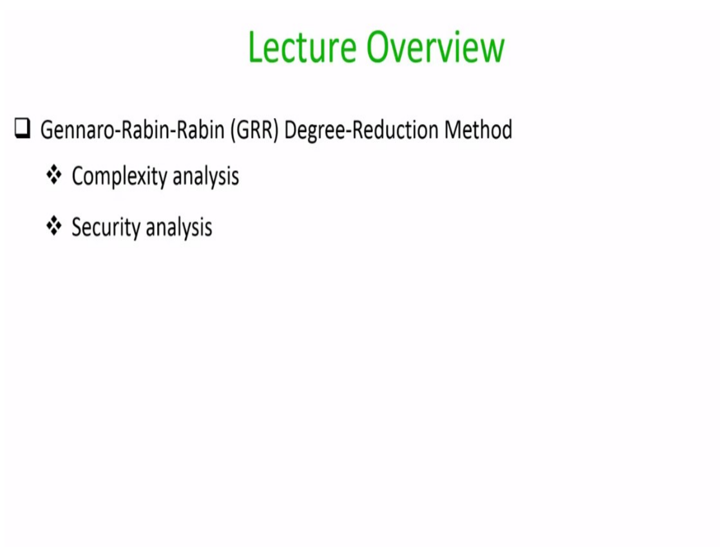
Secure Computation: Part 1
Prof. Ashish Choudhury
Department of Computer Science
Indian Institute of Science – Bengaluru

Lecture – 25
Analysis of the GRR, Degree-Reduction Method

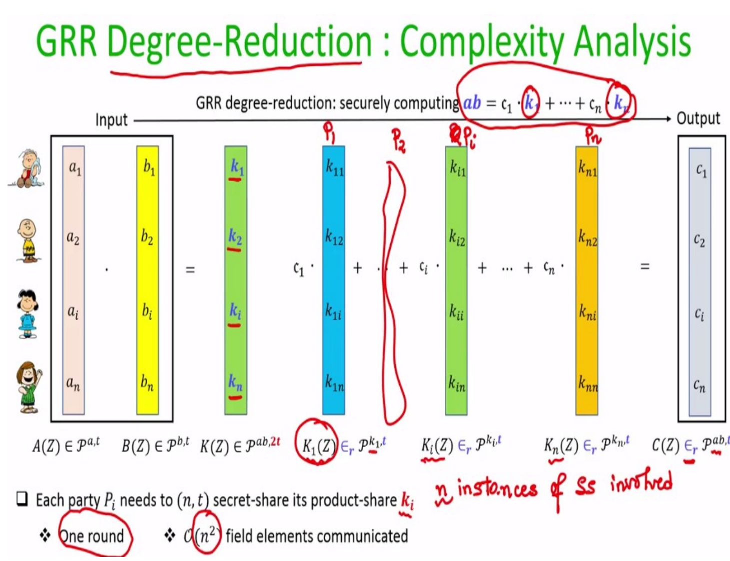
Hello everyone. Welcome to this lecture. So, in this lecture we will see the analysis of the GRR degree reduction method.

(Refer Slide Time: 00:37)



We will see the complexity analysis as well as the security analysis.

(Refer Slide Time: 00:42)



So, this is the summary of the GRR degree reduction method starting with secret sharing of a and b which are secret shared through degree t. We compute a secret sharing of a b also with degree t and the resultant shares lie on a random polynomial of degree t and basically the idea behind the GRR degree reduction method is to securely compute this linear functions. So, you can see the beauty of the method.

To do the degree reduction we actually use the fact that we have to compute a linear function of private inputs available with the respective parties and linear functions can be very easily computed if the parties just secret share their respective inputs. So, what will be the complexity here namely how many rounds will be required to do the reduction and how much communication is needed.

So, if you see here each party P i has to secret share its private input k i for this function that I have circled here. So, for instance, party 1 has to secret share k 1, party p 2 has to secret share k 2 party i has to secret share k i and party n has to secret share k n. So, P 1 will be secret sharing k 1 by picking a random polynomial of degree t and distributing shares similarly P 2 will be doing, P i will be doing and P n will be doing.

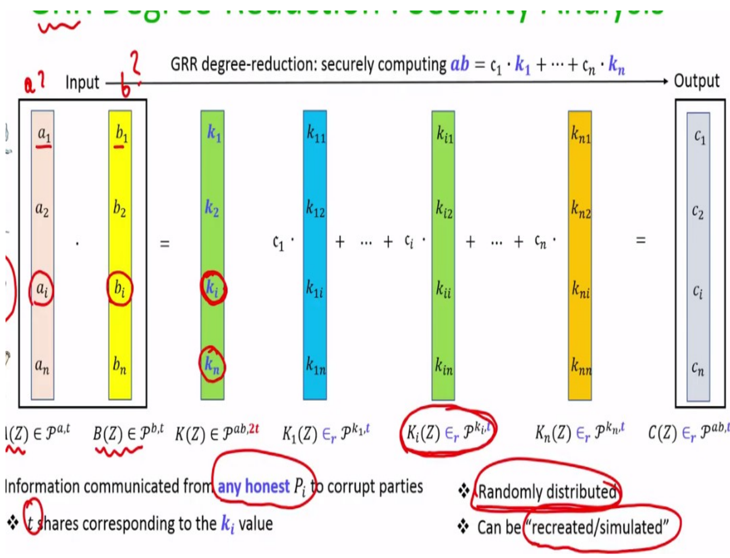
And then finally we take the linear combination. So, now even though in the slide I have shown that k 1 is picked and then k 2 is picked and then k i is secret shared and then k n is secret shared that need not be the case when we are actually implementing it because P i when P 1 is secret sharing the value k 1 through this K 1 polynomial independently and in parallel P i, P 2 any other party can independently pick it secret sharing polynomial.

And start distributing the shares of the corresponding k i values. So, that means in terms of interaction this requires only one round of communication that means if P 1 starts secret sharing its input k 1 at the same time P 2 can start secret sharing its input P i can start secret sharing its input k i, P 2 can start secret sharing its input k 2 and P n can start secret sharing its input k n.

So, that is why in terms of number of rounds this required only one communication round and there are n instances of secret sharing here involved because there are n inputs of this function each of which has to be secret shared. So, there are n instances of secret sharing involved. We know that for one instance of secret sharing order n field elements have to be communicated because n number of shares have to be communicated or distributed.

So for n instances of Shamir secret sharing total n square field elements have to be communicated that means to apply the GRR degree reduction method once the parties have to interact for one round and n square field element have to be communicated.

(Refer Slide Time: 04:21)



Now what about the security analysis? So, remember in the degree reduction problem the goal was that no additional information about a and b should be revealed. The parties should start with their components of the a vector, b vector and finally should obtain their component of the c vector. In the process no additional information about a or b or c should be revealed.

But you can see in the process the GRR degree reduction method there are several instances of secret sharing which are involved. During the secret sharing instances the parties or the adversary will be receiving shares of the k 1 value, the shares of the k 2 value, the shares of the k n value. The question is will learning those share reveal anything additional about a or b.

So, as I said if I consider the party P i to be honest imagine P i is the honest party. So, earlier as part of the input a i and b i are held only by the party P i and hence k i is held only by the party P i for the A polynomial, t shares were available with the adversary for the B polynomial t shares were available with the adversary and for this k i value adversary will now obtain t shares assuming that say the first t parties are the bad parties I do not know the control of the adversary.

But will learning those t shares help the adversary to learn or conclude anything about k i and answer is no because this k i is shared by an instance of Shamir secret sharing and as part of the instance of Shamir secret sharing the adversary will obtain t shares and those t shares are randomly distributed. They could be the shares of any candidate k i from the field. Adversary cannot pinpoint what exactly is the value of k i which has been secret shared.

In the same way if I consider the nth party to an honest party corresponding to k n the advisory will obtain t random shares and the property of Shamir secret sharing is that you take any t shares available with the adversary generated as part of the Shamir secret sharing they are randomly distributed. They could be the shares for k n being 0, they could be the shares for k n being 1 for any candidate k n from the field.

That means even though information about k 1, k 2, k i, k n are made available in the form of their shares adversary will have only t shares for each of the k i values which are under the control of the honest parties and hence those k i values remain unknown from the viewpoint of the adversary that means adversary does not learn or infer any additional information when they receive the shares of those k i values.

And hence we can easily say that they can be easily recreated or simulated by a simulator. So, remember when we formally defined the privacy property for a genetic MPC protocol the idea behind the security definition was to capture the fact that whatever information that party is received from the honest party as part of its view they can be recreated without even talking to the honest parties.

That means if we give the similar that okay this is the input and output of the corrupt guys based on this can you recreate whatever values, whatever messages the honest parties would have communicated in the real execution of the MPC protocol the simulators should be able to do that only then we can say that the interaction from the honest parties for the corrupt guys is of no use.

So, now what we are seeing here is what we are arguing here is that even in the GRR degree reduction method whatever were the shares of a and b that are under the control of adversary they will be known, but whatever information regarding the k i values that adversary is receiving as part of sharing of those k i values by the Ith party those values are randomly distributed.

And hence they can be easily simulated, they can be easily recreated even without actually talking with the Ith party that means without talking with the Ith party I mean in the sense that adversary will already know that the t shares that I am going to receive that they are random field element from the field. So, I can myself write down what are those values in the sense I can easily write down the probability distribution of those t values that I will be receiving.

That means interacting with honest P i in this GRR degree reduction method is of no advantage for the adversary. So that shows that the degree reduction method the GRR degree reduction method indeed satisfies the privacy requirement as well that means without even revealing any additional information about the a values, b values we can convert the shares of the a values and the shares of the b value into the shares of the c value lying on random t degree polynomial. Thank you.