Tolik Zinovyev

tolik@bu.edu

Research Experience

Approximate Lower Bound Arguments	Boston University	2023-2024

Approximate Lower Bound Arguments (ALBA) is a new cryptographic primitive for efficiently proving knowledge of many data elements of some kind (e.g. signatures) in a succinct, non-interactive manner. The proof is approximate because there is gap between the number of elements the prover knows and the number of elements the verifier is convinced the prover knows. This gap enables very efficient schemes. Applications include weighed multisignatures in the decentralized setting and designing universally composable SNARKs.

Joint work with Pyrros Chaidos, Aggelos Kiayias and Leo Reyzin.

Scalable routing in ad hoc networks	GMU / BU / Independent	Ongoing

A scalable routing protocol for dynamic ad hoc networks. The network structure is similar to Kademlia DHT. Whereas Kademlia requires direct connectivity between any two nodes (provided by IP), this protocol satisfies the Kademlia invariant and guarantees connectivity between any two nodes for any physical network topology.

A constant routing stretch and a sub-linear bound on space complexity has been proven for the static case for a (realistic) restricted class of graphs. Constant routing stretch and sub-linear communication, space, and computational complexity in the dynamic case is expected.

The distributed protocol has been implemented using ns-3 for discrete-event network simulation and tangible performance benefits have been observed over a traditional state-link routing protocol for networks with a couple thousand nodes. The ongoing efforts are focused on increasing the number of nodes simulated to demonstrate practical scalability. As part of this, many performance optimizations and bug fixes for ns-3 have been developed and many of those accepted upstream.

Education

PhD in Computer Science	Boston University	Started Fall 2018

- Current GPA – 3.9
- Courses: Cryptography (1, 2), Distributed Systems, Abstract Algebra, Computational Complexity, Communication Complexity, Randomness in Computing, Coding Theory, Algorithms, Programming Languages
- Teaching Assistant: Discrete mathematics CS 131 (Fall 2018, Spring 2019, Fall 2019, Summer 2020), Algorithms CS 330 (Spring 2020), Graduate Networks CS 655 (Fall 2020), Distributed Systems CS 651 (Spring 2021)

B.S. in Mathematics	George Mason University	Graduated May 2018

- GPA – 3.93
- Mathematics courses: Calculus, Discrete Mathematics, Linear Algebra, Number Theory, Numerical Analysis, Differential Equations, Probability, Combinatorics, Abstract Algebra
- Other technical courses: Theoretical CS, Data Structures

Employment

Software Engineer	Algorand	2021-2022

- Indexer (repo, commits)
  - Designed and executed a thorough refactor of the block import code, a large part of Indexer, to improve correctness by utilizing go-algorand's ledger accounting code; no production issues since then
  - Improved block import performance (in transactions per second) by 10X compared to the version before refactoring using techniques such as batched queries, parallelism, etc
  - Worked with product managers to design a policy for deprecating old database migrations to reduce maintenance burden
- go-algorand (repo, commits in master, commits in feature/320-rounds)
  - Implemented two-stage catchpoint (ledger snapshot) generation as part of a larger project to reduce memory usage and improve the blockchain throughput: for a round (block) X catchpoint, snapshot account state at round X-320 and later package it with the hash of block X (320 being the balances lookup lookback needed by the consensus algorithm)
  - Performed a security review of the new transaction propagation protocol; found several vulnerabilities and general bugs
  - Found and fixed an accounting bug in Algorand rewards calculation that could lead to a network stall

Software Engineering Intern	Google PS1	Summer 2018

- Designed and implemented a scheduling policy for a map reduce like system
  - Designed a policy that prioritizes users with little resource usage history
  - Implemented the scheduling algorithm for use in production

Software Engineering Intern	Google Spanner	Summer 2017

- Designed and implemented a multi-user cache replacement policy
  - Designed a policy that stands in the middle between the global Least Recently Used policy (global efficiency) and the max-min fairness (fairness).
  - Proved a fairness property.
  - Implemented a prototype: a container-like C++ class that allows insertion by copying, moving and constructing elements in place.
- Implemented a schema linter
  - Spanner is not a usual database and using it efficiently requires knowing some of its specifics. This tool warns the user of some of the common pitfalls.

Software Engineering Intern	Google Fiber	Summer 2016

- Co-designed and co-implemented a multicast delivery system
  - co-designed a file multiplexing protocol
  - co-designed distributed backend services and implemented a part of them
  - reviewed code
- Quickly fixed priority bugs
- Received a peer bonus

Data Science Intern	Resonate	Summer 2015

- Improved average models' accuracy by 4% by experimenting with different gradient descent parameters
- Performed feature clustering on sparse data to reduce dimensionality; improved average models' accuracy by 1%
- Reduced time complexity of an algorithm on sparse matrices from O(N*M*log(NNZ/N)) to O(NNZ), thus bringing down data preprocessing time from 45 minutes to 1 minute

Publications

- Approximate Lower Bound Arguments by Pyrros Chaidos, Aggelos Kiayias, Leonid Reyzin, Anatoliy Zinovyev. Appeared in EUROCRYPT 2024.
- Space-stretch tradeoff in routing revisited by Anatoliy Zinovyev. Appeared in DISC 2022.

Awards

- II place in the national programming competition (Ukraine, 2013)
- III place in the national programming competition (Ukraine, 2012)
- III place in the regional mathematics competition (Ukraine, 2012)

Programming Languages and Technologies

- C++, Go, Rust
- Bazel
- Linux