УДК 004.7

М. Г. Сірик, студент факультету ІКТ, V курс, гр. ПІ-41
Науковий керівник – Морозов А.В., к.т.н., декан факультету інформаційно-комп'ютерних технологій
Житомирський державний технологічний університет

НЕГРАФІЧНІ ОБЧИСЛЕННЯ НА ГРАФІЧНИХ ПРОЦЕСОРАХ ЗА ДОПОМОГОЮ ТЕХНОЛОГІЇ NVIDIA CUDA

Охарактеризуємо основні відмінності між архітектурами CPU (Graphical Processor Units) і GPU (Central Processor Units), які впливають на особливості математичних обчислень. Ядра CPU створені для виконання одного потоку послідовних інструкцій з максимальною продуктивністю, а GPU проектуються для швидкого виконання великої кількості паралельно виконуваних потоків інструкцій. GPU відрізняється від CPU ще й за принципами доступу до пам'яті. У GPU він пов'язаний і легко передбачуваний. Якщо з пам'яті зчитується тексель текстури, то через деякий час прийде час і для сусідніх текселей. Та й при записі те ж – піксель записується під фреймбуфер, і через кілька тактів буде записуватися розташований поруч з ним. Тому організація пам'яті відрізняється від тієї, що використовується в CPU. І відеочіпу, на відміну від універсальних процесорів, просто не потрібна кеш-пам'ять великого розміру, а для текстур потрібні лише кілька (до 128-256 в нинішніх GPU) кілобайт.

Робота з пам'яттю у GPU і CPU дещо відрізняється. Не всі центральні процесори мають вбудовані контролери пам'яті, а у всіх GPU зазвичай є по кілька контролерів, аж до восьми 64-бітних каналів в чіпі NVIDIA GT200. Крім того, на відкритих застосовується більш швидка пам'ять, і в результаті відеочіпам доступна в рази більша пропускна здатність пам'яті, що також досить важливо для паралельних розрахунків, що оперують з величезними потоками даних.

Обчислення на GPU розвивалися і розвиваються дуже швидко. І надалі, два основних виробника відеочіпів, NVIDIA і AMD, розробили і анонсували відповідні платформи під назвою CUDA (Compute Unified Device Architecture) і CTM (Close To Metal або AMD Stream Computing), відповідно. На відміну від попередніх моделей програмування GPU, ці були виконані з урахуванням прямого доступу до апаратних можливостей відеокарт. Платформи не сумісні між собою, CUDA – це розширення мови програмування C, а CTM – віртуальна машина, виконуюча асемблерний код. Зате обидві платформи ліквідували деякі з важливих обмежень попередніх моделей GPGPU, що використовують традиційний графічний конвеєр і відповідні інтерфейси Direct3D або OpenGL.

Щоб зрозуміти, які переваги приносить перенесення розрахунків на відеочипи, наведемо усереднені цифри, отримані дослідниками по всьому світу. В середньому, при перенесенні обчислень на GPU, у багатьох задачах досягається прискорення в 5–30 разів, у порівнянні з швидкими універсальними процесорами. Найбільші цифри (близько 100-кратного прискорення і навіть більше) Досягаються на коді, який не дуже добре підходить для розрахунків за допомогою блоків SSE, але цілком зручний для GPU.

Найбільш відомий BrookGPU, компілятор потокового мови програмування Brook, створений для виконання неграфічних обчислень на GPU. До його появи розробники, що використовують можливості відеочіпів для обчислень, вибирали один з двох поширених API: Direct3D або OpenGL. Це серйозно обмежувало застосування GPU, адже в 3D-графіці використовуються шейдери і текстури, про які фахівці по паралельному програмуванню знати не зобов'язані, вони використовують потоки і ядра. Brook зміг допомогти в полегшенні їх завдання. Ці потокові розширення до мови C, розроблені в Стенфордському університеті, приховували від програмістів тривимірний API, і представляли відеочіп у вигляді паралельного співпроцесора. Компілятор обробляв файл .br з кодом C ++ і розширеннями, виробляючи код, прив'язаний до бібліотеки з підтримкою DirectX, OpenGL або x86.

Надалі, деякі дослідники з проекту Brook влилися в команду розробників NVIDIA, щоб представити програмно-апаратну стратегію паралельних обчислень, відкривши нову частку ринку. І головною перевагою цієї ініціативи NVIDIA стало те, що розробники відмінно знають всі можливості своїх GPU до дрібниць, і у використанні графічного API немає необхідності, а працювати з апаратним забезпеченням можна безпосередньо за допомогою драйвера. Результатом зусиль цієї команди стала NVIDIA CUDA - нова програмно-апаратна архітектура для паралельних обчислень на NVIDIA GPU.

Технологія CUDA – це програмно-апаратна обчислювальна архітектура NVIDIA, заснована на розширенні мови C, яка дає можливість організації доступу до набору інструкцій графічного прискорювача і управління його пам'яттю при організації паралельних обчислень. CUDA допомагає реалізовувати алгоритми, здійснимі на графічних процесорах відеоприскорювачів GeForce восьмого покоління і старше (серії GeForce 8, GeForce 9, GeForce 200), а також Quadro і Tesla.

Перелічимо основні характеристики CUDA:

- уніфіковане програмно-апаратне рішення для паралельних обчислень на відеочіпах NVIDIA;
- великий набір підтримуваних рішень, від мобільних до мультичіпових;

- стандартна мова програмування C;
- стандартні бібліотеки чисельного аналізу FFT (швидке перетворення Фур'є) і BLAS (лінійна алгебра);
- оптимізований обмін даними між CPU і GPU;
- взаємодія з графічними API OpenGL і DirectX;
- підтримка 32- і 64-бітових операційних систем: Windows XP, Windows Vista, Linux і MacOS X;
- можливість розробки на низькому рівні.

Програмно-апаратна архітектура для обчислень на GPU компанії NVIDIA відрізняється від попередніх моделей GPGPU тим, що дозволяє писати програми для GPU на справжньому мові C із стандартним синтаксисом, покажчиками і необхідністю в мінімумі розширень для доступу до обчислювальних ресурсів відеочіпів. CUDA не залежить від графічних API, і володіє деякими особливостями, призначеними спеціально для обчислень загального призначення.

Переваги CUDA перед традиційним підходом до GPGPU обчислень:

- інтерфейс програмування додатків CUDA ґрунтується на стандартній мові програмування C з розширеннями, що спрощують процес вивчення і впровадження архітектури CUDA;
- CUDA забезпечує доступ до поділюваного між потоками пам'яті розміром в 16 Кб на мультипроцесор, яка може бути використана для організації кеша з широкою смугою пропускання, в порівнянні з текстурними вибірками;
- більш ефективна передача даних між системною і відеопам'яттю;
- відсутність необхідності в графічних API з надмірністю і накладними витратами;
- апаратна підтримка цілочисельних і бітових операцій.

Основні обмеження CUDA:

- відсутність підтримки рекурсії для виконуваних функцій;
- мінімальна ширина блоку в 32 потоки;
- закрита архітектура CUDA, що належить NVIDIA.

Цілком імовірно, що в силу широкого розповсюдження відеокарт у світі, розвиток паралельних обчислень на GPU сильно вплине на індустрію високопродуктивних обчислень. Ці можливості вже викликали великий інтерес у наукових колах, та й не тільки в них. Адже потенційні можливості прискорення добре піддаються розпаралелюванню алгоритмів (на доступному апаратному забезпеченні, що не менш важливо) відразу в десятки разів бувають не так часто.

ВІДОМОСТІ ПРО АВТОРІВ:

МОРОЗОВ Андрій Васильович, кандидат технічних наук, декан факультету інформаційно-комп'ютерних технологій, доцент кафедри комп'ютерної інженерії Житомирського державного технологічного університету. Наукові інтереси: задачі маршрутизації на графах та мережах, паралельні та розподілені системи, сучасні Інтернет-технології. E-mail: pzs.ztu@gmail.com

СІРИК Микола Григорович, студент групи ПІ-41 кафедри програмного забезпечення систем Житомирського державного технологічного університету. Наукові інтереси: сучасні Інтернет-технології.