

第 5 讲 并行算法设计

1. 试证明 Brent 定理: 令 $W(n)$ 是某并行算法 A 在运行时间 $T(n)$ 内所执行的运算数量, 则 A 使用 p 台处理器可在 $t(n) = O(W(n) / p + T(n))$ 时间内执行完毕。

2. 假定 $P_i (1 \leq i \leq n)$ 开始时存有数据 d_i , 所谓累加和指用 $\sum_{j=1}^i d_j$ 来代替 P_i 中的原始值

d_i 。

算法 PRAM-EREW 上累加和算法

输入: P_i 中保存有 $d_i, 1 \leq i \leq n$

输出: P_i 中的内容为 $\sum_{j=1}^i d_j$

begin

for $j = 0$ **to** $\log n - 1$ **do**

for $i = 2^j + 1$ **to** n **par-do**

(i) $P_i = d_{i-(2^j)}$

(ii) $d_i = d_i + d_{i-(2^j)}$

endfor

endfor

end

(1) 试用 $n=8$ 为例, 按照上述算法逐步计算出累加和。

(2) 分析算法时间复杂度。

3. 在 APRAM 模型上设计算法时, 应尽量使各处理器内的局部计算时间和读写时间大致与同步时间 B 相当。当在 APRAM 上计算 M 个数的和时, 可以借用 B 叉树求和的办法。

假定有 j 个处理器计算 n 个数的和, 此时每个处理器上分配 n/p 个数, 各处理器先求出自身的局和; 然后从共享存储器中读取它的 B 个孩子的局和, 累加后置入指定的共享存储单元 SM 中; 最后根处理器所计算的和即为全和。算法如下:

算法 APRAM 上求和算法

输入: n 个待求和的数

输出: 总和在共享存储单元 SM 中

Begin

(1) 各处理器求 n/p 个数的局和, 并将其写入 SM 中

(2) **Barrier**

(3) **for** $k = \lceil \log_B(p(B-1) + 1) \rceil - 2$ **downto** 0 **do**

3.1 **for all** $P_i, 0 \leq i \leq p-1$, **do**

if P_i 在第 k 级 **then**

P_i 计算其 B 个孩子的局和并与其自身局和相加, 然后将结果写入 SM 中

endif

end for

3.2 **barrier**

end for

End

(1)试用 APRAM 模型之参数, 写出算法的时间复杂度函数表达式。

(2)试解释 Barrier 语句的作用。

4. 在给定时间 t 内, 尽可能多的计算输入值的和也是一个求和问题, 如果在 $\log P$ 模型上求此问题时, 要是 $t < L+2 \cdot 0$, 则在一个单处理机上即可最快地完成; 要是 $t > L+2 \cdot 0$ 时, 则根处理器应在 $t-1$ 时间完成局和的接收工作, 然后用一个单位的时间完成加运算而得最终的全和。而根的远程子节点应在 $(t-1) - (L+2 \cdot 0)$ 时刻开始发送数据, 其兄妹子节点应依次在 $(t-1) - (L+2 \cdot 0+g)$, $(t-1) - (L+2 \cdot 0+2g)$, \dots 时刻开始发送数据。图示出了 $t=28$, $p=8$, $L=5$, $o=2$, $g=4$ 的 $\log P$ 模型上的通信(即发送/接收)调度树。试分析此通信调度树的工作原理和图中节点中的数值是如何计算的?

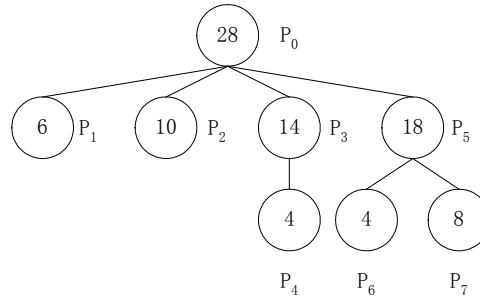


图 1.50 $t=28$, $p=8$, $L=5$, $o=2$, $g=4$ 的通信调度树

欲在 8 个处理器的 BSP 模型上, 计算两个 N 阶向量内积:

①试画出各超级步的计算过程(假定 $N=8$);

②并分析其时间复杂度。

5. 以下哪个关于并行算法设计过程的描述是正确的?
- A. 映射的策略是使得任务可以被不同处理器并发地执行, 增强局部性, 将通信频繁的任务放到同一个处理器上, 增强并发性
 - B. 重复计算减少了通信量, 但增加了计算量, 应保持恰当的平衡
 - C. 在通信阶段, 要尽可能将局部通信转换为全局通信
 - D. 域分解的划分对象是计算, 功能分解的划分对象是数据
6. DNS 并行矩阵算法的输入是两个 $n \times n$ 的矩阵, 算法输出为 $n \times n$ 的矩阵。当 $p = n^3$ 时, 该算法的时间复杂度为:
- A. $O(\log n)$
 - B. $O(p \log n)$
 - C. $O(n)$
 - D. $O(n^3 / p)$
7. 请结合实例说明, 你觉得对并行算法设计而言, 最重要的一个步骤是什么?
8. 对于一个具有良好可扩充性的并行算法, 任务的规模(或是任务的个数)会不会随着问题的规模的增加而增加? 为什么?
9. 并行算法设计的一般过程 PCAM 是指什么? 各个步骤中的主要判据是什么?
10. 什么是域分解和功能分解? 如何将全局通信转换为局部通信? 什么是表面-容积效应和重复计算? 映射的策略是什么?