Challenges of integrating parallel and distributed computing topics into Undergraduate CSE curriculum of AUST: Bangladesh Perspective

Kazi A Kalpoma
Department of Computer Science and Engineering
Ahsanullah University of Science and Technology (AUST)
Dhaka, Bangladesh.
Email: kalpoma@gmail.com

Md. Shahriar Mahbub
Department of Computer Science and Engineering
Ahsanullah University of Science and Technology (AUST)
Dhaka, Bangladesh.
Email: shaikatcse@gmail.com

This paper presents planning and challenge issues, approaches and experiences related to incorporating parallel and distributed computing (PDC) topics into the undergraduate CSE curriculum of the Department of Computer Science and Engineering, Ahsanullah University of Science and Technology (AUST), Dhaka, Bangladesh. The main goal is ensuring that all the CSE graduates of AUST, at least able to think parallelism in their coding in software development, applications or design issues. In Bangladesh, most undergraduate computer science (CS/CSE) programs do not teach parallel computing concepts, and CS undergraduates typically exclusively trained to think and program sequentially. IEEE/ACM joint task force on Computing Curricula has recommended integration of PDC topics in undergraduate curriculum and increased focuses on these important topics. Our effort is motivated by the iPDC Summer Institute: Integrating Parallel and Distributed Computing in Introductory Programming Classes held at Tennessee Tech University on July 2018, which was funded by National Science Foundation (NSF) [1]. The main objective of their effort was to create a collaborative repository of readily available PDC modules that can be easily integrated into existing CS courses without much effort required by the instructors. Integration of PDC topics needs the addition of new content to a course may require removing existing content due to the density of topics covered in our existing course. We plan to use Mike Rogers et. al. models of PDC education [2] partially and periodically in several courses. After attending iPDC Summer Institute, a draft plan has submitted at the end of the workshop. However, this session will describe the plan and challenges in detail.

Ahsanullah University of Science and Technology (AUST) is a private university in Bangladesh established in 1995. It has nine departments including Computer Science and Engineering (CSE). The CSE department offers a Bachelor of Science degree, which required completing 161.25 credits over four years, and each year has two semesters.

In order to integrate the related PDC topics within our curriculum, we would have faced many difficulties: 1) lack of teaching experience of the faculty members about the topic. Therefore, proper training is required to understand the methodology for teaching the new platform. 2) Lack of teaching resources. We can use PDC syllabus and other resources from the universities who are already offering the PDC topics. 3) Lack of empty space to spare in CSE curriculum. Adding new content of PDC needs to remove some content from existing course content, which will be difficult for us to do. We are choosing those opportunities that apply thinking in parallel to traditional topics like Mike Rogers et. al. models of PDC education [2]. 4) Lack of proper infrastructure to teach the parallelism. We do not have any HiPC lab, but the problem can overcome by building an HPC cluster testbed using the Raspberry Pis as recommended in iPDC Summer Institute 2018 [2].

Our main goal is to ensure that every AUST CSE student is exposed to parallel and distributed computing. In particular, we focus on teaching students parallel thinking. We want every student to be exposed to fundamental issues in parallel and distributed computing from the algorithmic and programming perspectives. Students should also develop skills to analyze and problem solve in parallel and distributed environments.

Students of AUST are not well prepared to get the topic quickly. We, therefore, would like to introduce concepts of PDC periodically. We want to incorporate PDC concepts within 5 existing courses. All the details of these 5 courses including credit hours, the offered year and semester of the course and the semester that we are planning to incorporate the PDC concepts are shown in I below. Every 3 credit theory course includes three hours lectures and 1.5 credit lab course includes two and half hours of lab work.

We hope to initiate very basic concepts of PDC in the 1st year and 1st semester courses. CSE1108: Introduction to Computer System is the first course offered to the students who are just admitted into CSE bachelor program. The course includes most of the basic concepts of the components of a computer system including hardware, software, and networking. We realized that this course is the ideal place to introduce very first theoretical concepts of PDC. Moreover, in the same semester, we can introduce parallel programming approaches

TABLE I
ACTIVITY PLAN FROM COMING SEMESTER, FALL 2018

Course No	Course Title	Credit hour	Offered Year and Semester	Planned Semester
CSE1108	Introduction to Computer Systems	1.5	1st year, 1st semester	Fall 2018
CSE1101 CSE1102	Elementary Structured Programming Elementary Structured Programming Lab	3.0 1.5	1st year, 1st semester	Fall 2018
CSE1205 CSE1206	Object-Oriented Programming Object-Oriented Programming Lab	1.5	1st year, 2nd semester	Spring 2019
Contest/ Event	Practice PDC concepts and gathering feedback through university computer club	N/A	N/A	Fall 2019
CSE2103 CSE2104	Data structures Data structures Lab	3.0 1.5	2nd year, 1st semester	Spring 2020
CSE2207 CSE2208	Algorithms Algorithms Lab	3.0 1.5	2nd year, 2nd semester	Fall 2020

TABLE II
IPDC MODULES PLAN ACCORDING TO MIKE ROGERS ET. AL. MODEL AS IN [2], UNPLUGGED MODULE (KNOWLEDGE/COMPREHENSIVE)

Module Level (Course Name)	Module Name	PDC Concept: Bloom Level
Elementary Structured Programming Elementary Structured Programming Lab	Arrays in Parallel	Data Dependency: K/C Serial vs Parallel: K
Object Oriented Programming Object Oriented Programming Lab	PB&J Making	Serial vs Parallel: K Congestion: K

TABLE III
IPDC MODULES PLAN ACCORDING TO MIKE ROGERS ET. AL. MODEL AS IN [2], PLUGGED MODULE (KNOWLEDGE/COMPREHENSIVE/APPLYING)

Module Level (Course Name)	Module Name	PDC Concept: Bloom Level
Data Structure Data Structure Lab	M&M Sorting	Data Partitioning: K Parallel Overhead: K Serial vs Parallel: K
Algorithm Algorithm Lab	Matrix Multiplication	Concurrency: C Data Parallel: A Cache Locality: A

in the course CSE1101: Elementary Structured Programming, where students learn structural program, develop programming skills including variables, control structures, functions, and array. We plan to initiate basic PDC programming ideas such as parallel array to the course CSE1101. In the lab course CSE1102: Elementary Structured Programming Lab, students would implement the parallel array sum. CSE1205: Object-oriented programming is a course where students learn to solve a programming problem in an object-oriented manner. The main focus of the course is to teach a student basic object-oriented design concepts. In this course, we will introduce basic ideas of message passing interface through a Java platform (e.g. omp4j).

In the 2nd year and 1st semester, CSE2103: Data structures course is offered where we are planning to introduce different parallel merging algorithms. In the CSE2104: Data structures Lab course, students would implement the parallel sorting. In the 2nd year and 2nd semester, CSE2207: Algorithms course, the students learn the basic algorithms (e.g., Divide and Conquer, Greedy method, Dynamic programming, Back-tracking) for problem-solving and to analyze the complexity of an algorithm. We want to incorporate some advanced concepts such as concurrency, sequential dependency, synchronization through some parallel examples (e.g. matrix multiplication) as mentioned in III below.

As it is mentioned earlier that the teachers have lack of experience about the topic, we are in the process to train the corresponding course teachers according to the Mike Rogers et. al. proposed model as in [2]. At the same time, the evaluation/assessment policy for the students has to adopt similarly which is provided and suggested in iPDC summer institute [1]. Pre-course assessment will be performed to identify the students' areas of interest. Moreover, in-between assessment and post-course assessment will be carried out to evaluate the integration of PDC. Now the challenges are: Q/A setup to assess the students, what will be the assessment method? We may ask them to give new real life examples of parallelisms. Other challenge is. how to set the assignments/home works/online offline discussions?

Our effort is a multiple year continuous processing plan, involving at least 5 existing courses to which parallel topics will be added or expanded. The Fall 2018 semester represents our first steps of including parallel topics in two existing courses in our undergraduate CSE curriculum at AUST. Initial evaluations from both student and faculty perspectives will indicate that how the integration of PDC meet its proposed goals. Evaluation of student lab works and exams will indicate that students could recognize the difference between parallel and sequential algorithm analysis and could identify the benefits of parallel algorithms from a theoretical perspective. After one year, some events or contests may arrange to practice PDC topics. It will encourage students to learn parallel topics and we will be able to get the students feedback. Overall, we feel our initial implementation and evaluation of our curricular changes need more attention and evaluation to get the success. In future, we will plan to continue integrating parallel topics throughout the curriculum.

REFERENCES

[1] https://www.nsf.gov/awardsearch/showAward?AWD_ID=1730417&HistoricalAwards=false

[2] iPDC Modules, https://www.csc.tntech.edu/pdcincs/index.php/ipdc-modules/, Last accessed: Nov, 2018