

Keith Brian Gallagher, Ph.D.

6587 Canal Road Melbourne, Florida 32904
kbgphd@gmail.com +1 321-505-9252
US Citizen Top Secret Clearance (inactive)
February 14, 2017

Current Position

- 2010: Associate Professor of Software Engineering.
School of Computing.
Florida Institute of Technology.
150 West University Blvd. Melbourne, FL 32901
kgallagher@fit.edu
<http://cs.fit.edu/~kgallagher>
+1 321-674-8395

Past Appointments

- Director of Software Engineering, Dept. of Computer Sciences and Cyber Security, Florida Institute of Technology, 2011 - 2016.
- 2005 to 2010: Senior Lecturer, Department of Computer Science, Durham University.
- 1985 to 2005: Associate Professor, Computer Science Department, Loyola University Maryland.
- 1982 to 1985: Assistant Professor of Computer Science, Houghton College, New York.

Education

- Ph.D. Computer Science. The University of Maryland Graduate School at Baltimore.
- Master of Science. Computer and Communication Sciences, The University of Michigan.
- Master of Science. Mathematics. The University of Michigan.
- Bachelor of Arts. Mathematics. Bucknell University, Lewisburg, Pennsylvania.

Research Appointments

- 2004: Visiting Research Fellow in Computer Science, University of Durham, England.
- 1997 to 1998: Visiting Senior Research Engineer at the Commonwealth Scientific and Industrial Research Organization (CSIRO). Canberra, Australia.
- 1992 to 2000: Faculty Research Associate, Information Technology Laboratory, National Institute of Standards and Technology.

Research Interests

- *Areas:* Program Slicing; Software Maintenance and Evolution; Empirical Studies; Program Comprehension; Software Visualization; Software Testing.
- *Current Focus:* High Volume Automated Testing.
- *Guiding Principle:* What can I do to help software engineers “in the trenches?”

Works in Progress

- Evaluating the Use of Sound in Program Comprehension. With L. Berman & S. Kozaitis.
- HiVAT: High Volume Automated Testing. With C. Kaner, F. Hull II, M. Fioravanti & C. Oliver.
- Flaky Tests are Just Failures.
- High volume automated testing with algebraic axiomatic specifications. With S. F. Hull & M. Fioravanti.
- An emerging architecture for flexible high-volume automated testing. With M. Fioravanti, C. Oliver & C. Kaner.
- Long sequence randomized unit testing with MaadiJ. With S. F. Hull & M. Fioravanti.
- Instructing introductory software engineering students with live open source projects. With M. Fioravanti & A. Marcoux.
- Restructuring secure code and preserving integrity.

Recent Courses

- Software Maintenance and Evolution
- Introduction to Software Engineering (undergraduate)
- Software Engineering I & II (graduate)
- Software Testing II (structure, tools and automation)
- Software Design
- Agile Methods
- Program Slicing
- Non-functional Requirements

Completed Supervised Graduate Students

- Bradley Rees, Ph.D. Florida Institute of Technology. *Ego-Based Overlapping Communities Detection: A New Paradigm*. 2015.
- Mohammad Abdullah, Ph.D. Durham University. *A Weighted Grid for Measuring Program Robustness*. 2010.
- Amir Ngah, Ph.D. Durham University. *Regression Test Selection by Exclusion*. 2010.
- Lewis Berman, Ph.D. Durham University. *Program Comprehension Through Sonification*. 2009.
- Robert Konczynski. M.Sc. Florida Institute of Technology. *The Importance of data validation in migrating to SOA*. 2015.
- Scott Forest Hull. M.Sc. Florida Institute of Technology. *Long sequence xUnit testing*. 2015.
- Anand Gopalakrishnan M.Sc. Florida Institute of Technology. *Conquest: An Interface for Test Automation Design*. 2012.

Professional Activities

- IEEE & ACM. Senior Membership applications submitted.
- Editorial Board of the *Journal of Software Maintenance and Evolution*.
- Steering Committee & Program Committee for the International Workshop on Source Code and Manipulation.
- Program Committee for the International Conference on Software Maintenance.

- Program Committee for the International Conference on Program Comprehension. Program Co-chair for 2010 Conference.
- Review for *IEEE Transactions on Software Engineering*, *Journal of Software Maintenance*, *IEEE Software*.

Funded Grant Proposals

- National Science Foundation (0629454) Learning Units on Law and Ethics in Software Engineering. Co-PI. \$268,000. 2012.
- Constructing the Surgeon's Assistant, *National Institute of Standards and Technology, Information Technology Laboratory*, 1994-1996. \$375,000.
- Evaluating Decomposition Slicing as a Software Maintenance Methodology, *Research Initiation Award of the National Science Foundation*, with Research Experience for Undergraduates supplement, 1991-1993. \$59,500

Invited Program Presentations

- “A Peek Under the Bonnet: One Programmer’s Psychology” Keynote Presentation, Psychology of Programming Interest Group (PPIG), Limerick, Ireland, June, 2009.
- “Techniques for Understanding and Mining Legacy Assets.” Invited program presentation, the International Workshop on Program Comprehension, Paris, France, 2002.
- “The Comprehender’s Workbench.” Invited program presentation, the International Workshop on Program Comprehension, Limerick, Ireland, 2000.
- “Software Surgery.” A Tutorial. Presented at:
 - 1999 International Conference on Software Maintenance, Keable College, Oxford, England.
 - 1998 International Conference on Software Maintenance, Washington, DC.
 - 1998 International Conference on Software Engineering, Kyoto, Japan.
- “More of What I Want From an Evolution System” at the *Durham Workshop on Program Transformation for Software Evolution*, Durham, England, 1998.
- “What I Want From an Evolution System” at the *ICSE-17 Workshop on Program Transformation for Software Evolution*, Seattle, WA, 1996.

Publications

- [1] Bradley S Rees and Keith B Gallagher. Detecting overlapping communities in complex networks using swarm intelligence for multi-threaded label propagation. In *Complex Networks*, pages 111–119. Springer Berlin Heidelberg, 2013.
- [2] Giuliano Antoniol and Keith B Gallagher. Preface to the special issue on program comprehension. *Empirical Software Engineering*, 2013.

- [3] Bradley S Rees and Keith B Gallagher. Egocustering: overlapping community detection via merged friendship-groups. In *The Influence of Technology on Social Network Analysis and Mining*, pages 1–20. Springer Vienna, 2013.
- [4] Bradley S Rees and Keith B Gallagher. Overlapping community detection using a community optimized graph swarm. *Social Network Analysis and Mining*, 2(4):405–417, 2012.
- [5] Keith Gallagher, C Kaner, and Jenifer Deignan. The law and reverse engineering. In *2012 19th Working Conference on Reverse Engineering (WCRE)*, . IEEE, 2012.
- [6] Mohammad Abdallah, Malcolm Munro, and Keith Gallagher. A static robustness grid using MISRA C2 language rules. In *ICSEA 2011, The Sixth International Conference on Software Engineering Advances*, pages 65–69, 2011.
- [7] Mohammad Abdallah, Malcolm Munro, and Keith Gallagher. Certifying software robustness using program slicing.. In *26th International Conference on Software Maintenance*, 2010.
- [8] Bradley S Rees and Keith B Gallagher. Overlapping community detection by collective friendship group inference. In *Advances in Social Networks Analysis and Mining (ASONAM), 2010 International Conference on*, pages 375–379, . IEEE, 2010.
- [9] Mark Harman, David Binkley, Keith Gallagher, Nicolas Gold, and Jens Krinke. Dependence clusters in source code. *ACM Transactions on Programming Languages and Systems (TOPLAS)*, 32(1), 2009.
- [10] Keith Gallgher and David Binkley. Program slicing. In Hausi Muller, editor, *2008 Frontiers of Software Maintenance*. IEEE Press, Oct. 2008. ISBN 978-14244-2655-3. Invited Paper.
- [11] K. Gallagher, A. Hatch, and M. Munro. Software architecture visualization: An evaluation framework and its application. *IEEE Transactions on Software Engineering*, 34(2):260 – 270, March/April 2007. ISSN 0098-5589. DOI 10.1109/TSE2007.70757.
- [12] K. Gallagher. Desert island reading. *Automated Software Engineering*, 14(4):465 – 470, December 2007. ISSN 0928-8910 (print) 1573-7335 (online). Invited paper.
- [13] K. Gallagher, T. Hall, and S. Black. Reducing regression test size by exclusion. In L. Tahvildari and G. Canfora, editors, *23rd International Conference on Software Maintenance*, pages 157 – 166, Paris, France, 2007. IEEE. ISBN 1-4244-1256-0.
- [14] K. Gallagher, D. Binkley, and M. Harman. Stop-list slicing. In *6th IEEE Workshop on Source Code and Analysis, SCAM-6*, pages 11 – 20, . IEEE Press, September 2006. ISBN 0-7695-2353-6.
- [15] L. Berman and K. Gallagher. Listening to program slices. In *International Conference on Auditory Display ICAD-06*, June 2006. On-line proceedings <http://www.icad.org>.
- [16] L. Berman and K. Gallagher. The sound of software: Using sonification to aid comprehension. In *International Conference on Program Comprehension*, 2006. Working session.

- [17] K. Gallagher, A. Hatch, and M. Munro. A framework for software architecture visualization assessment. In S. Ducasse, M. Lanza, A. Marcus, J. Maletic, and M-A. Storey, editors, *Third IEEE Workshop on Visualizing Software (VISSOFT 2005)*, pages 76 – 82, . IEEE Press, September 2005. ISBN 0-7803-9540-9.
- [18] K. Gallagher. MonkeySort. *The Journal of Computing Sciences in Colleges*, 15(3):70 – 81, February 2005.
- [19] K. Gallagher. Some notes on interprocedural program slicing. In *The Fourth Workshop on Source Code and Analysis, SCAM-4*, pages 36 –42, September 2004.
- [20] K. Gallagher, M. Harman, and S. Danicic. Guaranteed inconsistency avoidance during software evolution. *Journal of Software Maintenance and Evolution: Research and Practice*, 15:393–415, 2003.
- [21] K. Gallagher and L. Layman. Are decomposition slices clones?. In *The 11th International Workshop on Program Comprehension*, 2003.
- [22] K. Gallagher and D. Binkley. An empirical study of computation equivalence as determined by decomposition slice equivalence. In *The 10th Working Conference on Reverse Engineering, WCRE-03*, 2003.
- [23] K. Gallagher and L. O’Brien. Analyzing programs via decomposition slicing. In *International Workshop on Empirical Studies of Software Maintenance, WESS*, 2001.
- [24] K. Gallagher and N. Fulton. Using program slicing to estimate software robustness. In *International Systems Software Assurance Conference, ISSAC*, 1999.
- [25] S. Woods, L. O’Brien, T. Lin, K. Gallagher, and A. Quilici. An architecture for interoperable program understanding tools. In *The 6th International Workshop on Program Comprehension*, 1998.
- [26] M. Hutchens and K. Gallagher. Improving visual impact analysis. In *The 1998 International Conference on Software Maintenance-98 ICSM-98*, 1998.
- [27] M. Harman and K. Gallagher. Program slicing. *Journal of Information and Software Technology*, 40(11&12), 1998.
- [28] M. Harman and K. Gallagher, editors. *Journal of Information and Software Technology*, volume 40. Wiley, 1998. Special issue on program slicing.
- [29] K. B. Gallagher and L. O’Brien. Reducing visualization complexity using decomposition slices. In *The 1997 Software Visualization Workshop, SoftVis97*, Dec 1997. ISBN 0725806303.
- [30] D. Binkley and K. Gallagher. A survey of program slicing. In M. Zelkowitz, editor, *Advances in Computers*. Academic Press, 1996.
- [31] K. B. Gallagher. Visual impact analysis. In *The International Conference on Software Maintenance - 1996 ICSM-96*, 1996.

- [32] K. Gallagher. The surgeon's assistant. In *Software Engineering Research Forum*, November 1995.
- [33] B. Kuhn, D. Smith, and K. Gallagher. The decomposition slice display system. In *The 1995 Conference on Software Engineering and Knowledge Engineering, SEKE '95*, June 1995.
- [34] J.R. Lyle, D.R. Wallace, J.R. Graham, K.B. Gallagher, J.E. Poole, and D.W. Binkley. *A CASE tool to evaluate functional diversity in high integrity software*. U.S. Department of Commerce, Technology Administration, National Institute of Standards and Technology, Computer Systems Laboratory, Gaithersburg, MD, 1995.
- [35] K. Gallagher and L. Berman. Applying metric-based object-oriented process modeling techniques to configuration management. In *The Fourth International Workshop on Software Configuration Management*, June 1993.
- [36] K. B. Gallagher and J. R. Lyle. Program slicing and software safety. In *The Eighth Annual Conference on Computer Assurance*, June 1993. COMPASS '93.
- [37] K. Gallagher. Evaluating the surgeon's assistant: Results of a pilot study. In *The International Conference on Software Maintenance - 1992 ICSM-92*, November 1992.
- [38] K. B. Gallagher. Using program slicing to eliminate the need for regression testing. In *Eighth International Conference on Testing Computer Software*, May 1991.
- [39] K. B. Gallagher. Conditions to assure semantically correct consistent software merges in linear time. In *The Third International Workshop on Software Configuration Management*, May 1991.
- [40] K. B. Gallagher and J. R. Lyle. Using program slicing in software maintenance. *IEEE Transactions on Software Engineering*, 17(8), August 1991.
- [41] K. B. Gallagher. Surgeon's assistant limits side effects. *IEEE Software*, May 1990.
- [42] K. B. Gallagher, C. Mair, A. Ramina, R. Tom, and F. Gauthier. A tool to guide safe modifications. In *International Conference on Software Engineering*, May 1989.
- [43] K. B. Gallagher and J. R. Lyle. A program decomposition scheme with applications to software modification and testing. In *Proceedings of the 22nd Hawaii International Conference on System Sciences*, January 1989.
- [44] K. B. Gallagher. *Using Program Slicing in Software Maintenance*. PhD thesis, University of Maryland, Baltimore, Maryland, December 1989.
- [45] K. B. Gallagher and J. R. Lyle. Using program decomposition to guide modifications. In *Conference on Software Maintenance - 1988*, October 1988.