GPU 引发的计算化学革命

鲍建樟¹ 丰鑫田² 于建国^{2,*}(¹北京师范大学管理学院系统科学系, 北京 100875; ²北京师范大学化学学院, 北京 100875)

摘要: 综述了图形处理器(GPU)在计算化学中的应用和进展. 首先简单介绍了 GPU 在科学计算中应用的发展, 然后分别详细讲述了迄今几个使用 GPU 和 CUDA(compute unified device architecture, 显卡厂商 Nvidia 推出的计算平台)开发工具设计的量子化学计算和分子动力学(MD)模拟的算法和程序, 尤其对目前唯一完全使用 GPU 技术开发的量子化学计算软件 TeraChem 做了完备的介绍, 包括算法、实现的细节和程序目前的功能. 此外, 本文还对 GPU 在计算化学上将会发挥的作用做出了极为乐观的展望.

关键词: GPU; CUDA; 计算化学; 分子动力学; TeraChem

中图分类号: O641

GPU Triggered Revolution in Computational Chemistry

BAO Jian-Zhang¹ FENG Xin-Tian² YU Jian-Guo^{2,*}(¹Department of Systems Science, School of Management, Beijing Normal University, Beijing 100875, P. R. China;²College of Chemistry, Beijing Normal University, Beijing 100875, P. R. China)

Abstract: Over the last 3 years, the use of graphics processing units (GPU) in general purpose computing has been increasing because of the development of GPU hardware and programming tools such as CUDA (compute unified device architecture). Here, we summarize the progress in algorithms and the corresponding software with regard to computational chemistry using GPU including quantum chemistry and molecular dynamics simulations in detail. We introduce and explore the newly developed TeraChem program, which is unique quantum chemical software and we discuss the algorithms, implementations, and functionality of the program. Finally, we give an optimistic outlook for the use of GPU in computational chemistry.

Key Words: GPU; CUDA; Computational chemistry; Molecular dynamics; TeraChem

1 引言

计算化学是基于计算科学的原理来解决化学问题的一个化学分支学科. 计算科学和技术的任何进步, 都会迅速应用于计算化学, 为计算化学带来长足的进步. 而敏锐的计算化学家们, 也时刻关注着计算科学的每一个新进展, 切盼这些进展能对使用计算机从微观上了解化学和生命现象带来突破和契机. 无疑, 自 2007 年起 GPU 在计算化学上的应

用和导致的一些具有革命意义的进展(且该浪潮的势头还正方兴未艾)对任何与计算化学有关的学科的科学工作者, 都是应该密切关注和深入了解的. 概要介绍 GPU 用于科学计算的基本技术和原理, 穷极搜罗至今有关的成果, 条分缕析地归类分析, 为计算化学家和关注并需要计算化学成果的非计算化学家提供一个引导性的材料, 使他们能把握前进的脉搏, 而争取迅速迎头赶上, 并在其中争得一席

Received: April 15, 2011; Revised: June 9, 2011; Published on Web: June 24, 2011.

*Corresponding author. Email: jianguo_yu@bnu.edu.cn; Tel: +86-10-58802051.

The project was supported by the National Natural Science Foundation of China (20733002, 20873008, 21073014).

国家自然科学基金(20733002, 20873008, 21073014)资助项目

© Editorial office of *Acta Physico-Chimica Sinica*

之地,是写这篇综述的初衷。

在这篇综述中,第二节将简单介绍GPU以及它在数值计算中应用发展的几个阶段及加速在GPU上的计算软件开发的工具;第三节将逐一介绍迄今使用GPU改进量子化学计算算法和程序的努力和一个极有前景的完全基于GPU的量子化学计算软件TeraChem;第四节介绍使用GPU对MD算法的改进和迄今能在GPU上运行的MD程序。最后简单总结并展望基于GPU的计算化学算法和程序的辉煌前景。

2 GPU在数值计算中的应用和软件开发工具

价格合理的GPU (graphics processing units, 图形处理器)最先是为能在计算机游戏中生动真切显示三维图形而设计的计算机硬件。为了更好地显示三维(3D)图形, GPU需要对每个要显示的画面进行大量的浮点运算。GPU最先的设计者们完全没有意识到,他们在每一个GPU中安排的数百个计算单元具有能够进行数量可观的并行运算的特征,使得GPU在今天成了科学计算的宠儿。相比而言,一直在计算机中担负计算使命的每一个CPU (central processing unit, 中央处理器)到今天也最多只能有数个运算单元(多核)。GPU的设计特性使得巨大数目的并行事件同时进入GPU的RAM (random-access memory, 随机存取存储器),即相同操作能在GPU的内存不同部分执行。这属于SIMD (single instruction, multiple data, 单指令多数据)类型。典型的GPU包含许多分列成组的计算单元,每组共享能快速进

入的内存和一个指令单位。这些高度密集的计算单元需要更大的cache和控制单元。GPU在数值计算中表现出的优异性能与已经趋于平缓发展的CPU相比展现出了令人耳目一新的表现(见图1, 数据¹收集到2008年6月)。而之后不到三年的时间(到2011年3月), Nvidia的GeForce GTX 590的GPU的浮点运算峰值和内存带宽就又冲到了 $2488.3 \text{ GFlop} \cdot \text{s}^{-1}$ 和 $2 \times 163.87 \text{ GB} \cdot \text{s}^{-1}$ 。²

GPU结构的本身具备进入科学计算的基本需求。最早将GPU用在数值计算中的企图可以追溯到20年前。³ 尽管之后各种努力和尝试不绝如缕,^{4,5} 但很长时间以来一直没有突破性的进展,原因是缺少科学计算编程的有效工具。当时一般是用特定的汇编语言或图形取向的应用编程接口(API, application programming interface), 如OpenGL或DirectX等编程, 工作量之巨大让人望而生畏。值得一提的工作有最早的Yang等⁶使用OpenGL作为载体编制出程序,在GPU上进行了分子动力学的模拟计算;之后有Anderson等⁷在GPU上进行量子蒙特卡罗(QMC)计算的尝试。

由于GPU硬件快速发展的促动和科学计算高速化的迫切需求,在GPU上用于科学计算简易快捷的编程工具应运而生。它们是ATI Stream Technology⁸和Nvidia的CUDA (compute unified device architecture)。⁹ Stream计算机编程范式(paradigm)最先是ATI公司为在多个计算单元或多种计算硬件,如CPU和GPU间容易进行平行计算设计的,不需特别规定不同计算单元间的交流和传递。之后ATI公司

图1 GPU与CPU的浮点运算峰值和内存带宽的比较

Fig.1 Comparisons of peak GFlop and memory bandwidth of GPU and CPU

The data come from NVIDIA website.¹ GT200=GeForce GTX 280, G92=GeForce 9800 GTX, G80=GeForce 8800 GTX, G71=GeForce 7900 GTX, G70=GeForce 7800 GTX, NV40=GeForce 6800 Ultra, NV35=GeForce FX 5950 Ultra, NV30=GeForce FX 5800

在 2006 年被 AMD 公司收并, 直到 2010 年后期, ATI 品牌彻底被 AMD 取代. CUDA 是一个类似于 C 语言的高级、直观的编程语言, 并能被 C 语句调用. NVIDIA 公司最近还推出了 Parallel Nsight,⁹ 使 CUDA 能在微软公司最为普及的软件开发平台 Visual Studio 上使用. 这惠及了数百万的软件开发工作者. 可以毫不夸张地说, CUDA 给 GPU 在科学计算上应用的飞速发展插上了翅膀. 一时间诸多的与数值计算有关的科学家趋之若鹜, 这已经在改变着科学计算的面貌了.

现有数值计算的程序编码已是浩如烟海, 无论如何方便的在 GPU 上的编程工具, 如 CUDA, 仍无法重复所有已有的编码和算法. 若有简便的工具将现有的数值计算的程序转换到 GPU 上, 将善莫大焉. 在这个方向上的尝试和进展包括能在 GPU 上计算的量子化学和固体物理中常用的一些算法中的子程序库, 如进行傅里叶变换的 CUFFT¹⁰ 和线性代数库 CUBLAS¹¹ 和 MAGMA.¹²

一开始影响 GPU 在科学计算上普及使用的另一障碍是早期 GPU 较低的数值精度, 随着 NVIDIA 公司的支持 32 位浮点运算精度的 G80 系列 GPU 卡的推出和 64 位的 GeForce 8800 GTX GPU 卡的推出, 用于科学计算的精度问题得到了彻底的解决. 尽管在 64 位浮点运算精度下, 运算速度会有所损失.

3 使用 GPU 的量子化学计算方法和程序

量子化学计算的基本方法, 从头算(*ab initio*)和密度泛函理论(DFT)的最主要计算工作, 是求得双电子排斥积分(也包括 DFT 中的交换-相关积分的数值求取)和生成 Fock 矩阵以及矩阵的对角化.

Yasuda¹³ 第一个认识到可使用 GPU 来提高计算电子排斥积分(ERI)的速度. 他使用 GeForce 8800 GTX GPU 卡, 发展出了使用 GPU 计算 DFT 的 **KS** 矩阵的 Coulomb 积分的算法, 但仅适用于 *s* 和 *p* 基函数. 这个算法在 Taxol 和 Valinomycin 的小基组和 LSDA DFT 计算里, 使用 GPU 计算 Coulomb 势, 仅需 CPU 常规计算时间的 1/3. 尽管使用的 GPU 仅支持单精度浮点数运算, 但仍然有可以接受的精度. Yasuda 的算法是在 Gaussian 03 程序¹⁴ 上完成的.

GPU 对积分的计算只能包括 *s* 和 *p* 基函数是远远不够的. 将 GPU 上的积分计算扩展到 *d* 及更高角量子数的基函数的困难在于目前最大的单个 GPU

也无法容纳这些高角量子数的计算壳层. 如对于一个(*fff*)双电子积分的壳层, 在中间计算时共需要 5376 个浮点数, 它们要做多次的乘法运算, 最终需约 10^4 个浮点数. 对于双精度, 这相应于 123008 bytes, 远远超出了目前最大单片 GPU 的容量. 有人在等待着很快有更大的 GPU 上市, 但 Asadchev 等¹⁵ 另辟蹊径, 近来提出了一个使用 CUDA 能计算到 *g* 型基函数的算法. 他们将计算的中间结果存放在装置的内存, 而不是 GPU 中, 并将这些壳层块的积分重新排序, 满足这些积分的频繁调用. 这样做带来的问题是为了适应所有积分类型的有效排列的方式, 原来在 CPU 下使用的大量程序需要重写. 他们采用了由一个个模块自动生成的方式来解决这个问题. 使用 GTX275 和 Tesla T10 GPU 卡, 他们与 GAMESS¹⁶ 比较了双电子排斥积分的运算速度. CPU 能达到 $1 \text{ GFlop} \cdot \text{s}^{-1}$ (每秒钟进行 1 giga 浮点运算), 而 GPU 在单精度和双精度运算时能分别达到 50 和 25 $\text{GFlop} \cdot \text{s}^{-1}$.

对 DFTGGA 中的 XC(交换-相关)势的数值计算, 仍是 Yasuda¹⁷ 首先提出了在 GPU 上计算的算法. 他采用的策略是在仅将 XC 较费时的步骤交由 GPU, 其余仍由 CPU 处理. 使用 NVIDIA 的 GeForce 8800 GTX 卡, 与 2.8 GHz 的 Intel Pentium 4 相比, 该步计算的加速能达到 40 倍.

Aspuru-Guzik 等^{18,19} 将开发基于 GPU 的量子化学算法的注意力放在了包括相关能校正的 RI-MP2 (RI: resolution-of-the identity)方法上. 除了对于含过渡金属的体系, MP2 计算的平衡几何构型优于广为使用的 DFT. MP2 计算中最为耗时的是原子轨道基的积分向分子轨道基的变换. 这一般是体系基函数数目 *N* 的 5 次方 ($O(N^5)$). 引入 RI 积分近似可减少计算时间而又没有明显的精度损失. Aspuru-Guzik 等分析 RI-MP2 计算各个步骤对计算机时间的需求, 认识到 RI-MP2 计算中主要的时间消耗是在从半变换的三指标积分变换为分子轨道积分时的矩阵相乘. 他们使用 CUBLAS¹⁶ 的函数在 GPU 上执行矩阵相乘的运算(将 CUBLAS 的函数引入量子化学计算这还是首次). 据他们自己的测试, 在单精度下, 计算速度能达到原来使用 CPU 的 4.3 倍.¹⁸ 使用 NVIDIA Tesla C1060 GPU 和 4 GB 的内存, 他们测试了有 168 个原子的分子 Valinomycin, 在单精度、混合精度和双精度下分别能比 CPU 提速 13.8、10.0 和 7.8 倍, 在双精度下, 误差接近于零. 他们的计算是植根于另

一广为使用的商品化量子化学计算程序 Q-Chem.²⁰

在此还应提及一个不被化学家熟悉的、使用 Daubechies 波, 而不是 Gaussian 型的基函数的程序包 BigDFT.²¹ BigDFT 使用 CUDA 在 GPU 上编程, 达到了极高的并行效率, 使用 GPU 的版本能比 CPU 的提速达 6 倍.

量子化学计算程序 Firefly²² 的前身是著名的量子化学程序 GAMESS¹⁶ 的变种 PC GAMESS. 近来 Firefly 在它的 MP4(SDTQ) 计算中使用了 GPU, 对于 Cl⁻(HF)₅ 分子使用 6-311+G(2d,2p) 基组进行单点计算, 在使用四个 GPU(2 GTX295+2 C1060) 时, 计算速度提高了近 10 倍.²³

通用的量子化学程序包 Q-Chem²⁰ 的一个突出特点是具有运算速度非常快的 DFT 计算功能. 近来 Q-Chem 的开发者们 Gan 等²⁴ 也将 GPU 用于进一步提高 DFT 计算速度. 结合已有的可用于 DFT 一些步骤的 GPU 算法和 Q-Chem 在 DFT 计算上的改进, 提出一个方案(task based programming, object-oriented design, OPENMP for multithreading, MPI for cross node communications, 即基于作业编程、目标取向设计、用 OPENMP 作多线程和用 MPI 处理节点间的交通). 作者自己的测试计算对 Taxol 分子(C₄₇H₅₁NO₁₄), 使用一片 C1060 GPU, 计算速度提高了 24 倍, 若使用 4 片 GPU, 速度提高达到了 102 倍.

对将 GPU 使用到量子化学计算极为关注和付出努力的还有美国 Iowa 州立大学的 Gordon 教授. Gordon 教授的研究小组是 GAMESS 程序¹⁶ 的维护者, 并不断开发和发展. 最先对 GPU 计算精度的不足曾影响了 Gordon 教授对之的投入, 随着双精度的 GPU 卡的出现, Gordon 教授和 GAMESS 研发小组变成了积极的参与者. Gordon 教授和他的研发小组正在重写 GAMESS 中的程序, 使之能在 GPU 上高效使用, 相信很快就有这样的 GAMESS 版本推出.²⁵

以上提到的近几年提出的使用 GPU 提高量子化学计算速度的算法中, 除了 BigDFT,²¹ 其余都是利用现有软件包, 改善其中部分算法, 在某一特定计算步骤, 使计算速度得到提高. 而刚于 2010 年推出的 TeraChem 程序则完全使用 CUDA 概念, 重写了 GPU 上的量化计算的基本程序.²⁶ TeraChem 是由美国 Stanford 大学的 Ufimtsev 和 Martinez 编写完成的, 由美国 PetaChem 公司于 2010 年推出商业化版本.²⁷ 由于他们在 GPU 的量子化学程序的前驱性的工作和突出贡献, Martinez 教授被 Gordon 称为美国

量子化学家的 GPU 的领头人.²⁵

Ufimtsev 和 Martinez²⁸ 提出的计算双电子排斥积分基于 McMurchie-Davidson 框架, 检验了三种分配 GPU/CPU 工作和处理任务分块的模式(1B1CI、1T1CI 和 1T1PI), 先进行由 H 原子组成的体系的(ss|ss)型积分的运算, 然后推广到包括由 *p* 函数组成的基的排斥积分. 他们发现若从 Gaussian 基的双电子排斥积分直接进入 Fock 矩阵, 而不是如传统的将 Gaussian 轨道基的排斥积分收缩到 Slater 轨道(STO), 在 GPU 下能有极高的效率. 对于由 64 个 H 组成的体系, 使用 6-311G 和三种模式计算排斥积分, 分别耗费 7.086、0.675 和 0.428 s, 加上用 CPU 的预计算的 0.009 s 和 GPU-CPU 转换的 0.883 s, 最快的 1T1PI 模式共需 1.320 s. 而使用 GAMESS 在单 CPU 下则需 170.8 s. Ufimtsev 和 Martinez²⁹ 随后报道了完全使用 GPU 进行 Hartree-Fock 计算的方法. 基于以前提出的 1T1PI 处理双电子排斥积分的方法²⁸, 他们提出了 Hartree-Fock 的 *J* 矩阵和 *K* 矩阵在 GPU 下依据大小分块处理的算法, 还研究了多 GPU 的并行化处理: 每个 GPU 处理自己的 *J* 矩阵子块, 计算 *K* 矩阵时每个 GPU 处理 *K* 矩阵的不同的行. 使用三个 GPU 卡(GTX280)做检验, 能达到一个 GPU 卡的 2.0–2.8 的效率. 他们测试了从 24 个原子到 453 个原子组成的分子的 Hartree-Fock 单点能量计算. 使用 3-21G 和 6-31G 基组以及一个 GPU 卡(GeForce 280 GTX), 在双精度下, 与 GAMESS 相比, 速度能提高 40 到 600 多倍, 而总能量的差别可忽略不计.

Ufimtsev 和 Martinez 系列文章的第三篇²⁶ 是关于能量解析梯度计算的 GPU 处理, 包括 Coulomb 和交换部分计算的处理. 加上多 GPU 卡的并行计算的处理, 该方法可用于 Hartree-Fock 能量和 DFT 能量的解析梯度的计算. 使用 GPU 程序和单个 GPU 卡(GeForce 295 GTX), 与在 4 核 2.66 GHz CPU 上运行的 GAMESS 程序相比, 从小到大的分子, GPU 仍然可有数倍到超过百倍的速度提高.

这样, 他们提出的在 GPU 上运行的量子化学计算的算法, 涵盖了基本的从头算/DFT 的全部三个耗时步骤(双电子积分、Hartree-Fock/Kohn-Shan 矩阵处理和能量的解析梯度). 基于此, Martinez 推出了他们的 TeraChem 程序.²⁶ 2010 年春推出的为 1.05 版, 能进行闭壳层的 Hartree-Fock 和 DFT 的单点能量计算, 基于解析梯度的几何构型极小化和过渡态优化的计算, 以及基于以上计算(能量和解析梯度)

的动力学模拟(MD)和简单的QM/MM计算. 使用TeraChem, 很容易检验和实现上述他们的三篇论文的结果. 最近(2011年4月)推出的v1.45版已能进行包括 d 轨道和含过渡金属原子体系的HF和DFT计算, 并能达到与 sp 轨道差不多的速度提高. 具体算法的改进和实现还未见有详细报道.

TeraChem更为不凡, 需引起密切关注的原因是这是第一个完全基于GPU, 从零开始编制的量子化学计算程序. 本文引述的在Ufimtsev和Martinez的工作之前或之后的关于GPU的量子化学计算的算法的文章, 虽都基于某一特定量子化学程序包实现, 但至今还都没有能进入公开发行的程序包中, 有些或许只能永远停留在纸面上. 而TeraChem一进入人们的视野, 就以完备的商品软件的态度出现. 据PetaChem网站披露, TeraChem在2010年春天一推出, 短短半年就有超过4000份的试用版本发出, 展现出蓬勃的生命力.

量子蒙特卡罗(QMC)是求解与时间无关的薛定谔方程的精确方法之一,³⁰ 并且QMC很容易并行化. Anderson等⁷在2007年就提出了在GPU上加速QMC计算的方案. 使用CUDA核优化缓存使用, 在GPU上进行需求很大的并行计算. 对于8–32个原子的体系, 比CPU上的执行, 速度能有5倍的提高. Meredith等³¹在单精度的GPU上用量子簇近似, 使用规整晶格上的二维Hubbard近似研究超导在临界温度下的混乱效应, 在GPU上使用CUBLAS库执行矩阵乘法, 得到与CPU上的双精度可比的结果.

4 分子动力学算法和程序中的GPU

MD计算对生物分子的模拟自McCammon等对酶的计算为肇基,³² 当时的计算水平是500个原子, 模拟的时间10 ps. 之后飞速发展, 到了今天, 模拟的生物体系包括数百万个原子, 模拟时间能持续到微秒都已成常规. 而科学家对生命现象的深入了解仍在向计算化学家不断提出更高的要求. 如对蛋白质折叠的MD模拟研究, 折叠行为约需毫秒来完成, 如以纳秒为步长做模拟, 且每步计算花费一天计, 这样的模拟就需执行一百万天, 即差不多3000年. 这确实是对计算化学家提出了严峻的挑战. 虽然有人花费巨资搭建豪华计算机系统, 如日本的总耗资超过一千五百万美元的MDGRAPE-3系统,^{33,34} 这是使用专门设计的硬件加速搜寻表以加速直接空间非键的计算. 但这样的耗费巨资并为某一特定

问题专门设计的硬件完全无法做广泛的推广, 且不说耗资惊人了. 而GPU适时的出现为MD对生物的模拟计算带来了曙光.

由于在GPU下编程的困难, 早期试图在GPU上提高MD计算速度的努力有许多是走入迷途的, 报导的结果也是使人信疑参半. 至今被普遍接受的GPU上MD算法的成功改进仍属本文开始提及的Yang等⁶的工作, 尤其值得赞许的是该工作是在CUDA推出之前, 直接使用OpenGL编程, 在GPU上实现需惊人的工作量, 并达到了10倍以上速度的提高. 虽然他们的算法相当简捷, 在模拟过程中仅包含Lennard-Jones相互作用和建立一个邻接目录, 而缺乏对生物体系模拟的许多计算步骤, 但这个先驱性的工作仍有力地证实了GPU能在MD模拟上发挥重要作用.

早期使用GPU改进MD运算速度的尝试大都局限于使用简化的模型, 或者只针对于特定的问题做处理, 而不是着眼于对真正MD的全面改进. Liu等³⁵是第一个使用CUDA来对MD的计算进行改进的. 他们仅在GPU上计算简化的范德华势, 而避免了对凝聚态体系MD模拟的复杂的乘积运算. Stone等³⁶在他们的文章里仔细讨论了一系列用于分子模拟的算法, 提出了第一个直接空间范德华势和静电势在GPU上合并处理的方法, 但该方法并不包括任何实际的MD计算, 而仅仅是在被模拟体系临近库仑势的时间平均的计算中使用GPU, 并引入到MD程序包NAMD³⁷中, 诞生了NAMD的GPU加速的新版.

之后还有一些使用GPU对MD算法的局部改进,^{38,39} 直到Anderson等⁴⁰将共价项的贡献与使用GPU计算的范德瓦尔和简谐键势项加入到他们的HOOMD程序包⁴¹中, 用于研究非离子液体. 可以认为, GPU版的HOOMD朝向真正完整的GPU的MD程序前进了一大步.

真正能称之为使用GPU的MD程序产品是到2009/2010年才出现的. 至今只有三个MD程序包可认为基本满足, 它们分别是Harvey等⁴²的ACEMD、Friedrichs等⁴³的OpenMM和Philips等³⁷的NAMD.

ACEMD可谓是第一个支持全部GPU加速功能的处理凝聚相的MD程序包, 可处理周期性边界和短程、长程的静电作用. 使用GPU加速后, 有很明显的速度提高. OpenMM是一个程序库, 起先仅是对于小的和中等大小的体系使用非键项的间接求

和方法来计算溶剂的广义 Born 模型, 之后 Eastman 和 Pande⁴⁴ 对之改进, 显性计算进行溶剂的 MD 模拟. 最新推出(2011 年 3 月 11 日)的 OpenMM 版是 v3.0. 著名 MD 程序包 GROMACS⁴⁵ 新近的 GPU 版将 OpenMM 纳入其中, 可在 GPU 上大大提高 MD 计算的速度. NAMD 的 v2.7b2 版也添加了 GPU 加速的明确进行溶剂化计算的功能, 在对一个由 25095 个原子组成的生物体系的模拟计算, 速度提高到了 8.6 倍. NAMD 最新的版本是 v2.8b1, 仍是以 CPU 计算为主, 在多节点间进行 MD 计算, 而将之置于 GPU 的环境, 使用单片或多片 GPU 来提高 CPU 的单个节点内的计算效率以达到提高 MD 计算速度的目的.

AMBER 程序可谓是 MD 计算机程序家族中最著名的一员,⁴⁶ 由一些使用 AMBER 力场⁴⁷ 进行与 MD 有关计算的程序包组成. 最近 AMBER 程序家族的 PMEMD 程序在显性溶剂 PME (particle mesh Ewald) 和广义 Born 模型 (GB, generalized Born) 的处理中使用 Nvidia 的 GPU, 并因之推出了 AMBER11 版,⁴⁸ 但使用 GPU 后的速度提高和计算精度还都有待于检验.

最近推出的另一 MD 程序 HOOMD-BLUE 0.9.2⁴¹ 也声称支持 GPU 的运算. 算法的作者声称他们的整个 MD 计算都在 GPU 上进行, 单片 GPU 运算的速度相当于 30 个微处理器.⁴⁰

尽管有诸多的对 MD 在 GPU 上应用的算法的创立和改进, 和相应的能在 GPU 上运行的 MD 程序的推出, 其中某些还声称是完全使用了在 GPU 上基于 CUDA 开发的新算法, 但相对于 GPU 的量子化学计算程序能对速度有 50–100 倍的提高, GPU 上的 MD 程序的速度提高还不尽如人意(最高速度提高的倍数大约为 10, 但据我们验证, 除特殊情况, 基本无法达到这个速率), 这是由 MD 计算的本质决定的. 尽管今天的 CPU, 甚至 GPU 的计算速度已足够快, 但硬件间数据传递的速度仍是瓶颈, 而 MD 计算却是非常依赖和需要这些传输. GPU 运算速度的提高无助于这些传输, 甚至加重了节点间数据传输的压力. 这就是为什么至今 GPU 上的 MD 程序仅限于单个 GPU (OpenMM 和 AMBER) 或单个节点内的多个 GPU (ACEMD). 仅 NAMD 试图在多节点上应用 GPU 改进 MD 的计算速度, 试图模拟非常大的体系, 而不是优化单节点内的运行, 这需要数量众多的 GPU 卡, 而仅有差强人意的速度提高, 且“性价比”还相当低.

5 结 语

近年由 CUDA 带动的 GPU 在科学计算上的蓬勃应用和发展, 不可能不进入计算化学领域. 自 2010 年以来, 接连有 GPU 在计算化学(包括量子化学计算和 MD 模拟)的综述文章发表, 如 Götz 等⁴⁹ 题为“Quantum Chemistry on GPU”的综述, 和 Xu 等⁵⁰ 对 GPU 在生物分子 MD 模拟的进展的综述(本文的一些材料间接引自这两篇综述), 最近美国 Chemical & Engineering News 上的一篇题为“GPU 革命 (The GPU Revolution)”的短文也对 GPU 将对计算化学的贡献和前景给出了极为乐观的评价和期许.²⁵

纵观由三年来多方努力和如雨后春笋般的在 GPU 上的量子化学计算和 MD 模拟算法和程序的呈现, 仅有 Ufimtsev 和 Martinez 辛勤创立的 TeraChem 软件是完全从零的基础上、完全为 GPU 计算设计和实现, 却也达到了计算速度的可喜提高, 而其它量子化学程序和差不多所有的 MD 程序都还有各种各样的缺憾和不足. TeraChem 或许会成为量子化学计算的一个新里程碑. 如同四十年前 Gaussian 70 和九十年代初量子化学软件的商品化大潮, 呈现以二十年为周期的飞跃.

同时, Nvidia 会不断推出容量更大、速度更快的 GPU, 而价格也会愈来愈低, CUDA 也会不断扩充新功能, 并愈来愈完善, 使得在 GPU 上编制科学计算程序成为常规. 而 AMD 的 ATI Stream Technology 和相应的硬件或许在很近的将来会成为 Nvidia 的 GPU 卡和 CUDA 的有力竞争者, 而且 GPU 在科学计算上的应用也将出现百花齐放的繁盛局面. 这些都使我们相信, 我们处在一个计算化学新时代来临的前夜.

References

- (1) NVIDIA CUDA. Compute Unified Device Architecture Programming Guide Version 3.0. http://www.nvidia.com/object/cuda_develop.html (accessed March 6, 2010).
- (2) Comparison of Nvidia Graphics Processing Units. http://en.wikipedia.org/wiki/Comparison_of_Nvidia_graphics_processing_units (accessed March 6, 2010).
- (3) Lengyel, J.; Reichert, M.; Donald, B. R.; Greenberg, D. P. *Comput. Graph.* **1990**, *24*, 327.
- (4) Bohn, C. A. *Joint Conference on Intelligent Systems 1999 (JCIS'98)* **1998**, *2*, 64.
- (5) Hoff, K. E., II.; Culver, T.; Keyser, J.; Ming, L.; Manocha, D. Fast Computation of Generalized Voronoi Diagrams Using Graphics hardware. In *Proceeding of SIGGRAPH 99*, Danvers,

- August 8–13, 1999; Assison-Wssley Publishing Company, 1999, 277–286.
- (6) Yang, J.; Wang, Y.; Chen, Y. *J. Comput. Phys.* **2007**, *221*, 799.
- (7) Anderson, A. G.; Goddard, W. A., III.; Schroder, P. *Comput. Phys. Commun.* **2007**, *177*, 265.
- (8) ATI Stream Technology, <http://www.amd.com/US/PRODUCTS/TECHNOLOGIES/STREAM-TECHNOLOGY/Pages/stream-technology.aspx> (Accessed April 13, 2011).
- (9) CUDA: Santa Clara, CA. http://www.nvidia.com/object/cuda_home_new.html (accessed April 13, 2011).
- (10) NVIDIA: Santa Clara, CA, CUFFT Library. http://developer.download.nvidia.com/compute/cuda/2_3/toolkit/docs/CUFFT_Library_2.3.pdf (accessed March 6, 2010).
- (11) NVIDIA: Santa Clara, CA, CUBLAS Library 2.0. http://developer.download.nvidia.com/compute/cuda/2_0/docs/CUBLAS_Library_2.0.pdf (accessed March 6, 2010).
- (12) Innovative Computing Laboratory, University of Tennessee, Matrix Algebra on GPU and Multicore Architectures. <http://icl.cs.utk.edu/magma> (accessed March 6, 2010).
- (13) Yasuda, K. *J. Comput. Chem.* **2008**, *29*, 334.
- (14) Frisch, M. J.; Trucks, G. W.; Schlegel, H. B.; *et al.* *Gaussian 03*, Revision B.01; Gaussian Inc.: Pittsburgh, PA, 2003.
- (15) Asadchev, A.; Allada, V.; Felder, J.; Bode, B. M.; Gordon, M. S.; Windus, T. L. *J. Chem. Theory Comput.* **2010**, *6* (3), 696.
- (16) (a) Schmidt, M. W.; Baldridge, K. K.; Boatz, J. A.; Elbert, S. T.; Gordon, M. S.; Jensen, J. H.; Koseki, S.; Matsunaga, N.; Nguyen, K. A.; Su, S.; Windus, T. L.; Dupuis, M.; Montgomery, J. A. *J. Comput. Chem.* **1993**, *14*, 1347.
- (b) Gordon, M. S.; Schmidt, M. W. Advances in Electronic Structure Theory: In *Theory and Applications of Computational Chemistry: the First Forty Years*; Dykstra, C. E., Frenking, G., Kim, K. S., Scuseria, G. E., Eds.; Elsevier: Amsterdam, 2005; p 1167.
- (17) Yasuda, K. *J. Chem. Theory Comput.* **2008**, *4*, 1230.
- (18) Kermes, S.; Olivares-Amaya, R.; Vogt, L.; Shao, Y.; Amador-Bedolla, C.; Aspuru-Guzik, A. *J. Phys. Chem. A* **2008**, *112*, 2049.
- (19) Olivares-Amaya, R.; Watson, M. A.; Edgar, R. G.; Vogt, L.; Shao, Y.; Aspuru-Guzik, A. *J. Chem. Theory Comput.* **2010**, *6*, 135.
- (20) Shao, Y. H.; Fusti-Molnar, L.; Jung, Y. S. *et al. Phys. Chem. Chem. Phys.* **2006**, *8*, 3172.
- (21) Genovese, L.; Ospici, M.; Deutsch, T.; Mehaut, J. F.; Neelov, A.; Goedecker, S. *J. Chem. Phys.* **2009**, *131*, 34103.
- (22) Granovsky, A. A. Firefly version 7.1.G. <http://classic.chem.msu.su/gran/firefly/index.html> (accessed April 4, 2011).
- (23) <http://classic.chem.msu.su/gran/gamess/cuding.html> (accessed April 4, 2011).
- (24) Gan, Z.; Shao, Y.; Kong, J.; Olivares-Amaya, R.; Aspuru-Guzik, A. http://www.nvidia.com/content/GTC/documents/1050_GTC09.pdf (accessed April 4, 2011).
- (25) Wolf, L. *Chemical and Engineering News* **2010**, *88*, 27.
- (26) Ufimtsev, I. S.; Martinez, T. J. *J. Chem. Theory Comput.* **2009**, *5*, 2619.
- (27) TeraChem. <http://www.petachem.com> (accessed March 6, 2010).
- (28) Ufimtsev, I. S.; Martinez, T. J. *J. Chem. Theory Comput.* **2008**, *4*, 222.
- (29) Ufimtsev, I. S.; Martinez, T. J. *J. Chem. Theory Comput.* **2009**, *5*, 1004.
- (30) Ceperley, D.; Alder, B. *Quantum Monte Carlo. Science* **1986**, *231*, 555.
- (31) Meredith, J. S.; Alvarez, G.; Maier, T. A.; Schulthess, T. C.; Vette, J. S. *Parallel Comput.* **2009**, *35*, 151.
- (32) McCammon, J. A.; Gelin, B. R.; Karplus, M. *Nature* **1977**, *267*, 585.
- (33) Susukita, R.; Ebisuzaki, T.; Elmegreen, B. G.; Furusawa, H.; Kato, K.; Kawai, A.; Kobayashi, Y.; Koishi, T.; McNiven, G. D.; Narumi, T.; Yasuoka, K. *Comput. Phys. Commun.* **2003**, *155*, 115.
- (34) Narumi, T.; Ohno, Y.; Noriyuk, F.; Okimoto, N.; Suenaga, A.; Yanai, R.; Taiji, M. In *From Computational Biophysics to Systems Biology: MDGRAPE-3*; Meinke, J., Zimmermann, O., Mohanty, S., Hansmann, U. H. E. Eds.; J. von Neumann Institute for Computing: Jülich, 2006; p 29.
- (35) Liu, W.; Schmidt, B.; Voss, G.; Müller-Wittig, W. In *High Performance Computing-HiPC 2007: Lecture Notes in Computer Science*; Aluru, S., Parashar, M., Badrinath, R., Prasanna, V. K. Eds.; Springer, Berlin/Heidelberg, 2007; Vol. 4873, p 185.
- (36) Stone, J. E.; Phillips, J. C.; Freddolino, P. L.; Hardy, D. J.; Trabuco, L. G.; Schulten, K. *J. Comput. Chem.* **2007**, *28*, 2618.
- (37) Phillips, J. C.; Stone, J. E.; Schulten, K. Adapting a Message-Driven Parallel Application to GPU Accelerated Clusters. In *SC '08: Proceedings of the 2008 ACM/IEEE conference on Super Computing*, 1–9, IEEE Press, Piscataway, NJ, USA, 2008.
- (38) van Meel, J. A.; Arnold, A.; Frenkel, D.; Portegies Zwart, S. F.; Belleman, R. G. *Mol. Simulat.* **2008**, *34*, 259.
- (39) Rapaport, D. C. *Comput. Phys. Commun.* **2011**, *182*, 926.
- (40) Anderson, J. A.; Lorenz, C. D.; Travesset, A. *J. Comput. Phys.* **2008**, *227*, 5342.
- (41) HOOMD: General Purpose Molecular Dynamics on Multiple GPUs Implemented Using CUDA Joshua A. Anderson Path to Petascale: Adapting GEO/CHEM/ASTRO Applications for Accelerators and Accelerator Clusters, April 2009. <http://www.ncsa.uiuc.edu/Conferences/accelerators/agenda.html> (accessed April 4, 2011).
- (42) Harvey, M. J.; Giupponi, G.; De Fabritiis, G. *J. Chem. Theory Comput.* **2009**, *5*, 1632.
- (43) Friedrichs, M. S.; Eastman, P.; Vaidyanathan, V.; Houston, M.; Le Grand, S.; Beberg, A. L.; Ensign, D. L.; Bruns, C. M.; Pande, V. S. *J. Comput. Chem.* **2009**, *30*, 864.

- (44) Eastman, P.; Pande, V. S. *J. Comput. Chem.* **2010**, *31*, 1268.
- (45) GROMACS <http://www.gromacs.org> (accessed April 4, 2011).
- (46) Pearlman, D. A.; Case, D. A.; Caldwell, J. W.; Ross, W. S.; Cheatham, T. E., III.; DeBolt, S.; Ferguson, D.; Seibel, G.; Kollman, P. *Comp. Phys. Commun.* **2005**, *91*, 1.
- (47) Cornell, W. D.; Cieplak, P.; Bayly, C. I.; Gould, I. R.; Merz, K. M., Jr.; Ferguson, D. M.; Spellmeyer, D. C.; Fox, T.; Caldwell, J. W.; Kollman, P. A. *J. Am. Chem. Soc.* **1995**, *117*, 5179.
- (48) Case, D. A.; Darden, T. A.; Cheatham, T. E., III.; Simmerling, C. L.; Wang, J.; Duke, R. E.; Luo, R.; Walker, R. C.; Zhang, W.; Merz, K. M.; Roberts, B.; Wang, B.; Hayik, S.; Roitberg, A.; Seabra, G.; Kolossváry, I.; Wong, K. F.; Paesani, F.; Vanicek, J.; Liu, J.; Wu, X.; Brozell, S. R.; Steinbrecher, T.; Gohlke, H.; Cai, Q.; Ye, X.; Wang, J.; Hsieh, M. J.; Cui, G.; Roe, D. R.; Mathews, D. H.; Seetin, M. G.; Sagui, C.; Babin, V.; Luchko, T.; Gusarov, S.; Kovalenko, A.; Kollman, P. A. *AMBER 11*; University of California: San Francisco.
- (49) Götz, A. W.; Wölfe, T.; Walker, R. C. *Annual Reports in Computational Chemistry*; Elsevier: Amsterdam, 2010; Vol. 6, p 21.
- (50) Xu, D.; Williamson, M. J.; Walker, R. C. *Annual Reports in Computational Chemistry*; Elsevier: Amsterdam, 2010; Vol. 6, p 1

中国化学会第28届学术年会第一轮通知

中国化学会第28届学术年会将于2012年4月13–16日在四川省成都市举行。本届年会由中国化学会主办,四川大学承办,会期4天,4月12日报到。2012年正值中国化学会成立八十周年,届时将举办相关庆祝活动。

本届年会设立学术分会19个、专题论坛4个。学术交流形式包括大会特邀报告(PL)、分会邀请报告(I)、口头报告(O)和墙报(P)。年会面向全国广大化学工作者征集学术论文(摘要),并印制论文摘要集,欢迎踊跃投稿,积极参会。

年会期间将举办新技术、新产品与新仪器成果展览以及人才交流活动,欢迎相关企业、高校、科研院所积极参与。

会议详细内容请见会议网站: <http://www.ccs.ac.cn/confab/huiyi.html>