# Environmental Education in the University of California Computer Science and Computer Engineering Curriculum

Mason Kwiat

University of California, Santa Barbara

Author Note

This paper was prepared for Writing 50, taught by Peter Huk.

## Environmental Education in the University of California

## Computer Science and Computer Engineering Curriculum

Computer Science and Computer Engineering students are increasingly graduating from college to work in fields pertaining to environmental sciences, energy technology, and energy research. Moreover, nearly all commercial computation and programming positions in the current career market require complex computer systems and servers where energy efficiency is in high demand. Computer Science and Engineering students at University of California campuses can take courses on parallel computing and generally learn to focus on computational efficiency as a goal of reducing computation time, but the curriculum is currently lacking specific courses that cover many important topics such as power management, virtualization, and sustainable hardware solutions. The University of California, Santa Barbara's curriculum provides an exemplary case study of an institution with strong Engineering and Environmental Science departments that could integrate more environmental instruction in Computer Science and Engineering fields. Courses focused on sustainability are necessary to prepare today's undergraduate students for research and work in growing environmental occupations and departments.

Professionals and Academics within the scientific and computing community agree that sustainability is a pressing issue for the Computer Science and Engineering fields. John Vig, President of the Institute of Electrical and Electronics Engineers (IEEE), created the President's Sustainability Initiative; he explained, "I see sustainability as a major issue of the millennium, one that's of increasing interest to society as well as to IEEE members… It involves many, many issues, including political, economic, and technological. Where IEEE can help is in technology" (Vig, 2009). Computer technology is integral to the future coexistence of humanity and the global environment. Why then, do university curricula seem to neglect this connection? If more academic institutions were to support sustainable engineering education, how would such universities go about integrating sustainability topics into their Computer Science and Computer Engineering departments? The two main methods for bringing in new topics to a curriculum are by incorporating the topics into existing classes that cover relevant material and by creating new courses. Integrating new information into existing courses would minimize cost and resources required for a separate course, and would keep the curriculum from becoming too bloated. The new material, however, could potentially cut into other important topics that teachers of existing classes must cover. The second option of creating a new course would increase the focus on sustainability. Although new courses would provide sustainability and provide the most relevant knowledge for Computer Science and Engineering students, many universities already include general Environmental Science and Ecology courses. At the University of California, Santa Barbara (UCSB), Computer Science students must take upper division statistics courses because many employers favor a strong statistics background. Furthermore, the exposure to a common professional field allows students to discover whether they would be interested in pursuing such a job, and it provides a useful background even though students are not directly involved in programming or circuitry in the course. Moreover, curriculum administrators for the engineering departments could expand the Environmental Science and Ecology classes already in place for majors in those disciplines, to allow engineering students a background in the subject.

General background classes would provide good experience for Computer Science and Engineering students, but some specific sustainability themes are more interrelated to computing, and would work better as new courses or as integrated material in computer specific courses. One area in which Computer Science and Engineering students should have more instruction is in the environmental cost of computer hardware and natural resources used in manufacturing computers. Many computer components require exotic metals as well as some toxic chemical compounds. Nearly all modern computers use lead solder for wire and circuit connections, and many use toxic flame-retardants, both of which make dismantling and recycling of computers more difficult and potentially harmful to the environment. Also prevalent are both steel or plastic cases and housings for components, which influence the environment through invasive mining in the case of steel and other metals, and through petroleum drilling and molding for plastics. Eric Williams, a student at United Nations University in Japan, estimated in 2004 that the average desktop computer with a 17" CRT (cathode ray tube) monitor would take 6400 mega joules of energy and 260kilograms of fossil fuels to make. This means the weight of the fossil fuels used in production are about 11 times greater than the weight of the product, compared to one or two times greater for other manufactured goods. Furthermore, a study of 1350 Japanese web users found that the average time between computer purchases was approximately two years (Williams, 2004). Though manufacturing costs of computer systems are still very high, some advancements have minimized the material needed for many components, especially the advent of smaller LCD (liquid crystal display) monitors over CRT monitors, which can contain up to eight pounds of lead (Royte, 2005). Similarly, companies are more often using leadless solder for circuit connections, and some are using recyclable plastics. It is highly important that Computer and Electrical Engineering students that plan to focus on computer design and manufacturing know the environmental impacts of the components they use in addition to the simple economic costs.

The high environmental costs of manufacturing computers are exacerbated by the generally short lifespan of most machines and poor means of disposal employed by most organizations and individuals. Though all computer users should learn how to properly deal with their unwanted electronics, students in computer-based majors are particularly likely to be involved in electronic disposal. According to the Environmental Protection Agency (2010), only 18 percent of electronic waste was recycled in 2006 and 2007. This is an improvement over previous years, but it still leaves over 1.8 million tons of televisions, cell phones, and various computer products in landfills (EPA, 2010). Independent recycling programs as well as some responsible manufacturers are trying to encourage users to recycle and repurpose their unused electronics. While these programs are beginning to take hold with the public, it would also benefit Computer Engineering students to learn more about the recycling process so that they may better understand how to safely take apart and reuse computer and electronic components.

Another sustainability subject, specific to Computer Science and Engineering students, is the problem of server and data center design and management. Almost all modern academic institutions and companies in computer technology depend on complex network server clusters for parallel computation, distribution of software services, or data manipulation and storage. According to David F. Carr, a 2006 estimate suggested that Google runs approximately 450,000 servers (Carr, 2006). Though Google does not publish specific numbers, they likely run many more servers today considering their growth. Academic institutions also use high performance server clusters for research and education, such as the San Diego Supercomputer Center at the University of California, San Diego. These systems can consume great quantities of electricity, but proper techniques can reduce power usage and increase efficiency of certain computations. Jeffery J. Evans of Purdue University has emphasized the issue of power usage in high performance computing (HPC) systems such as server clusters and the issue of power management systems reducing load. New methods are continually being developed, including systems that utilize both hardware and software to manage and optimize both processor usage and temperature control among specific nodes in high performance server clusters (Evans, 2010). In these systems, many individual computers, referred to as “nodes”, are networked together to share data and/or computational power in order to solve extensive problems or, in the case of web servers, distribute the processing of requests from outside users connecting to a website. Often the nodes in these server clusters are not all working in equal magnitude. If a certain program or process is using only a few number of nodes in a server cluster, then the intelligent server management software could switch between nodes being used to reduce time each node is under stress, and adjust temperature control to focus cooling on nodes currently under heavy load. Computer processors operate most efficiently when cool, and they generate a lot of heat, so proper cooling can greatly increase the efficiency and reduce the power usage of a server cluster. Greater detail on implementing such systems would have a lot of impact on both Computer Engineering students who go on to design server hardware systems and Computer Science students who will implement management algorithms for server control programs.

Another solution for mitigating the intense power demand and ecological impact of server computing is through server virtualization and cloud computing. In computing, virtualization refers to the act of running multiple “virtual machines” on a single physical machine. This can be useful when you need a number of servers running different processes, but only have a limited number of physical machines available. The virtual machines share the computing power of the single node. Proper use of virtualization can reduce the number of machines needed for a task, or for multiple tasks. This can reduce both power consumption and cost. Ringling College of Art and Design students, David Przybyla and Mahmoud Pegah, used the servers at their college as a case study for ecologically responsible computing. After first using virtualization software in 2006, they found they could consolidate many services onto a single server, and continue to increase its capacity without buying more machines. They estimated that one of their current high efficiency servers could take on the same roles that would otherwise require seven to ten less powerful servers (Przybyla, 2007).

Cloud computing is an idea related to virtualization. “The Cloud” generally refers to an abstraction of server computation resources that allows users to host content such as complex web applications in a way that reduces required knowledge of intricate server hardware details and allows for a more dynamically scalable infrastructure for responding to fluctuating or growing customer demand. Most often cloud computing is considered to be a web application hosting service, which companies like Amazon use to offer more advanced web hosting options over the internet without the user needing to worry about buying physical servers, installing necessary software, and adjusting to increased web traffic and demand. Though businesses like Amazon offer cloud services, it is also possible for companies to create private clouds on their own servers. Many technology companies, particularly those based on online applications and services, can utilize virtualization or cloud computing. With the growing trend of rich web applications and online interaction, Computer Science and Engineering students are more likely going to be interested in and work with virtual network systems and cloud-based applications. UCSB currently offers upper division electives in networking and database systems, but does not specifically cover server optimization using virtualization and cloud computing.

The process of controlled approximation is another means of computing a wide variety of problems in a more efficient and sustainable way. In software programming, it is often the case that there are many ways to implement a solution to a particular problem. The main idea behind controlled approximation is to implement a solution that is much more efficient, but it may not give the best or most accurate solution. There are many instances where exact solutions are not as important, and a more efficient program could be written. One example of such a problem is image compression. When represented in a computer program, an image consists of thousands of points of data that determine the color of a given point in a rectangular image. For large images, this can take a lot of space to store, and subsequently require a lot of processing power and time whenever the image is loaded or analyzed in a program. Various methods can approximate an image to a slightly lesser degree of accuracy while significantly minimizing the disk space needed to store the image. For example, if I have a website that displays an overview of many pictures from a photo album, I would not want the page to load all of the images at the highest resolution just to show a small representation of all the photos. Doing so would put a greater load on both the user's computer and my server that is sending the website. By utilizing image compression, I can save power and create a better user experience. This type of approximation also has many mathematical computing applications. Woongki Baek of Stanford University explained that programmers could use heuristics and make other approximations under specific circumstances to obtain a reasonable answer to a problem while using less computational power. However, Baek added that this method is usually implemented at the programmer's discretion, without any standardized means of controlling the quality and precision of the output. The majority of the problems that can benefit from approximation consist of looping over a system of equations or instructions, where the accuracy of the answer increases as the loop runs. Baek suggests that programmers can also use a general system that allows for control of answer accuracy by deciding when to break out of a loop early. By using this standard system, programmers could easily adjust levels of approximation as they fit into different situations and requirements of the software they are designing (Baek, 2010). In-depth instructions on how to implement such systems would be invaluable to Computer Science students. This subject could also be added into advanced algorithms classes at UCSB, providing students with greater control over the sustainability of their programming without greatly affecting their curriculum.

In addition to working toward greater control of software power usage and reduction of the negative effects of computing on the world environment, students and graduates also have the opportunity to use computing to positively affect human and environmental sustainability. One of the biggest issues facing modern science is the challenge of creating viable and renewable energy through technology. A big focus of energy research is design of new solar panels and wind turbines that will run more efficiently and cost less. Engineering students, particularly in Electrical Engineering, would benefit from learning about how to design circuits with photovoltaic arrays for solar panels, and the details of how those cells work, so that they may become interested in doing their own research after graduation to improve these designs. Other energy research areas include creating sustainable fusion power. Though this is still in varying early stages of viable use, Computer Engineering students could have a significant interest in learning the design intricacies of the powerful lasers required for creating fusion reactions. Furthermore, a lot of energy, research and development require complex simulations to model the physical systems, find errors, and determine more efficient models before implementing expensive test hardware. In addition to energy research, computation majors can affect sustainability of homes and other human occupied structures by developing control systems that manage lighting and heating to minimize power usage. A number of technologies can be used to create smart building networks, such as motion sensors or RFID (Radio Frequency Identification) tags that can tell if someone enters or exits a room. Students interested in both hardware and software would profit from learning to integrate systems like these by building architecture. All Computer Science and Engineering students could learn more about these sustainable energy solutions.

It is clear through my research that there are many subjects in which computing and environmental sustainability are tightly intertwined, and students would be interested in learning about these topics that are highly relevant to their future career opportunities. The important hurdle is determining how to integrate these expansive topics into the intricacies of existing curriculum such as the one at UCSB. A few universities have already implemented some of these computational sustainability topics and can serve as examples for curriculum integration. Dr. Yu Cai of Michigan Technological University developed an undergraduate course, Green Computing and Network Services, which demonstrated the potential success of involving sustainability in engineering education. Dr. Cai's course covered general concepts of green computing, background and motivation for sustainability in computing, server virtualization technology and its use in energy conservation, energy saving data center practices, hardware and software efficiency innovation, disk management and benchmarking, and electronic disposal and recycling (Cai, 2010). This course was built off an existing networking class in order to minimize impact on the existing curriculum. Similar methods could be applied to networking courses in the UC system. The student response to the new class at Michigan Technological University was very positive. Students that had enrolled in the fall 2008 and summer 2009 courses took an exit survey to assess their understanding of the material as well as their interest in carrying knowledge gained from the course to later projects. The ratings set the course among the top 10 percent of all courses at Michigan Technological University (Cai, 2010). This existing model provides a useful precedent that a green computing course can be very successful.

The Michigan Technological University provides a good model for a green networking class, but other topics could require a different approach. Moreover, different sustainability topics would be variably relevant to different computer-based majors at UCSB. The main electronics majors at UCSB are Electrical Engineering (EE), Computer Engineering (CE), and Computer Science (CS). EE students focus on the more physical size of computing, such as circuit design, processor design, and other hardware systems. CS majors study more theoretical computing, and concentrate more on software programming. CE students cover an equal mix of programming and hardware design. Of the topics I covered in my research, a few would be applicable to all majors, but others would only be relevant to either hardware design or software design. Learning about the environmental impact of various components, and proper disposal or reuse of such components, would be most relevant to Electrical and Computer Engineering students who focus more on using these building blocks in their circuits, though Computer Science students would also benefit from a fundamental understanding of electronic recycling. Building energy efficient server clusters and systems would be useful knowledge in particular for EE and CE majors. Computer Science majors, however, would be involved in designing smart power management software to optimize computer clusters. CS students would also be interested in virtualization and cloud computing topics. Likewise, controlled approximation is a method applied mostly to software programming. Sustainable energy technology such as wind and solar power would be most relevant to Electrical and Computer Engineering students who would be designing the hardware for these systems. However, CS students could also be involved in simulations for energy research, as well as programming control systems for arrays of solar panels and wind turbines. All engineering students would benefit from a basic background in Environmental Science.

The main challenge of creating a new curriculum is finding the best way to integrate these varying levels of interest for different students. The simplest adjustment that department advisors can make is to encourage or require that engineering students take an existing Environmental Science course. This would provide a substantial background for the subject to acclimatize and perhaps pique the interest of students who had not considered Environmental Science as a field related to computation. However, this is not enough for a fully sustainable computer education. UCSB and other academic institutions would need to create new classes to teach subjects such as sustainable hardware design practices, design of renewable energy systems, and energy conservation in networked server clusters. In addition to creating new classes, some topics could work better when integrated into existing classes. Algorithms for controlled approximation, for example, would work well within a course on other advanced algorithms. Similarly, professors could discuss energy saving practices in existing courses and explain the relationship between computational efficiency and energy conservation. This would likely be clear to most students with an understanding of processor hardware, but should be emphasized in early engineering classes. Through a combined process of creating a strong Environmental Science background, creating new courses for computer sustainability, and integrating sustainable topics and practices into existing courses, the University of California, Santa Barbara academic system can better prepare Computer Science, Computer Engineering, and Electrical Engineering students for the pressing environmental predicaments that they will face in the technology engineering fields.

# References

Baek, W. (2010). Green: a framework for supporting energy-conscious programming using controlled approximation. *Association for Computer Machinery, 45,* 198-209. doi:10.1145/1809028.1806620

Berger, I. (2009, February 6). IEEE takes the initiative on sustainability. *IEEE.* Retrieved from http://www.theinstitute.ieee.org/

Cai, Y. (2010). Integrating sustainability into undergraduate computing education. *Association for Computer Machinery*, *41*, 524-528. doi:10.1145/1734263.1734439

Carr, D. (2006). How google works. *Baseline Magazine*. Retrieved from
http://www.baselinemag.com/

Evans, J. (2010). On performance and energy management in high performance computing systems. *IEEE 2010 39th International Conference on Parallel Processing Workshops*. Doi: 10.1109/ICPPW.2010.66

Moloney, C. (2010). "Understanding understanding" across the disciplines: towards strategies for sustainable engineering education for the 21st century. *IEEE Xplore*. *6-9*, 1-31. Doi: 10.1109/TEE.2010.5508815

Przybyla, D. (2007). Dealing with the veiled devil: eco-responsible computing strategy. *Proceedings of the 35th annual ACM SIGUCCS fall conference*. Doi: 10.1145/1294046.1294116

Royte, E. (2005). E-gad! americans discard more than 100 million computers, cell phones and other electronic devices each year. As "e-waste" piles up, so does concern about this growing threat to the environment. *Smithsonian Magazine.* Retrieved from http://www.accessmylibrary.com/

Unknown Author. (2010). Statistics on the management of used and end-of-life electronics. *U.S. Environmental Protection Agency*. Retrieved from http://www.epa.gov/

Williams, E. (2004). Energy intensity of computer manufacturing: hybrid assessment combining process and economic input-output methods. *Environmental Science and Technology.* Doi: 10.1021/es035152j