

Associate Professor, Teaching Stream
David R. Cheriton School of Computer Science
University of Waterloo

Contact Information

Full Name: David Andrew Douglas Tompkins
Citizenship: Canadian
Mailing Address: David R. Cheriton School of Computer Science
University of Waterloo
200 University Avenue West
Waterloo, ON, Canada N2L 3G1
Phone: (cell) +1 (226) 792-4335
(fax) +1 (519) 885-1208
email: dtompkins@uwaterloo.ca
website: <http://www.cs.uwaterloo.ca/~dtompkin/>

Education

2010 PhD, Computer Science
University of British Columbia
Dynamic Local Search for SAT: Design, Insights and Analysis
Supervisor: Holger H. Hoos

2000 MSc, Electrical & Computer Engineering
University of British Columbia
Rate Control in Bi-Level Image Coding
Supervisor: Faouzi Kossentini

1996 BSc, Electrical Engineering
University of Western Ontario
Gold Medallist (top graduating student)

1994 BSc, Computer Science
University of Western Ontario

Academic Employment History

University of Waterloo

2024.09–present Associate Professor, Teaching Stream
2019.07–2024.08 Continuing Lecturer
2011.09–2019.06 Lecturer
(on parental leave from 2013.09–2013.12 and 2018.10–2019.02)

University of British Columbia

2010.09–2011.08 Postdoctoral Research & Teaching Fellow
1998.01–2010.08 Research Assistant, Teaching Assistant, Sessional Lecturer (intermittent appointments)
(on leave from 2000.08–2001.08 and 2008.09–2009.08)

Other Relevant Employment History

2000.08–2001.08 Senior Software Developer
UB Video, Inc., Vancouver, BC
Development of libraries (JPEG2000, MPEG4) for PCs & embedded DSPs

1997.05–1997.12 Systems Analyst
Netcon Technologies, London, ON
Developed software and system solutions for clients

1996.05–1997.04 (Student Council) President
University Students' Council, UWO, London, ON
Managed a variety of people and services for a large organization

1995.01–1995.12 Systems Analyst
MIS Solutions, Inc., London, ON
Developed software and system solutions for clients

1992.05–1994.08 IT Support & Development
Labatt Breweries of Canada, London, ON
IT Services: customer support, network administration, application development

1988.05– Sole Proprietor
synervations.com
Consulting and development with numerous projects and clients

Teaching Experience

Course Coordinator

2015.01–present CS 100 (Introduction to Computing through Applications)
2013.05–2022.08 CS 136 (Elementary Algorithm Design and Data Abstraction)
2013.05–2016.06 CS 135 (Designing Functional Programs)

Teaching Record – University of Waterloo

Term	Course	Class Size	Lecture hours/week	“Effectiveness” Evaluation †
2025 Fall	CS 100	62	3.0	
2025 Spring	CS 100	37	3.0	
2025 Winter	CS 100	92	3.0	
2024 Fall	CS 100	88	3.0	
2024 Fall	CS 100	51	3.0	
2024 Spring	CS 100	57	3.0	
2024 Winter	CS 100	162	online	
2023 Fall	CS 100	234	online	
2023 Spring	CS 100	186	online	
2023 Winter	CS 100	264	online	
2023 Winter	CS 136	60	1.5	
2023 Winter	CS 136	60	1.5	
2023 Winter	CS 136L	60	2.0	
2023 Winter	CS 136L	60	2.0	
2022 Fall	CS 135	95	3.0	
2022 Fall	CS 135	91	3.0	
2022 Fall	CS 100	287	online	
2022 Spring	CS 100	184	online	
2022 Winter	CS 136	88	3.0 [◊]	
2022 Winter	CS 136	74	3.0 [◊]	
2022 Winter	CS 136	35	3.0 [◊]	
2021 Fall	CS 136	195	online [◊]	
2021 Fall	CS 100	166	online	
2021 Spring	CS 100	170	online	
2021 Winter	CS 136	1439	online [◊]	
2020 Fall	CS 136	169	online [◊]	
2020 Fall	CS 100	193	online	
2020 Spring	CS 100	264	online	
2020 Winter	CS 136	88	3.0 [◊]	94
2020 Winter	CS 136	87	3.0 [◊]	99
2019 Fall	CS 136	72	3.0	91

Term	Course	Class Size	Lecture hours/week	“Effectiveness” Evaluation †
2019 Fall	CS 136	82	3.0	92
2019 Fall	CS 136	58	3.0	96
2019 Spring	CS 100	237	online	
2018 Spring	CS 136	88	3.0	95
2018 Spring	CS 136	85	3.0	96
2018 Winter	CS 136	90	3.0	95
2018 Winter	CS 136	90	3.0	97
2018 Winter	CS 136	88	3.0	98
2017 Fall	CS 135	83	3.0	96
2017 Fall	CS 135	80	3.0	98
2017 Fall	CS 100	389	online	
2017 Spring	CS 100	226	online	
2017 Winter	CS 136	91	3.0	98
2017 Winter	CS 136	88	3.0	98
2016 Fall	CS 136	78	3.0	95
2016 Fall	CS 136	89	3.0	99
2016 Fall	CS 136	88	3.0	96
2016 Fall	CS 100	357	online	
2016 Spring	CS 100	234	online	
2016 Winter	CS 136	89	3.0	99
2016 Winter	CS 136	90	3.0	98
2016 Winter	CS 100	344	online	
2015 Fall	CS 135	74	3.0	99
2015 Fall	CS 135	82	3.0	99
2015 Fall	CS 100	361	online	
2015 Spring	CS 100	225	online	
2015 Winter	CS 100	362	online	
2014 Fall	CS 135	116	3.0	96
2014 Fall	CS 135	108	3.0	98
2014 Fall	CS 135	117	3.0	96
2014 Winter	CS 136	109	3.0	98
2014 Winter	CS 136	110	3.0	98
2013 Spring	CS 136	89	3.0	93
2013 Spring	CS 350	53	3.0	95
2013 Spring	CS 350	56	3.0	94
2013 Winter	CS 136	103	3.0	97
2013 Winter	CS 490	25	3.0	97
2012 Fall	CS 135	92	3.0	96
2012 Fall	CS 350	60	3.0	97
2012 Spring	CS 350	56	3.0	94
2012 Spring	CS 350	37	3.0	91
2012 Winter	CS 115	89	3.0	96

Term	Course	Class Size	Lecture hours/week	“Effectiveness” Evaluation †
2012 Winter	CS 115	92	3.0	93
2012 Winter	CS 490	41	3.0	90
2011 Fall	CS 115	107	3.0	92
2011 Fall	CS 135	73	3.0	92

◊These terms were affected by COVID-19 and were temporarily taught online and/or were transitional terms that included both in-person and online components.

† “Effectiveness” Evaluation: average (mean) evaluation for the *Overall Effectiveness* question when asked (The University of Waterloo has stopped asking this question). Results are normalized to 100. For example, on a 5-point Likert scale, ‘Strongly Disagree’ is weighted 1 (20/100) and ‘Strongly Agree’ and is weighted 5 (100/100).

Full teacher evaluations are available online : <http://www.cs.uwaterloo.ca/~dtompkin/teaching>

Sessional Instructor / Postdoctoral Research & Teaching Fellow (UBC)

Term	Course	Class Size	Lecture hours/week	“Effectiveness” Evaluation †
2011 Summer	CPSC 121	63	7.5	70
2010 Fall	CPSC 211	70	3.0	84
2002 Fall	EECE 467	21	2.0	89
2002 Spring	EECE 467	50	2.0	91
2001 Fall	EECE 467	27	2.0	94
2001 Spring	EECE 467	50	2.0	96
2000 Fall	EECE 467	31	2.0	90
2000 Spring	EECE 467	33	2.0	85

Teaching Assistant (UBC)

Term	Course	“Effectiveness” Evaluation †
2008 Spring	CPSC 121	96
2007 Spring	CPSC 121	96
2006 Fall	CPSC 320	89
2006 Spring	CPSC 320	97
2005 Summer	CPSC 121	94
2005 Summer	CPSC 320	
2004 Fall	CPSC 320	90
2004 Spring	CPSC 220	
2003 Fall	EECE 466	
2002 Summer	EECE 466	
1999 Fall	EECE 467	

Course Calendar Descriptions (Waterloo)

- CS 100 **Introduction to Computing through Applications**
Using personal computers as effective problem solving tools for the present and the future. Effective use of spreadsheets to process, manipulate, and visualize numeric and textual information. Introduction to the Internet, World Wide Web, HTML, and XML. Algorithms underlying the functional components of web search engines and their influence on data access. Using wikis to publish, reshape, and organize data collaboratively.
- CS 115 **Introduction to Computer Science 1**
An introduction to the fundamentals of computer science through the application of elementary programming patterns in the functional style of programming. Function definition and application. Tracing via substitution. Design, testing, and documentation. Recursive data definitions. Lists and trees. Functional and data abstraction.
- CS 135 **Designing Functional Programs**
An introduction to the fundamentals of computer science through the application of elementary programming patterns in the functional style of programming. Syntax and semantics of a functional programming language. Tracing via substitution. Design, testing, and documentation. Linear and nonlinear data structures. Recursive data definitions. Abstraction and encapsulation. Generative and structural recursion. Historical context.
- CS 136 **Elementary Algorithm Design and Data Abstraction**
This course builds on the techniques and patterns learned in CS 135 while making the transition to use of an imperative language. It introduces the design and analysis of algorithms, the management of information, and the programming mechanisms and methodologies required in implementations. Topics discussed include iterative and recursive sorting algorithms; lists, stacks, queues, trees, and their application; abstract data types and their implementations.
- CS 136L **Tools and Techniques for Software Development**
This course introduces students to tools and techniques useful in the software development lifecycle. Students learn to navigate and leverage commands and utilities in the Linux Command Line Shell. Students gain experience in version control software, writing scripts to automate tasks, and creating effective test cases to identify bugs. Tracing and debugging strategies are discussed. Students also gain experience in using built-in support for version control, testing, debugging, build automation, etc. in integrated development environments (IDEs).
- CS 350 **Operating Systems**
An introduction to the fundamentals of operating system function, design, and implementation. Topics include concurrency, synchronization, processes, threads, scheduling, memory management, file systems, device management, and security.
- CS 490 **Information Systems Management**
The integration of business and technical considerations in the design, implementation and management of information systems. Topics include: IS planning and development; business, management, executive, and strategic information systems, including case studies of selected large- scale systems; decision support systems; end-user training and development; systems security, disaster planning and recovery. Practical examples of information systems in industry.

Course Calendar Descriptions (UBC)

- CPSC 101 **Connecting with Computer Science**
Fundamentals of computer science and their connections with the arts, psychology, and biology. Historical, cultural, and gender perspectives of important contributions to the field will be discussed. (Equivalency: WMST201)
- CPSC 121 **Models of Computation**
Physical and mathematical structures of computation. Boolean algebra and combinations logic circuits; proof techniques; functions and sequential circuits; sets and relations; finite state machines; sequential instruction execution.
- CPSC 211 **Introduction to Software Development**
Software design and the development of robust abstractions; design practices, data abstractions, inheritance, testing, concurrency and distributed computing.

- CPSC 220 Introduction to Discrete Structures
An introduction to computer science applications of discrete mathematics. Sets; logic; functions and relations; induction; program correctness; mathematical rigour; algorithms and applications.
- CPSC 320 Intermediate Algorithm Design and Analysis
Systematic study of basic concepts and techniques in the design and analysis of algorithms, illustrated from various problem areas. Topics include: models of computation; choice of data structures; graph-theoretic, algebraic, and text processing algorithms.
- EECE 466 Digital Signal Processing Systems
DSP fundamentals; digital filter FIR and IIR structures; filter design; DSP architectures; DSP applications.
- EECE 467 Real-Time Implementation of DSP Algorithms
DSP systems overview; DSP architectures; programming DSPs in 'C' and Assembly, I/O issues; real-time operating systems; host interfaces; code mapping and optimization; testing; DSP solutions in speech, audio, radar, telecommunications and control.

Awards and Scholarships

Performance Awards

- 2022 Outstanding Performance Award
(Faculty)
- 2019 Outstanding Performance Award
(Faculty)

Teaching Awards

- 2025 1st Place, Computer Science Teach-Off
(Departmental)
- 2009 UBC Killiam Graduate Teaching Award
\$1,000 (Institutional)
- 2007 UBC Computer Science Graduate TA Award
(Departmental)
- 2006 UBC Computer Science Graduate TA Award
(Departmental)

Best Paper Awards

- 2006 Best Paper Award
The Nineteenth Canadian Conference on Artificial Intelligence (AI-06)
(National Conference)

Best Poster Awards

- 2003 BC Advanced Systems Institute Exchange Communication (Poster) Award
\$500 (Regional)

1999 BC Advanced Systems Institute Exchange Communication (Poster) Award
\$200 (Regional)

Service Awards

2010 UBC Graduate Student Society Service Award
\$500 (Institutional)

2008 UBC Computer Science Service Award
(Departmental)

Service to the University Community

Administrative and Committee Chair Appointments

2024.07–present Chair, Faculty Council (Mathematics)

2021.09–present Associate Director, Undergraduate Studies (Computer Science)
Undergraduate Academic Plans Committee (ex officio)
Undergraduate Recruiting Committee (ex officio)

2022.01–2022.08 Chair, Undergraduate Recruiting Committee (Computer Science)

2015.09–2021.06 Chair, Commons Committee (Computer Science)

2012.01–2018.10 Undergraduate Academic Advisor (Computer Science)
Special Advisor for Double Degree Students (Waterloo BCS & Laurier BBA)

Selected Committee Memberships

2024.09–present Compensation Strategy Committee (Faculty Association)

2021.09–present Staff Hiring Committees (Various Positions)

2025.12–2026.02 Compensation Negotiation Team (Faculty Association)

2022.09–2025.08 School Advisory Committee on Appointments, Teaching Stream (Computer Science)

2013.07–2015.06 Commons Committee (Computer Science)

University of British Columbia

1998.01–2011.08 Alma Mater Society of UBC

1998.01–2009.12 Graduate Student Society of UBC

2003.01–2008.04 UBC President's Committee on the Naming of Facilities

2001.10–2006.12 Institute for Computing, Information & Cognitive Systems (ICICS)
Student Advisory Committee

2006.12–2006.12 ICICS Director Selection Committee

- 1999.03–2004.03 UBC Senate
Committees: Budget, Teaching and Learning, Library, Appeals on Academic Standing
- 1999.03–2004.03 UBC Faculty of Graduate Studies Council
- 2000.01–2000.03 UBC Teaching and Learning Enhancement Fund committee

Service to the Academic Community

Program Committees

- 2011 Twenty-Second International Joint Conference on Artificial Intelligence (IJCAI 11)

(Anonymous) Peer Reviewing

- 2014 Artificial Intelligence Journal
- 2013 Sixteenth International Conference on Theory and Applications of Satisfiability Testing (SAT 2013)
- 2013 Journal of Heuristics
- 2012 Fifteenth International Conference on Theory and Applications of Satisfiability Testing (SAT 2012)
- 2011 Ninth Metaheuristics International Conference (MIC 2011)
- 2011 Fourteenth International Conference on Theory and Applications of Satisfiability Testing (SAT 2011)
- 2011 Twenty-Second International Joint Conference on Artificial Intelligence (IJCAI 2011)
- 2008 Computers & Operations Research (Journal)
- 2008 Eleventh International Conference on Theory and Applications of Satisfiability Testing (SAT 2008)
- 2007 Twentieth International Joint Conference on Artificial Intelligence (IJCAI 2007)
- 2007 International Symposium on Combinatorics, Algorithms, Probabilistic and Experimental Methodologies (ESCAPE 2007)
- 2006 Twelfth International Conference on Principles and Practice of Constraint Programming (CP 2006)
- 2006 Ninth International Conference on Theory and Applications of Satisfiability Testing (SAT 2006)
- 2004 Nineteenth National Conference on Artificial Intelligence (AAAI 2004)
- 2003 Ninth International Conference on Principles and Practice of Constraint Programming (CP 2003)
- 2003 Sixth International Conference on Theory and Applications of Satisfiability Testing (SAT 2003)
- 2003 IEEE Transactions on Image Processing (Journal)

Professional Development

- 2012 Instructional Skills Workshop

- 2010 From TeachScheme to ReachJava: How to Design Programs (HtDP) teaching workshop.
- 2001 Stochastic local search workshop, IJCAI-01.
- 2001 Empirical analysis and methods workshop, IJCAI-01.
- 1997 Dale Carnegie course in professional communication.

Publications

Refereed Conference Proceedings

- [1] Sam Bayless, Dave A. D. Tompkins, and Holger H. Hoos. Evaluating instance generators by configuration. In Panos M. Pardalos, Mauricio G. C. Resende, Chrysafis Vogiatzis, and Jose L. Walteros, editors, *Proceedings of the Eighth International Conference on Learning and Intelligent Optimization (LION-14)*, volume 8426 of *Lecture Notes in Computer Science*, pages 47–61. Springer Berlin / Heidelberg, 2014. (*acceptance rate: 48%*).
- [2] Dave A. D. Tompkins, Adrian Balint, and Holger H. Hoos. Captain Jack: New variable selection heuristics in local search for SAT. In Karem Sakallah and Laurent Simon, editors, *Proceedings of the Fourteenth International Conference on Theory and Applications of Satisfiability Testing (SAT 2011)*, volume 6695 of *Lecture Notes in Computer Science*, pages 302–316. Springer Berlin / Heidelberg, 2011. (*acceptance rate: 44%*).
- [3] Dave A. D. Tompkins and Holger H. Hoos. Dynamic scoring functions with variable expressions: New SLS methods for solving SAT. In Ofer Strichman and Stefan Szeider, editors, *Proceedings of the Thirteenth International Conference on Theory and Applications of Satisfiability Testing (SAT 2010)*, volume 6175 of *Lecture Notes in Computer Science*, pages 278–292. Springer Berlin / Heidelberg, 2010. (*acceptance rate: 38%*).
- [4] Dave A. D. Tompkins and Holger H. Hoos. On the quality and quantity of random decisions in stochastic local search for SAT. In Luc Lamontagne and Mario Marchand, editors, *Proceedings of the Nineteenth Conference of the Canadian Society for Computational Studies of Intelligence (AI 2006)*, volume 4013 of *Lecture Notes in Artificial Intelligence*, pages 146–158. Springer Berlin / Heidelberg, 2006. (**Best Paper Award**) (*acceptance rate: 21%*).
- [5] Dave A. D. Tompkins and Holger H. Hoos. UBCSAT: An implementation and experimentation environment for SLS algorithms for SAT and MAX-SAT. In Holger Hoos and David Mitchell, editors, *Revised Selected Papers from the Seventh International Conference on Theory and Applications of Satisfiability Testing (SAT 2004)*, volume 3542 of *Lecture Notes in Computer Science*, pages 306–320. Springer Berlin / Heidelberg, 2005. (*acceptance rate: 33%*).
- [6] Dave A. D. Tompkins and Holger H. Hoos. Warped landscapes and random acts of SAT solving. In *Proceedings of the Eighth International Symposium on Artificial Intelligence and Mathematics (AI&MATH 2004)*, 2004. (*acceptance rate: 47%*).
- [7] Dave A. D. Tompkins and Holger H. Hoos. Scaling and probabilistic smoothing: Dynamic local search for unweighted MAX-SAT. In Yang Xiang and Brahim Chaib-draa, editors, *Proceedings of*

the Sixteenth Conference of the Canadian Society for Computational Studies of Intelligence (AI 2003), volume 2671 of *Lecture Notes in Artificial Intelligence*, pages 145–159. Springer Berlin / Heidelberg, 2003. (*acceptance rate: 28%*).

- [8] Frank Hutter, Dave A. D. Tompkins, and Holger H. Hoos. Scaling and probabilistic smoothing: Efficient dynamic local search for SAT. In Pascal Van Hentenryck, editor, *Proceedings of the Eighth International Conference on Principles and Practice of Constraint Programming (CP 2002)*, volume 2470 of *Lecture Notes in Computer Science*, pages 233–248. Springer Berlin / Heidelberg, 2002. (*acceptance rate: 30%*).
- [9] Dave A. D. Tompkins and Faouzi Kossentini. A fast segmentation algorithm for bi-level image compression using JBIG2. In *Proceedings of the 1999 IEEE International Conference on Image Processing (ICIP 1999)*, volume 1, pages 224–228. IEEE Computer Society Press, 1999. (**Invited Paper**).
- [10] Magesh Valliappan, Brian L. Evans, Dave A. D. Tompkins, and Faouzi Kossentini. Lossy compression of stochastic halftones with JBIG2. In *Proceedings of the 1999 IEEE International Conference on Image Processing (ICIP 1999)*, volume 1, pages 214–218. IEEE Computer Society Press, 1999. (**Invited Paper**).

Theses

- [11] Dave A. D. Tompkins. *Dynamic Local Search for SAT: Design, Insights and Analysis*. PhD Thesis, University of British Columbia, October 2010.
- [12] Dave A. D. Tompkins. *Rate Control in Bi-Level Image Coding*. Master of Applied Science, University of British Columbia, August 2000.

Technical Reports

- [13] Dave A. D. Tompkins. MAX-SAT 2012: ubcsat-irots. Solver Description, MAX-SAT 2012 Competition Booklet, 2012. (**Won Several Categories in the Incomplete Solver Track – 2012**).
- [14] Adrian Balint, Andreas Fröhlich, Dave A. D. Tompkins, and Holger H. Hoos. Sparrow2011. Solver Description, SAT 2011 Competition Booklet, 2011. (**Won 2 Gold Medals – 2011**).
- [15] Holger H. Hoos and Dave A. D. Tompkins. Novelty⁺ and Adaptive Novelty⁺. Solver Description, SAT 2004 Competition Booklet, 2004. (**Won 2 Gold Medals – 2004**).
- [16] Dave A. D. Tompkins, Frank Hutter, and Holger H. Hoos. Scaling and Probabilistic Smoothing (SAPS). Solver Description, SAT 2004 Competition Booklet, 2004. (**Won 2 Silver Medals – 2004**).
- [17] Dave Tompkins and Faouzi Kossentini. JBIG2 halftones: Analysis and considerations for T.89. Technical Submission, ITU-T SG8, November 1999.
- [18] Dave Tompkins and Faouzi Kossentini. Additional extension segments in JBIG2. Technical Report N1318, ISO/IEC JTC1 SC29 WG1, July 1999.

- [19] Faouzi Kossentini, Dave Tompkins, Soeren Forchhammer, Bo Martins, Ole Jensen, Ian Caven, and Paul Howard. JBIG-like coding of bi-level image data in JPEG-2000. Technical Report N1014, ISO/IEC JTC1 SC29 WG1, October 1998.
- [20] Dave Tompkins and Faouzi Kossentini. Coding of numerical data in JBIG-2. Technical Report N862, ISO/IEC JTC1 SC29 WG1, July 1998.

Posters & Non-Refereed Conference Papers

- [21] Dave Tompkins. Dynamic local search for SAT. Poster, British Columbia Advanced Systems Institute Exchange, 2003. (**Best Poster Award**).
- [22] Dave A. D. Tompkins and Faouzi Kossentini. Lossless JBIG2 coding performance. In James A. Storer and Martin Cohn, editors, *Proceedings of the 1999 Data Compression Conference (DCC 1999)*, page 553. IEEE Computer Society Press, 1999. (abstract).
- [23] Dave Tompkins. Document image compression. Poster, British Columbia Advanced Systems Institute Exchange, 1999. (**Best Poster Award**).

Selection of Invited Talks, Panels & Presentations

- 2024.09 “Panel: The Future of CS Education.” Cheriton Research Symposium.
- 2017.05 “SAT, Incomplete Algorithms and Graph Colouring.” Canadian Computing Olympiad.
- 2016.04 “Panel: Engage, Click, Discuss.” Math Teaching Series.
- 2010.11 “Scientific Citizens.” UBC Fall Graduation Ceremony.
- 2010.11 “The Satisfiability Problem.” Vancouver Mensa Group.
- 2005.03 “Adventures in SAT Solving.” UBC Graduate Recruitment Talk.
- 2000.04 “JBIG2 Halftoning.” University of Texas at Austin.