

Resume: Jeffrey Howard Kingston

Contact details

*<http://sydney.edu.au/engineering/it/~jeff/>
jeff@it.usyd.edu.au*

Date and Place of Birth

30 December 1958; Sydney, NSW, Australia.

Citizenship

Australian.

Academic Qualifications

1. B. Sc. (Hons. 1st class), University of Sydney, 1981; majoring in computer science and mathematics.
2. Ph. D., University of Sydney (Basser Department of Computer Science), 1986.

Appointments

1. Visiting Assistant Professor, Department of Computer Science, The University of Iowa (1984–7).
2. Lecturer, Basser Department of Computer Science, The University of Sydney (1988–91).
3. Senior Lecturer, Basser Department of Computer Science, The University of Sydney (1992–2000).
4. Associate Professor, Basser Department of Computer Science, The University of Sydney (2001–2002).
5. Honorary Associate, School of Information Technologies, The University of Sydney, August 2002–.

Research Interests

Algorithms and complexity, timetable construction, document formatting.

Awards

1. University of Sydney Excellence in Teaching Award, 1999. With Kathryn Crawford, Tony Greening, Alan Fekete, and Judy Kay.

Grants

1. University of Sydney / Australian Research Council Small Grants Scheme, 1994–96: *Consistency and extensibility in interactive document editing* (\$21,000)
2. University of Sydney Information Technology Committee Development Grant, 1999: *Facilitating group work in first year Computer Science* (\$12,000)

Books

1. Jeffrey H. Kingston. *Algorithms and Data Structures: Design, Correctness, Analysis*. Addison-Wesley, 1990. Reprinted 1991, 1995.
2. Jeffrey H. Kingston. *Algorithms and Data Structures: Design, Correctness, Analysis*. Addison-Wesley. Second Edition, 1997.
3. Jeffrey H. Kingston. *A User's Guide to the Lout Document Formatting System (Version 3)*. Basser Department of Computer Science, The University of Sydney, 1994–2000. URL <ftp://ftp.cs.su.oz.au/jeff/lout/lout-3.24.user.ps.gz>.
4. Jeffrey H. Kingston (ed. and principal author). *An Introduction to Computer Programming with Blue*. Basser Department of Computer Science, The University of Sydney, 1998.
5. Jeffrey H. Kingston (ed. and principal author). *An Introduction to Computer Programming with Java*. School of Information Technologies, The University of Sydney, 2002.

Book Chapters

1. Edmund Burke, Jeffrey H. Kingston, and Dominique de Werra. Applications to Timetabling. In Jonathon L. Gross and Jay Yellen (eds.), *Handbook of Graph Theory*. CRC Press, 2004. (Also Second Edition, 2013, ISBN 9781439880180)
2. Jeffrey H. Kingston. Educational Timetabling. In A. Sima Uyar, Ender Özcan, and Neil Urquhart (eds.), *Automated Scheduling and Planning: From Theory to Practice*, pages 91–108. Springer, 2013. ISBN 978-3-642-39303-7

Computer Software

1. Jeffrey H. Kingston. The Lout Document Formatting System (Version 3). Computer program, 1998. URL <ftp://ftp.cs.su.oz.au/jeff/lout/lout-3.24.tar.gz>.
2. Jeffrey H. Kingston. Kedsal: Kingston's Eiffel Data Structures and Algorithms Library, 1996. URL <ftp://ftp.cs.usyd.edu.au/jeff/kedsal/kedsal.1.0.tar.gz>.
3. Jeffrey H. Kingston. The KTS high school timetabling web site (Version 1.6), October 2007. URL <http://www.it.usyd.edu.au/~jeff>.
4. Jeffrey H. Kingston. *The KHE High School Timetabling Engine*. School of Information Technologies, The University of Sydney, 2010. URL <http://www.it.usyd.edu.au/~jeff/khe>.

5. Jeffrey H. Kingston. The HSEval High School Timetable Evaluator, 2010. URL <http://www.it.usyd.edu.au/~jeff/hseval.cgi>.

Refereed Conference and Journal Publications

1. Jeffrey H. Kingston. Analysis of tree algorithms for the simulation event list. *Acta Informatica* **22**, 15–33 (1985).
2. Jeffrey H. Kingston. Analysis of Henriksen’s algorithm for the simulation event set. *SIAM Journal on Computing* **15**, 887–902 (1986).
3. Jeffrey H. Kingston. The amortized complexity of Henriksen’s algorithm. *BIT* **26**, 156–163 (1986).
4. Jeffrey H. Kingston. A new proof of the Garsia-Wachs algorithm. *Journal of Algorithms* **9**, 129–136 (1988).
5. Jeffrey H. Kingston. The design and implementation of the Lout document formatting language. *Software—Practice and Experience* **23**, 1001–1041 (1993).
6. Tim B. Cooper and Jeffrey H. Kingston. The solution of real instances of the timetabling problem. *The Computer Journal* **36**, 645–653 (1993).
7. Diab Abuaiadh and Jeffrey H. Kingston. Are Fibonacci heaps optimal?. In *ISAAC’94: Fifth Annual International Symposium on Algorithms and Computation, Beijing, August 25–27, 1994*, pages 442–450. Springer-Verlag (Lecture Notes in Computer Science 834), 1994.
8. Tim B. Cooper and Jeffrey H. Kingston. A program for constructing high school timetables. In *Proceedings 1st International Conference on the Practice and Theory of Automated Timetabling*, Napier University, Edinburgh, UK, 1995.
9. Tony Greening, Judy Kay, Jeffrey H. Kingston, and Kathryn Crawford. Trialling a problem-based learning approach to first year Computer Science. In *Proceedings of the First Australasian Conference on Computer Science Education*. The University of Sydney, Australia, July 1996.
10. Tim B. Cooper and Jeffrey H. Kingston. The complexity of timetable construction problems. In *Practice and Theory of Automated Timetabling (First International Conference, PATAT’95, Edinburgh, August 1995)*, pages 283–295. Springer Lecture Notes in Computer Science 1153, 1996.
11. Edmund K. Burke, Jeffrey H. Kingston, and Paul W. Pepper. A standard data format for timetabling instances. In *Practice and Theory of Automated Timetabling II*. Springer Lecture Notes in Computer Science 1408, 1997.
12. Tony Greening, Judy Kay, Jeffrey H. Kingston, and Kathryn Crawford. Results of a PBL trial in first year Computer Science. In John Hurst (ed.), *Proceedings of the Second Australasian Conference on Computer Science Education*, pages 201–206. ACM Press, Melbourne, July 1997.

13. Alan Fekete, Tony Greening, and Jeffrey H. Kingston. Conveying technical content in a curriculum using problem based learning. In *Proceedings of the Third Australasian Conference on Computer Science Education*, pages 198–202, Brisbane, July 1998.
14. Edmund Burke, Kirk Jackson, Jeffrey H. Kingston, and Rupert Weare. Automated timetabling: the state of the art. *The Computer Journal*, 565–571 (1997).
15. Michael Barg, Kathryn Crawford, Alan Fekete, Tony Greening, Owen Hollands, Judy Kay, and Jeffrey H. Kingston. Problem-based learning for foundation Computer Science courses. *Computer Science Education* **10**, 109–128 (2000).
16. Jeffrey H. Kingston and Nicholas P. Sheppard. On reductions for the Steiner problem in graphs. In *Proceedings of the 10th Australasian Workshop on Combinatorial Algorithms AWOCA*, pages 34–43, 1999. To appear in *Journal of Discrete Algorithms*, 2(1).
17. Jeffrey H. Kingston and Nicholas P. Sheppard. Obtaining maximum reduction for the Steiner problem in graphs. *Journal of Discrete Algorithms* (2000).
18. Jeffrey H. Kingston and Nicholas P. Sheppard. On the hardness of computing maximum self-reduction sequences. *Discrete Mathematics* **226/1–3**, 243–253 (2000).
19. Alan Fekete, Judy Kay, Jeffrey H. Kingston, and Kapila Wimalaratne. Supporting reflection in introductory Computer Science. In *Proceedings of the ACM Technical Symposium in Computer Science Education*, Austin, Texas, March 2000.
20. Jeffrey H. Kingston and Benjamin Yin-Sun Lynn. A software architecture for timetable construction. In *Practice and Theory of Automated Timetabling III*, pages 342–350. Springer Lecture Notes in Computer Science 2079, 2001.
21. Jeffrey H. Kingston. Modelling timetabling problems with STTL. In *Practice and Theory of Automated Timetabling III*, pages 309–321. Springer Lecture Notes in Computer Science 2079, 2001.
22. David Corne and Jeffrey H. Kingston. Addressing the availability-based laboratory/tutorial timetabling problem with heuristics and metaheuristics (abstract). In *Proceedings of the 3rd International Conference on the Practice and Theory of Automated Timetabling*, Konstanz, August 2000.
23. David J. Abraham and Jeffrey H. Kingston. Generalizing edge colouring to solve real instances of the timetabling problem. In *Practice and Theory of Automated Timetabling IV (4th International Conference, PATAT 2002, Gent, Belgium)*, pages 288–298. Springer Lecture Notes in Computer Science 2740, 2002.
24. Jeffrey H. Kingston. A tiling algorithm for high school timetabling. In *Practice and Theory of Automated Timetabling V (Springer Lecture Notes in Computer Science 3616)*, pages 208–225, 2005.

25. Jeffrey H. Kingston. The KTS high school timetabling system. In *Practice and Theory of Automated Timetabling VI (Sixth International Conference, PATAT2006, Czech Republic, August 2006, Selected Papers)*, pages 308–323. Springer Lecture Notes in Computer Science 3867, 2007.
26. Jeffrey H. Kingston. Hierarchical timetable construction. In *Practice and Theory of Automated Timetabling VI (Sixth International Conference, PATAT2006, Czech Republic, August 2006, Selected Papers)*, pages 294–307. Springer Lecture Notes in Computer Science 3867, 2007.
27. Jeffrey H. Kingston. Resource assignment in high school timetabling. In *PATAT2008 (Seventh international conference on the Practice and Theory of Automated Timetabling, Montreal, August 2008)*, 2008.
28. Samad Ahmadi, Sophia Daskalaki, Jeffrey H. Kingston, Jari Kyngäs, Cimmo Nurmi, Gerhard Post, David Ranson, and Henri Ruizenaar. An XML format for benchmarks in high school timetabling. In *PATAT08 (Seventh international conference on the Practice and Theory of Automated Timetabling, Montreal, August 2008)*, 2008.
29. Gerhard Post, Jeffrey H. Kingston, Samad Ahmadi, Sophia Daskalaki, Christos Gogos, Jari Kyngäs, Cimmo Nurmi, Nysret Musliu, Nelishia Pillay, Haroldo Santos, and Andrea Schaerf. XHSTT: an XML archive for high school timetabling problems in different countries. *Annals of Operations Research* (November 2011).
30. Jeffrey H. Kingston. Solving the general high school timetabling problem. In *PATAT2010 (Eighth international conference on the Practice and Theory of Automated Timetabling, Belfast, August 2010)*, 2010.
31. Jeffrey H. Kingston. Timetable construction: the algorithms and complexity perspective (invited plenary presentation). In *PATAT2010 (Eighth international conference on the Practice and Theory of Automated Timetabling, Belfast, August 2010)*, 2010.
32. Jeffrey H. Kingston. Timetable construction: the algorithms and complexity perspective. *Annals of Operations Research* **218**, 249–259 (2014). DOI 10.1007/s10479-012-1160-z
33. Jeffrey H. Kingston. Repairing high school timetables with polymorphic ejection chains. In Dag Kjenstad, Atle Riise, Tomas Eric Nordlander, Barry McCollum, and Edmund Burke (eds.), *PATAT 2012 (Ninth international conference on the Practice and Theory of Automated Timetabling, Son, Norway, August 2012)*, pages 16–30, 2012.
34. Jeffrey H. Kingston. Resource assignment in high school timetabling. *Annals of Operations Research* **194**, 241–254 (2012).
35. Gerhard Post, Luca Di Gaspero, Jeffrey H. Kingston, Barry McCollum, and Andrea Schaerf. The Third International Timetabling Competition. In Dag Kjenstad, Atle Riise, Tomas Eric Nordlander, Barry McCollum, and Edmund Burke (ed.s), *PATAT 2012 (Ninth international conference on the Practice and Theory of Automated Timetabling, Son, Norway, August 2012)*, pages 479–484, 2012.

36. Gerhard Post, Luca Di Gaspero, Jeffrey H. Kingston, Barry McCollum, and Andrea Schaerf. The Third International Timetabling Competition. *Annals of Operations Research*. DOI 10.1007/s10479-013-1340-5
37. Jeffrey H. Kingston. KHE14: An algorithm for high school timetabling. In Ender Özcan, Edmund Burke, and Barry McCollum (eds.), *PATAT 2014 (Tenth international conference on the Practice and Theory of Automated Timetabling, York, UK, August 2014)*, pages 269–291, 2014.
38. Jeffrey H. Kingston. Integrated student sectioning. In Ender Özcan, Edmund Burke, and Barry McCollum (eds.), *PATAT 2014 (Tenth international conference on the Practice and Theory of Automated Timetabling, York, UK, August 2014)*, pages 489–492, 2014.

Other Publications

1. Jeffrey H. Kingston. *Analysis of Algorithms for the Simulation Event List*. Ph.D. thesis, The University of Sydney, 1984.
2. Alan Fekete, Judy Kay, and Jeffrey H. Kingston. Teaching exceptionally able students in a mass CS1 course. Tech. Rep. (1992), Basser Department of Computer Science, The University of Sydney.
3. Jeffrey H. Kingston. A refined analysis of Christofides' travelling salesman heuristic. Tech. Rep. (1992), Basser Department of Computer Science, The University of Sydney.
4. Jeffrey H. Kingston. *An Expert's Guide to the Lout Document Formatting System (Version 3)*. Basser Department of Computer Science, The University of Sydney, 1994.
5. Jeffrey H. Kingston. A practical introduction to the Lout document formatting system. Overhead transparencies (1994), Basser Department of Computer Science, The University of Sydney.
6. Diab Abuaiadh and Jeffrey H. Kingston. The shortest path problem with negative edges. Tech. Rep. (1995), Basser Department of Computer Science, The University of Sydney.
7. Jeffrey H. Kingston (ed.). Bibliography on the practice and theory of automated timetabling. Tech. Rep. (1995), Basser Department of Computer Science, The University of Sydney. URL <ftp://ftp.cs.su.oz.au/jeff/timetabling/timetabling.bib.gz>.

Seminars and workshops

1. Jeffrey H. Kingston. *A fresh start for first year Computer Science*. Computer Science Department, Royal Melbourne Institute of Technology, 12 June 1997.
2. Jeffrey H. Kingston. *A fresh start for first year Computer Science*. ITITAB Workshop, Bardon Conference Centre, Bardon, Queensland, 23 October 1997. The ITITAB Workshop was attended by a group of senior people from around Australia in the Information Technology area of TAFE (Technical and Further Education).

3. Alan Fekete, Judy Kay, and Jeffrey H. Kingston. *Problem-based learning: some issues*. The University of Sydney Vice-Chancellor's Showcase of University Teaching, Education Building, University of Sydney, 11 November 1998.
4. Alan Fekete, Judy Kay, and Jeffrey H. Kingston. *Clarifying the goals: innovative assessment in Computer Science*. The University of Sydney Vice-Chancellor's Showcase of University Teaching, Education Building, University of Sydney, 11 November 1998.
5. Jeffrey H. Kingston. *The Lout document formatting system*. Canon Information Systems Research Australia, North Ryde, NSW, Australia, 18 November 1998.
6. Jeffrey H. Kingston. *Assessment of problem-based learning in introductory Computer Science*. Academic Forum of the University of Sydney, Sydney, NSW, Australia, 28 May 2001.
7. Jeffrey H. Kingston. *The KTS high school timetabling system*. One-day workshop for timetable coordinators from New South Wales high schools, 4 August 2005, at the University of Sydney.

Conference Series Steering Committees

1. Founding member of conference series steering committee, Practice and Theory of Automated Timetabling, 1995–.

Conference Programme Committee Memberships

1. 1st International Conference on the Practice and Theory of Automated Timetabling, Napier University, Edinburgh, UK, August 1995.
2. 2nd International Conference on the Practice and Theory of Automated Timetabling (PATAT97), University of Toronto, Canada, August 1997.
3. 3rd International Conference on the Practice and Theory of Automated Timetabling (PATAT2000), Constanz, 2000.
4. 4th International Conference on the Practice and Theory of Automated Timetabling (PATAT2002), Gent, Belgium, August 2002.
5. Multidisciplinary International Conference on Scheduling: Theory and Applications (MISTA03), Nottingham, UK, 17–21 August 2003.
6. 5th International Conference on the Practice and Theory of Automated Timetabling (PATAT2004), Pittsburgh, PA, August 2004.
7. Multidisciplinary International Conference on Scheduling: Theory and Applications (MISTA05), 2005.
8. 6th International Conference on the Practice and Theory of Automated Timetabling (PATAT2006), Brno, Czech Republic, August 2006.

9. Multidisciplinary International Conference on Scheduling: Theory and Applications (MISTA07), 2007.
10. 7th International Conference on the Practice and Theory of Automated Timetabling (PATAT2008), Montreal, Canada, August 2008.
11. 8th International Conference on the Practice and Theory of Automated Timetabling (PATAT2010), Belfast, UK, August 2010.
12. 9th International Conference on the Practice and Theory of Automated Timetabling (PATAT2012), Son, Norway, August 2012.
13. 10th International Conference on the Practice and Theory of Automated Timetabling (PATAT2014), York, UK, August 2014.

Programming Competition Directorships

1. Joint Director, Third International Timetabling Competition (ITC2011).

Journal Editorial Boards

1. Member of editorial board, Journal of Scheduling, Wiley, 1998–2001.
2. Member of guest edition editorial board, European Journal of Operational Research.

Journal Associate Editorship

1. Journal of Scheduling, Wiley, 2001–.

Ph.D. Candidates Supervised

1. Stephen C. Hirst (1995). Vector trees: a compact data structure for sets.
2. Diab Abuaiadh (1996). On the complexity of the shortest path problem.
3. Nicholas Sheppard (2000). Self-reduction for combinatorial optimisation.

Honours and M.Sc. (Prelim) Candidates Supervised

1. Saritha Roach (1988). Backtracking document layout in Prolog.
2. Tom Allchin (1988). The further development of Lout.
3. Jim Donovan (1989). A software package for timetabling.
4. Philip Rutherford (1989). Efficient backtracking algorithms for document layout.

5. Andrew Heading (1990). Timetabling a high school.
6. Tim B. Cooper (1992). High school timetabling.
7. Thomas Lapins-Silvirs (1992). High school timetabling.
8. Chung-Li Chen (1993). The addition of horizontal galleys to Lout.
9. Yvette F. Clancey (1993). A graphical user interface for timetabling.
10. David M. Roberts (1993). University timetabling.
11. Richard T. Plunkett (1994). University timetabling.
12. Gabor Inokai (1995). Optimal document layout in Lout.
13. Vincent Tan (1999). Implementation of the Nonpareil language.
14. Soon Leong (2001) (MIT project). Genetic algorithms in a general framework for timetabling.
15. Mark Wotton (2001). Implementation of the Nonpareil language.
16. David Abraham (2001–2). The high school timetable construction problem.
17. Bradley Baetz (2002). Nonpareil: a strongly typed object oriented functional programming language.

Special Studies Program and Summer Scholarship Students Supervised

1. Adam Wootton (February 1997). Bipartite matching.
2. Michael Lim (February 1997). Optimal document layout.
3. Mike Barg (February 1997). An extensible GUI for document editing.
4. Ben Lin (1997/1998). A specification language for timetabling problems.
5. Vincent Tan (Summer 1997/1998). A PDF back end for Lout.
6. Ben Lin (February 1998). An object-oriented framework for timetabling.
7. Ben Lin (July 1998). Comparison of operations research models with local search on a timetabling problem.
8. Vincent Tan (1998/1999). Implementation of a functional object-oriented language.
9. Jessica Preston (1998/1999). Implementation of a two-dimensional graphics package for document formatting.

10. Mark Gebski (February 2001). Implementation of a web site for accepting applications from potential tutors.
11. Mark Gebski (July 2001). A network flow application to timetable construction.

Teaching Experience

At Iowa, I taught courses on data structures, programming language concepts (including syntax, axiomatic semantics, Ada, Lisp and Prolog), discrete structures, and a graduate level course in analysis of algorithms. At Sydney:

- 1988* CS2 Theoretical Computer Science (18 lectures)
CS3 Complexity Theory (18 lectures, 17 tutorials)
CS4 Advanced Analysis of Algorithms (18 lectures)
- 1989* CS1 Overview of Computer Science (13 tutorials)
CS2 Languages and Automata (13 lectures)
CS2 Large Programming Project 2 (two weeks' preparation and marking)
CS3 Complexity Theory (13 lectures, 12 tutorials)
CS4 Advanced Analysis of Algorithms (12 lectures)
- 1990* CS2 Design and Data Structures (28 lectures)
CS2 Large Programming Project 1 (two weeks' preparation and marking)
CS2 Languages and Automata (13 lectures)
CS2 Large Programming Project 2 (two weeks' preparation and marking)
- 1991* CS1 Challenges (13 lectures)
CS2 Design and Data Structures (28 lectures, 13 tutorials)
CS2 Large Programming Project 1 (two weeks' preparation and marking)
CS4 Advanced Analysis of Algorithms (12 lectures)
- 1992* CS1 Challenges (13 seminars)
CS2 Design and Data Structures (28 lectures, 13 tutorials)
CS2 Large Programming Project 1 (two weeks' preparation and marking)
CS4 Advanced Analysis of Algorithms (12 lectures)
- 1993* CS1 Challenges (11 seminars)
CS2 Design and Data Structures (28 lectures, 13 tutorials)
CS2 Large Programming Project 1 (two weeks' preparation and marking)
- 1994* CS1 Challenges (11 seminars)
CS1 Programming 1 (first 13 lectures)
CS2 Large Programming Project 1 (two weeks' preparation and marking)
- 1995* CS1 Challenges (11 seminars)
CS1 Programming 1 (first 13 lectures)
- 1996* CS1 Programming 1 (first 13 lectures)
On sabbatical leave rest of year

- 1997 COMP 101 Introductory Programming (Section leader, 14 seminars, tutorials and labs)
COMP 102 Introductory Computer Science (Section leader, as above)
COMP 308 Object-Oriented Systems, half of (13 lectures, 7 tutorials)
- 1998 COMP 1901 Introductory Programming Advanced (Section leader, 14 seminars, tutorials and labs)
COMP 1001 Introductory Programming (Section leader, as above)
COMP 3008 Object-Oriented Systems, half of (13 lectures, 7 tutorials)
COMP 3101 Theory of Computation (27 lectures, 14 tutorials)
- 1999 COMP 1901 Introductory Programming Advanced (Section leader, 14 seminars, tutorials and labs; also SSP project supervision of 15 students)
COMP 1001 Introductory Programming (Section leader, as above)
- 2000 On sabbatical leave first semester
COMP 1002 Introductory Computer Science (Section leader, 26 seminars, 12 tutorials and labs)
Computer Science Honours course: Intractable Problems in Theory and Practice (5 students, 20 hours lectures)
- 2001 INFO 2000 System Analysis and Design (coordinator, no student contact)
COMP 1002 Introductory Computer Science (coordinator, no student contact)
Computer Science Honours course: Intractable Problems in Theory and Practice (7 students, 20 hours lectures)
- 2002 SOFT 1001 Software Development 1 (coordinator, no student contact)
SOFT 1901 Software Development 1 Advanced (Section leader, 13 seminars)
Computer Science Honours course: Intractable Problems in Theory and Practice (6 students, 20 hours lectures)
- 2003 Computer Science Honours course: Intractable Problems in Theory and Practice (6 students, 20 hours lectures)
- 2004 Computer Science Honours course: Intractable Problems in Theory and Practice (2 students, 20 hours lectures)
- 2006 Computer Science Honours course: Intractable Problems in Theory and Practice (7 students, 20 hours lectures)

Professional Development

Attendance at seminars: 'Supervising Postgraduate Students,' 'Effective Tutoring,' 'Theory and Practice of Teaching and Learning,' given by the Centre for Teaching and Learning at the University of Sydney, 1988-9.

Attendance at all-day seminar: 'Adopting Problem-Based Approaches to Learning,' given at the Centre for Teaching and Learning at the University of Sydney on 17th July 1995 by Penny Little from PROBLARC.

Attendance at conference: 'The University of Sydney Vice-Chancellor's Showcase of University Teaching,' 11–12 November 1998.

Attendance at workshop: 'Performance Management and Development Workshop for Supervisors and Reviewers,' University of Sydney Staff Support and Development Unit, 13 November 2001.

Departmental Administrative Appointments

1. Director, Diploma in Computer Science (1988–93)
2. Chairman, Courses Committee (1989–91)
3. Director, Computer Science 3 (1992–95)
4. Undergraduate Director (1997–1999)
5. Associate Undergraduate Director (July–August 2000)
6. Undergraduate Director (September–December 2000)
7. Undergraduate Director and Associate Head of Department (2001–July 2002)

Research Projects since 2002

Since retiring from paid employment in 2002 I have been carrying out research at home, dividing my time between two projects: the Nonpareil document formatting system, not yet published, and automated timetable construction, published as listed above.