Incorporating Sustainability into Computing Education

Preprint of Sustainability column in *IEEE Intelligent Systems*, vol. 31, no. 5, 2016

Douglas H. Fisher, Zimei Bian, Selina Chen

Computing is ubiquitous, and integral to many fields. So it is not surprising that higher education in the environmental sciences, for example, will typically include computing (e.g., for earth-system simulations) in its coursework. What of integrating sustainability, also pervasive in scope, into computer science education?

Until sustainability is in the bedrock of computing education, few students, professionals, and researchers will be conscious of the possibilities of the computing/sustainability nexus to transform society. This article surveys the integration of environmental and societal sustainability into computer science curricula at colleges and universities, with a particular focus on intelligent systems educational materials.

Levels of Integration

Currently, integration of sustainability into computer science higher education happens at two levels of granularity -- the *course level* and at the course *component level*.

Course-level integration happens by introducing computer science courses that are focused on topics at the intersection of computing and sustainability. For example, courses that we survey here include those named "*Computing, Energy, and the Environment*" and "*Introduction to Computational Sustainability*". At some institutions it may also be beneficial to consider courses in other majors, such as earth and environmental sciences, that include a considerable and integral computing component. For example, a course on "*Agent-Based Modeling*" that is taught by a department of earth and environmental sciences might serve as a solid elective that adds a sustainability component to a computer science curriculum.

In contrast, component-level integration happens by introducing course components, such as lectures, exercises, and projects, with sustainability themes into a computer science course that does not have a sustainability focus, such as courses in computer organization, database, and artificial intelligence.

We dive deeper into component-level and course-level integration, and then speculate on the future of sustainability-themed computer science education, to include *curriculum-level* integration, which is concerned with larger-than-course curricular constructs, such as concentrations, minors, and majors.

Component-Level Integration

There are many ways of infusing fine-grained sustainability elements into a computing course, to include homework exercises, lecture and textbook examples, supplemental readings, exam questions, and team-projects. Here are a few examples.

- A several-week project and sequence of exercises in a database course, taught by one of the authors and taken by the other two, has students reverse engineer the database used in a Dorm Energy monitoring project [1].

- Patterson and Hennessey's third edition of *Computer Organization and Design* [2], includes extended examples of "Computers in the Real World", most of which have sustainability themes.

- Closer to the home of intelligent systems, homework exercises of varying complexity can have students apply a machine learning algorithm being studied (e.g., decision tree induction, support vector machines) to data sets in sustainability-relevant areas such as smart electricity grids, transportation, and flora and fauna preservation.

- Another intelligent systems assignment can have students design an optimization algorithm that constructs protected areas for wildlife conservation, and compare their algorithms to those in the research literature.

Resources for sustainability-themed computer science course components, and intelligent systems specifically, can be found on the Web (e.g., *Artificial Intelligence for Computational Sustainability: A Lab Companion* [3] and *Teaching Sustainability in Computer Science* [4]). These follow in the tradition of more general resource repositories, such as *Nifty Assignments* [5], *IEEE Real World Projects* [6], and *MERLOT* educational repositories [7]. Ideally, fine-grained educational materials in sustainability-focused computer science will find their way into these more general repositories as well.

Course-Level Integration

At a higher level of granularity is the introduction of entire sustainability-themed courses into computing curricula. As noted earlier, this might include courses in sustainability-area disciplines that have a significant computing component. Here we limit our attention to computing courses offered in computer science or similar departments..

Courses to Date

Our survey of sustainability-themed computing courses, together with the term of offering, include

- Sustainability and Assistive Computing (Bryn Mawr, Fall 2010),
- Computing and the Environment (Vanderbilt, Spring 2011),
    - Computing, Energy, and the Environment (Vanderbilt, Fall 2016),
- Topics in Computational Sustainability (Cornell University, Spring 2011),
- Computational Methods in Sustainable Energy (Carnegie Mellon, Fall 2012),
- Computational Sustainability (U British Columbia, Winter 2013-2014),
- Computational Sustainability (Georgia Tech, Spring 2014),
- Seminar on Computational Sustainability: Algorithms for Ecology and Conservation (UMass Amherst, Spring 2014), and
- Topics in Computational Sustainability (Stanford, Spring 2016).

These courses typically have a seminar format, with pedagogical elements such as (a) readings from the research and popular literatures, (b) responses to these readings through short papers and in-class discussions, and (c) semester projects. Most of the courses are for upper-division undergraduate and graduate students.

Content of Sustainability-Themed Computing Courses

Collectively, the courses cover sustainability areas of wildlife conservation, water distribution and safety, climate change, energy, agriculture, urban design, disaster response, poverty and human health. Computing areas that are covered collectively across all courses include machine learning, optimization, simulation, systems, hardware, and algorithmic analysis. The reading lists of these courses can be found online [8], and they are dominated by intelligent systems.

Table 1 gives a list of topics for characterizing the previous crop of computational sustainability courses. In particular, we ran the Mallet topic modeling tool [9] on all the abstracts (or introductory sections) of the readings from these courses. Mallet uses word frequencies and other factors to find topics that cover the document collection. The details are not vital for our purposes, but we experimented with various numbers of topics, then used Mallet to discover the best five topics as reflected in the abstracts, after we (the authors) did some basic preprocessing (e.g., stemming and hyphenation). Ten characteristic words are listed for each topic by Mallet in Table 1. The authors gave each topic a name. The topic words are dominated by sustainability-related words, rather computing words, which merits further investigation; using the entirety of articles from courses' reading lists, rather than just abstracts or introductory sections as we did, may yield somewhat different results.

Table 1: *Topics covered by computational sustainability courses discovered using Mallet.*

TOPICS		
#	**Topic Words**	**Topic Name**
0	model data species climate prediction ecology monitoring distribution methods	Ecological Modeling
1	energy emissions energy-consumption environmental data planetary-boundaries systems reduce smart-meter	Energy
2	environmental foodweb health economic food development complex services sustainable	Human Health
3	infrastructure fish urban optimization ocean agriculture water cost systems	City and Agriculture
4	networks optimization algorithms conservation habitat wildlife-corridors graph budget planning	Optimization and Planning

Mallet can also take a single document and characterize it by topics that it has discovered. We appended the abstracts of each course into a single "document" per course and used Mallet to assign each course proportions using the topics of Table 1. For a given course, the topic proportions sum to 1.0. Table 2 gives a visual representation of each course's topic coverage. We have highlighted in green the topics of each course with a proportion that exceeds the average of 0.20 = 1.0 divided by 5 (topics), which might be considered the most characteristic topics of the course. For example, Vanderbilt's course is most strongly characterized by topics 1 and 4.

Table 2: *Topic proportions of previous computational sustainability courses. Syllabi and reading lists of these and other courses are in [8].*

COURSE CONTENT BREAKDOWN					
	Topic 0	Topic 1	Topic 2	Topic 3	Topic 4
Bryn Mawr	0.25	0.17	0.25	0.05	0.28
Vanderbilt	0.15	0.33	0.17	0.05	0.29
Cornell	0.18	0.00	0.42	0.32	0.08
UBC	0.05	0.52	0.10	0.29	0.03
Georgia Tech	0.27	0.10	0.12	0.22	0.30
UMass Amherst	0.68	0.00	0.00	0.00	0.32

Instructors of subsequent courses can add to the online list of course readings and other resources [8], as appropriate. This list is intended as a resource for instructors to compare and select readings as they plan and revise their own courses.

Evolution of Computational Sustainability Education

We have briefly surveyed educational materials and courses to date, and pointed to online resources that instructors and students can share and expand. We close with some thoughts on the future of education at the nexus of computing and sustainability. *Computational sustainability* is quickly becoming the commonly accepted name for this nexus [10], and we use it here to signify a maturing field.

On Campus and Online Courses

A seminar format, which dominates current computational-sustainability courses, is typical in a new area where the instructor is exploring the territory, at least pedagogically. As computational sustainability matures, we can expect that courses with more formal designs will emerge, presumably following along with the "bottom-up" development of a rich set of educational resources as part of component-level integration. Most importantly, textbooks in computational sustainability can be both a driver and manifestation of greater maturity in the field.

Online courses and modules are another avenue for creating educational computational-sustainability materials, allowing instructors to share their lectures and other resources with a global audience. Many online courses follow a formal undergraduate course design, with lectures, assignments and exams that are auto-graded or peer-graded, but there are examples of very successful online courses in areas such as literary analysis that have significant seminar design components like the computational-sustainability courses of today, including live, virtual seminar discussions by students and instructors across the globe.

The online medium also offers unique opportunities for instructors, as well as students, to collaborate across traditional institutional boundaries [11]. The time seems right for a global course in computational sustainability, engaging many of the researchers and instructors who have pioneered the field to date. Such a course would undoubtedly be hybrid, with much of the content delivered online, but with local cohorts meeting physically, and perhaps customizing the computational sustainability content to the environmental and infrastructure challenges of their region.

Curriculum-Level Integration: Tracks, Minors, and Majors

As more courses develop and become differentiated in computational sustainability, instructors will recognize and design complementary courses that can be arranged into minors, tracts or concentrations, and perhaps even entire majors, for on-campus, online, and hybrid settings. For example, a specialized track might cover "Machine Learning for Computational Sustainability", with courses within the track differentiated by sustainability topics (e.g., Machine Learning for Urban Operations, Machine Learning for Wildlife Conservation). But we can expect more than courses and programs that are obvious intersections of the parent areas of computing and sustainability, and more too than prerequisite courses from each area (e.g., Data Structure Techniques from computing and Basic Hydrology from sustainability).

What will be the design principles for larger curricular constructs in computational sustainability education? One ideal is that computational sustainability majors be designed around frameworks of evidence-based decision making [12]. After all, it is the myopia of human decision making that computational sustainability is intended to correct.

Concluding Remarks

The nexus between sustainability and computing is natural and important -- computing is ubiquitous, and sustainability concerns, knowledge, and strategies should be ubiquitous too. Just as many fields are integrating computing into their educational curricula, it seems wise to integrate sustainability science and engineering into computing education.

We have surveyed existing resources and courses that integrate environmental and societal sustainability into computing higher education, particularly with respect to intelligent systems. Interested readers are encouraged to use the online resources [3, 8] for more details, and to add to them.

Acknowledgments

Douglas Fisher, Zimei Bian, and Selina Chen are supported by NSF Grant #1521672 for research, education, and reporting on advances in computing and sustainability.

References

1. John E. Petersen, Vladislav Shunturov, Kathryn Janda,Gavin Platt and Kate Weinberger, "Dormitory Residents Reduce Electricity Consumption when Exposed to Real-Time Visual Feedback and Incentives" *International Journal of Sustainability in Higher Education*, vol. 8, no. 1, 2007, pp. 16-33 http://www.emeraldinsight.com/doi/pdfplus/10.1108/14676370710717562

[2] D. A. Patterson & J. L. Hennessy, *Computer Organization and Design: The Hardware/Software Interface*, Third Edition, Morgan Kaufmann Publishers, 2005.

[3] Wikibooks Contributers, *Artificial Intelligence for Computational Sustainability: A Lab Companion*, Retrieved June 24, 2016 from https://en.wikibooks.org/wiki/Artificial_Intelligence_for_Computational_Sustainability:_A_Lab_Companion, Wikibooks, The Free Textbook Project.

[4] *Sustainability Improves Student Learning: Teaching Sustainability in Computer Science*, Retrieved June 24, 2016 from http://serc.carleton.edu/sisl/sustain_in_cs.html .

[5] *Nifty Assignments*, Retrieved June 24, 2016 from http://nifty.stanford.edu/.

[6] *IEEE Real World Engineering Assignments*, Retrieved June 24, 2016 from http://www.realworldengineering.org/ .

[7] *MERLOT II: Multimedia Educational Reseource for Learning and Online Teaching*, Retrieved June 24, 2016 from https://www.merlot.org/merlot/index.htm.

[8] Wikibooks Contributers, A*rtificial Intelligence for Computational Sustainability: A Lab Companion/List of Computational Sustainability Courses*, Retrieved June 24, 2016 from https://en.wikibooks.org/wiki/Artificial_Intelligence_for_Computational_Sustainability:_A_Lab_Companion/List_of_Computational_Sustainability_Courses , Wikibooks, The Free Textbook Project.

[9] Andrew Kachites McCallum, *MALLET: A Machine Learning for Language Toolkit*, Retrieved June 24, 2016 from http://mallet.cs.umass.edu, 2002.

[10] Carla Gomes, "Computational Sustainability: Computational Methods for a Sustainable Environment, Economy, and Society." *The Bridge,* vol. 39, no. 4, 2009, pp. 5-13.

[11] Douglas H. Fisher, D., & Armando Fox, *Report on the CCC-CRA Workshop on Multidisciplinary Research for Online Education*, Feb 11-12, 2013 Washington, DC, Retrieved June 24, 2016 from http://www.cra.org/ccc/files/docs/CCC-MROE-Report.pdf.

[12] Douglas H. Fisher, "Recent Advances in AI for Computational Sustainability", *IEEE Intelligent Systems*, vol. 31, no. 4, 2016, pp. XX-YY. Preprint at https://my.vanderbilt.edu/csx892/files/2016/06/AISustainability.3.14.2016.preprintpublic.pdf

Douglas H. Fisher is an associate professor in the Department of Electrical Engineering and Computer Science at Vanderbilt University. Contact him at douglas.h.fisher@vanderbilt.edu. **Zimei Bian** and **Selina Chen** are senior undergraduate students in Computer Science at Vanderbilt University.