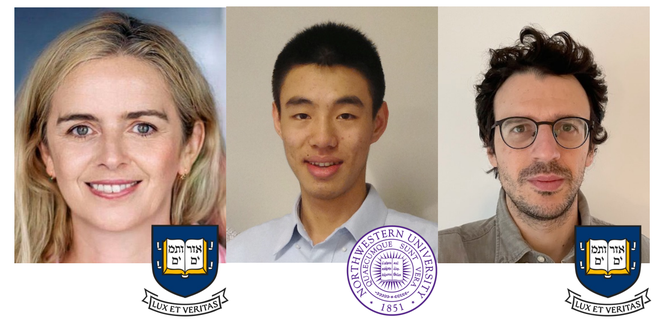
Automating and Synthesizing Parallel Zero-Knowledge Protocols

Ruzica Piskac (Yale), Xiao Wang (Northwestern), Timos Antonopoulos (Yale)



https://www.nsf.gov/awardsearch/showAward?AWD_ID=2318974
https://www.nsf.gov/awardsearch/showAward?AWD_ID=2318975

Context: Zero-knowledge proofs (ZKPs) enable a prover to convince a verifier about the truth of some statement without revealing why. Recent advances by the cryptographic research community have brought a tremendous improvement in its efficiency and numerous creative applications. However, various practical challenges make the adoption of ZK unfeasible for tasks of realistic sizes.

Our solution: A comprehensive toolkit of programming languages and compilers providing a full suite of effective, intuitive, parallel-aware, and general methods for writing complex statements intended for efficient ZKPs.

Challenges

1) Automatically **identify the best distribution strategy** for ZKP protocols.

2) Securely **support randomized challenges** and enable the distribution of statements.

3) Scale our framework to handle **ZKP statements of large size.**

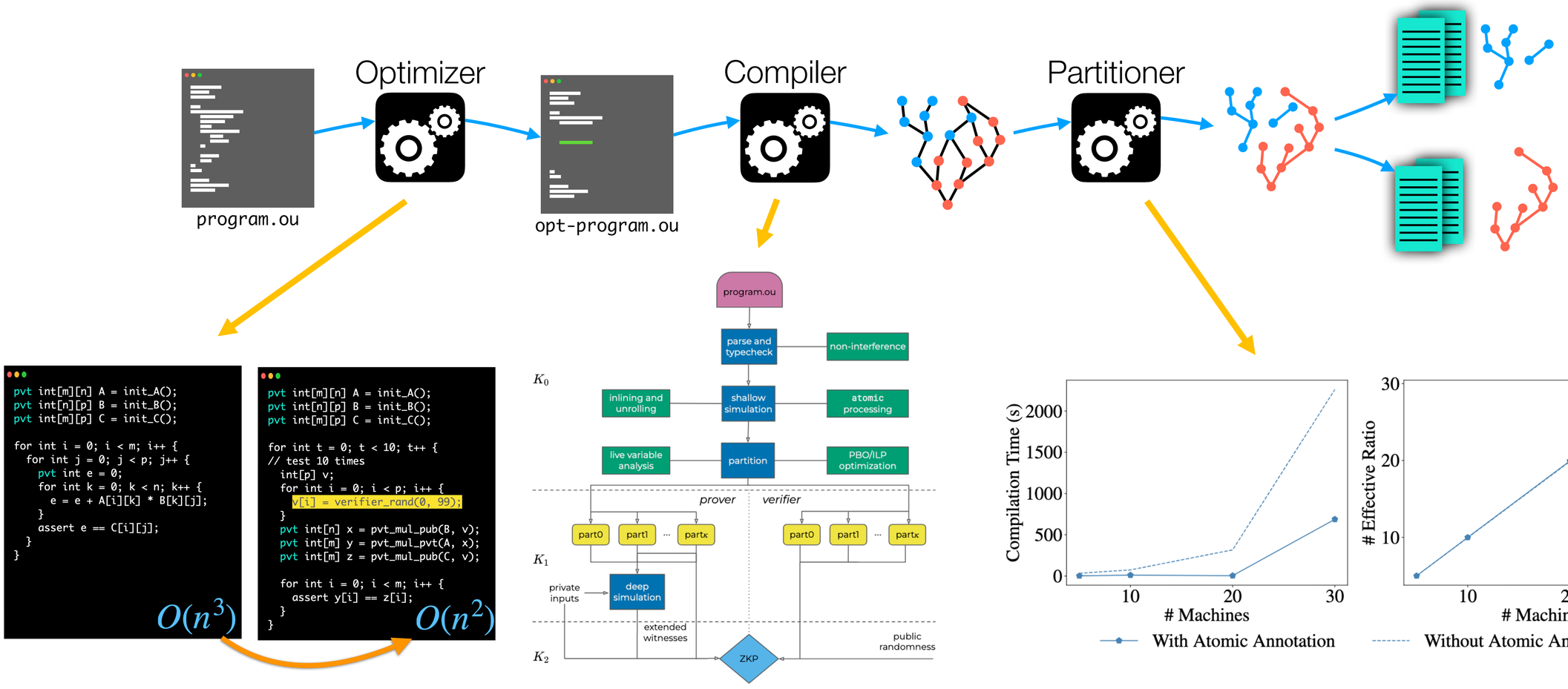
4) Use formal methods techniques to prove that a given program and its optimized version **exhibit the same functionality and security**.

5) **Automatically synthesize ZK-specific optimizations** taking advantage of prover-provided hints and verifier-provided randomness?

Scientific Impact

- *Formal Methods:* Static and dynamic analysis to infer properties of programs that lead to ZK-specific optimizations. *E.g.,* automatically inferring a computation is faster with a prover-provided witness.
- *Program Analysis:* Live variable analysis and simulations, based on prover witnesses and verifier-provided randomness, to find optimal circuit. Allow programmers to specify knowledge-levels and assist compiler with atomic processing.
- *Optimization:* Convert partitioning to pseudo-Boolean optimization to optimize cuts.

Major Publications: Sang et al. *Ou: Automating the Parallelization of Zero-Knowledge Protocols* [CCS 2023]



Impact on Society

Our project will **accelerate the deployment of ZKP**. Our language and framework will **enable non-expert programmers** to write efficient and intuitive ZK programs. This would allow bringing auditability and transparency to **legal, financial and healthcare** systems among others.

Impact on Education

- Ning Luo, PhD student with PI Piskac and Postdoc with PI Wang, joined UIUC as a tenure-track assistant professor
- Joint curriculum development between formal methods and cryptography

Broader Participation

- Open-source analysis tools, compilers and languages. Tools often offered through web interfaces.
- Involving Undergraduate Researchers. E.g., undergraduate student at Yale currently building Vehicle miles traveled application using Ou