

高性能计算与传统学科交叉教学探索

赵广鹏 叶小涛

(河南理工大学信息化建设与管理中心, 河南焦作 454003)

摘要: 高性能计算作为第三类研究方法, 在众多领域已经得到广泛应用, 但目前大多数传统学科的学生缺乏对高性能计算的基本认知和知识储备。在考虑学生学习和科研工作实际需求基础上, 结合高性能计算系统管理和理论教学经验, 对基础知识体系进行构建, 对课程内容进行梳理, 并对教学活动进行设计, 以提升学生的学习兴趣, 提高学生解决实际问题的能力和科研素养。

关键词: 高性能计算; 学科交叉

0. 引言

高性能计算已被公认为继理论科学和实验科学之后, 人类认识世界改造世界的第三大科学研究方法, 是科技创新的重要手段。高性能计算相对于理论科学和实验科学, 有其独特的优越性。[1]

鉴于高性能计算对传统学科研究的巨大提升作用, 越来越多的学科投入到高性能计算的怀抱中来, 各高校科研院所一般都建立了规模不等的高性能计算集群。本文根据多年集群管理和计算理论经验, 从知识体系开始构建, 研究探索课程内容和教学设计, 提升学生学习兴趣, 提升学生学习和科研能力。

1. 高性能计算基础知识体系结构构建

高性能计算是一个庞大的知识体系, 里面涉及众多的知识点, 任意一个分支都可作为一门学科进行研究。但是对于学生日常使用来说, 无需掌握庞大驳杂的知识体系, 只需掌握其中一部分基础知识, 然后在实践中逐渐深入即可。在这样的前提下, 我们构建高性能计算基础知识体系结构, 着重增加作业调度系统、并行编程、脚本编写的内容, 对操作系统、集群优化方面不需要过多深入。主要包括以下几个方面:

1.1 认识高性能计算

高性能计算(High performance computing, 缩写 HPC) 指通常使用很多处理器或者某一集群中组织的几台计算机的计算系统和环境。有许多类型的 HPC 系统, 其范围从标准计算机的大型集群, 到高度专用的硬件。大多数基于集群的 HPC 系统使用高性能网络互连。基本的网络拓扑和组织可以使用一个简单的总线拓扑, 在性能很高的环境中, 网状网络系统在主机之间提供较短的潜伏期, 所以可改善总体网络性能和传输速率。

1.2 操作系统

高性能计算集群一般运行的都是 Linux 操作系统, 而且一般都是远程使用。因此, 学习高性能计算之前一定要熟悉 Linux 系统的基本操作。这其中包括操作系统的一些基本概念, 比如进程和线程的区别, 地址空间, 机器负载, cpu 利用率等基本知识; Linux 目录结构、文件格式、权限管理、常用命令使用、shell 脚本编程等[2]。

1.3 并行计算基础知识

高性能计算本质上依赖于并行计算, 通过同构或者异构模式, 实现高并发, 缩短计算时间, 提高计算效率, 节约时间成本。与学习其他编程语言一样, 了解并行编程语言其基本语法, 基本实现方式和执行方式, 了解其基本函数和数据结构, 进行程序编写和调试。

1.4 作业调度系统

目前高性能计算集群一般都采用作业调度系统来对计算资源进行统一管理。调度系统主要是由资源管理器和调度器组成的。资源管理器提供了作业底层的开始、持有、取消和监控操作。如果没有这些底层开销, 仅仅靠单一的调度器无法完成对作业的控制。调度器应该确保作业在什么时间什么地点什么方式运行能够保证整个系统运行最佳[3]。

2. 课程以及教学活动设计

2.1 课程设计

基于以上基础知识体系结构来设计高性能计算学习课程。首先应该是对高性能计算的基本认识, 包括发展过程, 目前进展, 我国目前所处地位等。第二, Linux 操作系统及其基本操作。了解操作系统基本原理, Linux 基本操作。在此基础上, 学习 shell 脚本的编写, 提升学生的实践能力和兴趣。第三, 并行编程语言基本介绍, 侧重点以及所擅长领域, 着重介绍 MPI 编程语言。第四, 基本调度系统介绍, 比如 LSF, PBS, SLURM 等。基于河南理工大学高性能计算平台, 以 PBS 为基础介绍其架构和工作原理, 队列设置的基本原则, 以及基本使用方法, 包括 PBS 脚本编写, 作业提交, 作业监控以及基本错误处理。

2.2 教学活动设计

除了传统的理论授课, 采用多媒体方式讲解知识点, 在课堂中穿插实践活动。通过学生自带笔记本电脑或者机房授课的方式, 搭建测试集群, 让学生自己动手远程登录, 实践操作。在课下注重学生成果的展示, 可以是 shell 脚本的编写、并行程序的编写、专业软件的安装等, 让学生撰写实现文档, 可以极大提升学生的学习热情和科研素养[4]。针对每个学科, 讲解典型实例, 让学生既对高性能计算有一定认识, 也知道该怎么利用高性能计算来进行科学计算。对于学生的考核方式, 也不局限于课堂提问, 可以设置一些实践目标, 或者通过论文撰写的方式让学生自己找出问题, 加深理解, 自主学习, 激发学习兴趣。

3. 结束语

在高性能计算体系中, 学生不仅仅要掌握本学科的知识, 还要对以上高性能计算基础知识有一定的了解, 才能更好的发挥出高性能计算的优势。如果只掌握本学科知识, 面对高性能计算只能望洋兴叹; 如果只掌握计算机知识, 没有相应学科背景, 那么高性能计算也就没有价值。这两者是相辅相成, 目前随着高性能计算的蓬勃发展, 这方面人才非常稀缺, 使得高校开设相关课程变得十分迫切。本文介绍的基础知识体系结构和课程设计, 可以使传统学科学生在短时间内了解并使用高性能计算, 在教学过程中获得了不错的反响。当然, 随着新技术和新理论的出现, 这些知识体系和课程设计也应该随之不断丰富, 保持理论和技术的先进性。

参考文献:

- [1]周毓麟, 沈隆钧. 高性能计算的应用及战略地位[J]. 中国科学院院刊, 1999, 3: 184-187
- [2]高性能计算课程教学内容设置与教学实践[J]. 盛乐标. 计算机时代, 2019(07)
- [3]Maui Administrator Guide. Supercluster Research and Development Group. 2002
- [4]蒋永荣, 方成, 夏金虹. 基于学科交叉培养本科生创新能力的探索[J]. 高教论坛, 2015(7): 45-48.

作者简介: 赵广鹏 (1986.1-), 男, 汉族, 河南新乡人, 讲师, 硕士, 主要从事高性能计算以及信息化建设与管理工作