

Jooyong Yi (Lee), Ph.D.

Assistant Professor
Institute of Technologies and Software Development
Innopolis University

Email: j.yi@innopolis.ru
Webpage: <http://www.jooyongyi.com>

SUMMARY

- My research areas are in *software engineering* and *programming languages*.
- I develop pragmatic methodologies and tools to *detect and fix software bugs automatically*.
- An automatic bug fixing tool I co-developed can fix the Heartbleed bug of OpenSSL automatically. In this sense, my research is also related to *software security*.
- My work has been published in ICSE, TOSEM, FSE, ASE (conference), ASE (journal), ISSTA, AOSD, IST, ICSME, and so on.

EDUCATION

- Ph.D., Computer Science
 - BRICS,¹ Aarhus University, Aarhus, Denmark
 - September 2003 – April 2007
 - Thesis: Program Validation by Symbolic and Reverse Execution
 - Advisers: Prof. Mogens Nielsen and Prof. Olivier Danvy
- M.Sc., Computer Science and Engineering
 - Theory and Formal Methods Lab., Korea University, Seoul, South Korea
 - September 2000 – August 2002
 - Thesis: Linkage of Model Checking to Debugger Using Extended JPDA
 - Adviser: Prof. Jin-Young Choi
- B.Sc., Computer Science and Engineering
 - Korea University, Seoul, South Korea
 - March 1996 – August 2000

PUBLICATIONS

- Refereed Journal Articles
 - J1. Software Change Contracts
Jooyong Yi, Dawei Qi, Shin Hwei Tan, and Abhik Roychoudhury
ACM Transactions on Software Engineering and Methodology (TOSEM'15). 24(3), 2015
 - J2. A case for dynamic reverse-code generation to debug non-deterministic programs
Jooyong Yi
Electronic Proceedings in Theoretical Computer Science (EPTCS'13) 129 – Semantics, Abstract Interpretation, and Reasoning about Programs: Festschrift for Dave Schmidt, pages 419–428, 2013.
 - J3. Efficient and formal generalized symbolic execution
Xianghua Deng, Jooyong Lee, and Robby
Automated Software Engineering (ASE'12), 19(3): pages 233-301, 2012.

¹BRICS, a Centre for Basic Research in Computer Science, specialized on Algorithmics and Mathematical Logic

Jooyong Yi (Lee)

- J4. Towards an industrial grade IVE for Java and next generation research platform for JML
Patrice Chalin, Robby, Perry R. James, Jooyong Lee, and George Karabotsos
Software Tools for Technology Transfer (STTT'10), 12(6): pages 429–446, 2010.
- J5. Filtering false alarms of buffer overflow analysis using SMT solvers
Youil Kim, Jooyong Lee, Hwansoo Han, and Kwang-Moo Choe
Information and Software Technology (IST'10), 52(2): pages 210–219, February 2010.
- Refereed International Conference Publications
 - C1. Angelix: Scalable Multiline Program Patch Synthesis via Symbolic Analysis
Sergey Mechtaev, Jooyong Yi, and Abhik Roychoudhury
ACM/IEEE International Conference on Software Engineering (ICSE'16), pages 691–701
2016.
 - C2. DirectFix: Looking for Simple Program Repairs
Sergey Mechtaev, Jooyong Yi, and Abhik Roychoudhury
ACM/IEEE International Conference on Software Engineering (ICSE'15), pages 448–458,
2015.
 - C3. Dynamic Inference of Change Contracts
Tien-Duy B. Le, Jooyong Yi, David Lo, Ferdian Thung, and Abhik Roychoudhury
IEEE International Conference on Software Maintenance and Evolution (ICSME'14), pages
451–455, 2014.
 - C4. Expressing and checking intended changes via software change contracts
Jooyong Yi, Dawei Qi, Shin Hwei Tan, and Abhik Roychoudhury
ACM International Symposium on Software Testing and Analysis (ISSTA'13), pages 1–11,
2013.
 - C5. Past expression: encapsulating pre-states at post-conditions by means of AOP
Jooyong Yi, Robby, Xianghua Deng, and Abhik Roychoudhury
ACM International Conference on Aspect-Oriented Software Development (AOSD'13), pages
133–144, 2013.
 - C6. Software change contracts
Dawei Qi, Jooyong Yi, and Abhik Roychoudhury
ACM SIGSOFT Symposium on Foundations of Software Engineering (FSE'12), NIER track,
Article No. 22, 2012.
 - C7. Bogor/Kiasan: A k-bounded symbolic execution for checking strong heap properties of open
systems
Xianghua Deng, Jooyong Lee, and Robby
IEEE/ACM International Conference on Automated Software Engineering (ASE'06), pages
157–166, 2006.
 - C8. Linkage of model checking to debugger using extended JPDA.
Joo-Yong Lee, Ki-Seok Bang, and Jin-Young Choi.
International Conference on Computer and Information Science (ICIS'02), pages 197–201,
2002.
 - C9. Systematic testing of Java programs using extended JPDA and reflection.
Joo-Yong Lee, Ki-Seok Bang, and Jin-Young Choi
Software Engineering Research and Practice (SERP'02), pages 301–306, 2002.
- Refereed International Workshop Publications
 - W1. Dynamic reverse code generation for backward execution.
Jooyong Lee.
Proceedings of the Workshop on Verification and Debugging, pages 37–54, 2006.
 - W2. Reverse code generation for Java program model checking.
Jooyong Lee.
Winter School on MOdelling and VERifying parallel Processes (MoVeP), pages 89–95, 2004.

Jooyong Yi (Lee)

- Refereed Local Conference Publications

- L1. Model Checking Object-Oriented Software Using Bandera Toolset
Ki-Seok Bang, Joo-Yong Lee, and Jin-Young Choi
KIISE (Korea Institute of Information Scientists and Engineers) Conference, pages 466–468, 2001.
- L2. A Study on Formal Verification of complex network protocol
Joo-Yong Lee, Ki-Seok Bang, and Jin-Young Choi
KIPS (Korea Information Processing Society) Conference, pages 673–676, 2001.
- L3. Case Study on Development Methodology of Safety-Critical System Using Formal Method
ChangHun Sung, JooYong Lee, Na-Young Lee, and Jin-Young Choi
KIPS (Korea Information Processing Society) Conference, pages 595–597, 2001.

- Reports

- R1. The Java Modeling Language (JML).
Gary T. Leavens, Peter H. Schmitt, and Jooyong Yi.
NII Shonan Meeting Report No. 2013-3, Available at
<http://www.nii.ac.jp/shonan/wp-content/uploads/2011/09/No.2013-3.pdf>
- R2. Tutorial on JIR (JML Intermediate Representation).
Jooyong Lee, Robby, and Patrice Chalin. 2010.

PROFESSIONAL ACTIVITIES

- Organizer

- The NII Shonan meeting on JML, 2013 with Gary T. Leavens and Peter H. Schmitt

- Reviewer (Journals)

- IEEE Transactions on Software Engineering (TSE), 2013, 2015, 2016.
- International Journal on Software Tools for Technology Transfer (STTT), 2016.
- Information and Software Technology (IST), 2014.
- Aircraft Engineering and Aerospace Technology, 2013

- PC Member

- PSC (Programming for Separation of Concerns) track of the ACM Symposium on Applied Computing (SAC), 2015.
- Foundations of Software Engineering (FSE2014) Artifact Evaluation.
- PSC (Programming for Separation of Concerns) track of the ACM Symposium on Applied Computing (SAC), 2014.
- International Workshop on Harnessing Theories for Tool Support in Software (TTSS 2013).

- External Reviewer

- International Symposium on Software Testing and Analysis (ISSTA 2016)
- International Symposium on Software Testing and Analysis (ISSTA 2016) Artifact Evaluation
- International Conference on Software Engineering (ICSE 2016)
- International Symposium on Software Testing and Analysis (ISSTA 2015)
- International Conference on Automated Software Engineering (ASE 2014)
- International Symposium on Formal Methods (FM 2014)
- European Symposium on Programming (ESOP 2014)
- International Symposium on Software Testing and Analysis (ISSTA 2013)

Jooyong Yi (Lee)

- International Conference on Automated Software Engineering (ASE 2013)
- International Conference on Fundamental Approaches to Software Engineering (FASE 2013)
- India Software Engineering Conference (ISEC 2013)
- Conference on Object-Oriented Programming Systems, Languages, and Applications (OOP-SLA 2009)
- European Symposium on Programming (ESOP 2007)
- International Static Analysis Symposium (SAS 2006)

WORK EXPERIENCE

- Assistant Professor
 - Innopolis University
 - January 2017 – Present
- (Senior) Research Fellow
 - School of Computing, National University of Singapore
 - December 2011 – December 2016 (promoted to a Senior Research Fellow as of January 2014)
- Research Professor
 - College of Information and Communications, Korea University
 - March 2011 – December 2011
- Research Fellow
 - College of Information and Communications, Korea University
 - March 2010 – February 2011
- Research Fellow
 - SANToS, Department of Computing and Information Sciences, Kansas State University
 - February 2008 – February 2010
- Research Fellow
 - Division of Computer Science, KAIST
 - August 2007 – January 2008

TEACHING EXPERIENCE

- Part-time Lecturer
 - Introduction to Information and Society (CNCE150)
 - Korea University, fall 2011
 - Undergraduate, elective, 2 credits, 45 enrollments
 - Software Security (CRE641)
 - Korea University, spring 2011
 - Graduate, 3 credits, 30 enrollments
 - Theory and Models in Systems (CRE512)
 - Korea University, fall 2010
 - Graduate, 3 credits, 39 enrollments
 - Media Programming (ND26000)
 - Sungshin Women's University, fall 2010

Jooyong Yi (Lee)

- Undergraduate, required, 2 credits, 27 enrollments
- Advanced Simulation Modeling (CRE602)
 - Korea University, spring 2010
 - Graduate, 3 credits, 21 enrollments
- Teaching Assistant
 - Software Testing and Debugging (CS4218)
 - National University of Singapore, spring 2015
 - Undergraduate, 4 credits, 53 enrollments
 - Role: tutor for two of three lab groups (group size: 14 and 21 students, respectively)
 - Parallel and Concurrent Programming (CS3211)
 - National University of Singapore, spring 2013
 - Undergraduate, 4 credits, 45 enrollments
 - Role: tutor for one of three tutorial groups (group size: 16 students)

STUDENT SUPERVISION

- PhD Students
 - Sergey Mechtaev (Co-supervised with Abhik Roychoudhury)
 - Shin Hwei Tan (Co-supervised with Abhik Roychoudhury)
 - Youil Kim (Co-supervised with Hwansoo Han)

SOFTWARE

- Angelix: a scalable semantics-based automatic program repair tool
- DirectFix: a semantics-based automatic program repair tool generating provably minimal repairs
- SemFix: the first semantics-based automatic program repair tool
- Extension of OpenJML – a flagship tool set for JML (Java Modeling Language)
 - Extension for software change contracts
 - Extension for past expressions
- JmlEclipse: an Eclipse-based tool set for JML
- OpenJIR: a bridge tool between OpenJML and JmlEclipse via JIR.
 - JIR (JML Intermediate Representation) is an intermediate representation introduced to decouple the front-end (e.g., parsing) and the back-end (e.g., checking) of JML tools
- Kiasan: a JML checker based on an efficient symbolic execution technique
- Raccoon: an abstract interpretation tool used for the study of false alarm filtering

AWARDS

- Graduate Study Abroad Program funded by NRF (National Research Foundation of Korea), 2003–2006