pp. 373-380, 2009.
7. Галушкин А.И., Коробкова С.В. и др., Программный комплекс «Нейроматематика», Рабочий проект ОКР по теме ИТ-13.3/001 «Разработка прикладных технологий для инженерных задач с плотными системами уравнений и сверхбольшим числом неизвестных», Москва, 2006

Качественная визуализация на основе шейдеров.

Гаврилов Н.И., Турлапов В.Е.

Нижегородский Государственный Университет им. Н.И. Лобачевского

В научной визуализации часто приходится иметь дело с данными, заданными в пространстве. Такими данными в физике часто являются наборы трёхмерных матриц, представляющие поля в различных физических процессах. Эти поля, как правило,

получают в результате численного эксперимента на кластерных системах. В медицине также существуют трёхмерные данные. Результатом компьютерной томографии является множество слоёв, т.е. двумерных массивов данных, которые вместе образуют трехмерное скалярное поле.

Научная визуализация должна помочь извлечь максимум полезной информации из таких данных. Объем данных, которые нужно визуализировать в реальном времени или интерактивно обычно равен 0.25-1ГБ. Реальное время означает вывод на экран не менее 25-30 кадров в секунду. Поэтому визуализация в реальном времени, с возможностью быстрой смены ракурса, смены параметров визуализации и т.д., при указанном объеме данных представляет определенную проблему.

Некоторые методы визуализации

Объёмный рендеринг – это одна из техник визуализации трёхмерного массива данных. В медицине обычно оперируют поперечными сечениями объёмного снимка, регулируя только яркость и контрастность чёрно-белого изображения. Такой способ мед-визуализации удобен, информативен и его можно считать традиционным, однако графические процессоры (GPU) уже позволяют достаточно быстро работать с объёмными снимками в пространстве.

Наибольшего качества и информативности объёмного рендеринга позволяет достичь метод обратной трассировки лучей. Для определения каждого пиксела искомого изображения генерируется луч, представляющий из себя точку в пространстве (например, позиция наблюдателя) и направление. Двигаясь в этом направлении через объём данных с некоторым шагом, луч “накапливает” цвет пиксела. С помощью этого метода можно визуализировать данные как набор изоповерхностей или туман. Изначально трассировка лучей не была методом, рассчитанным на работу в реальном времени, т.к. его реализации были только на CPU.

Реализация трассировки лучей для объёмной визуализации (рендеринга) на GPU

Наиболее удобной структурой хранения пространственных массивов данных на GPU является 3D-текстура, аналогичная 3D-массиву действительных чисел. Размер такой текстуры сегодня ограничен величиной 512x512x512 (128 миллионов значений). Для CPU удобнее может оказаться иерархическая структура, например октодерево. Для доступа к элементам дерева процессору необходимо пройти множество ветвлений, что приемлемо для CPU, но убивает производительность для GPU, к тому же, в программах для GPU нет рекурсии. Всё это плата за высокую производительность GPU. Программа, реализующая трассировку выполнена в среде MS Visual Studio 2008 как шейдерная программа для GPU, в среде OpenGL и GLSL (шейдерный язык для OpenGL).

Предложен метод ускорения вычислений, заключающийся в использовании иерархии текстур: помимо основного массива, 512x512x512, в GPU загружается меньший массив – двухканальная текстура 16x16x16. В пространстве каждому элементу этой текстуры соответствует свой блок 32x32x32 исходных данных. Текчтура хранит минимальное и максимальное значения данных соответствующего блока.

В итоге, обращаясь к маленькой текстуре, луч может пропускать «неинтересные» области пространства, двигаясь с большим шагом. Около 20-30% лучей так и не найдут “интересные” области, 50-80% будут в ней находится небольшое число шагов (например, до столкновения с изоповерхностью, либо при насыщении, т.е. непрозрачности получившегося цвета).

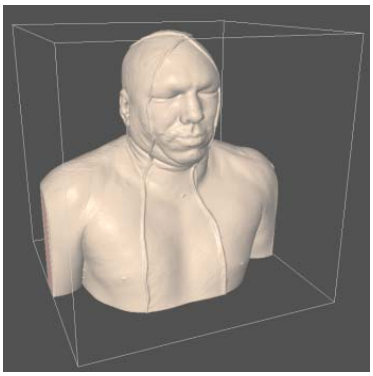
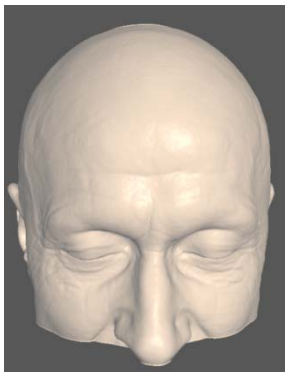
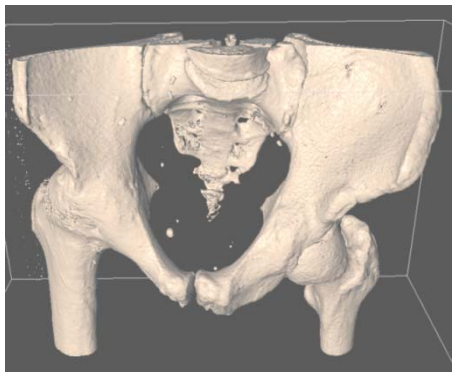
Увеличения производительности также можно добиться путём введения дополнительного экранного буфера, в котором можно хранить информацию о каждом луче. Эта информация должна сократить количество итераций для каждого луча, а главное

– улучшить качество изображения, сведя к минимуму “промахи” лучей мимо изоповерхности из-за недостаточно малого шага. Для визуализации данных, которые не помещаются в одну текстуру, можно работать с несколькими текстурами. Распределение массива данных по нескольким текстурам также может ускорить доступ к ним из-за их меньшего размера.

Результаты

Тестовые данные предоставлены ИПФ РАН, Московским мединститутом имени И.М. Сеченова, а также получены из Интернет-ресурса [1]. Это как результаты численного моделирования в физике, так и результаты компьютерной томографии. В физике достаточно массива данных 128^3, хотя в самом численном моделировании использовалась сетка 512^3. Для анализа CT-снимка нужно минимум 256^3, а лучше 512^3. Ниже представлена производительность, полученная на видеокарте GeForce 8600 GT.

Размер	128^3	256^3	512^3	512x512x690
FPS	50-60	30-40	10-15	7-10

Представленные результаты показывают, что предложенный подход позволил реализовать интерактивную визуализацию трехмерного поля данных большого объема.

1. The scientific computing and imagine institute at the university of Utah http://www.sci.utah.edu/download/IV3DData.html
2. Friedemann Roßler, Eduardo Tejada, Thomas Fangmeier, Thomas Ertl, Markus Knauff, *GPU-based Multi-Volume Rendering for the Visualization of Functional Brain Images*// Institute for Visualization and Interactive Systems, University of Stuttgart, D-70569 Stuttgart, Center for Cognitive Science, University of Freiburg, Germany, 2006 http://www.vis.uni-stuttgart.de/ger/research/pub/pub2006/simvis06-roessler.pdf
3. Ralph Brecheisen, Anna Vilanova i Bartroli, Bram Platel, Bart ter Haar Romeny. *Flexible GPU-Based Multi-Volume Ray-Casting* // Technical University of Eindhoven http://www.yp.wtb.tue.nl/pdfs/9881.pdf

GLSL и OPENCL – сравнительный опыт вычислений

Д.К. Боголепов[1], М.М. Захаров[2], О.Д. Блохин[1], Д.П. Сопин[1]

[1]*Нижегородский государственный университет им. Лобачевского*
[2]*Нижегородский государственный технический университет им. Алексеева*

Введение. OpenCL является *первым* открытым межплатформенным стандартом для параллельных вычислений на современных центральных процессорах и графических ускорителях. Целью данной работы являлось исследование программируемости

графической аппаратуры с помощью OpenCL и оценка производительности в сравнении с другими решениями: шейдерные языки и технология CUDA от компании NVIDIA. В качестве тестовой была выбрана задача моделирования динамики N точечных зарядов, помещенных в магнитное поле, при этом рассчитывались взаимодействия между *всеми парами* частиц. Основными физическими моделями являются закон Кулона и сила Лоренца. Для интегрирования системы дифференциальных уравнений использовался метод Эйлера, который является стандратным для тестовых программ такого рода.

Теоретическая оценка производительности. Перед анализом фактической производительности полезно дать некоторые теоретические оценки. Для этого следует выделить основной вычислительный код и оценить число *тактов* и *элементарных операций*, которые затрачиваются на его выполнение. Данные величины в общем случае *различны*: число тактов зависит от архитектуры, в то время как число элементарных операций является характеристикой алгоритма. В рассматриваемой задаче *основной* объем вычислений связан с расчетом *ускорения* частицы под воздействием остальных частиц системы. Далее приводится псевдокод соответствующей функции:

Функция расчета ускорения для одной частицы p и q – положение и заряд текущей частицы	Элементарных операций	Циклов ГПУ
`1 | for ( int i = 0; i < N; ++i ) {`	–	–
`2 |  float3 r = p – p [i];`	3	3
`3 |  float dis = dot ( r, r );`	5	3
`4 |  float inv = inversesqrt ( dis + SOFTENING );`	2	2
`5 |  float cube = inv * inv * inv;`	2	2
`6 |  float s = q * q [i] * cube;`	2	2
`7 |  acc += s * r;`	6	3
`8 | }`	–	–
Итого	20	15

В данной работе функции вычисления обратного квадратного корня (строка *5*) назначается *минимальная возможная* оценка в *один* такт, что позволяет оценить производительность в GFLOPS *снизу* и число итераций в секунду *сверху*. Необходимо отметить, что современные ГПУ поддерживают инструкцию *mad*, выполняя сложение и умножение за *один* такт (строки *4* и *8*).

Поскольку тело цикла в функции расчета ускорения выполняется N раз, на обработку одной частицы затрачивается $20 \cdot N$ элементарных операций, в то время как расчет одной итерации требует $20 \cdot N^2$ элементарных операций.

В качестве примера, построим временную оценку одной итерации для графической карты NVIDIA Quadro FX 5600 1.5 Гб и системы из $N = 2^{14} = 16384$:

$$IPS = \frac{\textit{Число ядер} \times \textit{Частота процессора}}{\textit{Число частиц} \times \textit{Число тактов на частицу}} = \frac{128 \times 1350\ \textit{МГц}}{16384 \times 16384 \times 15} \approx 43\ \textit{итерации} / c\ .$$

Результаты экспериментов. Для проведения экспериментов на базе технологии CUDA и шейдерного языка GLSL использовались реализации [1] и [2] соответственно.

Таблица 1. Производительность в задаче N тел (ATI/AMD Radeon HD 4890 1 Гб, драйвер 10.3)

Число частиц	OpenCL		GLSL	
	FPS	GFLOPS	FPS	GFLOPS
2048	316	26.5	3109.0	260.71

4096	**152.9**	**51.3**	**1617.0**	**542.57**
8192	**67.3**	**90.3**	**392.2**	**526.40**
16384	**16.9**	**90.7**	**106.9**	**573.91**

Таблица 2. Производительность в задаче *N* тел (NVIDIA Quadro FX 5600 1.5 Гб, драйвер 196.21)

Число частиц	OpenCL		CUDA		GLSL	
	FPS	GFLOPS	FPS	GFLOPS	FPS	GFLOPS
2048	**1147.7**	**96.27**	**1398**	**117.27**	**907.6**	**76.13**
4096	**583.8**	**195.89**	**672.19**	**225.55**	**319.9**	**107.34**
8192	**148.3**	**199.04**	**172.44**	**231.45**	**106.9**	**143.47**
16384	**37.3**	**200.25**	**43.39**	**232.98**	**27.8**	**149.25**

Заключение. Шейдеры обеспечивают минимальные возможности в задачах общего назначения (в частности, не поддерживается *локальная* память), а платформы OpenCL и CUDA практически идентичны. Поэтому от шейдерной реализации следует ожидать наименьшей производительности, в то время как CUDA и OpenCL должны демонстрировать схожие результаты. Данные соображения в целом реализуются для карты NVIDIA Quadro FX 5600 1.5 Гб. Незначительное отставание OpenCL можно объяснить недостаточной проработкой текущей реализации. При использовании CUDA достигается пиковая оценка производительности в *данной* задаче; при использовании OpenCL достигается 87% от пика. Для карты ATI/AMD Radeon 4890 1 Гб производительность находится на крайне низком уровне: в качестве базового ориентира такой оценки мы рассматриваем производительность шейдеров – 570 GFLOPS. Столь скромный результат объясняется еще и тем, что семейство ускорителей ATI/AMD Radeon 4800 *не поддерживает* локальную память на физическом уровне – она отображается на область глобальной памяти.

Платформа OpenCL продолжает активно развиваться и может претендовать на роль стандарта для параллельных вычислений на гетерогенных системах. Мы столкнулись с рядом проблем текущих реализаций (например, реализация NVIDIA не поддерживает исполнение на центральном процессоре, а платформа ATI/AMD не поддерживает текстуры), однако рекомендовать OpenCL к использованию и получать приемлемые результаты можно уже сейчас.

1. Erich Elsen V., Vishal Mike Houston и др. N-Body Simulations on GPUs. http://arxiv.org/pdf/0706.3060
2. Боголепов Д.К., Турлапов В.Е. Вычисления общего назначения на графических процессорах с использованием шейдерных языков // Труды международной научной конференции “Параллельные вычислительные технологии”, Нижний Новгород, 30 марта – 3 апреля 2009 г., с. 339-410.

http://omega.sp.susu.ac.ru/books/conference/PaVT2009/papers/short_papers/012.pdf

Разработка гибридной высокопроизводительной вычислительной платформы для прикладного компьютерного моделирования задач взаимодействия макромолекул

И.И. Морозов[1], И.Б Коваленко[3], В.А. Барсегов[1,2], А.А. Жмуров[1,2], А.М. Казенов[1], С.В. Трифонов[1], Я.А. Холодов[1].

[1] Кафедра вычислительной математики, Московский физико-технический институт, Институтский пер. 9, Долгопрудный, МО, Россия, 141700

[2] Department of Chemistry, University of Massachusetts, Lowell, MA 01854

[3] Биологический ф-т, кафедра биофизики, Московский государственный университет им. М.В.Ломоносова, Ленинские горы, г. Москва, Россия, 119992

IvanMorozov@rambler.ru, kovalenko78@mail.ru

На сегодняшний день существует достаточно большое количество разнообразных систем моделирования задач молекулярной динамики с различной производительностью, областью применимости, возможностями настройки и функциональностью. Обзор имеющихся пакетов и их функциональности можно посмотреть, например, на сайте:

http://en.wikipedia.org/wiki/List_of_software_for_molecular_mechanics_modeling.

В данном проекте реализуется программная платформа, функциональность которой рассчитана для детального численного моделирования определенного класса прикладных задач взаимодействия макромолекул.

Нами предложены и реализованы некоторые основные идеи для увеличения производительности разрабатываемой системы и её функциональности:

- отдельно реализовано пользовательское приложение, которое включает в себя графическую систему работы с данными;
- применяется многомасштабное иерархическое разбиение рассчитываемой системы по различным классам моделей и уровням детализации;
- как расчетная платформа берется гибридная высокопроизводительная архитектура, основанная на одновременном использовании графических вычислительных модулей совместно с многоядерными системами с общей и распределенной памятью.

Существует также ряд актуальных проблем, которые возникают при разработке и поддержке такого рода пакетов, а именно:

- производительность;
- расширяемость функционала;
- доступность для использования сторонними исследователями.

В настоящее время основное внимание при реализации данного проекта уделяется следующим прикладным направлениям и задачам:

- моделирования образования белок-белковых комплексов;
- моделирование денатурации больших молекулярных комплексов под воздействием механической силы;
- моделированию поведения вирусных капсул.

В последние годы на кафедре биофизики Биологического факультета МГУ разработан комплекс прямых многочастичных компьютерных моделей, описывающих

процессы взаимодействия белков – компонентов фотосинтетической цепи. Моделируемые молекулы перемещаются в соответствии с законами броуновской динамики, ориентируются друг относительно друга в электростатическом поле и образуют комплексы на трехмерной компьютерной сцене, сконструированной в соответствии с нашими знаниями о пространственной организации фотосинтетической мембраны. Адекватность модели проверяется соответствием кинетических характеристик наблюдаемых процессов экспериментальным данным. Преимуществом прямого моделирования является возможность изучения влияния формы и геометрических характеристик реакционного объема, числа и расположения взаимодействующих молекул, ионной силы и pH раствора и других характеристик на скорость изучаемой реакции взаимодействия белков. Прямое многочастичное моделирование позволяет понять, каким образом физические механизмы молекулярных взаимодействий (диффузия, электростатика и проч.) определяют общую динамику процессов в клетке.

Разрабатываемая программная платформа позволит проводить прямое многочастичное моделирование намного быстрее и с большей точностью и достоверностью получать значения скоростей изучаемых реакций взаимодействия белков.

Опыт применения графических вычислений в задачах молекулярной биофизики

Кондратьев М.С., Кабанов А.В., Самченко А.А., Хечинашвили Н.Н., Комаров В.М.

Институт биофизики клетки РАН, Россия, 142290, г. Пущино.

Тел. 8(4967)73-94-04, ma-ko@bk.ru

В течение нескольких последних лет, благодаря, главным образом, появлению вычислений на графических процессорах, произошло существенное расширение возможностей для проведения расчетов в рамках исследований структуры и динамического поведения биомолекул методами компьютерной химии. По-своему уникальным подразделением ИБК РАН Пущинского научного центра является Лаборатория структуры и динамики биомолекулярных систем, поскольку в моделировании здесь используется наиболее широкий арсенал методов: квантово-химические, молекулярно-динамические, гибридные и Монте-Карло. В рамках некоторых направлений исследований нами используются расчетные методы на GPU NVIDIA. Остановимся кратко на ряде работ.

Несколько лет назад коллективом авторов [Хечинашвили и др. 2006] был предложен возможный механизм повышения термостабильности малых глобулярных белков – за счет увеличения количества водородных связей между боковыми группами аминокислотных остатков на поверхности молекулы. С целью использования выявленных закономерностей в создании биокатализаторов-ферментов с более высокой термостабильностью, нами был выполнен ряд работ, первой из которых стало теоретическое исследование пероксиредоксина 6 человека (1PRX) и попытка повышения его термостабильности. На основании данных по пространственной структуре этого сравнительно недавно открытого представителя класса антиокислительных ферментов и его гомологов, а также исходя из представлений о механизме повышения термостабильности белков, нами была предложена и изучена новая модификация белка (с заменами лишь четырех аминокислотных остатков). Они должна обладать повышенной термостабильностью - за счет увеличенного количества точек для образования альтернативных водородных связей на поверхности глобулы. Для нативной и модифицированной форм пероксиредоксина 6 на системах с GPU Nvidia были рассчитаны траектории молекулярной динамики (при разных температурах), анализ которых подтвердил наличие повышенной (по сравнению с исходным ферментом) устойчивости предложенных модификаций пероксиредоксина 6

человека к тепловой денатурации. Сейчас белок с предложенными нами аминокислотными заменами уже синтезирован и проходит экспериментальную проверку на активность и термостабильность. По аналогичной схеме в настоящий момент выполняются исследования ряда малых липаз, некоторые из которых обладают способностью эффективно разлагать пластмассы.

Вычисления на графических процессорах используются и в решении фундаментальных биофизических проблем, таких как, например, определение предпочтительной конформации и локализации пептидных фрагментов на ДНК, что важно для регуляции работы фермента РНК-полимеразы. Позиционирование этого фермента осуществляется при участии гибкого междоменного линкера, соединяющего C- и N-концевые домены альфы, однако в настоящее время этот пептидный линкер практически не изучен. В ходе исследования наиболее вероятных конформационных состояний изучаемого пептидного фрагмента было рассмотрено несколько вариантов топологий хода цепи и сделано предположение о его преимущественно альфа-спиральном строении. В результате последующего GPU-расчета жесткого докинга линкера с поверхностью участка ДНК была выявлена преимущественная локализация пептида на малой бороздке нуклеиновой кислоты, с примыканием к рецептору положительно заряженной стороной альфа-спирали. Изучение сворачивания самой альфа-спирали представляло собой еще одну область применения расчетов на графических процессорах.

До недавнего времени, несмотря на существование шаблонов альфа-спирали (набора аминокислот, обычно входящих в ее состав), механизм перестроек структуры линейной цепи пептида, приводящий к образованию спирали, оставался недостаточно детально описанным. Выполненные нами расчеты и анализ структуры аминокислот и пептидов, а также изучение динамического поведения отдельных последовательностей и сравнение формализованных временных и энтальпийно-энтропийных характеристик таких процессов, позволили констатировать реализацию двух путей спирализации – прямого (от N- к C-концу цепи, что совпадает с направлением выхода пептида из рибосомы) и обратного (от C- к N-концу). Примечательно, что выбор того или иного направления процесса зависит от типа аминокислотного остатка, его запускающего – кислого или основного, ярко выраженной особенностью боковых групп которых является эффективное образование бифуркационных (вилочковых) водородных связей с остовом основной цепи.

Показано, что оба способа альфа-спирализации, прямая и обратная, реализуются по единому механизму, который сопровождается заметным снижением не только полной, но и гиббсовой энергии.

Отыскание глобального минимума функции потенциальной энергии атомного кластера Морса.

Деянов Р.З.

МГУ имени М.В.Ломоносова, факультет ВМК

deyanov@math-lab.ru

Подавляющее число задач обработки данных эксперимента (также и численного) требуют применения методов минимизации. Так, в методе МНК наименьшая невязка теоретической модели и данных эксперимента обеспечивает наилучшие значения искомых параметров. Целый класс задач, моделирующих динамические процессы, требуют также поиска оптимальных значений параметров модели, исходя из принципа минимума потенциальной энергии. При этом, достижение глобального минимума является принци-

пиальным. Для этих целей разработан и апробирован алгоритм поиска глобального минимума функции многих переменных, когда число локальных минимумов конечно.

В работах [1-3] дано обоснование метода, приведены тестовые примеры. Суть алгоритма сводится к построению стратегии выхода из точки локального минимума (куда легко попадаем посредством стандартной программы) в область притяжения другой точки локального минимума со значением функции меньшим, чем в предыдущей.

Эффективность работы алгоритма покажем на примере определения оптимальной конфигурации атомного кластера, дающей минимум потенциала Морса. Выражение потенциала Морса(безразмерное) для n атомов, согласно [4], рассматривалось как:

$$f(x_1,...,x_n;\ \rho) = \sum_{i=1}^{n-1}\sum_{j=i+1}^{n}[(\ \mathrm{e}^{\rho(1-\|x_i-x_j\|)} - 1)^2 - 1]$$

Значение параметра ρ задается исходя из сорта атома. Задача состоит в определении декартовых координат n атомов - $x_k \in \mathbf{R}^3$, всего $3n$ параметров. Для сравнения получаемых результатов рассматривались данные Cambridge Cluster Database (http://physchem .ox.ac.uk/~doye/jon/structures/Morse/tables.html) . Для n=80 и ρ=3 в этой таблице приведено значение потенциала: -690.577890 . У нас получилось значение:

-690.588928 . Указанная таблица заканчивается при n=147 . Для n=148 у нас получено:

-1544.389404 (ρ=3). Счет проводился на ПК с частотой процессора 2.94GHz. Глобальный минимум находился за считанные минуты, при этом, основное время уходило на поиск более меньшего значения функции, которого уже не существовало. Полученные нами координаты атомов можно скачать на сайте http://math-lab.ru/ .

Будем признательны за новые постановки практических проблемных задач минимизации с большим количеством переменных.

1. Деянов Р.З. Программа глобальной минимизации R-фактора в методе газовой электронографии. / Алгоритмы и программы. Информ. бюлл. ВНИЦТИ, П006154, N3(54), с.50, 1983.
2. Деянов Р.З., Щедрин Б.М. Поиск глобального минимума функции нескольких переменных, заданной на компакте. /Сб.: Библиотека программ по структурному анализу. – М.: изд-во Моск. Ун-та, с.133-140, 1984.
3. Деянов Р.З, Щедрин Б.М. Алгоритм последовательного спуска по системе локальных минимумов. Прикладная математика и информатика № 30 , М.: Изд-во факультета ВМиК МГУ, с.46-54, 2008.
4. J.P.K.Doye, D.J.Wales. Structural Consequences of the Range of the Interatomic Potential: a Menagerie of Clusters. J. Chem. Soc., Faraday Trans., 93, 4233, 1997.

Возможности высокопроизводительной вычислительной инфраструктуры МФТИ для решения прикладных задач численного моделирования

А.М. Казенов[1], А.Е. Алексеенко[1], И.И. Морозов[1], Я.А. Холодов[1].

[1] Научно-образовательный центр по высокопроизводительным вычислениям, Московский физико-технический институт, Институтский пер. 9, Долгопрудный, МО, Россия, 141700

kazennov@gmail.com

В последнее время высокопроизводительные вычислительные системы стали неотъемлемой частью научной жизни. Практически во всех областях науки, так или иначе, возникают задачи численного моделирования или численной обработки данных. Для решения такого рода задач часто требуются большие вычислительные мощности, поэтому создаются различного вида вычислительные кластеры и суперкомпьютеры.

В данном докладе основной акцент будет сделан на возможностях достаточно нового типа высокопроизводительных вычислительных систем, а именно – гибридных. Под гибридной понимается система, которая в качестве основного вычислительного ресурса использует не центральный многоядерный процессор, а некие другие устройства, взаимодействующие с ним. В нашем случае используется связка центральный процессор плюс графические вычислительные модули (ГВМ).

Так как данная технология появилась сравнительно недавно, то не существует единого принятого способа её тестирования, так же как не существует единого мнения, по поводу того, какая схема использования ГВМ наиболее эффективна. Поэтому целью данной работы было проведение ряда синтетический и прикладных тестов для исследования различных схем использования ГВМ при проведении научных расчетов.

В 2009-2010 гг. МФТИ были приобретены три вычислительные системы:

1. Гибридный вычислительный кластер факультета молекулярной и биологический физики, который состоит из пяти вычислительных узлов следующей конфигурации: MB - Tyan S2915A2NRF-E; 2 x CPU - AMD 2427(Six-core 2.2GHz); RAM - 16Гб SEC M393T5660QZA; 2 x GPU - Tesla C1060; Net - Mellanox MHQH19-XTC, D-Link DGE-528T.
2. Восьми-карточная вычислительная машина кафедры вычислительной математики: MB - TYAN S7025; 2 x CPU - Intel 5520 (Eight-core 2.3GHz); RAM - 32Гб DDR3Dimm 10660 ECC Registered Kingston; 8 x GPU - GeForce 480GTX.
3. Четырех-карточная вычислительная машина кафедры вычислительной математики: MB - TYAN Transport FT72; 2 x CPU - Intel 5520 (Eight-core 2.3GHz); RAM - 32Гб DDR3Dimm 10660 ECC Registered Kingston; 4 x GPU - Radeon 5870.

Тестирование проводилось четырьмя различными тестами. Тесты были выбраны, так что бы покрыть как можно больший спектр задач по моделированию и обработке данных. 1) CPU HPL является общепринятым тестом для проверки работы центрального процессора. 2) The Scalable Heterogeneous Computing (SHOC) benchmark suite является набором синтетических тестов для проверки работоспособности ГВМ с различными технологиями (CUDA, OpenCl) в том числе на гибридных вычислительных системах. 3) N-body test – представляет из себя прикладную задачу, которая может рассматриваться как модельная для широкого класса вычислительных задач, по расчету гравитационного взаимодействия между частицами при помощи технологий CUDA и OpenCl. 4) Gas dynamic test – также является прикладной задачей по решению многомерных уравнений газовой динамики, этот тест полезен с точки зрения оценки эффективности использования данных систем при расчете сеточных задач. Таким образом, такой набор тестов позволяет сформировать достаточно полное представление о производительностях различных архитектур гибридных вычислительных систем. Поскольку для ГВМ скорость расчета для одинарной (float) и двойной (double) точности различаются, то проводилось две серии тестов для ГВМ (с использованием float и double).

Результаты тестирования показали, что самой эффективной архитектурой с точки зрения применения графических вычислительных модулей оказалась 8-ми карточная машина, однако такой подход к построению вычислительных систем не ориентирован на создание больших вычислительных кластеров и тем самым самым сужает количество

программ, которые можно эффективно выполнять на данной системе. Поэтому для применения классических методов распараллеливания (MPI, OpenMP) или смешанных (MPI, OpenMP, CUDA, OpenCl) имеет смысл использовать гибридные вычислительные кластеры с распределенной памятью.

Применение параллельных вычислений в моделировании адсорбции фармацевтически активных веществ пористыми матрицами.

Колноченко А.В.
РХТУ им.Д.И.Менделеева

Внедрение фармацевтически активных веществ в пористые матрицы позволяет создавать новые композиты, уникальные свойства которых востребованы современной фармацевтикой. Целью работы стало исследование адсорбционных свойств аэрогелей по отношению к ряду нестероидных противовосполительных средств. Для моделирования адсорбции использована модель, допускающая высокую степень параллелизма -- клеточные автоматы с окрестностью Марголуса (КАОМ). Модель обобщена и развита на случай взаимодействия между молекулами активных веществ между собой и со стенками пор. Учёт взаимодействий разнородных объектов потребовал серьёзной переработки и оптимизации классической модели КАОМ для применения графических ускорителей в качестве расчётных устройств.

Реализация модели с использованием архитектуры CUDA позволила повысить скорость вычислений более чем в 100 раз, по сравнению с реализацией на традиционных последовательных технологиях.

Исследование проницаемости модельных мембранных систем методом молекулярной динамики.

М.Ю. Антонов, К.В. Шайтан.
Кафедра биоинженерии Биологического факультета МГУ имени М.В.Ломоносова
mikhail@moldyn.org

В работе при помощи управляемой молекулярной динамики был изучен процесс диффузии набора низкомолекулярных соединений, представляющих интерес с биологической точки зрения, через модельные мембранные системы. В качестве мембранных систем рассматривались липидные бислои различного состава. Поскольку самопроизвольная диффузия малых молекул через липидные бислои протекает достаточно медленно, в силу ограничений метода практически невозможно проследить этот процесс в рамках полноатомной равновесной молекулярной динамики, в особенности это касается полярных молекул. Поэтому в работе был применен метод неравновесной управляемой молекулярной динамики, который позволяет за достижимые времена численного эксперимента дать количественную оценку параметрам, характеризующим физические механизмы диффузии в микрогетерогенных структурах.

Для проведения расчета молекулярного моделирования, использовалось программное обеспечение ПУМА-Б, расчеты проводились в силовом поле amber99. В качестве модельных мембран использовались липидные бислои, собранные из трех типов липидов в различных пропорциях. В неравновесном эксперименте к пробным молекулам прикладывалась дополнительная постоянная сила F_{ext}, действующая в направлении нормали к поверхности мембраны. Исходя из полученных данных, производилась оценка

усредненного локального коэффициента трения, который далее пересчитывался в терминах эффективной микровязкости среды или коэффициента диффузии, используя формулу Стокса-Эйнштейна. Следует отметить, что применение в данном случае приближения сплошной среды для сложной гетерогенной молекулярной системы следует понимать как подход, позволяющий количественно оценить характеристику локальных диссипативных свойств липидного бислоя в единицах вязкости. Вычисляемые таким способом кинетические параметры, позволяют изучать проницаемость мембран различного липидного состава для лигандов различной химической природы и проводить сравнительное изучение эффективных вязкостных характеристик модельных мембранных систем и диффузионных свойств малых молекул различной химической природы.

Было показано, что при увеличении содержания ненасыщенных липидов возрастает проницаемость модельной мембраны, что вероятно связано с дефектностью структуры липидного бислоя. Наличие двойных связей в алкильных хвостах способствует уменьшению упорядоченности бислоя и приводит к разрыхлению полукристаллического состояния мембранной структуры

Изучение неоднородности микровязкости по отношению к уровню нормали бислоя показало, что локальная микровязкость значительно различается в разных частях бислоя. При этом область липидных голов оказывается наиболее вязкой и чувствительной к химической природе пробных молекул, выполняя селективную роль. Вязкость в гидрофобной сердцевине бислоя значительно меньше и может отличаться на порядок и более. При этом она более чувствительна к размеру молекулы, и менее чувствительна к ее полярным свойствам.

Моделирование пространственно_временных паттернов активности нейрональных систем с динамической архитектурой.

Д.Ф. Иудин, В.Б. Казанцев, И.В. Мухин, А.Б. Терентьев

ННГУ, ИПФ РАН, НГМА. Текст тезисов отсутствует

Новые перспективы для молекулярного моделирования методом Монте Карло при использовании GPU Nvidia

В.С. Дуняшев, А.К. Лященко, В.Л. Дуняшев
ИОНХ РАН. Текст тезисов отсутствует