



DOI:10.1145/3639575

Leah Hoffmann

Q&A

Verifying Correctness

Yael Tauman Kalai on her career, proof systems, and certifying correctness.

CRYPTOGRAPHER AND 2022 ACM Prize winner Yael Tauman Kalai is keenly aware of the trade-offs that often must be made between security and computational efficiency. Kalai, who works as a Senior Principal Researcher at Microsoft Research and an adjunct professor at the Massachusetts Institute of Technology (MIT), has developed groundbreaking methods for succinctly verifying the correctness of a computation. Here, she explains how they work.

Your work on proof systems dates back to graduate school, where you began studying the security of the Fiat-Shamir paradigm.

Fiat-Shamir is a paradigm that reduces interaction in interactive protocols. Adi Shamir was my master's thesis advisor at the Weizmann Institute. I thought the paradigm was beautiful, and I was working really hard to try to prove that it was sound.

Can you give a quick overview of how interaction and Fiat-Shamir work?

Interaction is a very powerful tool that we use for proofs. If I want to prove something, I can give you a classical, mathematical proof. That tends to be very long, and it's hard to verify. But allowing interaction and randomness reduces the number of bits you need to exchange, and it makes verifying much easier. But in practice, in many scenarios, we can't use this interactive setting since an interactive proof convinces a single person—the one the prover is interacting with. If you want to prove something to the



world, you'd need to interact with each and every person.

And that's where Fiat-Shamir comes in.

The Fiat-Shamir paradigm is a very simple, elegant way to eliminate interaction from interactive protocols. It was introduced in the 1980s, and it's used all over the place. But the question is, if you apply Fiat-Shamir to an interactive protocol, do you get something that's sound?

Yet in the course of trying to prove soundness, you ended up finding an example for which it's insecure.

I worked really hard, and after failing for a long time, I told my dad—who is also an academic, but knows nothing about cryptography—that I was

stuck, and he said, “Maybe it's the time to look for a counterexample.”

So, fast forward a little bit, we did get a counterexample. Together with my Ph.D. adviser at the time, Shafi Goldwasser, we showed you can't prove in general that this paradigm is sound. Which is not to say that the use of Fiat-Shamir in practice is not sound; it's just that if you want to come up with a proof of security, you need to limit the scope.

This is what drew you into the realm of proof systems and interactive proofs, which have enormous relevance to distributed platforms like blockchain technologies.

There was a lot of work in the 1990s showing that interactive protocols enable you to [CONTINUED ON P. 107]

PHOTO BY DANIEL JACKSON/MIT CSAIL

[CONTINUED FROM P. 108] verify very powerful computations. But the prover needed a huge amount of resources to convince a verifier that the computation was correct.

At the time, nobody cared about prover runtime, because this was all theoretical. They even called the prover “Merlin,” like a wizard. Of course, in reality, we don’t have wizards, and even very powerful machines are bounded. So Shafi Goldwasser, a fellow student at the time Guy Rothblum, and I embarked on this journey of what we call “doubly efficient” interactive proofs. The goal is to make the verifier efficient without putting too much overhead on the prover.

How did your work unfold from there?

The holy grail would be to say: for any computation that takes time T in Space S , we have an interactive proof where the prover runs in time close to T , and the verifier runs in space close to S . That’s still an open problem. But we did construct an interactive proof for any computation that requires small depth—meaning one that’s very efficiently parallelizable—the verifier runs in time linear in the input and the depth of the computation, and the prover’s overhead is very small. This is known as the GKR protocol, after its inventors.

Subsequently, you focused on developing not just proofs but succinct certificates that would certify correctness of a computation.

There are two approaches to these certificates. One is to take the doubly efficient interactive proofs and reduce interaction via the Fiat-Shamir paradigm. Indeed, recently, together with Jawale, Khurana, and Zhang, we were able to prove that applying the Fiat-Shamir paradigm to the GKR protocol is sound under standard assumptions.

The other approach starts with two provers.

This line of work originated with a model that was introduced in the late 1980s. Suppose you have two provers that don’t interact with each other. Let’s say I’m a verifier, and those two provers give me the outcome of a very difficult computation. I say, “How do I know that their outcome is the correct one?” And they say, “We’ll prove it to you.”

“The way these two-prover systems work is very simple. The verifier sends a question to each prover, and each prover sends an answer.”

I send Prover One to one room and Prover Two to another room. They can’t talk to each other, so I can interrogate each prover separately and verify the correctness of the computation. As it turns out, it is very hard to cheat in this model, which makes verifying very efficient. If the computation takes time T , I can verify it in time almost $\log T$.

From there, you developed an approach that uses cryptography, essentially, to reduce these two provers to one.

Let me just say the way these two-prover systems work is very simple. The verifier sends a question to each prover, and each prover sends an answer.

There is a beautiful heuristic for converting this two-prover system to a single prover using fully homomorphic encryption. Let’s say I only have one prover. I’ll give that prover both questions, encrypted. Turns out, we have encryptions that allow the prover to generate an encryption of the answers without actually knowing the questions. It’s kind of magical. And then the verifier can decrypt the answers and see if they match.

This approach does not generate a certificate, because you need the secret keys to verify. But you can think of it as a designated verifier certificate, because the verifier can say, “I’m a verifier. Here are my two encrypted questions. Anyone who wants to prove things to me, just send me the encrypted answers.”

Is it secure?

Together with Ran Raz and Ron Rothblum, we tried to analyze it for a


very long time. It seemed like it should be secure, because when the prover gives you the answer to Question One, Question Two is completely unknown. It’s encrypted with a different key. How can he cheat?

Turns out, this model is not necessarily secure, which surprised us, because it seemed counterintuitive. And the reason is, in the two-prover model, the answers are local, meaning that Answer One is only a function of Question One, and Answer Two is only a function of Question Two. Now we give the prover both questions encrypted. Encryption guarantees that when you return Answer One, it does not signal any information about the other question. But it’s not necessarily local.

As it turns out, there is a mind-blowing connection here with quantum physics, even though we are completely in the classical world. In quantum physics, there’s this notion of non-signaling strategies. It has to do with quantum entanglement, which I don’t want to go into. But if the two provers share some quantum entanglement, then you can’t argue that they’re completely local. Einstein called it “spooky interaction.”

However, we found that if we start with a two-prover interactive proof that is sound, even if the two provers interact, as long as each answer does not signal information about the other question, then this transformation is sound. We also constructed a two-prover interactive proof with such non-signaling soundness for any (deterministic) computation, which can then be used to generate a designated verifier succinct certificate.

So it was an accidental adventure in quantum.

Absolutely. It was really not the question I was interested in, but it was a technique that I stumbled upon to get this designated verifier scheme to work. And now we use this technique—together with a very nice recent paper by Choudhuri, Jain, and Jin—to make what we call a SNARG, which stands for Succinct Non-interactive ARGument and is a succinct certificate that’s publicly verifiable. 

Leah Hoffmann is a technology writer based in Piermont, NY, USA.

© 2024 ACM 0001-0782/24/3