Proposers:

Matthew Johnson (mattjjohnson@acm.org)
Robert H. Liao (liao_r@hotmail.com)
Alexander Rasmussen (alexras@acm.org)
Ramesh Sridharan (ramesh_s@berkeley.edu)
Daniel D. Garcia (ddgarcia@cs.berkeley.edu)
Brian Harvey (bh@cs.berkeley.edu)
Address:
EECS Department
University of California Berkeley
253 Cory Hall
Berkeley, CA 94720-1770
Phone: (510) 642-3214
Fax: (510) 643-7846
WWW: http://www.eecs.berkeley.edu/

Topic: Infusing Parallelism into Introductory Computer Science Curriculum using MapReduce

Abstract

We have incorporated cluster computing fundamentals into the introductory computer science curriculum at UC Berkeley. For the first course, we have developed coursework and programming problems in Scheme centered around Google's MapReduce. To allow students only familiar with Scheme to write and run MapReduce programs, we designed a functional interface in Scheme and implemented software to allow tasks to be run in parallel on a cluster. The streamlined interface enables students to focus on programming to the essence of the MapReduce model and avoid the potentially cumbersome details in the MapReduce implementation, and so it delivers a clear pedagogical advantage.

Content/Significance

Computer science today is rapidly moving towards parallelism as a means to surmount increasing problem sizes and the declining rate of clock speed improvements. Despite the changing environment, undergraduate curriculum, particularly at the introductory level, often provides little or no coverage of basic parallel programming concepts. Our project introduces parallelism to the lower-division computer science courses at UC Berkeley, primarily our very first course, CS61A. We address cluster computing instead of multithreaded parallelism because the details associated with multiprocessor systems would exceed the scope of the introductory courses, and many of the data-parallel programming concepts remain the same. There is an entire upper-division course dedicated to multi-core concurrency issues (see http://www.cs.berkeley.edu/~yelick/cs194f07/) By covering parallelism in introductory courses,

students gain an earlier exposure to and greater appreciation of its advantages, disadvantages, and uses.

In CS61A, we chose to teach cluster computing by introducing Google's MapReduce (http://labs.google.com/papers/mapreduce.html) at a high level of abstraction, as it provides both a practical and intellectually compelling example of a highly parallelized system. MapReduce is presented entirely through a Scheme interface which works with Hadoop (an open-source implementation of MapReduce) to execute student programs on a new cluster. We also designed instructional modules for two other lower-division courses, CS61B and CS61C, which introduce MapReduce and the Message Passing Interface (MPI), respectively, at a lower level of abstraction. Together, the curriculum for the three courses provides students with a broad exposure to cluster computing concepts.

In this poster we focus primarily on CS61A, as this course received the most curriculum development and required the most infrastructure. We describe the Scheme-Hadoop interface and outline our implementation, as well as provide examples of the course material we developed.