

关联维数的并行求解算法^{*}

蒋廷耀^{1,2} 李庆华² 杨景华¹

(三峡大学电气信息学院 宜昌443000)¹ (华中科技大学计算机科学与技术学院 武汉430074)²

摘要 关联维数的求解是分形理论中的一个重要问题,标准算法由于其巨大的计算量,不能满足实时任务的需要。过去的改进算法集中在串行地减少求解多个关联维数时的重复计算量,并未从根本上降低 $O(N^2)$ 次的向量距离计算、距离比较和求和次数,其应用范围和性能改善程度是有限的。本文给出了两个并行算法:基于PRAM模型的花费 $O(N^2/p + \log p)$ 时间 p 个处理机的算法,和基于LARPBS模型的花费 $O(N^2/p)$ 时间 p 个处理机的算法。相对纯理论的PRAM算法,LARPBS算法是实际可行的,它是目前时间复杂度最低的算法,并且是最优可扩展和成本最优的。

关键词 关联维数,并行,算法

The Research of Parallel Correlation Dimension Calculation

JIANG Ting-Yao^{1,2} LI Qing-Hua² YANG Jing-Hua¹

(College of Electric and Information, Sanxia University, Yichang 443000)¹

(College of Computer Sci. & Tech., Huazhong University of Science & Technology, Wuhan 430074)²

Abstract The calculation of correlation dimension is a key problem of the fractal dimension. The standard algorithm requires $O(N^2)$ computations. The previous improvement methods endeavor to sequentially reduce redundant computation on condition that there are many different dimensional phase spaces. The application area and performance improvement degree are limited, e.g., they are not adequate for real time tasks. This paper presents a $O(N^2/p + \log p)$ time p processors parallel PRAM algorithm and a $O(N^2/p)$ time p processors parallel LARPBS algorithm. The speedup of parallel algorithms is efficient. Compared with the PRAM algorithm, The LARPBS algorithm is practical. It is optimally scalable and cost optimal. To our best knowledge, it is the fastest algorithm for correlation dimension calculation so far.

Keywords Correlation dimension, Parallel, Algorithm

1 引言

用分形^[1]来描述自然界中一些极不规则、极不稳定、有高度复杂结构的现象时有非常好的效果,在混沌信号处理,图像处理,地理、生态等领域有着重要的应用。关联维数的求解是分形理论中的一个重要问题^[2,3],标准算法由于其 $O(N^2)$ 的巨大计算量(N 是样本点个数),不能满足实时任务的需要。过去的改进算法^[4~7]是串行的,集中在假定嵌入维的维数是不确定的情况下减少进行向量距离计算时的重复计算量,性能的改善随着所需计算的维数变化而变化;另外这些算法并未从根本上降低 $O(N^2)$ 次的向量距离计算、比较和求和次数,其应用范围和性能改善程度都是有限的。本文给出了两个并行算法:基于CRCW(concurrent read concurrent write)PRAM(parallel random access machine)模型的花费 $O(N^2/p + \log p)$ 时间 p 个处理机的算法,和基于LARPBS(linear arrays with a reconfigurable pipelined bus systems)模型^[8~12]的花费 $O(N^2/p)$ 时间 p 个处理机的算法。并行算法的加速较串行算法是高效率的。相对纯理论的PRAM算法,LARPBS算法是实际可行的,并且是最优可扩展和成本最优的,它是目前时间复杂度最低的算法。

2 标准算法

定义1 关联函数 $C(r)$ 定义如下:
$$C(r) = \frac{1}{N^2} \sum_{i=1}^N \sum_{j=1, j \neq i}^N \Theta(d_{ij}, r)$$

(d_{ij}, r)

$$\Theta(d_{ij}, r) = \begin{cases} 1 & d_{ij} < r \\ 0 & d_{ij} \geq r \end{cases} \quad d_{ij} = |\vec{x}_i - \vec{x}_j|$$

其中, $\vec{x}_i, \vec{x}_j$ 是用时间延迟技术^[13]从一维时间序列中构造的 m 维的向量, m 是嵌入维数, r 是向量间距离阈值。

定义2 关联维数 D_2 定义如下:
$$D_2 = \lim_{r \rightarrow 0} \frac{\log C(r)}{\log r}$$

显然,标准的关联维数 D_2 的计算量主要在关联函数 $C(r)$ 的计算,一次 $C(r)$ 的计算需进行 $O(N^2)$ 次向量间距离计算、距离比较和求和操作,只有有效降低 $C(r)$ 的计算时间复杂度才能有效缩短 D_2 的计算时间。这是过去的串行算法尚未解决的,而采用并行算法却可以很好地解决。

3 并行计算模型

PRAM是一个纯理论的并行计算模型,它忽略了处理机间的通行开销,本文将采用CRCW PRAM模型实现第一个并行算法。LARPBS是近年出现的一种高效的并行计算模型,与理想的PRAM模型不同,LARPBS是现实可行的。LARPBS使用光纤总线来互连电子处理机,用光波导(waveguide)代替电子总线在处理机间传递消息。除了光的高速传播特性外,光的传播还具有单向传递和单位长度上可预知的传播延迟特性,因此消息可以并发地以流水线形式在光纤总线上传递,最大的消息传递延迟是光脉冲在光纤上的端到端的传递延迟,称为一个光纤总线周期。

^{*})本文得到国家自然科学基金资助(No. 60273075)。蒋廷耀 博士研究生,研究方向:高性能计算,计算机通信;李庆华 教授,博导,研究方向:高性能计算机。

LARPBS 模型的一个实例如图1所示,处理机用光纤总线互连,每个处理机可在光纤总线的上半段由发射耦合器向光纤总线注入信号脉冲而在下半段由接受耦合器接受信号脉冲。所有处理机可互连作为一个线性阵列(或称为总线系统),也可分段成为多个独立子线性阵列。RST(i)和 RSR(i)是可重配置交换机,有直通和交叉2种状态,图1中 RST(2)、RSR(2)和 RST(4)、RSR(4)是交叉状态,其它重配置交换机处于直通状态,则整个总线系统分裂成为2个独立的子系统,互不

影响。光纤总线包含了3个波导:一个用于发送消息(message waveguide),另2个用于地址选通(reference waveguide 和 select waveguide)。如图2所示,条件延迟器在 select 波导上用一个2X2的光纤交换机实现,每个光纤交换机有直通和交叉2种状态。在交叉状态引入一个 ω 延迟(ω 是一个光脉冲的时间宽度)。下半段 reference 和 message 波导每经过一个处理机引入一个 ω 延迟。当 select 和 reference 脉冲同时到达一个处理机时,该处理机接受 message 波导上的消息帧。

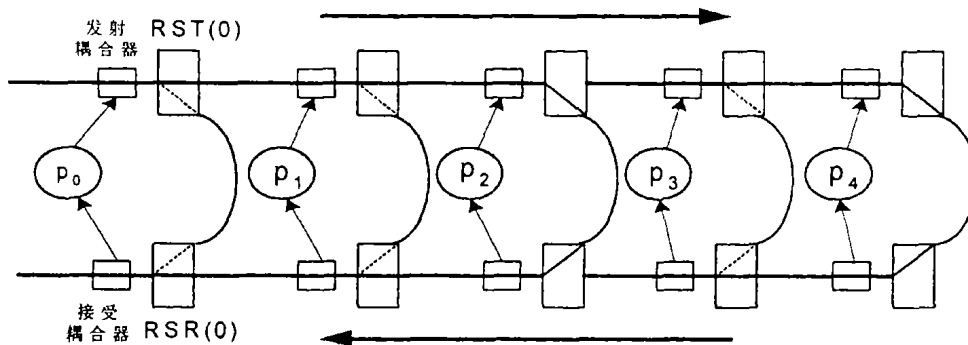


图1 5个处理机分成2个子阵列的一个 LARPBS 模型

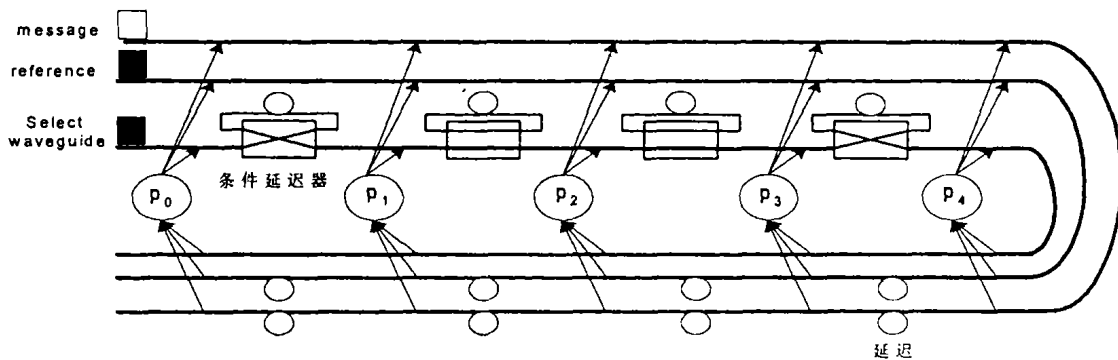


图2 波导和延迟

作为一个同步并行计算系统,LARPBS 的计算由一个交替的全局通信步和局部计算步系列完成。只要局部计算步的时间边界是个常数并大致等于一个光纤总线周期,则算法的时间复杂度就可以用光纤总线周期的个数来度量。LARPBS 能够在常数时间($O(1)$ 个总线周期)内实现一些基本的通信、数据移动和全局操作。这里列举了本文所需的几个操作,为了帮助理解,我们以多播通信为例说明了其操作过程,其它操作参见文[8~12]。

(1)多播通信 多播通信是一个处理机同时向系统中的其它多个处理机发送消息。图2中,若 p_1 向 p_0 、 p_4 多播消息,操作过程如下:所有条件延迟器处于直通状态。在总线周期的开始时刻 t_{ref} , p_1 发射 reference 脉冲和 message 帧,并在 t_{ref} 时刻和 $t_{ref} + 4\omega$ 时刻各发射一个 select 脉冲。3种脉冲在各自的波导上传播,message 帧、reference 脉冲和第一个 select 脉冲会同时到达 p_4 , p_4 接受 message 帧;光脉冲继续往下传播,message 帧、reference 脉冲和第二个 select 脉冲会同时到达 p_0 , p_0 接受 message 帧。这样多播通信在一个总线周期内即可完成。

(2)一对一通信 一个处理机向另一个处理机发送消息。若处理机 i 向处理机 j 发送消息,则 select 脉冲的发送时刻 $t_{sel}(j) = t_{ref} + (P - j - 1)\omega$,与发送处理机的位置无关(P 是处理机数,处理机 i 右边的所有条件延迟器处于直通状态)。

(3)广播通信 一个处理机向总线系统上的其它所有处

理机发送消息。

(4)组多播通信 多个多播操作,多播组之间无交叠。

(5)二进制前缀和操作 P 个二进制数 x_i ($0 \leq i \leq P-1$) 分布在 P 个处理机上,各处理机计算 $s_i = x_0 + x_1 + \dots + x_i$ 。

4 并行关联维数 D_2 求解算法

本文的 CRCW PRAM 算法和 LARPBS 算法基于 SIMD (单指令流多数数据流,即所有处理机同时执行相同的程序)机器。设有 p 个处理机,顺序编号为 $0, 1, \dots, p-1$,每个处理机有自己的编号 pid 。

4.1 PRAM D_2 算法

For all processors parallel do

For $k=1$ to N^2/p

$t = pid * N^2/p + k$

$i = \lceil t/N \rceil$

$j = (t - (i-1) * N) \bmod (N+1)$

If $(i=j)$ then

$h_{pid} = 0$

Else

If $(|\vec{x}_i - \vec{x}_j| < r)$ then $h_{pid} = 1$ else $h_{pid} = 0$

$\Theta_{pid} = \Theta_{pid} + h_{pid}$

Endfor

For $k=0$ to $\log p - 1$

If $(pid \bmod 2^k = 0 \text{ and } 2 * pid + 2^k < p)$ then $\Theta_{2 * pid} = \Theta_{2 * pid} + \Theta_{pid + 2^k}$

Endfor

(下转封四)