Hi everyone, my name is Hai Nguyen. I am going to talk about Secure Computation based on Leaky Correlations: High Resilience Setting. This is a joint work with Alexander Block and Prof. Hemanta Maji.

1. Let me start with the notion of correlated private randomness, in short correlation. Correlation is a fundamental cryptographic resource that helps parties compute securely over their private data. In an offline preprocessing phase, a trusted dealer generates correlated secret shares (r\_a, r\_b) from a joint distribution, and provides r\_a to Alice and r\_b to Bob. The joint distribution here is called a correlation. During the online secure computation phase, parties use their respective secret shares in an interactive protocol to securely compute the intended functionality. Note that the preprocessing phase is independent of the functionality or the inputs fed to the functionality by the parties. However, the secret shares are vulnerable to leakage attacks. For example, Alice can leakage additional information about her secret share r\_a to Bob, and vice versa, Bob can leakage information about his secret share r\_B to Alice. In the case of leakage happen, the correlation is called “leaky correlation”.
2. A prominent and well-studied correlation is the random oblivious transfer correlation, represented as ROT. I believe that most of you are familiar with oblivious transfer. OT protocol take as input 2 bits x\_0 and x\_1 from Alice, and a bit b from Bob. It outputs x\_b to Bob. The security requirements of this protocol are that Alice does not know the choice of the bit b, and Bob does not know the other bit x\_(1-b) .

ROT is an input-less functionality that implements a randomized version of oblivious transfer functionality. It samples 3 bits x\_0, x\_1, b independently and uniformly at random, then provides (x\_0, x\_1) to Alice and provides (b, x\_b) to Bob.

1. As I mentioned earlier, secret shares are vulnerable to leakage attacks. To address this problem, Ishai, Kushilevitz, Ostrovsky, and Sahai introduced the notion of correlation extractor. Roughly speaking, a correlation extractor takes leaky correlations as input and produces secure independent copies of oblivious transfer. More concretely, a (n,m,t,epsilon) correlation extractor for a correlation (R\_A, R\_B) is a two-party interactive protocol that takes n-bit secret shares as inputs, produces m independent copies of OT. It allows t-bits of leakage on the secret share, and is secure against semi-honest adversaries with simulation error epsilon. Note that the number of the output OT samples (m) and their high security (epsilon) are crucial for the secure computation protocol.
2. Next, let me briefly summary prior work and our contributions. The first result from Ishai et al (IKOS 09) takes n/2 samples from ROT correlation. It gives asymptotically linear in production rate, linear in leakage resilience, exponential high in security, and 4 rounds protocol but the constants \alpha, \beta, \gamma are extremely small.

The subsequent work of Gupta et al (GIMS 15) improves the resilience to roughly n/4 but trade-off the security for high production rate, and consequently achieves only negligible insecurity. They also consider a new correlation, namely inner product correlation. In general, an inner product correlation over a field F is a correlation in which each party gets a random vector in F^n such that their vectors are orthogonal. They construct a correlation extractor for the inner-product correlation with resilience n/2 and exponential high security. It produces one ROT sample as output. Note that the correlation extractor takes only one sample from the big correlation.

Ideally, we want to achieve linear in production rate, linear in leakage resilience, exponential high in security, and 2 rounds protocol.

Our work shows that the inner-product correlation over an appropriately large ﬁeld admits a correlation extractor that has high concrete production rate n^{}, is resilient to n/2 bits of leakage, and has exponentially high security. I will go over the construction of the correlation extractor in the next few slides. Next, I will state our results more formally.

1. The first result we show that for every constant \delta in the range 0 to 1/2, there exists a correlation (R\_A, R\_B) and a two rounds correlation extractor with m = n^{1 – o(1)} for the correlation such that if the number of leaky bits t equals to (1/2 -g) n, then the simulation error is exponential small.

One natural question is that does there exist a correlation extractor for IP that can achieves over ½ fractional leakage resilience. More generally, can we meaningfully upper bound the maximum leakage resilience of any correlation? We have answer for the first question, and have a conjecture for the second one.

1. Our second result shows that there exists a positive universal constant \epsilon\* such that for any arbitrary field F and any (n, 1, n/2, \epsilon)-correlation extractor for the inner product correlation, the simulation error \epsilon is at least \epsilon\*.

It means that for any inner product correlation with secret shares of size n, if we leak n/2 bits from the secret shares, we cannot securely extract even only one OT.

Note that this result proves the optimality of the leakage resilience achieved by our correlation extractor and the [GIMS15] extractor.

1. Next, let me go over the construction of our correlation extractor. But before that let me introduce an important concept that needed for the construction. Instead of producing OTs, our correlation extractor produces OLEs, which is functionally equivalent to OT. An OLE over a field F, represented as OLE(F), takes two field elements (A,B) from Alice and a field element from Bob as inputs, and provides Z = AX+B to Bob.

Note that at the end of the protocol Alice gains no additional advantage in predicting X, similarly, Bob gains no additional advantage in predicting A.

Note also that OT is functionally equivalent to OLE over GF[2]

1. Now, let me go over the construction of our correlation extractor. Our goal is that Alice and Bob want to produce m copies of OLE over GF(2). The construction is the composition of two main steps. First, from a t-bit leaky inner product correlation over a large field K, we get a sample of oblivious linear-function evaluation over the same field. Then we extract m copies of OLE over GF[2] from that.

The first step is a natural generalization of the GIMS15 protocol. We consider inner product correlation over a large field instead of GF[2].

In the second step, we embed m copies of OLE over GF[2] into one OLE over the large field K. Note that this embedding relies on finding solutions of a new combinatorial problem, which I will introduce in the next slide. The larger the value of m, the better the production rate.

1. Let me state our combinatorial problem formally. We want to find.
2. Let me talk about the hardness result, upper-bounding leakage resilience. One natural question is that which techniques can be used to upper bound the maximum leakage resilience of a correlation? A common method is the partition argument. In fact this technique can be applied for the ROT correlation to prove the max. leaky resilience bound. Can we apply partition argument for inner product correlation? Unfortunately, the answer is No. That technique applies only to multiple independent samples of small correlations? What if the secret shares are sampled from globally correlated correlations, for example, the inner product correlations?
3. To answer this question, we introduce a new measure called simple partition number. Let me introduce some definitions. A simple graph is a bipartite graph such that each of its connected component is a biclique. Using the definition of simple graph, we define simple partition number as follow. The simple partition number of a bipartite graph G is the minimum number of simple graphs needed to partition its edges. For instance, this graph can be decomposed into 2 simple graphs.
4. Why do we care about these graphs? In fact, correlation can be alternatively represented as graphs. A correlation is a weighted bipartite graph, in which the left partite set L is the set of all possible secret shares for Alice, the right partite set R is the set of all possible secret shares for Bob. The weight of the edge (r\_A, r\_B) is the probability of getting (r\_A, r\_B) from the correlation. In this talk, we consider only uniform correlation, so you can ignore the weight and think about correlation as a un-weighted bipartite graph. For example, this graph represents the inner product correlation over GF(2)^2.
5. Next, I will show the connection between Max. Leakage Resilience and simple partition number. Basically, it says that an upper bound on the simple partition number implies an upper bound on the max. leakage resilience. In the next slide, I will show the estimation of simple partition number of some common correlations.
6. For the ROT^(n/2) correlation, the secret share size is n, the simple partition number is 2^{n/4}. It means that the max. fractional leakage resilience is at most ¼. Note that this technique subsumes partition argument technique. For the inner product correlation IP(F^n), the secret share is nlogF, the simple partition is F^(n/2). It implies that the max. fractional leakage resilience is at most ½.