

УДК 004.93

С.А. Зори, канд. техн. наук, доц.,

В.В. Лисеенко, магистр

ГВУЗ «Донецкий национальный технический университет», г. Донецк, Украина

zori@pmi.dgtu.donetsk.ua

Синтез изображений рельефов и ландшафтов методом ROAM на кластерной и CUDA- архитектурах вычислительных систем

В работе рассмотрена задача синтеза реалистичных изображений рельефов и ландшафтов и эффективность ее решения в реальном времени методом ROAM на популярных современных архитектурах параллельных вычислительных систем. Выполнено исследование эффективности реализации алгоритма ROAM на параллельном GPU-мультипроцессоре с использованием технологии CUDA и кластерной архитектуре NeClus с использованием технологии MPI. Показано, что использование GPU-мультипроцессора для реализации ROAM является неэффективным, а использование кластерной архитектуры – эффективным и позволяющим удовлетворить требования реального времени решением.

Ключевые слова: рельеф, ландшафт, реалистичная визуализация, синтез изображений, ROAM, графический мультипроцессор, CUDA, кластер, NeClus, эффективность.

Введение

Технологии визуализации рельефов и ландшафтов в реальном времени сегодня применяют в самых разнообразных сферах деятельности человека как часть прикладных задач, решаемых системами виртуальной и дополненной реальности, географическими информационными системами, предоставляющими возможности для сбора, хранения, анализа и графической визуализации географических данных, системами проектирования дизайнерских решений и многими другими.

Современный объем обрабатываемых данных, напрямую связанный с пространственным разрешением реальных и модельных гео- данных, сегодня очень велик, что приводит к непосредственному росту сложности модели сцены, и, как следствие, к уменьшению производительности алгоритмов синтеза.

В связи с этим, необходимо отметить, что используемые в этой области традиционные подходы к решению задачи рассчитаны на традиционные архитектуры компьютерных систем, и не в состоянии решать ее в реальном времени на актуальных сегодня объемах и сложностях данных и моделей.

Ввиду уже отмеченной высокой вычислительной сложности решения этой задачи и ее высокой зависимости от пространственного разрешения сеток данных, современные системы синтеза изображений и визуализации рельефов и ландшафтов должны максимально полно использовать производительность вычислительного оборудования.

Целью работы является определение эффективности синтеза изображений рельефов и ландшафтов методом ROAM в реальном времени с использованием популярных параллельных архитектур вычислительных систем.

Основные алгоритмы синтеза изображений рельефов и ландшафтов

Классификация основных методов и алгоритмов синтеза изображений рельефов и ландшафтов в системах визуализации приведена на рис. 1.



Рисунок 1 – Классификация методов синтеза изображений рельефов и ландшафтов

Наиболее обобщенными LOD- алгоритмами построения аппроксимаций являются иррегулярные сети треугольников, однако особенности вычислений не позволяют эффективно использовать вычислительную мощность современной аппаратуры, вследствие чего данный алгоритмы и его вариации не могут быть использованы на целевых объемах данных.

Техники с ограниченной адаптивностью могут применяться для решения конкретных задач визуализации, но с увеличением размеров обрабатываемых карт высот линейно растет их вычислительная сложность, что не только не позволяет решить поставленную задачу в полном объеме, но и существенно лимитирует масштабируемость в дальнейших применениях.

Анализ литературных источников позволяет утверждать, что одними из главных (по эффективности) используемых современных методов синтеза изображений рельефов и ландшафтов являются:

- методы, использующие иррегулярные сети треугольников (Shroeder, Rossignac, Borrel и др. [1, 2]);
- методы с ограниченной адаптивностью (Willem H. de Boer, Losasso и Норре [3, 4]);
- методы иерархической триангуляции на основе квадродеревьев и бинарных деревьев треугольников (Duchaineau и Ulrich, [5, 6]).

При этом, наибольшую эффективность при визуализации статического и динамического рельефа, показали методы, использующие иерархическую триангуляцию на основе квадродеревьев и бинарных деревьев треугольников, а также новые адаптивные кластерные подходы к ним, самым известным из которых является ROAM [5].

Алгоритм ROAM фактически является модификацией классического алгоритма бинарных деревьев треугольников и основан на бинарном дереве, которое является специальным случаем итеративной бисекции основания треугольника по базовой вершине. В процессе детализации пара треугольников разбивается по общей базовой вершине, находящейся на смежных основаниях (рис.2).

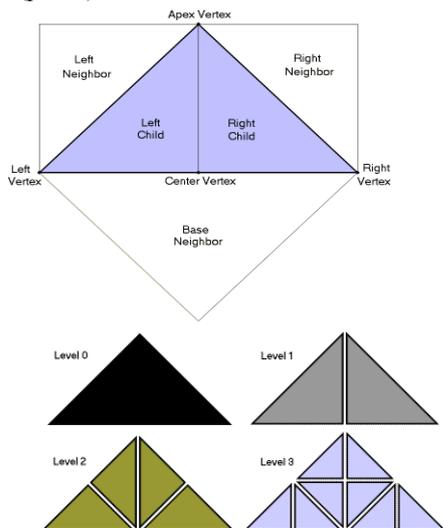


Рисунок 2 – Иллюстрация метода ROAM

Синтез изображений рельефов и ландшафтов методом ROAM на базовой вычислительной системе

Авторами была проведена программная реализация и исследования характеристик метода ROAM (время загрузки карты, производительность от количества выводимых треугольников) на ПК с четырех ядерным процессором (рис. 3, 4).

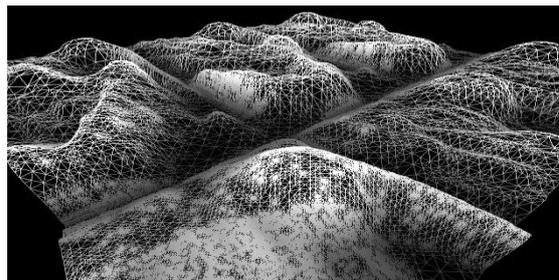


Рисунок 3 – Результат синтеза изображения рельефа в режиме сетки

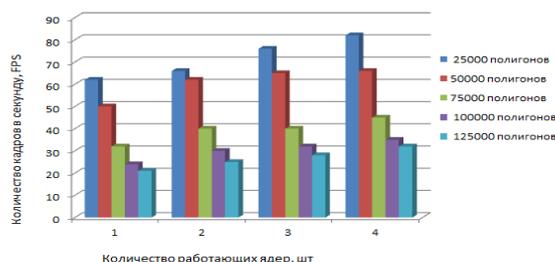


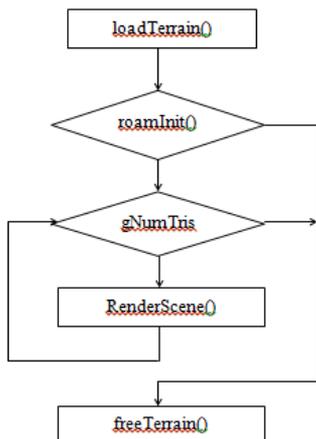
Рисунок 4 – Зависимость FPS от количества ядер и сценарных полигонов

Исследования показали, что с ростом размера карты (что необходимо для улучшения визуального восприятия рельефа - реалистичности), а, следовательно, увеличением количества сценарных полигонов, существенно увеличивается время рендеринга сцены, и использование реальных карт рельефов делает синтез в реальном времени невозможным. С другой стороны, увеличение количества работающих ядер заметно ускоряет производительность алгоритма, что свидетельствует о возможности его распараллеливания (подключение каждого ядра увеличивало производительность на 16-17%).

Выполненный анализ показал, что распараллеливание алгоритма возможно за счет разбиения загружаемой карты высот на блоки и дальнейшей параллельной обработки блоков на параллельных вычислительных архитектурах. Работы велись в направлениях исследований возможности распараллеливания алгоритма ROAM и перенесения вычислений на блоки графических мультимикропроцессоров видеоадаптеров с технологией GPGPU и кластерных архитектур вычислительных систем [7].

Синтез изображений рельефов и ландшафтов методом ROAM на архитектуре GPGPU (CUDA)

Была разработана программная реализация и проведены исследования временных характеристик частей ROAM, реализованных с помощью технологии CUDA, таких как время выполнения соответствующих частей, время загрузки и выгрузки данных (рис. 5).



```

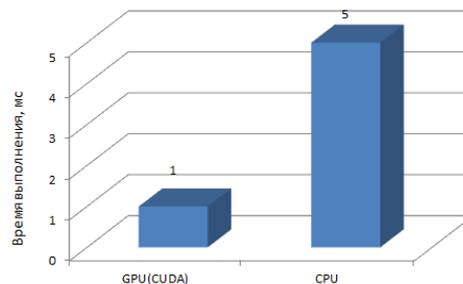
__global__ void TextInit(unsigned char
*d_data_r, unsigned char* d_data_g, unsigned
char *d_data_b,float *rand_arr)
{
int idx=threadIdx.x+blockIdx.x*blockDim.x;
d_data_r[idx]=128*sinf(rand_arr[idx]);
d_data_g[idx]=(int)(128.0+(40.0 *
rand_arr[idx])/RAND_MAX);
d_data_b[idx]=0;
}
  
```

Рисунок 5 – Программное моделирование ROAM на CUDA

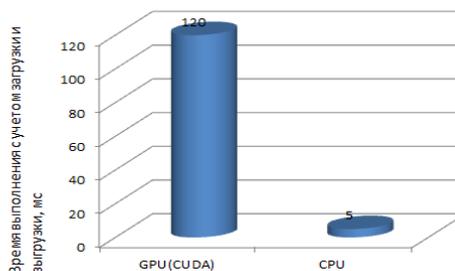
Анализ результатов показал, что ввиду отсутствия поддержки технологией CUDA рекурсивных вычислений и динамических древовидных структур в памяти GPU, использование CUDA является неэффективным, а в некоторых случаях и негативным для общей производительности алгоритма.

Например, время выполнения реализованных функций на видеокарте в 4-5 раз меньше времени выполнения на процессоре базовой системы, но загрузка данных в память видеокарты повышает общее время выполнения в 24 раза! (частая загрузка-выгрузка данных из оперативной памяти процессора в память видеокарты, на что уходит порядка 90-98% общего времени) (рис. 6).

Таким образом, эффективная реализация ROAM с использованием параллельной архитектуры графического мультипроцессора видеокарты ПК является невозможной.



Время выполнения вычислительной части функций: - GPU в 4-5 раз быстрее



Общее время выполнения функций с учетом времени загрузки/выгрузки данных: - CPU в 24 раза быстрее

Рисунок 6 – Реализация ROAM на CUDA

Синтез изображений рельефов методом ROAM на параллельной кластерной вычислительной системе NeClus

Была создана программная параллельная реализация основных частей метода ROAM с использованием технологии MPI на кластере NeClus ДонНТУ (рис. 7).

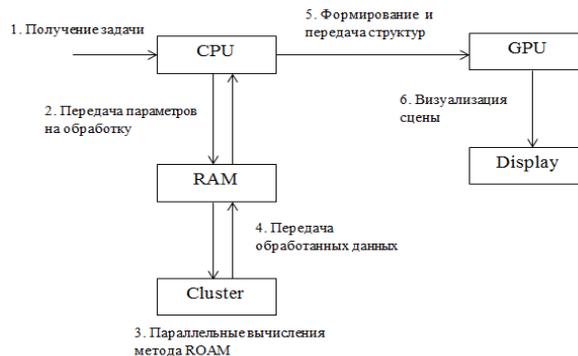


Рисунок 7 – Структура системы визуализации на кластере

Алгоритм синтеза ROAM переносится на параллельную архитектуру вычислительного кластера NeClus путем (рис. 8):

- а) разбиения карты высот на части (патчи);
- б) параллельной обработки патчей на вычислительных узлах кластера:

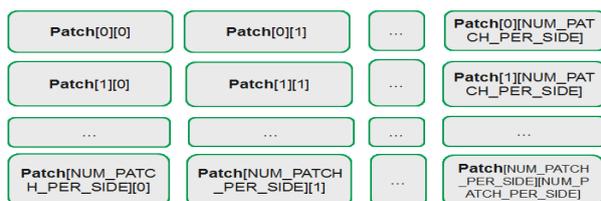


Рисунок 8 – Карта высот, разбитая на патчи

Реализованные основные функции выполняют инициализацию патчей (Init), построение точной аппроксимированной модели (Tessellate) и ее рендеринг (Render) (рис. 9). Используемая для исследований карта высот имеет размер 1024x1024, при размере патча 32x32, для распараллеливания требуется 2^k узлов вычислительного кластера. Так как кластер ограничивает количество используемых узлов для одной задачи, то каждый узел будет обрабатывать $32x32/k$ патчей, где k – количество используемых узлов.

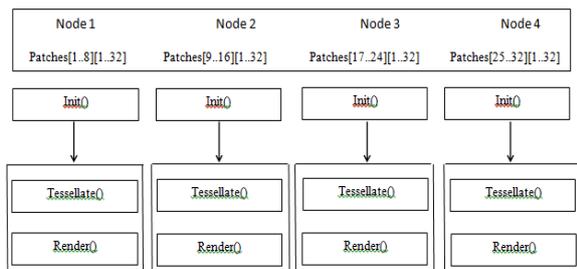


Рисунок 9 – Реализация ROAM на кластере NeClus (4 узла)

Исследования показали, что использование MPI для организации параллельных вычислений на вычислительном кластере является эффективным. Так, использование 4-х узлов кластера позволяет увеличить производительность алгоритма в 2 раза при рендеринге 125000 треугольников (время выполнения алгоритма - 21мс, время реализации на базовой архитектуре – 40 мс, время реализации с использованием технологии CUDA - 750мс).

Увеличение количества используемых узлов кластера приводит к увеличению производительности алгоритма и уменьшению времени выполнения основных его частей (рис. 10) - время синтеза на уменьшается на 85-90% с каждым двукратным увеличением числа узлов, и возможен синтез в реальном времени.

Выводы

В работе исследованы современные методы синтеза реалистичных изображений рельефов и ландшафтов. Показаны главные проблемные места этих алгоритмов – высокая вычислитель-

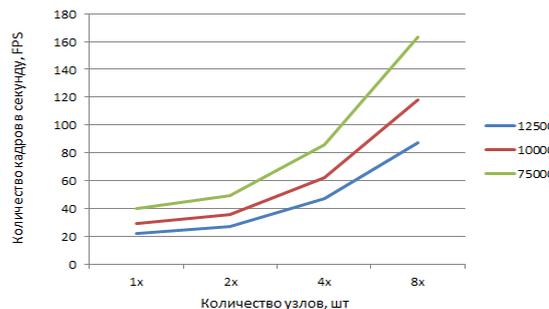
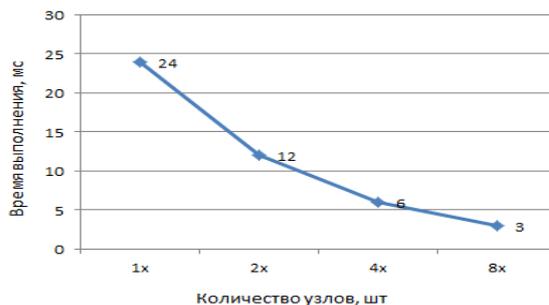


Рисунок 10 – Характеристики реализации ROAM на кластере NeClus

ная сложность, напрямую зависящая от пространственного разрешения данных, и интенсивное использование рекурсивных вычислений для организации уровней детализации.

Показано, что одним из эффективных методов решения поставленной задачи является метод ROAM. Проведено исследование эффективности реализации метода на базовой системе (ПК) и параллельных архитектурах вычислительных систем – параллельного GPU-мультипроцессора в составе современных видеокарт ПК (технология CUDA) и кластерной архитектуре NeClus (технология MPI), получены показатели процесса синтеза.

Показано, что ввиду отсутствия поддержки технологией CUDA рекурсивных вычислений и динамических древовидных структур в памяти GPU, использование CUDA является неэффективным (сами вычисления выполняются в 4-5 раз быстрее, чем вычисления на базовой системе, однако на частую загрузку-выгрузку данных из оперативной памяти процессора в память видеокарты уходит порядка 90-98% общего времени).

Показано, что использование MPI для организации параллельных вычислений на вычислительном кластере является эффективным, увеличение количества используемых узлов кластера приводит к увеличению производительности алгоритма и уменьшению времени выполнения основных его частей (время синтеза на уменьшается на 85-90% с каждым двукратным увеличением числа узлов), и возможен синтез в реальном времени.

Список літератури

1. Decimation of triangle meshes / W.J. Schroeder, J.A. Zarge, and W.E. Lorenzen // SIGGRAPH '92 Proc.: Computer Graphics. - July 1992. - 26(2). - PP. 65–70.
2. Jarek Rossignac. Multi-resolution 3D approximations for rendering complex scenes / Jarek Rossignac, Paul Borrel // In B. Falcidieno and T. Kunii, editors, Modeling in Computer Graphics: Methods and Applications. – 1993. - PP. 455–465.
3. Willem H. de Boer. Fast Terrain Rendering Using Geometrical MipMapping / Willem H. de Boer // E-mersion Project. – October 2000.
4. Hugues Hoppe. View-dependent refinement of progressive meshes / Hugues Hoppe // In SIGGRAPH '97 Proceedings. - 1997.
5. ROAMing terrain: Realtime optimally adapting meshes / [M. Duchaineau, M. Wolinsky, D.E. Sigtet, M.C. Miller, C. Aldrich, M.B. Mineev-Weinstein] // In Proceedings IEEE Visualization. – 1997. - PP. 81–88.
6. Ulrich T. Rendering massive terrains using chunked level of detail / T. Ulrich // In: Super-size-it! Scaling up to Massive Virtual Worlds (ACM SIGGRAPH Tutorial Notes). ACM SIGGRAPH, 2000.
7. Зори С.А. Методи синтезу реалістичних зображень рельєфів і ландшафтів для паралельних висчислювальних систем тривимірної комп'ютерної графіки / С.А. Зори, В.В. Лисеєнко // Моделювання та комп'ютерна графіка: матеріали 4-ї міжнародної науково-технічної конференції (м. Донецьк, 5-8 жовтня 2011 р.) / Міністерство освіти та науки, молоді та спорту України. — Донецьк: ДонНТУ, 2011. – С.114-118.

Надійшла до редакції 26.04.2013

С.А. ЗОРИ, В.В. ЛИСЕЄНКО

ДВНЗ «Донецький національний технічний університет»

СИНТЕЗ ЗОБРАЖЕНЬ РЕЛЬЄФІВ І ЛАНДШАФТІВ МЕТОДОМ ROAM НА КЛАСТЕРНІЙ І CUDA-АРХІТЕКТУРАХ ОБЧИСЛЮВАЛЬНИХ СИСТЕМ

В даній роботі розглянута задача синтезу реалістичних зображень рельєфів і ландшафтів та ефективність її вирішення в реальному часі методом ROAM для популярних сучасних архітектур паралельних обчислювальних систем. Виконано дослідження ефективності реалізації алгоритму ROAM на паралельному GPU-мультипроцесорі з використанням технології CUDA і кластерної архітектури NeClus з використанням технології MPI. Показано, що використання GPU-мультипроцесору для реалізації ROAM є неефективним, а використання кластерної архітектури - ефективним рішенням, що дозволяє задовольнити вимоги реального часу.

Ключові слова: *рельєф, ландшафт, реалістична візуалізація, синтез зображень, ROAM, графічний мультипроцесор, CUDA, кластер, NeClus, ефективність.*

S.A. ZORI, V.V. LISEIENKO

Donetsk National Technical University

IMAGE-SYNTHESIS OF RELIEFS AND LANDSCAPES BY ROAM- METHOD ON CLUSTER AND CUDA- COMPUTING ARCHITECTURES

The paper considers the problem of relief and landscapes image-synthesis and the efficiency of real time relief image-synthesis by algorithm ROAM for popular modern parallel computing systems. The analysis and classification of the basic image-synthesis methods for reliefs and landscapes were made. We studied the efficiency of realization of ROAM-algorithm on parallel GPU with the use of CUDA-technology and cluster architecture NeClus with the use of MPI-technology and obtained some parameters of the synthesis process. The conclusion is that the use of GPU-multiprocessor for ROAM-realization is inefficient, while the use of cluster architecture is an effective solution, which meets the requirements of real time.

Key words: *relief, landscape, realistic visualization, image-synthesis, ROAM, GPU-multiprocessor, CUDA, cluster, NeClus, efficiency.*