Post-quantum zero-knowledge with depth-efficient simulation

Gaspard Damoiseau-Malraux, Alex B. Grilo, Damien Vergnaud

Sorbonne Université, CNRS, LIP6

Zero-knowledge proofs

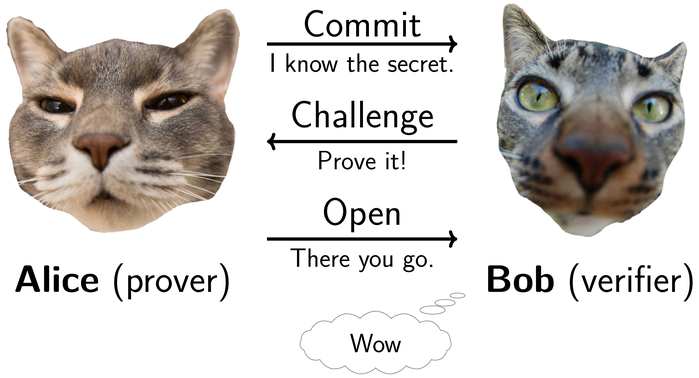
In cryptography, **zero-knowledge proofs** [3] are interactive protocols between a prover $\mathcal{P}$ and a verifier $\mathcal{V}$. They are essential primitives, allowing $\mathcal{P}$ to prove a mathematical statement to $\mathcal{V}$, **without revealing anything else than the fact that this statement is true.**

- *What the verifier sees in the protocol (even if it cheats) should be something which it could have computed itself.* [2]

Textbook examples of zero-knowledge proofs (ZKPs) are the graph 3-coloring proof [1], the quadratic nonresidue proof, or the graph isomorphism problem [3].

More generally, **any language $\mathcal{L} \in NP$ admits a ZKP** [2] — even if $\mathcal{L} \in P$, although these are trivial and not really useful in this case.

Diagram of a Σ-ZKP in action



Bob **learned nothing**, except that Alice knows the secret.

How to recognize a zero-kowledge proof?

A zero-knowledge protocol must respect 3 fundamental properties. Informally, they are as follows:

- **Completeness:** the protocol should work: if Alice knows the secret, then Bob must accept.
- **Soundness:** the protocol must be secure: Bob should refuse all cheating provers (except with negligible probability).
- **Zero-knowledge:** the protocol should not reveal anything else than the fact that Alice knows the secret.

Simulators

To prove that a ZKP is *zero-knowledge*, we must exhibit a simulator S. A simulator is a **polytime algorithm, which output is indistinguishable from the transcript of a real execution of the protocol.**

→ The intuition is that the simulator, devoid of the secret, evidently cannot leak any secret information. If its output is indistinguishable from a true transcript, then it must be that the protocol does not leak any information either.

S can perform black-box queries[a] to the verifier in order to be able to perform its task. In addition, it can also make parallel operations, even possible parallel queries to $\mathcal{V}$.

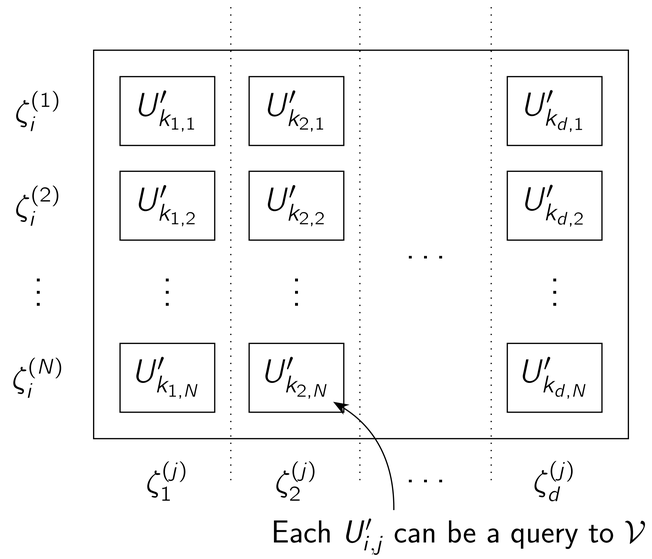
[a] Meaning that it can query $\mathcal{V}$ with no restriction on the context or the order of the queries.

Post-quantum zero-knowledge

Cryptology is at a turning point due to the advent of the quantum threat. In post-quantum zero-knowlege (PQZK), the adversary (namely, the verifier) is a quantum algorithm. Therefore, the simulator must also be quantum.

In this situation, is it difficult to find *efficient* simulators. An *efficient* simulator is defined as one with the same complexity as $\mathcal{V}$. Efforts are made in the community to improve the complexity of simulators.

Diagram of a parallel simulator



Research goal

One measure of quantum time complexity is circuit depth. The depth of a circuit corresponds to the number of "slices" of a quantum circuit one can make, or seen differently, the amount of "steps" that some qubits go through before measurement (even though these steps can have parallel operations). In particular, we are interested in simulators which are *depth-efficient*, i.e. which **depth is constant in that of the verifier**. The open question we are working on is the following:

- **What kinds of languages admit zero-knowledge protocols, where there exist depth-efficient simulators?**

The goal of our research is to prove that **only languages in BQP[a] admit simulators whose depth remains constant** — that is, there exist no constant-depth simulators for nontrivial languages.

- **Conjecture:** *Let $\mathbb{L}$ be any language that admits a black-box ZKP with a constant-depth simulator. Then $\mathbb{L} \in$ **BQP**.*

[a] **BQP** (*Bounded-error Quantum Polynomial time*) is a complexity class that contains probabilistic quantum algorithms with a polynomial execution time, a quantum equivalent to complexity class **P**.

Why is it important?

As mentioned earlier, a protocol in P or BQP would be useless; hence, proving that the existence of a constant-depth verifier implies membership to **BQP** would help characterize and differentiate which protocols are possible, and which ones are useful.

In addition, it would also help to know what can be expected from existing quantum devices. One main difficulty is indeed the lifespan of qubits, because it is currently hard for experimentalists to keep qubits coherent for more than a few milliseconds. **The impossibility of a constant-depth circuit would therefore hinder near future implementations.**

References

[1] O. Goldreich and A. Kahan.
How to construct constant-round zero-knowledge proof systems for np.
Journal of Cryptology, 9(3):167–189, June 1996.

[2] O. Goldreich, S. Micali, and A. Wigderson.
Proofs that yield nothing but their validity or all languages in np have zero-knowledge proof systems.
J. ACM, 38(3):690–728, July 1991.

[3] S. Goldwasser, S. Micali, and C. Rackoff.
The knowledge complexity of interactive proof-systems.
In Proceedings of the seventeenth annual ACM symposium on Theory of computing, STOC '85, page 291–304, New York, NY, USA, Dec. 1985. ACM.