

СИСТЕМА ИМИТАЦИОННОГО МОДЕЛИРОВАНИЯ ВОЗДЕЙСТВИЯ ВЫСОКОСКОРОСТНЫХ ЧАСТИЦ НА ГЕРМЕТИЧНЫЕ КОНСТРУКЦИИ КОСМИЧЕСКОГО АППАРАТА С ИСПОЛЬЗОВАНИЕМ ТЕХНОЛОГИИ CUDA

С.А. Крестьянсков, Л.С. Зеленко

Самарский государственный аэрокосмический университет им. С.П. Королева

Введение

Создание высоких технологий и разработка новой техники требуют все более точных знаний о количественных характеристиках нестационарных явлений при интенсивных нагрузках, которые инициированы процессами физического или химического взрыва и высокоскоростного удара. Эти процессы, в отличие от квазистатических, являются кратковременными и быстропротекающими и сопровождаются образованием ударных волн, волн сжатия и разрежения. Взаимодействие и отражение волн от различных дезориентированных поверхностей и движущихся тел и сред приводит к большим деформациям, локальному перегреву вещества, изменению кристаллической структуры и свойств нагружаемых сред, множественным разрушениям и другим необратимым эффектам. Поэтому численные методы в задачах физики взрыва и удара в настоящее время являются важным и весьма *эффективным инструментом моделирования процессов*, протекающих в экстремальных условиях нагружения газов, жидкостей и твердых деформируемых сред [1].

Целью работы является создание системы имитационного моделирования для проведения вычислительных экспериментов, позволяющих исследовать быстропротекающие процессы, такие как взрыв и высокоскоростной удар. Система будет использоваться для тестирования конструкций в условиях интенсивных нагрузок, сопровождающихся образованием ударных волн, разрушениями и переходами из одного фазового состояния в другое.

Актуальность разработки системы обусловлена отсутствием в большинстве проектных организаций и исследовательских лабораторий эффективных высокопараллельных средств для проведения численных расчетов задач механики сплошных сред с использованием параллельных средств обработки данных, реализуемых на персональных компьютерах с графическими процессорами Nvidia.

Численный метод решения – метод гидродинамики гладких частиц

В качестве численного метода для моделирования задачи высокоскоростного соударения был использован бессеточный свободно-лагранжев *метод гладких частиц* (smoothed particle hydrodynamics method – SPH), предложенный в 1977 г. Монаганом и Джингольдом для решения астрофизических проблем [2]. Он обеспечивает достаточную точность расчетов, достигаемую при использовании имитационной модели. Кроме того, данный метод универсален и алгоритмически реализуем с применением параллельных вычислений, так как расчет параметров гладкой частицы может осуществляться независимо от остальных частиц системы.

Особенности реализации системы

В связи с тем, что метод SPH универсален и частицы можно считать независимыми, авторами было принято решение использовать для вычислений параллельную архитектуру графического процессора GPU и технологию CUDA, которая хорошо подходит для обработки больших объемов данных, обеспечивает доступ к набору инструкций графического ускорителя и позволяет управлять его памятью при организации параллельных вычислений.

На каждом итерационном шаге программы перед следующей прорисовкой системы частиц необходимо вычислять новые параметры гладких частиц через старые параметры частиц-соседей.

В процессе разработки системы было выяснено, что использование **технологии The CUDA/OpenGL interoperability** позволяет заметно увеличить производительность системы. Данная технология дает возможность разделять память между CUDA и OpenGL, т.е. вычислять новые параметры системы частиц в CUDA и производить рендеринг из той же памяти с использованием OpenGL. При сравнительном анализе решения одной задачи на GPU (CUDA) и GPU (CUDA/ OpenGL Interoperability) было получено, что в некоторых случаях FPS снижается практически в два раза при применении технологии CUDA/ OpenGL Interoperability.

Поиск соседей

Как любая задача системы множества взаимодействующих тел, метод гладких частиц сталкивается с проблемой поиска соседних частиц, находящихся на расстоянии взаимодействия r_c . В случае решения проблемы полным перебором всех пар мы получим сложность порядка n^2 . Это значит, что при увеличении числа частиц в 10 раз, время, затраченное на поиск взаимодействующих частиц, увеличивается в 100 раз.

Для ускорения поиска используется алгоритм разбиения пространства сеткой (рис. 1).

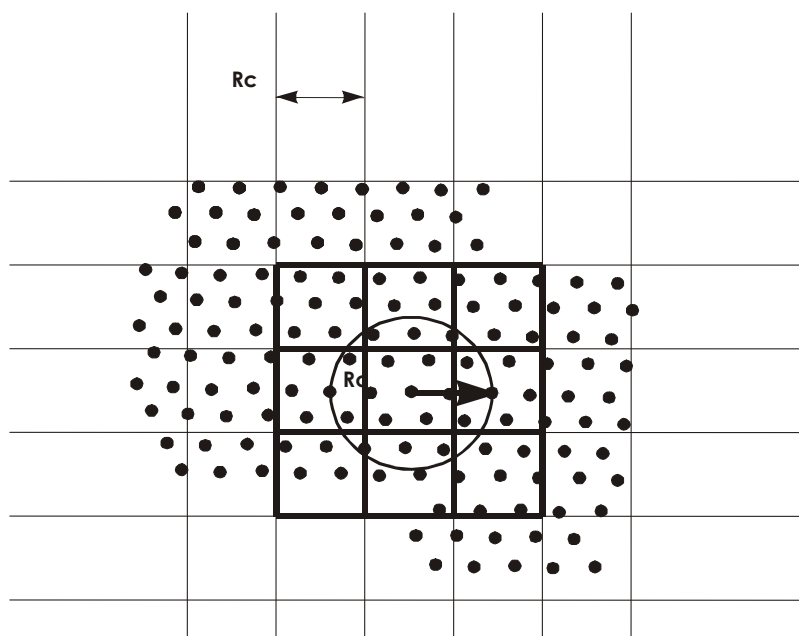


Рис. 1. Пространственная сетка

Алгоритм состоит в следующем:

- 1) все пространство частиц разбивается регулярной сеткой, равной максимальному радиусу взаимодействия, представляющей собой трехмерный массив;

- 2) на каждой итерации алгоритм перебирает все частицы и добавляет их в список частиц ячейки пространства, в которой они геометрически находятся; сложность этапа имеет порядок n ;
- 3) при непосредственном поиске соседей для некоторой частицы просматриваются только те частицы, которые располагаются в текущей ячейке и соседних ячейках. Сложность этапа k_n , где k не зависит от n и равно среднему числу атомов в текущей и соседних ячейках.

Таким образом, алгоритм поиска соседей имеет порядок, равный числу частиц. Поскольку поиск соседних частиц остается наиболее вычислительно сложным этапом из всего алгоритма расчета итерации, то целесообразным является решение распараллелить цикл поиска соседей.

Чтобы успешно осуществить распараллеливание процесса поиска соседей, необходимо предварительно загрузить в память видеокарты начальные данные о системе частиц, т.е. параметры регулярной сетки и распределение частиц по ней (рис. 2). На каждом итерационном шаге перед тем, как вычислять параметры отдельных частиц, необходимо обновлять данные об их соседях. Для этого каждая из вычислительных нитей по новому состоянию частицы, а именно ее позиции в пространстве, определяет ее принадлежность к новой или старой ячейке сетки.

Загрузка данных о системе частиц осуществляется один раз в начале работы программы, далее эти данные модифицируются в памяти видеокарты.

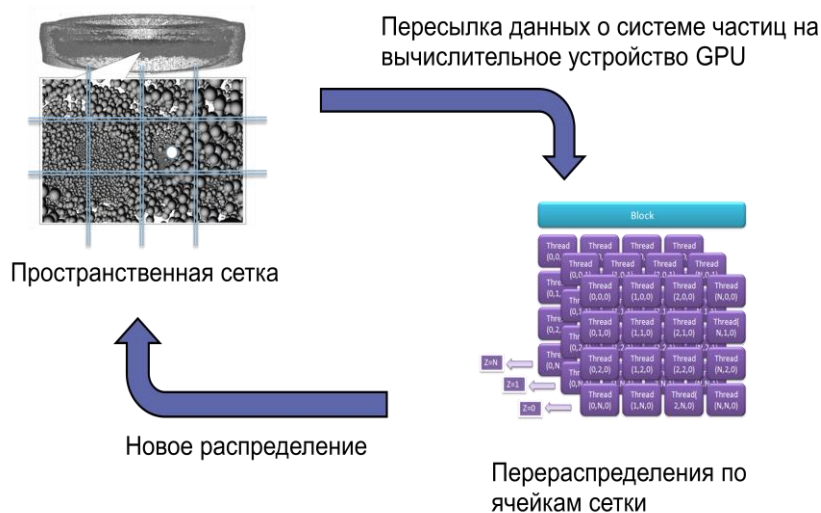


Рис. 2. Принцип реализации параллельного алгоритма (поиск соседей)

Описание работы системы

Для создания начальных параметров эксперимента в трехмерном пространстве используется 3D Studio MAX 7. По открытию открывается стандартное окно программы, где пользователь уже может добавлять геометрические объекты и задавать их материал (см. рис. 3).

Вычислитель представляет собой консольное приложение, принимающее на вход конечный файл с данными о начальном состоянии системы частиц и использующее для визуализации отдельное окно отрисовки OpenGL. На рис. 4 изображен пример визуального эксперимента.

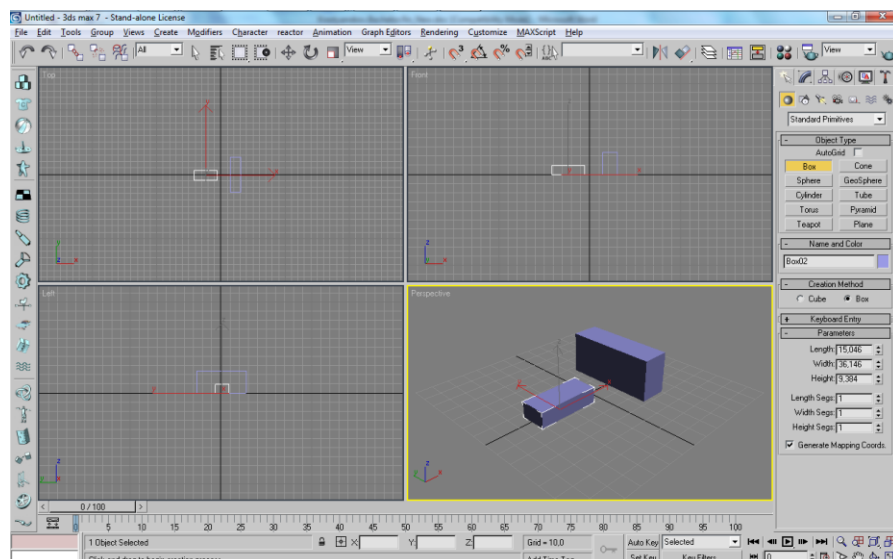


Рис. 3. Создание модели в 3D Studio Max 7

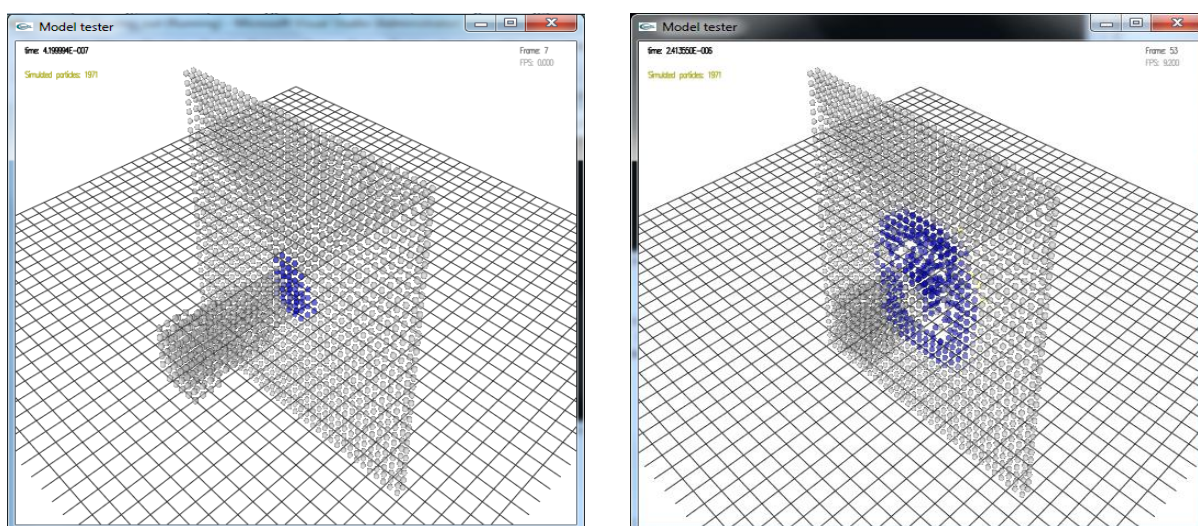
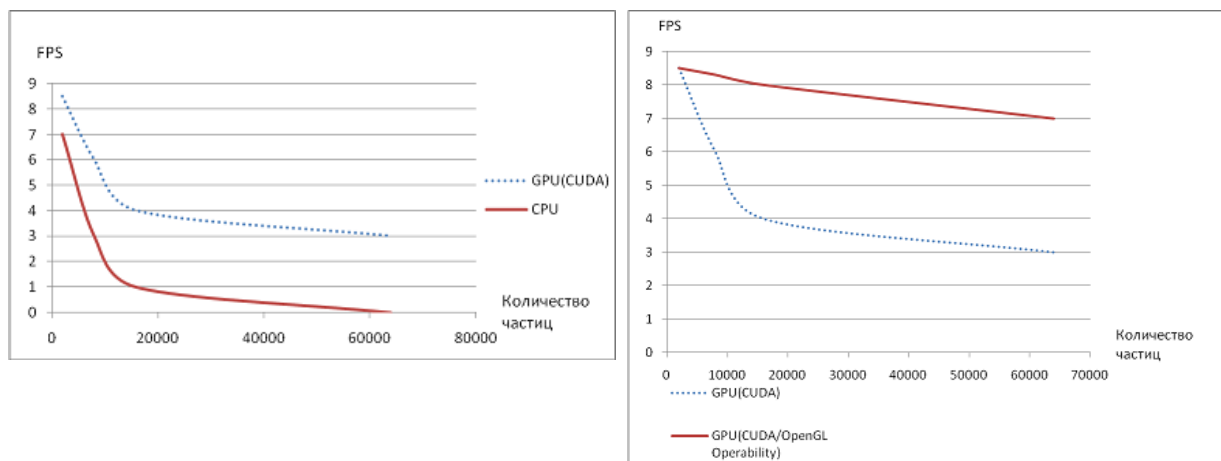


Рис. 4. Визуализация эксперимента

Вычислительные эксперименты

При использовании технологии CUDA было достигнуто значительное ускорение, что дало существенный выигрыш в производительности. На рис. 5а) приведен график производительности вычислительной системы при использовании данного подхода.

Как было отмечено, при реализации численных алгоритмов приходится сталкиваться с проблемой пересылки данных о частицах с GPU на CPU для последующей отрисовки. Технология The CUDA/OpenGL interoperability позволяет избежать этого и дает заметные преимущества. На рис. 5б) приведен график производительности вычислительной системы при использовании данной технологии.



а) при использовании CPU и GPU

б) при использовании CUDA/OpenGL interoperability

Рис. 5. Сравнительный анализ производительности

Перенесение задачи поиска соседей на GPU дает еще более существенный выигрыш в производительности (см. рис. 6).

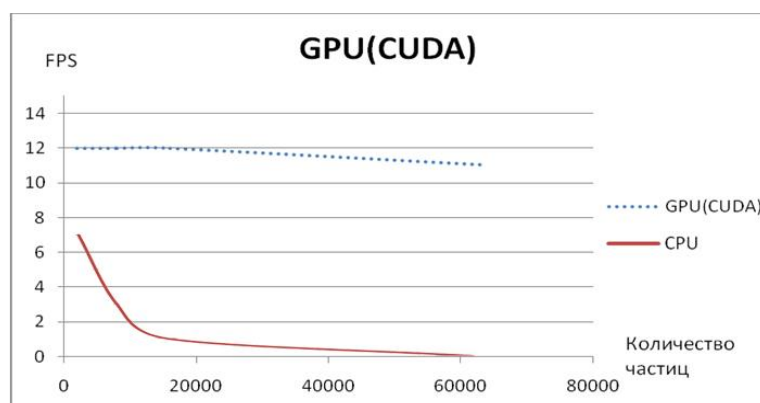


Рис. 6. Сравнительный анализ производительности

Заключение

Разработанная система имитационного моделирования позволяет проводить исследования воздействия высокоскоростных частиц на конструкции космических аппаратов с применением технологии CUDA в качестве параллельной модели обработки данных.

Литература

1. Бабкин А.В. Прикладная механика сплошных сред. Т. 3. Численные методы в задачах физики взрыва и удара / А.В. Бабкин, В.И. Колпаков, В.Н. Охитин, В.В. Селиванов. – М.: Изд-во МГТУ им. Н.Э. Баумана, 2000.
2. Gingold R.A. Smoothed Particle Hydrodynamics. Theory and Application to Non-spherical Stars / R.A. Gingold, J.J. Monahan. – Monthly Notices Royal Astr. Soc., 1977. – 181 p.
3. Parshikov A.N. Improvements in SPH methods by means of interparticle contact algorithm and analysis of perforation tests at moderate projectile velocities / A.N. Parshikov, S.A. Medin, I.I. Loukashenko, V.A.Milekhin // International Journal of Impact Engineering. – 2000. – V. 24, № 15. – P. 779-796.

4. Parshikov, A. N. Smoothed particle hydrodynamics using interparticle contact algorithms / A.N. Parshikov, S.A. Medin // Journal of Computational Physics. – 2002. – V. 12, № 180. – P. 358-382.