Zero-Knowledge and Interactive Proofs

Joshua Koo

July 2023

Table of Contents

NP

Informal Definition

NP is the set of problems that can solved like the following:

- **Completeness**: If a statement is true, there is a proof that the prover can send to the verifier such that the verifier becomes convinced that the statement is true.
- **Soundness**: If a statement is not true, there is no proof that the prover can send to convince the verifier that the statement is true.

Example: Hamiltonian Cycles are NP

Definition

In a graph, a **Hamiltonian cycle** is a cycle that visits each vertex exactly once.

Why are these in NP? If there is in fact a Hamiltonian cycle in a given graph, then the prover can propose to the verifier a Hamiltonian cycle and the verifier can check whether the cycle is valid by simulating the cycle. If there no Hamiltonian cycle, there is no way for the prover to provide the verifier a working cycle.

Interactive Proofs

Definition

A problem admits an **interactive proof** if it satisfies the following three conditions:

- **Efficiency**: The number of interactions between the prover and verifier is polynomial and the verifier runs in randomized polynomial time.
- **Completeness**: If a statement is true, there is a sequence of interactions between the prover and verifier such that the verifier becomes convinced that the statement is true with high probability.
- **Soundness**: If a statement is not true, with a high probability over a sequence of interactions then the verifier cannot be convinced.

Example

Definition

Two graphs G and H are **isomorphic** if the nodes of G can be reordered so that it is identical to H.

Here, we will look at the complement of isomorphism. We will show that if two graphs aren't isomorphic, a prover can convince a verifier of this fact through an interactive proof system.

Protocol

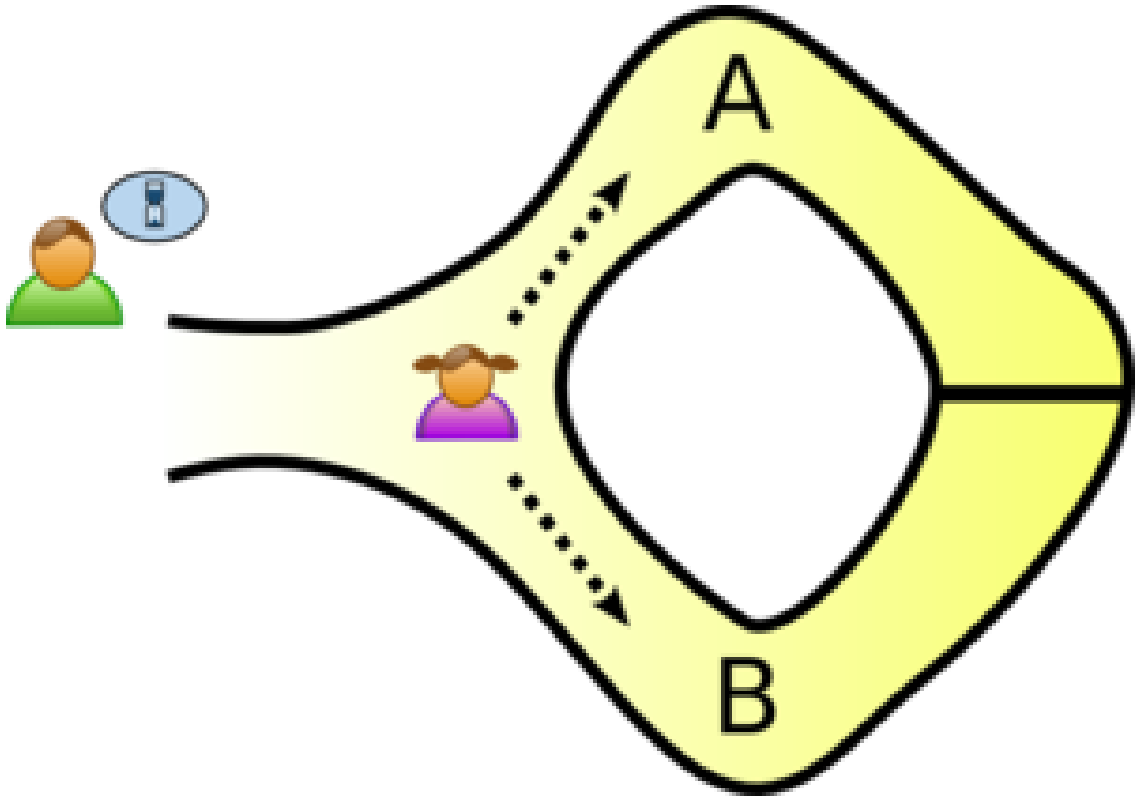
Let the two graphs we consider be G_1 and G_2. The verifier can then randomly select one of the graphs and randomly reorder the nodes of them. Let us call this new graph H. The verifier will then send H to the prover and the prover must decide whether H was obtained from G_1 or G_2. If G_1 and G_2 were indeed nonisomorphic, then the prover should always be able to be correct.

Zero-Knowledge Proof

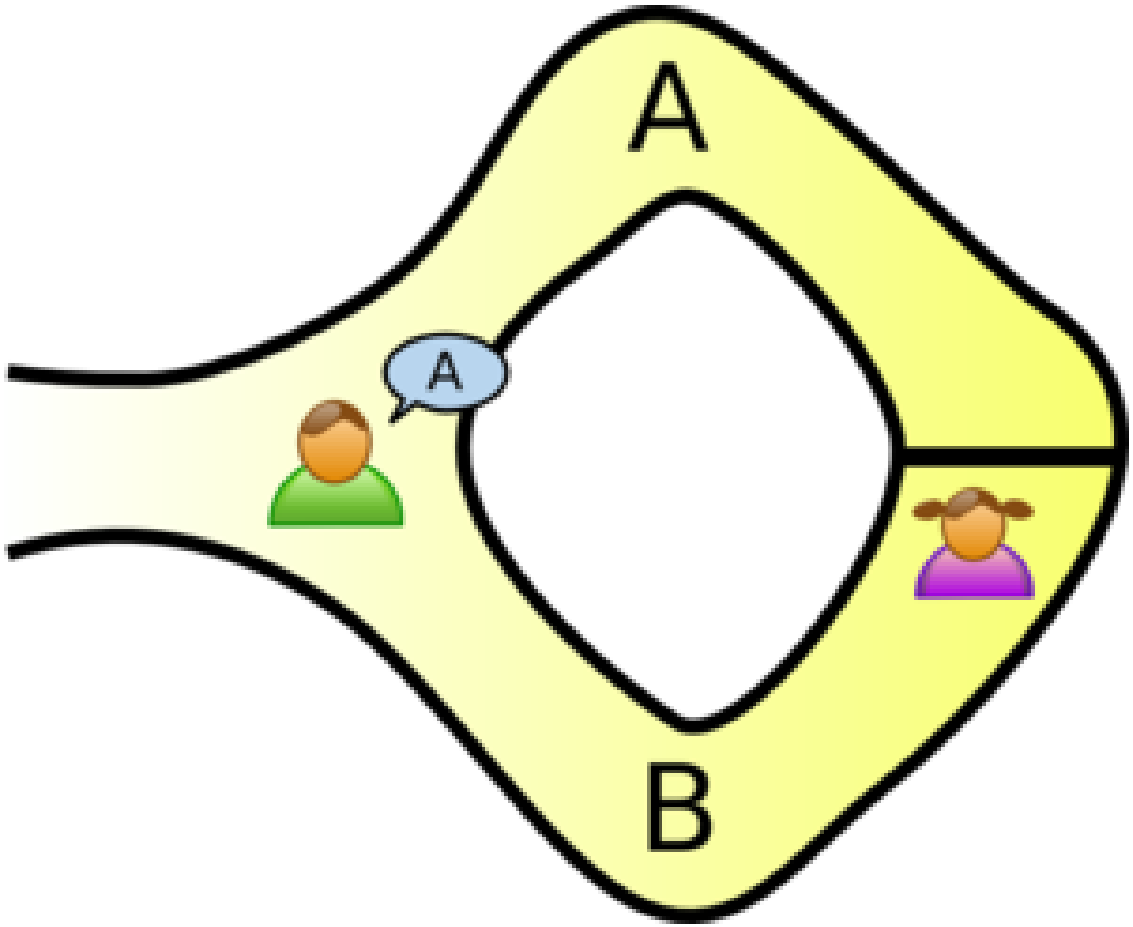
Definition

- **Completeness**: If a statement is true, there is a sequence of interactions between the prover and verifier such that the verifier becomes convinced that the statement is true with high probability.
- **Soundness**: If a statement is not true, with a high probability over a sequence of interactions then the verifier cannot be convinced.
- **Zero-Knowledge**: If a statement is true, the verifier learns nothing other than the fact that the statement is true.

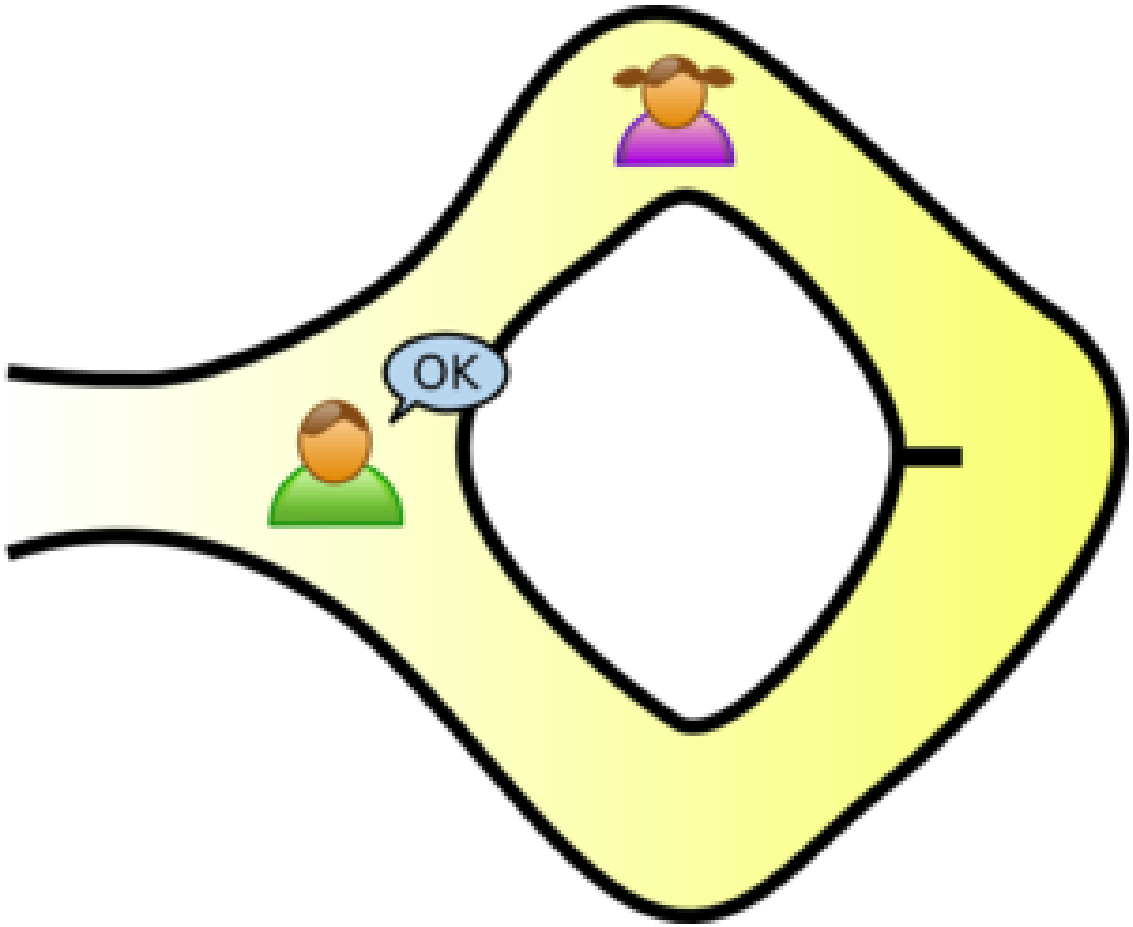
Abstract Example



Abstract Example



Abstract Example



Example: Hamiltonian Cycles!

- Peggy creates a graph H isomorphic to G.
- On a piece of paper, for each edge of H, Peggy writes down the two vertices that the edge joins (after the vertices are all labeled). The pieces of paper are then put face down.
- Victor then randomly asks Peggy to either show the isomorphism or show a Hamiltonian cycle in H.
- If Peggy is asked to show that H and G are isomorphic, she flips all the papers and then provides the vertex translations that map G to H.
- If Peggy is asked to prove that she knows a Hamiltonian cycle in H, she translates her Hamiltonian cycle from G onto H and only reveals the edges of the Hamiltonian cycle.

Checking the Requirements and Applications

- Completeness

Blockchain (Anonymous payments, Identity protection, authentication)

Checking the Requirements and Applications

- Completeness
- Soundness

Blockchain (Anonymous payments, Identity protection, authentication)

Checking the Requirements and Applications

- Completeness
- Soundness
- Zero-Knowledge

Blockchain (Anonymous payments, Identity protection, authentication)