Физическое моделирование на OpenCL

|  |  |
| --- | --- |
| Боголепов Д. bogdencmc@inbox.ru | Блохин О. blohin.o.d@gmail.com |
| Захаров М. maxim.zakharov@inbox.ru | Сопин Д. sopindm@gmail.com |
| Удалова Т. udalova.t@gmail.com | Калишев Г. gleb-kalishev@rambler.ru |

# Введение

В конце 2008 года организация Khronos Group утвердила новый стандарт OpenCL. Он предназначен для осуществления вычислений на гетерогенных системах. В апреле 2009 года Nvidia и AMD, выпустили драйвера с поддержкой OpenCL, но до сих пор мало информации о производительности решений на базе этой технологии.

# Постановка задачи

Для тестирования производительности была выбрана задача моделирования из класса N-body problems, задачи такого типа хорошо поддаются распараллеливанию.

Рассматривается эволюция системы N взаимодействующих точечных зарядов в магнитном поле. Действие зарядов друг на друга определяется по закону Кулона [3 p129], а влияние магнитного поля на частицу – силой Лоренца [3 p182].

# Описание решения

Система дифференциальных уравнений, которая определяет изменение ускорения частиц, интегрируется при помощи метода Эйлера. Основную долю вычислений при решении задачи составляет расчёт ускорения каждой частицы.

Мы хотим узнать, насколько теоретическую производительность реально достичь на практике, используя OpenCL.

При подсчёте, мы предполагаем, что будут произведены все возможные оптимизации исходного кода. Таким образом, получим оценку для числа затраченных операций снизу, а для частоты кадров – сверху.

|  |  |  |
| --- | --- | --- |
| Процедура расчета ускорения для одной частицы | flops | GPU cycles |
| 1|for ( int i = 0; i < N; ++i ) |  |  |
| 2|{ |  |  |
| 3| // Calculate acceleration caused by particle j  on particle i |  |  |
| 4| float3 r = p - pi; | 3 | 3 |
| 5| float dist = dot ( r, r ); | 5 | 5 |
| 6| float invDist = inversesqrt ( dist + SOFTENING ); | 2 | 5 |
| 7| float invDistCube = invDist \* invDist \* invDist; | 2 | 2 |
| 8| float s = q \* qi \* invDistCube; | 2 | 2 |
| 9| // accumulate effect of all particles |  |  |
| 10| acc += s \* r; | 6 | 3 |
| 11|} |  |  |
| Итого: | 20 | 20 |

В системе команд GPU есть инструкция одновременного умножения и сложения (mad – multiple add), которая .

Мы получили, что на расчет движения одной частицы приходится 20∙N flops, а на одну итерацию по методу Эйлера – 20∙N2 flops. Число операций растёт пропорционально квадрату количества частиц.

# Результаты экспериментов

Для теоретической оценки производительности мы пользовались формулой:

Таблица . Характеристики оборудования

|  |  |  |
| --- | --- | --- |
| Устройство | Число потоковых процессоров | Частота одного процессора, MHz |
| ATI Radeon HD 4890 | 800 | 850 |
| Nvidia Quadro FX5600 |  |  |

Таблица 2. Производительность в теории

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Число частиц | ATI Radeon HD 4890 | | Nvidia Quadro FX5600 | |
| FPS | GFLOPS | FPS | GFLOPS |
| 2048 |  |  |  |  |
| 4096 |  |  |  |  |
| 8192 |  |  |  |  |
| 16384 |  |  |  |  |
| 2048 |  |  |  |  |

Таблица 3. Производительность на практике

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| ATI Radeon HD 4890 | | | | |
| 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 |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Nvidia Quadro FX5600 | | | | | | |
| 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

Несмотря на то, что стандарт продолжает развиваться и не везде поддерживается в полной мере, OpenCL предоставляет приятные перспективы развития отрасли. На тестах новая технология показывает близкие и, иногда, даже более хорошие результаты, по сравнению с существующими решениями.

# Литература

1. *Курс Физики. Трофимова Т. И. 1990г.*
2. *Erich Elsen V., Vishal Mike Houston и др. N-Body Simulations on GPUs*.  
   <http://arxiv.org/pdf/0706.3060>