

# Yuxiang Lei, Ph.D. in Software Engineering

✉ [youthyounggray@gmail.com](mailto:youthyounggray@gmail.com)

🌐 <http://kisslune.github.io>

🆔 0000-0002-4484-8172

🌐 [LinkedIn](#)



## Summary

Yuxiang Lei, PhD, graduated from the School of Computer Science at the University of Technology Sydney (UTS) and has recently completed a one-year postdoctoral fellowship at the University of New South Wales (UNSW). His research focuses on formal languages, compiler theory, and program analysis. Noteworthy achievements include publications in top-tier conferences such as PLDI, OOPSLA, and SAS. Along with his academic research, he actively participates in the open-source community. He is in charge of developing and maintaining of two open-source program analysis projects, *POCR* and *SVF*, which are recognized with over 1k stars on GitHub and widely used by peers, with over 20 papers published on top-tier conferences.

## Employment History

2023 – 2024    📌 **Postdoc Fellow**, University of New South Wales, NSW, Australia.  
**Supervisor:** A.Prof. Yulei Sui.  
**Research field:** formal language and compiler theory.  
**Focus:** syntax analysis, parser, graph-based analysis, performance optimization.

## Education

2018 – 2023    📌 **Ph.D. in Software Engineering**, University of Technology Sydney, NSW, Australia.  
**Research field:** program analysis.  
**Supervisor:** A.Prof. Yulei Sui.  
**Focus:** static analysis, C/C++, program abstraction, performance.  
**Thesis title:** *Improving the Efficiency of Graph-Based Static Analysis*.

## Awards

2022    📌 ACM SIPLAN Distinguished Artifact Award.  
2019    📌 Radhia Cousot Young Researcher Best Paper Award.



## Research Publications

1. **Yuxiang Lei**, Camille Bossut, Yulei Sui and Qirun Zhang, “Context-free language reachability via skewed tabulation”, *Proceedings of the ACM on Programming Languages*, vol. 8, PLDI 2024. (CCF-A)
2. Pei Xu, **Yuxiang Lei\***, Yulei Sui and Jingling Xue, “Iterative-epoch online cycle elimination for context-free language reachability”, *Proceedings of the ACM on Programming Languages*, vol. 8, pp. 1437–1462, OOPSLA1, 2024. (CCF-A)
3. **Yuxiang Lei**, Yulei Sui, Shin Hwei Tan and Qirun Zhang, “Recursive state machine guided graph folding for context-free language reachability”, *Proceedings of the ACM on Programming Languages*, vol. 7, pp. 318–342, PLDI 2023. (CCF-A)



4. **Yuxiang Lei**, Yulei Sui, Shuo Ding and Qirun Zhang, “Taming transitive redundancy for context-free language reachability”, *Proceedings of the ACM on Programming Languages*, vol. 6, pp. 1556–1582, *OOPSLA2 2022*. (CCF-A)
5. **Yuxiang Lei** and Yulei Sui, “Fast and precise handling of positive weight cycles for field-sensitive pointer analysis”, *Static Analysis: 26th International Symposium, SAS 2019*. (CCF-B)
6. Jin Gou, **Yuxiang Lei**, Wangping Guo, Yiqiao Cai and Wei Luo, “A novel improved particle swarm optimization algorithm based on individual difference evolution”, *Applied Soft Computing*, vol. 57, pp. 468–481, 2017.

## Projects




### Tools

- 2020 – now      **POCR**, a context-free language reachability analysis tool.  
**Role:** Creator and developer.  
**Award:** ACM SIPLAN Distinguished Artifact Award (2022).  
**Recognition:** The techniques implemented in this tool were recognized with 4 papers published at top-tier conferences PLDI (CCF-A) and OOPSLA (CCF-A). This tool is also widely used by peers, with 7 papers published at top-tier conferences.
- 2018 – now      **SVF**, an LLVM-based C/C++ program analysis tool.  
**Role:** Developer.  
**Awards:** Radhia Cousot Young Researcher Best Paper Award (2019).  
**Recognition:** 1.3k stars on GitHub. 18 papers were accepted and published in top-tier conferences based on this tool.

### Research Projects

- 2023 – now      *Program Analysis with Precise Abstractions*.  
**Sponsor:** U.S. National Science Foundation  
**Role:** Participant. In charge of boosting the precision and efficiency of program analysis via optimizing the abstract graph models of programs.
- 2022 – now      *Learning Software Security Analysers with Imperfect Data*.  
**Sponsor:** Australian Research Council.  
**Role:** Participant. In charge of optimizing the logic of the online analyzer.

## Skills

- Programming      C/C++, Python, Matlab, GNU, LLVM, Git, Docker, Linux, SQL,  $\text{\LaTeX}$ .
- Technical      formal language, automata, program analysis, compiler theory software security.
- Misc.      academic research, supervision, technical writing, teaching.