向量算法与并行算法

张丽君　金绥更　编著

国防工业出版社

向量算法与并行算法

张丽君　金绥更　编著

国防工业出版社

(京)新登字 106 号

图书在版编目（CIP）数据

向量算法与并行算法/张丽君，金绥更编著．—北京：国防工业出版社，1993.

ISBN 7-118-01133-9

Ⅰ．向…

Ⅱ．①张…②金…

Ⅲ．①电子计算机--计算方法 ②线性代数计算法

Ⅳ．TP301.6

向量算法与并行算法

张丽君　金绥更　编著

赵克英　责任编辑

*

国防工业出版社出版发行

（北京市海淀区紫竹院南路23号）

（邮政编码 100044）

新华书店经售

北京市飞龙印刷厂印刷

*

850×1168毫米　开本32　印张$6^{3}/_{4}$　173千字

1993年10月第一版　1993年10月第一次印刷　印数：0001—1500 册

ISBN 7-118-01133-9/TP·145　定价：7.00元

国防科技图书出版基金
第一届评审委员会组成人员

致 读 者

本书由国防科技图书出版基金资助出版。

国防科技图书出版工作是国防科技事业的一个重要方面。优秀的国防科技图书既是国防科技成果的一部分，又是国防科技水平的重要标志。为了促进国防科技事业的发展，加强社会主义物质文明和精神文明建设，培养优秀科技人才，确保国防科技优秀图书的出版，国防科工委于1988年初决定每年拨出专款，设立国防科技图书出版基金，成立评审委员会，扶持、审定出版国防科技优秀图书。

国防科技图书出版基金资助的对象是：

1．学术水平高，内容有创见，在学科上居领先地位的基础科学理论图书；在工程技术理论方面有突破的应用科学专著。

2．学术思想新颖，内容具体、实用，对国防科技发展具有较大推动作用的专著；密切结合科技现代化和国防现代化需要的高新技术内容的专著。

3．有重要发展前景和有重大开拓使用价值，密切结合科技现代化和国防现代化需要的新工艺、新材料内容的科技图书。

4．填补目前我国科技领域空白的薄弱学科和边缘学科的科技图书。

5．特别有价值的科技论文集、译著等。

国防科技图书出版基金评审委员会在国防科工委的领导下开展工作，负责掌握出版基金的使用方向，评审受理的图书选题，决定资助的图书选题和资助金额，以及决定中断或取消资助等。经评审给予资助的图书，由国防工业出版社列选出版。

国防科技事业已经取得了举世瞩目的成就。国防科技图书承担着记载和弘扬这些成就，积累和传播科技知识的使命、在改革

开放的新形势下，国防科工委率先设立出版基金，扶持出版科技图书，这是一项具有深远意义的创举。此举势必促使国防科技图书的出版，随着国防科技事业的发展更加兴旺。

设立出版基金是一件新生事物，是对出版工作的一项改革。因而，评审工作需要不断地摸索、认真地总结和及时地改进，这样，才能使有限的基金发挥出巨大的效能。评审工作更需要国防科技工业战线广大科技工作者、专家、教授，以及社会各界朋友的热情支持。

让我们携起手来，为祖国昌盛、科技腾飞、出版繁荣而共同奋斗！

国防科技图书出版基金
评审委员会

前　言

现代科学技术的发展，提出了许多大型科学计算问题，这些课题要求用高速度、大容量的计算机系统加以处理。于是从70年代开始相继出现了多种不同类型的并行计算机。随之而来的是要研究能充分发挥这些机器并行处理能力的新算法——并行算法。近年来并行算法的研究已取得很多成果，并且逐渐成为计算数学领域里的一门崭新的学科。

并行计算机系统可分为单指令流多数据流（SIMD）和多指令流多数据流（MIMD）两种类型，相应的并行算法也截然不同。本书对适用于这两类计算机的算法都作了介绍，使读者既能同时掌握两类算法，又能通过比较，了解算法对机器的依赖性。

本书内容取材于国内外近年来发表的有关文献，包括编者的工作。既有理论分析也有具体方法，其中有些方法构思新颖、并行效率高，是近代发表的新方法。例如并行预处理共轭斜量法、异步迭代法、并行多分裂方法等。这些方法不但可供读者直接使用，并且能启发读者作进一步研究。特别，第三、四、五、六各章的内容，在国内外已出版的并行算法专著中，尚未见到过有系统的阐述。

本书是作者在为计算数学专业研究生多次讲课的讲稿基础上，根据教学经验和国内外的最新科研成果不断进行修改补充而成，在文字上也多次整理加工，因此是容易阅读和理解的。可作为读者开展并行算法研究的理论基础。

本书第一章由金绥更编写，其余各章由张丽君编写。全书由张丽君统稿。

由于编者水平有限，书中难免有错误和不确切之处，欢迎读者批评指正。

编者　　1992年 5月

内 容 简 介

本书重点介绍在SIMD和 MIMD 两种类型并行计算机上求解线性代数方程组的最新并行数值计算方法。全书共八章：第一章是预备知识；第二、三两章是适用于SIMD类型计算机的向量解法，特别突出并行预处理共轭斜量法；第四、五、六章介绍适用于 MIMD 类型计算机的同步与异步并行算法，包括任务系统的最优调度问题、异步迭代法和最近发展的并行多分裂方法，第七章介绍特征值问题的并行计算；最后一章是方程求根和非线性递推计算并行性分析的理论结果。

本书可供从事并行算法研究的工作者参考，也可作为计算数学专业研究生的教材或参考书。

目　　录

第一章　引　　言

并行算法依赖于一个简单而又重要的事实：独立的计算可以同时进行。通常根据问题本身的并行性来设计并行算法，或由识别一个串行算法的内在并行性加以改造而得到。对于给定的并行算法，则希望知道它的计算复杂性、并行效率、数值稳定性等，用以判定算法的优劣。这些性质与计算机的结构特征相关。故本章先对并行计算机作一简单介绍，然后介绍有关并行算法的一些基础知识。

1.1　并行计算机简介

1.1.1　阵列式处理机

阵列式处理机是重复设置多个同样的处理单元，通过互连网络以一定方式互相连接，根据控制器发出的命令，各台处理机对不同数据并行地完成同一条指令。所以是属于单指令流多数据流（简记 SIMD）类型的计算机。

阵列式处理机的存储器有多种组成方案。有的是把内存的各个并行存储体分布地设置在各处理单元中，称为处理单元存储器，合起来称为阵列存储器。对这种存储器存储数据时，要求合理分配，使各处理单元都可以主要依靠自身存储器中的数据进行运算。有的是设置一个共享的多体并行存储器，通过互连网络与各处理单元相连，可以最少受存储冲突的影响。这一方案在处理单元数目不太大的情况下是很理想的。

ILLIAC IV 计算机是阵列式处理机，它由 64 个处理单元、64 个处理单元存储器、存储器逻辑部件（总称为 64 个处理部件）及一个公共的控制器所组成。这 64 个处理部件排成 8×8 的方

阵，每一个与上下左右四个处理部件直接连接，如图 1-1 所示。

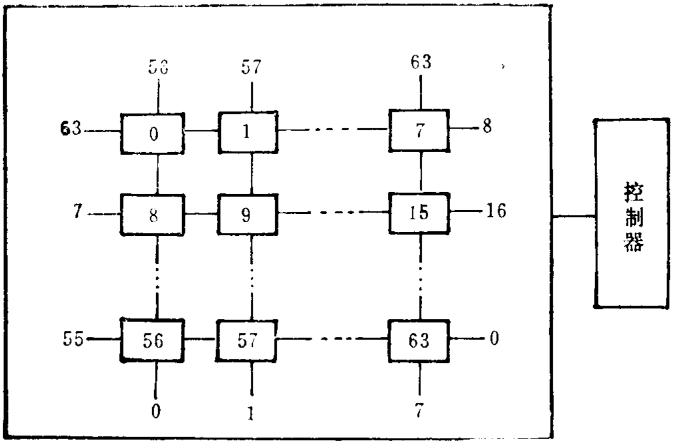

图1-1 ILLIAC IV 处理部件连接图

设在 ILLIAC IV 计算机上计算

$$c_i = a_i + b_i \quad (i = 1, 2, \cdots, N)$$

假定加法运算需要一个单位时间，不考虑数据存取、传送等开销，则计算时间为

$$T = \left\lceil \frac{N}{64} \right\rceil$$

（$\lceil x \rceil$表示取不小于x的最小整数）。一般地，若有P个处理单元，则N对数进行一次加法运算的时间是

$$T = \left\lceil \frac{N}{P} \right\rceil$$

又 ILLIAC IV 阵列有公共数据总线，用作向 64 个处理单元同时传送公共数据的通路。但当数据需要从一个处理单元传送到不是直接连接的另一个处理单元中时，就会引起存取的延迟，有可能严重影响程序的执行时间。为了提高速度，必须从算法角度考虑数据的存储方式、调度等问题。本书不作详细分析。下面举一个存储方式的例子。

设 $A = (a_{ij})$ 是 64 阶的方阵，在 ILLIAC IV 计算机上用两种存储方式：

（1）直阵存储法

$$
\begin{array}{cccc}
M_0 & M_1 & \cdots & M_{63} \\
\hline
a_{0,0} & a_{0,1} & \cdots & a_{0,63} \\
a_{1,0} & a_{1,1} & \cdots & a_{1,63} \\
\cdots & \cdots & & \\
a_{63,0} & a_{63,1} & \cdots & a_{63,63}
\end{array}
$$

即第 i 列元素存于第 i 个处理单元存储器内。这样，当对 A 作按行运算时，64 个处理单元可以同时工作，而对 A 作按列运算时，64 个元素在同一存储器内，同步运算发生了困难。

（2）斜阵存储法

$$
\begin{array}{cccc}
M_0 & M_1 & \cdots & M_{63} \\
\hline
a_{0,0} & a_{0,1} & \cdots & a_{0,63} \\
a_{1,63} & a_{1,0} & \cdots & a_{1,62} \\
 & \cdots & \cdots & \\
a_{63,1} & a_{63,2} & \cdots & a_{63,0}
\end{array}
$$

这种存储方式，无论对 A 作按行运算或按列运算，64 个处理单元都能同时工作。

1.1.2 流水线处理机

流水线处理机也属于 SIMD 类型，但其速度的提高主要是利用操作步骤并行，而不是设置多个处理单元。基本思想是将复杂的运算分解成若干个子运算，运算对象逐个地经过各子运算站，最后完成复杂的运算。由于各子运算站可以同时工作，所以运算速度大大地提高。例如可假设浮点加法分为求阶差、对阶、尾数加、规格化四个子运算站。当第一对操作数完成第一个子运算进入第二个子运算站时，第二对操作数就进入第一个子运算站进行加工，依次顺序进行，可顺次得到各个结果。

流水线计算机上的向量运算执行时间由两部分组成，即初始的延迟时间和结果顺次流出的间隔时间。记为

$$T = \sigma + N\tau$$

式中　σ——起步时间；

τ——结果顺次流出的间隔时间；

N——向量长度。

为进一步提高向量处理的速度，采用向量链接特性。例如CRAY-1计算机[1]，由于它的向量寄存器可以在同一时钟周期内既接受一个功能部件送来的运算结果，又可把这一结果作为下一次运算的源操作数送给另一个功能部件，这样就可把两个或两个以上的功能部件连接起来形成一条链子进行流水线处理。这就是CRAY-1的链接特性。并且计算机能自动检查每一条向量指令是否可以与前一条指令形成链。如果满足条件，则在前一条指令的第一个运算结果到达寄存器并可以用作本条指令的操作数时，立即启动本条指令而形成链。如果这条指令的操作数寄存器正是前一条指令的结果寄存器，但由于功能部件或另一操作数寄存器已被占用而不能满足条件，那么这条指令就必须等待到结果寄存器释放后才能开始执行。而CRAY—X—MP计算机的链接特性又有了改进[2]。

1.1.3　多处理机

多处理机是属于多指令流多数据流（简记MIMD）类型的计算机，是由多台处理机组成。多处理机同时对多条指令及其各自有关的数据进行处理，而SIMD类型的计算机只对多数据流执行同一条指令操作。由于各台处理机执行不同的运算，计算的并行性进一步提高，然而这给算法和系统软件的设计都带来困难。例如算法运行过程中，如果进程之间有数据往来或控制依赖，那末执行过程中有的进程就要中途停顿下来进入等待状态，直到它所依赖的执行条件满足为止。这就要求多处理机采取特殊的同步措施，才能使并行进程之间保持程序所要求的正确顺序。下面是个简单的例子。

例　计算

$$S=(A\times B)\times C+(D\times E)$$

A、B、C、D、E全是$N\times N$矩阵。用两台处理机P_1、P_2进行计算。

P_1:　　$X=(A\times B)\times C$

P_2:　　$Y=D\times E$，$S=X+Y$

显然P_2完成$D\times E$操作后，必须等待P_1工作结束，且X值由P_1传送到P_2，才能进行$S=X+Y$的操作。

1.2 并行算法的评价标准

为避免局限于个别计算机的特征而造成研究并行算法的困难，常常对具有下列假设的理想化机器进行分析。

常用的假设有：

（1）处理机台数不受限制（或向量寄存器长度不受限制）。

（2）每台处理机的运算执行时间相同，且忽略数据传送时间。

（3）在任何时刻有任意多个存储单元可供每台处理机利用，不存在任何存取冲突。

具有以上假设的机器为理论模型。

一个算法使用P台处理机去计算，理想的情况是应该比使用一台处理机计算快P倍，但除极少数简单问题外，实际速度提高要少得多。因此建立度量并行算法有效性的标准是很重要的。对于阵列式模型和多处理机模型有下面三个常用的评价标准，流水线模型可参照定义。

1.2.1 加速

对于一个给定的问题，用T_P表示所讨论的并行算法使用P台处理机的运行时间，T_1表示该算法在单处理机上的运行时间（有时T_1表示最优串行算法在单处理机上的运行时间）。定义

$$S_P=\frac{T_1}{T_P}$$