

陈国良. 并行计算: 结构 • 算法 • 编程 [M].
1999.

屈彬

2019 年 7 月 8 日

1 并行计算机系统及其结构模型

1.1 并行计算 (2019 年 7 月 2 日)

1.1.1 并行计算与计算科学

定义 1.1 (应用需求) 应用需求分为三类, 计算密集 (*Compute-Intensive*) 型应用, 如大型科学与工程计算与数值模拟; 数据密集 (*Data-Intensive*) 型应用, 如数字图书馆、数据仓库、数据挖掘和计算可视化等; 网络密集 (*Network-Intensive*) 型应用, 如协同工作、遥控和远程医疗诊断等。

定义 1.2 (高性能计算和通信) *High Performance Computing and Communication, HPCC*。

定义 1.3 (加速战略计算创新) *Accelerated Strategic Computing Initiative, ASCI*。

定义 1.4 (美国能源部三大高性能计算实验室) *Lawrence Livermore, Los Alamos, Sandia*。

1.2 并行计算系统互连 (2019 年 7 月 3 日)

1.2.1 系统互连

定义 1.5 (机群网络分类) 1. 系统域网络: (*System Area Network, SAN*), 连接短距离 ($\leq 25m$) 内的节点。

2. 局域网: (*Local Area Network, LAN*), 连接企事业单位内 (*500m 2km*) 内的节点。
3. 节点内网络: 由处理器总线、局部 (本地) 总线、存储器总线构成。
4. 小型机系统接口: (*Small Computer System Interface, SCSI*) 由 *I/O* 总线、系统总线构成。

定义 1.6 (工作站机群) (*Cluster of Workstations, COW*), 通过 *SAN/LAN* 互连的工作站组。

1.2.2 静态互连网络

定义 1.7 (对剖宽度) (*Bisection Width*): 将网络划分为两等分所必须剪去的最少边数。

1.2.3 动态互连网络

定义 1.8 (总线分类) 常用总线包括 *PCI*、*VME*、*Multibus*、*SBus*、*Microchannel* 和 *IEEE Futurebus*。

1. 本地总线: (*Local Bus*), CPU 板级上的总线。
2. 存储器总线: 存储器板级上的总线, 主要指内存与 CPU 互连的总线。
3. 数据总线: *I/O* 与通信板级上的总线, 例如硬盘与内存之间的总线, 以及网卡。

1.2.4 标准互连网络分类

1. 光纤分布式数据接口 (*Fiber Distributed Data Interface, FDDI*), 采用双向光纤令牌环提供 100 200Mb/s 的数据传输。
2. 快速以太网主流的互连网络, 第三代以太网数据传输速度可达 1Gb/s。
3. **myrinet** 由 Myricom 公司生产的千兆位包开关网。
4. 高性能并行接口 (*High Performance Parallel Interface, HiPPI*), 主要用于构筑异构计算机系统。
5. 异步传输模式 (*Asynchronous Transfer Mode, ATM*), 在光纤通信基础上建立起来的一种新的宽带综合业务数字网 (*B-ISDN*) 的交换技术。
6. 无线带宽 (*InfiniBand*)。

1.3 并行计算机体系结构（2019 年 7 月 4 日）

1.3.1 并行计算机结构模型

大型并行机系统一般可分为六类机器：

1. 单指令多数据流：（Single Instruction Multiple-Data, SIMD）。
2. 并行向量处理机：（Parallel Vector Processor, PVP）。
3. 对称多处理机：（Symmetric Multiprocessor, SMP）。
4. 大规模并行处理机：（Massively Parallel Processor, MPP）。
5. 工作站机群：（Cluster of Workstations, COW）。
6. 分布共享存储多处理机：（Distributed Shared Memory, DSM）。

除 SIMD 外，其余五种皆为 MIMD。

1.3.2 并行计算机访存模型

1. 均匀存储访问：（Uniform Memory Access, UMA），其特点包括：物理存储器被所有处理器均匀共享；所有处理器访问任何存储单元取相同的时间；每台处理器可带私有 cache；外围设备也可以一定形式共享。这种系统由于高度共享资源又被称为紧耦合系统（Tightly Coupled System）。多核同构 CPU 系统的存储模型属于典型的 UMA 模型。
2. 非均匀存储访问：（Nonuniform Memory Access, NUMA）。
3. 全高速缓存存储访问：（Cache-Only Memory Access, COMA）。
4. 高速缓存一致性非均匀存储访问：（Coherent-Cache Nonuniform Memory Access, CC-NUMA）。
5. 非远程存储访问：（No-Remote Memory Access, NORMA）。

1.3.3 并行计算机存储组织

定义 1.9 (层次存储技术) 利用程序局部性，设置多级存储系统。底层的存储器通常具有容量大、速度慢、成本低的特点，高层的存储器通常具有容量小、速度快、成本高的特点。层次存储技术出现的目的在于寻找性能与成本之间的平衡。

定义 1.10 (高速缓存一致性问题) 当处理器将新数据写入高层存储器时, 需保证其他层次中存储器的数据对应的位置具有相同的副本。

2 当代并行计算机系统介绍

跳过第 2 章, 它的内容与第 1 章存在大量重复。

3 并行计算性能评测

3.1 并行计算机的一些基本性能指标 (2019 年 7 月 5 日)

3.1.1 CPU 和存储器的某些基本性能指标

名称	符号	含意	单位
机器规模	n	处理器的数目	
时钟速率	f	时钟周期长度的倒数	MHz
工作负载	W	计算操作的数目	Mflop
顺序执行时间	T_1	程序在单处理机上的运行时间	s (秒)
并行执行时间	T_n	程序在并行机上的运行时间	s (秒)
速度	$R_n = W/T_n$	每秒百万次浮点运算	Mflops
加速	$S_n = T_1/T_n$	衡量并行机有多快	
效率	$E_n = S_n/n$	衡量处理器的利用率	
峰值速度	$R_{peak} = nR'_{peak}$	所有处理器峰值速度之积	Mflops
顺序峰值速度	R'_{peak}	单个处理器的峰值速度	Mflops
利用率	$U = R_n/R_{peak}$	可达速度与峰值速度之比	
通信延迟	t_0	传送 0-字节或单字的时间	μs
渐进带宽	r_{inf}	传送长消息的通信速率	MD/s

3.1.2 通信开销测量方法

1. 乒-乓方法: (Ping-Pong Scheme) 适用于一对节点之间通信开销的测量。
2. 热土豆法: (Hot-Potato), 也称作救火队法 (Fire-Brigade), 适用于测量多个节点 (两个以上) 以环形方式通信, 数据包传递一圈的通信开销。

理论点到点通信开销 $t(m)$ 是消息长度 m (字节) 的线性函数:

$$t(m) = t_0 + m/r_{inf}$$

其中 t_0 是通信启动时间 (μs), r_{inf} 是渐进带宽 (MB/s)。

3.1.3 机器的成本、价格与性能/价格比

内容比较零碎, 跳过。

3.2 加速比性能定律 (2019 年 7 月 6 日)

3.2.1 Amdahl 定律

3.2.2 Gustafson 定律

3.2.3 Sun 和 Ni 定律

4 并行算法的设计基础

出于 PAC 比赛的需求, 暂时先跳过第三章。

4.1 并行算法的基础知识

4.2 并行计算模型 (2019 年 7 月 7 日)

并行计算模型是硬件和软件之间的一种桥梁 (大雾, 什么叫作 “桥梁”), 使用它能够设计分析算法, 在其上高级语言能被有效地编译且能够用硬件实现。(我觉得作者没有把并行计算模型的用途讲清楚)

4.2.1 PRAM 模型

即并行随机存储机器 (Parallel Random Access Machine, PRAM), 是一种抽象的并行计算同步模型。在这种模型中, 假定存在着一个容量无限大的共享存储器; 有多个同构处理器; 在任何时刻各处理器均可通过共享存储单元相互交换数据以实现同步。根据对处理器读写时序的限制, 又可分为:

1. 不允许同时读和同时写 (Exclusive-Read and Exclusive-Write) 的 PRAM 模型, 记作 **PRAM-EREW**。
2. 允许同时读不允许同时写 (Concurrent-Read and Exclusive-Write) 的 PRAM 模型, 记作 **PRAM-CREW**。
3. 允许同时读和同时写 (Concurrent-Read and Concurrent-Write) 的 PRAM 模型, 记作 **PRAM-CRCW**。

不过，允许同时写显然是不现实的，所以对 PRAM-CRCW 模型做了进一步的约定：

1. **CPRAM-CRCW**：公共的 PRAM-CRCW，只允许所有的处理器同时写相同的数。
2. **PPRAM-CRCW**：优先的 PRAM-CRCW，只允许最优先的处理器先写。
3. **APRAM-CRCW**：任意的 PRAM-CRCW，允许任意处理器自由写。

4.2.2 异步 PRAM 模型

记作 APRAM，它的特点是分相（Phase），每个处理器都有其私有存储（书里称作“局存”）、私有时钟（书里称作“局部时钟”）和局部程序；处理器之间通过共享全局存储器进行通信；无全局时钟，各处理器异步地独立执行各自的指令；若处理器之间存在任何时序依赖关系（书中称作“时间依赖关系”），须通过插入同步路障（Synchronization Barrier）以保持依赖源和目标的执行顺序。

APRAM 模型具有四种访存指令：

1. **全局读**：将全局存储单元中的内容读入私有存储单元中。
2. **局部操作**：对局存中的数执行操作，其结果是存入局存中。
3. **全局写**：将局存单元中的内容写入全局存储单元中。
4. **同步**：同步是计算中的一个逻辑点，在该点各处理器均需等待别的处理器到达后才能继续执行其局部程序。

4.2.3 BSP 模型

整体同步计算模型（Bulk Synchronous Parallel），字面含义是“大”同步模型，该模型使用了三个属性描述：模块（Components）、选路器（Router）和同步路障器执行时间 L （中文又称“超级步”）。BSP 是一种 MIMD-DM（DM：distributed memory）模型。

在一个超级步中，每个处理器执行各自的局部计算，通过选路器接收和发送消息；然后作一全局检查（同步），以确定当前超级步是否已由所有的处理器完成；然后前进到下一个超级步。

在分析 BSP 模型的性能时，假定局部操作可在一个超级步内完成。在每一超级步中，一个处理器至多发送或接收 h 条消息（称为 **h-relation**）。假定 s

是传输建立时间， g 定义为每秒处理器所能完成的局部计算数目与每秒选路器所能传输的数据量之比，那么传送 h 条消息的时间开销为 $gh + s$ ，若 $gh \geq 2s$ ，则超级步步长 $L \geq gh$ （为什么？）。硬件可设置 L 的下限（例如宽的通信带宽减小 g ），而软件可设置 L 的上限（例如并行粒度越大， L 越大）。

4.2.4 logP 模型