Multiparty Computation (MPC) with Guaranteed Output Delivery Implementation and Optimization

Albert Gao

11/08/2020

1 Project Description

1.1 People

I will be primarily working with Vipul Goyal (`https://www.cs.cmu.edu/~goyal/`), an Associate Professor in the Computer Science Department at CMU with a research focus on cryptography and theoretical computer science. He teaches the undergraduate theory course on Introduction to Cryptography, which I have taken in the Fall 2019 semester.

One of his past mentees enrolled in the Master's Program at CMU was Hanjun Li, now a PhD student at the University of Washington. Hanjun has partially started this project before graduating from CMU, but the concrete results did not match with the state-of-the-art theoretical performance. I will communicate with him regarding the past progress.

Lastly, another undergraduate student at CMU, Fan Pu Zeng, is independently interested in this project with professor Goyal. I will collaborate closely with him throughout the next semester.

1.2 Background

Multiparty computation has a relatively long history of research dating back to the early 1980s. Intuitively, the goal of MPC is to compute a generic function based on inputs from different parties who may not trust each other, without sacrificing the privacy of the secret inputs during or after the computation.

Over the past three decades, dozens of papers have emerged regarding the communication complexity of MPC, with a few varients for the its model.

This proposed project, as titled, indeed involves achieving concrete communication complexity for MPC under a series of specific assumptions. In particular, we assume that

- Adversaries are malicious. This means that the corrupt parties in the MPC scheme are not merely curious about the private information of other honest parties, but can also deviate from the MPC protocols.
- Adversaries are computationally unbounded. This means that we are in an information-theoretical setting and do not rely on other cryptographic assumptions.
- Adversaries are the minority. The assumption of "honest majority" exists since this model is optimal — had the adversaries taken majority, we would not be able to give guarantees for the honest parties.
- Adversaries do not have the abort operation. The adversaries may not halt the computation after obtaining their own outputs.

1.3 Overview

The theoretical foundation for this project has been established in the most recent paper "Guaranteed Output Delivery Comes Free in Honest Majority" by Vipul Goyal, Yifan Song, and Chenzhi Zhu. Importantly, it is hypothesized that the complexity required for our current model is linear with respect to the number of parties involved in the MPC protocol, a strict improvement over the past results. Specifically, the complexity is shown to be $O(Cn\phi)$, where C denotes the number of gates in the arithmetic circuit used for our computation, n denotes the number of parties in the MPC protocol, and ϕ denotes the maximum number of bits needed to represent one element of the finite field we are computing on.

An implementation of this result, as discussed earlier, has been attempted but did not achieve the theoretical complexity. My research project will focus on a practical implementation that achieves, improves upon, or contradicts the 5.5 field element per party per gate concrete complexity as hypothesized in the paper. Oftentimes new problems arise when we bring theory to practice in the field of cryptography, and additional research opportunities may become apparent — especially when this implementation here has never been achieved before.

This implementation project will serve as a model for future MPC protocols with similar assumptions and I hope to achieve better concrete complexity results and/or better clarity for the theoretical proposals. Despite a lack of tangible theoretical motivation on this topic at the moment, I believe this project will prove to be intellectually stimulating enough to provide insights for other seemingly unrelated areas in cryptography. Finally, I will appeal to a quote whose source I cannot seem to find:

> In theory there is no difference between theory and practice. In practice there is.

1.4 Website

My project information and progress will be documented at `http://adbforlife.github.io/mpc`.

2 Project Goals

It is somewhat implausible to predict the exact results of this project, given the exploratory nature of this endeavor itself. Hence, I roughly outline three levels of completeness for this project here, and all of these are subject to change given progress during my readings / reflections.

- (**75%**). Fully implement and test the proposed protocol in the 2020 paper by Vipul, Yifan, and Chenzhi, achieving the concrete complexity of 5.5 field element per party per gate.
- (**100%**). Propose a change in the existing protocol that achieves better concrete complexity OR pinpoint the part of the protocol that could not be implemented efficiently in practice OR if the previous two goals are not possible, provide a suite of open source MPC implementations for varying assumptions.
- (**125%**). Prove a lower bound on the communication complexity for the current model OR propose a new protocol for MPC under different assumptions that is asymptotically better than the state-of-the-art.

3 Milestones

This is mostly a sketch of the different parts of the problem I intend to resolve, and all of these are subject to change.

- **End of Semester**. Carefully peruse all papers relevant to the topic listed under Resources section.
- **February 15th**. Contact Hanjun Li and discuss previous progress and difficulties on implementations. Set up meetings and coordinate work with collaborator Fan Pu Zeng.
- **March 1st**. Provide an implementation for the secret sharing schemes and establish infrastructure for distributed algorithms.
- **March 15th**. Provide an implementation for the paper "Scalable and Unconditionally Secure Multiparty Computation" published in 2007 by Ivan Damgard and Jesper Buus Nielsen. This is a necessary foundation for the other implementations.
- **March 29th**. Provide an implementation for the paper "Malicious Security Comes Free in Honest-Majority MPC" published in 2020 by Vipul Goyal and Yifan Song. This is a necessary foundation for the final protocol.
- **April 12th**. Provide 50% implementation for the paper "Guaranteed Output Delivery Comes Free in Honest Majority MPC".
- **April 26th**. Complete implementation for the paper "Guaranteed Output Delivery Comes Free in Honest Majority MPC".
- **May 10th**. Test existing implementations and explore improvements to the existing protocols for the current model or otherwise.

4 Resources

4.1 Literature

There is **significant** background material to read and internalize. There has been almost 40 years of continued interest in MPC from the resource community of the world. The complexity of protocols and arguments has only increased on a yearly basis. It would be difficult to read most modern papers on this topic without specifically understanding their references. Here's a minimal list of literature that will be essential for this project.

- (Andrew C. Yao. 1982.) Protocols for secure computations.
- (O. Goldreich, S. Micali, and A. Wigderson. 1987.) How to play ANY mental game.
- (Ivan Damgard and Jesper Buus Nielsen. 2007.) Scalable and unconditionally secure multiparty computation.
- (Vipul Goyal and Yifan Song. 2020.) Malicious Security Comes Free in Honest-Majority MPC.
- (Vipul Goyal, Yifan Song, and Chenzhi Zhu. 2020.) Guaranteed Output Delivery Comes Free in Honest Majority MPC.

4.2 Software

It is **optional** to work with existing infrastructure provided by Hanjun Li. Otherwise, no specific software knowledge is needed, since we will be providing the software!