

矩阵相乘并行算法的 MPI 实现

A MPI-Program for Matrix Multiplication Parallel Algorithm

张锦雄

Zhang Jinxiong

(广西大学计算机与电子信息学院, 广西南宁 530004)

(Coll. of Comp. & Elec. Info., Guangxi Univ., Nanning, Guangxi, 530004, China)

摘要: 介绍行列划分算法和矩阵相乘并行算法 MPI 程序, 给出基于矩阵相乘并行算法的 MPI 实现, 分析和讨论处理器数目、复杂性、矩阵划分、B 子块传递、死锁避免和矩阵数据的获取等问题。

关键词: 矩阵相乘 并行算法 MPI

中图分类号: TP39

Abstract: A matrix multiplication parallel algorithm is introduced. A MPI-Program for matrix multiplication parallel algorithm is discussed in number of processors, complexity, partition of matrix, B-block transfer, deadlock avoidance, matrix data collecting.

Key words: matrix multiple, parallel algorithm, MPI

在并行机上实现矩阵相乘必须研究并行算法, 目前矩阵相乘并行算法有多种, 它们主要取决于矩阵的划分, 根据不同的划分有行列划分算法、行行划分算法、列列划分算法、列行划分算法和 cannon 算法^[1]. 本文采用行列划分算法, 文中给出基于矩阵相乘并行算法 MPI 实现.

1 行列划分算法^[1]

设矩阵 $A_{m \times k}$ 和 $B_{k \times n}$, 求解矩阵 $C_{m \times n} = A_{m \times k} \times B_{k \times n}$.

假设有 p 个处理器, $m = m' \times p, n = n' \times p$, 则矩阵 A 和 B 分别划分为行块子矩阵和列块子矩阵:

$$A = [A_0^T, A_1^T, \dots, A_{p-1}^T]^T,$$

$$B = [B_0, B_1, \dots, B_{p-1}],$$

此时 $C = (C_{i,j}) = (A_i \times B_j)$, 其中 $C_{i,j}$ 是 $m' \times n'$ 矩阵, A_i, B_j 和 $C_{i,j}, j = 0, 1, \dots, p-1$ 存放在 P_i 中. 每台处理器每次计算出一个 $C_{i,j}$, 各处理器一次计算出的 $C_{i,j}$ 按对角线分布, 每台处理器计算完一个 $C_{i,j}$ 后则按逻辑环的方式传递 B 的列块子矩阵, 整个算法共进行 p 次的计算、传递才完成矩阵 C 的计算. 矩阵 C 的子块矩阵 $C_{i,j}$ 的计算过程如图 1 所示.

图 2 为图 1 迭代步 p 中的虚线下三角阵调整位

置后所得的矩阵 C .

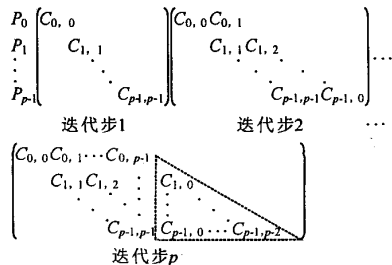


图 1 矩阵 C 的计算过程

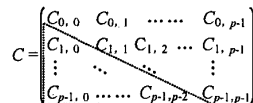


图 2 矩阵 C

2 矩阵相乘并行算法 MPI 程序

MPI 是消息传递接口 (Message Passing Interface) 的简称, 它作为一种标准已为各大并行机制造商所支持, MPI 程序可不加修改地在所有的并行机上运行. MPI 是一个函数库, 它采用 Fortran 和 C 语言进行绑定. MPI 是一种消息传递编程模型, 广泛应用于多类并行机的模式, 它适用于 SIMD、MIMD、SPMD 和 MPMD 等并行机类型. MPI 作为消息传递编程模型的代表和事实上的标准, 有多种实现版本, MPICH 是其中一种, 且 MPICH 随 MPI 标准的更新相应推出新版本, 本文算法采用 MPICH

-1.2.5.

2.1 MPI 程序的模式

MPI 程序是 SPMD 程序,也是 MPMD 程序^[2],二者的表达能力相同.对于 MPI 的 SPMD 程序,对等模式问题易于理解和接受,而主从模式问题,完全可以以 SPMD 形式通过分支控制语句指派处理器的功能角色来实现.因此 SPMD 是 MPI 程序的主要模式,SPMD 程序刻画各处理器行为的共性及处理器之间的通信关系,以实现数据并行和相互通信,功能并行通过分支控制语句实现角色分配.

MPI 程序要求有相对于 C 语言实现的头文件 mpi.h, MPI 程序的开始和结束采用 MPI_Init 和 MPI_Finalize. MPI 程序的程序体包括各种 MPI 过程调用语句和 C 语句, MPI_Comm_rank 返回当前正在运行进程的标识号, MPI_Comm_size 返回所有参加运算的进程数, MPI_Get_processor_name 返回进程所在机器的名称, MPI_COMM_WORLD 标识 MPI 通信域,以上是大部分 MPI 程序所共有的成分,它们反映进程在 MPI 通信域中的身份地位,进程在 MPI 通信域中的通信关系是由问题逻辑决定的,编程人员只需要调用相应的 MPI 函数即可实现.进程是一个任务实体,它占用处理器时间片,若把进程理解为伪处理器,则进程与第 3 节提到的处理器等同.

2.2 矩阵相乘并行算法描述

算法中指定某个处理器用于矩阵划分,并将子矩阵传递给其它处理器,当每个处理器都接到子矩阵后即开始计算,一次计算完毕则按逻辑环的方式传递 B 的列块子矩阵,如此重复计算、传递 p 次后每个处理器输出各自计算的矩阵 C 的行块子矩阵.处理器 P_i 执行程序的流程图如图 3 所示.

3 分析和讨论

3.1 处理器数目

消息传递企望减少通信开销,对工作站尤其如此.将问题分解为几个并行部分,分解到一定程度后,通信时间在总的执行时间中将占主导地位.故在保持足够并行性的同时,尽量增大计算/通信比^[3].本文提出的矩阵相乘并行算法中,由于处理器数目越多,每台处理器的计算量越少,每次通信交换的信息量也越少,但计算/通信比的变化仍难以直观判断.由于一次传递的时间 $T_c = T_{启动} + t \times n' = T_{启动} + t \times n/p$, $T_{启动}$ 是启动时间, t 是传递一列数据所需

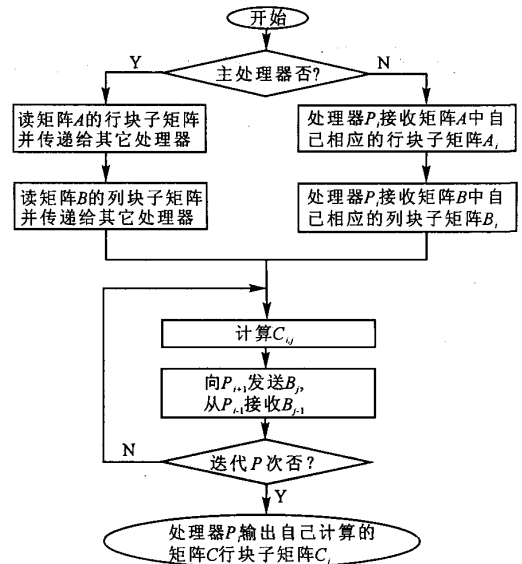


图3 程序流程

时间,因此,为提高数据通信效率,一次应传递多列.一次传送的列数在问题规模一定时与处理器数目有关,则总的通信时间 $t_r = 2p \times T_c = 2p \times T_{启动} + 2t \times n$,可见处理器数目越多,通信开销越大.

以上两点讨论都要求每台处理器应有相当的计算量,但不能因追求并行性而盲目加大处理器数目.另一方面,处理器数目不应比矩阵 A 的行数或矩阵 B 的列数多,即 $p \leq \min(m, n)$,否则出现有些处理器没有计算量却在通信环路中的情况,即计算/通信比为零.

3.2 复杂性分析

迭代步的执行时间有两部分,一部分为矩阵子块的计算时间,其乘法次数为 $m' \times k \times n'$,另一部分为传递数据的时间,数据传递次数为 2,算法共迭代 p 次.故有并行执行时间: $t_p = t_c + t_r = p \times (m' \times k \times n' + 2 \times O(k \times n')) = m \times k \times n/p + 2 \times O(k \times n)$.

由此看出,算法总的的数据交换量相当于 2 个矩阵 B 的数据交换量,算法的并行执行时间在计算时间占主导地位情况下,可通过增加处理器数目方式减少.

并行执行代价为: $C = p \times t_p = m \times k \times n + 2 \times O(k \times n) \times p = O(m \times k \times n)$.

3.3 矩阵划分

在矩阵划分时, m' 、 n' 并不一定是整数,此时在接近均分的基础上可让某些处理器多分配一行或一列.例如,设有 14 行按 4 个处理器划分,则有 2 个处理器分到 4 行,另 2 个处理器分到 3 行.

3.4 B 子块的传递

由上分析可知, B 的列块子矩阵可能列数不同, 可通过发送方将待发送子块列数先行发送来了了解接收子块的处理器接收的列数. 另外, 子块的列数应与待发送子块捆绑发送, 减少一次发送启动, 降低发送开销.

3.5 死锁避免

本文算法采用阻塞方式发送和接收数据, 由于所有处理器都采用先发送后接收的方式传递数据, 会产生死锁现象. 要避免死锁必须指定其中 1 个处理器以先接收后发送的方式传递数据(图 4), 以破坏死锁产生的环路条件.

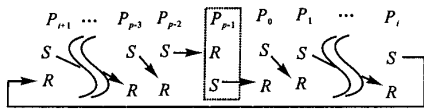


图 4 接收发送方式

P 表示处理器; S 表示发送; R 表示接收.

3.6 矩阵数据的获取

本文提出的算法处理的矩阵数据为事先输入, 并且以文件方式存储, 矩阵 A 以行顺序存储, 而矩阵 B 以列顺序存储, 其目的是使得列的划分问题转

化为行的划分, 从而使行、列的划分采用同样的代码实现. 为便于划分, 在向文件写入矩阵数据之前将矩阵 A 的行数、列数及矩阵 B 的列数先行写入文件.

4 结束语

现实世界中许多现象都表现出并行性, 众多问题的求解过程都有并行的可能性, 但由于人们习惯用 SISD 计算模型上的思维, 使得编写并行机执行程序变得不合常规, 其实, 底意识的并行才更接近问题. MPI 程序的 SPMD 编程模式给人们进行并行思维以很好的训练, MPI 的通信机制为人们在连网工作站上编写并实现并行程序提供了舞台, 使得问题求解变得更加自然.

参考文献:

- 1 孙家昶, 张林波, 迟学斌, 等编著. 网络并行计算与分布式编程环境. 北京: 科学出版社, 1996.
- 2 都志辉编著. 高性能计算并行编程技术——MPI 并行程序设计. 北京: 清华大学出版社, 2001.
- 3 Barry Wilkinson, Michael Allen 著. 并行程序设计. 陆鑫达, 等译. 北京: 机械工业出版社, 2002.

(责任编辑: 黎贞崇)

(上接第 216 页)

各个国家的研究人员对现有的语音编码算法进行深入研究, 也取得一定的成绩(如 0.8~1.5 比特/采样的窄带语音编码), 但各种输入信号的编码的统一模型还没有形成. 在不久的将来, 需要更为灵活的高质量数字宽带语音编码, 这将对宽带语音算法提出更高的要求.

参考文献:

- 1 ITU-T. Recommendation G. 723. Extensions of Recommendation G. 721 adaptive differential pulse code modulation to 24 and 40 kbit/s for digital circuit multiplication equipment application, 1988.
- 2 ITU-T. Recommendation G. 729-Coding of Speech at 8kbit/s using conjugate-structure algebraic-code-excited linear-prediction (CS-ACELP) 1996. Annex A: Reduced complexity 8kbit/s CS-ACELP speech codec, 1996.
- 3 CCITT. 7KHZ Audio Coding within 64kbit/s. In Recommendation G. 722, vol. III. 4 of Blue Book, Melbourne, 1998. 269~341.
- 4 Schnitzler Ju Krgen, Vary Peter. Trends and perspectives in wideband speech coding. Elsevier Science Signal Processing, 2000, (80): 2267~2281.
- 5 ITU-T. Recommendation G. 711, Pulse Code Modulation (PCM) of Voice Frequencies, 1988.
- 6 ITU-T. Recommendation G. 721, 32 kbit/s adaptive differential pulse code modulation (ADPCM). 1988.

- 7 Guyader A Le, Lamblin C, Boursicaut E. Embedded algebraic CELP/VSELP coders for wideband speech coding. Speech Communication, 1995, (16): 319~328.
- 8 Schnitzler Jurgan. A 13.0 kbit/s inband speech codec based on sb-acelp. Proc of ICASSP, 1998.
- 9 Combes Pierre, Schnitzler J. 16, 24, 32 kbit/s Wideband Speech Coder Based On ATCELP ICASSP, Phoenix, AZ, USA, IEEE, 1999.
- 10 Ubale Anil, Gersho Allen. A Low-delay Wideband Speech Coder At 24 kbps. In Proc. IEEE Int Conf Acoustics, Speech, Signal Processing, 1998. 165~168.
- 11 Kataoka A, Kurihara S, Sasaki S, et al. A 16-kbit/s wideband speech codec scalable with G. 729 Proc EURO-SPEECH'97, 1997. 1491~1494.
- 12 Asakawa Yoshiaki, Okamoto Toru, Susa Akiko, et al. A 24-kbit/s Wideband Speech Coder for Videoconferencing Systems. 1990. 1671~1674.
- 13 McCree A V. A 2.4Kb/s MELP coder candidate for the new U. S. Federal standard. Proc. ICASSP, 1996, 200~203.
- 14 McCree A V, Barnwell T P. A mixed excitation LPC vocoder moder for low bit rate speech coding. IEEE Trans. Speech & Audio Processing, 1995, 3 (4): 242~250.
- 15 覃团发, 刘运毅, 张淑仪. G. 729A 声码器的 VC++ 面向对象封装实现. 电声技术, 2003, (12): 13~16.

(责任编辑: 黎贞崇)