Preparing the Future Workforce for Careers in Science and Engineering

Steven I. Gordon
sgordon@osc.edu

XSEDE
Extreme Science and Engineering Discovery Environment
We are Failing to Meet National Needs

- Although the numbers have been rising recently, the number of undergraduate and graduate degrees in science and engineering are not meeting national needs.

Source: KEY SCIENCE AND ENGINEERING INDICATORS: 2010 DIGEST
Filling the Gap

- We are increasingly dependent on foreign born graduates to fill the gap especially at the graduate level.
Problems Attracting and Retaining Students

• Students switching to non-science majors
  – Over 90% indicated poor instruction among reasons for switching
  – 26% had trouble learning the basic concepts (Seymour and Hewitt, 1997)

• Engineering program at Wright State University improved the four-year graduation rate by 40% by introducing an applied mathematics course to precede the traditional calculus sequence

• Recent National Research Council report on undergraduate physics education recognized the importance of inquiry-based methods in retaining students by improving student’s conceptual understanding
Computation is Central to How Science is Done

- Computation lets us explore phenomena that are too big or complex to experiment, too small, or changes too fast or too slowly.
- Computation allows us to explore more options more quickly.
Computation is how science is done
Marketing Computational Science
Challenges to Changing How and What We Teach

• We tend to teach in the way we were taught
• Computational science is interdisciplinary
  – Faculty workloads fixed on disciplinary responsibilities
  – Coordination across departments is superficial
  – Expertise at universities is spotty
• Major time commitments are required to negotiate new programs and develop materials
• Curriculum requirements for related fields leave little room for new electives
• Change is hard
Pathways to Reform

• There are many ongoing efforts at instructional reform in the STEM fields
  – Disciplinary oriented changes in math and science instruction
  – Comprehensive curriculum revisions

• Focus here on the efforts by the XSEDE project to facilitate the integration of computational science into the curriculum
  – This includes approaches which utilize inquiry-based learning
XSEDE Vision

The eXtreme Science and Engineering Discovery Environment (XSEDE):

enhances the productivity of scientists and engineers by providing them with new and innovative capabilities

and thus

*XSEDE accelerates open scientific discovery by enhancing the productivity of researchers, engineers, and scholars and making advanced digital resources easier to use.*
Promoting Formal Academic Programs

• XSEDE Education program is focused on assisting with the initiation and enhancement of formal computational science and engineering programs
  – Both undergraduate and graduate programs
  – Most sustainable way to help achieve the long-term project goals by producing a savvy workforce
  – Reduce the barriers to program adoption by
    • Providing program models
    • Solidifying a virtual community to share experiences
    • Providing faculty professional development
XSEDE Education Program Services

• Campus Visits
• Assistance with program creation
• Workshops for faculty and students
• Repository of shared materials
• Other resources
Initiating Services to Facilitate Change

• Campus visits
  – First discussions about integrating computational science into the curriculum
  – Discussion of formal programs
  – Opportunities for faculty professional development
  – Overview of related XSEDE services
Creating Computational Science Programs

• Inherently interdisciplinary
  – Science, engineering, or social science domain
  – Mathematics
  – Computer science
• Expertise often dispersed across multiple departments, colleges, institutions
• Difficulty of negotiating requirements, responsibilities, and institutional arrangements
What Do Students Need to Know?

- Considerable discussion across many disciplines
- Difficulty working from general conceptual ideas to specific skills and knowledge
- Several efforts focused on a competency based model to arrive at consensus of the essential knowledge base
- Competencies reviewed by both academic and non-academic experts
- See [http://hpcuniversity.org/educators/competencies/](http://hpcuniversity.org/educators/competencies/)
Ohio Minor Program Example

- Undergraduate minor program
  - 6-8 courses
  - Varies based on major
- Faculty defined competencies for all students
- Reviewed by business advisory committee
- Program started in Autumn 2007
- Agreements to share students at distance, instructional modules, revenues, and teaching responsibilities

<table>
<thead>
<tr>
<th>Competencies for Undergraduate Minor</th>
</tr>
</thead>
<tbody>
<tr>
<td>Simulation and Modeling</td>
</tr>
<tr>
<td>Programming and Algorithms</td>
</tr>
<tr>
<td>Differential Equations and Discrete Dynamical Systems</td>
</tr>
<tr>
<td>Numerical Methods</td>
</tr>
<tr>
<td>Optimization</td>
</tr>
<tr>
<td>Parallel Programming</td>
</tr>
<tr>
<td>Scientific Visualization</td>
</tr>
<tr>
<td>One discipline specific course</td>
</tr>
<tr>
<td>Capstone Research/Internship Experience</td>
</tr>
<tr>
<td>Discipline Oriented Courses</td>
</tr>
</tbody>
</table>
Example Competencies Simulation and Modeling

• Explain the role of modeling in science and engineering
• Analyze modeling and simulation in computational science
• Create a conceptual model
• Examine various mathematical representations of functions
• Analyze issues in accuracy and precision
• Understand discrete and difference-based computer models
• Demonstrate computational programming utilizing a higher level language or modeling tool (e.g. Maple, MATLAB™, Mathematica, other)
• Assess computational models
• Build event-based models
• Complete a team-based, real-world model project
• Demonstrate technical communication skills
Explain the role of modeling in science and engineering

Descriptors:
Discuss the importance of modeling to science and engineering
Discuss the history and need for modeling
Discuss the cost effectiveness of modeling
Discuss the time-effect of modeling (e.g. the ability to predict the weather)
Define the terms associated with modeling to science and engineering
List questions that would check/validate model results
Describe future trends and issues in science and engineering
Identify specific industry related examples of modeling in engineering (e.g., Battelle; P&G, material science, manufacturing, bioscience, etc.)
Discuss application across various industries (e.g., economics, health, etc.)

Example exercise
Flexibility in Implementation

• Adapt existing courses by adding computationally oriented modules
• Discipline oriented courses dependent on existing faculty expertise and interests
• Different subsets of required and optional competencies tied to major, required math, and example projects
Graduate Level Competencies

• Assumes some of the background of an undergraduate
• Focus more on research skills
• Core areas focus on the computer science and related modeling skills
• Need to branch into a wider array of specializations based on the nature of the graduate program
Graduate Competencies

Specializations
- Discipline-Specific HPC Simulation
- HPC Application Development
- Data Intensive Computing

Core Area 1
- Intermediate Scientific Computing
- Physical Sciences and Engineering
- Computer Science

Core Area 2
- High Performance Scientific Computing

Subject Areas
- Life Sciences and Bioinformatics
Developing Faculty Expertise

• Faculty professional development workshops
  – Two to six day workshops on a variety of topics
    • Computational thinking
    • Computational science education in science and engineering domains
  – Focus on local/regional audiences to reduce travel costs
  – Subsidies for faculty to travel to workshops at other sites
Special Workshops for Faculty and Students

• Development of synchronous and asynchronous education and training sessions
  – Multi-site broadcasts of workshops
  – Online training and education modules
  – Experimenting with full courses that can be widely shared for credit and non-credit inclusion in curricula (e.g. https://www.xsede.org/xsede-offers-free-online-parallel-computing-course)
Repository of Shared Materials

• Developing a repository of computational science education materials
  – Reviewed by professional staff and faculty
  – Indexed by subject and a detailed competency-based ontology
  – Goal: trusted, comprehensive source of information for computational science educators
  – http://hpcuniversity.org/resources/search/
Some Other Opportunities

• Journal of Computational Science Education
  – www.jocse.org
  – Peer reviewed articles on computational science education experiences

• Become a reviewer or contributor to the online repository

• Use the XSEDE online materials
  – www.xsede.org
Questions

• What can be done to encourage our universities to support increased interdisciplinary instruction and coordination?

• What kinds of campus activities and services from projects like XSEDE would advance computational science education on your campus?

• What strategies might work best for integrating computational science into the curriculum in the face of limitations on total credit hours for a degree?

• Are there infrastructure barriers that inhibit the integration of computational modeling into instruction?

• How can we revise the instruction in large, introductory lecture classes to integrate inquiry-based learning?

• Are the computational science competencies for undergraduates presented at http://hpcuniversity.org/educators/competencies/ appropriate for students in your discipline?