Popular Science Summary

Secure Multi-party Computation (SMPC) allows computations where multiple parties participate, but only the result is revealed in the end. Imagine that you are a member of an exclusive board, voting for a new member to join. Only a unanimous decision allows a prospect to join, but it can be sensitive to share who have voted no. In the analogue case there is someone who counts the votes on pieces of paper. An electronic alternative is preferable in many situations.

Upgrading from pieces of paper to pieces of data could surely have been made in the reaping process of Suzanne Collins' *The Hunger Games* as well. In choosing what child is to compete in the Hunger Games, a piece of paper is taken from a bowl with children's names, some appearing many times as the poor families can trade food for their child's name, increasing the likelihood that the child is chosen.

Both of these scenarios could have been solved electronically with the help of the two protocols suggested by Orlandi, Ravi and Scholl. The neat thing about these protocols is that they make the Bottleneck Complexity (BNC) independent on how many parties there is in the protocol. This means that even if a new party joins the protocol, the party that is already doing the most communication will not have to increase how many bits it sends or receives.

A protocol for the first usecase was implemented in the Python programming language. It makes use of a technique called Garbled Circuits (GC) that enables a party to perform computations without knowing what numbers are involved, only the result. The implementation showed that the BNC was in fact independent on the number of parties in the protocol. The second protocol, which could have been used in the Hunger Games, was not implemented because of the great complexity to create GC for this protocol. The GC would be used to perform encryption and decryption of an Additively Homomorphic Encryption (AHE) scheme. Encryption and decryption are two, usually very mathematically complex, algorithms that takes a message and turns it into a ciphertext, and then transforms it back into the original message. AHE is a very powerful technique when it comes to secure computing, allowing addition of the original numbers while keeping them secret as they are encrypted. Someone can do the computation without actually knowing the numbers, much like the GC itself.