

大规模科学计算

陈俊清

jqchen@math.tsinghua.edu.cn

清华大学数学科学系

March 1, 2013

- 大规模科学计算简介
- Linux编程基础
- 消息传递编程接口MPI
- 程序性能评价与优化
- 并行算法设计与实现
- 自适应有限元方法
- PHG简介
-

- 并行计算导论, 张林波等, 清华大学出版社, 2006
- Introduction to Parallel Computing(并行计算导论), 2nd, Ananth Grama et al.机械工业出版社,2003
- 并行计算综论(Sourcebook of parallel Computing), Jack Dongarra等, 莫则尧等译, 电子工业出版社, 2005
- 数值并行算法与软件, 李晓梅, 吴建平, 科学出版社, 2007
- 并行程序设计导论, Peter S. Pacheco, 机械工业出版社, 2011
- ...

- 先修课程： 数值分析， 数值线性代数， 偏微分方程数值解
- 编程语言： C语言
- 考核方式：
 - 平时作业 50%
 - 期末报告 50%

特别说明： 请大家按时完成作业和报告！

Outline

- 1 大规模科学计算简介
- 2 高性能并行计算机简介
- 3 自适应计算
- 4 并行编程模式
- 5 操作系统与并行编程环境
- 6 并行算法
- 7 Linux操作系统
- 8 Linux基本操作
- 9 实习与思考

Outline

- 1 大规模科学计算简介
- 2 高性能并行计算机简介
- 3 自适应计算
- 4 并行编程模式
- 5 操作系统与并行编程环境
- 6 并行算法
- 7 Linux操作系统
- 8 Linux基本操作
- 9 实习与思考

什么是大规模科学计算

Large-Scale Scientific Computing: 没有明确的定义，但是有一些特点，比如空间上全三维，全物理过程，高分辨率， 总之对计算资源的需求巨大。催生了另外一门学科**并行计算**的发展。

- 名称:并行计算, Parallel Computing(concurrent computing)
- 定义: 通过网络相互联接的两个以上的处理机相互协调, 同时计算同一个任务的不同部分, 从而提高问题求解速度, 或者求解单机无法求解的大规模问题。

并行计算的要点

- 通过网络联接的两个以上的处理机；
- 处理机之间相互协调通信，同时执行同一个任务的不同部分；
- 任务可以分解为被并行执行的多个子任务；
- 两个目的：提高问题求解速度，或者提高求解问题的规模；
- 意义：解决一批具有重大挑战性计算任务的科学与工程计算问题，满足不断增长的应用问题对速度和内存需求；

两个方面的认识

- 单机性能不可能满足大规模科学与工程计算需求（例如Power4/1.5GHz的 峰值性能仅为60亿次/秒，目前最快的单机不超过千亿次/s），并行计算是解决挑战性问题的唯一途径；
- 同时性与并行性是物质世界的一种普遍属性，具有实际应用物理背景的计算问题在许多情况下都可以分解为能并行计算的 多个子任务。

Distributed Computing, 通过网络相互联接的两个以上的处理机相互协调, 以**Client-Server**方式, 同时执行不同的任务, 但这些任务之间存在相互依赖关系, 从而达到协调资源访问, 提高资源访问效率的目的, 但无法提高同一个任务的求解速度。复杂的应用系统需要并行与分布式计算互相配合协调。

例子

- 例1.1（并行计算） P台处理机完成N个数相加；
- 例1.2（并行计算） 二维规则区域上的Dirichlet问题 $-\Delta u = f$ ，5点格式有限差分离散， $N_x * N_y$ 个网格单元， $P_x * P_y$ 台处理机，Jacobi迭代(Gauss-Seidel迭代?)，提高计算速度，求解单机无法求解的问题；
- 例1.3（分布式计算）观众点播，远程驾驭式可视化，电视会议等。

并行计算的研究内容

这是一个交叉学科：大规模应用 + 计算数学（物理） + 并行计算机。

- 并行计算机的高性能特征抽取，理解和抽取当前并行计算机体系结构的高性能特征，提出实用的并行计算模型和并行性能评价方法。
- 并行算法设计与分析，针对具体应用的离散方法，设计高效的并行算法
- 并行实现技术，主要包含并行程序设计和并行性能优化。并行编程环境：消息传递平台MPI，共享存储平台OpenMP
- 并行应用，这是并行计算的最终目的。

并行计算发展的两个主要动力

- 大规模科学与工程计算应用需求;
- 微电子技术与大規模集成电路 (VLSI) 的发展:
 - Moore定理: 微处理器速度每3年翻4倍, 内存容量每年翻3-4倍, 该定理目前还在延续;
 - 互联网络技术飞速发展.

大规模科学计算对并行计算的需求是无止境的

- 全球气象预报中期天气预报模式：网格点635万，内存>1TB，总计算量需要25万亿次浮点 运算/秒，要求在24小时内完成48小时天气预报；
- 核武器数值模拟：ASCI计划，万亿次、十万亿次、30万亿次、100万亿次，4个阶段；
- 国家安全；
- 石油地震勘探；
- 生物信息处理；
- 新药研制；
- 直接湍流模拟；
- 基础理论研究；
- ...

- 工业制造;
- 燃料燃烧;
- 大规模数据管理;
- 网络游戏
- ...

- 以太网 (Ethernet) ,10Mbps,100Mbps,1Gbps
- 商用交换机 (Switch) , 可对 $N/2$ 对端口提供100MB/s的直接连接通路, N 为交换机端口数。
- 专用微机机群互联网络
 - Myrinet,点对点带宽 250MB/s
 - Quadrics,点对点带宽 400MB/s
 - InfiniBand, 点对点带宽 1.25GB/s

高性能计算与通信计划 (HPCC: High performance Computing and Communication)

- 确保美国在高性能计算领域的绝对领先地位;
- 1983, 配合星球大战战略防御计算的战略计算机计划 (SCP), 10亿/s的CRAY计算机
- 1993年, 美国科学、工程、技术联邦协调委员会向国会提交了题为“重大挑战性项目: 高性能计算与通信 (HPCC)”计划, 目的是研制能够提供3T性能目标 (1Tflops计算能力、1TB内存容量和1TB/s的I/O带宽, $1T=10^{12}$) 的高性能并行机, 解决科学与工程计算中的重大挑战性课题, 其中包括新药设计、磁记录技术、高速民航、催化作用、燃料燃烧、海洋建模、臭氧耗损、数字解析、大气污染、蛋白质结构设计、图像理解、密码破译等。
- 世界上第一台峰值速度超过1Tflops的高性能计算机是由Intel公司于1996年12月研制成功的。

加速战略计算创新 (ASCI: Accelerated Strategic Computing Initiative) 计划:

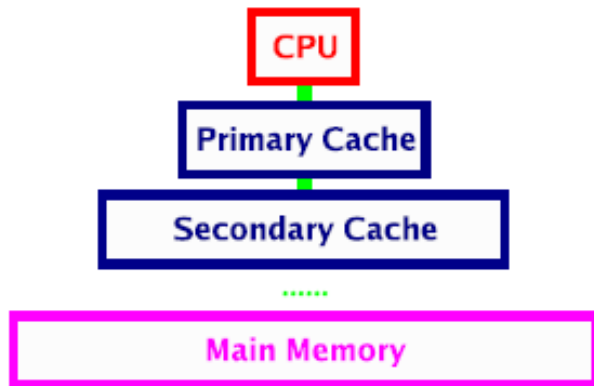
- 核武器库存数值模拟, 在全面核禁试条约签定后, 继续保持美国的核威慑力量
- 1996年6月, 美国能源部联合美国三大核武器实验室 (Los Alamos国家实验室、Lawrence Livermore国家实验室和Sandia国家实验室) 共同提出了“加速战略计算创新 (ASCI) 计划”, 提出通过数值模拟评估核武器的性能、安全性、可靠性、更新等, 要求数值模拟达到高分辨率、高逼真度、三维、全物理、全系统的规模和能力。
- 三大实验室分别向美国三大公司 (Intel、IBM和SGI) 预定了峰值速度超过1Tflops (百万亿) 的并行机, 计划分四个阶段, 分别实现万亿次、10万亿次、30万亿次和100万亿次的高性能并行机。目前, 这四个阶段已完全实现, 并出现了千万亿次 (万万亿次) 的高性能并行机。

- 欧洲 万亿次计算计划
- 日本 真实世界计算计划, 2002 Earth Simulator
- 我国在针对ASCI计划、HPCC计划也采用相应的高性能计算对策（863计划），极大地鼓舞了国内并行计算的发展

Outline

- 1 大规模科学计算简介
- 2 高性能并行计算机简介
- 3 自适应计算
- 4 并行编程模式
- 5 操作系统与并行编程环境
- 6 并行算法
- 7 Linux操作系统
- 8 Linux基本操作
- 9 实习与思考

微处理器的结构



Cache结构对程序性能的影响

例1.4,矩阵乘法中不同循环对程序性能的影响

$$c_{ij} = \sum_{k=1}^n a_{ik} b_{kj}, i, j = 1, 2, \dots, n$$

```
DO J=1,N
  DO I=1,N
    C(I,J)=0.0
  ENDDO
ENDDO
DO I=1,N
  DO J=1,N
    DO K=1,N
      C(I,J)=C(I,J)+A(I,K)*B(K,J)
    ENDDO
  ENDDO
ENDDO
```

并行计算机体系结构

- 结点(node):每个节点由多个处理器构成, 可以直接输入输出(I/O)
- 互联网络(interconnect network), 所有结点通过互联网络相互连接相互通信
- 内存(memory): 或在节点内, 或与节点对称地分布在网络两侧

并行计算机的规模: 结点总数或CPU总数。

并行计算机分类

并行计算机的分类是随并行计算机体系的发展而发展的。按经典指令与数据流（Flynn法）分

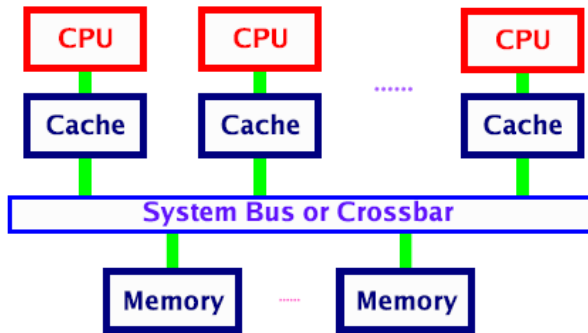
- SISD，单指令单数据流，单处理器
- SIMD，单指令多数据流，按同一条指令计算机的不同功能部件处理不同的数据，已退出历史舞台
- MIMD，不同的处理器可同时对不同的数据执行不同的指令，目前所有并行机都属这一类
- MISD，至今没有出现

按内存分布与访问模型不同，MIMD又可分为

- 对称多处理器或共享存储并行计算机(SMP)
- 分布共享内存并行计算机(DSM)
- 分布存储并行计算机 (DM)
 - 大规模并行机(MPP)
 - 机群(Cluster)
 - 星群(Constellation)

共享内存SMP型并行计算机

- 对称多处理器（Symmetric Multi-Processors），或共享内存处理器（Shared Memory Processors）（90's）
- 多个处理器通过系统总线或交叉开关共享一个或多个内存模块
- 优点：使用简单，维护方便
- 缺点：受系统总线带宽限制，只能支持少量处理器（一般十几个）
- 并行编程方式：通常采用OpenMP，也可使用消息传递（MPI/PVM）及HPF
- 代表机型：SGI Power Challenge, Sun E10000等



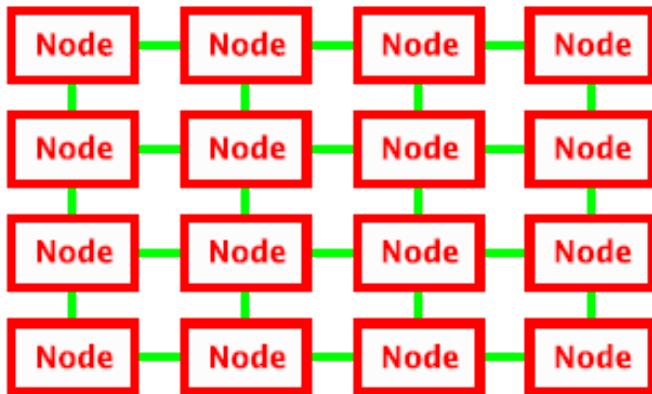
分布式内存DSM型并行计算机

分布共享内存: Distributed Shared Memory

- 多个物理上具有独立内存的处理单元, 通过高速网络联接在一起
- 逻辑上作为共享内存并行机使用
- 也称为NUMA结构 (NonUniform Memory Access)
- 不同处理单元间内存的共享通过特殊的硬件/软件实现
- 具有比SMP型并行机有更好的可扩展性 (超过100个CPU)
- 代表机型: SGI Origin 2000/3000,
CRAY-T3E(2048cpu, 2.5Tflops)

大规模并行机系统 (MPP)

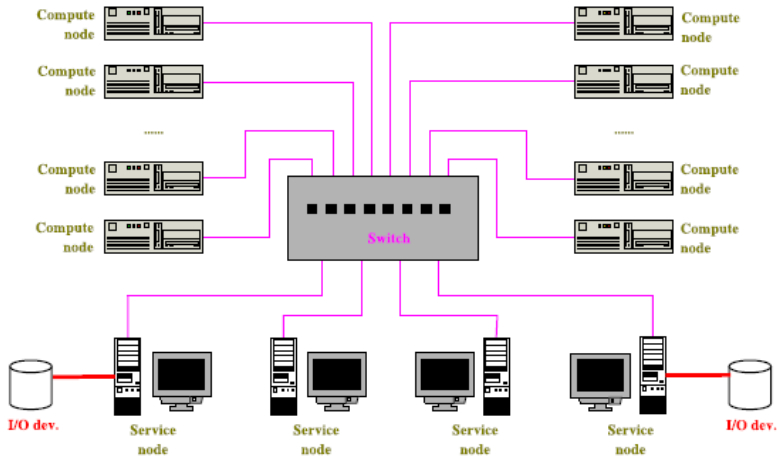
- Massively Parallel Processors的简称(2000s)
- 指由大量具有局部内存的计算节点通过高速系统网络联接而构成的并行计算机系统
- MPP 系统的系统网络通常具有某种拓扑结构 (如Tree, mesh, torus, hypercube)
- IBM Blue Gene/L, 天河1A



- 微机机群（PC cluster，又称为Beowulf cluster，2000年以来），工作站机群（NOW，Network Of Workstation，90年代中后期），将联网的多台微机或工作站组织成一台并行计算机，目前常用的网络有以太网,Myrinet,Quadrics,infiniband等
- 适合于构造中等规模的并行系统（多达数百个处理器）
- 根据机群中所使用的机型可分为同构型和异构型两种
- 根据机群的使用方式又可分为专用型和兼用型，前者指专门用于并行计算
- 微机／工作站机群的优点：价格便宜，配置灵活。缺点：规模和并行效率受网络设备的制约

配以适当的系统管理工具及作业调度、性能监控、并行程序调试开发环境等，可以达到与商用MPP系统一样的使用效果

微机／工作站机群



如果将上述Beowulf结构中的每个节点替换为一个SMP系统，便称这种系统为SMP集群或星群（constellation）

- 将多台SMP并行机通过互联网络连接而成
- 目前国内外最高性能的并行机大多是这种结构

兼具SMP与DM两种结构的共同优点，可以通过消息传递和共享存储两种模式编程，灵活性强且成本较低。

超级计算机排行榜

从1993年开始，每年分别在6月和11月公布当时全世界最快的超级计算机前排名，给出计算机的Linpack测试峰值，安装地，生产商等信息。

www.top500.org

- 目前世界上最快的是美国能源部位于橡树岭国家实验室的 Titan - Cray XK7，有560640个核心，710144 GB内存， Linpack峰值性能17590.0 TFlop/s（1.76万万亿次/秒）。
- 国防科大研制的Tianhe-1A排名第8，共有186368个核心,229376 Gb内存，峰值性能达2.56Petaflop/s（2千万亿次/秒）
- 80%左右为Cluster架构，20%左右为MPP架构，其它架构非常少。
- 20%教育， > 50% 工业应用，20%科研，其余为政府部门和生产厂商
- 芯片：Intel，IBM，AMD。

我国也在2002年开始了自己的高性能计算机排行榜

www.samss.org.cn/sites/shuxue/pcgg.jsp

fs,FLOPS:floating point operations per second

超级计算机排行榜



PRESENTED BY
UNIVERSITY OF
MANNHEIM

ICL
INNOVATIVE
COMPUTING LABORATORY
at UNIVERSITY OF TORONTO



Lawrence Berkeley
National Laboratory

FIND OUT MORE AT
www.top500.org

	NAME	SPECS	SITE	COUNTRY	CORES	RMAX PFLOP/s	POWER MW
1	TITAN	Cray XK7, Operon 6274 16C 2.2 GHz + Nvidia Kepler GPU, Custom interconnect	DOE/OS/ORNL	USA	560,640	17.6	8.3
2	SEQUOIA	IBM BlueGene/Q, Power BQC 16C 1.60 GHz, Custom interconnect	DOE/NNSA/LLNL	USA	1,572,864	16.3	7.9
3	K COMPUTER	Fujitsu SPARC64 VIIIfx 2.0GHz, Custom interconnect	RIKEN AICS	Japan	705,024	10.5	12.7
4	MIRA	IBM BlueGene/Q, Power BQC 16C 1.60 GHz, Custom interconnect	DOE/OS/ANL	USA	786,432	8.16	3.95
5	JUQUEEN	IBM BlueGene/Q, Power BQC 16C 1.60 GHz, Custom interconnect	Forschungszentrum Jülich	Germany	393,216	4.14	1.97

Figure: 2012年11月top500前5名

千万亿此计算: Petaflops

2005年世界各国推出每秒千万亿次规模计算系统的研发计划, 目标2010年左右实现Petaflops计算。目前美国 (CRAY XT5" Jaguar", IBM BladeCenter" RoadRunner", IBM Blue Gene/P), 中国 (TianHe-1, TianHe-1A) 已实现这一计划, 日本也于2011年研制出自己的Petaflops计算机。

并且出现了万万亿次规模的计算系统: 美国 (Cray XK7, BlueGene/Q), 日本 (K computer)

Outline

- 1 大规模科学计算简介
- 2 高性能并行计算机简介
- 3 自适应计算**
- 4 并行编程模式
- 5 操作系统与并行编程环境
- 6 并行算法
- 7 Linux操作系统
- 8 Linux基本操作
- 9 实习与思考

千万亿此计算的挑战

2009年10月29日，我国首台千万亿次超级计算机系统“天河一号”在国防科技大学建成，使得我国成为世界上第二个掌握千万亿次超级计算机系统技术的国家。

- >10万个处理核心，每个核心处理100万个自由度（未知量）⇒可以处理千万亿个未知量。
- 全球系统模式：地球表面积约5亿（ 5×10^8 ）平方公里，每平方公里1个网格单元，垂直于地球方向分为200层计算，需1000亿个网格
- 大气环流模式：大气速度，压力，温度和湿度。⇒7000亿自由度
- 1秒的时间步长，100—1000年的时间跨度⇒超过千万亿次计算机的能力！

任何计算机能处理的问题的规模总是有限的。在有限能力的基础上发挥算法的作用：自适应方法，让计算机自动实现自由度的最优分配。

- 为保证计算精度，计算机所能处理的自由度不需要均匀分布在计算区域上，比如天气预报，可以在台风，暴雨和我们所关心的地方分布更多的自由度，仍然保证计算结果的可靠性。
- 能带来差不多两三个数量级的计算能力提高。
- 并行自适应有限元平台PHG， Deal.II等。

Outline

- 1 大规模科学计算简介
- 2 高性能并行计算机简介
- 3 自适应计算
- 4 并行编程模式**
- 5 操作系统与并行编程环境
- 6 并行算法
- 7 Linux操作系统
- 8 Linux基本操作
- 9 实习与思考

自动并行与手工并行

- 在SMP及DSM并行机上编译系统通常具有一定的对用户程序（C/FORTRAN）进行自动并行化的功能，但常常需要人工干预（通过指导语句、命令行选项等），以达到理想的并行效率，并行主要针对循环并行（细粒度并行）
- 在分布式内存并行机上尚无通用高效的自动并行工具，主要依靠人工编写并行程序。
- 并行算法的设计及并行程序的编制已成为目前制约大规模并行计算机应用的主要障碍。

在串行程程序的循环语句前插入特定的指导语句，告诉编译系统一些有助于对该循环进行并行的信息以及／或是强制编译系统按指定的方式将该循环并行化

- 主要限于SMP及 DSM型的并行机，现在也已发展到一些MPP系统
- 通常结合编译系统的自动并行化功能使用
- 也有一些自动并行化工具，他们对程序结构和数据流进行分析，然后插入适当的OpenMP语句
- OpenMP的优点是学习及使用简单，很多情况下不需要对原有算法及程序作大的修改，缺点是只适合一类特定的机型，并且程序的可扩展性通常不如消息传递方式编写的并程序
- 对一些具有强数据相关性的计算过程需要通过改变计算顺序或修改算法甚至设计新的算法来实现并行计算

- 基于数据并行的编程语言，即用户通过指导语句指示编译系统将数据分布到各处理器上，编译系统根据数据分布的情况生成并行程序
- 主要面向Fortran 90/95
- 优点：编程相对简单，串并行程序一致
- 适合于SMP/DSM/MPP/cluster
- 主要缺点是并行过程对用户的透明度不高，程序的性能很大程度上依赖于所用的编译系统及用户对编译系统的了解，需要针对不同的HPF编译器做不同的优化，影响了程序的可移植性。

消息传递并行编程模式

- 并行程序由一组独立运行的进程构成，进程间通过相互发送消息来实现数据交换
- 可以说消息传递并行编程是并行应用程序开发的最底层编程方式之一。很多其它并行开发语言或工具（如一些HPF编译器）将程序化成消息传递型并行程序来实现并行。
- 常用的消息传递平台有PVM（Parallel Virtual Machine）和MPI（Message Passing Interface）
- 程序通用性好，用PVM或MPI编写的程序可以在任何并行计算机上运行
- 能够达到很高的并行效率，具有很好的可扩展性
- 缺点：程序的编制与调试比较困难，许多情况要对程序甚至算法作大的改动。

Outline

- 1 大规模科学计算简介
- 2 高性能并行计算机简介
- 3 自适应计算
- 4 并行编程模式
- 5 操作系统与并行编程环境**
- 6 并行算法
- 7 Linux操作系统
- 8 Linux基本操作
- 9 实习与思考

操作系统和编程语言

- UNIX操作系统几乎是当前所有高性能并行计算机采用的标准操作系统，其中包括SGI的IRIX，IBM的AIX，SUN的Solaris和自由软件Linux等。
- 编程语言：SMP，DSM和MPP 一般提供 Fortran 77/90, C/C++, HPF等，机群系统一般都有免费的GNU Fortran 77/95, GNU C/C++等

进程可表示成四元组 (P, C, D, S) :P程序代码, C进程的控制状态, D是进程的数据, S是进程的执行状态。是操作系统资源调度的基本单位。

任何进程总是和程序联系在一起, 程序一旦在操作系统中运行就变成了进程。

各进程具有独立的执行环境, 且各个进程不能直接访问其他进程的局部内存空间

- 资源特征。计算资源: 程序代码、内存空间等等
- 执行特征。进程执行中的动态改变特征, 比如执行指令序列, 进程的控制等

即使同一个程序, 不同的执行也将产生不同的进程。

多个进程可以在同一台处理机中（或不同的处理机中），拥有独立的执行环境，在操作系统的调度下，分时共享计算机资源，但内存指令和数据地址空间必须互不相交。

- 通信：进程间的数据传递
- 同步：是位于相同或不同的处理机中不同的多个进程间相互等待的操作（通信的一种方式）。
- 聚集（归约）：将位于相同或不同处理机上的多个进程的局部结果综合起来，产生一个新的结果。

消息：进程间通信所操作的数据；消息传递：具体操作。

进程间通信的实现

- 在共享存储环境中，通过读／写操作系统提过的共享数据缓存区来实现
- 在分布式存储网络环境中，通过网络通信来实现。

但是具体实现对编程用户来说都是屏蔽的，我们能接触的就是一堆API接口。

进程具有独立的局部内存空间，使得操作系统对它的管理非常费时，影响并行计算的性能发挥，不适合细粒度共享存储并行程序设计。为在共享存储环境下有效开发应用程序的细粒度并行度，将一个进程分解成两部分，其中一部分由其资源特征构成，仍称为进程，另一部分由其执行特征构成，称为线程，或称为轻量级进程。

特点：

- 同一个进程中可以有多个线程
- 多个线程并行执行
- 多个线程共享同一个进程的资源

多线程编程：OpenMP, pthreads等

Outline

- 1 大规模科学计算简介
- 2 高性能并行计算机简介
- 3 自适应计算
- 4 并行编程模式
- 5 操作系统与并行编程环境
- 6 并行算法**
- 7 Linux操作系统
- 8 Linux基本操作
- 9 实习与思考

并行算法

并行算法是适合在并行计算机上实现的算法，一个好的并行算法应该具有充分发挥并行计算机潜在性能的能力。

例如：给定一个描述实际问题的偏微分方程，采用一定的格式数值离散后，剩下的就是设计合适的并行算法，编制并行程序，在并行计算机上并行求解

并行算法分类

根据运算基本对象的不同分为

- 数值并行算法：主要是为数值计算方法而设计的并行算法，矩阵的并行算法，FFT等
- 非数值并行算法：为符号运算而设计的并行算法，如图论算法，遗传算法等

并行进程间相互执行的顺序

- 同步并行算法，进程间由于运算执行顺序而需要相互等待的并行算法
- 异步并行算法：进程间相互独立，不需要相互等待的算法，根据最新消息决定进程的继续与终止。
- 独立并行算法：进程间完全独立，不需相互通信。

根据各进程计算粒度的不同

- 细粒度：通常指基于向量和循环级的并行
- 中粒度：通常指基于较大的循环级的并行
- 大粒度：通常指基于子任务级的并行，比如区域分解算法

并行算法的粒度是一个相对的概念，如果处理器的计算功能强大，则原来的粗粒度的计算也可认为是细粒度的。

并行算法的发展阶段

- 基于向量运算的并行算法设计阶段（70—80，向量机，Cray-1，YH-1）
- 基于多向量处理机的并行算法设计阶段（YH-2）
- SIMD类并行机上并行算法设计阶段（80s）
- MIMD类并行机上并行算法设计阶段（80s—，区域分解算法）
- 现代并行算法设计阶段：以MIMD为主流，并要求具有可扩展性和可移植性。

Outline

- 1 大规模科学计算简介
- 2 高性能并行计算机简介
- 3 自适应计算
- 4 并行编程模式
- 5 操作系统与并行编程环境
- 6 并行算法
- 7 Linux操作系统**
- 8 Linux基本操作
- 9 实习与思考

Linux操作系统

当前流行的计算机主要有节点上安装Linux操作系统的Cluster构成，所以需要大家了解一些基本的Linux操作系统的使用。

- Linux操作系统可以认为是一个小型的、适合于个人计算机的UNIX操作系统。
- 准备一套Linux安装盘，随便哪一个发行版都可以，Fedora，Ubuntu，Debian，OpenSuse等。
- 电脑上有足够的磁盘空间，如果已经安装了Windows操作系统，可以通过删除系统的某个分区或是通过一些工具，如PartitionMagic, AcronisDiskDirector等工具调整Windows分区来得到所需空间，10GB左右的文件系统，0.5—1GB交换分区。
- 随提示安装。（注意不要把已经安装的系统覆盖掉！）
- 随时准备在网上寻求帮助。
- 参阅系统手册（man）。

尽量留足空间，把发行版上提供的软件尽量多的软件装上，等熟悉后再卸载那些不需要的软件包！

根目录下的子目录

bin	dev	home	lib64		media	opt	root	sbin
boot	etc	lib	lost+found		mnt	proc	run	srv

- /usr: 主要的系统文件和软件
- /home: 普通用户的家目录
- /bin: 常用的可执行文件
- /mnt: 外挂设备的挂载点(或是/media目录下)
- /root: 超级用户的家目录

Outline

- 1 大规模科学计算简介
- 2 高性能并行计算机简介
- 3 自适应计算
- 4 并行编程模式
- 5 操作系统与并行编程环境
- 6 并行算法
- 7 Linux操作系统
- 8 Linux基本操作**
- 9 实习与思考

Linux常用命令

- 显示当前路径: `pwd`
- 切换目录: `cd`
- 列出当前目录: `ls`
- 删除文件: `rm -f (filename)`
- 创建目录: `mkdir (dirname)`
- 删除目录: `rm -rf (dirname)`
- 移动目录或文件: `mv` (相当于重命名文件或文件夹, 或将文件移动到目标文件夹)
- 查看文件内容: `more`, `less`, `cat`(比较`more`与`less`的区别)
- 文件拷贝: `cp (src) (dest)`
- 目录拷贝: `cp -r (src) (dest)`
- 文本编辑: `(g)vim`, `(X)emacs`, ...
- 帮助: `man (命令名称)`; `info(命令名称)`

vi基本概念

- 命令模式 (command mode) :控制屏幕光标的移动, 字符的删除
- 底行模式 (last line mode) : 将文件保存或退出vi等
- 插入模式 (insert mode) : 文字输入, 按ESC键回到命令模式
- 可视模式 (visual mode)

vi 基本操作

- 进入vi: vi filename
- 切换至插入模式编辑文档: 在命令模式下按字母键i
- 退出vi及保存文件: 在命令模式下按: 键进入底行模式
 - :w filename 将文件以指定的文件名filename保存
 - :wq 存盘并退出vi
 - :q! 不存盘强制退出vi

Outline

- 1 大规模科学计算简介
- 2 高性能并行计算机简介
- 3 自适应计算
- 4 并行编程模式
- 5 操作系统与并行编程环境
- 6 并行算法
- 7 Linux操作系统
- 8 Linux基本操作
- 9 实习与思考**

- 访问top500网站: www.top500.org, 了解目前并行机发展的动态
- 访问我国的top100网站。
- 例1.2中的Gauss-Seidel迭代具有并行度么?
- 当前有一个很流行的名词云计算, 通过google等收索工具, 理解一下它与并行计算有什么关系。

试着在你自己的计算机上安装Linux操作系统, 并熟悉之。