Involving Undergraduates in Computational Science and Engineering Research: Successes and Challenges

R.M. Kirby, C.R. Johnson, and M. Berzins

School of Computing and Scientific Computing and Imaging Institute,
University of Utah, Salt Lake City, UT, USA

Abstract. The undergraduate years of one's educational career are among the most formative in terms of education and research outlook. The university education should both broaden the student's perspective, while simultaneously sharpening critical thinking skills. Though classroom experience is beneficial, it is incomplete. The purpose of this paper is to address some of the issues associated with providing computational science and engineering education at the undergraduate level by surveying efforts made at the University of Utah. Specifically, we will describe a program that allows students to become involved in research during their undergraduate years, and will provide two success stories demonstrating its efficacy. We also discuss efforts being made at associated undergraduate curriculum reform.

1 Introduction

The mission of the university at the undergraduate level is to educate and train. The first effort attempts to develop critical thinking skills that transcend any specific discipline; the second effort attempts to instill discipline-specific practical skills that help students to be productive members of the work-force and of society. It is well understood in many fields that although the classroom experience is necessarily, is it incomplete. It is our contention that education in the sciences and engineering can be made complete by introducing research opportunities at the undergraduate level. Research experience for undergraduates provides a means of honing critical thinking and research skills under the tutelage of faculty interested in both the student's research and academic development. Traditionally project-work towards the end of an undergraduate degree provides some experience of research for students. However there are a multitude of benefits at the student, faculty and university level for undergraduate students to be immersed in a research environment for a lengthy period of time. For instance, at the student level, the benefits are:

- Many students find being part of a research cohort to be intellectually stimulating. As part of a mentor's research group, they are able to participate with others in the process of discovery; in group meetings to update members involved on the research project; and to obtain access to faculty who often treat the student more like a graduate student.

V.N. Alexandrov et al. (Eds.): ICCS 2006, Part II, LNCS 3992, pp. 147–154, 2006.

- They are able to advance their personal/professional goals (i.e. enhancement of cognitive/communication skills, letters of recommendation, graduate school training, etc.)
- Integration within a research laboratory often helps motivate students to study harder in their classes by giving them context for what they are learning. Many of us have had the experience of being presented with a seemingly abstract topic with little motivation, e.g. infinite series expansions or functional analysis, and thus did not expend much time on the subject, only to find out later that this material is now key to understanding many important methods. Students not stimulated by the university curricula often find their research experience to be a far more invigorating blend of application and theory.

The student is not the only one to benefit. Benefits of this experience for the faculty and university are:

- It helps to expand the university's commitment to scholarly activity and/or scientific inquiry, and as such provides a good source of "advertisement" to attract new students.
- The undergraduate helps the advisor in advancing his/her career and/or research environment, and many undergraduates are capable of functioning at a level similar to that of many graduate students for lower funding needs.
- In the end, we believe that the students that participate in a research experience receive a superior education, and that is good for all concerned.

In this paper we describe two efforts at expanding computational science and engineering at the University of Utah. The first effort we will discuss is the Engineering Scholars Program, a College of Engineering program geared to get undergraduate students into the research environment as early as possible. We will present some details of the program and will discuss two student success stories. The second effort is our attempts to revamp the view of scientific computing undergraduate education through new undergraduate courses that present the computational science pipeline.

2 Engineering Scholars Program

The focus of the Undergraduate Engineering Scholars Program (ESP) is the engagement of between ten and twenty *first year* undergraduate engineering students in research opportunities. This early exposure to the research environment enhances the engineering education experience and better trains these future engineers for positions in industry, government, and advanced graduate training.

The quest of the ESP is to leverage the exciting engineering-based research at the University by exposing first year students to actual research activities. The addition of a research component to the first year student's experience allows the student to see over the horizon offered by prerequisite courses. By giving the first year student a more in-depth look into engineering, it is hoped that we will increase the enthusiasm level of the students with the results being significantly improved retention rates and better trained students. Higher retention rates will translate into more, better-trained and much-needed engineers and computer scientists.

The program is designated as a one-year program during the freshmen year where the students are exposed to engineering research through a set of tours and demonstrations in the Fall semester. During Spring semester, students are placed in Engineering and Computer Science labs where they work between 10-20 hours per week participating in ongoing research projects. Although they may not be able to do the most complicated parts of the research, the opportunity to participate in associated tasks allow them to gain an understanding of the research process.

A unique aspect to the ESP is that students are engaged in research their freshman year. Several universities have programs that engage undergraduates during their senior year (in the form of a senior project or senior thesis). While we think that any engagement in research during undergraduate training is positive, we have noted that ESP students, as is demonstrated below by the student examples, often continue to work within research laboratories throughout their entire undergraduate training. There have been multiple instances where ESP students have co-author journal and conference papers, as well as given presentations at research conferences. It is almost always the case that people are astonished that undergraduate students are doing such a level of research. We too have been impressed at the level of research that some of the ESP students can tackle when properly challenged and motivated.

An additional benefit of the Engineering Scholars Program is that the students participating in the program are introduced to fellow peers in the program, and indeed some are even housed in the same building. Events such as social gatherings and field trips are regularly scheduled for both current members of the program and the program alumni. A highlight is the annual inaugural luncheon where students and parents are brought together with participating professors, top-level College of Engineering administrators and program sponsors.

2.1 Engineering Scholar – Lindsey Healy

Lindsey started her research experience working in a Bioengineering lab, but then migrated to working with Radiology Professor Sam Browd, on a project involving finding a way to visualize the cervical vertebral body in children with Down Syndrome.

Children with Down Syndrome are more prone to congenital abnormalities of the cervical spine than their normal age-matched counterparts. These abnormalities of the cervical spine leave the Down Syndrome population with an increased susceptibility to spinal injury from normal activity. In order to analyze the origin of spinal instability of the atlanto-occipital joint, Lindsey performed segmentation on CT images taken prior to surgical fixation of the unstable joint complex. After segmenting and creating a reconstruction of the first cervical vertebral body for the patient and control population, the superior particular facet was

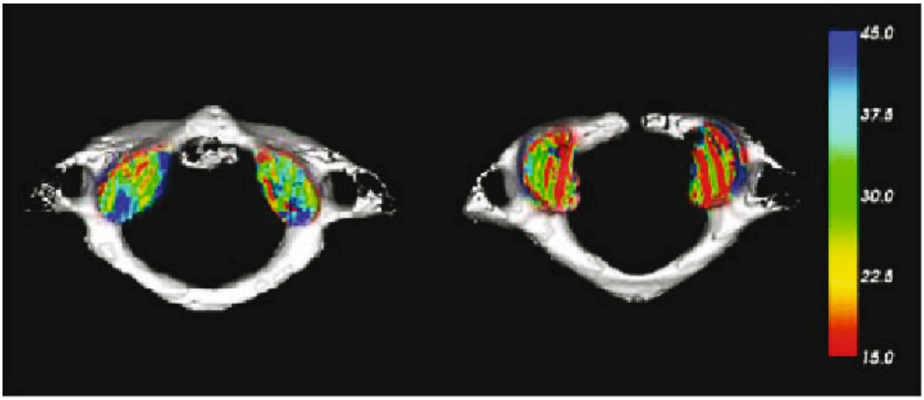

Fig. 1. Representative data comparing the normal age-matched control (left), to the Down syndrome patient (right). Numbers on color scale indicate degrees between the perpendicular at any particular point and the vertical. Blue therefore indicates areas closer to parallel (more sloped) with the vertical while red indicates areas closer to perpendicular (less sloped) with the vertical.

color mapped according to the slope to visualize differences between subjects and age-matched controls as shown in Figure 1. In addition, quantitative measurements were taken and length over depth ratios were calculated. The results showed a significant difference ($p = 0.01$) in the morphometric measurements between the two groups as shown in Table 1. As part of this research, Lindsey worked closely with members of the Pediatric Neurosurgery Department at Primary Children's Medical Center in Salt Lake. This research was accepted for an oral presentation at the AANS Pediatric Section1 annual meeting in December 2004 and has been submitted for publication. It is the hope that this research will provide an avenue for further research and software development for improved and preventative screening within the Down Syndrome population.

Because of Lindsey's positive experience, she has gone on to work on multiple research projects during her undergraduate education, including a segmentation and visualization project with geneticists Mario Capecchi and Charles Keller that resulted in a paper that was recently submitted to Nature Genetics, and then worked with Radiology Professor Norman Foster, who directs the Center for Alzheimer's Care, Imaging, and Research at the University of Utah.

Lindsey graduated in 2005 with a B.S. in Bioengineering. As an undergraduate, she was awarded nine scholarships. At the time of this paper, she is applying for graduate programs in Bioengineering.

2.2 Engineering Scholar – Curtis Hamman

Curtis Hamman began his Mechanical Engineering studies at the University of Utah in the Fall of 2003. By the end of that first Fall semester, Curtis had obtained a research position directed by Professor Mike Kirby at the Scientific

Computing and Imaging Institute. Curtis has been active in two interrelated projects presented below.

Project 1 – Visualization and Analysis of Turbulent Flows: The study of fluid flows is typically based on the three vector velocity components $\boldsymbol{u} = (u, v, w)^T$ and the scalar pressure field p. These quantities can adequately describe the state of any incompressible fluid flow, nevertheless, the computation and visualization of derived quantities such as vorticity $\boldsymbol{\omega}$ can improve our understanding about the flow dynamics.

In the same spirit, Curtis has been involved in the study of a newly appreciated derived quantity, the divergence of the lamb vector. At a time in their student careers when most are attempting to grasp velocity and vorticity as concepts, Curtis was using research experience to solidify his understanding of fluid mechanics concepts and their applications within engineering.

As an example – Curtis was challenged with the task of understanding both mathematically and through numerical evidence the differences between isosurfaces of vorticity and isosurfaces of the divergence of the lamb vector. His working hypothesis was that the divergence of the lamb vector is a far more localized quantity than vorticity, and that as such is far more amenable to modeling. To support his hypothesis, he post-processed and visualized data from a fully-developed turbulent channel flow with two homogeneous directions with $Re_\tau \approx 180$ and $Re_\delta \approx 3300$ as shown in Figures 2 and 3. These figures demonstrate that divergence of the lamb vector is a far more localized quantity (as shown in Figure 3) than vorticity (as shown in Figure 2). Presentation of these results to faculty in the Mechanical Engineering department here at the University stimulated an ongoing inter-disciplinary research effort to understand the differences between vorticity and the lamb vector.

Project 2 – High-Performance Computing Applied to Channel Flow Simulations: This project investigated the mathematical and computational

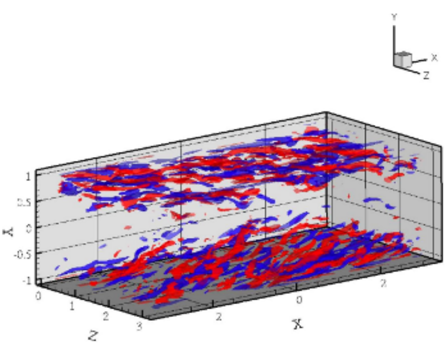

Fig. 2. Streamwise vorticity isocontours at $|w_x| = 2.5$. Red is positive, and blue is negative

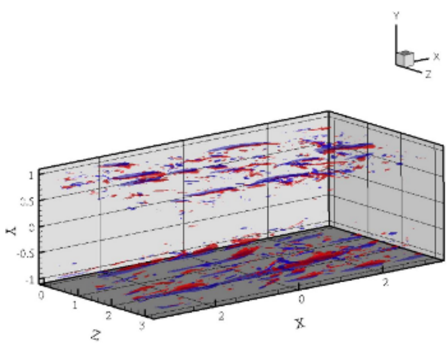

Fig. 3. Lamb vector divergence isocontours at $|n| = 250$. Red is positive, and blue is negative

Table 1. Wall clock time, speedup, and efficiency for varying numbers of processors for a simulation with $N_e = 32$, $P = 8$, $N_x = 64$, and $N_z = 64$. The predicted efficiency is given by an equation derived as part of the study.

# Processors	Wall clock time (s)	Speedup	Efficiency	Predicted Efficiency
1	20.47	1	1	1
2	10.72	1.91	0.96	0.98
4	5.74	3.57	0.90	0.92
8	3.08	6.65	0.83	0.83
16	1.90	10.78	0.67	0.67

properties of a one-dimensional Helmholtz solver with high-order spectral element discretizations. The principal benefit of this study is to elucidate the salient aspects of this particular parallel implementation in order to deduce its effectiveness in the Direct Numerical Simulation (DNS) of turbulent channel flow where wall-normal Helmholtz solves are conjugate to the streamwise-spanwise Fourier transforms. As part of the project, a parallel channel code was developed and computational models were formulated to predict efficiency. The results of a study of one of the stages of the flow solver are shown in Table 1.

Although these results are for relatively small numbers of processors they are significant in that they also show a close relationship between the predicted scalability and the actual scalability. More significantly the same code has been extended and shown to scale nicely on as many as 1024 processors [2]. Although the underlying parallel modeling methodology was that discussed in [1], the study was independent of a taught course, moved more quickly, and used many more processors by being linked to the existing DOE CSAFE research program at Utah.

3 Computational Science and Engineering Course Development

Many universities have undergraduate classes titled "Introduction to Scientific Computing" or "Introduction to Numerical Methods" or similar. Such classes provide an introduction to numerical techniques and sometimes make use of scientific software tools such as Matlab or Mathematica. In some cases however, such classes would be difficult to distinguished from a traditional numerical analysis course taught twenty years ago. A question we are asking is: is this the right way to expose undergraduate students to scientific computing?

It is our thesis that scientific computing deals with the application of the computational science pipeline to the study of science and engineering problems, and that a scientific computing class at the undergraduate level should expose students to this process. This idea has precedents in which problems and solutions were both considered [3], but has been extended to the complete computational science pipeline.

What distinguishes the computational science pipeline? Historically, the scientific method was formulated around the idea of postulating a model of natural phenomenon, making observations to validate one's model, and correcting the model based upon discrepancies between the phenomenon and nature. Later, the scientific process was extended to include the idea of the controlled experiment. No longer was the scientist limited to passively observing the world around him to deduce the correctness of the model. This gave rise to the idea of devising controlled experiments designed to evaluate the correctness of the hypothesis in a systematic manner. This systematic process allowed the model to be updated based upon the lessons learned through the experiment. With the advent of modern computing, a new paradigm called computational science has emerged, in which *experiment* now employed within the scientific method consists of the computational solution of the model. The computational science pipeline consists of the following stages:

– Scientific Problem of Interest (Problem Identification): Statement of the scientific or engineering problem of interest. Questions should be developed in such a way that quantifiable metrics for determining the level of success of the computational science endeavor can be evaluated.
– Modeling: The development of a model which abstracts the salient features of the problem of interest in such a way that exploration and evaluation of the model allows an answer to the questions specified concerning the problem of interest. Modeling techniques include, but are not limited to, deterministic or probabilistic, discrete or continuous mathematical models. Means of validating the model (determining the error introduced due to the model abstraction of the real phenomenon) should be established.
– Computation: The generation of algorithms and implementations which accurately and efficiently evaluate the model over the range of data needed to answer the questions of interest. This simulation of the physical phenomenon by computational expression of the model provides the experiment upon which the computational science pipeline hinges.
– Evaluation: The distillation and evaluation of the data produced through computational simulation to answer the questions of interest and to provide quantifiable determination of the success of the experiment. Methods such as, but not limited to, scientific visualization provide a means of tying the simulation results back to the problem of interest. Typically this stages leads to a revised model or computation and the process being repeated.

We are undertaking the creation of a new undergraduate version of semester long Introduction to Science Computing course that will expose students to the computational science method and the associated tools and techniques. As such, we will reduce the amount of numerical methods and add sections on mathematical and geometric modeling, visualization, and statistical evaluation. In addition we will stress the entire process by a few well chosen driving computational science problems.

An example of such a problem is the classic chemical morphogenesis work of Turing [4]. Following Turing, a basic idea of the chemical processes can be presented to the students, followed by the mathematical modeling tools which allow one to simplify the problem of studying morphogenesis down to understanding a system of ordinary differential equations. Once can then introduce basic time-stepping methods and concepts such as stability, accuracy and error control. Visualization is then used as a means of understanding the numerical results of the system, with the goal of connecting it back to the original problem of interest – chemical morphogenesis. This is just one of a multitude of such examples which help students to see the entire computational science pipeline in action.

4 Summary and Discussion

Multidisciplinary research has become an integral part of the research landscape, and its importance will continue to grow in the future. How discipline-centered university programs adapt to the changing nature of research will directly impact scientific and engineering progress in this next century. More tightly coupled integration of research and teaching is mandatory – especially at the undergraduate level. The University of Utah's Engineering Scholars Program and undergraduate computational science course as described in this paper demonstrate how undergraduates can be exposed to research ideas through both the classroom and laboratory environment.

References

[1] M. Berzins, R.M. Kirby, and C.R. Johnson, "Integrating Teaching and Research in HPC: Experiences and Opportunities", Proceedings of the International Conference on Computational Science, 2005.
[2] C.W. Hamman, R.M. Kirby and M.Berzins, "Parallelization and scalability of spectral element solver", SCI Institute Report UUSCI-2005-011.
[3] T.A. Grandine, "The numerical methods programming projects book", *Oxford University Press*, 1990.
[4] A.M. Turing, "The chemical basis of morphogensis", *Phil. Trans. Roy. Soc. Lond.*, B237:37-72, 1952.