КВАНТОВОХИМИЧЕСКИЕ ВЫЧИСЛЕНИЯ С ИСПОЛЬЗОВАНИЕМ ПРОГРАММНО-АППАРАТНОЙ АРХИТЕКТУРЫ CUDA

И.И. Устюгов, Л.В. Парфенова, Л.М. Халилов

Институт нефтехимии и катализа РАН, Уфа

Сочетание теории функционала плотности (DFT) в задачах квантовой химии с возможностью применения мощных графических ускорителей NVIDIA и программно-аппаратной архитектуры CUDA может обеспечить беспрецедентную производительность и точность в расчетах сложных молекулярных систем.

Введение

Современные квантовохимические задачи предъявляют все большие требования к вычислительным возможностям центральных процессоров (Central Processing Unit, CPU). С развитием индустрии видеоигр произошел большой скачок в технологиях производства мощных графических ускорителей NVIDIA. В связи с этим, стало возможным их применение в высокопроизводительных вычислениях (High Performance Computing, HPC).

Графические ускорители, такие как nVidia GeForce 8800GTX, могут быть охарактеризованы как потоковые процессоры [1]. Потоковая обработка – это обобщение архитектуры SIMD (Single Instruction Multiple Data, «одна инструкция – много данных») [2], которая впервые была реализована в суперкомпьютере Cray-1 [3]. Принцип SIMD-архитектуры заключается в том, что приложения объединяются в потоки (streams) и ядра (kernels), представляя собой блоки данных и преобразования кода соответственно. Потоки данных обрабатываются параллельно на нескольких процессорах, используя небольшое количество ядер (порядка 1-2).

В данном примере nVidia 8800GTX имеет 128 потоковых процессоров, объединенных в 16 мультипроцессорных единиц, каждый из которых включает в себе 8 устройств обработки данных. Тактовая частота 1 потокового процессора 1.35 ГГц. Таким образом, суммарная мощность потоковых процессоров nVidia 8800GTX составляет 172.8 ГГц, что сравнимо с традиционными научными расчетными кластерами, в которых используются CPU, например, серверные процессоры AMD Opteron.

На сегодняшний день стоимость графической карты NVIDIA значительно ниже стоимости одного CPU, который используется в научных расчетных кластерах. Это обуславливает экономическую выгоду использования графических ускорителей в квантовохимических расчетах.

Результаты и обсуждение

Ранние попытки использования графических процессоров для «неграфических» вычислений [4-6] были ограничены точностью и сложностью их программирования. Эти проблемы были решены в более современных графических ускорителях, так как они стали поддерживать арифметические операции с 32-битными числами с плавающей точкой (10^{-38}... $\approx 10^{38}$, числа с одинарной точностью). Следующие поколения графических ускорителей и потоковых процессоров NVIDIA и AMD расширили эту под-

держку до арифметики 64-битных чисел с плавающей точкой (10^{-308}... $\approx 10^{308}$, числа с двойной точностью).

Проблему сложности программирования графических процессоров (GPU, Graphical Processing Unit) решила корпорация NVIDIA, представив программно-аппаратную архитектуру CUDA (Compute Unified Device Architecture). Эта архитектура предоставила разработчикам простой программируемый интерфейс для написания исходного кода на языке «Си» (C). С развитием CUDA многие другие языки программирования (такие как Fortran, C#, C++, и т.д.) получили поддержку данной архитектуры.

Результаты расчетов на модельных соединениях (кофеин, холестерол, фуллерен C_{60}, таксол, ваниломицин, CLN025 – «искусственный протеин» [7], олестра) наглядно представлены в работах [3, 8]. В качестве сравнения приводились результаты, полученные в программе *GAMESS v11 Apr 2008 (R1)* на одноядерном CPU Intel Pentium D 3 ГГц.

В расчетах с использованием итеративного алгоритма прямого метода самосогласованного поля (прямой SCF-метод, базис 6-31G) на видеокарте nVidia GeForce 280GTX, nVidia G80 и трех видеокартах nVidia GeForce 280GTX получены впечатляющие результаты [8] (табл. 1).

Таблица 1. Результаты квантовохимического бенчмарка формирования матрицы Фока

		Коэффициент ускорения GPU по сравнению с CPU		
Модельное соединение	**Процессорное время GAMESS, с**	**nVidia G80**	**nVidia 280GTX**	**3 x nVidia 280GTX**
кофеин ($C_8N_4H_{10}O_2$)	7.6	19	28	42
холестерол ($C_{27}H_{46}O$)	113.9	40	56	120
фуллерен (C_{60})	589.9	37	57	155
таксол ($C_{45}NH_{49}O_{15}$)	476.8	46	64	145
ваниломицин ($C_{54}N_6H_{90}O_{18}$)	1226.3	67	90	222
CLN025, «искусственный протеин» ($C_{62}N_{11}H_{97}O_{32}$)	2274.7		92	225
олестра ($C_{156}H_{278}O_{19}$)	14079.2	199	352	696

Результаты приведены для первой, наиболее затратной по времени итерации алгоритма прямого SCF-метода. На GPU использована арифметика 32-битных чисел с одинарной точностью.

Из табл. 1 видно, что при использовании GPU в квантовохимических расчетах дает более чем 100-кратный прирост производительности.

В настоящее время в квантовой химии широкое распространение получила теория функционала плотности (Density Functional Theory, DFT). На данный момент DFT применяется для расчетов энергий связи в молекулах, их минимальной энергии, а так же в физике твердого тела.

Сверхпроводимость, релятивистские эффекты в тяжелых элементах и атомных ядрах, классические жидкости, магнитные свойства сплавов – все это было изучено с помощью теории функционала плотности [9].

Популярность данного подхода обуславливается меньшим временем расчетов (по сравнению с другими квантовохимическими методиками: SCF, MP2, MP3, и т.д.) с незначительным увеличением погрешности вычислений. Следовательно, DFT в сочетании с программно-аппаратной архитектурой CUDA обеспечит беспрецедентную производительность и точность.

Таким образом, на данный момент CUDA представляет собой простую, удобную, быструю и экономически выгодную архитектуру программирования для квантовохимических DFT-вычислений сложных молекулярных систем.

Литература

1. Kapasi U.J., Rixner S., Dally W.J., Khailany B., Ahn J.H., Mattson P., Owens J.D. Programmable Stream Processors // Computer 2003. 36, 54.
2. Russel R.M. The Cray-1 Computer System. Comm // ACM. 1978. 21, 63.
3. Ufimtsev I.S., Martinez T.J. Quantum Chemistry on Graphical Processing Units. 1. Strategies for Two-Electron Integral Evaluation // J. Chem. Theory Comput. 2008. 4, 2.
4. Fatahalian K., Sugerman J., Hanrahan P. Understanding the Efficiency of GPU Algorithms for Matrix-Matrix Multiplication. In Graphics Hardware; Akenine-Moller, T., McCool, M., Eds.; A.K. Peters: Wellesley, 2004.
5. Hall J., Carr N., Hart J. Cache and Bandwidth Aware Matrix Multiplication on the GPU. – University of Illinois Computer Science Department Web Site, 2003. [http://graphics.cs.uiuc.edu/jch/papers/UIUCDCS-R-2003-2328.pdf].
6. Bolz J., Farmer I., Grinspun E., Schroder P. Sparse matrix solvers on the GPU: Conjugate gradients and multigrid // ACM Trans. Graph. 2003, 22, 917.
7. Honda S., Akiba T., Kato Y.S. et al. Am. Chem. Soc. 2008, 130, 15327.
8. Ufimtsev I.S., Martinez T.J. Quantum Chemistry on Graphical Processing Units. Direct Self-Consistent-Field Implementation // J. Chem. Theory Comput. 2009. 5. P. 1004–1015.
9. Klaus C. A Bird's-Eye View of Density-Functional Theory // Brazilian Journal of Physics, 2006. 36. 4A. P. 1318–1341.