Notes on Complexity Theory Last updated: May, 2020

Quantum Zero Knowledge

Jonathan Katz

The language of *graph isomorphism* (GI) consists of pairs of graphs (G_0, G_1) such that G_0 and G_1 are isomorphic. GI was one of the first languages shown to have a classical zero-knowledge (ZK) proof, i.e., a proof that is zero-knowledge against a polynomial-time classical verifier. Here we describe Watrous's result [2] that GI has a *quantum* ZK proof, i.e., a proof that is zero-knowledge against a polynomial-time *quantum* verifier. We assume the reader has some basic background in quantum computing, but otherwise try to make the presentation as elementary as possible.

1 The Classical ZK Proof for GI

We begin by describing the classical ZK proof for GI [1]. Let us first recall the relevant definitions. For a protocol specified by interactive algorithms $\mathcal{P}, \mathcal{V}$, we let $\langle \mathcal{P}(w), \mathcal{V}(z)\rangle(x)$ denote $\mathcal{V}$'s output after running an execution of the protocol where $\mathcal{P}$ (resp., $\mathcal{V}$) uses private input w (resp., z), and the parties have common input x. An interactive proof for an $\mathcal{NP}$ language L with $\mathcal{NP}$ relation R_L consists of a pair of probabilistic polynomial-time (PPT) interactive algorithms $\mathcal{P}, \mathcal{V}$ (the prover and verifier, respectively) such that:

Completeness: if $(x, w) \in R_L$, then $\langle \mathcal{P}(w), \mathcal{V}\rangle(x) = 1$. That is, the honest prover who knows a witness w can always make $\mathcal{V}$ accept on common input $x \in L$.

Soundness: if $x \notin L$, then for all (even all-powerful) $\mathcal{P}^*$ we have

$$\Pr[\langle \mathcal{P}^*, \mathcal{V}\rangle(x) = 1] \leq 1/2.$$

That is, if $x \notin L$ then no cheating prover can convince $\mathcal{V}$ to accept with probability better than $1/2$.

Of course, the soundness error can be made exponentially small using sequential repetition.

Zero knowledge. An interactive protocol is *zero knowledge* if a cheating verifier "learns nothing" from an execution of the protocol. This is formalized by requiring the existence of an efficient simulator who can generate transcripts (without any interaction with $\mathcal{P}$) that have the same distribution as in a real execution of the protocol, even if the verifier is malicious. Formally, we say that $\mathcal{P}, \mathcal{V}$ is *statistical zero-knowledge for a classical verifier* if for any PPT verifier $\mathcal{V}^*$ there is a PPT simulator Sim such that for all $(x, w) \in R_L$ and all z the distributions $\langle \mathcal{P}(w), \mathcal{V}^*(z)\rangle(x)$ and $\mathsf{Sim}(x, z)$ are within statistical difference $2^{-|x|}$. Note that, without loss of generality, we may assume that $\mathcal{V}^*$ is deterministic, and simply outputs its entire view at the end of the execution.

A ZK protocol for GI. Let $x = (G_0, G_1)$ be a pair of isomorphic graphs, and let $w = \sigma$ be a permutation of the vertices with $\sigma(G_1) = G_0$. The 3-round ZK protocol for GI is defined as:

1. $\mathcal{P}$ chooses a uniform permutation π, and sends $H = \pi(G_0)$.
2. $\mathcal{V}$ responds with a uniform $b \in \{0, 1\}$.

3. $\mathcal{P}$ sends $\pi' = \pi \circ \sigma^b$.

4. $\mathcal{V}$ outputs 1 iff $\pi'(G_b) = H$.

Since $G_0 = \sigma^b(G_b)$, we have $\pi'(G_b) = \pi(\sigma^b(G_b)) = \pi(G_0) = H$, and so completeness holds. Moreover, if for some H a cheating prover can send correct responses π_0, π_1 for each possible challenge $b = 0, 1$, then $\pi_0(G_0) = \pi_1(G_1)$ and so G_0 and G_1 are isomorphic. This proves soundness (since if $x \notin L$ then a cheating $\mathcal{P}^*$ will be able to respond correctly to at most one challenge).

We now prove zero knowledge. Fix some PPT verifier $\mathcal{V}^*$. For notational convenience, let $\mathcal{V}^*_{x,z}(H)$ represent the 1-bit challenge sent by $\mathcal{V}^*$, when run using inputs x, z, and receiving first message H from the prover. (We assume $\mathcal{V}^*_{x,z}$ always outputs a bit, treating an abort as a 0.) The simulator Sim, on input x, z, does as follows:

- Let $n = |x|$. For $i = 1, \ldots, n$ do:
  1. Choose uniform $a \in \{0, 1\}$ and uniform permutation π. Set $H = \pi(G_a)$.
  2. Let $\mathcal{V}^*_{x,z}(H) = b$. If $b = a$ then output (H, π) and stop. Otherwise return to step 1.
- If all n iterations of the loop failed, output $\perp$.

The idea of running $\mathcal{V}^*$ repeatedly is often called "rewinding." (I.e., we may say that Sim runs $\mathcal{V}^*_{x,z}(H)$ to see whether the result is a, and if not then it rewinds $\mathcal{V}^*_{x,z}$ to its initial state and then runs another iteration.) But it is important to realize that there is nothing magical going on here; all Sim is doing is running the (fixed, deterministic) algorithm $\mathcal{V}^*_{x,z}$ repeatedly, on different inputs.

Claim 1 *For all $x \in L$ and any z, the probability that Sim outputs $\perp$ is $2^{-|x|}$.*

Proof This follows since H is independent of a (we leave this to the reader to verify); thus, regardless of what $\mathcal{V}^*$ does, the probability that $b = a$ in any iteration is exactly 1/2. ■

Claim 2 *Fix $(x, \sigma) \in R_L$ and z. Conditioned on the event that $\mathsf{Sim}(x, z)$ does not output $\perp$, its output is distributed identically to that of $\langle \mathcal{P}(\sigma), \mathcal{V}^*(z) \rangle(x)$.*

Proof In each iteration of Sim, if we condition on $b = a$ then the output (H, π) from that iteration is distributed identically to that of $\langle \mathcal{P}(\sigma), \mathcal{V}^*(z) \rangle(x)$. (We leave verification of this to the reader.) Since the iterations are independent of each other, this proves the claim. ■

These two claims show that the interactive proof above is statistical zero knowledge. (Note if we allow Sim to run in *expected* polynomial time, we can obtain a *perfect* simulation. It is not known how to obtain perfect ZK [with perfect completeness] with a strict polynomial-time simulator.)

2 Proving Quantum Zero Knowledge

Our goal is to show that the previous protocol is zero knowledge even against a *quantum* polynomial-time (QPT) verifier. We will continue to use the same protocol as above. So, in particular, the honest prover $\mathcal{P}$ is still classical, as is the witness σ used by $\mathcal{P}$ and the common input x. The cheating verifier $\mathcal{V}^*$, however, is now allowed to be a QPT algorithm, and the verifier's auxiliary input z is now a quantum state $|z\rangle$. (We assume the auxiliary input is a pure state for simplicity. One can verify that the proof does not change significantly if the auxiliary input is a mixed state.) Fixing some common input x, we may now view the interaction of $\mathcal{V}^*$ with the honest prover as follows:

1. $\mathcal{P}$ chooses a uniform permutation π, and sends $H = \pi(G_0)$.

2. Let $\mathcal{V}^*_H$ denote the quantum operator the verifier uses when receiving H as the first message. The verifier applies $\mathcal{V}^*_H$ to $|z\rangle\,|0\rangle$ to obtain

$$\mathcal{V}^*_H|z\rangle\,|0\rangle \;= \sum_{z',b} \alpha^H_{z',b}\,|z'\rangle\,|b\rangle\,.$$

   The verifier measures the final register to obtain a bit b, and sends b to $\mathcal{P}$. Denote the residual state after the measurement by $|\psi\rangle$.

3. $\mathcal{P}$ sends $\pi' = \pi \circ \sigma^b$.

4. The "view" of $\mathcal{V}^*$ is H, π', and the residual state $|\psi\rangle$.

Let $\mathcal{V}^*$ be an operator that takes into account the verifier's dependence on H; i.e.,

$$\mathcal{V}^*|H\rangle\,|z\rangle\,|0\rangle \;= \sum_{z',b} \alpha^H_{z',b}|H\rangle\,|z'\rangle\,|b\rangle\,.$$

We can encapsulate the distribution of the verifier's output in a single quantum state. To do this, take the superposition $(N!)^{-1/2}\sum_\pi |\pi(G_0)\rangle\,|\pi\rangle$ over $\mathcal{P}$'s choices in step 1; then apply $\mathcal{V}^*$; then apply step 3 of the honest prover algorithm. (No measurement is done.) This gives the state

$$|\mathsf{real}\rangle\,{}^{\mathcal{V}^*}_{\sigma,|z\rangle} \stackrel{\text{def}}{=} \frac{1}{\sqrt{N!}} \cdot \sum_\pi \sum_{z',b} \alpha^{\pi(G_0)}_{z',b}\,|\pi(G_0)\rangle\,|z'\rangle\,|b\rangle\,|\pi\circ\sigma^b\rangle\,.$$

By measuring the first and third registers of the above state, we obtain an output that is distributed identically to the output of the real interaction.

It will be useful to rewrite the state above. For a graph H, let $\Pi(H,b)$ be the set of all permutations that map G_b to H. Then we have

$$|\mathsf{real}\rangle\,{}^{\mathcal{V}^*}_{\sigma,|z\rangle} \;= \frac{1}{\sqrt{N!}} \cdot \sum_H \sum_{\pi\in\Pi(H,0)} \sum_{z',b} \alpha^H_{z',b}\,|H\rangle\,|z'\rangle\,|b\rangle\,|\pi\circ\sigma^b\rangle\,.$$

Quantum zero knowledge. For quantum ZK (QZK), we allow the simulator to also run in quantum polynomial time. Although one might consider weaker definitions, for our purposes we say that $\mathcal{P},\mathcal{V}$ is *perfect zero-knowledge for a quantum verifier* if for any QPT verifier $\mathcal{V}^*$ there is a QPT simulator Sim such that for all $(x,\sigma)\in R_L$ and all $|z\rangle$ the output of $\mathsf{Sim}(x)$ when run on $|z\rangle$ is identical to the state $|\mathsf{real}\rangle\,{}^{\mathcal{V}^*}_{\sigma,|z\rangle}$ defined above.

Proving QZK. The previous proof of zero knowledge fails in the quantum setting. Informally, the reason is that we cannot "rewind" a quantum algorithm $\mathcal{V}^*$. But as we have noted before, "rewinding" simply means running an algorithm again, so this is not by itself the full source of the problem. The issue is more subtle: while we can run $\mathcal{V}^*$ multiple times, we cannot run $\mathcal{V}^*$ multiple times *on the same auxiliary input* $|z\rangle$ (at least, not if we want to also perform a measurement on the resulting state each time).

Watrous showed how to fix this using a clever idea [2, 3]. From 10,000 feet, the idea is to have the simulator do a "partial rewind," and do so at most once. In a bit more detail: the simulator

will generate a superposition over all possible first messages of the classical simulator from before. (Recall that this involves choosing both a random permutation π as well as a random guess a for the verifier's challenge.) It then runs $\mathcal{V}^*$ on the result (along with $|z\rangle$). Next, it measures in the basis that checks whether the guess a and the challenge of $\mathcal{V}^*$ are equal. If they are, the simulation is done. If not, the simulator will "partially undo" its computation to obtain some state which is *not* identical to the starting state. Nevertheless, by suitably "adjusting" this state, it can apply $\mathcal{V}^*$ once more to obtain a good simulation. We now make the description of Sim precise:

1. Let S be an operator that maps $|\mathbf{0}\rangle$ to $(2n!)^{-1/2}\sum_{\pi,a}|\pi(G_a)\rangle|a\rangle|\pi\rangle$. Compute

$$\mathsf{S}|\mathbf{0}\rangle\otimes|z\rangle|0\rangle = (2n!)^{-1/2}\sum_{\pi,a}|\pi(G_a)\rangle|a\rangle|\pi\rangle|z\rangle|0\rangle.$$

   Note that we can equivalently rewrite this as

$$(2n!)^{-1/2}\sum_{H,a}\sum_{\pi\in\Pi(H,a)}|H\rangle|a\rangle|\pi\rangle|z\rangle|0\rangle.$$

2. Apply $\mathcal{V}^*$ to the result of the previous step to obtain

$$|\psi_{\mathsf{all}}\rangle \stackrel{\text{def}}{=} (2n!)^{-1/2}\sum_{H,a,z',b}\sum_{\pi\in\Pi(H,a)}\alpha^H_{z',b}|H\rangle|a\rangle|\pi\rangle|z'\rangle|b\rangle. \tag{1}$$

3. Measure in the basis that checks if the second and last registers are equal (i.e., whether $b=a$). This returns a result along with a residual quantum state $|\psi\rangle$. There are now two cases.

   If the result is "equal," then the resulting state is

$$|\psi_{\mathsf{good}}\rangle \stackrel{\text{def}}{=} (n!)^{-1/2}\sum_{H,z',b}\sum_{\pi\in\Pi(H,b)}\alpha^H_{z',b}|H\rangle|b\rangle|\pi\rangle|z'\rangle|b\rangle.$$

   Trace out the second register of $|\psi_{\mathsf{good}}\rangle$, output the resulting state, and halt.

   If the result is "not equal," then the resulting state is

$$|\psi_{\mathsf{bad}}\rangle \stackrel{\text{def}}{=} (n!)^{-1/2}\sum_{H,z',b}\sum_{\pi\in\Pi(H,\bar{b})}\alpha^H_{z',b}|H\rangle|\bar{b}\rangle|\pi\rangle|z'\rangle|b\rangle.$$

   In this case, do:

   (a) Apply the inverse of $\mathcal{V}^*$ to $|\psi_{\mathsf{bad}}\rangle$, followed by the inverse of S.
   (b) Apply the operator F that flips the sign of every state except $|\mathbf{0}\rangle$.
   (c) Apply S followed by $\mathcal{V}^*$.
   (d) Trace out the second register, and output the resulting state.

Fix isomorphic G_0, G_1, and note that the probability the measurement returns "equal" is $1/2$. We show that in any case, the output state is identical to $|\mathsf{real}\rangle\,\mathcal{V}^*_{\sigma,|\psi_z\rangle}$ (up to reordering of the registers). This is fairly immediate if the result is "equal," using the fact that $\Pi(H,b)=\Pi(H,0)\circ\sigma^b$ for all H,b. The case when the result is "not equal" is more complex, and addressed next.

We show that applying steps 3(a)–3(c) to $|\psi_{\mathsf{bad}}\rangle$ results in $|\psi_{\mathsf{good}}\rangle$, which completes the proof. The intuition is that if the result of the measurement is "not equal" then we are in a superposition of states where the simulator guessed *wrong*. But using the magic of quantum mechanics, we can "flip" this to end up in a superposition of states where the simulator guessed right.

Let us step through the computation of the simulator when the result of the measurement was "not equal." Immediately after this measurement, the state is $|\psi_{\mathsf{bad}}\rangle$. Note that $|\psi_{\mathsf{bad}}\rangle$ and $|\psi_{\mathsf{good}}\rangle$ are orthogonal, and

$$|\psi_{\mathsf{all}}\rangle = \frac{1}{\sqrt{2}} \cdot |\psi_{\mathsf{good}}\rangle + \frac{1}{\sqrt{2}} \cdot |\psi_{\mathsf{bad}}\rangle . \tag{2}$$

Set $U \stackrel{\text{def}}{=} \mathcal{V}^*\mathsf{S}$. Consider the effect of taking the state $|\psi_{\mathsf{bad}}\rangle$, applying U^*, applying the "flip" operator F, and then applying U. (This is what the simulator does in steps 3(a)–3(c).) This gives

$$\begin{aligned} UFU^*|\psi_{\mathsf{bad}}\rangle &= UFU^* \cdot \left(\frac{1}{2} \cdot (|\psi_{\mathsf{bad}}\rangle + |\psi_{\mathsf{good}}\rangle) + \frac{1}{2} \cdot (|\psi_{\mathsf{bad}}\rangle - |\psi_{\mathsf{good}}\rangle) \right) \\ &= \frac{1}{2} \cdot (UFU^* (|\psi_{\mathsf{bad}}\rangle + |\psi_{\mathsf{good}}\rangle) + UFU^* (|\psi_{\mathsf{bad}}\rangle - |\psi_{\mathsf{good}}\rangle)) . \end{aligned} \tag{3}$$

Now,

$$U^* (|\psi_{\mathsf{bad}}\rangle + |\psi_{\mathsf{good}}\rangle) = \sqrt{2}\, U^* |\psi_{\mathsf{all}}\rangle = \sqrt{2}\, |\mathbf{0}\rangle \otimes |z\rangle\, |0\rangle ,$$

because we are just inverting the first two steps of Sim. Since F has no effect on $|\mathbf{0}\rangle$, this means

$$UFU^* (|\psi_{\mathsf{bad}}\rangle + |\psi_{\mathsf{good}}\rangle) = |\psi_{\mathsf{bad}}\rangle + |\psi_{\mathsf{good}}\rangle .$$

Moreover, since $|\psi_{\mathsf{bad}}\rangle + |\psi_{\mathsf{good}}\rangle$ and $|\psi_{\mathsf{bad}}\rangle - |\psi_{\mathsf{good}}\rangle$ are orthogonal, $U^* (|\psi_{\mathsf{bad}}\rangle - |\psi_{\mathsf{good}}\rangle)$ is orthogonal to $U^* (|\psi_{\mathsf{bad}}\rangle + |\psi_{\mathsf{good}}\rangle) = \sqrt{2}\, |\mathbf{0}\rangle \otimes |z\rangle\, |0\rangle$, and is therefore negated by F. So,

$$UFU^* (|\psi_{\mathsf{bad}}\rangle - |\psi_{\mathsf{good}}\rangle) = |\psi_{\mathsf{good}}\rangle - |\psi_{\mathsf{bad}}\rangle .$$

Plugging into Eq. (3) shows that

$$\begin{aligned} UFU^*|\psi_{\mathsf{bad}}\rangle &= \frac{1}{2} \cdot (UFU^* (|\psi_{\mathsf{bad}}\rangle + |\psi_{\mathsf{good}}\rangle) + UFU^* (|\psi_{\mathsf{bad}}\rangle - |\psi_{\mathsf{good}}\rangle)) \\ &= \frac{1}{2} \cdot \big((|\psi_{\mathsf{bad}}\rangle + |\psi_{\mathsf{good}}\rangle) + (|\psi_{\mathsf{good}}\rangle - |\psi_{\mathsf{bad}}\rangle) \big) \\ &= |\psi_{\mathsf{good}}\rangle , \end{aligned}$$

as desired.

References

[1] O. Goldreich, S. Micali, and A. Wigderson. Proofs that yield nothing but their validity, or all languages in $\mathcal{NP}$ have zero-knowledge proof systems. *J. ACM* 38(3): 691–729, 1991.

[2] J. Watrous. Zero-knowledge against quantum attacks. *SIAM J. Computing* 39(1):25–58, 2009.

[3] J. Watrous. Zero-knowledge against quantum attacks. Presentation dated Jan. 16, 2006. Available at `https://qipconference.org/2006/slides/watrous.pdf`