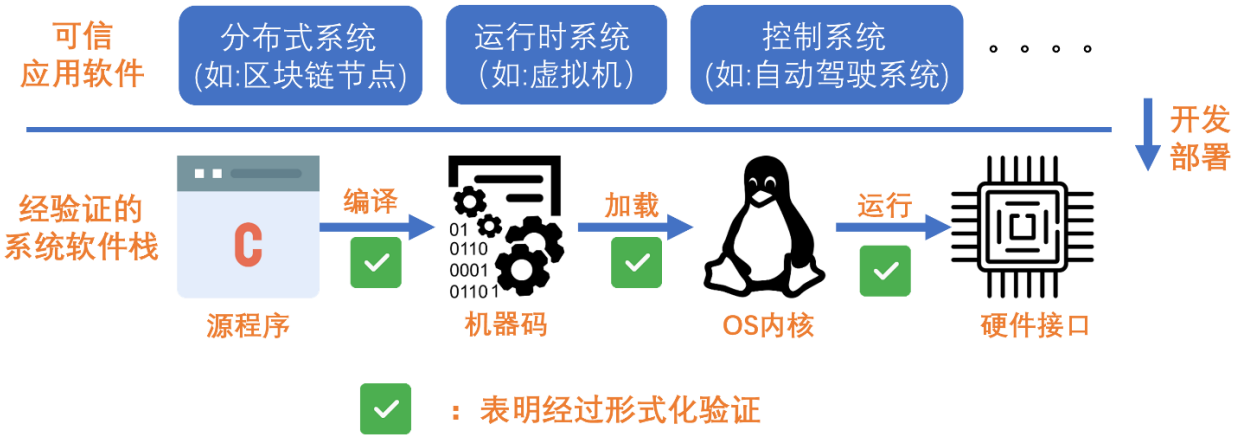
汪宇霆

单位：上海交通大学约翰•霍普克罗夫特计算机科学中心
职位：长聘教轨副教授
地址：上海市闵行区东川路 800 号，上海交通大学闵行校区电信群楼 1-203 室
邮箱：yuting.wang@sjtu.edu.cn
主页：http://jhc.sjtu.edu.cn/~yutingwang

研究领域介绍

本人主要研究软件的形式化验证，包括形式化验证理论（包括程序设计语言理论、类型论、证明论、逻辑框架等）、基于这些理论的工具开发，以及它们在可信软件（Certified Software）中的应用。

本人目前专注研究构建可信系统软件（如编译器和操作系统）的形式化方法，以此得到经过完整形式化验证的系统软件平台，为可信应用软件的开发和灵活部署提供扎实的基础。如下图所示：



本人还积极研究开发新一代定理证明工具，该部分研究围绕着 Abella 定理证明器（http://abella-prover.org/）展开，其特点是支持高阶抽象语法（Higher-Order Abstract Syntax），从根本上简化复杂程序的验证复杂度。Abella 已被成功应用于学术界多个重要的程序验证项目。

教育背景

- 明尼苏达大学双城校区，计算机科学，博士研究生，2011.09 - 2016.12，美国

  导师：Gopalan Nadathur

  论文：A Higher-Order Abstract Syntax Approach to the Verified Compilation of Functional Programs （获得**明尼苏达大学研究生院博士论文奖学金**）

- 康涅狄格大学，计算机科学与工程，硕士研究生，2009.09 – 2011.08，美国

  导师：Laurent Michel

  论文：AMIBE: an Imperative Programming Language with First Class Continuations
- 上海交通大学，电力系统及其自动化，硕士研究生，2006.09 – 2009.02，中国
- 上海交通大学，电气工程与自动化，本科生，2002.09 – 2006.06，中国

  （**2006 年上海交通大学优秀毕业生**）

工作经历（包括实习经历）

- 上海交通大学约翰•霍普克罗夫特计算机科学中心，长聘教轨副教授，2020.03 - 至今，中国
- 耶鲁大学计算机系，博士后研究员，2016.12 – 2019.12，美国

  指导教授：Zhong Shao（邵中，耶鲁大学计算机系主任）

  参与项目：**深度规约的科学研究** (The Science of Deep Specification https://deepspec.org)，该项目为普林斯顿大学、耶鲁大学、麻省理工和宾夕法尼亚大学共同合作，为期五年的 NSF 探索性计算（Expedition in Computing）项目，研究使用深度规约形成完整的可信软件生态链。本人主要参与其中**可信编译器和操作系统的研究。**
- 法国国家信息与自动化研究所（INRIA）Saclay 分部，暑期实习研究生，2012、2014 年夏，巴黎，法国

  指导教授：Kaustuv Chaudhuri，Dale Miller

  参与项目：**Abella 定理证明器的理论基础及其实现研究**

代表性论文

如无特殊说明，作者按贡献大小排序，带#号标注为共同第一作者，带*号标注为通讯作者。计算机领域国内评价体系由中国计算机学会(CCF)制定，分为 A-C 类会议和期刊。

- Ling Zhang; **Yuting Wang***; Jinhua Wu; Jeremie Koenig; Zhong Shao. Fully Composable and Adequate Verified Compilation with Direct Refinements between

Open Modules. *Proceedings of the ACM on Programming Languages*, 2024, 8(**POPL**): 72:1-72:31.

注： CCF 评级 A 类会议（以期刊形式发表）

- Jinhua Wu; **Yuting Wang***; Meng Sun; Xiangzhe Xu; Yichen Song. Verified Transformation of Continuation-Passing Style into Static Single Assignment Form. In *Proceedings of the 21st Asian Symposium on Programming Languages and Systems (**APLAS**)*, pages xx-xx, Taipei, Taiwan (China), 2023.

  注： CCF 评级 C 类会议

- Siyu Liu; **Yuting Wang***. Verified Transformation of Continuation-Passing Style into Static Single Assignment Form. In *Proceedings of the 17th International Symposium on Theoretical Aspects of Software Engineering (**TASE**)*, pages 20-37, Bristol, UK, 2023.

  注： CCF 评级 C 类会议

- **Yuting Wang**; Ling Zhang; Zhong Shao; Jeremie Koenig. Verified Compilation of C Programs with a Nominal Memory Model. *Proceedings of the ACM on Programming Languages*, 2022, 6(**POPL**): 25:1-25:31.

  注： CCF 评级 A 类会议（以期刊形式发表）

- Xiangzhe Xu; Jinhua Wu; **Yuting Wang***; Zhenguo Yin; Pengfei Li. Automatic Generation and Validation of Instruction Encoders and Decoders. In *Proceedings of the 33rd International Conference on Computer-Aided Verification (**CAV**)*, pages 728-751, Virtual Conference, 2021.

  注： CCF 评级 A 类会议

- **Yuting Wang**; Xiangzhe Xu; Pierre Wilke; Zhong Shao. CompCertELF: Verified Separate Compilation of C Programs into ELF Object Files. *Proceedings of the ACM on Programming Languages*, 2020, 4(**OOPSLA**): 197:1-197:28.

  注： CCF 评级 A 类会议（以期刊形式发表）

- **Yuting Wang**; Pierre Wilke; Zhong Shao. An Abstract Stack Based Approach to Verified Compositional Compilation to Machine Code. *Proceedings of the ACM on Programming Languages*, 2019, 3(**POPL**): 62:1-62:30.

  注： CCF 评级 A 类会议（以期刊形式发表）

- Gopalan Nadathur[#]; **Yuting Wang**[#]. Schematic Polymorphism in the Abella Proof Assistant. *In Proceedings of the 20th International Symposium on Principles and Practice of Declarative Programming (**PPDP**)*, pages 15:1-15:13, Frankfurt am Main, Germany, 2018.

  注：两位作者贡献相同，为共同第一作者，按姓氏首字母排序

- **Yuting Wang**; Gopalan Nadathur*. A Higher-Order Abstract Syntax Approach to Verified Transformations on Functional Programs. In *Proceedings of the 25th European Symposium on Programming (**ESOP**)*, pages 752-779, Eindhoven, The Netherlands, 2016.

  注：CCF 评级 B 类会议

- **Yuting Wang**; Kaustuv Chaudhuri. A Proof-theoretic Characterization of Independence in Type Theory. In *Proceedings of the 13th International Conference on Typed Lambda Calculi and Applications (**TLCA**)*, pages 332-346, Warsaw, Poland, 2015.

- David Baelde[#]; Kaustuv Chaudhuri[#]; Andrew Gacek[#]; Dale Miller[#]; Gopalan Nadathur[#]; Alwen Tiu[#]; and **Yuting Wang**[#]. Abella: A System for Reasoning about Relational Specifications. *Journal of Formalized Reasoning (**JFR**)*, 2014, 7(2): 1-89.

  注：JFR 特邀文章，作者贡献相同，按姓氏首字母排序

- **Yuting Wang**; Kaustuv Chaudhuri; Andrew Gacek; Gopalan Nadathur. Reasoning about Higher-Order Relational Specifications. In *Proceedings of the 15th Symposium on Principles and Practice of Declarative Programming (**PPDP**)*, pages 157–168, Madrid, Spain, 2013.

- **Yuting Wang**; Gopalan Nadathur. Towards Extracting Explicit Proofs from Totality Checking in Twelf. *In Proceedings of the 8th ACM SIGPLAN International Workshop on Logical Frameworks and Metalanguages: Theory and Practice (**LFMTP**)*, pages 55-66, Boston, USA, 2013.

重要软件开发项目

- **Stack-Aware CompCert** （https://certikos.github.io/compcertmc/）:

  CompCert（http://compcert.inria.fr/）作为应用最为广泛的经形式化验证的 C 语言编译器，其目标语言为一种抽象的汇编语言。从该汇编语言到机器代码的编译过程依赖于未经验证的外部工具（如 GNU 汇编器）。Stack-Aware CompCert 通过扩展 CompCert 实现了从源程序到机器代码完整编译过程的验证。其关键创新点如下：通过扩展 CompCert 的内存模型使其支持明确的栈结构，以证明程序的栈空间消耗能被编译过程保存，最终将栈空间和数据及代码段映射到实际机器语言支持的有限内存模型。通过上述扩展，Stack-Aware CompCert 实现了从 CompCert 汇编到机器代码的验证，并结合 CompCert 实现了从 C 程序到机器代码的端到端编译过程的验证。Stack-Aware CompCert 同时还提供了一种名为 Contextual Compilation 的模块化编译方法，为经验证的操作系统内核 CertiKOS 提供了模块化编译验证的基础。Stack-Aware CompCert 由本人与 Pierre Wilke，Zhong Shao 合作开发。基于 Stack-Aware CompCert，本人还设计开发了首个能将 C 程序编译到标准格式机器码(ELF 格式)的经验证编译器 CompCertELF（详见 OOPSLA 2020 文章)。

- **CeriKOS** （http://flint.cs.yale.edu/certikos/）：

  CertiKOS 是耶鲁大学计算机系系主任 Zhong Shao 教授的小组开发的新一代基于形式化验证的操作系统内核。其系统构架基于深度规约（Deep Specifications)。该类规约以数学逻辑精确描述操作系统各模块的功能，支持模块之间的横向及纵向组合，从而使得模块化的操作系统验证第一次成为可能。本人在 CertiKOS 项目中负责研究使用 Stack-Aware CompCert 为 CertiKOS 提供模块化编译功能，以及验证用户程序的加载和其在 CertiKOS 内核中的正确运行。

- **Abella** （http://abella-prover.org）：

  Abella 是明尼苏达大学和法国国家信息与自动化研究所（INRIA）联合开发的基于高阶抽象语法（Higher-Order Abstract Syntax）的逻辑证明框架。其特点是支持复杂的高阶语言和逻辑系统的形式化建模及验证，已被成功应用于多种程序语言的形式化设计、程序语言元理论的验证、以及函数式编译器验证。本人从 2013 年起作为 Abella 开发和维护的核心团队成员，负责其基础理论研究、验证方法设计、以及这些方法的实现。主要研究成果包括在 Abella 2.0.0 版本中加入对高阶关系式规约的支持（与 INRIA 研究员 Kaustuv Chaudhuri 的合作成果）以及在 2.0.6 版本中加入对多态验证的支持（与 Gopalan Nadathur 的合作成果）。

- **AMIBE** （http://digitalcommons.uconn.edu/gs_theses/142/）：

  AMIBE 是一门基于约束式编程（Constraint Programming）的命令式语言，由本人与 Laurent Michel 教授共同开发。作为该工作的一部分，本人在 C++中设计并实现了 AMIBE 编译器。该编译器利用编译器基础框架 LLVM（https://llvm.org/）提供的对于尾函数调用的优化，实现了约束式编程中关键控制结构的高效调用。

- **VSTLF** (https://github.com/ldmbouge/vstlf):

  Very Short Term Load Forecasting (VSTLF) 是一款基于机器学习的电力系统负载预测软件。其功能是通过由大量历史数据训练而成的多重过滤式神经网络对地区电网的负载提供短期实时预测（从分钟到小时），从而为电力市场提供竞价标准。 从 2009 年到 2011 年，本人是 VSTLF 程序的主要开发和调试人员。该系统最终被法国阿尔斯通公司购买并整合进入该公司的 e-terraplatform 平台，用于提供地区电网操作中心监控系统的解决方案。VSTLF 使用 Java 编写，在 Laurent Michel 和 Peter Luh 教授的指导下进行开发。

教学经历

- 任课教师：CS171 - 程序设计思想与方法, 上海交通大学，2020 年秋；

- 课程助教：CSCI4011 - Formal Languages and Automata Theory (形式语言及自动机理论), 明尼苏达大学双城校区，2015 年秋，美国
- 本科论文指导：Automating the Proofs of Strengthening Lemmas in the Abella Proof Assistant，Dawn Michaelson，明尼苏达大学双城校区，2016 年秋，美国

  注：Dawn Michaelson 现为明尼苏达大学博士研究生
- 暑期实习指导：Andrew Wu (吴昊泽), Davidson College，2017 年夏，美国

  注：吴昊泽现为斯坦福大学博士研究生

学术活动

- 学术委员会成员：
  - Certified Programs and Proofs, 2023，波士顿，美国
  - International Conference on Formal Techniques for Distributed Objects, Components, and Systems, 2023, 里斯本，葡萄牙
  - Logical Frameworks and Meta-Languages: Theory and Practice (LFMTP 2018, associated with FLoC 2018)，2018.07，牛津大学，英国
  - International Conference on Functional Programming (ICFP 2019, Artifact Evaluation Committee)，2019.08，柏林，德国
  - International Conference on Functional Programming (ICFP 2020, Artifact Evaluation Committee)，2020.08，新泽西，美国
- 学术组织：
  - 中国计算机学会形式化方法专业委员会通讯委员

获奖经历

- 明尼苏达大学博士论文奖学金(Doctoral Dissertation Fellowship)，2014-2015
- 明尼苏达大学研究生理事会旅行奖学金，2016
- NSF-INRIA REUSSI 旅行奖学金, 2012
- 上海交通大学光华奖学金二等奖，2008

- 上海交通大学优秀毕业生，2006

主要国际合作者

- Zhong Shao，耶鲁大学计算机系主任（http://www.cs.yale.edu/homes/shao/）
- Dale Miller，INRIA 科研主任（http://www.lix.polytechnique.fr/Labo/Dale.Miller/）
- Gopalan Nadathur，明尼苏达大学教授（https://www-users.cs.umn.edu/~ngopalan/）
- Kaustuv Chaudhuri，INRIA 研究员（https://chaudhuri.info/）

学术演讲

- 面向一阶语言的可组合编译器验证。中国软件大会. 2023.12，上海，中国
- 端到端的可组合编译器验证。CCF 系统软件专委编译器技术论坛. 2023.7，天津，中国
- 基于抽象内存模型的 C 程序编译验证。中国计算机大会. 2022.12，线上会议
- 基于名义内存模型的 C 程序编译验证。中国软件大会. 2021.12，线上会议
- 指令编码译码的形式化验证方法。中国计算机大会. 2021.12，线上会议
- Verified Compilation of C Programs with a Nominal Memory Model. POPL 2022. 2022.01, 在线会议.
- End-to-End Verified Compilation of Heterogenous Programs. CCF 形式化方法专委会 2021 战略研讨会. 2021.10，中国科学院软件所，北京，中国
- 从 C 程序到二进制对象文件分离编译的验证方法研究。SKLCS Seminar. 2020.12，中国科学院软件所，北京，中国
- CompCertELF: Verified Separate Compilation of C Programs into ELF Object Files. The 2020 ACM Conference on Object-oriented Programming, Systems, Languages, and Applications. 2020.11，在线会议
- Stack-Aware CompCert: Verified Compositional Compilation of C Programs into Machine Code. *The DeepSpec@PLDI 2019 workshop*. 2019.06，凤凰城，美国

- Compilation Support for End-to-End Formal Verification of Software Systems. *The 4th SJTU Future Information Technology International Forum for Young Scholars (SIFYS)*. 2019.06，上海交通大学，上海，中国
- Stack-Aware CompCert: Verified Compilation of C Programs into Machine Code. *Seminar in John Hopcroft Center*. 2019.02，上海交通大学，上海，中国
- Schematic Polymorphism in the Abella Proof Assistant. *The 20th International Symposium on Principles and Practice of Declarative Programming*. 2018.09，法兰克福，德国
- The Higher-Order Abstract Syntax Approach to Verified Transformations on Functional Programs. *The 25th European Symposium on Programming (ESOP)*. 2016.04，埃因霍温，荷兰
- A Framework for Verified Compilation of Functional Programs. *Seminar in Prof. Zhong Shao's group (invited talk)*. 2016.03，耶鲁大学，纽黑文，美国
- A Proof-Theoretic Characterization of Independence in Type Theory. *The 13th International Conference on Typed Lambda Calculi and Applications*. 2015.07，华沙大学，华沙，波兰
- Verified Transformations of Functional Programs. *Midwest Verification Day 2014*. 2014.10，密苏里大学，哥伦比亚，美国
- Verified Functional Program Transformations Using Higher-Order Abstract Syntax. *Parsifal Seminar*. 2014.06, 巴黎综合理工学院，巴黎，法国
- Towards Extracting Explicit Proofs from Totality Checking in Twelf. *The 8th ACM SIGPLAN International Workshop on Logical Frameworks and Metalanguages: Theory and Practice.* 2013.09, 波士顿，美国
- The Abella Approach to Specifying and Reasoning about Formal Systems. *Midwest Verification Day 2012.* 2012.09, 堪萨斯大学，劳伦斯，美国

- New Developments with the Abella System. *Workshop on Abella and Bedwyr*. 2012.07, 巴黎综合理工学院，巴黎，法国
- AMIBE: an Imperative Programming Language with First Class Continuations. *Midwest Verification Day 2011*. 2011.09, 明尼苏达大学，明尼阿波利斯，美国