

Michael J. Bannister

Employment

Assistant Professor	Santa Clara University	9/2016–Present
Taught courses in programming and the theory of algorithms. Conducted research in graph drawing and network visualization. Received consistently high teaching evaluations. Served on departmental and university committees. Worked with C++14, Python3, SDL and SFML.		
Visiting Assistant Professor	Pomona College	7/2015–6/2016
Taught courses in data structures, computational photography and discrete mathematics. Received consistently high teaching evaluations. Supervised projects in cryptography, game tree search and visualization. Worked with Java, C, MATLAB, make, ASan, UBSan, Valgrind and OS X.		
Graduate Student	University of California, Irvine	9/2010–6/2015
Served as a Graduate Student Researcher (GSR) and as a Teaching Assistant (TA). Won the best presentation award at the 21st International Symposium on Graph Drawing out of the 40+ presentations given.		
Graduate Student Instructor	University of California, Irvine	6/2014–9/2014
Taught courses in programming and data structures. Worked with C++11, make, SDL, Valgrind and Linux.		
Adjunct Faculty	Santiago Canyon College	6/2008–5/2010
Developed and taught courses in algebra and trigonometry.		
Adjunct Faculty	Orange Coast College	1/2008–5/2010
Developed and taught courses in algebra, calculus, differential geometry, topology and the theory of computation.		

Education

PhD in Computer Science	University of California, Irvine	2015
Thesis: Lower Bounds and Fixed-Parameter Tractability of Drawing Graphs. Advisor: David Eppstein.		
MA in Mathematics	University of California, Los Angeles	2007
Emphasis: abstract algebra and mathematical logic. Jointly earned with BS.		
BS in Mathematics	University of California, Los Angeles	2007
Summa cum laude. Departmental highest honors. Sherwood Prize in Mathematics. Phi Beta Kappa.		

Selected Projects

Game tree search for Connect-K

An AI player for tic-tac-toe, Connect-4 and generalizations of these games. Uses several tree search methods and board evaluation heuristics. Implemented using C++ and SDL.

Darwin tournament simulator

A high-performance simulator of Darwin's World, a programming assignment created by Nick Parlante. In this assignment, students program creatures intent on taking over the World. The simulator is used to run a round-robin tournament of all creatures submitted by the students. Implemented using C++, SDL and gperftools.

Graph drawing testbed

A tool for entering small input graphs and experimenting with various graph visualization algorithms. Used to design new algorithms for the visualization of social networks. Implemented using Python, JSON, NetworkX and wxPython.

Selected Research

Randomized speed up of the Bellman-Ford algorithm

A variant of the Bellman-Ford algorithm which is faster than the previously-best variant (Yen, 1970) by a factor of $2/3$ in expectation with high probability. Currently the fastest known variant of the Bellman-Ford algorithm.

Windows into relational events

Data structures for the analysis of timestamped events in a social network. The data structures support several common counting queries on subgraphs formed by the relational events in a given time interval.

Confluent orthogonal drawings of syntax diagrams

An algorithm and implementation for producing a syntax diagram from a context free grammar. The syntax diagram is optimized for several legibility criteria.