Spring-12: Teaching Parallel Processing Using Multicore Computers While Looking Into the Cloud

Jie Liu
Computer Science Division
Western Oregon University
Monmouth, OR, USA
liuj@wou.edu

In the past few years, the development in nanotechnology has allowed chip manufacturers, such as Intel and IBM, to produce CPUs with more and more cores. For example, NVIDIA Tesla C2075 video card has 448 cores on a single GPU, and Intel's Knights Corner 3120A delivers 1.003 double precision teraflops with 60 1.053 GHz cores. Nevertheless, software is still not designed to utilize the additional cores. Checking into our school's 40 capstone projects for seniors for the past five years, only one project applied parallel processing.

With the help of the Spring-12 Early Adopters - NSF/IEEE-TCPP Curriculum Initiative on Parallel and Distributed Computing for Undergraduates program, we are systematically and purposely introducing PDC topics in several classes. Still, the class with most concentrated PDC concepts is our CS474 Concurrent Systems. In this poster, we will discuss our covering of PDC topics in this course, evaluation of our efforts, and our plan for the course's future directions.

The topics covered in our Concurrent Systems class

CS 474 is offered every Winter term and is a strongly recommended elective class for students concentrating on software engineering and is designed to study parallel architectures, parallel algorithms, and parallel programming. In this course, students have written parallel programs on UMA, Beowulf clusters, multi-core parallel computers, and GPUs. In the recent years, students have mainly programmed on Intel's multi-core and IBM's Cell processors on a PS2. This year, we added programming on multi-core GUPs using NVIDIA Tesla C2075 video card. Through coding, students gain firsthand experience on concepts such as process communication, data dependencies, scheduling, load balancing, locking, synchronizations, and effects of granularity. The class also covers many PDC proposed topics such as parallel computer organizations (mesh, hyper-tree, butterfly, hypercube, hypercube ring, Shuffle-Exchange networks, etc.), parallel algorithms on PRAM (finding Max in a *constant* time, fan-in, parallel prefix sum, list ranking, and parallel merging, etc.), parallel programming concepts (shared memory vs. distributed memory, message passing, speedup, cost of parallel algorithms, NC and P-Complete classes, Amdahl's law and Gustafson's Law, barrier and semaphore synchronizations, data parallelism vs. control parallelism, and Brent Theorem etc.), and parallel sorting algorithms (Bitonic sort, parallel quick sort, and random sampling, etc.). In addition, students have term projects and several programming assignments to practice many key concepts and algorithms discussed in lectures. Currently, we are mostly using Microsoft's Visual Studio Task Parallel Library.

A key component of our Concurrent Systems class is the required term projects. Right before the midterm, students are each assigned a project related to parallel processing. Students are allowed to find their own suitable projects when possible. The projects help students' learning in many ways. First, students need to conduct research either to learn some new tools or to find parallel

solutions to problems with well define sequential solutions. Second, students have to implement the parallel algorithms or to experiment with the new tools. Third, students have to give presentations. This not only sharpens their presentations skills, which is an important career preparation step, but also allows students to learn from each other on a wide variety of topics outside the text books. Some of the topics students currently working on are advanced GPU Programming, Parallel programming using F#, Parallel Processing in the Cloud -- the current state, Parallel Sorting by Regular Sampling, Parallel Gaussian Elimination, and False Coloring. Outstanding projects have been selected to be presented in school's Academic Excellence Showcase to award student's excellent academic achievements and to demonstrate the abilities of our students to their peers around campus, professors of other disciplines, and school administrators.

Evaluation and the survey

We are using several means to evaluate the effectiveness of integrating PDC topics and about our teaching. One measurement is how many projects being selected to present in the Academic Excellence Showcase. We also survey students to find out what they think about the class.

Out school's Academic Excellence Showcase, organized by the Honor Society of Phi Kappa Phi and the Program for Undergraduate Research Experiences, is a one day event dedicated to presentations of student's scholarly activities. Selected students in each division demonstrate their achievements. Generally, a large proportion of students in our Concurrent Systems course are selected to show their course projects. The Showcase serves several purposes. First, students are motivated when they know their projects are more likely to be selected. Second, the campus in large can assess our integration efforts and teaching effectiveness. And third, the process shows other students what students learned PDC concepts are capable of doing. The simple action of showing a 100% CPU utilization chart excites many juniors and sophomores and has actually attracted a few capable students to switch their majors to Computer Science.

Attached to student's final is an anonymous survey. The table below shows a subset of questions with student's answers in percentage. Clearly, student responses are overwhelmingly positive. Answers on questions #8 indicate students are honest when answering the survey. Answers to questions #10 are extremely encouraging, almost 90% students are glad they are taking the class.

Question	Strongly Agree	Agree	No Opinion	Disagree	Strongly Disagree
1. Every CS student should take this class.	37.5	50	12.5	0	0
3. The class discussed interesting topics.	75	25	0	0	0
8. We should have had more programming assignments	0	37.5	37.5	25	0
9. Bitonic sorting is an elegant algorithm and is very interesting	37.5	50	0	12.5	0
10. I am glad I have decided to take the classes	87.5	12.5	0	0	0

Evolving of the course to cover new concepts

Understanding the Cloud is a migration certainty, as well stated by Gartner's analyst Ken McGee, we are looking into introducing HPC in the Cloud to the class and already looking into research topics and issues of conducting HPC in the Cloud. For example, we have contacted HP Cloud Services to find out ways to create instances on the same CPU or the same motherboard whenever possible. We are also looking into parallel processing on Intel's Knights Corner 3120A.