COM S 6830 – Cryptography Tuesday, October 20, 2009

Lecture 15: Interactive Proofs

Instructor: Rafael Pass *Scribe: Chin Isradisaikul*

In this lecture we discuss a new kind of proofs that involves interaction between the prover and the verifier and then extend it to zero-knowledge protocols. We then construct interactive, zero-knowledge proofs for graph nonisomorphism and graph isomorphism, provided that the verifier is honest.

1 Interactive Proofs

1.1 Definitions

Before we define an interactive proof, let us recall the traditional proof. Intuitively, we can obtain interactive proofs by relaxing the requirements posed by traditional proofs:

Traditional Proof (NP-proofs)		**Interactive Proof**
- non-interactive	RELAXATION	- interactive
- can never "prove" false statement	$\longrightarrow$	- can "prove" false statement with "small" probability

Having the intuition, we now formally define traditional proofs.

Definition 1 V is an *NP-verifier* for L if V is polynomial-time in the length of the first input and that the following two properties hold:

- (completeness) If $x \in L$, $\exists \pi : V(x, \pi) = 1$.
- (soundness) If $x \notin L$, $\forall \pi : V(x, \pi) = 0$.

In this definition, π is a certificate for x. Notice that this definition states that if $x \in L$, there is a certificate that the verifier can use to ensure that $x \in L$. Otherwise, if $x \notin L$, then there should be no such certificate.

Remark: Because V is polynomial-time in $|x|$, it is necessary that $|\pi| \leq p(|x|)$, where p is a polynomial.

Remark: This definition is equivalent to the other definition of NP (which states that there exists a nondeterministic polynomial-time algorithm that decides whether $x \in L$) because we can view the nondeterministic tape on an accepting path as a certificate. Conversely, if we have a certificate π, we can construct a nondeterministic algorithm that simply guesses π.

Now, we consider relaxing some of the requirements for an NP-verifier. If we only relax completeness and soundness (e.g., that if $x \in L$, then $\exists \pi : V(x, \pi) > \frac{2}{3}$, and if $x \notin L$, then

$\forall \pi : V(x, \pi) < \frac{1}{3}$), the resulting V will be a BPP algorithm. Therefore, we also have to relax the non-interactive requirement as well to arrive at a new kind of proof system. It seems that in doing so, we arrive at the following definition:

Definition 2 (INCORRECT) (P, V) is said to be an *interactive proof* for L if V is PPT (in the length of the input) and that the following two properties hold:

- (completeness) $\forall x \in L \exists y \in \{0, 1\}^*$:

$$\Pr[]\text{out}_V[P(x, y) \leftrightarrow V(x)] = 1 = 1$$

- (soundness) $\exists$ negligible function $\varepsilon \forall x \notin L \forall y \in \{0, 1\}^*$:

$$\Pr[]\text{out}_V[P(x, y) \leftrightarrow V(x)] = 1 \leq \varepsilon(|x|)$$

where $P(x, y) \leftrightarrow V(x)$ denotes a random variable indicating the interaction between P and V (both probabilistic) and out_V denotes a random variable indicating the output of V.

In this definition, the completeness requirement states that there is some string y that P can use to convince V with probability 1, and the soundness requirement states that no string can P use to convince V with nonnegligible probability.

For the interaction, P and V interact for a number of rounds. In the end, V outputs either 1 or 0. Also note that we did not restrict the complexity of P. That is, P can be unbounded in time. Later we will require that P be PPT so that the proof is efficient and can be used in practice.

Remark: Since ε is negligible, it might be the case that for small $|x|$, the probability that V is convinced when $x \notin L$ might be nonnegligible. To fix this, we can simply pad the instance x to be long enough that the probability of successful "cheating" becomes negligible.

Nonetheless, the definition above allows a dishonest prover to convince V that $x \in L$ even though it is not. For example, in the interactive protocol shown in Figure **??**, P that never says Hello can convince V with nonnegligible probability when $x \notin L$.

$P(x, w)$		$V(x)$
	$\xrightarrow{\text{Hello}}$	
	$\xleftarrow{\text{Hello}}$	
	$\xrightarrow{w}$	
		If P said Hello, accept if $(x, w) \in R_L$; otherwise, always accept.

Figure 1: An interactive protocol in which the prover can convince the verifier with nonnegligible probability, even though $x \notin L$

Therefore, we should redefine soundness to prevent this kind of anomaly.

Definition 3 (P, V) is said to be an *interactive proof* for L if V is PPT (in the length of the input) and that the following two properties hold:

- (completeness) $\forall x \in L \exists y \in \{0,1\}^*$:

$$\Pr[]\text{out}_V[P(x,y) \leftrightarrow V(x)] = 1 = 1$$

- (soundness) $\exists$ negligible function $\varepsilon \forall P^* \forall x \notin L \forall y \in \{0,1\}^*$:

$$\Pr[]\text{out}_V[P^*(x,y) \leftrightarrow V(x)] = 1 \leq \varepsilon(|x|)$$

where P^* is any algorithm, $P(x,y) \leftrightarrow V(x)$ denotes a random variable indicating the interaction between P and V (both probabilistic), and out_V denotes a random variable indicating the output of V.

Remark: We can restrict P^* in the above definition to be only nuPPT algorithms. In this case, the resulting definition

$$\forall \text{ nuPPT } P^* \exists \text{ negligible function } \varepsilon \forall x \notin L \forall y \in \{0,1\}^* :$$

$$\Pr[]\text{out}_V[P^*(x,y) \leftrightarrow V(x)] = 1 \leq \varepsilon(|x|)$$

is called *computational soundness*. An *interactive argument* is an interactive system such that completeness (as defined in the definition) and computational soundness hold.

For cryptography, we only require that P be PPT, which will suffice to provide proofs for problems in NP.

Before we move on to the next section, we present some results where P might be unbounded for the completeness of the topic.

1.2 Example: Graph Nonisomorphism

Recall that two graphs are isomorphic if there is a way to relabel the nodes of one so that they are identical to the other's. The Graph Isomorphism problem is in NP as we can consider a permutation of nodes as a witness. Given two graphs, how can Alice (prover) convince Bob (verifier) that the two graphs are isomorphic or nonisomorphic?

First, we consider the nonisomorphism. Consider the interactive protocol shown in Figure **??**. We claim that this protocol is an interactive proof. That is, completeness and soundness holds for this protocol.

- Completeness: If $G_1 G_2$, then there is only one b' such that $H \approx G_{b'}$. In this case, P always answers b' correctly. Hence, $b = b'$ always, and $\Pr[]\text{out}_V[P \leftrightarrow V] = 1 = 1$.

Input: $x = G_1, G_2$, where $G_1 G_2$

P		V
		Pick $b \in \{1, 2\}$ randomly. Pick a random permutation $\pi \in S_n$, where $n = \|V_b\|$. Compute a new graph $H = \pi(G_b)$.
	$\xleftarrow{H}$	
Compute b' such that $H \approx G_{b'}$		
	$\xrightarrow{b'}$	
		Accept if $b = b'$.

Repeat this interaction $|x|$ times.

Figure 2: An interactive proof for graph nonisomorphism

- Soundness: If $G_1 \approx G_2$, then $\{\pi(G_1)\} \equiv \{\pi(G_2)\}$ for random π because $G_2 = \rho(G_1)$ for some permutation ρ, so $\{\pi(G_2)\} \equiv \{\pi(\rho(G_1))\} \equiv \{\pi'(G_1)\}$ for some permutation π'. But then P cannot determine exactly whether b' is 1 or 2, so P needs to guess b' with probability $1/2$ of being correct in each round. Hence, the probability that P guesses b' correctly for all the $|x|$ rounds is $1/2^{|x|}$, which is negligible.

Note that the prover in this protocol needs not run in polynomial time. This protocol is zero-knowledge for an honest verifier because a simulator that picks random b and π, generates H, and answers $b' = b$ runs in polynomial time. It turns out that this protocol is not zero-knowledge for a dishonest verifier.

1.3 Interactive Proofs with Efficient Provers

Definition 4 An interactive proof (P, V) for $L \in \mathrm{NP}$ has an *efficient* prover if P is PPT and completeness holds whenever P receives (x, y), where y is a "witness" for x, for some witness relation on L.

Now we consider an interactive proof for Graph Isomorphism in Figure **??** We claim that this protocol is an interactive proof with efficient prover.

- Completeness: Suppose $G_1 \approx G_2$. Then there exists a permutation ρ such that $G_2 = \rho(G_1)$. Let $H = \pi(G_1)$, where π is a permutation. Then there is a correspondence among G_1, G_2, H as shown in Figure **??**. Therefore, if $c = 1$, $G_1 = \pi^{-1}(H)$, and if $c = 2$, $G_2 = \rho \circ \pi^{-1}(H)$. Hence, $G_c = \sigma(H)$ always, so V always accepts.

- Soundness: If $G_1 G_2$, then H is isomorphic to only G_1. Picking c at random, V will discover that $G_c H$ with probability $1/2$. That is, with probability $1/2$, P can convince

Input: $x = G_1, G_2$, where $G_2 = \rho(G_1)$
Variable: n is the number of vertices in each graph (which is the same).

$P(G_1, G_2, \rho)$		$V(G_1, G_2)$
Pick $\pi \leftarrow S_n$. Compute $H = \pi(G_1)$.		
	$\xrightarrow{H}$	
		Pick $b \leftarrow \{1, 2\}$ randomly. (Ask P: How to go from H to G_c?)
	$\xleftarrow{c}$	
Compute $\sigma = \begin{cases} \pi^{-1} & \text{if } c = 1 \\ \rho \circ \pi^{-1} & \text{if } c = 2 \end{cases}$		
	$\xrightarrow{\sigma}$	
		Accept if $G_c = \sigma(H)$.

Repeat this interaction n times.

Figure 3: An interactive proof for graph isomorphism

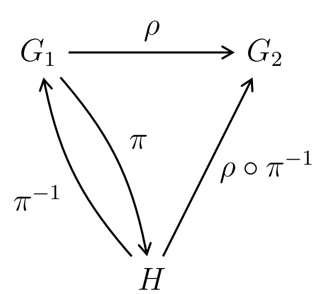

Figure 4: Correspondences among G_1, G_2, and H

V even though the two graphs are not isomorphic. Therefore, the probability that P convinces V for all the n rounds is $1/2^n$, which is negligible.

Note: This protocol will not be zero-knowledge if H is repeated (i.e., the same π is chosen twice) and both $c = 1, 2$ are chosen for this H.

Proposition 5 This interactive protocol is zero-knowledge for honest verifier.
Proof Idea: If the protocol is zero-knowledge, V needs to generate (H, c, σ) so that

- $\sigma(H) = G_c$,
- c is uniform in $\{1, 2\}$, and
- H is a uniform graph isomorphic to G_1, i.e., $\pi \leftarrow S_n$.

This can be done by generating c first, then H, so that $G_c \approx \sigma(H)$ always. $\square$

Note that for dishonest V, V can fix c and the above proof would not follow (because c is not uniformly chosen).