Adoptions and Outcomes of NSF/IEEE TCPP PDC Curriculum at College of Staten Island

Yumei Huo, and Feng Gu
Department of Computer Science, College of Staten Island, City University of New York
Yumei.Huo@csi.cuny.edu, and Feng.Gu@csi.cuny.edu

Abstract

The Department of Computer Science at College of Staten Island, City University of New York, has been offering three different levels of PDC courses: lower undergraduate level, upper undergraduate level, and graduate level. After examining all our Parallel and Distributed Computing (PDC) courses and comparing them with NSF/IEEE TCPP Curriculum initiatives, we proposed some modifications to our current curriculum. We have gradually experimented and implemented those ideas in our PDC related curriculum. The modifications on some courses have been introduced. In this poster, we will mainly focus on the new developed PDC course Shared Memory Programming and some results we have achieved.

Introduction

Among twenty-three institutions of The City University of New York (CUNY), College of Staten Island (CSI) is one of the colleges that started teaching the parallel and distributed computing (PDC) courses. The Department of Computer Science at CSI has been offering three different levels of PDC courses: lower undergraduate level, upper undergraduate level, and graduate level. The early undergraduate level PDC course, CSC 229 Introduction to High Performance Computing, is a two-credit elective course and covers the basics of parallel computing. With the prerequisite of the first programming class, this course is intended for someone who is just becoming acquainted with the subject. This course include 1 lecture hour and three lab hours, in which students learn about concepts and terminologies associated with parallel computing, and obtain a lot of hand-on experience programming Message Passing Interface. The high undergraduate level PDC course, CSC 429 Advanced High Performance Computing, is a four-credit elective course and emphasizes parallel algorithms design and analysis. With the prerequisite of data structures, this course is intended for someone who has learned basic algorithm design (such as divide-and-conquer) and fundamental time complexity analysis. In this course (3 lecture hours and 2 lab hours), students will obtain both theoretical foundation for parallel algorithms and programming experience on MPI programming. The graduate PDC course, CSC 770 Parallel Computing, is a three-credit elective graduate course and covers parallel programming and parallel algorithm analysis and design in addition to the basic concepts of parallel architectures. This course is intended for students from computer science, engineering, mathematics, finance etc., who are interested in high performance and parallel computing.

After examining all our PDC courses and comparing them with NSF/IEEE TCPP Curriculum initiatives, we realize that our PDC course structure needs to be further improved. We proposed to modify our current curriculum including complementing some contents in our current PDC courses, introducing a couple of PDC topics into our core course CSC 326 Data Structure and developing a new PDC course Shared Memory Programming. Since Fall 2014, we have gradually experimented and implemented those ideas in our PDC related curriculum and the results are exciting. The modifications on courses CSC229, CSC429, and CSC326 have been introduced. In this poster, we will mainly focus on the new developed PDC course Shared Memory Programming and some results we have achieved. Our experience may give some colleagues an insight on developing their PDC related courses.

CSC4XX Shared Memory Programming

In this course, students will learn about parallel computing based on shared memory platform. The emphasis will be on shared memory architecture, OpenMP and GPU programming, analysis of correctness and performance of an OpenMP and GPU program, and algorithms that can be used on shared memory systems. The course will include both theoretical components and programming components.

The learning goals and assessment plans are presented as follows:

GOALS	ASSESSMENT
Understanding shared memory parallel computer architecture	The part I of each homework assignment will be focused on the fundamental concepts.
Learning how to program OpenMP	Programming assignment 1 on parallel computers
Learn GPU programming	Programming assignment 2 on parallel computers
Effectively modify or extend the classic algorithms to solve variants of problems	Corresponding homework problems will be served on assessing students' such ability.
Able to analyze the correctness and performance of an OpenMP or GPU program	Corresponding homework problems will be served on assessing students' such ability
Apply algorithmic techniques to research – tasks partitioning, data partitioning, etc.	Demonstrate practical skills by completing the programming assignment 1&2

The sample course schedule is presented as follows:

Week	Topic
1	1. Discussion of the syllabus, requirements, topics to be covered, etc. 2. Introduction to shared memory parallel computer architecture
2-3	1. Overview of OpenMP: multithreading 2. OpenMP programming styles correctness and performance consideration 3. Writing a first OpenMP program
4-5	1. OpenMP language features 2. Sharing the work among threads in an OpenMP program 3. Clauses to control parallel and work-sharing constructs 4. OpenMP synchronization constructs
6	1. How to get good performance by using OpenMP
7	1. Using OpenMP in the real world
8	**1. Midterm exam (in class)** **2. Presentation for programming assignment 1**
9	1. GPU introduction
10-12	1. Parallel programming in CUDA C
13	1. GPU programming performance evaluation
14	1. Using GPU in the real world
15	1. Review **2. Presentation for programming assignment 2**
16	**Final Exam (in class)**

Outcomes

One of the goals of this PDC education is to prepare necessary background for students to start the career in PDC, and conduct research in high performance computing. Two undergraduates Mohammad Butt and Rida Syeda, who took both CSC 229 and CSC 429, were hired as interns by CUNY High Perfomance Computing Center. Other students from PDC courses are interested in the intern positions and in the process of applying.

Furthermore, students gain interest in high performance computing by taking PDC courses and the PDC infused course, and actively participate in faculty's research. Junxiong Huang worked with Dr. Huo in the research project "Metaheuristics on High Performance Computers for Bi-criteria Scheduling on Two-Machine Flow Shop Subject to Availability Constraints" and their research results have been accepted by the Sixth Workshop on Parallel Computing and Optimization at IEEE-IPDPS 2016. The undergraduates Mohammad Butt and Rida Syeda joined in the Dr. Feng Gu's research and published two conference papers in 2015 Summer Simulation Multi-Conference and 2016 Spring Simulation Multi-Conference as coauthors. Rida Syeda applied NSF REU programs on high performance computing and were accepted by two programs, and she attended the NSF REU program to conduct HPC research at University of Texas at Austin.