

<<面向千万亿次计算的算法与应用>>

图书基本信息

书名：<<面向千万亿次计算的算法与应用>>

13位ISBN编号：9787302177227

10位ISBN编号：7302177228

出版时间：2008-8-1

出版时间：清华大学出版社

作者：（美）巴德尔

页数：356

译者：都志辉 等

版权说明：本站所提供下载的PDF图书仅提供预览和简介，请支持正版图书。

更多资源请访问：<http://www.tushu007.com>

## &lt;&lt;面向千万亿次计算的算法与应用&gt;&gt;

## 前言

在过去的几十年内，在高性能计算应用、算法和体系结构的支持下，在科学、工程和社会领域发生了许多突破性的进展。

这些强有力的工具为研究人员、教育工作者和实践工作者提供了这样的能力，就是通过计算，可以把从全球收集获得的数据，转换为解决我们社会最具有挑战性问题的解决方案。

持续推动高性能计算发展的一支重要的力量是针对“前沿性里程碑”问题产生的一些社团联盟，比如它们会提出代表这一领域下一阶段进展的技术目标。

在19世纪90年代，高性能计算组织希望实现万亿次（每秒 $10^{12}$ 的浮点操作）的计算能力，万亿次计算导致了一些新的发现，比如设计新的药物来对抗艾滋病和其他的疾病；对自然现象的仿真可以达到前所未有的精度，比如地震和飓风；对大系统，比如宇宙，和小系统，比如比细胞还小的一些系统有了更好的理解；现在我们将要在于万亿次（每秒 $10^{15}$ 的浮点操作）级别的机器上进行计算。

一些研究组织正在考虑亿亿次（每秒 $10^{18}$ 的浮点操作）级别的计算将会是什么样子，他们处在这一研究的早期阶段。

在走向“下一个前沿计算”的过程中，一个假设就是高性能计算体系结构的下一个研究前沿将会提供新的极大的容量和能力，这将使得以前需要大量资源但是又得不到满足的用户直接受益，同时对其他的用户将带来长期的裨益。

但是大规模高性能计算用户都知道，高效使用前沿超级计算系统能力，同不断增加系统容量和能力相比，如果不是同等重要，就是重要性更高，我们需要投入大量的时间、人力、软件和硬件基础设施来充分利用这些非同寻常的系统。

经验表明，通过开发可扩展的算法，模型，仿真，分析，库和应用组件，可以充分利用这些前沿系统所提供的容量和能力，因此它与建造和部署该前沿系统本身一样，是非常具有挑战性的工作。

对于未来几年内可以支持千万亿次操作的应用程序代码，不管用什么样的处理器技术，将会需要几十万甚至更多的处理器核心。

目前，几乎还没有可以很容易扩展到这种规模的高性能计算程序代码，大量的程序代码开发工作对于挖掘新的千万亿次系统的潜能是非常重要的。

扩展到千万亿次系统涉及的工作包括：提高物理模型，数学抽象，近似和其他的应用组件。

我们需要提高各种解决方案的算法来增加结果的准确性。

输入数据集需要在分辨率上进行提高（产生更多的数据），或者是用于测量的输入数据的精度需要提高。

每一个应用或者算法的提高都会给开发千万亿次代码带来实质性的挑战，会从新应用执行得到的领域结果中推动新的计算机科学的发现。

本书是那些在千万亿次计算领域前沿性研究成果的结晶。

作者试图通过汇集在高性能计算和计算科学领域最成熟和已积累丰富经验的应用的最新成果，来解决在开发千万亿次系统应用代码中的一些挑战性问题，这些应用能够在新的千万亿次系统产生和部署之前，就提前针对其体系结构的特点来进行开发。

开发千万亿次应用将会需要丰富的关于千万亿次系统硬件和系统软件方面的知识，我们需要大量的时间来做扩展，测试，评估以及优化千万亿次代码、库、算法、系统软件的工作，所有这些工作都将基于即将构建的新系统来进行。

## <<面向千万亿次计算的算法与应用>>

### 内容概要

在过去的几十年内，在高性能计算应用、算法和体系结构的支持下，科学、工程和社会领域发生了许多突破性的进展。

本书是那些在千万亿次计算领域前沿性研究成果的结晶。

作者试图通过汇集在高性能计算和计算科学领域最成熟和已积累丰富经验的应用的最新成果，来解决在开发千万亿次系统应用代码中的一些挑战性应用，这些应用能够在新的千万亿次系统产生和部署之前，就提前针对其体系结构的特点来进行开发。

开发千万亿次应用需要丰富的关于千万亿次系统硬件和系统软件方面的知识，需要大量的时间来做扩展、测试、评估以及优化千万亿次代码、库、算法、系统软件的工作。

所有这些工作都将基于即将构建的新系统来进行。

对于高性能计算来说，这是一个激动人心的时代，这个时代将会在一个新的规模和尺度上产生大量史无前例的新发现，这些发现将会对科学和社会带来切实的好处。

本书可以让你对千万亿次应用和算法前沿性挑战性研究工作的第一次浪潮有一个概观，为目前和未来的千万亿次研究提供非常必要的基础。

## <<面向千万亿次计算的算法与应用>>

### 作者简介

David A Bader是高性能计算所的所长，教授，长期从事计算科学与工程领域研究，该研究所隶属于佐治亚理工学院计算学部。

他在1996年从马里兰大学（University of Maryland）获得博士学位，后赢得美国国家自然科学基金支持在实验计算机科学方面从事博士后研究。

他获得了美国自然科学基金CAREER支持，是IEEE计算机学会优秀访问人员计划中的杰出演讲者，是IBM PERCS组DARPA高吞吐率计算机系统计划项目组的成员，佐治亚理工学院sony-Toshiba-IBM Cell处理器中心的主任。

他还是Internet2研究顾问委员会的成员，是IPDPS和HiPC会议指导委员会的成员，作为主席组织了很多高性能计算以及计算科学与工程方面的会议。

Bader博士是许多高级重要出版物的编辑，包括IEEE Transactions on Parallel and Distributed Systems，ACM Journal of Experimental Algorithmics，IEEE DSOonline，以及Parallel Computing。

他是IEEE计算机协会的资深会员，ACM会员。

Bader博士在用高性能计算解决生物信息学以及计算基因问题方面是取得前沿性研究成果的科学家。

他作为联合主席主持了高性能计算生物学IEEE International Workshop on High-Performance Computational Biology（HiCOMB）的系列会议，曾为几本书撰写了部分章节，联合编辑Journal of Parallel and Distributed Computing（JPDC）和IEEE Transactions on Parallel and Distributed Systems（TPDS）在高性能计算生物学方面研究的专刊。

他还是80余篇被同行审阅过的期刊论文、会议论文的作者，他的主要研究领域是并行算法，组合优化，计算生物学与计算基因学。

## &lt;&lt;面向千万亿次计算的算法与应用&gt;&gt;

## 书籍目录

第1章 千万亿次计算科学应用的性能特征 1.1 介绍 1.2 测试的各种体系结构 1.3 科学应用概述 1.4 GTC : Particle-in-Cell磁融解 1.5 ELBM3D : 晶格玻耳兹曼流体动力学 1.6 Cactus : 通用的相对天体物理学 1.7 PARATEC : 材料科学的第一原理 1.8 HyperCLaw : 双曲AMR气体动力学 1.9 总结与结论 1.10 致谢 参考文献第2章 千万亿次的计算对NASA未来使命的影响 2.1 介绍 2.2 Columbia超级计算机 2.3 航空宇宙分析及计算 2.3.1 方法论 2.3.2 结果 2.3.3 NASA使用千万亿次计算的好处 2.4 推进子系统分析 2.4.1 方法 2.4.2 结果 2.4.3 千万亿次计算给NASA带来的益处 2.5 飓风预测 2.5.1 方法 2.5.2 结果 2.5.3 千万亿计算对NASA的益处 2.6 瓶颈 2.7 总结 参考文献第3章 多物理模拟与千万亿次计算 3.1 引言 3.2 下一代超级计算机 3.3 适用于大规模并行机的编程模型 3.3.1 新型并行语言 3.3.2 MPI-2 3.3.3 协作式并行 3.3.4 协作式并行的应用实例 3.4 多尺度算法 3.4.1 并行的多重网格方法 3.4.2 ALE-AMR离散化 3.4.3 离散-连续混合算法 3.5 目前及将来的应用 3.5.1 万亿次仿真的技术现状 3.5.2 通过协作并行进行多物理模拟 3.6 未来展望 3.7 致谢 参考文献第4章 针对Uintah多物理程序代码的可扩展并行AMR算法研究 4.1 前言 4.2 自适应网格优化 4.3 Uintah程序框架 4.3.1 仿真组件 4.3.2 负载均衡器 4.3.3 调度器 4.4 网格重构器 4.5 提高性能 4.6 将来的工作 4.7 致谢 参考文献第5章 使用Enzo对宇宙进化进行仿真 5.1 宇宙结构的形成 5.2 Enzo的编码 5.2.1 物理层建模和数值算法 5.2.2 自适应网格细化 5.2.3 实现 5.2.4 并行化 5.2.5 快速的邻居网格搜索 5.2.6 Enzo的I/O 5.3 在万亿次平台上的性能和可扩展性 5.3.1 单网格应用 5.3.2 AMR应用 5.3.3 并行展 5.4 将Enzo运行在万亿次计算机平台上 5.4.1 新的AMR数据结构 5.4.2 混合型并行 5.4.3 天体运动和宇宙射线之间的隐性关联 5.4.4 内部数据关系分析工具 5.5 致谢 参考文献第6章 重大影响天气现象数值预测：千万亿次计算的重要动力 6.1 引言 6.2 计算方法和工具 6.2.1 区域性天气预测模型 6.2.2 千万亿系统中的内存和性能问题 6.2.3 分布式内存并行和消息传递 6.2.4 负载均衡 6.2.5 时间消耗和可扩展性 6.2.6 NWP系统中其他重要的组件 6.2.7 其他问题 6.3 NWP实际应用例子 6.3.1 大规模的天气预报 6.3.2 高分辨率的龙卷风仿真 6.3.3 通过观测现象对龙卷风进行预测 6.4 数值天气预测的挑战和需求 6.5 总结 6.6 致谢 参考文献第7章 千万亿次气象科学应用的软件设计 7.1 介绍 7.2 气象科学 7.3 千万亿次计算机的体系结构 7.4 区域气象系统模型CCSM(Community Climate System Model) 7.4.1 当前CCSM概述 7.4.2 区域大气模型CAM(Community Atmosphere Model) 7.4.3 并行海洋程序POP(Parallel Ocean Program) 7.4.4 区域陆地模型 7.4.5 社区海洋冰川模型 7.4.6 模型的耦合 7.5 总结 7.6 致谢 参考文献第8章 迈向分布式千万亿次计算 8.1 引言 8.2 网格计算 8.3 基于网格的千万亿次计算 8.4 虚拟银河 8.4.1 银河的多物理学模型 8.4.2 银河仿真的性能模型 8.4.3 千万亿次虚拟银河仿真 8.5 讨论与总结 参考文献第9章 千万亿次计算时代的生物分子建模 9.1 引言 9.2 NAMD的设计 9.2.1 混合分解 9.2.2 动态负载均衡 9.3 面对千万亿次的挑战与所需的改进 9.3.1 目前的性能 9.3.2 在未来千万亿次机器上的性能 9.3.3 协处理器加速 9.4 生物分子应用 9.4.1 水通道蛋白 9.4.2 钾通道 9.4.3 病毒 9.4.4 核糖体 9.4.5 色素体 9.4.6 BAR域囊泡 9.5 总结 9.6 致谢 参考文献第10章 用于分子动力学模拟的千万亿次计算机 10.1 介绍 10.2 MDGRAPE-3的硬件 10.3 MDGRAPE-3进行的计算 10.4 MDGRAPE-3的芯片 10.4.1 力计算流水线 10.4.2 粒子j的内存和控制单元 10.4.3 芯片说明 10.5 系统结构 10.6 MDGRAPE-3的软件 10.7 MDGRAPE-3的性能 10.8 总结和展望 10.9 致谢 参考文献第11章 在千万亿次超级计算机上进行生物分子仿真 11.1 引言 11.2 机遇 11.2.1 研究更大生物分子系统的能力 11.2.2 研究更长时间范围的能力 11.2.3 混合量子与经典仿真 11.2.4 更精确的仿真 11.3 挑战 11.3.1 在大于100K数量的处理器上扩大生物分子模拟代码的规模 11.3.2 适应硬件的变化 11.3.3 容错性 11.3.4 包含可配置计算的多范型硬件 11.3.5 千万亿次计算带来的新的仿真方法 11.4 总结和展望 11.5 致谢 参考文献第12章 处理大规模图的多线程算法 12.1 引言 12.1.1 图运算中的问题 12.1.2 分布式存储图运算的扩展局限性 12.2 Cray MTA-2平台 12.2.1 并行性表示 12.2.2 对细粒度同步的支持 12.3 案例分析：最短路径算法 12.3.1 初步分析 12.3.2 -分步算法 12.3.3 Thorup算法 12.3.4 实验结果 12.4 案例分析：连通分量 12.4.1 传统PRAM算法 12.4.2 Kahan的多层次算法 12.4.3 性能比较 12.5 结论 12.6 致谢 参考文献第13章 千万亿次计算中的灾难恢复算法研究 13.1 FT-MPI：一个实现容错功能的MPI 13.1.1 FT-MPI概述 13.1.2 FT-MPI：一个实现容错功能的MPI 13.1.3 FT-MPI的使用 13.2 应用级的无盘检查点技术 13.2.1

## &lt;&lt;面向千万亿次计算的算法与应用&gt;&gt;

基于邻居的检查点方案 13.2.2 基于校验和的检查点方案 13.2.3 基于加权校验和的检查点方案 13.3 一种容错的递归方程求解器 13.3.1 有条件的共轭梯度算法 13.3.2 将容错机制添加到PCG算法中 13.4 实验评估 13.4.1 使用不同MPI实现的PCG算法的性能 13.4.2 设置检查点的性能开销 13.4.3 执行恢复操作的性能开销 13.4.4 恢复操作中的舍入错误所带来的数值影响 13.5 讨论 13.6 结论和未来工作 参考文献第14章 Tsubame的研制与未来发展 14.1 引言-2通向Tsubame之路 14.2 Tsubame的架构需求 14.3 Tsubame一瞥 14.4 Tsubame之旅——使世人皆能超级计算的性能和操作 14.5 结论和展望—Tsubame 2.0 参考文献第15章 通过SMP模块构造千万亿次的性能 15.1 引言 15.2 OpenMP编程体系结构 15.3 通过OpenMP实现的循环级并行 15.4 C++与OpenMP 15.4.1 迭代循环 15.4.2 ccNUMA的关键问题 15.4.3 并行化面向对象代码 15.4.4 线程安全性 15.5 应用OpenMP实现嵌套并行化 15.5.1 目前OpenMP规范中的嵌套并行化 15.5.2 FIRE的基于目录图像修复 15.5.3 多块CFD数据集中3D关键点的计算 15.5.4 TFS流体求解器 15.6 结论与展望 参考文献第16章 千万亿次系统的性能及其复杂性分析 16.1 引言 16.2 千万亿次系统体系结构的发展趋势及其并发度 16.3 性能特征和基准测试的现状 16.3.1 基准测试创新 16.3.2 应用程序的性能特征 16.3.3 性能复杂性和性能效能的测量 16.4 APEX-MAIP 16.4.1 APEX-Map的设计原则 16.4.2 并行编程范式与APEX-Map的对比 16.5 性能复杂性特征描述 16.5.1 性能复杂度定义 16.5.2 性能模型选择 16.5.3 若干并行系统的性能复杂性分析 16.6 小结 参考文献第17章 高度可扩展的性能分析工具 17.1 引言 17.2 性能分析概念回顾 17.3 Paradyne 17.4 SCAI/ASCA 17.5 Vampir Next Generation 17.6 Periscope 17.6.1 体系结构 17.6.2 ASL性能属性描述 17.6.3 Periscope节点代理 17.6.4 性能属性搜索 17.6.5 Periscope高层代理 17.6.6 代理通信基础构造 17.6.7 评价 17.7 工具对比和未来研究 参考文献第18章 面向千万亿次计算规模的多级有限元求解器 18.1 引言 18.1.1 概述 18.1.2 千万亿次架构示例 18.2 设计范例 18.2.1 分层混合格网 18.2.2 ParExPDE 18.3 评估与比较 18.4 结论 参考文献第19章 高效有限元代码开发的混合方法 19.1 简介 19.2 高级应用代码 19.3 代码生成 19.3.1 元编程 19.3.2 变分问题的实时编译 19.3.3 FFC 19.3.4 SyFi 19.4 有限元集成的统一框架 19.4.1 有限元集成 19.4.2 UFC接口 19.4.3 实现UFC接口 19.5 总结 19.6 致谢 参考文献第20章 使用Charm++编写千万亿次应用程序 20.1 动机 20.2 Charm++和AMPI: 编程模型 20.2.1 动态负载均衡 20.2.2 投影 20.2.3 其他特性概述 20.3 Charm++应用程序 20.3.1 NAMD 20.3.2 LeanCP 20.3.3 ChaNGa 20.3.4 其他应用 20.4 大型系统仿真 20.5 新型并行语言 20.6 总结 20.7 致谢 参考文献第21章 基于注解的高产出率和性能移植性 21.1 引言 21.2 实现 21.2.1 总体设计 21.2.2 注解语法 21.2.3 系统扩展 21.2.4 代码生成模块 21.3 性能研究 21.3.1 STREAM基准测试 21.3.2 AXPY操作 21.4 相关工作 21.4.1 自调节的库和代码 21.4.2 编译器方法 21.4.3 性能相关的用户注解 21.5 总结与未来的方向 21.6 致谢 参考文献第22章 高效能编程语言的局部性感知特性 22.1 引言 22.2 Chapel中关于数据并行化的基本概念 22.2.1 域 22.2.2 数组 22.3 数据分布 22.3.1 基本方法 22.3.2 “分布”接口 22.3.3 局部存储对象上的分配策略 22.4 实例与讨论 22.4.1 一个负载平衡块分布 22.4.2 一个稀疏数据分布 22.5 实现 22.5.1 编译器实现进展 22.5.2 分布实现策略 22.6 相关工作 22.7 结论和未来展望 22.8 致谢 参考文献第23章 体系结构与程序设计方法对获得持续千万亿次计算性能的影响 23.1 引言 23.2 数值计算和计算机发展的历史简介 23.2.1 20世纪60年代 23.2.2 20世纪70年代 23.2.3 20世纪80年代 23.2.4 20世纪90年代 23.2.5 2000年及以后 23.3 体系结构 23.3.1 处理器发展 23.3.2 Cell Broad Engine处理器 23.3.3 ClearSpeed卡 23.3.4 类向量体系结构 23.3.5 能耗和成本因素 23.3.6 通信网络 23.3.7 通信协议和并行范式 23.4 超大型计算机的算法 23.4.1 大规模计算机 23.4.2 Linpack局限性 23.4.3 选择算法实现方法 23.5 其他技术的影响 参考文献第24章 Cactus框架: 从黑洞到伽玛射线脉冲 24.1 相对天体物理学目前的挑战和伽玛射线脉冲问题 24.2 Cactus框架 24.3 时空代码和流体动力学代码 24.3.1 Cctie: 时空进化 24.3.2 Whisky: 广义相对论流体动力学 24.4 并行化的实现和格网细化 24.4.1 PUGH 24.4.2 使用Carpet实现自适应格网细化 24.4.3 I/O 24.5 当前机器的扩展性 24.5.1 浮点性能 24.5.2 I/O性能 24.6 在千万亿次计算上的发展 24.6.1 物理学: 辐射传输 24.6.2 扩展性 24.6.3 工具 24.7 致谢 参考文献

## 章节摘录

1.5 ELBM3D：晶格玻耳兹曼流体力学LBM：(Lattice-Boltzmann methods)晶格玻耳兹曼方法已经证明是传统数值方法的一种很好的替代，也可以用于模拟流体以及对流动物理学进行建模[29]。其基本的想法就是开发一个简单的动力学模型，结合内在的物理过程，可以重新产生正确的宏观平均特性。

从19世纪80年代中期以来，这些算法得到了广泛的应用，用于模拟Navier-Stokes流，最近扩展到可以处理多项流、反作用流、扩散过程和磁发电机流体力学。

正像从显式算法中可以得到的那样，当一个算法推向更高的雷诺数的时候，LBM趋向于数值非线性不稳态。

这些数值不稳态会上升，因为没有强加的约束会促使分布函数保持非负。

熵LBM算法可以保持分布函数的非负性，即使在任意小的传输系数约束的情况下也是这样，该算法已被用于Navier-Stokes湍流[2]，而且已经被结合到了最近开发的代码之中[30]。

LBM方式使得很困难的边界几何（比如通过使用边界返回的策略来模拟没有滑动墙的情况）的实现变得比较容易。

这里我们介绍周期边界条件下的三维模拟情况，其空间格网和相空间向量格子相互重叠在一起。

每一个格子点和一些mesoscopic变量关联在一起，其值与流方向数量成比例，并且存储在向量中。

该格子被划分到一个三维笛卡儿处理器网状拓扑结构上，用MPI来实现通信，图1-3(a)展示的是一个通信拓扑的情况，重点强调了相对稀疏的通信模式。

对这种情况的大多数模拟，都是通过扩展边界区域空间来存储从邻居处理器获得的数据拷贝。

对于ELBM3D，为了使得碰撞过程满足一定的约束条件，需要在每一个迭代步对每一个格网点求解一个非线性方程。

由于这一方程需要对分布函数的各个组成部分求对数，因此整个算法的性能就特别受限于 $\log(\ )$ 函数的性能。

图1-3(a)展示的ELBM3D的连接关系，在结构上与Cactus非常接近，但是由于代码周期性边界条件的不同，其通信模式上就有细微的差异。

从同构性的角度看，该拓扑与格网或者环互连拓扑有很大不同，但是对于胖树或者互连开关这样的全互连网络，它对可得的对分带宽的占用非常有限。

图1.3(c)表明，和Cactus一样，ELBM3D主要的通信形式是点到点通信，但是图1.3(b)表明点到点消息缓冲区的空间很大，因此它受带宽的限制很大。

## <<面向千万亿次计算的算法与应用>>

### 编辑推荐

对于高性能计算来说，这是一个激动人心的时代，这个时代将会在一个新的规模和尺度上产生大量史无前例的新发现，这些发现将会对科学和社会带来切实的好处。

《面向千万亿次计算的算法与应用》可以让你对千万亿次应用和算法前沿性挑战性研究工作的第一次浪潮有一个概观，为目前和未来的千万亿次研究提供非常必要的基础。



版权说明

本站所提供下载的PDF图书仅提供预览和简介，请支持正版图书。

更多资源请访问:<http://www.tushu007.com>