

《大规模并行处理器编程实战》

图书基本信息

书名：《大规模并行处理器编程实战》

13位ISBN编号：9787302342724

10位ISBN编号：7302342725

出版时间：2013-11

出版社：清华大学出版社

作者：David B. Kirk, Wen-mei W. Hwu

页数：432

译者：赵开勇, 汪朝辉, 程亦超

版权说明：本站所提供下载的PDF图书仅提供预览和简介以及在线试读，请支持正版图书。

更多资源请访问：www.tushu000.com

《大规模并行处理器编程实战》

作者简介

David B. Kirk院士：美国国家工程院院士，NVIDIA院士、前首席科学家，也是CUDA技术的创始人之一，2002年曾荣获ACM SIGGRAPH计算机图形成就奖。他拥有麻省理工学院的机械工程学士学位和硕士学位，加州理工学院的计算机科学博士学位。Kirk是50项与图形芯片设计相关的专利和专利申请的发明者，发表了50多篇关于图形处理技术的论文，是可视化计算技术方面的权威。

Wen-mei W. Hwu(胡文美)教授：拥有美国加州大学伯克利分校计算机科学博士学位，担任美国伊利诺伊大学厄巴纳-香槟分校(UIUC)协调科学实验室电气与计算机工程AMD创始人Jerry Sanders讲席教授(Walter J. Sanders Advanced Micro Devices Endowed Chair)。胡文美教授还是IEEE(国际电气电子工程师学会)院士，ACM(美国计算机学会)院士。

书籍目录

第1章 引言 1

- 1.1 异构并行计算 2
- 1.2 现代GPU的体系结构 6
- 1.3 为什么需要更高的速度和并行化 8
- 1.4 应用程序的加速 9
- 1.5 并行编程语言和模型 11
- 1.6 本书的总体目标 12
- 1.7 本书的组织结构 13

参考文献 16

第2章 GPU计算的发展历程 19

- 2.1 图形流水线的发展 19
 - 2.1.1 固定功能的图形流水线时代 20
 - 2.1.2 可编程实时图形流水线的发展 23
 - 2.1.3 图形与计算结合的处理器 25
- 2.2 GPGPU：一个中间步骤 27
- 2.3 GPU计算 28
 - 2.3.1 可扩展的GPU 29
 - 2.3.2 发展近况 29
 - 2.3.3 未来发展趋势 30

参考文献与课外阅读 30

第3章 CUDA简介 35

- 3.1 数据并行性 36
- 3.2 CUDA的程序结构 37
- 3.3 向量加法kernel函数 39
- 3.4 设备全局存储器与数据传输 41
- 3.5 kernel函数与线程 46
- 3.6 小结 50
 - 3.6.1 函数声明 50
 - 3.6.2 启动kernel函数 50
 - 3.6.3 预定义变量 51
 - 3.6.4 运行时API 51
- 3.7 习题 51

参考文献 53

第4章 数据并行执行模型 55

- 4.1 CUDA的线程组织 56
- 4.2 线程与多维数据的映射 59
- 4.3 矩阵乘法——一个更加复杂的kernel函数 65
- 4.4 线程同步和透明的可扩展性 70
- 4.5 线程块的资源分配 73
- 4.6 查询设备属性 74
- 4.7 线程调度和容许时延 75
- 4.8 小结 78
- 4.9 习题 79

第5章 CUDA存储器 81

- 5.1 存储器访问效率的重要性 82
- 5.2 CUDA设备存储器的类型 83
- 5.3 减少全局存储器流量的一种策略 89

5.4 分块矩阵乘法的kernel函数	93
5.5 存储器——限制并行性的一个因素	98
5.6 小结	100
5.7 习题	101
第6章 性能优化	103
6.1 WARP和线程执行	104
6.2 全局存储器的带宽	111
6.3 执行资源的动态划分	118
6.4 指令混合和线程粒度	120
6.5 小结	121
6.6 习题	121
参考文献	124
第7章 浮点运算	127
7.1 浮点格式	128
7.1.1 M的规范化表示	128
7.1.2 E的余码表示	129
7.2 能表示的数	130
7.3 特殊的位模式与IEEE格式中的精度	134
7.4 算术运算的准确度和舍入	135
7.5 算法的优化	136
7.6 数值稳定性	137
7.7 小结	141
7.8 习题	141
参考文献	142
第8章 并行模式：卷积	143
8.1 背景	144
8.2 一个基本算法：一维并行卷积	148
8.3 常数存储器和高速缓存	149
8.4 使用光环元素的分块一维卷积	153
8.5 一个更简单的分块一维卷积——通用高速缓存	158
8.6 小结	160
8.7 习题	161
第9章 并行模式：前缀和	163
9.1 背景	164
9.2 简单并行扫描	165
9.3 考虑工作效率	169
9.4 工作高效的并行扫描	170
9.5 任意输入长度的并行扫描	174
9.6 小结	177
9.7 习题	177
参考文献	178
第10章 并行模式：稀疏矩阵-向量乘法	179
10.1 背景	180
10.2 使用CSR格式的并行SpMV	183
10.3 填充与转置	184
10.4 用混合方法来控制填充	186
10.5 通过排序和划分来规则化	189
10.6 小结	191
10.7 习题	191

参考文献 192

第11章 应用案例研究：高级MRI重构 193

11.1 应用背景 194

11.2 迭代重构 197

11.3 计算FHD 198

11.4 最终评估 214

11.5 习题 217

参考文献 218

第12章 应用案例研究：分子可视化和分析 219

12.1 应用背景 220

12.2 kernel函数简单的实现方案 221

12.3 线程粒度调节 225

12.4 存储器合并 227

12.5 小结 230

12.6 习题 231

参考文献 232

第13章 并行编程和计算思想 233

13.1 并行计算的目标 234

13.2 问题分解 235

13.3 算法选择 238

13.4 计算思想 243

13.5 小结 244

13.6 习题 244

参考文献 244

第14章 OpenCL简介 245

14.1 背景 246

14.2 数据并行性模型 247

14.3 设备的体系结构 249

14.4 kernel函数 250

14.5 设备管理和启动kernel 251

14.6 OpenCL中的静电势图谱 254

14.7 小结 258

14.8 习题 258

参考文献 259

第15章 OpenACC并行编程 261

15.1 OpenACC与CUDA C的比较 261

15.2 执行模型 263

15.3 存储器模型 265

15.4 基本的OpenACC程序 266

15.4.1 并行构造 266

15.4.2 循环构造 267

15.4.3 kernels构造 272

15.4.4 数据管理 275

15.4.5 数据构造 276

15.4.6 异步计算和数据传输 278

15.5 OpenACC的发展方向 279

15.6 习题 280

第16章 Thrust：一个面向效率的CUDA编程库 281

16.1 背景简介 282

- 16.2 动机 284
- 16.3 Thrust的基本特性 284
 - 16.3.1 迭代器和内存空间 286
 - 16.3.2 互操作性 286
- 16.4 泛型编程 288
- 16.5 抽象的益处 290
 - 16.5.1 编程效率 290
 - 16.5.2 鲁棒性 291
 - 16.5.3 真实性能 291
- 16.6 最佳范例 293
 - 16.6.1 融合 293
 - 16.6.2 数组结构体 294
 - 16.6.3 隐式范围 296
- 16.7 习题 297
- 参考文献 298
- 第17章 CUDA FORTRAN 299
 - 17.1 CUDA FORTRAN和CUDA C的区别 300
 - 17.2 第一个CUDA FORTRAN程序 301
 - 17.3 CUDA FORTRAN中的多维数组 303
 - 17.4 用通用接口重载主机/设备端例程 304
 - 17.5 通过iso_c_binding调用CUDA C 307
 - 17.6 kernel循环指令和归约操作 309
 - 17.7 动态共享存储器 310
 - 17.8 异步数据传输 311
 - 17.9 编译和性能剖析 316
 - 17.10 在CUDA FORTRAN中调用Thrust 317
 - 17.11 习题 321
- 第18章 C++ AMP简介 323
 - 18.1 C++ AMP核心特性 324
 - 18.2 C++ AMP执行模式详解 329
 - 18.2.1 显式和隐式的数据复制 330
 - 18.2.2 异步操作 331
 - 18.2.3 本节小结 333
 - 18.3 加速器管理 333
 - 18.4 分块执行 335
 - 18.5 C++ AMP图形特性 338
 - 18.6 小结 340
 - 18.7 习题 341
- 第19章 异构集群编程 343
 - 19.1 背景简介 344
 - 19.2 运行示例 344
 - 19.3 MPI基础 346
 - 19.4 MPI点对点通信模型 348
 - 19.5 重叠计算和通信 355
 - 19.6 MPI集合通信模型 362
 - 19.7 小结 363
 - 19.8 习题 363
 - 参考文献 364
- 第20章 CUDA动态并行 365

- 20.1 背景 366
- 20.2 动态并行简介 367
- 20.3 重要细节 368
 - 20.3.1 启动环境变量设置 369
 - 20.3.2 API错误和启动失败 369
 - 20.3.3 事件 369
 - 20.3.4 流 369
 - 20.3.5 同步范围 370
- 20.4 内存可见性 371
 - 20.4.1 全局内存 371
 - 20.4.2 零拷贝内存 371
 - 20.4.3 常量内存 371
 - 20.4.4 局部内存 371
 - 20.4.5 共享内存 372
 - 20.4.6 纹理内存 372
- 20.5 一个简单示例 373
- 20.6 运行时限制 376
 - 20.6.1 内存占用 376
 - 20.6.2 嵌套深度 376
 - 20.6.3 内存分配和生存周期 376
 - 20.6.4 ECC错误 377
 - 20.6.5 流 377
 - 20.6.6 事件 377
 - 20.6.7 启动池 377
- 20.7 一个更复杂的示例 378
 - 20.7.1 线性贝塞尔曲线 378
 - 20.7.2 二次贝塞尔曲线 378
 - 20.7.3 贝塞尔曲线计算(非动态并行版本) 378
 - 20.7.4 贝塞尔曲线计算(使用动态并行) 381
- 20.8 小结 384
- 参考文献 384
- 第21章 结论与展望 385
 - 21.1 重点回顾 385
 - 21.2 存储器模型的演变 386
 - 21.2.1 大型虚拟和物理地址空间 386
 - 21.2.2 统一的设备存储空间 388
 - 21.2.3 可配置的缓存和暂时存储器 388
 - 21.2.4 提高原子操作的速度 389
 - 21.2.5 提高全局内存的访问速度 389
 - 21.3 kernel函数执行控制过程的演变 389
 - 21.3.1 kernel函数内部的函数调用 389
 - 21.3.2 kernel函数中的异常处理 390
 - 21.3.3 多个kernel函数的同步执行 390
 - 21.3.4 可中断的kernel函数 391
 - 21.4 内核的性能 391
 - 21.4.1 双精度的速度 391
 - 21.4.2 更好的控制流效率 392
 - 21.5 编程环境 392
 - 21.6 美好前景 392

《大规模并行处理器编程实战》

参考文献 393

附录A 矩阵乘法主机版的源代码 395

附录B GPU的计算能力 407

《大规模并行处理器编程实战》

精彩短评

- 1、作者在Coursera上开了门课 Heterogeneous Parallel Programming <https://class.coursera.org/hetero-002#>
- 2、理论高于实战！建议读完GPU高性能编程CUDA实战，再来啃这本

《大规模并行处理器编程实战》

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:www.tushu000.com