A *refutation* of a statement P is a proof of the statement NOT P. For example, a refutation of "formula ϕ is satisfiable" is a proof that ϕ is not satisfable. A refutation of "graphs G_0 and G_1 are isomorphic" is a proof that G_0 and G_1 are non-isomorphic.

This lecture is about the existence of efficient (polynomial-time verifiable) refutations for NP problems. It is believed that no such refutations exist for NP-complete problems like SAT. In the last lecture we say an interactive refutation for graph isomorphism (i.e. an interactive proof for non-isomorphism). It turns out that such interactive refutations for any problem can be derived systematically from interactive *proofs* for the same problem, provided the proof possesses an additional property called *statistical zero-knowledge*.

1 Efficient refutations

The class coNP consists of those problems (YES, NO) such that (NO, YES) is in NP. These are the problems that have short and efficiently checkable refutations. The following are all coNP problems:

$\overline{\text{SAT}}$: Is boolean formula ϕ unsatisfiable?
$\overline{\text{PMATCH}}$: Does graph G have no perfect matching?
$\overline{\text{GI}}$: Are graphs G_0 and G_1 not isomorphic?

Let us compare the decision problems SAT and $\overline{\text{SAT}}$. For SAT, given any boolean formula, we can always provide a short and efficiently checkable proof that the formula is satisfiable: The certificate is simply the satisfying assignment. But what if the formula is not satisfiable? Do we still expect to have a proof that this is the case?

Consider, for instance, the formula:

$$(x_1 \text{ OR } \overline{x_2}) \text{ AND } (x_1 \text{ OR } x_3 \text{ OR } \overline{x_4}) \text{ AND } (\overline{x_1} \text{ OR } \overline{x_2}) \text{ AND } (x_2).$$

This formula is not satisfiable for the following reason: The clauses $(x_1 \text{ OR } \overline{x_2})$ and $(\overline{x_1} \text{ OR } \overline{x_2})$ can only be simultaneously satisfied if x_2 is false, while the clause (x_2) requires x_2 to be true. So no matter which assignment we choose, the formula will not be satisfied.

For this specific example we did manage to give a proof that the formula is unsatisfiable. Is it possible to provide such a proof for *every* unsatisfiable formula? This is possible if the "certificate" is exponentially long; an exponentially long certificate can encode a complete record of exhaustive search for solutions, thereby certifying its failure. However, it is not known whether we can do so with polynomial-time verifiable certificates of polynomial length in the input size.

Now let's look at $\overline{\text{PMATCH}}$: Can we get a certificate that a graph does not have a perfect matching? Here the answer is yes: Tutte's theorem says that a graph has no perfect matching *if and only if* there exists a subset of vertices S such that after removing S and all its incident edges, the rest of the graph has more than $|S|$ connected components with an odd number of vertices. So the set of vertices S is a certificate that the graph has no perfect matching: This set is of size polynomial in the description, and once we have S the conclusion of Tutte's theorem can be verified in polynomial time.

There is a more brutal way to certify that a graph has no perfect matching: Run Edmonds' perfect matching algorithm on the graph. If the algorithm does not find a perfect matching, we can take this as a certificate that the perfect matching does not exist (if it did exist, the algorithm would have found it).

These example illustrate the relationship between the classes P, NP, and coNP. In general, if a (promise) problem (YES, NO) is in P then (NO, YES) is also in P and therefore in NP, so P is a subclass of coNP. On the other hand, we do not know if SAT is in coNP, giving a potential example of a problem that is in NP but not in coNP. In fact SAT (or any other NP-complete problem) is hardest to refute within NP: If SAT can be refuted efficiently then so can all of NP.

Theorem 1. *If* SAT $\in$ coNP*, then* NP = coNP*.*

Proof. We showed that there is a polynomial-time reduction from every NP-search problem to SAT. This implies that there is a polynomial-time reduction between their decision versions. So for every NP decision problem (YES, NO) there is a reduction that maps *YES* instances to satisfiable formulas and *NO* instances to unsatisfiable formulas. If SAT is in coNP then there is an polynomial-time verifier that accepts unsatisfiable formulas (with a certificate of unsatisfiability) and rejects satisfiable ones (with any "certificate"). Therefore there is a polynomial-time verifier for (NO, YES), so (YES, NO) is in coNP. □

To summarize, we know for sure that P is in the intersection of NP and coNP, but it appears plausible that NP and coNP are distinct. Does the intersection of NP and coNP contain problems other than the ones in P? One potential example is the problem

> Does the number n have an odd number of prime factors?

The reason this problem is in both NP and coNP is that there is an efficient algorithm that decides if a number is prime. To prove that n has an even or odd number of prime factors, the certificate can consist of the prime factorization of n, and the verifier can check that all the factors given are indeed prime numbers. An efficient algorithm for this problem is not known.

Another plausible example related to our discussion today is Graph isomorphism: This is a problem that has both polynomial-time proofs and polynomial-time (interactive) refutations, but no known polynomial-time algorithms.

2 Statistical Zero-Knowledge

The interactive proof for graph non-isomorphism from the last lecture has one curious property: After interacting with the prover, the verifier does not learn anything about the graphs G_0 and G_1 beyond the fact that the two are not isomorphic. Recall that the verifier chooses a random bit $b \in \{0, 1\}$, sends a random graph isomorhphic to G_b to the prover and expects to receive b as an answer. So the verifier already knows the answer he is going to get (provided the graphs are indeed isomorphic and the prover follows the rules).

Contrast this with the standard proofs for SAT where the verifier does not merely find out that the formula is satisfiable, but also learns the satisfying assignment for it. Similarly, in a proof of graph isomorphism, the verifier learns not only that G_0 and G_1 are isomorphic, but also the isomorphism ϕ between the vertices of the two graphs. Is it possible to come up with an alternative proof that hides this additional information? Here is how to do this *interactively* for Graph Isomorphism:

Interactive proof for graph isomorphism

On input (G_0, G_1):

P: Apply a random isomorphism to G_0 and send the resulting graph G to the Verifier.

V: Send a random bit $b \sim \{0, 1\}$ to the Prover.

P: Send an isomorphism π such that $\pi(G_b) = G$.

V: If $\pi(G_b) = G$, accept, otherwise reject.

This proof is clearly complete: If G_0 and G_1 are isomorphic then an isomorphism between G_b and G will exist regardless of the value of b, so the verifier accepts yes instances with probability one. On the other hand, the soundness (the probability that the verifier accepts when G_0 and G_1 are not isomorphic) is at most half: Regardless of the choice of G, G_b and G fail to be isomorphic with probability at least $1/2$ over the choice of b, in which case the verifier rejects. So this is a valid interactive proof for graph isomorphism.

Now let's see what the verifier learns when G_0 and G_1 are isomorphic (beyond the fact that they are isomorphic). The verifier observes a graph G obtained by applying a random isomorphism to G_0 (or G_1) together with an isomorphism π from G_b to G. This is "information" that the verifier could have generated on his own in the following way: First choose b and π at random and then set G to equal $\pi(G_b)$.

Proofs in which the verifier learns nothing beyond the validity of the statement to be proved are called zero-knowledge proofs. For the general definition we need the following concepts:

- The *statistical distance* between two random variables X and Y is the maximum possible distinguishing advantage of any *computationally unbounded* distinguisher.
- The *view* of interactive Turing Machine A in an interaction with B consists of A's randomness and the sequence of messages exchanged between the two.
- A function f is *negligible* if for every polynomial p and all sufficiently large n, $f(n) \leq p(n)$.

Definition 2. An interactive proof (V, P) for promise problem (YES, NO) is *statistical zero-knowledge* if there exists a randomized polynomial-time Turing Machine S called *the simulator* such that for every $x \in YES$, the statistical distance between $S(x)$ and the view of V in the interaction with P on input x is negligible in $|x|$.

The job of the simulator is to produce a view that is *indistinguishable* from an actual verifier-prover interaction efficiently but without access to the all-powerful prover. The best way to understand how this is at all possible is to describe the simulators in the examples of graph isomorphism and graph non-isomorphism.

In the proof of graph non-isomorphism, when G_0 and G_1 are not isomorphic the verifier's view consists of a random bit b, a random permutation π, the graph $\pi(G_b)$ (sent to the prover) and the bit b' equal to b (sent by the prover). The simulator outputs $(b, \pi, \pi(G_b), b)$, which in this case is identically distributed to the verifier's view (i.e., the statistical distance is zero).

In the proof of graph isomorphism, when G_0 and G_1 are isomorphic the verifier's view consists of (G, b, π) where G is a random graph isomoprhic to G_0 and b, π are random conditioned on $\pi(G_b) = G$. The simulator outputs $(\pi(G_b), b, \pi)$ which is again identically distributed to the verifier's view. So both examples satisfy our definition of statistical zero-knowledge.

The class SZK consists of all (promise) problems that have statistical zero-knowledge interactive proofs, without limitation on the number of rounds. So graph isomorphism and graph non-isomorphism are both in SZK. Just like NP and AM, SZK has a "canonical" complete problem that we describe next.

3 Statistical difference

A *sampler* is a circuit $C\colon \{0,1\}^m \to \{0,1\}^n$ that takes a uniformly random input r and outputs a sample $C(r) \in \{0,1\}^n$. The *statistical distance* between two samplers C_0 and C_1 with outputs in $\{0,1\}^n$ is the statistical distance between their output distributions. We consider the following promise problem:

SD (STATISTICAL DIFFERENCE):
Input: Two samplers C_0 and C_1.
Yes instances: The statistical distance between C_0 and C_1 is at least $2/3$.
No instances: The statistical distance between C_0 and C_1 is at most $1/3$.

We argue that SD has a statistical zero-knowledge proof. First, we show that this is the case when the quantities $2/3$ and $1/3$ are replaced by $1 - \varepsilon$ and $1/3$, where ε is some negligible function of the input length (the sizes of C_0 and C_1). We then describe a reduction that implements this change of parameters.

The proof for statistical difference is similar to the one of graph non-isomorphism:

Interactive proof for statistical difference

On input (C_0, C_1):

V: Choose random $b \sim \{0,1\}$, random $r \sim \{0,1\}^m$ and send $y = C_b(r)$ to the prover.

P: Send $b' = D(y)$, where D is the best possible distinguisher between C_1 and C_0, i.e., one that maximizes the advantage $\Pr_r[D(C_1(r)) \text{ accepts}] - \Pr_r[D(C_0(r)) \text{ accepts}]$.

V: If $b' = b$, accept, otherwise reject.

If (C_0, C_1) is a yes instance of SD then $\Pr_r[D(C_1(r)) \text{ accepts}] - \Pr_r[D(C_0(r)) \text{ accepts}] \geq 1 - \varepsilon$, so both the events "$D(C_1(r))$ rejects" and "$D(C_0(r))$ accepts" have probability at most ε. Regardless of the choice of b, the prover makes a mistake with probability at most ε. To analyze no instances we use the following characterization of statistical distance.

Lemma 3. *The statistical distance between X and Y is δ if and only if there exists a joint distribution (X, Y) and an event E of probability $1 - \delta$ such that (a) X and Y are identically distributed given E; (b) X and Y are disjoint given* NOT *E.*

When the statistical distance between X and Y is zero, X and Y are identically distributed as in case (a). When it is one they are disjoint as in case (b). The lemma says that all other distances can be represented as a "combination" of these two cases.

If (C_0, C_1) is a no instance of SD then the verifier's first message can equivalently be described like this: The verifier samples $b \sim \{0,1\}$ and (Y_0, Y_1) from the joint distribution on the outputs Y_0, Y_1 of C_0, C_1 from Lemma 3 then sends Y_b to the prover. Conditioned on E, Y_b and b are independent (since Y_0, Y_1 are conditionally identically distributed, Y_b carries no information about b) and so are b' and b. The prover then suceeds with probability exactly half. The overall accepting probability of the prover is then at most $\frac{1}{2}\Pr[E] + \Pr[\text{NOT } E] \leq \frac{1}{2} \cdot \frac{2}{3} + \frac{1}{3} = \frac{2}{3}$. Therefore the described proof accepts yes instances with probability at least $1 - \epsilon$ and no instances with probability at most $2/3$.

It remains to argue that the proof is statistical zero-knowledge. If (C_0, C_1) is a yes instance, the verifier's view consists of b, r, $C_b(r)$, and b'. The simulator outputs b, r, $C_b(r)$, and b. Since $b' = b$ with probability $1 - \varepsilon$, there is a joint distribution under which the two views are identically distributed with probability $1 - \varepsilon$. By the other direction of Lemma 3, the statistical distance between the two views is at most ε, therefore negligible in the input size.

Amplification of Statistical Difference

We now show how to enlarge the statistical distance gap between yes instances and no instances from $2/3$ versus $1/3$ to $1 - \exp(s^{-\Omega(1)})$ versus $1/3$, where s is the instance size. We will apply two different transformations given in the next two lemmas.

Lemma 4. *Given two random variables Y_0 and Y_1, let Y_0' and Y_1' consists of two independent samples of Y_a and Y_b where a and b are random bits conditioned on $a \oplus b = 0$ and $a \oplus b = 1$, respectively. Then the statistical distance between Y_0' and Y_1' is the square of the statistical distance between Y_0 and Y_1.*

Proof. Let D' be a candidate distinguisher between Y_1' and Y_0' with advantage δ'. The probability that $D'(Y_a, Y_b)$ predicts $a \oplus b$ equals

$$\begin{aligned}
\Pr[D'(Y_a, Y_b) = a \oplus b] &= \frac{1}{2}\Pr[D'(Y_1') = 1] + \frac{1}{2}\Pr[D'(Y_0') = 0] \\
&= \frac{1}{2}\Pr[D'(Y_1') = 1] + \frac{1}{2}(1 - \Pr[D'(Y_0') = 1]) \\
&= \frac{1 + \delta'}{2}.
\end{aligned}$$

Therefore it is sufficient to show that if the statistical distance between Y_0 and Y_1 is δ, the best predictor of $a \oplus b$ from Y_a, Y_b has advantage $(1+\delta^2)/2$. By Lemma 3 there exist independent events A, B of probability $1-\delta$ each such that Y_a is independent of a given A and Y_b is independent of b given B. If either of A, B happens then (Y_a, Y_b) is independent of $a \oplus b$ and the advantage of any predictor is exactly half. If neither happens then Y_a determines a and Y_b determines b so $a \oplus b$ can be predicted with conditional probability 1. The maximum prediction advantage is therefore $(1-\delta^2) \cdot \frac{1}{2} + \delta^2 \cdot 1 = (1+\delta^2)/2$ as desired. □

Lemma 5. *Let Y_0' and Y_1' consist of k independent copies of Y_0 and Y_1, respectively. If the statistical distance between Y_0 and Y_1 is δ then the statistical distance between Y_0' and Y_1' is at most $k\delta$ and at least $1 - 2\exp(-k\delta^2/2)$.*

Proof. By Lemma 3 Y_0 and Y_1 are identically distributed with probability $1-\delta$. The probability that all k samples are identically distributed is then at least $1-(1-\delta)^k = 1-k\delta$. For the other inequality, if D is the best distinguisher between Y_1 and Y_0, i.e., $\Pr[D(Y_1) \text{ accepts}] - \Pr[D(Y_0) \text{ accepts}] \geq \delta$ and $(\Pr[D(Y_1) \text{ accepts}] + \Pr[D(Y_0) \text{ accepts}])/2 = p$ then by a Chernoff bound the probability that at least kp of the copies of Y_b are accepted by D is at least $1-\exp(-k\delta^2/2)$ if $b=1$ and at most $\exp(-k\delta^2/2)$ if $b=0$. So the two distribution can be distinguished with advantage at least $1-2\exp(-k\delta^2/2)$. □

Given samplers C_0 and C_1 of size s, Lemma 4 produces samplers of size $2s+O(1)$ whose statistical distance is the square of the original one. If we apply this lemma $\log \ell$ times, we obtain samplers (C_0', C_1') of size $O(\ell s)$ whose statistical distance is at least $(2/3)^\ell$ for yes instances (C_0, C_1) and at most $(1/3)^\ell$ for no instances. Now applying Lemma 5 with $k = 3^{\ell-1}$ to C_0' and C_1' we end up with a pair of samplers of size $O(\ell 3^\ell s)$ whose statistical distance is at most $1/3$ for no instances and at least

$$1 - 2\exp(-k(2/3)^{2\ell}/2) \geq 1 - \exp(-(4/3)^\ell/6)$$

for yes instances. Choosing $\ell = \log s$ gives a polynomial-time reduction with a negligible error for yes instances as desired.

4 Completeness of Statistical Difference

The *complement* $\overline{\text{SD}}$ of SD is hard for statistical zero-knowledge, namely:

Theorem 6. *For every promise problem (YES, NO) in* SZK *there is a polynomial-time reduction that on input x outputs a pair of samplers C_0, C_1 such that*

$$\begin{aligned} \text{If } x \in YES, &\quad \text{then } C_0 \text{ and } C_1 \text{ have statistical distance at most } 1/3, \\ \text{If } x \in NO, &\quad \text{then } C_0 \text{ and } C_1 \text{ have statistical distance at least } 2/3. \end{aligned}$$

Corollary 7 (Okamoto's Theorem)**.** *If a promise problem (YES, NO) is in* SZK *then its complement (NO, YES) is also in* SZK*.*

Proof. By Theorem 6, (YES, NO) reduces to $\overline{\text{SD}}$. Therefore (NO, YES) reduces to SD. In the last section we argued that SD is in SZK. As SZK is closed under polynomial-time reductions (completeness, soundness, and zero-knowledge are all preserved), (NO, YES) is also in SZK. □

In particular, $\overline{\text{SD}}$ itself is in statistical zero-knowledge. By reversing the role of the yes and no instances we also obtain that SD is complete for SZK. Since SD has a two-round interactive proof (which happens to be zero-knowledge) it is in AM. Combining all these observations we get the complexity class containment

$$\boxed{\text{SZK} \subseteq \text{AM} \cap \text{coAM}}$$

where coAM is the class of problems (NO, YES) such that (YES, NO) is in AM. One consequence is that it is unlikely that SAT (or any NP-complete problem) has statistical zero-knowledge proofs because it would

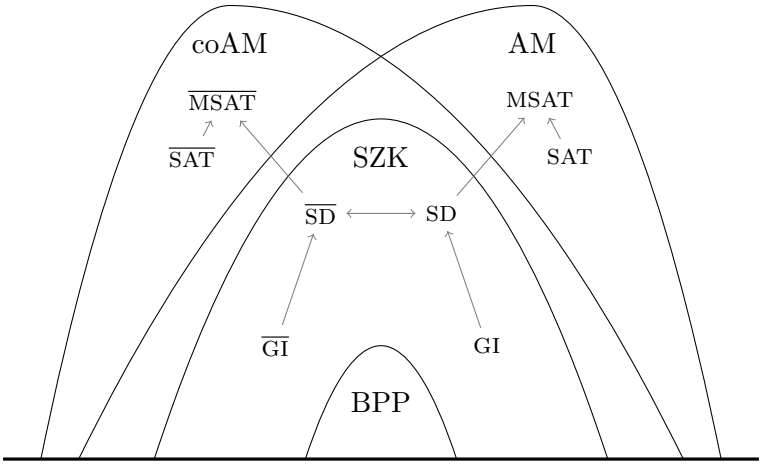

Figure 1: Complexity class containments and reductions. Under plausible derandomization assumptions BPP = P, AM = NP, and SAT and MSAT are equivalent (both NP-complete).

then have efficient refutations. Together with the results from last lecture we obtain the "complexity map" in Figure 1.

Before we sketch the proof of Theorem 6 let us explain how a restricted variant of graph non-isomorphism reduces to $\overline{\text{SD}}$. As in the example we gave in the last lecture, we will work under the promise that G_0 and G_1 have no automorphisms. We want a reduction that maps pairs of graphs (G_0, G_1) to pairs of circuits (C_0, C_1) so that if G_0 and G_1 are not isomorphic then C_0 and C_1 are statistically close, while if G_0 and G_1 are isomorphic then C_0 and C_1 should be statistically far.

Let us consider the distribution $X = \pi(G_b)$ where π is a random isomorphism and b is a random bit. If there are no automorphisms, and n is the number of vertices, then X is a flat distribution over a set of size $2n!$ when G_0 and G_1 are not isomorphic and $2n!$ when they are. If we take a sequence of six independent copies of X, the resulting distribution X^6 is also flat over support size $2^6(n!)^6$ and $(n!)^6$ for yes and no instances respectively. We now apply the following lemma to X^6:

Lemma 8. *Let Z be a random variable taking values in $\{0,1\}^n$. If Z is a flat distribution over a set of size at least 2^m and $H\colon \{0,1\}^n \to \{0,1\}^{m-2\ell}$ is a pairwise-independent hash function then the statistical distance between $(H, H(Z))$ and (H, U) is at most $2^{-\ell}$, where U is a uniform random variable independent of H.*

We apply Lemma 8 to $Z = X^6$ with $2^m = 2^6(n!)^6$ and $\ell = 2$. If G_0 and G_1 are not isomorphic by Lemma 8 the statistical distance between $(H, H(X^k))$ and (H, U) is at most $1/4$. If G_0 and G_1 are isomorphic then for every H there are at most $(n!)^6$ possible outputs for $H(X^k)$ given H and $4(n!)^6$ possible outputs for U. If D is the distinguisher that accepts the possible outputs of the form $(H, H(X^k))$ and only those, then D accepts $(H, H(X^k))$ with probability one and (H, U) with probability at most $1/4$, so the statistical distance is at least $3/4$.

Proof sketch of Theorem 6 The first step in the proof of Theorem 6 is a transformation of the statistical zero-knowledge proof for (YES, NO) into one in which the verifier uses public coins, like the one for graph isomorphism. We will not show this part of the proof and will assume that each verifier message consists of a sequence of public coins.

We will assume that the completeness and soundness gaps of the proof system are negligible. This can be arranged by repeating the proof independently several times. Assume the verifier goes first and the interaction terminates within $r(n)$ rounds on size-n inputs. Finally, we will modify the constants in the definition of $\overline{SD}$ from $1/3$ and $2/3$ to $1/3r(n)$ and negligible. These can be amplified to $1/3$ and $2/3$ using Lemma 5.

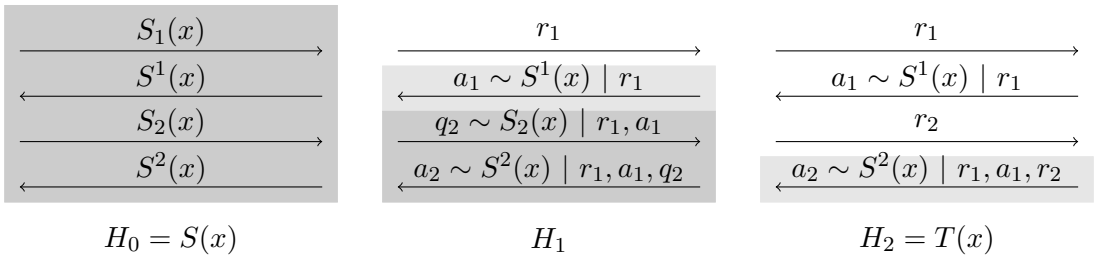

Figure 2: The distributions $S(x)$ and $T(x)$ and the hybrid H_1 when $r = 2$. The shaded part represents that output of $S(x)$ conditioned on the first $2i - 1$ messages being sampled as in $T(x)$. S_i and S^i denote the i-th question and answer output by the simulator, respectively.

On input x of (YES, NO), the output distributions of C_0 and C_1 will consists of two parts. The first part is a single bit: In C_0 first the verifier's view is sampled from $S(x)$ then the verifier's decision given this view is output. In C_1 this bit is always 1 (accept).

The second part is a partial interaction sampled independently as follows: In both C_0 and C_1 a random number I between 0 and $r(|x|) - 1$ is chosen. In C_0, the first $2I + 1$ messages $S(x)$ of the simulator are output. In C_1, the first $2I$ messages of $S(x)$ are output followed by an independent random string corresponding to the next message of the verifier. To summarize:

C_0 samples (V's output in $S(x)$, first $2I + 1$ messages of $S(x)$),
C_1 samples (1, first I rounds of $S(x)$ followed by an independent random question).

We show that if x is a yes instance then C_0 and C_1 are statistically close. First, the verifier must almost always accept the view provided by $S(x)$: The verifier accepts with probability close to 1 and the simulator's output must be indistinguishable by the zero-knowledge requirement. So the first bit is almost always 1 in both distributions. For the second part, by the zero-knowledge property for every i the first $2i + 1$ messages of the simulator are ε-close to the same messages in the actual interaction for some negligible ε. In the actual interaction, these consist of the first $2i$ messages followed by an independent random question asked by the verifier. Applying zero-knowledge again, they are therefore 2ε-close to the first $2i$ messages of the interaction followed by a random verifier message. We conclude that both parts of C_0 and C_1 are within negligible statistical distance.

Now suppose x is a no instance. Consider the following distribution $T(x)$ of verifier's views: First the verifier asks a random question r_1. Then the prover samples an answer a_1 by running $S(x)$ conditioned on the first message of $S(x)$ being equal to r_1. Then the verifier answers a random question r_2. Then the prover samples an answer a_2 by running $S(x)$ conditioned on the first three messages being equal to r_1, a_1, and r_2 respectively, and so on. We consider two cases.

If the statistical distance between $S(x)$ and $T(x)$ is at most $1/3$ then the verifier rejects the view $S(x)$ with probability at least $1/2$ because $T(x)$ represents an actual interaction between a verifier and a prover, so the probability that the verifier accepts this interaction is negligible. By statistical closeness, the probability that the verifier accepts $S(x)$ can be at most $1/2$. Then the first bit of C_0 is one with probability at most $1/2$, so the first bit distinguishes C_0 and C_1 with advantage at least half.

If, on the other hand, the statistical distance between $S(x)$ and $T(x)$ exceeds $1/3$ we claim that the statistical distance between the second part of C_0 and C_1 is at least $1/3r$, where $r = r(|x|)$. To see this consider the following hybrid distributions $H_0, \ldots, H_r$: In distribution H_i the first i rounds are sampled as in $T(x)$, but the remaining rounds are sampled from $S(x)$ conditioned on these first $2i$ messages. Equivalently, the first $2i - 1$ messages are sampled as in $T(x)$ and the remaining messages are sampled from $S(x)$ conditioned on them (see Figure 2).

Then $H_0 = S(x)$ and $H_r = T(x)$, so the statistical distance between H_0 and H_r is at least $1/3$. It follows that for a random I, the statistical distance between H_{I-1} and H_I is at least $1/3r$. This remains true if we truncate H_{I-1} and H_I after the first $2I - 1$ messages because the remaining rounds are sampled

from the same conditional distribution (and postprocessing cannot increase statistical distance). But then H_{I-1} and H_I become identical to the second part of C_0 and C_1 so the statistical distance between these two is at least $1/3r$.

References

Zero-knowledge was first defined and studied by Goldwasser, Micali, and Rackoff. The containments SZK $\in$ coAM and SZK $\in$ AM were shown separately by Fortnow and by Aiello and Håstad. The equivalence of public and private coins and the closure under complement (Corollary 7) were proved by Okamoto. The completeness of Statistical Difference and our proof of Theorem 6 follows the work of Sahai and Vadhan. Lemma 8 (the "leftover hash lemma") is due to Håstad, Impagliazzo, Levin, and Luby.