

华中科技大学出版社计算机丛书

多核计算与程序设计

Multi-core Computing and Programming

○ 周伟明 著



华中科技大学出版社
<http://www.hustp.com>

策划编辑：王红梅
责任编辑：王红梅
封面设计：秦 茹



本书特色内容：

- 并行遍历的基本方法。
- 常见并行算法如并行搜索、并行排序、并行数值计算等在多核系统中的实现。
- 共享资源分布式计算的基本编程模式和方法。
- 分布式队列，它能自动给每个线程赋予一个本地队列，它是基于偷取的共享队列和队列池来实现的。
- 分布式查找，包括分段锁的哈希表，动态负载平衡的多级查找等。
- 分布式内存管理，它自动给每个线程生成一个本地的内存管理器，并且几乎不需要使用锁进行内存分配和释放（抢夺式内存管理）。
- 任务图分解与调度及实现方法。
- 非嵌套任务调度，可用于网络服务器软件的任务调度。
- 嵌套任务调度，是另一种更广泛的任务调度方法，可用于实现各种并行计算。
- 各种程序和算法中的伪共享问题的处理。
- Lock-Free编程基础知识。

ISBN 978-7-5609-5096-9



9 787560 950969 >

定价：88.00元

华中科技大学出版社计算机丛书

多核计算与程序设计

Multi-core Computing and Programming

◦ 周伟明 著



华中科技大学出版社
(中国·武汉)

图书在版编目(CIP)数据

多核计算与程序设计/周伟明 著. —武汉:华中科技大学出版社,2009年3月
ISBN 978-7-5609-5096-9

I. 多… II. 周… III. 并行程序-程序设计 IV. TP311.1

中国版本图书馆 CIP 数据核字(2009)第 004442 号

多核计算与程序设计

周伟明 著

策划编辑:王红梅

责任编辑:王红梅

责任校对:周 娟

封面设计:秦 茹

责任监印:熊庆玉

出版发行:华中科技大学出版社(中国·武汉)

武昌喻家山 邮编:430074 电话:(027)87557437

录 排:武汉众心图文激光照排中心

印 刷:湖北新华印务有限公司

开本:787mm×1092mm 1/16

印张:43 插页:2

字数:900 000

版次:2009年3月第1版

印次:2009年3月第1次印刷

定价:88.00元

ISBN 978-7-5609-5096-9/TP·670

(本书若有印装质量问题,请向出版社发行部调换)

内容简介

本书主要介绍适应于多核（或多处理器）计算机系统的算法和程序，共分为五个部分。

第1部分介绍多核编程的基础知识，包括多核编程常见问题、锁竞争、加速比、负载均衡等基本概念，多线程退出算法、读写锁、旋转锁、原子操作等多线程编程基础知识，基于 OpenMP 标准的并行程序设计基础等。

第2部分介绍基础的数据结构与算法，包括数组、链表、哈希表、二叉树、AVL 树、复合二叉树等基本数据结构，还讲解了多线程并行遍历的基本方法。

第3部分介绍多核并行计算方面的基础知识，并行编程包括常用的编程模式，如分治模式、流水线模式、任务图分解与调度模式、动态任务调度模式等，并行搜索包括顺序搜索及终止检测算法、并行最短路径搜索等，并行排序包括并行快速排序、并行归并排序、并行基数排序等，并行数值计算包括并行矩阵乘法、并行前缀和计算等方面的内容。本部分介绍的各种并行算法和程序中，重点介绍如何解决多核系统中的计算随 CPU 核数的扩展性，CPU Cache 伪共享方面的问题。

第4部分介绍多核共享资源计算方面的内容，也是本书中最重要的内容，讲解了分布式计算设计模式如线程分组竞争模式、条件同步模式、批量私有化处理模式、数据本地化模式等。本部分中讲解了本书中几个最重要的程序：分布式队列中实现了自动让每个线程带有一个本地队列、分布式查找中介绍了分段锁的哈希表、动态负载平衡的分布式查找等，分布式内存管理则介绍了适应多核的内存管理方案，尤其是基于抢夺式的分布式内存管理算法，在分配和释放共享内存时也几乎不需要使用锁，性能优异。

第5部分介绍任务分解与调度方面的知识，这也是本书中最重要的内容，包括任务图分解与调度的实现方法，动态任务分解与调度的实现方法等。其中，还介绍了使用动态嵌套任务调度进行并行计算的方法，给出了用动态嵌套任务调度实现 Parallel_For ()、并行快速排序、并行归并的实例。

最后一章介绍了 Lock-Free 编程（使用 CAS 原子操作进行编程）的基础知识，如 ABA 问题，内存删除问题等，并给出了一个 Lock-Free 的队列的实现实例。

本书的特色内容如下。

- (1) 并行遍历的基本方法。
- (2) 常见并行算法如并行搜索、并行排序、并行数值计算等在多核系统中的实现。
- (3) 共享资源分布式计算的基本编程模式和方法。
- (4) 分布式队列，它能自动给每个线程赋予一个本地队列，它是基于偷取的共享队列和队列池来实现的。
- (5) 分布式查找，包括分段锁的哈希表，动态负载平衡的多级查找等。
- (6) 分布式内存管理，它自动给每个线程生成一个本地的内存管理器，并且几乎不需要使用锁进行内存分配和释放（抢夺式内存管理）。
- (7) 任务图分解与调度及实现方法。
- (8) 非嵌套任务调度，可用于网络服务器软件等地方进行任务调度。
- (9) 嵌套任务调度，是另一种更广泛的任务调度方法，可以用它实现各种并行计算。
- (10) 各种程序和算法中的伪共享问题的处理。
- (11) Lock-Free 编程基础知识。

序

我的工作要求我跟踪、了解软件技术各领域的重要进展。为此，除了频繁阅读最新的技术新闻，请教于各路高人，我还必须经常留心市面上的技术图书。

大约是在 2006 年 5 月，我在中关村图书大厦看到一本《多任务下的数据结构与算法》，作者周伟明，这本书当即引起了我的兴趣。

自从 2005 年初 Herb Sutter 发表名为《免费午餐已经结束》的著名文章以来，多核计算就一直是整个技术社群关注的热点问题之一。不过，大家的注意力更多地集中在诸如 Java. concurrent、MPI、OpenMP、Intel TBB 等程序库和 Pipelining、MapReduce 等编程模式上，希望能够借助一些工具来规避并发程序设计所带来的智力挑战。毫无疑问，由于并发程序所具有的复杂性和各种“诡异”问题，在软件项目实施中，应尽可能利用现成的模式和工具，而要对“土法炼钢”式的“创新”保持审慎。然而，这并不意味着开发者可以轻轻地“站在巨人肩膀上”；恰恰相反，如果没有自下而上地对并发多任务程序设计下一番苦功进行研究的话，这些模式和工具在我们手中反而可能成为无知和错误的放大器，贻害无穷。《多任务下的数据结构与算法》这本书的出发点，即是以并发和多任务的角度重新审视算法与数据结构这样最基本、最重要的知识，从根本上帮助读者建立对多任务程序本质的正确观念、知识体系和关键技能，对此我是非常认可的。

特别是看到书中有一段对微软 COM 体系的批评，言人所不敢言，透着一种自信磊落、敢想敢做的气质，绝无一般技术图书那种四平八稳的八股气，更觉得作者绝非普

通技术专家，一定是有心思、有见地的高人。当夜，我就在自己的 CSDN 博客上发表了一篇题为《一位作者对 COM 的严厉批判》的文章，对作者在技术上的求真务实表示钦佩。

由于这篇博文的缘故，我与这位思想锐利、技术精湛、职业生涯丰富奇特的专家成为朋友，也更加深了对他的了解。

周先生在技术上有自己的坚定信念，让万法纷繁的各种编程技术直趋软件技术的本质——算法。多年来，他曾服务于著名大型企业，也曾在创业型公司和硅谷高科技企业中颇有建树，职业生涯历经兴衰，然而却能心存一念而不堕，苦心孤诣于算法研究，功力自然非凡。

近二三年里，周先生专心钻研，不求闻达，为了寻找一片安静的研究和写作环境，他北上南下，几易居所；先是对软件测试进行了系统的研究并出版相关著作，随后又用一年的时间系统梳理了多任务数据结构及算法的有关知识，并取得若干重要创新成果。为了体现并发多任务程序设计方面的最新研究成果，他决定修订前书，并在写作之中与书成之后，与我进行了较为深入的讨论。

我认为《多核计算与程序设计》在保留《多任务下的数据结构与算法》精华的基础上，确有重大提升，值得再次推荐。细言之，这本书的优点有三。

其一为“整”。此书体系完整，内容精当，首先覆盖了专业教育及实际应用开发所需要的主要数据结构和算法技术，然后从并行、分布式角度考察这些数据结构的实际运用。而且门槛低，天花板高；读者按图索骥，很容易入门，认真钻研，则能在算法、数据结构和多任务计算方面达到较高造诣。

其二为“实”。此书系作者心血之作，每一段文字、每一段程序都是作者反复思考、反复优化、反复检验得来的，系经验之谈，绝无鹦鹉学舌、浮夸抄袭之嫌，读者随意翻开一读便知。此外，对比一般数据结构算法类著作，本书还有一个鲜明的“实在”特色：用工业级的技术解决方案作为案例进行讲解，例如讲到数组时介绍 Hook 技术实现方案，讲到哈希表时介绍 Web Server Cache 实现，讲到分布式查找时介绍多级分布式数据结构、并实现了分布式哈希数组，可谓实在之极。真正在一线从事开发工作的读者都能体会到这些内容的价值。此外，此书行文朴实，语言流畅，阅读起来毫无隔阂，体现了作者作为一流技术专家的务实求真。

其三为“新”。此书不但体现了周先生多年来对算法和并发程序设计的钻研经

验，而且总结了他最近两年来的一些最新研究成果，其中不乏领先性的模式和多核算法。2008年12月，作者在CSDN主办的“2008软件开发2.0技术大会”上，将这些创新成果与上百位与会技术专家和开发者进行了交流，受到大家的好评。作者在此书中无保留地公开了这些创新成果，有心的读者可循此进入并发编程技术的前沿领域。

如果说此书第一版出版的时候，多核与并行计算是“大势所趋”，那么今天，多核与并行计算已经不只是“势”，而已经成为现实。因此，并发程序的设计已经成为当代开发者必须了解的领域。

基于以上所述的原因，我愿意再次向读者推荐这本书，相信有大志的读者，定能从此书中获得知识与经验的丰收。

孟 岩

2008年12月

自序

——新的多核免费午餐开始

2005年3月，C++标准委员会主席 Herb Sutter 在《Dr. Dobbs's Journal》上发表了一篇著名的文章《免费午餐已经结束——软件历史性地朝并发靠拢》（*The Free Lunch Is Over, A Fundamental Turn Toward Concurrency in Software*），并预言自面向对象革命以来，并发性将是软件开发中最大的变化。一场由多核 CPU 引起的新的软件开发革命开始了。

从近几年多核软件开发方面的情况来看，确实如 Herb Sutter 所说：免费的午餐已经结束。在多核 CPU 面世后的这几年里，许多软件人员可能发现，以前写的程序拿到多核 CPU 上运行，性能得不到应有的提高，甚至出现个别性能下降的情况。也有许多软件人员在努力地将程序改写成能并发运行的软件，其中的少数在多核 CPU 机器上取得了很好的提高，但是大多数的努力并没有得到好的回报，他们写的并发性软件在多核 CPU 上运行的性能提高并不理想。

为什么会出现这种情况呢？并发性编程的难度高是一方面的原因，更重要的原因是近些年来多核软件技术方面日新月异的发展。尤其是多核共享资源分布式计算方面，最近几年的发展可以说是突飞猛进。换句话说，近些年来多核编程技术方面的变化太快，导致软件人员根本来不及掌握最新的编程技术是最主要原因。

那么，掌握了最新的多核共享资源分布式计算方面的编程技术，会不会还是和以前一样，难以写出性能好的软件呢？（这里说的性能并非单纯指速度，也包括扩展性等其他性能指标。）当然没有人保证一定写出性能好的软件，就像一把宝剑交到一个不会武功的人手里，它的威力是无法发挥的。不过，可以肯定的是，掌握了这些新的编程

技术，至少能比以前有一个很大的提高，也更容易写出具有高并发性的软件，并且能够大大降低并行编程的难度。从这个意义上讲，共享资源分布式计算技术，是端给编写并行程序的程序员的一道新的免费午餐。

这道新的免费午餐的一个初始版本在 2008 年初已经摆上了餐桌，那就是 Intel 公司提供的开源项目 TBB。也许有一些人已经品尝过这道还比较原始的免费午餐，当然味道好坏依据不同人的口味可能会有不同。

值得一提的是，多核的免费午餐仍然以单核时代的各种算法作为基础，并不像我们见过的许多并行算法一样完全另起炉灶；这使得学习和使用它的难度大大降低，并且可以充分复用以往的知识，程序员只需要在原有基础上向前进一步，并不需要完全从头开始学习全新的知识。

“授人以鱼不如授人以渔”。本书不仅是教大家如何品尝免费的午餐，更重要的是教大家如何烹调。了解烹调的方法后，才能更加深入地品尝。因此，本书不是对某个开源项目进行剖析，更不是讲解它的使用方法，事实上目前也没有任何一个开源项目可以做到将整个多核计算覆盖或者很好地实现。本书从多核计算依据的基本原理出发进行讲解，然后到算法实现、程序代码的实现，由浅入深地覆盖了多核计算的大部分内容。当然，其中也有一些重要算法是作者经过不懈努力、进行创新和改进的结果，希望读者在书中细细品尝。

预备知识

如果需要学习本书程序实现的细节，那么在阅读本书前，读者需要掌握 C/C++ 编程语言基础知识，掌握 C++ 模板的基本概念和使用方法，但并不需要掌握复杂的泛型编程语法。

如果只学习本书的算法思想和设计方法，那么并不需要掌握 C/C++ 编程语言方面的知识，有 Java 等类似语言经验者也可以阅读。

如果有多线程编程的实践经验，则对深入掌握和理解本书的内容有很大帮助。

学习本书的第 18 章内容需要少量的图论基础概念，其他章节则可以不需要数据结构算法方面的基础，本书中自带有所需的数据结构与算法基础知识。当然，如果要深入掌握，则数据结构与算法方面的基础越厚越好。

学习本书中加速比分析的内容，应掌握少量概率论方面的知识；需要进行深入掌握的读者，应了解概率论的一些基础知识。其他的涉及数学理论方面的内容，基本上具有高中的数学知识已足够。

本书内容介绍和导读

虽然本书是专门针对多核计算与程序设计而写，然而多核计算仍然需要继承以往单核计算的基础，尤其是多核多线程编程的基础；为了照顾一些算法基础较弱的读者，本书选用了作者所著的《多任务下的数据结构与算法》的许多内容，这部分内容约占本书的篇幅三分之一，另外有三分之二属新增加内容。

本书共分五个部分，其中第2部分基本上是原《多任务下的数据结构与算法》中的内容，其他部分基本上是新内容。各部分内容介绍如下。

第1部分为基础知识，讲解了多核编程的一些基本概念如加速比、锁竞争等，多线程编程的一些基本概念如线程、锁、原子操作等，以及 OpenMP 程序设计等内容。

第2部分为《原多任务下的数据结构算法》中的内容，包括数组、栈、队列、哈希表、普通树、二叉树、AVL 树、复合二叉树等基本数据结构与算法。除 AVL 树那一章添加了内容外，其他的章节都只是在原基础上做了少量的修改，例如对原书中的一些错误进行了更正，对难以理解之处进行重新阐述。

第3部分为并行计算内容，重点讲解针对多核系统的并行算法，如并行排序、并行查找、并行数值计算等。讨论了多核系统中并行计算的常用设计模式，特有的伪共享问题的处理等。

第4部分为共享资源分布式计算，是本书的核心内容之一。虽然很多地方仍然沿用老的叫法把它称作并行 (concurrent) 计算，但实际上叫做分布式计算更确切，因此本书将它叫做分布式计算。这一部分主要讲解多核系统中共享资源分布式计算设计模式，并给出分布式队列，分布式查找，分布式内存管理等数据结构与算法。

第5部分为任务分解与调度，也是本书的核心内容。这一部分主要讲解了三种任务调度算法：任务图分解与调度、非嵌套任务调度、嵌套任务调度，并给出了使用嵌套任务调度进行并行计算的实例。

最后一章中还讲解了无锁 (Lock-Free) 编程的一些基本概念。

如果您是初学者，只有少量的数据结构算法基础，那么按照本书章节顺序依次阅读即可。

如果您是原《多任务下的数据结构算法》的读者，那么第2部分的内容可以跳过不读。其他章节依次阅读，遇到熟悉的内容则跳过去即可。

如果您有丰富的并程序序和算法经验，那么建议您将第 1 部分的内容浏览一下，第 2 部分的链表那章中的并行遍历，复合二叉树等内容浏览一下，然后可以直接阅读第 4、5 部分的内容，再返回去参考一下第 3 部分并行计算的内容。

如果您是并行计算方面的高手，并有多核环境下的丰富实践经验，那么请直接阅读第 4、5 部分的内容即可。

本书代码说明

本书的代码用 C/C++ 写成，有些用 C，但有些用 C++。主要是因为有一些代码来自原《多任务下的数据结构与算法》，这部分代码仍然使用 C 语言的 void 指针方式实现；而后来添加的多核方面的代码，出于设计上的考虑，使用 C++ 模板有更好的扩展性，因此大都改为使用 C++ 模板实现。因此本书混合使用了 C 和 C++ 代码。

本书的代码主要供学习使用，只经过简单的调试和测试，因此不保证它的健壮性。如果要用于商业软件之中，请自行增加测试用例进行详尽的测试。

本书的代码发布在开源项目 CAPI 中，代码下载地址：<http://gforge.osdn.net.cn/projects/capi>。

为了阅读方便，书中可能会删除一些参数校验之类的代码等，因此书中代码和开源项目中代码存在一些不一致的地方。如果要使用，请以开源项目中的代码为准。如果将来发现 bug，会直接在开源项目中修正。

bug 报告：如果发现书中或代码中的 bug，请报告到开源项目 CAPI 中，或作者的博客中，作者博客地址：<http://blog.csdn.net/drzhouweiming>。

关于参考文献的说明

在国内搞过研究的人，尤其是像我这样不在学术界却要自行研究的人，一定知道在国内查阅参考文献的困难之处。为了查阅本书的参考文献，我去过深圳、北京、上海等地，许多图书馆的资料大多不全，并且大部分图书馆的网络速度较慢，有的慢至半个小时下载不了一篇文章，有的慢得根本无法将文章下载下来。

为了写作本书，共查阅了 1700 多篇参考文献，最终收入书中的有 500 余篇。单查阅文献的时间，估计可以折算成三个月左右的全部工作日时间。由于查阅文献的困难，并有许多文献无法下载原文，因此本书在参考文献方面的疏漏之处在所难免，如有遗漏某位前辈、高人、大师的重要文献的，敬请谅解！同时也请指出，以便日后更正。

有志于进一步研究多核计算的读者，可以将本书及所列出的参考文献作为起点，因为这些文献在国内基本上都可以下载原文。也欢迎有志者和作者进行联系和交流。

致谢

自2007年6月开始，作者花费了约17个月的全部工作日时间完成本书的写作，其间得到许多朋友的帮助。其中，比较重要的贡献来自我的大学同学、上海优控科技有限公司董事长王琦先生，这本书的大部分内容都是在他的公司里写作而成的。如果没有他提供的安静的办公场所，这本书也许要推迟很长时间才能和读者见面。

另外一个对本书有重要贡献的朋友是孟岩先生，在如何写作方面孟岩先生给予了非常有益的指导，如果本书在写作方面比作者所著的前两本书能够有所提高的话，那么这个功劳都要归功于孟岩先生。同时，孟岩先生还对本书提出了一些有益的修改意见。

当然，还要感谢许多对《多任务下的数据结构算法》一书报告 bug 和提出建议的网友，有了他们的贡献，才有机会将原书中的 bug 一一修正，使本书的质量得到较大的提升。在此，谨对他们表示诚挚的谢意！同时，希望有更多的网友能对本书提出更多的修改建议。

最后得感谢我的家人，如果没有他们在背后默默的奉献和牺牲，我是不可能专心致志地写作本书的。

当然还有很多朋友对本书有过贡献，限于篇幅，在此不一一列出，敬请谅解。

总 目 录

第 1 部分 基础知识

- 1 多核计算概述 (1)
- 2 多线程编程基础 (41)
- 3 OpenMP 程序设计 (75)

第 2 部分 基础数据结构与算法

- 4 数组 (125)
- 5 链表 (149)
- 6 哈希表 (191)
- 7 普通树与二叉树 (221)
- 8 AVL 搜索树 (245)
- 9 复合二叉树 (287)

第 3 部分 并行计算

- 10 并行程序设计模式 (315)
- 11 并行搜索 (323)
- 12 并行排序 (345)

13 并行数值计算 (379)

第 4 部分 共享资源分布式计算

14 分布式计算设计模式 (391)
15 分布式队列 (405)
16 分布式查找 (453)
17 分布式内存管理 (497)

第 5 部分 任务分解与调度

18 任务图分解与调度 (549)
19 动态任务分解与调度 (585)
20 Lock-Free 编程基础 (617)

附录 1 本书代码和 CAPI 开源项目源文件对照表 (639)

附录 2 多核编程的四层境界 (654)

目 录

第 1 部分 基础知识

1 多核计算概述	(1)
1.1 多核 CPU 概述	(2)
1.1.1 多核计算发展趋势	(2)
1.1.2 多核 CPU 硬件架构介绍	(4)
1.1.3 多核给程序员带来的机遇和挑战	(5)
1.2 多核编程会遇到的问题	(7)
1.2.1 并发性问题	(8)
1.2.2 CPU 饥饿问题	(8)
1.2.3 任务的分解与调度问题	(9)
1.2.4 加速比性能问题	(10)
1.2.5 节能、环保问题	(11)
1.2.6 扩展性问题	(12)
1.3 多核编程与单核多线程编程的区别	(13)
1.3.1 锁竞争导致的串行化的区别	(13)
1.3.2 线程分解与执行的区别	(14)
1.3.3 CPU 核负载平衡的区别	(14)
1.3.4 任务调度策略的区别	(15)
1.3.5 CPU Cache 存取的区别 (伪共享问题)	(16)
1.3.6 任务优先级抢占的区别	(17)
1.3.7 串行计算与并行及分布式计算的区别	(17)
1.4 多核编程与多机分布式编程的区别	(18)