

文章编号: 1000-5013(2008) 03-0357-03

OTIS 网络结构的并行矩阵乘算法

魏文红¹, 秦 勇², 李清霞¹

(1. 华南理工大学 计算机学院, 广东 广州 510640; 2. 茂名学院 信息与网络中心, 广东 茂名 525000)

摘要: 提出基于光交换互连系统(OTIS) 网络结构的矩阵乘并行算法, 分析它的时间复杂性. 采用一种新映射策略来处理一般 OTIS 网络结构上的矩阵映射, 即矩阵映射策略是根据基图中的哈密尔顿路径来分配处理器的. 通过 OTIS 网络的拓扑结构模拟实验, 结果表明, OTIS 网络矩阵乘算法的性能优于 Cannon 算法, 更加优于 $O(n^3)$ 串行矩阵乘算法.

关键词: 矩阵乘法; 并行算法; 光交换互连系统; 映射策略; 拓扑结构

中图分类号: TP 393. 02

文献标识码: A

1993 年, Marshde 等^[1] 提出光交换互连系统(Optical Transpose Interconnection System, OTIS) 网络模型来解决网络通信的瓶颈问题. 孙家昶等^[2] 论述了基于一维条形划分的矩阵乘并行算法, 陈国良^[3] 转载了著名的 Cannon 算法和 Fox 算法, 吴建平^[4] 将 Cannon 算法由正方形网格处理机阵列推广到长方形网格处理机阵列. 但这些并行矩阵乘算法几乎都是基于二维网孔结构的. 在 OTIS 网络结构中, Wang 等^[5] 开发了基于 OTIS-Mesh 网络结构的并行矩阵乘算法, 可该算法只适合于规则的 $N \times N$ 矩阵, 对于一般的矩阵却无法进行运算. 而且, 该算法也只适用于 OTIS-Mesh 网络结构, 对于一般的 OTIS 网络结构也无法适用. 本文提出了一种适用于一般的 OTIS 网络结构的并行 $M \times N$ 矩阵乘算法.

1 OTIS 网络体系结构

设 $G = (V_G, E_G)$ 为无向图, 则 OTIS 网络结构(OTIS- $G = (V, E)$) 定义为^[1]

$$V = \{ \langle g, p \rangle \mid g, p \in V_G \},$$

$$E = \{ (\langle g, p_1 \rangle, \langle g, p_2 \rangle) \mid g \in V_G, (p_1, p_2) \in E_G \} \cup \{ (\langle g, p \rangle, \langle p, g \rangle) \mid g, p \in V_G, \text{ 且 } g \neq p \}.$$

在该网络结构中, G 是 OTIS- G 的基图(或称为基图). 若 G 有 P 个节点, 则 OTIS- G 由 P 个具有 P 个节点的基图 G 组成(即共有 P^2 个节点), 分别表示为 G_0, G_1, \dots, G_{P-1} , 称之为组. OTIS- G 中的节点 $\langle g, p \rangle$ 对应于组 G_g 的第 p 个节点. 一般地, 对于节点 $\langle g, p \rangle$, 把 g 称为组地址, 把 p 称为处理器地址. OTIS- G 中的边 $(\langle g, p_1 \rangle, \langle g, p_2 \rangle)$ 采用电子线缆连接, 边 $(\langle g, p \rangle, \langle p, g \rangle)$ 采用光纤连接, 是节点间的交换连接, 称之为 OTIS 网络连接. 图 1 是基图 G 为 C_4 的 16 个处理器的 OTIS 网络结构图.

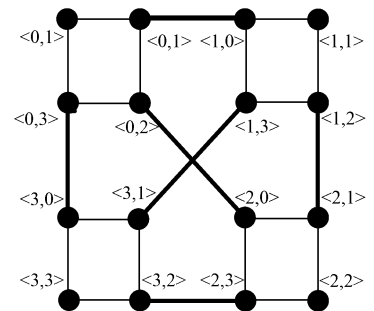


图 1 OTIS 网络结构

Fig. 1 The topology of OTIS network

2 并行矩阵乘算法

2.1 矩阵映射策略

已知矩阵 A, B, C , 其中, A 是 $M \times K$ 的矩阵, B 是 $K \times N$ 的矩阵, C 是 $M \times N$ 的矩阵. 假设在 P^2 个处理器的 OTIS 网络结构上进

收稿日期: 2007-11-04

作者简介: 魏文红(1977-), 男, 讲师, 博士研究生, 主要从事网络与并行分布式计算的研究. E-mail: hquw wh@ tom.com.

基金项目: 广东省自然科学基金资助项目(05011896); 广东省教育厅自然科学基金资助项目(Z03080)

行矩阵乘法 $C=A \times B, M>N>K$, 则矩阵乘法 $C=A \times B$ 为

$$D_{i,j}=\sum_{k=0}^{K-1}A_{i,k} \times B_{k,j}, \quad \forall i=0,1,\dots,M-1; \quad j=0,1,\dots,N-1. \tag{1}$$

对于如何把矩阵 A 和矩阵 B 分配给处理器, 在文[5]是采用 GRM 和 GSM 两种方式, 把矩阵映射到 OTIS-Mesh 网络中. 但这只适用于 OTIS-Mesh 网络结构, 并不适用于所有一般的 OTIS 网络结构. Parhami^[6] 证明了 OTIS 网络中存在哈密尔顿路径, 故矩阵映射策略就是根据基图中的哈密尔顿路径来分配处理器的. 在 OTIS 网络中, 矩阵的第 i 行映射到第 i 组, 矩阵行中的每个元素对应着一个处理器, 矩阵元素的列顺序是根据组(基图)中的哈密尔顿路径来排列. 如矩阵元素 $\langle i,j \rangle$ 就映射到 OTIS 网络中的节点 $\langle i,j \rangle$ 中, 当矩阵中的元素个数超过 OTIS 网络中处理器时, 可采用模运算来处理; 又如矩阵元素 $\langle i,j \rangle$ 映射到 OTIS 网络中的节点 $\langle x,y \rangle$ 中, 其中 $x=i \bmod P$ 和 $y=j \bmod P$. 整个矩阵的映射, 如图 2 所示.

2.2 算法实现

根据前面所述的映射策略对矩阵 A 和矩阵 B 进行映射, 把它们存储到 OTIS 网络中, 运算后得到的矩阵 C 也采用同样映射策略存储在 OTIS 网络中. 矩阵 A 映射到 OTIS 网络中的节点后, 对应的关系为矩阵 A 中的元素 $\langle i,k \rangle$ 映射到节点 $\langle x,y \rangle$, 其中 $x=i \bmod P$ 和 $y=k \bmod P$; 矩阵 B 映射到 OTIS 网络中的节点后, 对应的关系为矩阵 B 中的元素 $\langle k,j \rangle$ 映射到节点 $\langle x,y \rangle$, 其中 $x=k \bmod P$ 和 $y=j \bmod P$; 矩阵 C 的元素 $\langle i,j \rangle$ 映射到 OTIS 网络中的节点 $\langle x,y \rangle$, 其中 $x=i \bmod P$ 和 $y=j \bmod P$. 整个算法步骤有如下 4 个. (1) 在节点 $\langle x,y \rangle$ 中的矩阵元素 $B_{k,j}$ 通过组间连接与节点 $\langle y,x \rangle$ 中的矩阵元素 $B_{j,k}$ 进行交换. (2) 各组的节点广播其节点中 $A_{i,k}$ 和 $B_{k,j}$ 的值给同组的其它节点. (3) 在节点 $\langle y,x \rangle$ 中的所有元素 $B_{k,j}$ 通过组间连接与节点 $\langle x,y \rangle$ 中的所有元素 $B_{j,k}$ 进行交换. (4) 各组中的每个处理器节点都执行 $C_{i,j}=\sum_{k=0}^{K-1}A_{i,k}B_{k,j}$. 在算法中, $x,y=0,1,2,\dots,P-1; i=0,1,2,\dots,M-1; j=0,1,2,\dots,N-1; k=0,1,2,\dots,K-1$.

在算法执行过程中, 首先通过映射策略, 将矩阵 A 中的元素 $A_{i,k}$ 和 B 中的元素 $B_{k,j}$ 存储在 OTIS 网络的节点 $\langle x,y \rangle$ 中, 在执行步骤(1)之后, 矩阵 B 的元素 $B_{k,j}$ 存储在节点 $\langle y,x \rangle$ 中. 当算法执行步骤(2)之后, 对于任意的 $k=0,1,2,\dots,K-1$, 每个节点 $\langle x,y \rangle$ 中就存有矩阵 A 中元素 $A_{i,k}$ 的值; 每个节点 $\langle y,x \rangle$ 中就存有矩阵 B 中元素 $B_{k,j}$ 的值. 例如, 在节点 $\langle 0,1 \rangle$ 中存有 $A_{1,0}, A_{1,1}, \dots, A_{1,k}$ 和 $B_{1,0}, B_{2,0}, \dots, B_{k,0}$ 的值. 在步骤(3)之后, 节点 $\langle x,y \rangle$ 就存有了矩阵 B 中元素 $B_{k,j}$ 的值. 再例如, 在节点 $\langle 0,1 \rangle$ 中, 此时就存有 $A_{1,0}, A_{1,1}, \dots, A_{1,k}$ 和 $B_{0,1}, B_{1,1}, \dots, B_{k,1}$ 的值. 最后在执行步骤(4)时, 节点 $\langle x,y \rangle$ 按计算式(1)并把结果存储在节点 $\langle x,y \rangle$ 中. 整个算法结束后, 就可以在节点 $\langle x,y \rangle$ 中输出矩阵 C .

2.3 算法时间复杂度分析

设处理器平均传送每个数据的时间为 t_c , 平均每次四则运算需要的时间为 t_a , 则在算法中, 执行步骤(1)所需要的时间为 t_c . 这是因为所有的元素是同时进行移动的. 因此, 只要 1 个单位的传送数据时间即可. 步骤(2)为广播通信, 采用多对多的广播方式. 如果假定广播传送 1 个数据的时间为 t_b , 执行步骤(2)的时间为 t_b , 步骤(3)的执行时间同步骤(1), (4)为四则运算步. 每个处理器要做 k 次乘法和 $k-1$ 次加法, 因此执行该步需要的时间为 $(2k-1)t_a$. 所以算法在 OTIS 网络上的并行运行时间为

$$T_p=2t_c+t_b+(2k-1)t_a. \tag{2}$$

根据文[3], 在具有 P^2 个处理器的二维网孔上的 Cannon 算法的执行时间为

$$T'_{p,p}=(m+k)nt_c/p+2mnkt_a/p^2. \tag{3}$$

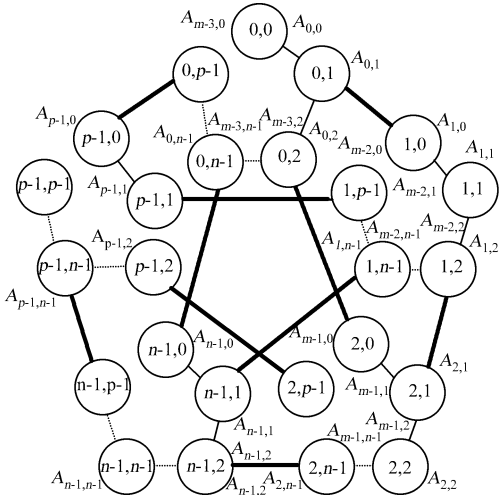


图 2 矩阵 A 映射到 OTIS 网络
Fig.2 The matrix A mapped onto OTIS network

比较式(2)和式(3), 显然两个算法的四则运算时间是相同的. 如果假定 OTIS 网络中的基图只有 1 条密尔顿路径的连接(此时广播传送数据的时间花费最大), 则 $t_b = (p - 1)t_c$, 式(2)进一步化简为

$$T_p = (p + 1)t_c + (2k - 1)t_a.$$

(4)

对比式(3)和式(4), 由于 $m \geq n \geq k \geq p$, 显然, 本文算法优于 Cannon 算法.

3 数值实验

按照图 1 所示的 OTIS 网络的拓扑结构模拟, 采用了 16 台 PC 机(P4 2.4 GHz, 512 MB)进行连接. 每台 PC 机都安装了 Linux 操作系统、MPICH 2 函数库(实验在 MPI 编程环境中进行). 同时, 采用了同样的 16 台 PC 机按照二维网孔的拓扑结构进行连接, 进行 Cannon 算法的实验, 此外, 还在单台 PC 机上进行了 $O(n^3)$ 串行矩阵乘算法的实验, 结果如图 3 所示. 从图 3 中可看出, OTIS 网络矩阵乘算法的性能优于 Cannon 算法, 更加优于 $O(n^3)$ 串行矩阵乘算法.

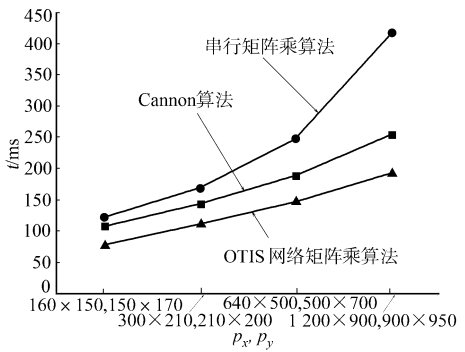


图 3 不同算法的执行时间比较
Fig. 3 Comparison between different algorithms in implementation time

4 结束语

本文结合 OTIS 网络结构的特点, 根据 OTIS-Mesh 网络结构中矩阵乘算法的启示, 开发出基于一般的 OTIS 网络的任意矩阵乘算法. 经过分析和实验证明, 该算法明显优于 Cannon 等算法.

参考文献:

[1] MARSDEN G, MARCHAND P, HARVEY P, et al. Optical transpose interconnection system architectures[J]. Optics Letters, 1993, 18(13): 1083-1085.

[2] 孙家昶, 张林波, 迟学斌, 等. 网络并行计算与分布式编程环境[M]. 北京: 科学出版社, 1996.

[3] 陈国良. 并行计算结构、算法、编程[M]. 北京: 高等教育出版社, 2003.

[4] 吴建平, 迟学斌. 分布式系统上并行矩阵乘法[J]. 计算数学, 1999, 21(1): 99-108.

[5] WANG C F, SAHNI S. Matrix multiplication on the OTIS-Mesh optoelectronic computer[J]. IEEE Trans on Computers, 2001, 50(7): 635-645.

[6] PARHAMI B. The Hamiltonicity of swapped (OTIS) networks built of Hamiltonian component networks[J]. Information Processing Letters, 2005, 95(4): 441-445.

Parallel Algorithm for Matrix Multiplication on the OTIS Network

WEI Wen-hong¹, QIN Yong², LI Qing-xia¹

(1. Department of Computer Science, South China University of Technology, Guangzhou 510640, China;
2. Information and Network Center, Maoming College, Maoming 525000, China)

Abstract: A parallel algorithm for matrix multiplication based on optical transpose interconnection system (OTIS) network is proposed, and the time complexity is analyzed. A new mapping scheme is used to map matrix to the general OTIS network, that is the matrix mapping scheme, processors are assigned according to Hamiltonian path in the basic graph of OTIS network. A simulation experiment about the topology of OTIS network is done, and the result shows that the algorithm for matrix multiplication on OTIS network is better than Cannon and the $O(n^3)$ serial algorithm.

Keywords: matrix multiplication; parallel algorithm; optical transpose interconnection system; mapping scheme; topology

(责任编辑: 黄仲一 英文审校: 吴逢铁)