Introduction

- Advent of multicore processors has completely changed the landscape of computing
  - Programmers used to be able to automatically take advantage of the microprocessors’ improvement
  - However the serial computing era has gone
- Parallel computing and distributed computing have now become universal
  - Multicore and GPGPU (General-Purpose Graphics Processing Units) computing on chip
  - Large-scale Grid computing and Cloud computing across wide-area networks
- Computer science undergraduate students need to
  - Study parallel and distributed computing principles
  - Learn how to write parallel programs
  - Prepare for their careers in this new computing era

Early Adoption Team at TTU

- From Left to Right: Dr. Yong Chen, Assistant Professor, Dr. Noé López-Benitez, Associate Professor, Dr. Yu Zhuang, Associate Professor
- Early Adopter status from Fall 2011
- We believe that it is crucial for undergraduate students to grasp the parallel and distributed computing principles and problem solving skills
- We found that the NSF/TCPP PDC Curriculum provides informative and invaluable guidance
- Aligns well with an earlier effort that was initiated to enhance the curriculum of parallel computing at TTU

Curriculum Design and Change

- Covers four undergraduate courses
  - Core course CS 3375 (Computer Architecture)
  - Elective course CS 4379 (Parallel and Concurrent Programming)
  - Core course CS 2350 (Computer Organization and Assembly Language Programming)
  - Elective course CS 4331 (High Performance Computing)
- Changes implemented primarily include
  - Cutting-edge new technologies
  - Application Programming Interfaces (APIs)
  - Programming examples
  - Algorithms
  - Intended to increase students’ enthusiasm about parallel and distributed computing

CS 3375 (Computer Architecture)

- Expanded content on instruction-level parallelism in the context of RISC machines
- Added content on multiprocessor and multi-computer architecture
- Dropped digital functional units while keeping data representation, I/O interfaces, memory, storage systems
- Possible due to the content change of a prerequisite the ECE 2372 (Modern Digital System Design)
  - Which has expanded coverage to include high-level circuitry and digital functional units beyond logic gates

CS 3250 (Computer Organization and ALP)

- Strengthened topics such as tasks and threads, and synchronization issues
- Added content regarding critical regions, concurrency issues such as deadlocks, spin locks, mutexes, etc.
- Although the major topics of CS2350 are not directly associated with PDC content, the course nevertheless provides important building blocks for PDC topics

CS 4379 Parallel and Concurrent Prog

- Moved multiprocessor and multi-computer architecture contents into CS 3375
- Increased the focus on parallel programming, as well as strengthened the coverage of parallel algorithm basics
- Covered programming paradigms for both shared-memory and distributed memory architectures
- Multi-thread programming using NVIDIA CUDA
- MPI for the message-passing paradigm
- Matrix and graph problems and algorithms are included and adapted for programming teaching examples

CS 4331 High Performance Computing

- A new elective course will be offered in Summer 2012
- Focus on hands-on experience and applications of parallel and distributed computing
- Prepare for a Student Cluster Competition at SC12

Early Adoption Experiences

- The NSF/TCPP PDC curriculum provides a valuable guidance for designing and improving undergraduate curriculum for parallel and distributed computing
- Most comprehensive and systematic curriculum design and guidance in PDC education
- Curriculum design including topics along the architecture, programming, algorithm, memory and distributed memory architectures

Possible Improvements

- Adopt real applications/real systems as the learning motivation and use them in teaching related topics
  - Will experiment in the summer CS 4331 elective
  - Provide project samples to have students reinforce and retain topics learned in class
  - Projects can be a systematic cross-course project to assist the integration of multiple courses
  - Would be difficult to come up without a coordinated effort like the PDC curriculum

Ongoing and Future Work

- In the process of making further improvements to the undergraduate curriculum at Texas Tech University
  - Further enhancements to CS 3375 that will cover latest advancements in CUDA GPU architectures
  - A new CS 4331 High Performance Computing
  - Enhancements to CS 4379 covering CUDA/OpenCL programming
- In the process of making CS 4379 a core course in the undergraduate curriculum
- Long-term goals if given continuous support
  - To elevate the need of PDC education to a level commensurate with its increasing importance
  - To carry out quality education of PDC topics in West Texas

Conclusion

- NSF/TCPP PDC curriculum has been adopted at TTU
- Early adoption experience has shown that the PDC curriculum provides a comprehensive resource for promoting parallel/distributed computing education
- Critical to elevate the PDC education as parallel/distributed computing become ubiquitous
- Seek further sponsorship/grants to continue the effort with a goal of delivering PDC education in West Texas

Acknowledgment: this work is sponsored by the National Science Foundation and IEEE Computer Society Technical Committee on Parallel Processing via the NSF-TCPP Early Adopter Status award made to the Texas Tech University.

Contact Information

Yong Chen (yong.chen@ttu.edu)
Yu Zhuang (yu.zhuang@ttu.edu)
Noé López-Benitez (noe.lopez-benitez@ttu.edu)