

并行计算

——结构•算法•编程

主讲教师：谢磊

第二篇 并行算法的设计

第四章 并行算法的设计基础

第五章 并行算法的一般设计方法

第六章 并行算法的基本设计技术

第七章 并行算法的一般设计过程

第七章 并行算法的一般设计过程

7.1 PCAM设计方法学

7.2 划分

7.3 通讯

7.4 组合

7.5 映射

7.6 小结

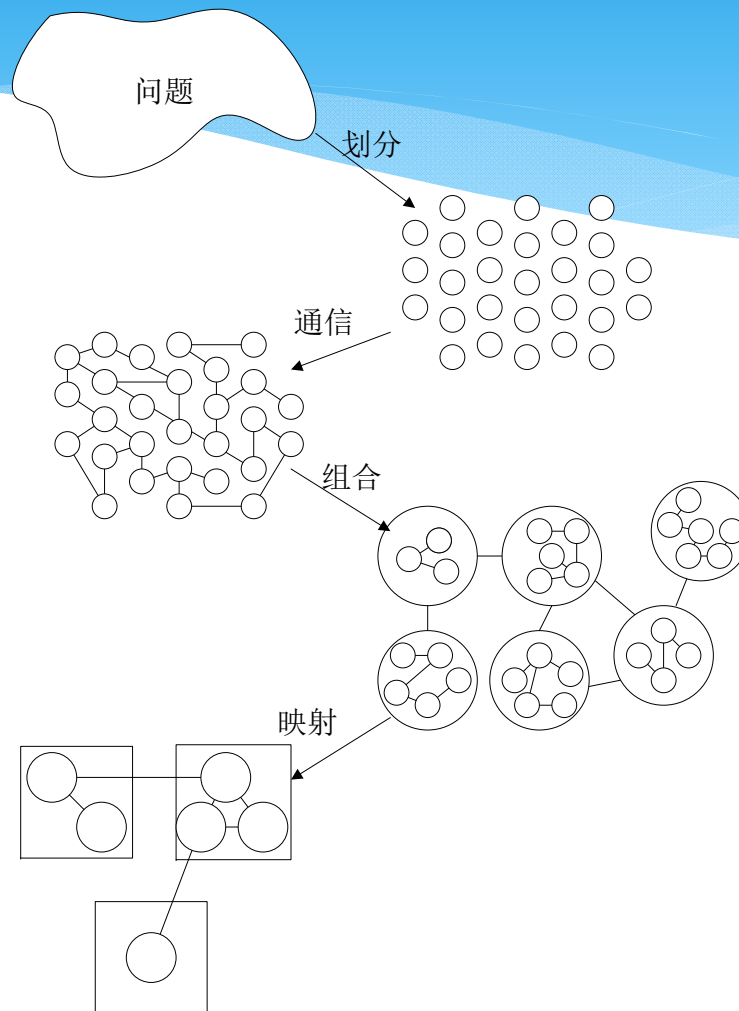
PCAM设计方法学

* 设计并行算法的四个阶段

- * 划分(Partitioning)
- * 通讯(Communication)
- * 组合(Agglomeration)
- * 映射(Mapping)

- * 划分： 分解成小的任务，开拓并发性；
- * 通讯： 确定诸任务间的数据交换，监测划分的合理性；
- * 组合： 依据任务的局部性，组合成更大的任务；
- * 映射： 将每个任务分配到处理器上，提高算法的性能。

PCAM设计过程



graph.

第七章 并行算法的一般设计过程

7.1 PCAM设计方法学

7.2 划分

7.3 通讯

7.4 组合

7.5 映射

7.2 划分

7.2.1 方法描述

7.2.2 域分解

7.2.3 功能分解

7.2.4 划分判据

划分方法描述

- * 充分开拓算法的并发性和可扩展性;
- * 先进行数据分解(称域分解), 再进行计算功能的分解(称功能分解);
- * 使数据集和计算集互不相交;
- * 划分阶段忽略处理器数目和目标机器的体系结构;
- * 能分为两类划分:
 - * 域分解(domain decomposition)
 - * 功能分解(functional decomposition)

7.2 划分

7.2.1 方法描述

7.2.2 域分解

7.2.3 功能分解

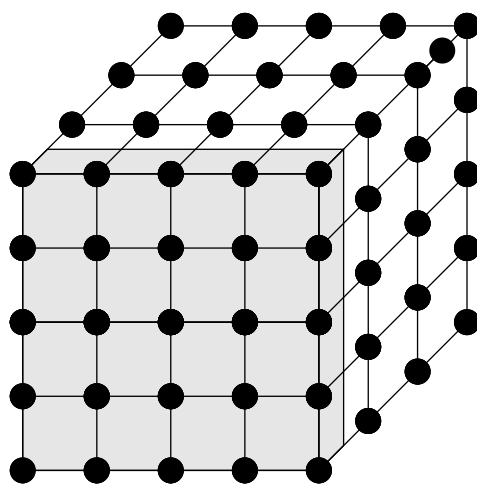
7.2.4 划分判据

域分解

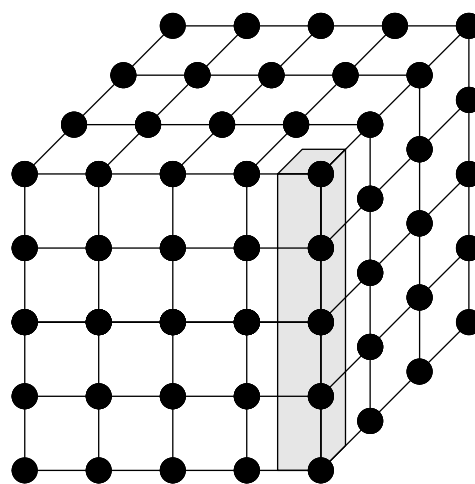
- * 划分的对象是数据，可以是算法的输入数据、中间处理数据和输出数据；
- * 将数据分解成大致相等的小数据片；
- * 划分时考虑数据上的相应操作；
- * 如果一个任务需要别的任务中的数据，则会产生任务间的通讯；交换
(Database \rightarrow join) .

域分解

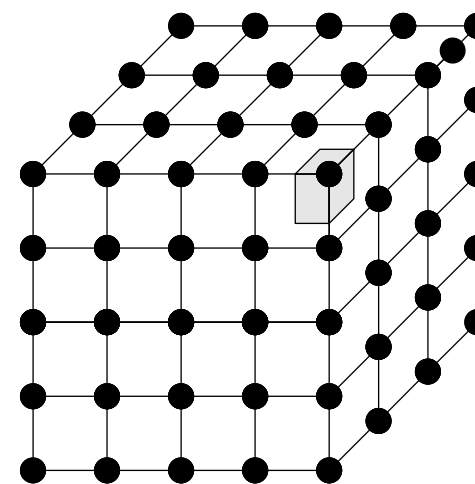
* 示例：三维网格的域分解，各格点上计算都是重复的。下图是三种分解方法：



1-D



2-D

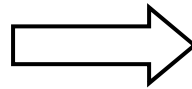
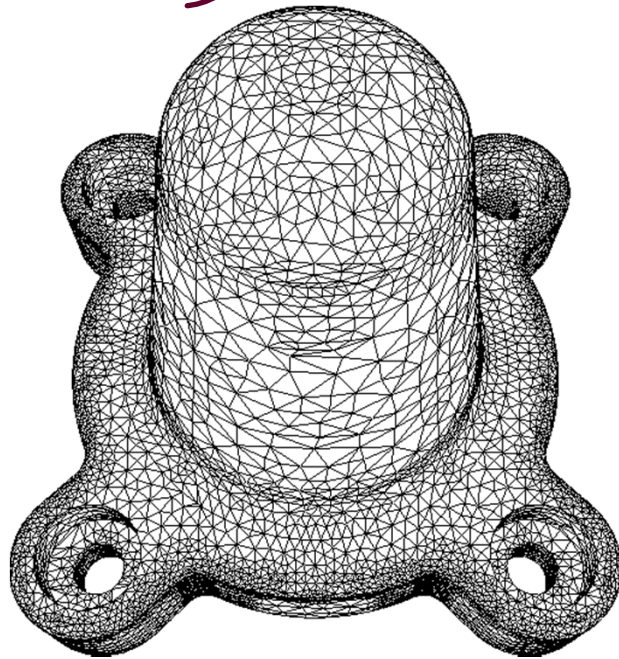


3-D

图7.2

域分解

* 不规则区域的分解示例：



7.2 划分

7.2.1 方法描述

7.2.2 域分解

7.2.3 功能分解

7.2.4 划分判据

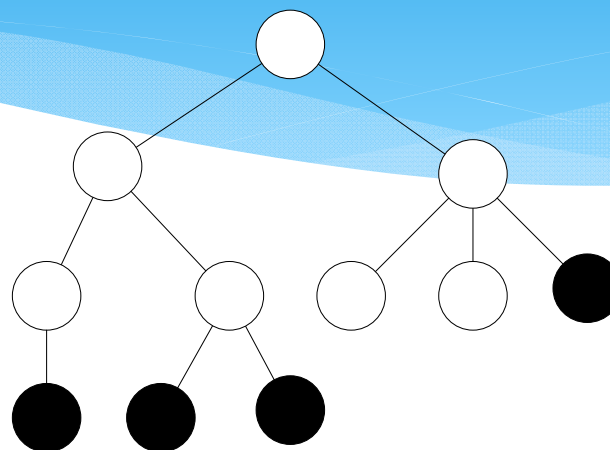
功能分解

从逻辑角度.

- * 划分的对象是计算，将计算划分为不同的任务，其出发点不同于域分解；
- * 划分后，研究不同任务所需的数据。如果这些数据不相交的，则划分是成功的；如果数据有相当的重叠，意味着要重新进行域分解和功能分解；
- * 功能分解是一种更深层次的分解。

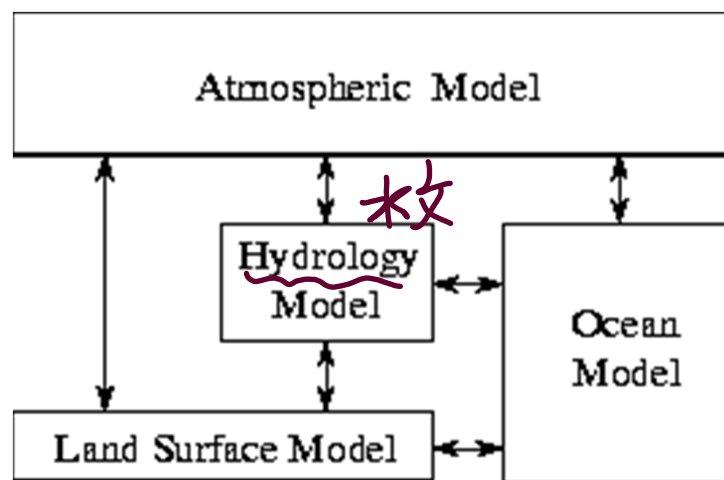
功能分解

* 示例1：搜索树



* 示例2：气候模型

冷启动.



7.2 划分

7.2.1 方法描述

7.2.2 域分解

7.2.3 功能分解

7.2.4 划分判据

划分判据

- * 划分是否具有灵活性？
- * 划分是否避免了冗余计算和存储？
- * 划分任务尺寸是否大致相当？
- * 任务数与问题尺寸是否成比例？
- * 功能分解是一种更深层次的分解，是否合理？

第七章 并行算法的一般设计过程

7.1 PCAM设计方法学

7.2 划分

7.3 通讯

7.4 组合

7.5 映射

7.3 通讯

7.3.1 方法描述

7.3.2 四种通讯模式

7.3.3 通讯判据

通讯方法描述

- * 通讯是PCAM设计过程的重要阶段；
- * 划分产生的诸任务，一般不能完全独立执行，需要在任务间进行数据交流；从而产生了通讯；
- * 功能分解确定了诸任务之间的数据流；
- * 诸任务是并发执行的，通讯则限制了这种并行性；

7.3 通讯

7.3.1 方法描述

7.3.2 四种通讯模式

7.3.3 通讯判据

四种通讯模式

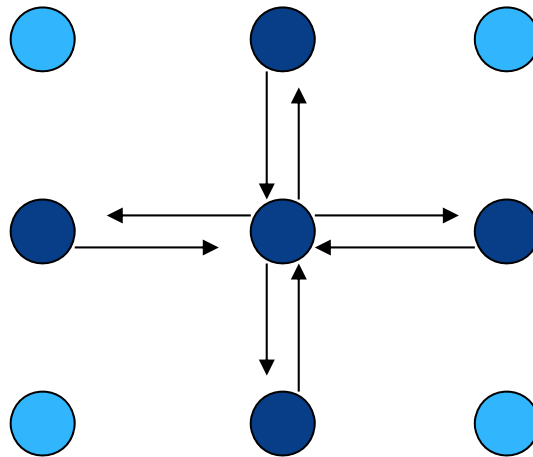
- * 局部/全局通讯
- * 结构化/非结构化通讯
- * 静态/动态通讯
- * 同步/异步通讯

局部通讯

* 通讯限制在一个邻域内

网格 两两之间交换.

多跳代替单跳
多次传输



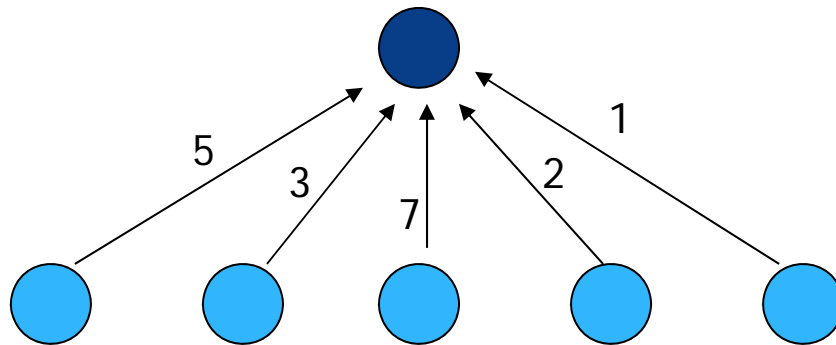
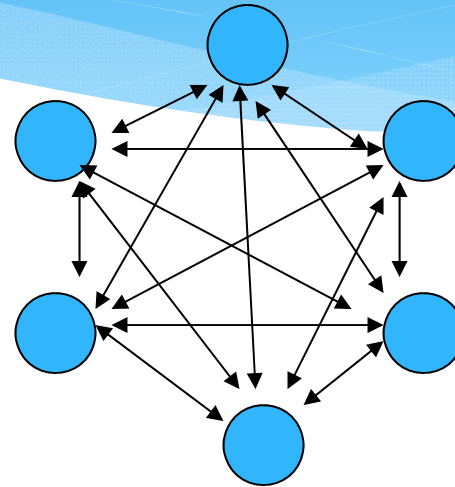
全局通讯 在一些比较重要的场合.

- * 通讯非局部的

- * 例如:

- * All to All

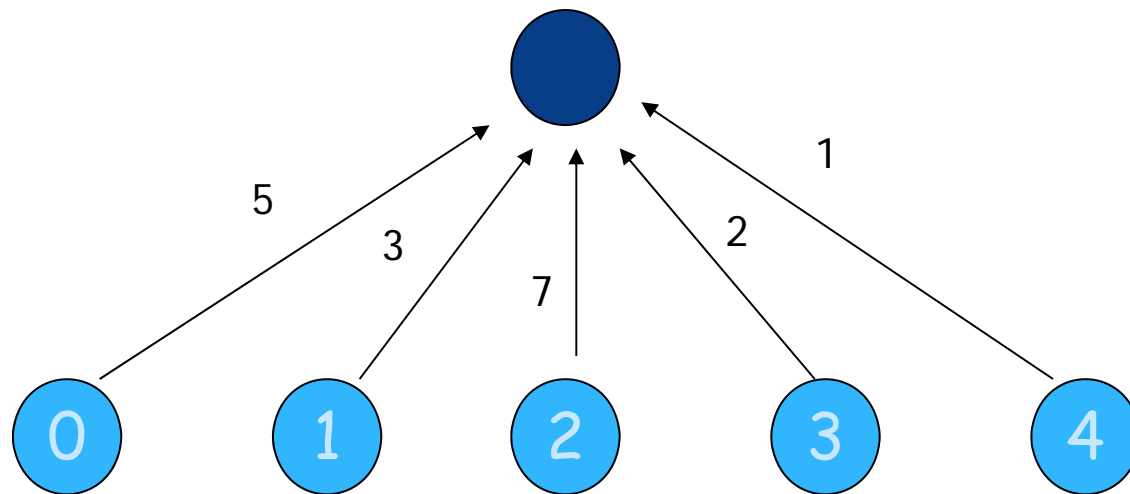
- * Master-Worker 专线.



全局通信

* 集中式顺序求和 $S = \sum_{i=0}^{N-1} x_i$

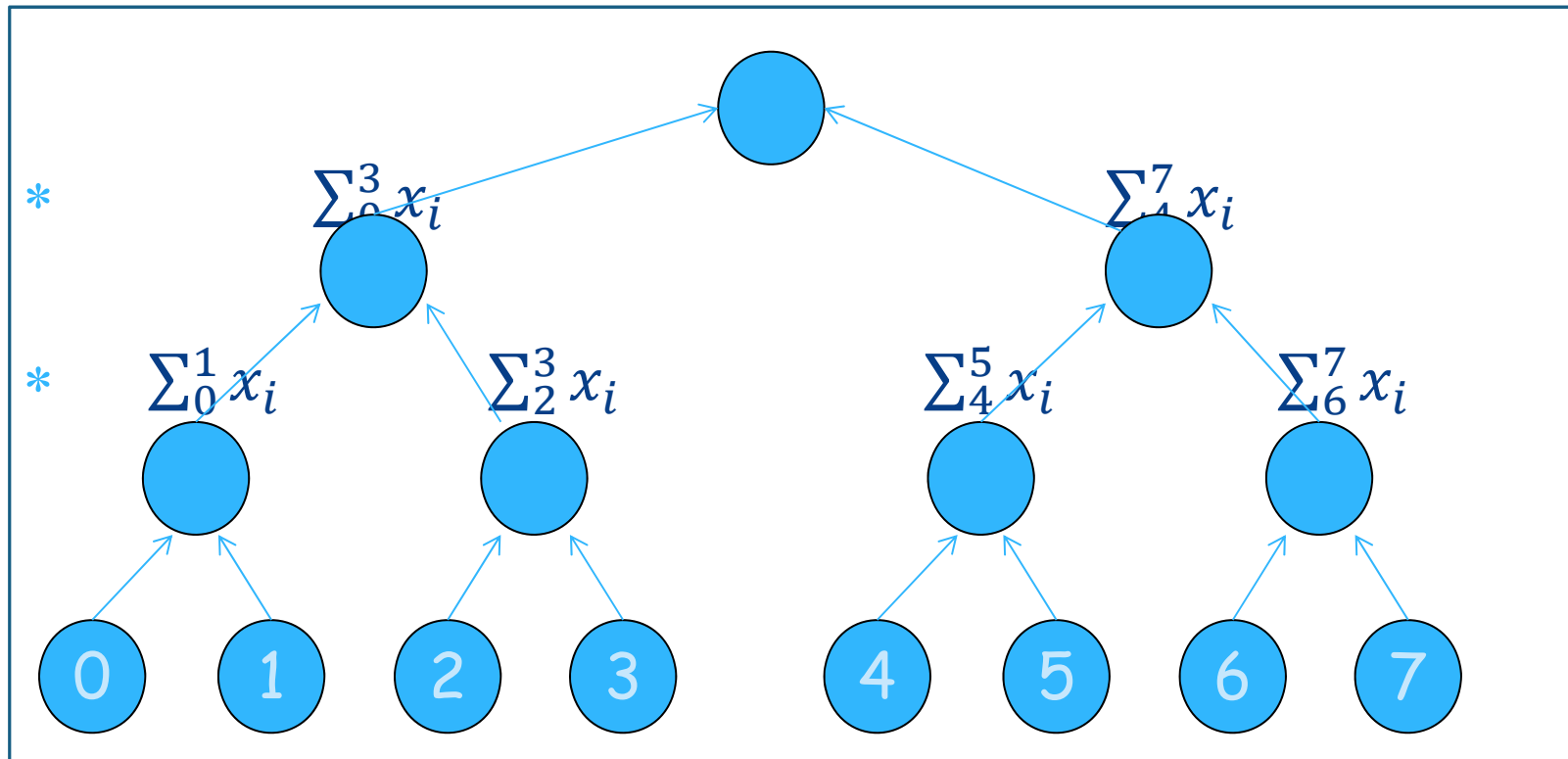
$$S_i = x_i + S_{i-1}$$



全局通信

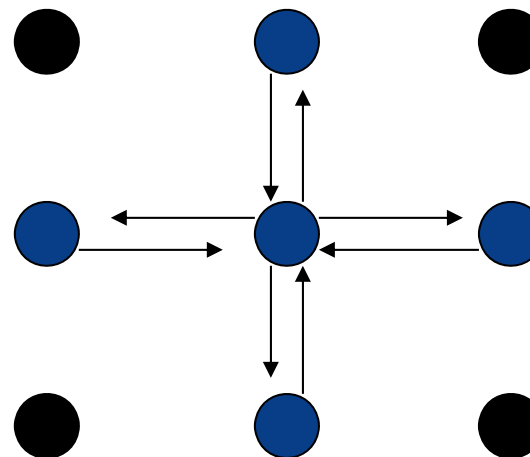
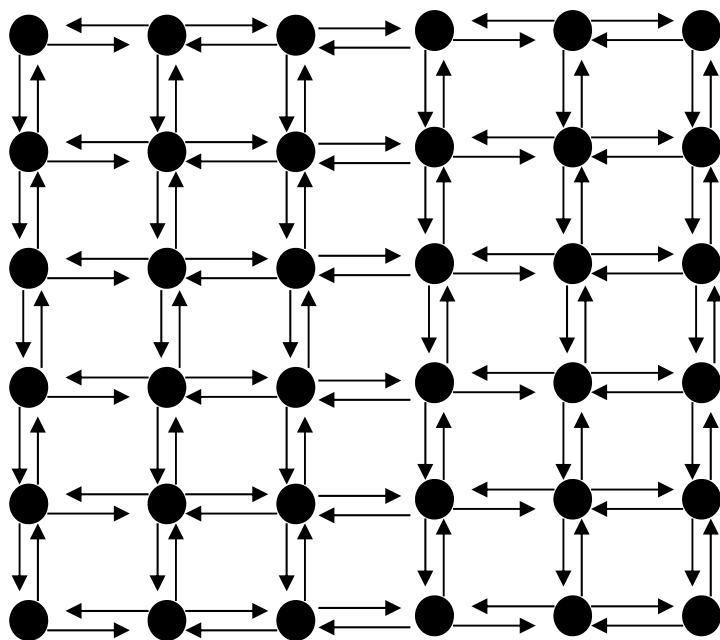
* 分治求和树

$$\sum_0^7 x_i$$



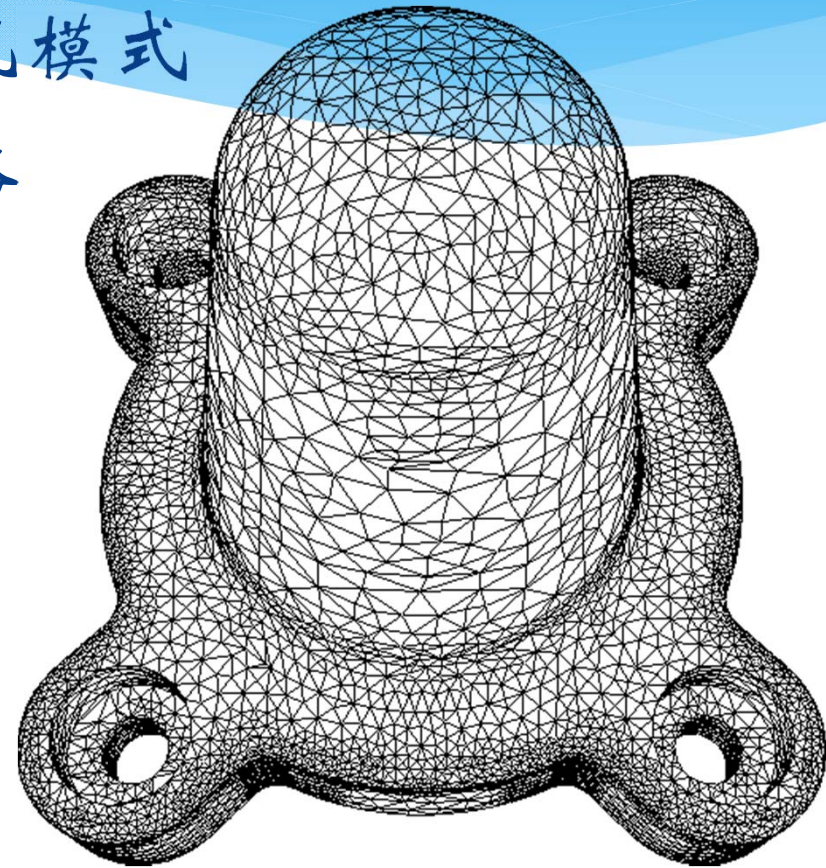
结构化通讯

- * 每个任务的通讯模式是相同的；
- * 下面是否存在一个相同通讯模式？



非结构化通讯

- * 没有一个统一的通讯模式
- * 例如：无结构化网格



7.3 通讯

7.3.1 方法描述

7.3.2 四种通讯模式

7.3.3 通讯判据

通讯判据

- * 所有任务是否执行大致相当的通讯?
- * 是否尽可能的局部通讯?
- * 通讯操作是否能并行执行?
- * 同步任务的计算能否并行执行?

第七章 并行算法的一般设计过程

7.1 PCAM设计方法学

7.2 划分

7.3 通讯

7.4 组合

7.5 映射

7.4 组合

7.4.1 方法描述

7.4.2 表面-容积效应

7.4.3 重复计算

7.4.4 组合判据

方法描述

- * 组合是由抽象到具体的过程，是将组合的任务能在一类并行机上有效的执行；
- * 合并小尺寸任务，减少任务数。如果任务数恰好等于处理器数，则也完成了映射过程；
- * 通过增加任务的粒度和重复计算，可以减少通讯成本；
- * 保持映射和扩展的灵活性，降低软件工程成本；

7.4 组合

7.4.1 方法描述

7.4.2 表面-容积效应

7.4.3 重复计算

7.4.4 组合判据

表面-容积效应

- * 通讯量与任务子集的表面成正比，计算量与任务子集的体积成正比；
- * 增加重复计算有可能减少通讯量；

表面-容积效应

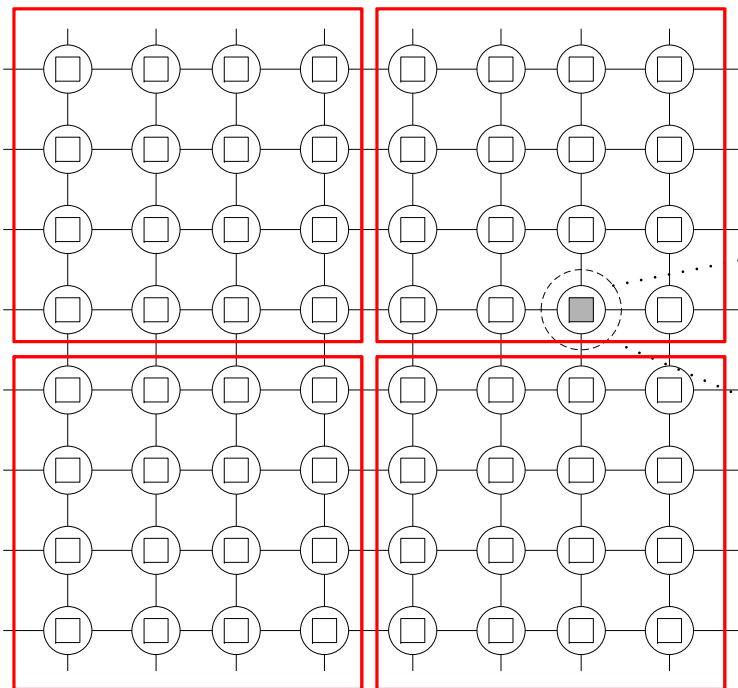
多邻居
↓
内存交换

* 细粒度二维网 (8*8网络)

* $8*8=64$ 个任务

* $64*4=256$ 次通信

* 每个任务通信4次，共传输
256个数据

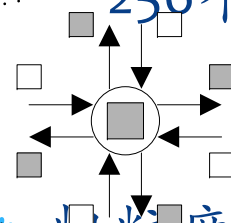


* 粗粒度二维网

* $2*2=4$ 个任务

* $4*4=16$ 次通信

* 每个任务通信4次，共传输16
个数据



7.4 组合

7.4.1 方法描述

7.4.2 表面-容积效应

7.4.3 重复计算

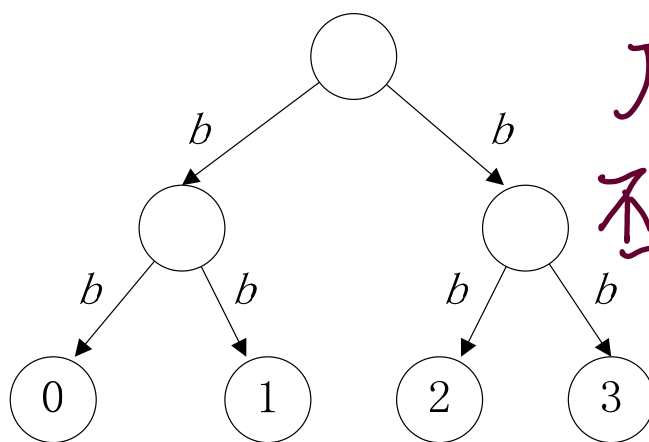
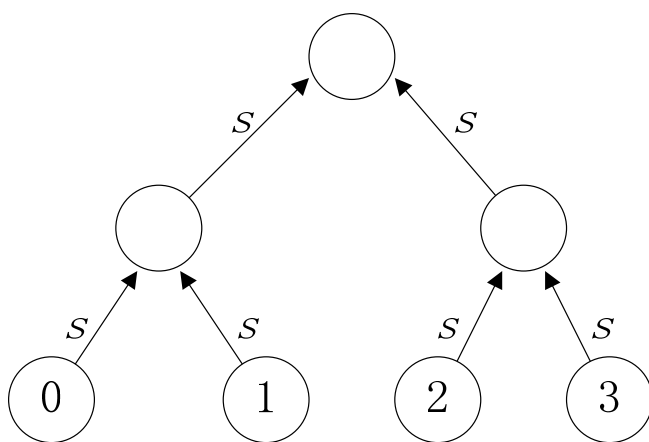
7.4.4 组合判据

重复计算

- * 重复计算减少通讯量，但增加了计算量，应保持恰当的平衡；
- * 重复计算的目标应减少算法的总运算时间；

重复计算

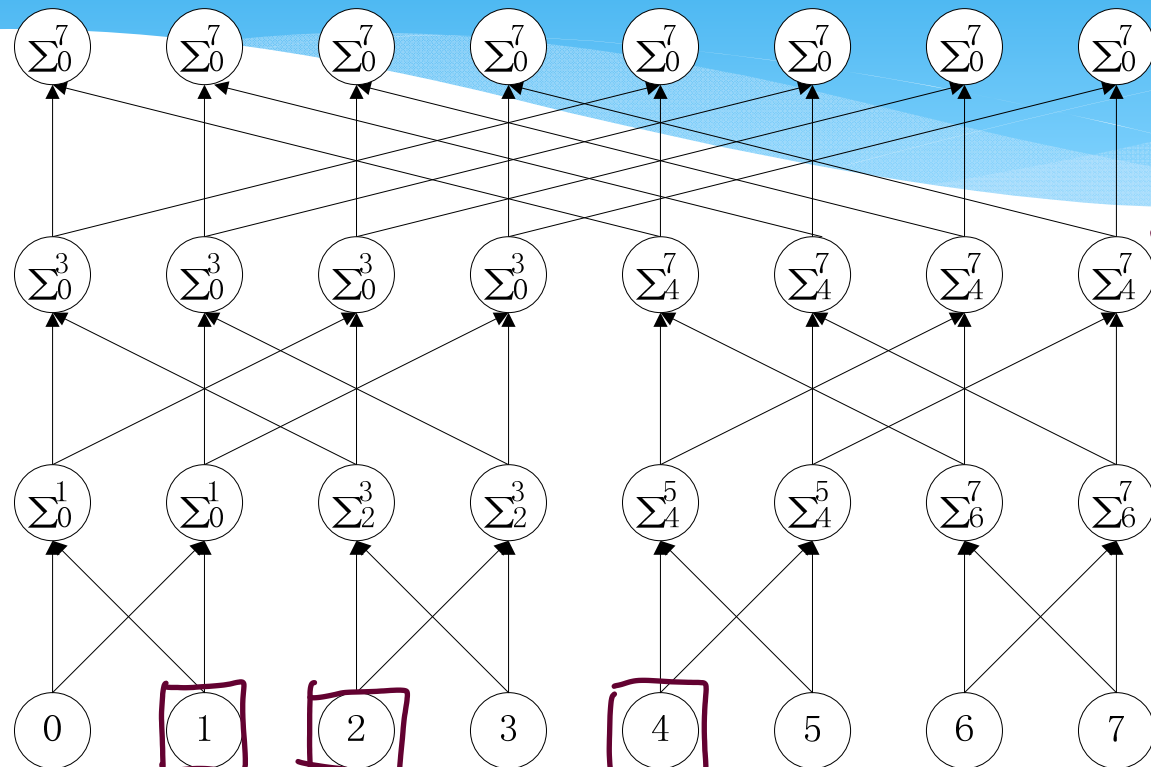
* 示例：二叉树上N个处理器求N个数的全和，要求每个处理器均保持全和。



广播?
不一定存在线性
局部.

二叉树上求和，共需 $2\log N$ 步

重复计算



蝶式结构求和，使用了重复计算，共需 $\log N$ 步
以总共施行 $O(N \log N)$ 次计算与通信操作为代价

7.4 组合

7.4.1 方法描述

7.4.2 表面-容积效应

7.4.3 重复计算

7.4.4 组合判据

组合判据

- * 增加粒度是否减少了通讯成本？
- * 重复计算是否已权衡了其得益？
- * 是否保持了灵活性和可扩放性？
- * 组合的任务数是否与问题尺寸成比例？
- * 是否保持了类似的计算和通讯？
- * 有没有减少并行执行的机会？

第七章 并行算法的一般设计过程

7.1 PCAM设计方法学

7.2 划分

7.3 通讯

7.4 组合

7.5 映射

7.6 其他

7.5 映射

7.5.1 方法描述

7.5.2 负载均衡算法

7.5.3 任务调度算法

7.5.4 映射判据

方法描述

- * 每个任务要映射到具体的处理器，定位到运行机器上；
- * 任务数大于处理器数时，存在负载平衡和任务调度问题；
- * 映射的目标：减少算法的执行时间
 - * 并发的任务 → 不同的处理器
 - * 任务之间存在高通讯的 → 同一处理器
- * 映射实际是一种权衡，属于NP完全问题；

7.5 映射

7.5.1 方法描述

7.5.2 负载平衡算法

7.5.3 任务调度算法

7.5.4 映射判据

负载均衡算法

- * 静态的：事先确定；
- * 概率的：随机确定；
- * 动态的：执行期间动态负载；
- * 基于域分解的：
 - * 递归对剖
 - * 局部算法
 - * 概率方法
 - * 循环映射

7.5 映射

7.5.1 方法描述

7.5.2 负载均衡算法

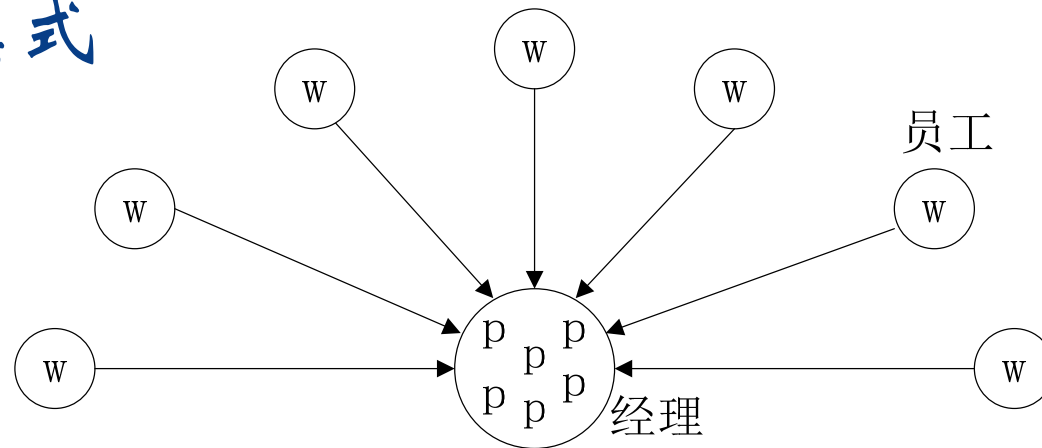
7.5.3 任务调度算法

7.5.4 映射判据

任务调度算法

* 任务放在集中的或分散的任务池中，使用任务调度算法将池中的任务分配给特定的处理器。下面是两种常用调度模式：

* 经理/雇员模式



* 非集中模式

7.5 映射

7.5.1 方法描述

7.5.2 负载均衡算法

7.5.3 任务调度算法

7.5.4 映射判据

映射判据

- * 采用集中式负载均衡方案，是否存在通讯瓶颈？
- * 采用动态负载均衡方案，调度策略的成本如何？

第七章 并行算法的一般设计过程

7.1 PCAM设计方法学

7.2 划分

7.3 通讯

7.4 组合

7.5 映射

7.6 小结

小结

- * 划分

- * 域分解和功能分解

- * 通讯

- * 任务间的数据交换

- * 组合

- * 任务的合并使得算法更有效

- * 映射

- * 将任务分配到处理器，并保持负载平衡