CMSC 858K — Advanced Topics in Cryptography		April 15, 2004
Lecture 21		
Lecturer: Jonathan Katz	*Scribe(s):*	*Omer Horvitz* *Zhongchao Yu* *John Trafton* *Akhil Gupta*

1 Introduction

In a previous lecture, we saw how to construct a three-round *zero-knowledge* (ZK) proof system for *graph 3-colorability* with soundness error $1 - 1/|E|$ on a common input $G = (V, E)$. The soundness error can be made negligible, while maintaining zero knowledge, by repeating the protocol $|E| \cdot \omega(\log k)$ times sequentially (where k is the security parameter); unfortunately, this increases the round complexity of the protocol tremendously and in particular does not result in a constant-round protocol. On the other hand, repeating the protocol many times in parallel is not known to result in a zero-knowledge protocol (i.e., we do not know how to show a simulator for the resulting protocol). The resulting protocol, however, can be shown to satisfy honest-verifier zero knowledge, a weaker variant of zero knowledge.

In this lecture, we consider another weakening of the zero-knowledge property known as *witness indistinguishability* [1]. This notion is useful in its own right, and also leads to constructions of constant-round ZK proof systems (with negligible soundness error) as we will see in a later lecture.

2 Witness Indistinguishability

In general, an $\mathcal{NP}$ statement may have multiple witnesses. For example, a Hamiltonian graph may have multiple Hamiltonian cycles; a 3-colorable graph may have multiple (non-isomorphic) 3-colorings; etc. We are interested in proof systems (for languages in $\mathcal{NP}$) that do not leak information about which witness the prover is using, even to a malicious verifier. In the following, we let $\langle A(y), B(z)\rangle(x)$ denote the view (i.e., inputs, internal coin tosses, incoming messages) of B when interacting with A on common input x, when A has auxiliary input y and B has auxiliary input z.

Definition 1 Let $L \in \mathcal{NP}$ and let $(\mathcal{P}, \mathcal{V})$ be an interactive proof system for L with perfect completeness. We say that $(\mathcal{P}, \mathcal{V})$ is *witness-indistinguishable* (WI) if for every PPT algorithm $\mathcal{V}^*$ and every two sequences $\{w_x^1\}_{x\in L}$ and $\{w_x^2\}_{x\in L}$ such that w_x^1 and w_x^2 are both witnesses for x, the following ensembles are computationally indistinguishable:

1. $\left\{\left\langle \mathcal{P}(w_x^1), \mathcal{V}^*(z)\right\rangle(x)\right\}_{x\in L, z\in\{0,1\}^*}$
2. $\left\{\left\langle \mathcal{P}(w_x^2), \mathcal{V}^*(z)\right\rangle(x)\right\}_{x\in L, z\in\{0,1\}^*}$.

(Note: when the security parameter is not written explicitly, we simply take $|x| = k$.) In particular, we may have $z = (w_x^1, w_x^2)$. ♢

Remark WI is defined with respect to *auxiliary-input* verifiers (where the auxiliary input can include the witnesses which the prover might use). Although we have not done so before, zero knowledge can also be defined with respect to auxiliary-input verifiers (this is called *auxiliary-input zero knowledge*). In this lecture, when referring to a ZK proof system we will mean auxiliary-input zero knowledge by default.

Witness indistinguishability is clearly a weaker notion than zero-knowledge, and in particular there exists protocols which are trivially WI and definitely *not* zero-knowledge. For example, whenever there is only a *single* witness for a particular statement (e.g., a single Hamiltonian cycle in a graph) then a protocol in which the prover sends the witness to the verifier is WI but not (in general) ZK! To flesh this example out a bit, assume the existence of a length-preserving one-way permutation f and define the language $L_0 \stackrel{\text{def}}{=} \{y \mid \text{the first bit of } f^{-1}(y) \text{ is } 0\}$. (Note that every $y \in L_0$ has the unique witness $x = f^{-1}(y)$.) An interactive proof in which the prover sends $f^{-1}(y)$ on common input y is witness indistinguishable (trivially) but not zero knowledge (assuming f is indeed one-way).

On the other hand, one can show that zero knowledge is strictly stronger than witness indistinguishability:

Theorem 1 (ZK implies WI) *If an interactive proof system $(\mathcal{P}, \mathcal{V})$ for a language L is zero-knowledge, then it is also witness indistinguishable.*

Proof (Sketch) Let $\mathcal{V}^*$ be a PPT algorithm, and let Sim be the simulator guaranteed by the zero-knowledge property of $(\mathcal{P}, \mathcal{V})$. Then for any $x \in L$ and any witnesses w_1, w_2 for x, we have that:

- $\{\mathsf{Sim}(x,z)\}_{x\in L, z\in\{0,1\}^*} \stackrel{c}{\approx} \{\langle \mathcal{P}(w_1), \mathcal{V}^*(z)\rangle(x)\}_{x\in L, z\in\{0,1\}^*}$
  and
- $\{\mathsf{Sim}(x,z)\}_{x\in L, z\in\{0,1\}^*} \stackrel{c}{\approx} \{\langle \mathcal{P}(w_2), \mathcal{V}^*(z)\rangle(x)\}_{x\in L, z\in\{0,1\}^*}$.

By the transitivity of computational indistinguishability, it follows that:

$$\{\langle \mathcal{P}(w_1), \mathcal{V}^*(z)\rangle(x)\}_{x\in L, z\in\{0,1\}^*} \stackrel{c}{\approx} \{\langle \mathcal{P}(w_2), \mathcal{V}^*(z)\rangle(x)\}_{x\in L, z\in\{0,1\}^*},$$

as required. □

Next, we show that — in contrast to zero knowledge — witness indistinguishability is preserved under parallel composition (i.e., parallel repetition of the protocol). Before doing so, we first formally define parallel composition and, in the process, establish some notation.

Definition 2 Let $(\mathcal{P}, \mathcal{V})$ be an interactive proof system for a language L, and let $\ell = \ell(k)$. We define the ℓ-fold parallel composition of $(\mathcal{P}, \mathcal{V})$, denoted $(\mathcal{P}_\ell^{\|}, \mathcal{V}_\ell^{\|})$, as the protocol obtained by running ℓ *independent* executions of $(\mathcal{P}, \mathcal{V})$ in parallel. Namely:

- $\mathcal{P}_\ell^{\|}$, on input $1^k, x, w$ with $x \in L$ and w a witness for x, generates $\ell = \ell(k)$ independent random tapes $\omega_1, \ldots, \omega_\ell$. It then runs $\mathcal{P}(1^k, x, w; \omega_1), \ldots, \mathcal{P}(1^k, x, w; \omega_\ell)$ to generate $\vec{m}_1 \stackrel{\text{def}}{=} m_1^1, \ldots, m_1^\ell$. It sends $\vec{m}_1$ to the verifier as its first message.

- $\mathcal{V}_\ell^{\|}$, on (common) input x, chooses ℓ independent random tapes $\omega_1', \ldots, \omega_\ell'$. After receiving message $\vec{m}_1 = m_1^1, \ldots, m_1^\ell$ from the prover, it runs $\mathcal{V}(1^k, x, m_1^1; \omega_1')$, $\ldots$, $\mathcal{V}(1^k, x, m_1^\ell; \omega_\ell')$ to generate $\vec{m}_2 \stackrel{\text{def}}{=} m_2^1, \ldots, m_2^\ell$. It sends $\vec{m}_2$ to the prover.
- The execution continues in this way. It is stressed that each of the ℓ executions is "oblivious" to the other $\ell - 1$ executions, and all random coins are chosen independently.
- At the end of the protocol, $\mathcal{V}_\ell^{\|}$ accepts only if all ℓ invocations of $\mathcal{V}$ have accepted.

$\diamondsuit$

It is not difficult to see that perfect completeness is preserved under parallel composition, and that if $(\mathcal{P}, \mathcal{V})$ has soundness error $\varepsilon(k)$ (against an all-powerful prover — i.e., this is a *proof* system) then $(\mathcal{P}_\ell^{\|}, \mathcal{V}_\ell^{\|})$ has soundness error $\varepsilon(k)^{\ell(k)}$.[1] We have also mentioned already that zero knowledge is *not* necessarily preserved under parallel composition. However:

Theorem 2 (WI is preserved under parallel composition) *For any polynomial $\ell(\cdot)$, if $(\mathcal{P}, \mathcal{V})$ is witness indistinguishable then so is $(\mathcal{P}_\ell^{\|}, \mathcal{V}_\ell^{\|})$.*

Proof At its core, the proof is via a hybrid argument but some things are slightly more subtle here because we are dealing with an interactive process (also, contrary to intuition(?), ZK is not preserved under parallel composition and so we must check the details and not rely on our intuition).

We prove the theorem for $\ell = 2$ (and write $\mathcal{P}^{\|}$ instead of $\mathcal{P}_2^{\|}$); the proof extends for any polynomial ℓ. Assume to the contrary that there exists a PPT algorithm $\mathcal{V}^*$, an infinite sequence $\{(x_i, w_i^1, w_i^2, z_i\}_{i \in \mathbb{N}}$, and a PPT distinguisher D such that the following is not negligible:

$$\Big| \Pr\left[\mathsf{view} \leftarrow \left\langle \mathcal{P}^{\|}(w_k^1), \mathcal{V}^*(z_k) \right\rangle (x_k) : D(1^k, \mathsf{view}) = 1 \right] - \Pr\left[\mathsf{view} \leftarrow \left\langle \mathcal{P}^{\|}(w_k^2), \mathcal{V}^*(z_k) \right\rangle (x_k) : D(1^k, \mathsf{view}) = 1 \right] \Big| .$$

Let $\hat{\mathcal{P}}^{\|}(w, w')$ denote a prover who runs two parallel executions of $\mathcal{P}$ but uses witness w in the first execution and w' in the second (we assume these are both witnesses for the same x). Using this notation, $\hat{\mathcal{P}}^{\|}(w, w) = \mathcal{P}^{\|}(w)$. Thus, a standard hybrid argument shows that at least one of

$$\Big| \Pr\left[\mathsf{view} \leftarrow \left\langle \hat{\mathcal{P}}^{\|}(w_k^1, w_k^1), \mathcal{V}^*(z_k) \right\rangle (x_k) : D(1^k, \mathsf{view}) = 1 \right] - \Pr\left[\mathsf{view} \leftarrow \left\langle \hat{\mathcal{P}}^{\|}(w_k^1, w_k^2), \mathcal{V}^*(z_k) \right\rangle (x_k) : D(1^k, \mathsf{view}) = 1 \right] \Big|$$

or

$$\Big| \Pr\left[\mathsf{view} \leftarrow \left\langle \hat{\mathcal{P}}^{\|}(w_k^1, w_k^2), \mathcal{V}^*(z_k) \right\rangle (x_k) : D(1^k, \mathsf{view}) = 1 \right] - \Pr\left[\mathsf{view} \leftarrow \left\langle \hat{\mathcal{P}}^{\|}(w_k^2, w_k^2), \mathcal{V}^*(z_k) \right\rangle (x_k) : D(1^k, \mathsf{view}) = 1 \right] \Big|$$

[1] Interestingly, this is not necessarily true when soundness is defined for computationally-bounded provers (i.e., an *argument* system, which will be defined in a later lecture).

is not negligible. Without loss of generality, assume it is the former. We show that this contradicts the witness indistinguishability of the original proof system $(\mathcal{P}, \mathcal{V})$.

Construct the following PPT verifier $\mathcal{V}^{**}$ who gets auxiliary input (z_k, w_k^1) and interacts with $\mathcal{P}$:

$\underline{\mathcal{V}^{**}(1^k, x, z_k, w_k^1)}$
run $\mathcal{V}^*(1^k, x, z_k)$ as follows:
 for the second of the two parallel executions of $\mathcal{P}$ that $\mathcal{V}^*$ expects to see
  forward the appropriate messages to and from the *external* prover $\mathcal{P}$
 for the first of the two parallel executions
  run $\mathcal{P}$ *internally*, using the witness w_k^1
when done, output the view of $\mathcal{V}^*$ (which is easy to reconstruct)

Note that

$$\left\langle \mathcal{P}(w_k^1), \mathcal{V}^{**}(z_k, w_k^1) \right\rangle (x_k) \equiv \left\langle \hat{\mathcal{P}}^{\|}(w_k^1, w_k^1), \mathcal{V}^*(z_k) \right\rangle (x_k)$$

(that is, the distributions are identical), and

$$\left\langle \mathcal{P}(w_k^2), \mathcal{V}^{**}(z_k, w_k^1) \right\rangle (x_k) \equiv \left\langle \hat{\mathcal{P}}^{\|}(w_k^1, w_k^2), \mathcal{V}^*(z_k) \right\rangle (x_k).$$

But then D distinguishes between the two distributions on the left-hand sides of the equations above, contradicting the witness indistinguishability of $(\mathcal{P}, \mathcal{V})$. ■

Recall the three-round, zero-knowledge protocol for graph 3-colorability from a previous lecture. Theorem 1 implies that this protocol is witness indistinguishable, and Theorem 2 implies that its witness indistinguishability is preserved under parallel composition. We thus obtain:

Corollary 3 *There exists a three-round, witness indistinguishable proof system with negligible soundness error for the NP-complete language graph 3-colorability, and hence for any language in NP.*

We will see applications of witness indistinguishability, both on its own and also as a tool to construct zero-knowledge protocols, in the upcoming lectures.

3 Commitment Schemes

As mentioned previously, two "flavors" of commitment schemes can be considered:

- *Perfect* (or *computationally binding*) *commitments*, protecting against all-powerful receivers and polynomial-time senders.
- *Standard* (or *computationally hiding*) *commitments*, protecting against polynomial-time receivers and all-powerful senders.

In the basic, three-round proof system for graph 3-colorability, we need soundness to hold even for all-powerful provers. Thus, we need the prover to use a standard commitment scheme to commit to the coloring of the graph in the first round.

We now show an easy construction of a standard commitment scheme: Let $(\mathsf{Gen}, \mathcal{E}, \mathcal{D})$ be a public-key encryption scheme with perfect decryption (i.e., a decryption error never occurs). Consider the following, two-phase protocol between a sender $\mathcal{S}$ on input $b \in \{0,1\}$ and a receiver $\mathcal{R}$. In the first phase, $\mathcal{S}$ runs $(pk, sk) \leftarrow \mathsf{Gen}(1^k; r_{\mathsf{Gen}})$, computes $c \leftarrow \mathcal{E}_{pk}(b; r_{\mathcal{E}})$ and sends (pk, c) to $\mathcal{R}$. In the second phase, $\mathcal{S}$ sends $(b, r_{\mathsf{Gen}}, r_{\mathcal{E}})$ to the receiver.

Claim 4 *The above protocol constitutes a standard commitment scheme.*

Proof (Sketch) Hiding follows directly from the hiding property of the encryption scheme (if the sender is honest, then r_{Gen} and $r_{\mathcal{E}}$ are chosen uniformly at random, and hence b is hidden; a formal proof is left to the reader). Perfect binding follows directly from the perfect decryption property of the encryption scheme. It is essential here that we force the sender to send all its randomness in the second round, since otherwise it may be possible for the sender to "cheat" (for example, to send an invalid public key that could not possibly be output by Gen). □

Public-key encryption schemes with perfect decryption can be based on trapdoor permutations, and so standard commitments can be based on that assumption. Specifically, for a trapdoor-permutation generator Gen, the sender commits to a bit b by computing $(f, f^{-1}) \leftarrow \mathsf{Gen}(1^k; r_{\mathsf{Gen}})$, selecting $r \leftarrow \{0,1\}^n$, computing $y = f(r)$, and sending $(f, y, h(r) \oplus b)$ to $\mathcal{R}$, where h is hard-core for f. The sender decommits by sending (r_{Gen}, r).

In the next lecture, we will see how to construct standard commitment schemes from the much weaker assumption of the existence of one-way functions.

References

[1] U. Feige and A. Shamir. Witness Indistinguishable and Witness Hiding Protocols. In *22nd ACM Symposium on Theory of Computing*, pages 416–426, 1990.