

Trevor A. Brown, Assistant Professor at the University of Waterloo

Cheriton School of Computer Science
200 University Ave W
Waterloo, ON N2L 3G1, Canada

E-mail: [me \[at\] tbrown \[dot\] pro](mailto:me[at]tbrown[dot]pro)
Personal web: <http://tbrown.pro>
Lab web: <http://mc.uwaterloo.ca>

Education

University of Toronto, Toronto, Canada, Jan. 2013 – Jun. 2017

Ph.D. Computer Science (supervised by Faith Ellen); A+ GPA

Thesis: Techniques for constructing efficient lock-free data structures (312 pages)

University of Toronto, Toronto, Canada, Sep. 2011 – Jan. 2013

M.Sc. Computer Science (supervised by Faith Ellen); A+ GPA

Thesis: Pragmatic primitives for non-blocking data structures

York University, First Class with Distinction, Toronto, Canada, 2006 – 2011

B.Sc. Hons. Major Computer Science, Hons. Minor Mathematics; A+ GPA

Research Interests

- Concurrent data structures
- Non-blocking algorithms
- Memory management
- Non-volatile memory
- Transactional memory

Employment

Assistant Professor. University of Waterloo (Sep 2018 - present).

Affiliated with Algorithms and Complexity, Systems and Networking, and Programming Languages groups. Leading the [Multicore Lab](#).

Postdoctoral Researcher. IST Austria (Oct 2017 – Aug 2018).

Working with Professor Dan Alistarh.

Postdoctoral Researcher. Technion, Israel Institute of Technology (Mar – Oct 2017).

Worked with Professor Hagit Attiya. (A short reciprocal visit to work with Maya Arbel.)

Research Intern. Oracle Labs East, Scalable Synchronization Group (Summer 2015).

Designed and implemented adaptive transactional lock-elision algorithms and work delegation algorithms for large scale systems with non-uniform memory architectures.

Application Developer. UPS Canada (2005 – 2006, 2008 – 2009).

Redesigned internal software to scale under a hugely increased user load.

Contributed over 200,000 lines of production code.

Conference and Journal Publications

1. **PathCAS: An efficient middle ground for concurrent search data structures.**
William Sigouin, Trevor Brown and Dan Alistarh. *Under submission.*
2. **NBR: Neutralizing based reclamation.**
Ajay Singh, Trevor Brown and Ali Mashtizadeh. *Under submission.*
3. **Memory tagging: minimalist synchronization for scalable concurrent data structures.**
Dan Alistarh, Trevor Brown and Nandini Singhal. [[Paper](#)] [[Slides](#)] [[Video](#)]
32nd ACM Symposium on Parallel Algorithms and Architectures (SPAA'20), pp 37-49.
(32% acceptance; Qualis A2-rank, ERA A-rank)
4. **Non-blocking interpolation search trees with doubly-logarithmic running time.**
Trevor Brown, Aleksandar Prokopec and Dan Alistarh. [[Paper](#)] [[Slides](#)] [[Audio](#)] [[Code](#)]
25th ACM Symp. On Princ. and Pract. of Parallel Programming (PPoPP'20), pp 4:(1-16)
Received the [best paper award](#). (23% acceptance; Qualis A2-rank, ERA A-rank)
5. **Snapshot-based synchronization: a fast replacement for hand-over-hand locking.**
Eran Gilad, Trevor Brown, Mark Oskin and Yoav Etsion. [[Paper](#)]
24th Intl. European Conf. on Parallel and Distr. Computing (**EUROPAR'18**), pp 465-479.
6. **Relaxed schedulers efficiently parallelize sequential algorithms.**
Dan Alistarh, Trevor Brown, Justin Kopinsky and Giorgi Nadiradze. [[Paper](#)]
37th ACM Symp. on the Principles of Distributed Computing (**PODC'18**), pp 377-386.
7. **Distributionally linearizable data structures.**
Dan Alistarh, Trevor Brown, Justin Kopinsky, Giorgi Nadiradze and Jerry Li. [[Paper](#)]
30th ACM Symp. on Parallelism in Algorithms and Architectures (**SPAA'18**), pp 133-142.
8. **Getting to the root of concurrent search tree performance.**
Maya Arbel-Raviv, Trevor Brown and Adam Morrison. [[Paper](#)] [[Slides](#)] [[Audio](#)]
2018 USENIX Annual Technical Conference (**USENIX ATC'18**), pp 295-306.
9. **Harnessing epoch-based reclamation for efficient range queries.**
Maya Arbel-Raviv and Trevor Brown. [[Paper](#)] [[Slides](#)] [[Code](#)]
23rd ACM Symp. on Princ. and Practice of Parallel Programming (**PPoPP'18**), pp 14-27.
10. **Reuse, don't recycle: transforming lock-free algorithms that throw away descriptors.**
Maya Arbel-Raviv and Trevor Brown. [[Paper](#)] [[Slides](#)] [[Video](#)] [[Code](#)]
31st ACM Symposium on Distributed Computing (**DISC'17**), pp 4:(1-16).
11. **Cost of concurrency in hybrid transactional memory.**
Trevor Brown and Srivatsan Ravi. [[Paper](#)] [[Slides](#)]
31st ACM Symposium on Distributed Computing (**DISC'17**), pp 9:(1-16).
12. **A template for implementing fast lock-free trees using HTM.**
Trevor Brown. [[Paper](#)] [[Slides](#)] [[Code](#)]
36th ACM Symp. on the Principles of Distributed Computing (**PODC'17**), pp 293-302.

13. **PHyTM: persistent hybrid transactional memory.**
Hillel Avni and Trevor Brown.
Very Large Data Bases Journal Volume 10(4), 12 pages.
Full version of the paper at VLDB'17.
14. **PHyTM: persistent hybrid transactional memory.**
Hillel Avni and Trevor Brown. [[Paper](#)] [[Slides](#)]
43rd International Conference on Very Large Data Bases (**VLDB'17**), pp 409-420.
15. **Investigating the performance of hardware transactions on a multi-socket machine.**
Trevor Brown, Alex Kogan, Yossi Lev and Victor Luchangco. [[Paper](#)] [[Slides](#)]
28th ACM Symp. on Parallelism in Algorithms and Architectures (**SPAA'16**), pp 121-132.
16. **Reclaiming memory for lock-free data structures: there has to be a better way.**
Trevor Brown. [[Paper](#)] [[Slides](#)] [[Code](#)]
34th ACM Symp. on the Principles of Distributed Computing (**PODC'15**), pp 261-270.
17. **B-slack trees: space efficient B-trees.**
Trevor Brown. [[Paper](#)] [[Slides](#)] [[Code](#)]
14th Scandinavian Symp. and Workshops on Algorithm Theory (**SWAT'14**), pp 122-133.
18. **A general technique for non-blocking trees.**
Trevor Brown, Faith Ellen and Eric Ruppert. [[Paper](#)] [[Slides](#)] [[Code](#)]
19th ACM Symp. on Princ. and Practice of Parallel Programming (**PPoPP'14**), pp 329-342.
19. **Pragmatic primitives for non-blocking data structures.**
Trevor Brown, Faith Ellen and Eric Ruppert. [[Paper](#)] [[Slides](#)] [[Code](#)]
31st ACM Symposium on the Principles of Distributed Computing (**PODC'13**), pp 13-22.
20. **Range queries in non-blocking k-ary search trees.**
Trevor Brown and Hillel Avni. [[Paper](#)]
16th International Conf. on Principles of Distributed Systems (**OPODIS'12**), pp 31-45.
21. **Non-blocking k-ary search trees.**
Trevor Brown and Joanna Helga. [[Paper](#)] [[Slides](#)] [[Video](#)] [[Code](#)]
15th International Conf. on Principles of Distributed Systems (**OPODIS'11**), pp 207-221.

Workshop Papers

22. **Cost of concurrency in hybrid transactional memory.**
Trevor Brown and Srivatsan Ravi. [[Paper](#)] [[Slides](#)]
12th ACM SIGPLAN Workshop on Transactional Computing (**TRANSACT'17**), 8 pages.
Preliminary version of the conference paper at DISC'17.
23. **Persistent hybrid transactional memory.**
Hillel Avni and Trevor Brown. [[Paper](#)] [[Slides](#)]
11th ACM SIGPLAN Workshop on Transactional Computing (**TRANSACT'16**), 8 pages.
Preliminary version of the conference paper at VLDB'17.

24. **Investigating the performance of hardware transactions on a multi-socket machine.**
Trevor Brown, Alex Kogan, Yossi Lev and Victor Luchangco. [[Paper](#)] [[Slides](#)]
11th ACM SIGPLAN Workshop on Transactional Computing (TRANSACT'16), 8 pages.
Preliminary version of the conference paper at SPAA'16.

Short Papers and Posters

25. **Reuse, don't recycle: transforming lock-free algorithms that throw away descriptors.**
Maya Arbel-Raviv and Trevor Brown. [[Paper](#)] [[Poster](#)]
22nd ACM Symp. on Princ. and Practice of Parallel Programming (PPoPP'17), pp 429-430.
Poster and short version of conference paper at DISC'17.
26. **Concurrent data structures.**
Faith Ellen and Trevor Brown. [[Paper](#)] [[Slides](#)]
35th ACM Symp. on the Principles of Distributed Computing (PODC'16), pp 151-153.
Short paper to accompany an invited talk by Faith Ellen at PODC'16.
27. **Faster data structures in transactional memory using three paths.**
Trevor Brown. [[Paper](#)] [[Slides](#)]
29th ACM Symposium on Distributed Computing (DISC'15), pp 671-672.
Short version of conference paper at PODC'17.
28. **A general technique for non-blocking trees.**
Trevor Brown, Faith Ellen and Eric Ruppert. [[Paper](#)] [[Slides](#)]
27th ACM Symposium on Distributed Computing (DISC'13), pp 567-568.
Short version of conference paper at PPoPP'14.

Articles and Technical Reports

29. **Analysis and Evaluation of Non-Blocking Interpolation Search Trees.**
Aleksandar Prokopec, Trevor Brown and Dan Alistarh. [[Paper](#)]
ArXiv Computing Research Repository (CoRR), abs/2001.00413, 17 pages.
30. **On the cost of concurrency in hybrid transactional memory.**
Trevor Brown and Srivatsan Ravi. [[Paper](#)]
ArXiv Computing Research Repository (CoRR), abs/1907.02669, 17 pages.
31. **Reuse, don't recycle: transforming lock-free algorithms that throw away descriptors.**
Maya Arbel-Raviv and Trevor Brown. [[Paper](#)]
ArXiv Computing Research Repository (CoRR), abs/1708.01797, 32 pages.
Full version of conference paper at DISC'17.
32. **A template for implementing fast lock-free trees using HTM.**
Trevor Brown. [[Paper](#)]
ArXiv Computing Research Repository (CoRR), abs/1708.04838, 20 pages.
Full version of conference paper at PODC'17.

33. **Techniques for constructing efficient lock-free data structures.**
PhD thesis, University of Toronto. [[Paper](#)]
Committee: Faith Ellen (University of Toronto), Azadeh Farzan (University of Toronto), Vassos Hadzilacos (University of Toronto), Maurice Herlihy (Brown University), Ryan Johnson (University of Toronto), Sam Toueg (University of Toronto).
ArXiv Computing Research Repository (CoRR), abs/1712.05406, 312 pages.
34. **Reclaiming memory for lock-free data structures: there has to be a better way.**
Trevor Brown. [[Paper](#)]
ArXiv Computing Research Repository (CoRR), abs/1712.01044, 27 pages.
Full version of conference paper at PODC'15.
35. **B-slack trees: space efficient B-trees.**
Trevor Brown. [[Paper](#)]
ArXiv Computing Research Repository (CoRR), abs/1712.05020, 19 pages.
Full version of conference paper at SWAT'14.
36. **A general technique for non-blocking trees.**
Trevor Brown, Faith Ellen and Eric Ruppert. [[Paper](#)]
ArXiv Computing Research Repository (CoRR), abs/1712.06687, 41 pages.
Full version of conference paper at PPOPP'14.
37. **Pragmatic primitives for non-blocking data structures.**
Trevor Brown, Faith Ellen and Eric Ruppert. [[Paper](#)]
ArXiv Computing Research Repository (CoRR), abs/1712.06688, 47 pages.
Full version of conference paper at PODC'13.
38. **Range queries in non-blocking k-ary search trees.**
Trevor Brown and Hillel Avni. [[Paper](#)]
ArXiv Computing Research Repository (CoRR), abs/1712.05101, 17 pages.
Full version of conference paper at OPODIS'12.
39. **Non-blocking k-ary search trees.**
Trevor Brown and Joanna Helga. [[Paper](#)]
York University Technical Report CSE-2011-04, 52 pages.
Full version of conference paper at OPODIS'11.

Patents

40. **Adaptive techniques for improving performance of hardware transactions on multi-socket machines.** Oracle Labs, US patent 10,127,088. 34 pages.
Alex Kogan, Yossi Lev, Victor Luchangco and Trevor Brown. [[Link](#)]

Conference Talks and Invited Presentations

Memory tagging: minimalist synchronization for scalable concurrent data structures.

- 32nd ACM Symposium on Parallel Algorithms and Architectures, *Virtual conference* (July 2020). [[Slides](#)] [[Video](#)]

Non-blocking interpolation search trees with doubly-logarithmic running time.

- 25th ACM Symposium on Principles and Practice of Parallel Programming, San Diego, United States (Feb 2020). *Best paper award*. [[Slides](#)] [[Audio](#)]

Invited to speak for 60 minutes at Hydra'20. However, I had to decline due to timing.

Scalable infrastructure for next-generation data management systems.

- Invited presentation at the 2nd Waterloo-Huawei Joint Innovation Workshop (164 registrants), Waterloo, Canada (Jun 2020).

Practical aspects of multicore programming.

- Invited talk at the 2nd Summer School on the Practice and Theory of Distributed Computing (200+ attendees), St. Petersburg, Russia (Jul 2019).

Rated 2nd best out of 9 invited talks by highly distinguished speakers, including a Turing award winner. [[Talk](#)] [[Speakers list](#)]

Getting to the root of concurrent search tree performance.

- 2018 USENIX Annual Technical Conference, Boston, United States (Jul 2018). [[Slides](#)]

Getting to the root of concurrent BSTs.

- TU Wien, Vienna, Austria (June 2018). [[Slides](#)]

Towards correct and efficient multicore programming.

- University of Waterloo, Waterloo, Canada (Mar 2018). [[Slides](#)]
- University of California Santa Cruz, California, United States (Feb 2018).
- Simon Fraser University, Burnaby, Canada (Feb 2018).
- University of Waterloo, Waterloo, Canada (Jan 2018).

Good data structure experiments are R.A.R.E.

- Invited talk at the 1st Workshop on the Theory and Practice of Concurrency, Vienna, Austria (Oct 2017). *Held in conjunction with DISC'17*. [[Slides](#)] [[Video](#)]
- Oath/Yahoo! Labs, Haifa, Israel (Sep 2017). [[Slides](#)]

Reuse, don't recycle: transforming lock-free algorithms that throw away descriptors.

- 31st ACM Symposium on Distributed Computing, Vienna, Austria (Oct 2017). [[Slides](#)] [[Video](#)]

- Riot Games, Los Angeles, California (Nov 2019) – pre-recorded video talk

Cost of concurrency in hybrid transactional memory.

- 31st ACM Symposium on Distributed Computing, Vienna, Austria (Oct 2017). [[Slides](#)]

A template for implementing fast lock-free trees using HTM.

- 36th ACM Symposium on the Principles of Distributed Computing, Washington, United States (Jul 2017). [[Slides](#)]

Techniques for Constructing Efficient Lock-free Data Structures.

- Technion - Israel Institute of Technology, Haifa, Israel (May 2017). [[Slides](#)]
- University of Toronto, Toronto, Canada (Mar 2017). [[Slides](#)]

Investigating the performance of hardware transactions on a multi-socket machine.

- 28th ACM Symposium on Parallelism in Algorithms and Architectures, Monterey, United States (Jul 2016). [[Slides](#)]
- 11th ACM SIGPLAN Workshop on Transactional Computing, Barcelona, Spain (Mar 2016). [[Slides](#)]

Persistent hybrid transactional memory.

- 11th ACM SIGPLAN Workshop on Transactional Computing, Barcelona, Spain (Mar 2016). [[Slides](#)]

Faster data structures in transactional memory using three paths.

- 29th ACM Symposium on Distributed Computing, Tokyo, Japan (Oct 2015). [[Slides](#)]

Scalable transactions on NUMA systems.

- Oracle Labs East, Burlington, United States (Aug 2015). [[Slides](#)]

Reclaiming memory for lock-free data structures: there has to be a better way.

- 34th ACM Symposium on the Principles of Distributed Computing, San Sebastian, Spain (Jul 2015). [[Slides](#)]

Hardware transactional memory and the lemming effect.

- University of Toronto, Toronto, Canada (Apr 2015). [[Slides](#)]

Java Experiments on MTL: From past mistakes to best practices.

- York University, Toronto, Canada (Mar 2015). [[Slides](#)]

Memory reclamation for lock-free data structures.

- University of Toronto, Toronto, Canada (Aug 2014).

B-slack trees: space efficient B-trees.

- 14th Scandinavian Symposium and Workshops on Algorithm Theory, Copenhagen, Denmark (Jun 2014). [[Slides](#)]

A general technique for non-blocking trees.

- 19th ACM Symposium on Principles and Practice of Parallel Programming, Orlando, United States (Feb 2014). [[Slides](#)]
- 27th ACM Symposium on Distributed Computing, Jerusalem, Israel (Aug 2013). [[Slides](#)]

Pragmatic primitives for non-blocking data structures.

- 31st ACM Symposium on the Principles of Distributed Computing, Montreal, Canada (Jul 2013). [[Slides](#)]
- University of Toronto, Toronto, Canada (Jul 2013).

Building a non-blocking chromatic tree.

- TransForm School on Research Directions in Distributed Computing, Heraklion, Crete (Jun 2013). [[Slides](#)]

Range queries in non-blocking k-ary search trees.

- 16th International Conference on Principles of Distributed Systems, Rome, Italy (Nov 2012).

Non-blocking k-ary search trees.

- 15th International Conference on Principles of Distributed Systems, Toulouse, France (Oct 2011). [[Slides](#)] [[Video](#)]

Experiences with Intel's Multicore Testing Lab.

- York University, Toronto, Canada (2011).

Teaching

- Teaching – University of Waterloo (2018 – present)
CS341: algorithms (W2020 – two sections)
CS798-003: (graduate) [multicore programming](#) (F2019).
CS341: algorithms (W2019).
CS798-003: (graduate) multicore programming (F2018).
[Reading Group](#) attended by up to 15 students from multiple research groups
- *Teaching Assistant – University of Toronto* (2011 – 2014)
CSC2221: (graduate) theory of distributed computing (2013).
CSC263: data structures and analysis (2011, 2012, 2013, 2014).
CSC369: principles of operating systems (2012).
CSC265: enriched data structures and analysis (2011).
- *Programming contest coach – York University* (2009)

Supervision

- Abhirup Das: intern (UW, F2020 – present)
- Daewoo Kim: MMath (UW, F2020 – present)
- Anubhav Srivastava: MMath (UW, S2020 – present)
- Gaetano Coccimiglio: MMath (UW, F2019 – present)
- Ajay Singh: PhD (UW, F2019 – present)
- Rosina Kharal: PhD (UW, F2019 – present)
- William Sigouin: MMath (UW, W2018 – W2020). Now working at Huawei.
- Gautam Pathak: intern (UW, W2020 – S2020). Now working at Cisco.
- Daewoo Kim: intern (UW, W2019 – S2019)
- Jialin Song: intern (University of Toronto, 2014)
- Kenneth Hoover: intern (University of Toronto, 2013)

Software Artifacts

- **Concurrent lock-free interpolation search trees in C++**
Bleeding edge data structure suitable for use as an index in an in-memory database. Experimental software artifact to accompany a PPOPP'20 conference paper (facilitating replication of experimental results). 2019.
<http://ist.tbrown.pro>
- **Setbench: data structure test harness & benchmark [WIP]**
Introduced a rigorous performance test bed with micro and macro benchmarks. Provides tools for avoiding common mistakes made in previous test beds. Fixes errors in popular concurrent sets and implements correct memory reclamation. Includes tools for validating and explaining experimental results. 2018-present.
<http://setbench.tbrown.pro>
- **LLX/SCX primitives for C++**
This is the state of the art implementation of the LLX and SCX synchronization primitives, with many improvements over the original. 2018.
<http://scx.tbrown.pro>
- **Reusable descriptors for lock-free data structures in C++**
Provided a lock-free reusable descriptor library. Used this library to accelerate four advanced lock-free data structures. 2018.
<http://weak-descriptors.tbrown.pro>
- **Support for range query operations in C++**
Produced three novel algorithms for adding range query operations to data structures. Augmented seven different data structures with range query support (producing up to five variants of each data structure). Used these data structures to accelerate an in-memory database (DBx1000). 2018.
<http://range-queries.tbrown.pro>
- **C++ hardware transactional lock-free data structure library**
Implemented unbalanced BSTs and relaxed (a,b)-trees using LLX and SCX. Produced four different transactional memory based algorithms for each data structure. 2017.
<http://3path-htm.tbrown.pro>
- **Lock-free memory reclamation in C++**
Provided a record manager library with allocation, reclamation and object pooling plugins (including five allocators and four memory reclamation algorithms). Implemented lock-free BSTs and Chromatic trees using this library. 2015.
<http://debra.tbrown.pro>
- **Java lock-free data structure library**
Produced the first lock-free unbalanced binary search tree implementation. Produced the first implementation of LLX and SCX synchronization primitives. Also introduced: k-ary search trees, relaxed AVL trees, Chromatic trees, b-slack trees. Includes experimental test harness with support for 13 competing data structures. 2014.
<http://java.tbrown.pro>

Funding

NSERC Collaborative Research and Development (\$209,700) Trevor Brown (75%) and Ali Mashtizadeh (25%)	2020 – 2023
Huawei Waterloo Joint Innovation Grant (\$190,000) Trevor Brown (75%) and Ali Mashtizadeh (25%)	2019 – 2021
Ontario Research Fund: Research Infrastructure (\$101,000) Trevor Brown (25%) and Ali Mashtizadeh (75%)	2020 – 2021
CFI John Evans Leaders Fund Grant (\$101,000) Trevor Brown (25%) and Ali Mashtizadeh (75%)	2020 – 2021
NSERC Discovery Grant (\$234,000)	2019 – 2024
NSERC Discovery Launch Supplement (\$12,500)	2019 – 2023
Waterloo Startup Grant (\$120,000)	2018 – 2022
NSERC Postdoctoral fellowship (\$90,000) Second highest ranked applicant in Canada	2017 – 2018
NSERC Alexander Graham Bell CGS D3 (\$105,000)	2013 – 2016
University of Toronto PhD fellowship (\$31,000)	2013 – 2016
Travel grants: SGS'12, SRDC'13, PODC'13, DISC'13 PODC'15, DISC'15, SPAA'16, TRANSACT'16 (\$8,000+)	2012 – 2016
University of Toronto MSc fellowship (\$19,000)	2011 – 2012
NSERC Alexander Graham Bell CGS M (\$17,500)	2011
NSERC Undergraduate Research Awards (\$22,000)	2009 – 2011
York University Entrance/Continuing Scholarships (\$3,720)	2006 – 2010
York Professor Ruth Hill Memorial Award (\$1,045) Awarded to the top undergraduate student in the Faculty of Science and Engineering	2008

Recognition

- Best paper at PPOP'20 (2020).
- *Nominated* for Governor General's Gold Medal (U of T, 2017)
(I did *not* win, but a school can only nominate one CS student annually)
- Award for Excellence in Teaching Assistance (U of T CS Student Union, 2014).
- Faculty of Science and Engineering Silver Medal (York University, 2011).
- Honourable Mention for the Computing Research Association's Outstanding Undergraduate Researcher Award (2011).
- Represented York at the ACM International Collegiate Programming Contests, ECNA region (2007 – 2011). In 2009, led senior school team to finish 6th (of 115).
- Dean's Honour Roll (York University) – Science and Engineering (2007 – 2011).
- Chair's Honour Roll (York University) – Mathematics and Statistics (2010).

Service

- **Program committee member** for
 - [ICDCS'18](#)
 - [PPoPP'19](#) (ERC)
 - [PODC'19](#) (23 papers)
 - PODC'20 (19 papers)
 - PODC'21
 - PPoPP'21
- **Publication chair** for [PPoPP'19](#). A [rough guide that I wrote for subsequent pub chairs](#).
- **Artifact evaluation committee member** for [PPoPP'16](#).
- **Conference paper reviews** for PODC'11, DISC'12, PODC'13, DISC'14, PODC'14, PPoPP'16, SIROCCO'16, SPAA'16, DISC'16, SPAA'17, RANDOM'17, DISC'17, PODC'18 (4x), DISC'18 (3x), SPAA'19, ESA'19 (2x), DISC'19 (2x), DISC'20.
- **Journal reviews** for the ACM Journal of Distributed Computing (DC/DIST) [4x], the IEEE Journal of Transactions on Parallel and Distributed Systems (TPDS) [3x], and the Elsevier Journal of Logical and Algebraic Methods in Programming (JLAMP).
- **Committees**
 - Graduate recruiting committee (University of Waterloo, 2018)
 - School advisory committee on faculty appointments (University of Waterloo, 2019)
 - Graduate recruiting committee (University of Waterloo, 2020)
- **Thesis committee member** for
 - Bryce Sandlund (PhD, University of Waterloo, 2020)
 - Thierry Delisle (PhD, University of Waterloo, 2020)
 - Diego Cepada (MMath, University of Waterloo, 2020)
- **Discussion panel member** for
 - Graduate skills seminars (University of Waterloo, Sep 2020)
 - UROC undergraduate research conference (University of Waterloo, Sep 2019)
- **Search committee for Chair of Computer Science**, University of Toronto (2015)
- **Wrote tenure recommendation letters**, York University (2010)
- **Developed student study resources** (2009 – 2010)
Developed online student resources for courses in Operations Research and Japanese, and for the ACM ICPC programming contests. They have been accessed 250,000 times.
- **Volunteer note-taker** (2008 – 2009)
Tutored and took notes on behalf of students with accessibility needs.