

学术著作丛书

面向结构的并行算法

设计与分析

李晓梅等
编著



面向结构的并行算法 ——设计与分析

李晓梅 莫则尧 编著
文尚猛 窦 勇

国防科技大学出版社

图书在版编目(CIP)数据

面向结构的并行算法——设计与分析/李晓梅等编著. —长沙:国防科技大学出版社,1996.6

ISBN 7-81024-367-5

I 面向结构的并行算法——设计与分析

II 李晓梅、莫则尧、文尚猛、窦勇

III 计算技术-并行算法-面向结构

IV TP301.6

责任编辑:徐 飞 张序君

责任校对:何 晋

封面设计:陆荣斌

国防科技大学出版社出版发行

电话:(0731)4555681 邮政编码:410073

新华书店总店北京发行所经销

湖南大学印刷厂印装

*

开本 850×1168 毫米 1/32 印张:11.375 字数:285 千

1996年6月第1版第1次印刷 印数:1—2000册

*

ISBN 7-81024-367-5
TP·73 定价:20.00元

内 容 简 介

本书系统地介绍了各种面向结构的并行计算模型上的并行算法。首先介绍了当前流行的一些并行计算模型,阐述了它对并行算法设计与分析的重要性,随后介绍了阵列和树模型、树网模型、超立方体网络模型以及超立方体类型网络模型等,同时阐述了在这些模型上一些基本的数值和非数值并行算法的设计与实现。在写作上深入浅出,注意系统化和揭示其思想方法。

本书适合于各大专院校、大型工程与科学计算单位的从事于计算数学、并行计算的研究人员以及从事计算机体系结构、计算机系统软件、应用软件的开发和研制者,还可以作为一本将计算机结构与算法紧密结合的入门教程。

前 言

自从本世纪 70 年代初产生第一台巨型计算机 ILLAC-IV 以来,巨型计算机的发展已经历了四代。第一代单指令流多数据流 (SIMD) 阵列处理机 (简称 AP 机),以美国鲍勒斯公司于 1973 年研制成功的 ILLAC-IV 为代表机型,主要是为天气预报和原子能数据处理而设计的。尽管这一代巨型机没有得到顺利发展,但它起到了开拓性的作用。第二代单处理机内多运算流水线向量机 (简称 VP 机),以美国 CDC 公司 70 年代初研制成功的 STAR-100 和 CRAY 公司 1978 年研制成功的 CRAY-1 向量机为代表机型。我国 1983 年研制成功的 YH-1 也属于这一代巨型机。由于这一代计算机只有一台处理机,且硬件、并行编译、向量识别以及并行算法技术比较成熟,用户使用方便,因而发展很快而形成产业。第三代共享主存的并行多处理机系统 (简称 SPP 机),以美国 CRAY 公司 1983 年研制成功的 CRAY X-MP 系列机为代表。我国 1992 年研制成功的 YH-2,以及中科院计算所研制成功的“曙光-I”也属于这一代巨型机。由于这一代巨型机处理机数目少,且共享主存,加上并行编译、并程序序设计环境、多任务库以及并行算法有相当程度的发展,因此使用方便,效率也比较高。第四代分布主存的大规模并行机 (简称 MPP 机),以美国 Intel 公司的 IPSC/860,思维公司的 CM-2、CM-5, Ncube 公司的 Ncube 10 和 CRAY 公司的 T-3D 为代表机型,我国研制成功的“神州-II”也属于这一类机型。近年来,由于高性能微处理器和各种超大规模集成电路的发展,并行处理技术逐年提高,从而使得这一类并行计算机系统得到迅速发展。但是它们的有效使用还有待于并行软件和并行算法的进一步发展。

并行巨型机的发展,使得大型科学与工程计算,特别是高维复杂问题的计算成为可能,使得科学计算作为科学研究的一种有效手段,已上升为与科学理论和科学实验相并重的三大科学方法之一。近十几年来,国防科技和国民经济建设的许多领域已与并行巨型机紧密结合,共同推动着我国科学技术的不断发展。在核武器研制中,巨型机已成为发展核武器研制的必要手段和主要工具;在航空航天工业和飞行器设计中,计算空气动力学已发展成为与风洞实验、自由飞行一起构成获得飞行器气动力数据的三种手段;在现代高技术战争中,作战计划的制定、情报信息的处理、兵力的指挥部署、作战行动的协调以及后勤的保障都需要大型巨型机给予处理,以便在极短的时间内作出决策;在天气预报和石油工业中,更依赖于并行巨型机的使用。高分辨率的中长期数值天气预报只有在巨型机问世后才变为可能,并由此而推动气象科学的发展;同样石油地震资料处理、油藏数值模拟和测井数据处理都需要巨型机的支持。正是由于这些应用领域的需求,要求巨型机具有更快的速度和更大的信息吞吐量,这又促使当前并行计算机系统向高度并行化方向发展,以满足应用的需求。

科学计算的发展史表明,计算方法的设计与分析是与计算机技术的进步相互影响,相互促进的,是计算技术的一个重要组成部分。对于并行处理和超级计算环境而言,并行算法与并行计算机硬件、软件三者并存,缺一不可,共同构成了超级计算技术。由于向量机、共享存储和分布存储并行计算机系统的产生,它们向各个领域的计算数学工作者提出了挑战,也就是要求他们在设计算法和安排计算时,必须使具体的并行计算机结构的效率得到较好的发挥。原有的一些为手算而设计的最优串行算法也许会变得不尽人意,需要重新改造。而许多在串行机上不是最优的算法,由于其内在的并行性而焕发“青春”。过去十多年,以向量机为典型代表的巨型机的成功应用,已经说明了这一点,使得并行算法在科学计算、工程

设计、数据处理和人工智能等领域产生了显著的社会效益和经济效益。

尽管如此,回顾我们从事并行计算机和并行算法所走过的历程,不难看到仍然存在这么一个事实:计算机科学工作者专门从事研制和设计计算机硬件和系统软件,而应用领域的计算数学工作者根据给定的并行计算机设计并行算法,二者之间缺少必要的勾通。如果对于向量机而言,这一点可以忍受的话,那么,并行计算机发展到今天,尤其是并行编译系统、应用软件系统还不完善的条件下,要使 MPP 并行计算机系统的效率得到较好的发挥,不熟悉具体计算机系统的结构,了解其性能,是万万不行的。实际表明,算法与结构结合的好坏,直接影响到计算机系统的效率发挥。同样的问题,结合得好,可以获得很好的加速比;结合得不好,并行计算机的效率将得不到发挥。这就要求并行算法设计者们在设计算法的过程中,应对其所使用的并行计算机结构有一个明确的认识。也要求计算机科学工作者和各应用领域的计算数学工作者相互之间密切结合,互相渗透。本书恰恰是为了这个目的而编写的。在近一年的讨论和具体教学中,我们发现它在这方面能收到较好的效果,至少可以起到抛砖引玉的作用,以期望在这方面有更好的著作出现。

全书共分六章。第一章介绍并行计算模型和并行算法设计以及一些基本方法。并行计算模型是一类并行计算机共同基本特征的抽象,是并行算法设计与分析的基础。我们介绍本章的目的是让读者明确算法研究要与切合实际的并行计算模型相结合,以保证并行算法的有效性。同时也开拓思维,使我们可以脱离具体并行计算机,而按并行计算模型来设计算法,以保证并行算法适合于广泛的一类并行计算机结构,同时也简化算法设计。第二章介绍阵列和树模型。给出了从一维到高维阵列的基本概念、性质,引出了并行算法的性能评价准则以及在这种最简单的模型上基本算法的实现。第三章介绍树网模型。树网是由阵列和树组成的一种混合结

构,它克服了阵列和树模型的两个主要缺点,具有很小的网络直径和较大的等分宽度。在它上面设计的算法几乎在速度上都达到最优。第四章介绍我们熟知的超立方体网络,并将所有阵列和树网算法在超立方体中等效实现。本章讨论了阵列的树网结构在超立方体网络中的嵌入以及算法的直接映射。第五章介绍超立方体类型网络。它们都是为了克服超立方体的度无界性而由超立方体所派生出来的,与超立方体具有等价的计算功能。这一章的大部分篇幅也主要放在它们与超立方体的相互等价模拟证明上。第六章介绍超立方体类型网络上基本算法的实现,主要包括数据包路由选择算法、分类、FFT 算法等,从中发现了超立方体网络的计算通用性和强健性。同时也告诉读者如何在超立方体这么一类并行计算机系统上研制新的算法。

由于并行算法是与多学科有联系的交叉学科,研究范围十分广泛,国内 MPP 机并行算法研究还刚刚起步,我们在广泛研读了近几年来国外所发表的这方面有关文章和主要著作,特别是美国 F. T. Leighton 所著的“Introduction to Parallel Algorithms and Architecture”基础上,结合近几年的工作,写成了这本书。

本书第一章由李晓梅、窦勇编写,第二章由李晓梅、莫则尧编写,第三章由李晓梅、文尚猛编写,第四章由莫则尧、窦勇编写,第五、六章由莫则尧、文尚猛编写。全书由李晓梅统稿。本书难免挂一漏万,不足之处,敬请广大读者批评指正。

作 者

1995. 1. 10

符号说明

本书采用以下记号来描述函数的渐近性质,这与大多数科技文献是一致的:

1. $f(N)=O(g(N))$ 表示存在常数 N_0 及常数 C ,使得对任意 $N \geq N_0$ 均有 $f(N) \leq Cg(N)$.

2. $f(N)=\Omega(g(N))$ 表示存在常数 N_0 及常数 C ,使得对任意 $N \geq N_0$ 均有 $f(N) \geq Cg(N)$.

3. $f(N)=\theta(g(N))$ 表示存在常数 N_0 及常数 C_1, C_2 ,使得对任意 $N \geq N_0$ 均有 $C_1g(N) \leq f(N) \leq C_2g(N)$.

4. $f(N)=o(g(N))$ 表示 $\lim_{N \rightarrow \infty} f(N)/g(N)=0$. 即对任意正数 $C>0$,均存在 N_0 ,使得对任意 $N \geq N_0$ 均有 $f(N) < Cg(N)$.

5. $f(N)=\omega(g(N))$ 表示 $\lim_{N \rightarrow \infty} f(N)/g(N)=\infty$. 即对任意正数 $C>0$,均存在 N_0 ,使得对任意 $N \geq N_0$ 均有 $f(N) > Cg(N)$.

目 录

第一章 并行计算模型与并行算法设计

1.1 引言	(1)
1.2 PRAM 并行计算模型	(5)
1.2.1 SIMD 共享存储器 PRAM 并行计算模型(SIMD-PRAM)	(6)
1.2.2 MIMD 共享存储器 PRAM 并行计算模型(MIMD-PRAM)	(8)
1.2.3 SIMD-PRAM 和 MIMD-PRAM 算法设计举例	(10)
1.3 一种符合实际的并行计算模型——LogP	(15)
1.3.1 技术推动	(16)
1.3.2 LogP 模型	(18)
1.3.3 算法设计	(23)
1.3.4 模型与并行机的匹配	(28)
1.3.5 LogP 与 PRAM 模型的比较	(31)
1.4 其它并行计算模型	(33)
1.4.1 分布存储器多处理机体系结构的特点	(35)
1.4.2 分布存储器多处理机算法设计的特点	(38)
1.4.3 分布存储器多处理机中算法设计、分析的一般方法和原则	(38)
1.4.4 算法设计与分析举例	(39)

第二章 阵列和树

2.1 基本概念	(45)
----------------	------

2.1.1	阵列与树模型基本概念	(46)
2.1.2	固定连接网络的基本性质	(49)
2.1.3	算法性能评价准则	(50)
2.1.4	有关固定连接网络上并行算法的时间下界	(53)
2.1.5	一个示例——计数分类	(54)
2.2	整数运算	(60)
2.2.1	超前进位加法	(60)
2.2.2	并行前缀计算	(63)
2.2.3	进位保留加法	(67)
2.2.4	乘法与卷积	(70)
2.2.5	除法	(76)
2.3	矩阵运算	(78)
2.3.1	矩阵乘法	(78)
2.3.2	三角矩阵算法	(83)
2.3.3	三对角矩阵算法	(87)
2.3.4	高斯消去法	(93)
2.3.5	迭代法	(98)
2.4	分类算法	(104)
2.4.1	线性阵列上分类算法	(105)
2.4.2	奇偶比较交换分类	(113)
2.4.3	两种快速网格分类算法	(115)
2.4.4	网格分类算法的一个时间下界	(120)
2.5	数据包路由选择算法	(121)
2.5.1	贪心法	(122)
2.5.2	贪心法平均效能分析	(128)
2.5.3	随机路由选择算法	(137)
2.5.4	小队列确定型路由选择算法	(141)
2.5.5	其它路由选择算法	(144)

2.6 高维阵列	(147)
2.6.1 定义与性质	(147)
2.6.2 矩阵乘法	(149)
2.6.3 分类算法	(151)
2.6.4 数据包路由选择算法	(153)
2.6.5 低维阵列模拟高维阵列	(154)

第三章 树网

3.1 二维树网	(160)
3.1.1 定义和性质	(160)
3.1.2 $N \times N$ 树网的递归分解	(162)
3.1.3 $N \times N$ 树网可看作完全二分图 $K_{N \times N}$ 的近似	(162)
3.1.4 $N \times N$ 树网的变形	(163)
3.2 三维树网和高维树网	(165)
3.2.1 三维树网的定义和性质	(165)
3.2.2 高维树网的定义和性质	(166)
3.3 二维树网上一些基本算法的实现	(167)
3.3.1 路由选择	(167)
3.3.2 分类	(168)
3.3.3 矩阵向量乘	(169)
3.3.4 Jacobi 迭代	(169)
3.3.5 高斯主元消去法	(170)
3.3.6 卷积	(171)
3.3.7 整数运算	(173)
3.3.8 图论算法	(178)
3.4 三维树网上一些基本算法的实现	(183)
3.4.1 矩阵乘法	(183)

第一章 并行计算模型与并行算法设计

1.1 引言

研究并行算法的目的,就是探索在并行计算机上解决应用问题的高效方法。在求解应用问题的过程中,并行算法的设计是非常关键的一个环节,图 1-1 表示了一个应用问题使用并行计算机求解的基本过程。

1. 问题的抽象

适合于计算机处理的问题遍布社会生活的各个方面,计算机作为一种工具,已经渗透到人类生活的各个角落,如数值天气预报、石油

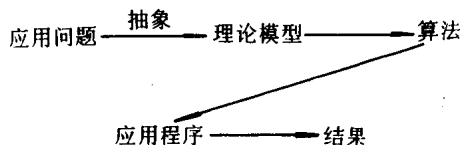


图 1-1 应用问题的基本求解过程

开采、生物医学等。计算机本身面对纷繁复杂的世界是无能为力的,只有先由科学家将特定领域的问题用形式化的公式、过程、定义加以抽象,建立一个模型(包括物理模型和数学模型),再由计算机做下一步的工作。因此求解问题的第一步是问题的抽象。

2. 模型的实现

进一步的工作是为建立的模型给出算法,应用问题是非常复杂的,虽然经过抽象、简化,但是其基本特点仍然保存在抽象模型中。典型的应用是由若干个相互作用的部分组成,每个部分可以看作是相对独立的模块。G. C. Fox 在 1989 年对 400 篇文章中所研

究的 84 种应用问题分析,将应用问题的模块分为如下五种形式:

同步模块:对大量数据做有规律的相同操作,如典型的矩阵运算、有限差分等。

松同步模块:对大量无规则数据进行操作,计算过程按时间或步分段进行,表现为宏观上的时间同步。典型的运算是对于不规则区域或者是异种数据区域的迭代计算。

异步模块:主要是时间不规则性,事件驱动的模式模拟问题就是典型的例子。

基本并行模块:由不相关联部分组成,可以独立执行。

松同步复合形模块:由若干同步模块组成,而且模块之间的关系是异步的。

对于具有不同特点的应用问题的模型,并行算法研究人员的工作就是研究能够在并行计算机上实现的高效方法,这部分工作是一个非常复杂的研究过程,它又可以分为以下几步(如图 1—2 所示)。

(1)算法分析:是为一个模型选择一个解决方法。如在数字信号处理问题中获取一个给定信号的光谱估计问题,算法分析从各种可能的解决方法中选择 FFT 算法来计算,但不涉及 FFT 算法细节。

(2)虚拟算法设计:确定要执行的计算。例如 FFT 算法的虚拟算法设计就是用数据流图或运算流图分析数据依赖关系以及理论上的最大并行性。

(3)理想算法:是基于最佳并行性分析得到理想计算方法。虽然没有实际效用,但是它可以启迪算法实现的策略和提供一些有价值的参考信息,是评价结构依赖性算法的标准。

(4)结构依赖性算法:确定理想算法在具体机器上的实现。结构依赖性算法的表现形式就是用程序设计语言描述的程序,实现结构依赖性算法要求算法的特点与机器的特性相互匹配,其主要

问题是如何保证计算负载的平衡,同时减少处理机间的通讯量,保证通讯量的平衡。算法转化为程序后,下一步的工作是程序在计算机上的运行求解。

3. 程序的执行

给定程序后,并行计算机系统是否能够正确、高效运行该程序,是计算机设计、实现人员的工作。其中包括软件人员对程序的优化、调度、管理,以及硬件设计人员在结构上的优化设计以支持程序正确、高效的执行,最后得到问题的解。

一个应用问题需要经过上述步骤后才能运用计算机这个工具得到解决。其中在算法的设计与实现、软件优化、硬件结构设计方面决定了计算机系统处理问题的能力。由并行处理技术来提高计算机的性能,主要就是研究这三方面的内容。

使用并行计算机解决一个

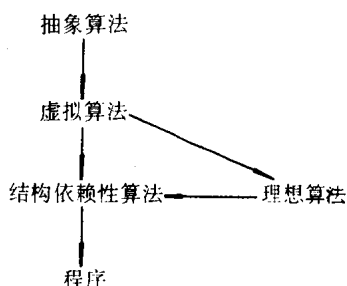


图 1-2 抽象算法的具体实现过程

应用问题时,并行算法的设计是非常重要的。而并行算法最终要成为一个由程序实现的结构依赖性算法,就特别需要一个抽象的并行计算机结构作为研究高效的结构依赖性算法的基础,以保证并行算法适应于广泛一类的并行计算机结构,并能够依照抽象的结构分析并行算法的效率,以及指导与并行机结构相匹配的并行算法的设计。并行计算模型就是为并行算法的设计、分析而研究并行计算机的抽象结构。

从并行计算机发展史来看,并行计算机的体系结构多种多样,因此构造一个合理的并行计算机的抽象模型是很困难的。过去的三十年中,人们针对不同类型的并行计算机给出了多种并行计算