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, we call the correlation “leaky correlation”, denoted as.
2. A prominent and well-studied correlation is the random oblivious transfer correlation, represented as ROT. Let me go over of 1-out-of-2 OT first. 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 correlation extractor. 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 ROT samples (m) and their high security (epsilon) are crucial for the secure computation protocol.
2. Prior work and our contributions. The first result from Ishai et al (IKOS 09) takes ROT correlation and 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. Note that if the field is GF[2], the secret shares are n-bit binary vectors. 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.

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 round 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 let F be an arbitrary field, there exists a positive universal constant \epsilon\* such that any (n, 1, n/2, \epsilon)-correlation extractor for the inner product correlation, the simulation error \epsilon is at least \epsilon\*.

We emphasis that even we want to extract only one OT, m = 1 in this case, the simulation error is constant.

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

1. Next, let me briefly summary the construction of our correlation extractor. But before that let me introduce an important concept that needed for the construction. Instead of producing m OTs, our correlation extractor produces m 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] since x\_b = (x\_1 – x\_0)b+ x\_0

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 combinatorial problem, which I will introduce in the next slide. The larger the value of m, the better the production rate.