Slido Q and A from Week 5

Arithmetisation in SNARKs does not involve execution trace, and polynomials are very very high degree, compared to STARKs, correct?

```
We do have a trace in SNARK as well.
```

How many sequencer are deployed Starknet OS on the chain?

```
The Sequencer is currently centralized, but
there are already plans to decentralized the
sequencer.
```

Is folding different than splitting? or is it the same concept?

```
different, Folding is to reduce the
verification of multiple steps into a single
```

```
step while splitting is used in inner products
arguments.
```

Is the concept of table from stark different from other protocol's table?

```
we dont have arithmetic gates in the same way
as we do in SNARKS
```

Is the distinction between Snark vs Stark very ambiguous now? i heard that snarks could use FRI or even be transparent? or that starks can use plonk

```
Indeed it is ambiguous nowadays.
```

So can you summarize again the major advantage that Cairo gives one over just using rust for zk work?

It's specifically made for provable programs and it's highlight optimized for STARK proofs.

So is the state diff changed by the fact registry or the starknet OS?

Once the verifier approves the validity of the proof, then the Starknet OS sends the state diff to L1. Note that Starknet OS is a Cairo Program.

So the goal of adding redundancy is basically to obfuscate our message (data points)?

I believe the goal is rather to enable error detection and correction.

Verification in STARKs involve proving the polynomial is of low degree, while in SNARKs,

its about evaluating polynomials at enough points to conclude they same Verification is

```
different for both STARK / SNARKS, but we will
cover this in this lecture.
```

what is k in that case?

```
k is carrying information
```

what is the purpose of Reed solomon codes?

```
We will cover this in a bit! But it enables
error detection and correction.
```

Custom gates mean less constraints?

```
Not necessarily. It's just not the standard
gate, and custom gates can reduce the circuit
size needed to compute a function.
```

How do you compute denominator in practice? That'll be very expensive, right?

```
Indeed, these would be expensive in practice, but there are some algorithms such as FFT that can help with these processes.
```

In Aztec Noir it is mentioned: "PLONK permits, at most, one trusted setup for the entire cryptosystem, " what does this means?

```
I believe it refers to the fact that you only have to perform the trusted setup once, and then it can be used for any number of different circuits within the ecosystem.
```

I remember the Cached Quotients concept from one of the podcasts in the homeworks. Are they in use at the moment?

```
I don't *think* it is part of standard PLONK, i can see a pull request , but looks like it hasn't been added yet
```

If Mina blockchain has constant size, how does it store the state? I'm talking about DA

```
The constant size relates to the amount needed to verify the blocks, the state is held separately in archive nodes, but onchain state in smart contract is fairly constrained also
```

What is the main difference between Plonk and Plonky2?

```
It is designed to be more optimized and decrease the proving time by combining PLONK and FRI, and also having recursive SNARKS.
```

Why people prefer plonkish arithmetisation over R1CS?

```
Great question, I would say simplification of the constraint systems , custom gates and having the execution trace.
```

Can it be used to publicly train on private data?

```
Yes, but currently it's not that efficient in
the ZK space.
```

Does this rely on the fact that starknet contracts are written in Cairo?

```
I would say one of the reasons, Cairo is indeed
a powerful general purpose language
```

For the "zkmic" this will prove that a certain audio has been detected but depends on a specific circuit so who will be the prover/verifier in this case?

```
You get a proof from this, which if you verify
the validity of it, you can say with certainty
that the audio file hasn't been
tampered/altered with.
```

I don't think it's correct that each "neuron" uses a linear regression activation function. Sigmoid and ReLU are more typical.

```
Thanks for pointing it out. Usually the activation functions can be found on the last layers of the NN and yes sigmoid, ReLU are more tipical
```

Is it the KZG commitment is not hiding? why?

```
It is hiding but it can be compromised if the secret value is found.
```

So for the trading bot you can avoid leaking the strategy or you have to show the strategy to the ppl that use the rockybot?

```
I believe the strategy can live offchain and you can show the validity proof in this case.
```

So the point is you can have a full bot onchain now on starknet for example?

```
Because usually we should have a offchain
software that will make the trade right?
I believe that would be the idea with the
rockybot.
```