The University of British Columbia

Curriculum Vitae for Faculty Members

Date: 2008/08/31 **Initials**:

1-4. Basic Information

Surname	Wolfman	**First Name**	Steven
Middle Name(s)	Andrew		
Department/School	Computer Science		
Faculty	Faculty of Science		
Present Rank	Instructor I	**Since**	August 1, 2004

5. Education

a) Degrees

Institution	Degree	Subject Area	Dates	
			Start	End
University of Washington	Ph.D.	Computer Science	Jan, 2000	Aug, 2004
University of Washington	M.S.	Computer Science	Sep, 1997	Dec, 1999
Duke University	B.S.E.	Elec Eng & Comp Sci	Sep, 1993	May, 1997

Dissertation: Understanding and Promoting Interaction in the Classroom through Computer Mediated Communication

Supervisor: Richard J. Anderson, Professor, Department of Computer Science & Engineering, University of Washington

6. Employment Record

b) At UBC

Employer	Department	Position	Dates	
			Start	End
The University of British Columbia	Computer Science	Instructor I	Aug, 2004	

8. Teaching & Supervision

a) Areas of special interest and accomplishments:

Active learning, inside and outside of the classroom

I believe students learn by engaging actively with their subject. I use a variety of methods to encourage engagement including: regular interactive problem sessions, votes, and discussions during lecture; "Kinesthetic Learning Activities", such as simulating an interface with a human "robot" in my 2005 CPSC 444: User Interface Design course; lecturing directly from a programming environment to try ideas on-the-fly (in CPSC 101: Connecting with Computer Science, CPSC 111: Intro. to Computation, CPSC 311: Intro. to Programming Languages, and CPSC 313: Computer Hardware and Operating Systems 2008) or with interactive, pen-based presentation technology I developed during my PhD (in most courses); and weaving domains of interest to students into course material, such as a focus on music in my 2006 CPSC 111: Intro. to Computation course.

Students need to engage with the subject outside the classroom as well, and in many cases, they need scaffolding to understand how to engage successfully. For example, I encourage students to read with pencil in hand, *interrogating* the textbook rather than just reading passively. To encourage this habit in my two 2006 CPSC 111: Intro. to Computation courses, I introduced brief "weekly reading questions": low-stakes, formative assessments in which students submit a written question about the week's reading. In my 2006 CPSC 322: Intro. to Artificial Intelligence course, I supplied extra credit points for publicly (i.e., on WebCT) identifying mistakes or improving wording in the textbook.

Challenging, open-ended assignments also encourage students to actively explore their discipline. For example, I worked with an undergraduate research assistant to develop and deploy a framework for open-ended assignments, which has so far been used in two offerings of CPSC 111 (taught by other instructors) and published at the premier CS Education conference. The framework supports assignments in which students solve a single problem in many different ways (e.g., creating a graphical puzzle or "skinning" text in a visually interesting way), and those disparate solutions are brought together into a single joint application available to students and the general public on the web (e.g., a "maze" of connected students' puzzles or a chat application with configurable student-built skins). As another example, in my CPSC 313 course, students pull together their systems knowledge to tackle "challenge problems" such as altering as many bytes of the Windows `notepad.exe` executable as possible without affecting its functionality.

I have shared my work on active learning with the CS Education community through a variety of venues including: a highly-reviewed series of workshops on Kinesthetic Learning Activities (KLAs) at the premier CS Education conference, a community-editable website on KLAs (http://ws.cs.ubc.ca/~kla/), and peer-reviewed and invited talks at various venues on my work with pen-based technologies.

Human-Computer Interaction

Although my focus is teaching, I have kept up with Human-Computer Interaction (HCI) research by attending weekly research group meetings, supervising undergraduate and graduate HCI researchers, and reviewing papers for the primary HCI research conference CHI. I have used this research to spur interest in the breadth of ideas in CS. For example, in my 2006-2007 CPSC 101: Connecting with Computer Science courses, I illustrated the connection between psychology, industrial design, and CS by showing real interface designs that interact poorly with human cognition taken from upper-level HCI courses and literature (such as an airplane with an oft-used, innocuous button that was altered with disastrous results to control jettisoning spare fuel tanks). In my 2006 CPSC 322: Intro. to Artificial Intelligence (AI) course, I emphasized the classic aim of AI to "augment" rather than recreate or supplant human intelligence and illustrated it with examples from an industrial research lab where I had worked.

This emphasis on HCI practice and research is especially important because the HCI subfield of CS has attracted a much more diverse population of practitioners than field as a whole; so, regular connections to HCI may work especially well to engage women and underrepresented minorities in the field.

Contextualizing Computer Science

I make regular, broad connections between students' course material and many subfields of CS (besides just HCI), other disciplines, and key current events. For example, my 2008 CPSC 313: Computer Hardware and Operating Systems course used the question: "Why is it that bathrooms in a building are usually directly above/below each other?" to draw out parallels between exposed and hidden functionality in (building) architecture and information hiding in computer architecture. In my 2005 CPSC 221: Algorithms and Data Structures course, I introduced a module covering proof techniques by proving Arrow's Impossibility Theorem (an economics theorem stating that no voting system is perfectly fair). I then gave an assignment in which students explored mathematical and practical properties of the British Columbia Single Transferable Vote, shortly before it came up for a provincial election. In my 2005 CPSC 444: User Interface Design course, I gave interface design examples from my daily experiences (like a clever shopping cart wheel at IKEA that allows customers to bring the carts on sloping slidewalks) and then invited students to share their own everyday design experiences at the start of every few classes. I designed an assignment for my Spring 2006 CPSC 111: Intro. to Computation course in which students use Google's programming interface to build an Artificial Intelligence agent that answers questions like "What is the tallest mountain in the world?" In my 2004 CPSC 111: Intro. to Computation class, I responded to surveys that showed substantial student interest in Biology by introducing a nested loops example in which we developed a fully functional program for analyzing DNA microarray experiments. In my 2005 CPSC 311: Intro. to Programming Languages course, we discussed the programming languages research of the most recent Turing Award winner (the top award in Computer Science).

In general, each of my classes makes at least one "connection" each week, whether it be to advanced research in an area, to another subfield of CS, or to another discipline.

Recruitment and Retention of Women in CS

Women are underrepresented across several science and engineering fields, but Computer Science has the dubious distinction of *losing* enrolment by women over time at the undergraduate level. Redressing this imbalance has been one of my central concerns. I am a long-standing member of the UBC Focus on Women in CS committee. I have contributed to many of the committee's projects and overseen three: the 2006 ChicTech program that brought eight teams of junior high school girls together with undergraduate mentors to design websites for non-profits, the 2005-2007 Alumni Panel series (for which I was the Principal Investigator) that sponsored panels of female and male CS alums to discuss their diverse career paths with current students, and an ongoing Institute for the Scholarship of Teaching and Learning grant (for which I am also the Principal Investigator) to study how we can improve recruitment and retention in our introductory courses. I am also helping to develop CPSC 101 (cross-listed with Women's Studies 201), which is designed to provide an open, diverse introduction to the field. Indeed, 60% of the enrolment in my summer 2006 and spring 2007 CPSC 101: Connecting with CS courses was women, compared with roughly 30% during an average offering of our introductory programming course and even lower numbers later in the curriculum. In 2006, I attended the Grace Hopper Celebration of Women in Computing conference to learn more about recruitment and retention issues, and in 2007 and 2008, I presented CPSC 101 and general UBC CS recruitment and retention work to the SIGCSE CS Education community in a panel, a birds-of-a-feather session, and a poster.

Enhancing the Computer Science Education Community

A variety of research has shown that students who socialize over their discipline outperform those who do not and that supportive communities are key to recruiting and retaining a diverse student population. In response to this, I have worked to create a tight and supportive Computer Science Education community.

One key contribution in this area has been the establishment in the summer of 2006 of an ongoing weekly lunch with undergraduate students (the "Eccentric Loup Munches", an anagram for "Computer Science Lunch"). The lunches bring together students from across the CS spectrum—including first-year and upper-level undergraduates, graduate students, and students from related areas like mathematics and cognitive science—to socialize with each other and discuss topics of interest. Each week, the students select a CS topic to discuss (from quantum computing and category theory to nifty programming tricks and CS-related board games). Whenever possible, I or one of the students arranges for a faculty member or other expert in the chosen topic to attend. Students learn CS material during these lunches; indeed, watching such a diverse group of students get excited about (for example) the connection between fourth-year CS theory and cryptography is incredibly rewarding! Even more importantly, however, they establish social ties that will help them in their studies and careers. For example, one introductory programming student is now regularly mentored by a sympathetic graduate student he met at the lunches and, in turn, provided a key reference that he had located for a previous lunch to a student in my Artificial Intelligence class. In another example, a mathematics graduate student found key mentoring she needed from faculty and students she met through the lunches to connect her mathematics thesis to related CS topics.

I also helped our CS students build and communicate their sense of community by running the vPortfolio project with the help of an undergraduate directed studies student. vPortfolio was a term-long program in which 18 CS students in 6 teams learned the tools and techniques of film production and applied them to create short films with CS themes ranging from the trials and tribulations of tech industry job interviews to the addictive nature of computer games. Students worked with speakers from the film industry and the UBC film program. Their final videos were feted at a gala reception attended by more than 80 students, faculty, and community members and posted to YouTube (under the keyword "vPortfolio").

With the help of two undergraduate students and one graduate student, I sponsored and ran the "Jade Brigade", which introduced nine female undergraduate students, ranging from first year to graduating, to the academic community through a trip to SIGCSE 2008. The trip was both an opportunity for the students to network with each other and with students and faculty from other universities at the conference. Students completed a worksheet during the conference that required them to meet new people—including luminaries of the community—and discuss their work, career paths, and life stories (i.e., to dive into the "hallway track") as well as reflecting on the talks they attended in the technical tracks.

Besides the lunches and the vPortfolio project, I have worked to improve the CS Education community by: establishing regular biweekly meetings among the department's teaching-oriented faculty; mentoring for the department's Tri-Mentoring program; sponsoring two undergraduate research assistants as participants in the UBC Multidisciplinary Undergraduate Research Competition; securing external funding for over a dozen students to attend Computer Science conferences; recruiting and hosting our first CS Education speaker in the department's Distinguished Lecture Series; attending a Positive Space workshop; and organizing and participating in a variety of outreach events to prospective, current, and former students while I was Chair of the department's Communications Committee.

Improving Education through Technology

Computing technology has and will continue to radically change education. My PhD dissertation contributed a popular system for flexible and interactive lecture presentation. While at UBC, I have continued to innovate with and adopt educational technology, including: a Tablet PC loaner program I initiated to encourage faculty and students to explore the educational potential of the device; extensive use of WebCT, such as private instructional staff and project team discussion topics and online formative surveys; use and analysis of a system for automated individual feedback on programming assignments in my Fall 2006 CPSC 111: Intro. to Computation course; and most recently use of PRS clickers in my Spring and Fall 2007 CPSC 101: Connecting with Computer Science courses. I previously supervised an undergraduate research assistant integrating camera phones as student devices into my own interactive lecture presentation system. That student's work led directly to an ongoing grant and research project led by researchers at the University of California, San Diego. I am currently co-supervising a Ph.D. student whose work helps instructors make pedagogically valuable use of multiple projectors in classrooms (and prototyping his software in my classes).

Curriculum Development

As a core member of Computer Science's Carl Weiman Science Education Initiative committee, I am working on a broad revision of the entire CS curriculum based on careful articulation of our goals, assessment of students' achievement, and understanding of modern teaching techniques. My development work in CPSC 101 is our prototype model for progress on the CWSEI project.

I have also contributed substantially to curriculum development in all the courses I've taught. In each class, I keep detailed "post-mortem" notes after my lectures and wrap these and other materials up into a post-mortem package for the whole term to hand off to subsequent instructors. Other instructors for CPSC 111, 121, and 312 have made extensive use of resources I passed on in these packages. Instructors of CPSC 444 and 344 (which was based on 444) found my post-mortem package helped considerably in curriculum revisions for those courses. Specific contributions to my courses include:

- I am part of an ongoing effort to revise the curriculum for CPSC 101: Connecting with CS, for which I have already designed learning goals for almost all of the course's modules and made substantial modifications to some of the lecture slides and labs. One representative change was to refocus our "Minds and Machines" module so that students read primary source material from Alan Turing and John Searle and then debate the intelligence of various real and imagined computer systems. The sole learning goal for this module is now for students to be able to "argue articulately for and against the intelligence of proposed systems (human, computer, or combinations) in your own words but appealing to the ideas laid out by Turing and Searle." I have also introduced pen-and-paper and PRS exercises throughout the course that encourage active student learning. Based on student feedback, I shifted the PRS exercises from their original quiz-like format to open-ended questions (often with no right answer) that introduce and frame material. For example, students might predict the result of a piece of code that uses a simple loop *before* we discuss loops for the first time. We then discuss how different models of the code's semantics could lead to the different predictions, which model(s) seem most natural, and which model matches the language's implementation, revisiting the question several times during our discussion. From a logistical standpoint, I improved the course's flow by retooling two labs previously run in Java to JavaScript (which students already learn in the first half of the course). Eliminating the dependency on an introduction to Java (which was so rushed that it had little pedagogical value) gives the instructor greater freedom to reorder course material and extra time to address more realistic learning goals.

  Independent studies of my use of learning goals in CPSC 101 verified that students appreciate the learning goals, that they rely heavily on the goals, and that the goals help students focus both during class time and during exam preparation time.

- In CPSC 312: Functional and Logic Programming, I redesigned the entire set of lecture slides to alter presentation style and introduce new active learning exercises (such as one that connects predicate logic to Grouse Mountain ski routes) and compelling examples (such as an overview of the formative research paper on automated logic programming). I reworked most assignments from scratch, created entirely new exams, and designed a substantial new course project in which students use Princeton's WordNet software to ask and answer rich natural language questions.

- In CPSC 121: Models of Computation, I designed a full term's slides based on the existing handwritten notes to ease sharing of lecture materials, to integrate active learning exercises directly into the presentation, and to explicitly link the course's labs to lectures (which had been identified in previous research as a key weakness of the course). I composed learning goals (as part of a formal CS-Science Education Initiative triad) and used them to highlight previously underemphasized components of the curriculum, e.g., requiring students to judge the clarity and correctness of existing proofs. I added new questions to homework assignments that explicitly tied discrete mathematics concepts to CS problems and ideas. I also helped revise the labs and exams and clarified assignment marking schemes for the students and the TAs. I attribute high course evaluations in this often ill-received course in part to my careful preparation of well-motivated and contextualized material.

- In CPSC 111: Intro. to Computation, I have designed an entire slate of assignments (each emphasizing the power and real-world connections of CS) from scratch, developed an almost entirely code-based teaching style for the course with heavily annotated notes, developed a series of midterm and final examinations, introduced a system for automated feedback on programming exercises, introduced a low-stakes weekly formative assessment (the "weekly reading questions"), and established good lines of communication and accountability for the numerous instructional staff. During two terms of CPSC 111 offerings, I have been the head instructor for the course, which entails substantial additional organizational and administrative duties (especially in the fall term offering I led with four instructors, fifteen TAs, and over 400 students).

- In CPSC 344/444: User Interface Design, I introduced a series of active learning exercises (which were enthusiastically re-adopted during subsequent curriculum revision), an entirely revised module on affective computing based on current research, and designed mostly new examinations. Student feedback in this course frequently cited the innovative interactive exercises I introduced as the most important and memorable components of the lectures. Also based on student feedback, I developed and shared with students clearly specified marking schemes for the course's highly open-ended projects (and shared these with subsequent course instructors).

- In CPSC 221: Data Structures and Algorithms, I introduced a series of running examples to reinforce class material and connect it to interesting problems, including a Kinesthetic Learning Activity for teaching hashing, a series of exercises related to the British Columbia Single Transferable Vote system, and a course-long example related to a clever magic trick with cards. I also designed new examinations and some new course slides and assignments.

- In CPSC 311: Intro. to Programming Languages and 322: Intro. to Artificial Intelligence, I introduced a series of examples of how programming languages and artificial intelligence (respectively) connected with other disciplines and ongoing research.

- In CPSC 313: Computer Hardware and Operating Systems, I drafted learning goals for the course, tuned the lectures to address those goals, and designed significant new assessment materials based on the goals. For example, my new final exam and its corresponding full-length practice exam are both annotated for each question with which learning goal the question targets. Students have told me that this helps to focus their studies, feedback that conforms to study results from my use of learning goals in CPSC 101. I also introduced a series of "challenge problems" (worth a small amount of credit) that encourage students to work with concepts and tools beyond the base set of learning goals assessed in the rest of the course. Although not all students accepted these challenges, those that did said they learned tools that will help them in the remainder of their careers (such as debuggers) and concepts that eluded them in lecture (such as the program call stack).

b) Courses Taught at UBC

Session	Course	Enrollment	Hours Taught per Week			
			Lecture	Tutorial	Lab	Office Hrs
Winter 2 2007-2008	CPSC 121: Models of Computation	67	3	0	0	2
Winter 2 2007-2008	CPSC 313: Computer Hardware and Operating Systems	78	3	0	0	2
Winter 1 2007-2008	CPSC 101/WMST 201: Connecting with CS	106	3	0	0	2.5
Winter 1 2007-2008	CPSC 344: Intro to HCI Methods	46	2	0	0	2.5
Winter 2 2006-2007	CPSC 101: Connecting with CS	129	3	0	0	3
Winter 1 2006-2007	CPSC 322: Intro Artificial Intelligence	93	3	0	0	2
Winter 1 2006-2007	CPSC 111: Intro to Computation	144	3	0	0	2
Summer 1 2006	CPSC 101: Connecting with CS	51	7.5	0	0	4
Winter 2 2005-2006	CPSC 121: Models of Computation	87	3	0	0	2
Winter 2 2005-2006	CPSC 111: Intro to Computation	82	3	0	0	2.5
Winter 1 2005-2006	CPSC 311: Intro Prog Languages	45	3	0	0	2
Winter 1 2005-2006	CPSC 111: Intro to Computation (BCS section)	21	3	0	0	1.5
Summer 1 2005	CPSC 221: Data Structures and Algs	49	7.5	0	0	2
Winter 2 2004-2005	CPSC 312: Functional & Logic Prog	50	3	0	0	2
Winter 2 2004-2005	CPSC 444: User Interface Design	37	3	0	0	1
Winter 1 2004-2005	CPSC 312: Functional & Logic Prog	44	3	0	0	2
Winter 1 2004-2005	CPSC 111: Intro to Computation (BCS section)	13	3	0	0	2

(c) Graduate Students Supervised and/or Co-Supervised

Role	Student	Degree	Supervisor	Thesis Title	Dates	
					Start	End
Committee Member	Joel Lanir	Ph.D. (CS)	Kellogg Booth	Presentation Tools for High-Resolution and Multiple Displays (*in progress*)	Jan 2007	*current*
Second Reader	Yamin Htun	M.Sc. (CS)	Kellogg Booth and Joanna McGrenere	The Annotator's Perspective on Co-Authoring with Structured Annotations	Jun 2007	Sep 2007
Second Reader	Qixing Zheng	M.Sc. (CS)	Joanna McGrenere	Structured Annotations to Support Collaborative Writing Workflow	Mar 2005	Dec 2005
Second Reader	Joseph Luk	M.Sc. (CS)	Karon MacLean	Using Haptics to Address Mobile Interaction Design Challenges	Oct 2005	Jul 2006

(e) Visiting Lecturer (indicate university/organization and dates)

Co-Instructor, CSE142 Introduction to Programming — UW, Seattle, WA — 2001
Taught a class section of over 200 students (one of two sections). Shared all other course duties including managing 19 TAs and other staff. Representative materials and reviews available upon request. Overall student rating: 4.54/5.0 (weighted rating: 4.94). URL: http://www.cs.washington.edu/education/courses/cse142/01sp/

Instructor, CSE326 Data Structures and Algorithms — UW, Seattle, WA — 2000
Taught a class of 55 students. Designed syllabus, delivered lectures, managed staff (two TAs), and performed all other duties of instruction. Representative materials and reviews available upon request. Overall student rating: 4.94/5.0 (weighted rating: 5.15). URL: http://www.cs.washington.edu/education/courses/cse326/00wi/

(f) Other: Undergraduate Students Supervised

Christopher Head, Undergraduate Academic Assistant, 2006/2007 Winter Term 1-current,
Flexible Framework for Creative, Collaborative Introductory CS Assignments.
Presenter, SIGCSE 2008 Symposium on Computer Science Education

Elizabeth Patitsas, Science One project (co-supervised w/Mark MacLean), 2007/8 Winter Term 2,
O(n lg n) Insertion Sort using Library Sort.

Wei "Athena" Li , Directed Studies Thesis (CPSC 448), 2007 Summer-2007/2008 Winter Term 1, The Insides of vPortfolio's Successful Debut.

Piam Kiarostami & Lisa Frey (co-supervised w/Anne Condon), Undergrad Academic Assts, 2007 Summer, CPSC 101 Curriculum Development

Erica Huang, Honours Thesis (CPSC 449), 2005/6 Winter Term 1-2005/6 Winter Term 2,
Mobile Phone Keypad Design for Fast Chinese Text Entry by Phonetic Spelling.
Winner Best Undergraduate Research, Grace Hopper 2006 ACM Student Research Competition
Presenter, UBC 2006 Multidisciplinary Undergraduate Research Program & Competition

Tian Lim, Co-op/URA, 2005 Summer-2005/6 Winter Term 2,
Presenter-On-Paper: The Camera Phone as an In-Class Educational Technology Tool.
Presenter, UBC 2006 Multidisciplinary Undergraduate Research Program & Competition

9. Scholarly & Professional Activities

(a) Areas of special interest and accomplishments

Human-computer interaction, artificial intelligence, Computer Science pedagogy, educational technology

(b) Research or equivalent grants

Granting Agency	Project Title	COMP	Amount	Years		Principal Investigator	Co-Investigators
				Start	End		
Jade Project	A First Conference for Undergraduate CS Students	C	$3700	2007	2008	Steven Wolfman	
Skylight, UBC CS	Flexible Framework for Creative, Collaborative Introductory CS Assignments	C	$5770	2007	2008	Steven Wolfman	
Skylight, Jade Project	A Modular Approach to Connecting With Computer Science	C	$5000	2007	2008	Steven Wolfman	Anne Condon
Skylight, UBC CS	vPortfolio – enhancing student learning through video portfolios	C	$5057	2006	2008	Steven Wolfman	Paul Carter, Kurt Eiselt, Gayle Mavor
UBC Institute for the Scholarship of Teaching and Learning	Research Collaboration Project: Retaining Women in Computer Science	C	~$12000	2006	2007	Steven Wolfman (replacing Tamara Munzner)	Joanne Nakonechny, Michele Ng
Skylight, UBC CS	Computer Science Outreach and Curriculum Development	C	$4600	2006	2007	Paul Carter	Giuliana Villegas, Kurt Eiselt, Steven Wolfman, Michele Ng
UBC Teaching & Learning Enhancement Fund	Pioneering Team Based Learning and Studio Methods in Computer Science	C	$41,217	2006	2007	Joanna McGrenere	Kellogg Booth, Giuseppe Carenini, Cristina Conati, Karon MacLean, Ron Rensink, Steven Wolfman
Skylight, Jade Project, UBC CS	"What to Do with a CS Education" Speaker Series	C	$2250	2005	2006	Steven Wolfman	Michele Ng
SIGCSE Outreach Project Grant	Kinesthetic Learning Activities	C	$1000	2005	2005	Rebecca Bates, Steven Wolfman	

(d) Invited Presentations

UBC Skylight Science Supper: "Teaching Assumptions: Just Click Here." (invited panel)
University of British Columbia. Vancouver, BC. Jan 2008.

University of Washington "Exploring Faculty Careers in Higher Education" Seminar: Faculty Panel: Advice For and From New faculty (invited panel)

University of Washington. Seattle, USA. May 2006.

Beyond "Chalk and Talk": Using Tablet PCs to Engage Students and Improve Student Understanding (invited talk)
Duke 2006 Instructional Technology Showcase. Durham, USA. Apr 2006.

Nifty Assignments (invited/peer-reviewed panelist)
Moderator: Nick Parlante. With Stuart Reges and Eric Roberts. SIGCSE'06: Technical Symposium on Computer Science Education. Houston, USA. Mar 2006.
http://nifty.stanford.edu/

The Learning Experience Project — Using Tablet PCs and Conferencing to Change the Classroom (invited talk and demonstration)
With Jay Beavers and Loring Holden. Microsoft Research Faculty Summit Plenary Session. Redmond, USA. Jul 2003.

Interaction/Use of Small Groups in Large Classes (invited panelist)
With Ann Baker on panel: "Interacting with Students in Large Classes". UW Center for Instructional Development and Research, Quarterly Forum. Seattle, USA. October 2002.

Motivating Active and Group Learning (invited panelist)
With Laurie Murphy, Kenneth Blaha, Tammy VanDeGrift, and Carol Zander on panel: "Active and Cooperative Learning Techniques for the Computer Science Classroom". CCSC-NW'02: Consortium for Computing Sciences in Colleges, Northwest Region. Seattle, USA. Oct 2002. (Notes published in *Journal of Computing Sciences in Colleges* 18(2):92–94, Dec 2002.)

(e) Other Presentations

New Paradigms for Introductory Computing Courses
(peer-reviewed birds-of-a-feather session; moderator)
With Heidi Ellis, Charles Kelemen, and Curt White. SIGCSE'07: Technical Symposium on Computer Science Education. Covington, KY, USA. Mar 2007.

New Paradigms for Introductory Computing Courses (peer-reviewed panel)
With Elliot Koffman (moderator), Heidi Ellis, Charles Kelemen, and Curt White. SIGCSE'07: Technical Symposium on Computer Science Education. Covington, KY, USA. Mar 2007.

Kinesthetic Learning in the Classroom (peer-reviewed workshop)
With Andrew Begel and Daniel D. Garcia. SIGCSE'06: Technical Symposium on Computer Science Education. Houston, USA. Mar 2006.

Kinesthetic Learning in the Classroom (peer-reviewed special session)
With Rebecca Bates. CCSC-NW'05: Consortium for Computing Sciences in Colleges, Northwestern Conference. Bothell, WA, USA. Oct 2005.

Kinesthetic Learning in the Classroom (peer-reviewed workshop)
With Andrew Begel and Daniel D. Garcia. SIGCSE'05: Technical Symposium on Computer Science Education, page 566. St. Louis, USA. Feb 2005.

Kinesthetic Learning in the Classroom (peer-reviewed special session)
With Andrew Begel and Daniel D. Garcia. SIGCSE'04: Technical Symposium on Computer Science Education, pp. 183–184. Norfolk, USA. Mar 2004.

A Real-Time, Unobtrusive, and Contextual Feedback System for the Classroom (peer-reviewed doctoral consortium)
Presentation as a member of the Doctoral Consortium of SIGCSE'02: Technical Symposium on Computer Science Education. Cincinnati, USA. Feb 2002

(g) Conference Participation (Organizer, Keynote Speaker, etc.)

Symposium Co-Chair, SIGCSE Technical Symposium on CS Education 2010

Symposium Chair (w/Gary Lewandowski) and member of the conference organizing committee. Mar 2010.

Program Co-Chair, SIGCSE Technical Symposium on CS Education 2009

Program Chair (w/Gary Lewandowski) and member of the program committee and conference organizing committee. Mar 2009.

Workshops Coordinator, SIGCSE Technical Symposium on CS Education 2008

Workshop Coordinator and member of the program committee and conference organizing committee. Mar 2008.

Judge, Grace Hopper 2006 ACM Student Research Competition

Judge for the graduate student division (posters and semi-finalist presentations) of the Student Research Competition at the Grace Hopper Celebration of Women in Computing. Oct 2006.

Volunteer Coordinator, SIGCSE Technical Symposium on CS Education 2006

Volunteer Coordinator (w/Lisa Kaczmarczyk) and member of the conference organizing committee: recruited, managed, and organized a team of more than 60 volunteers to provide support for the premier Computer Science Education conference with more than one thousand attendees. Feb 2006.

Keynote Panelist, University of Washington TA Conference 2003

Presented a talk entitled "Teaching to Inspire, a Pyramid Scheme that Works" and participated on Q&A panel for audience of all new UW graduate TAs. Sep 2003.

Other conference attendance and presentations (since August 2004):

Role	Conference or Other Event	Date
Presenter	SIGCSE Technical Symposium on CS Education, Covington, KY	Mar, 2007
Participant	Grace Hopper Celebration of Women in Computing, San Diego, CA	Oct, 2006
Presenter	Duke Instructional Technology Showcase, Durham, NC	Apr, 2006
Presenter	SIGCSE Technical Symposium on CS Education, Houston, TX	Mar, 2006
Presenter	Northwest Regional Conference of the Consortium for Computing Sciences in Colleges	Oct, 2005
Presenter	U. of Washington/Microsoft Research Tablet PC Workshop, Seattle, WA	Jul, 2005
Participant	Microsoft Research Faculty Summit, Bellevue, WA	Jul, 2005
Presenter	SIGCSE Technical Symposium on CS Education, Houston, TX	Mar, 2005
Participant	Western Canadian Conference on Computing Ed., Prince George, BC	May, 2005

Role	Conference or Other Event	Date
Participant	Conference on Object-Oriented Programming, Systems, Languages, and Applications, Vancouver, BC	Oct, 2004
Presenter	U. of Washington/Microsoft Research Tablet PC Workshop, Seattle, WA	Jul, 2004

10. Service to the University

(a) Memberships on committees, including offices held and dates

Departmental:

2008-current	**Curriculum Committee**
2008-current	**Computer Science Science Education Initiative Subcommittee (chair)**
2007-current	**CS Liaison to Carl Wieman Science Education Initiative**
2006-2007	***ad hoc* CS Carl Wieman Science Education Initiative Committee**
2006-current	***ad hoc* CPSC 101 Curriculum Revision Group**
2005-2007	**Communications Committee (chair)**
2005-2007	**Focus on Women in Computing Committee**
2005-2006	***ad hoc* High School Outreach Committee**

Faculty:

2007-2008	**Faculty of Science Killam Teaching Award Selection Committee**

University of Washington, Departmental, Graduate Student Committees:

2002	**Graduate Admissions**
2001	**Graduate Student Recruiting (chair)**
1999	**Graduate Student Orientation (chair)**
1998	**Graduate Student Orientation**

(b) Other service, including dates

2008	**UBC LEAD Educational Initiative Participant**
2007	**UBC CS Imagine Presentation (faculty organizer and presenter)**
2007	**Teaching mentor to KangKang Yin for CPSC 101, Summer 2007**
2007	**HCI/Industry Colloquium (Organizer and Master of Ceremonies)**
2007	**CS Representative to Science Ugrad Society Events, including "Meet the Profs"**
2006-2007	**Mentor for UBC CPSC Tri-Mentoring Program**
2005-2007	**CS 111 Challenge Exam (designed, invigilated, & marked)**
2006	**UBC CS Undergraduate Welcome Ceremony (Master of Ceremonies)**
2006	**UBC CS Prospective Undergraduate Visit Day (volunteer organizer)**
2006-2007	**UBC CS CSSS/CSGSA Industry Panel (Master of Ceremonies)**
2006	**UBC CS February Discovery Forum (Master of Ceremonies)**
2005-2006	**Mentor for UBC CPSC Tri-Mentoring Program**
2005-2006	**Various High School Outreach Presentations**
2005-2006	**UBC CS ChicTech Program for K-12 Girls in CS (co-organizer)**
2005-2006	**UBC CS Alumni Panel Series (co-organizer & Master of Ceremonies)**
2005	**UBC CS Imagine Presentation (co-organizer & co-presenter)**

2005	**UBC CS Undergraduate Welcome Ceremony (organizer & Master of Ceremonies)**

11. Service to the Community

(a) Memberships on scholarly societies, including offices held and dates

ACM	**Member**
ACM SIGCSE	**Member**
ACM SIGCHI	**Member**

(f) Reviewer (journal, agency, etc. including dates)

2007-current	**JERIC** (Journal on Educational Resources in Computing); **member of editorial board**
2008	**Ubicomp** (Int'l Conf. on Ubiquitous Computing); **1 paper refereed**
2007	**ICER** (Int'l Computing Education Research Workshop); **1 paper refereed**
2007	**ITiCSE** (Innovation and Technology in CS Education); **3 papers refereed**
2007	**CHI** (Human Factors in Computing); **2 papers refereed**
2007	**SIGCSE** (Computer Science Education SIG); **4 papers refereed**
2006	**SIGCSE** (Computer Science Education SIG); **3 papers refereed**
2006	**ITiCSE** (Innovation and Technology in CS Education); **2 papers refereed**
2005	**FIE** (Frontiers in Education); **3 papers refereed**
2005	**ITiCSE** (Innovation and Technology in CS Education); **4 papers refereed**
2005	**CHI** (Human Factors in Computing); **1 paper refereed**

12. Awards and Distinction

(a) Awards for Teaching (indicate name of award, awarding organizations, date)

2006/2007 Faculty of Science Killam Prize for Excellence in Teaching
(Awarded annually by the UBC Faculty of Science to ~three faculty in Science.)

UBC CPSC Incredible Instructor Award, 2004-2006
(Awarded annually by the department to approximately four faculty.)

University of Washington Excellence in Teaching Award, 2002
(Awarded annually by the President of the UW to two graduate TAs.)

College of Engineering Teaching Assistant Recognition Award, 2002
(Awarded annually by a committee of UW Engineering faculty to one TA.)

University of Washington Undergraduate ACM Teaching Award, 2000

The University of British Columbia

Publications Record

Steven Andrew Wolfman **Date**: 2008/08/31 **Initials**:

1. Refereed Publications

(a) Journals

A Study of Diagrammatic Ink in Lecture
Richard Anderson, Ruth Anderson, Crystal Hoyer, Craig Prince, Jonathan Su, Fred Videon, Steven Wolfman. Computers & Graphics, Vol. 29(4): 480–489, 2005.

Programming by Demonstration using Version Space Algebra
Tessa Lau, Steven A. Wolfman, Pedro Domingos, and Daniel S. Weld. *Machine Learning*, Vol. 53(1–2): 111–156, 2003.

Combining Linear Programming and Satisfiability Solving for Resource Planning
Steven A. Wolfman and Daniel S. Weld. *The Knowledge Engineering Review*, Vol. 16(1):85–99, 2001.

(b) Conference Proceedings
(* starred header indicates presented by Steven Wolfman; most papers use alphabetical author order)

Poogle and the Unknown-Answer Assignment: Open-Ended, Sharable CS1 Assignments
Christopher C. D. Head and Steven A. Wolfman. SICGSE'08: Technical Symposium on Computer Science Education, pp. 133–137. Portland, OR, USA. March 2008. (31% acceptance rate)

Speech, Ink, and Slides: The Interaction of Content Channels
Richard Anderson, Crystal Hoyer, Craig Prince, Jonathan Su, Fred Videon, and Steve Wolfman, pp. 796–803. MM'04: ACM International Conference on Multimedia. New York, USA. October 2004. (17% acceptance rate)

***A Study of Digital Ink in Lecture Presentation**
Richard J. Anderson, Ruth Anderson, Crystal Hoyer, Steven A. Wolfman. Long Paper, CHI'04: Human Factors in Computing Systems, pp. 567–574. Vienna, Austria. Apr 2004. (16% acceptance rate)

Experiences with a Tablet PC Based Lecture Presentation System in Computer Science Courses
Richard Anderson, Ruth Anderson, Beth Simon, Steven A. Wolfman, Tammy VanDeGrift, and Ken Yasuhara. SIGCSE'04: Technical Symposium on Computer Science Education, pp. 56–60. Norfolk, USA. Mar 2004. (28% acceptance rate)

***Promoting Interaction in Large Classes with Computer-Mediated Feedback**
Richard J. Anderson, Ruth Anderson, Tammy VanDeGrift, Steven A. Wolfman, and Ken Yasuhara. Short Paper, CSCL'03: Computer Support for Collaborative Learning, pp. 119–123. Bergen, Norway. Jun 2003. (30% acceptance rate; 25% as long papers & 5% as short papers)

***Interaction Patterns with a Classroom Feedback System: Making Time for Feedback**
Richard J. Anderson, Ruth Anderson, Tammy VanDeGrift, Steven A. Wolfman, and Ken Yasuhara. Interactive Poster, CHI'03: Human Factors in Computing Systems. Ft. Lauderdale, USA. Apr 2003. (38% acceptance rate among interactive posters)

***Making Lemonade: Exploring the Bright Side of Large Lecture Courses**
Steven A. Wolfman. SIGCSE'02: Technical Symposium on Computer Science Education, pp. 257–261. Cincinnati, USA. Feb 2002. (31% acceptance rate)

***Mixed Initiative Interfaces for Learning Tasks: SMARTedit Talks Back**
Steven A. Wolfman, Tessa Lau, Pedro Domingos, and Daniel S. Weld. IUI'01: Intelligent User Interfaces, pp. 167–174. Santa Fe, USA. Jan 2001. (32% acceptance rate)

***The LPSAT Engine & Its Application to Resource Planning**
Steven A. Wolfman and Daniel S. Weld. IJCAI'99: International Joint Conference on Artificial Intelligence, pp. 310–316. Stockholm, Sweden. Aug 1999. (26% acceptance rate)

(c) Other

An Issue of Identity: Women in Computer Science (concurrent session)
Suzanna Huebsch, Joanne Nakonechny, Michele Ng, and Steven Wolfman. 28th Annual Society for Teaching and Learning in Higher Education Conference. Winsdor, ON. June 2008.

A Survey of Practices, Effort, and Outcomes in UBC CS's Recruitment and Retention Programs (poster)
Pooja Viswanathan, Christine Kwan, Michele Ng, and Steven Wolfman. SICGSE'08: Technical Symposium on Computer Science Education, pp. 133–137. Portland, OR, USA. March 2008. (75% acceptance rate)

Understanding Diagrammatic Ink in Lecture (workshop paper)
Richard Anderson, Ruth Anderson, Crystal Hoyer, Craig Prince, Jonathan Su, Fred Videon and Steve Wolfman. AAAI Fall Symposium 2004 Workshop on Making Pen-Based Interaction Intelligent and Natural, pp. 22–28. Washington, D.C. October 2004.

Lecture Presentation from the Tablet PC (workshop paper)
Richard Anderson, Ruth Anderson, Crystal Hoyer, Beth Simon, Fred Videon, and Steve Wolfman. WACE'03: Workshop on Advanced Collaborative Environments. Seattle, USA. Jun 2003.

Activating Computer Architecture with Classroom Presenter (workshop paper)
Beth Simon, Richard Anderson, and Steve Wolfman. WCAE'03: Workshop on Computer Architecture Education. San Diego, USA. Jun 2003.

2. Non-Refereed Publications

(b) Conference Proceedings

Automatically Personalizing User Interfaces (invited paper)
Daniel S. Weld, Corin Anderson, Pedro Domingos, Oren Etzioni, Krzysztof Gajos, Tessa Lau, and Steve Wolfman. IJCAI'03: International Joint Conference on Artificial Intelligence. Acapulco, Mexico. Aug 2003.

Research on Statistical Relational Learning at the University of Washington (invited workshop paper)
P. Domingos, Y. Abe, C. Anderson, A. Doan, D. Fox, A. Halevy, G. Hulten, H. Kautz, T. Lau, L. Liao, J. Madhavan, Mausam, D. Patterson, M. Richardson, S. Sanghai, D. Weld and S. Wolfman. IJCAI'03 Workshop on Learning Statistical Models from Relational Data. Acapulco, Mexico. Aug 2003.

Classroom Presentation from the Tablet PC (invited poster)
Richard J. Anderson, Ruth Anderson, Tammy VanDeGrift, Steven A. Wolfman, and Ken Yasuhara. ITiCSE'03: Innovation and Technology in Computer Science Education. Thessaloniki, Greece. Jun–Jul 2003.

(c) Other

Tablet PC and Computing Curriculum (white paper)
Christine Alvarado (USD), Richard Anderson (UW), Ruth Anderson (UVA), Jane Prey (MSR), Beth Simon (UCSD), Joe Tront (VaTech), and Steve Wolfman (UBC). Microsoft Research University Relations Program White Paper. March 2005

3. Books

(c) Chapters

Learning Repetitive Text-editing Procedures with SMARTedit (invited chapter)
Tessa Lau, Steven A. Wolfman, Pedro Domingos, and Daniel S. Weld. In Lieberman, ed., Your Wish is My Command: Giving Users the Power to Instruct their Software, Morgan Kaufmann, 2001.