Non-Interactive Zero Knowledge (II)

CS 601.442/642 Modern Cryptography

Fall 2017

NIZKs for **NP**: Roadmap

- **Last-time:** Transformation from NIZKs in hidden-bit model to NIZKs in common random string model
- **Today:** NIZKs for **NP** in the hidden-bit model
- **Homework:** Non-adaptive NIZKs to Adaptive NIZKs

Hamiltonian Graphs

Definition (Hamiltonian Graph)

Let $G = (V, E)$ be a graph with $|V| = n$. We say that G is a Hamiltonian graph if it has a Hamiltonian cycle, i.e., there are $v_1, \ldots, v_n \in V$ s.t. for all $i \in [n]$:

$$(v_i, v_{(i+1) \bmod n}) \in E$$

Fact: Deciding whether a graph is Hamiltonian is **NP**-Complete. Let L_{H} be the language of Hamiltonian graphs $G = (V, E)$ s.t. $|V| = n$

Today: NIZK proof system for L_{H} in the hidden-bit model

Notation

Definition (Adjacency Matrix)

A graph $G = (V, E)$ with $|V| = n$, can be represented as an $n \times n$ adjacency matrix M_G of boolean values such that:

$$M[i,j] = \begin{cases} 1 & \text{if } (i,j) \in E \\ 0 & \text{otherwise} \end{cases}$$

Cycle Matrix: A cycle matrix is a boolean matrix that corresponds to a graph that contains a Hamiltonian cycle and no other edges

Permutation Matrix: A permutation matrix is a boolean matrix such that each row and each column has exactly one entry equal to 1

Fact: Every cycle matrix is a permutation matrix, but the converse is not true. For every n, there are $n!$ permutation matrices, but only $(n-1)!$ cycle matrices

NIZKs for L_{H} in Hidden-Bit Model

Two Steps:

Step I. NIZK $(\mathsf{K}_1, \mathsf{P}_1, \mathsf{V}_1)$ for L_{H} in hidden-bit model where K produces (hidden) strings r with a specific distribution: each r represents an $n \times n$ cycle matrix

Step II. Modify the above construction to obtain $(\mathsf{K}_2, \mathsf{P}_2, \mathsf{V}_2)$ where the (hidden) string r is uniformly random

Step I

Construction of $(\mathsf{K}_1, \mathsf{P}_1, \mathsf{V}_1)$ for L_{H}:

$\mathsf{K}_1(1^n)$: Output $r \leftarrow \{0,1\}^{n^2}$ s.t. it represents an $n \times n$ cycle matrix M_c

$\mathsf{P}_1(r, x, w)$: Execute the following steps:

- Parse $x = G = (V, E)$ s.t. $|V| = n$, and $w = H$ where $H = (v_1, \ldots, v_n)$ is a Hamiltonian cycle in G
- Choose a permutation $\varphi : V \to \{1, \ldots, n\}$ that maps H to the cycle in M_c, i.e., for every $i \in [n]$:

$$M_c[\varphi(v_i), \varphi(v_{(i+1) \bmod n})] = 1$$

- Define $I = \{\varphi(u), \varphi(v) | M_G[u, v] = 0\}$ to be the set of non-edges in G
- Output (I, φ)

Step I (contd.)

Construction of $(\mathsf{K}_1, \mathsf{P}_1, \mathsf{V}_1)$ **for** L_{H}**:**

$\mathsf{V}_1(I, r_I, \varphi)$: Execute the following steps:

- Parse $r_I = \{M_c[u,v]\}_{(u,v)\in I}$
- Check that for every $(u,v) \in I$, $M_c[u,v] = 0$
- Check that for every $(u,v) \in I$, $M_G(\varphi^{-1}(u), \varphi^{-1}(v)) = 0$
- If both the checks succeed, then output 1 and 0 otherwise

Completeness: An honest prover P can always find a correct mapping φ that maps H to the cycle in M_c

Soundness: If $G = (V, E)$ is not a Hamiltonian graph, then for any mapping $\varphi : V \to \{1, \ldots, n\}$, $\varphi(G)$ will not cover all the edges in M_c. There must exist at least one non-zero entry in M_c that is revealed as a non-edge of G

Step I (contd.)

Zero Knowledge: Simulator $\mathcal{S}$ performs the following steps:

- Sample a random permutation $\varphi : V \to \{1, \ldots, n\}$
- Compute $I = \{\varphi(u), \varphi(v) | M_G[u, v] = 0\}$
- For every $(a, b) \in I$, set $M_c[a, b] = 0$
- Output $(I, \{M_c[a, b]\}_{(a,b) \in I}, \varphi)$

It is easy to verify that the above output distribution is identical to the real experiment

Step II: Strategy

- Define a deterministic procedure Q that takes as input a (sufficiently long) random string r and outputs a biased string s that corresponds to a cycle matrix with inverse polynomial probability $\frac{1}{\ell(n)}$
- If we feed Q $n \cdot \ell(n)$ random inputs, then with high probability, at least one of the outputs will correspond to a cycle matrix
- In the NIZK construction, the (hidden) random string will be $r = r_1, \ldots, r_{n \cdot \ell(n)}$
- For every i, the prover will try to compute a proof using $s_i = Q(r_i)$
- At least one s_i will contain a cycle matrix, so we can use the NIZK proof system from Step I

Procedure Q

Let r be a random string s.t. $|r| = \lceil 3 \log n \rceil \cdot n^4$

Procedure $Q(r)$**:**

- Parse $r = r_1, \ldots, r_{n^4}$ s.t. $\forall i$, $|r_i| = \lceil 3 \log n \rceil$
- Compute $s = s_1, \ldots, s_{n^4}$, where:

$$s_i = \begin{cases} 1 & \text{if } r_i = 111 \cdots 1 \\ 0 & \text{otherwise} \end{cases}$$

- Define an $n^2 \times n^2$ boolean matrix M consisting of entries from s
- If M contains an $n \times n$ sub-matrix M_c s.t. M_c is a cycle matrix, then output (M, M_c), else output $(M, \perp)$

Analysis of Q

Notation. Let GOOD be the set of outputs of $Q(\cdot)$ that contain a cycle matrix and BAD be the complementary set

Lemma

For a random input r, $\Pr[Q(r) \in \text{GOOD}] \geqslant \frac{1}{3n^3}$

Let M be an $n^2 \times n^2$ matrix computed by Q on a random input r. We will prove the above lemma via a sequence of claims:

Claim 1: M contains exactly n 1's with probability at least $\frac{1}{3n}$

Claim 2: M contains a permutation sub-matrix with probability at least $\frac{1}{3n^2}$

Claim 3: M contains a cycle sub-matrix with probability at least $\frac{1}{3n^3}$

Analysis of Q (contd.)

Proof of Claim 1: Let X be the random variable denoting the number of 1's in M

- X follows the binomial distribution with $N = n^4$, $p = \frac{1}{n^3}$
- $\mathsf{E}(X) = N \cdot p = n$
- $\mathsf{Var}(X) = Np(1-p) < n$
- Recall Chebyshev's Inequality: $\Pr\left[|X - \mathsf{E}(X)| > k\right] \leqslant \frac{\mathsf{Var}(X)}{k^2}$
  Setting $k = n$, we have:

$$\Pr\left[|X - n| > n\right] \leqslant \frac{1}{n}$$

- Observe:

$$\sum_{i=1}^{2n} \Pr[X = i] = 1 - \Pr\left[|X - n| > n\right] > 1 - \frac{1}{n}$$

Analysis of Q (contd.)

Proof of Claim 1 (contd.):

- $\Pr[X = i]$ is maximum at $i = n$
- Observe:

$$\begin{aligned} \Pr[X = n] &\geqslant \frac{\sum_{i=1}^{2n} \Pr[X = i]}{2n} \\ &\geqslant \frac{1}{3n} \end{aligned}$$

Analysis of Q (contd.)

Proof of Claim 2: Want to bound the probability that each of the n '1' entries in M is in a different row and column

- After k '1' entries have been added to M,

$$\Pr[\text{new '1' entry is in different row and column}] = \left(1 - \frac{k}{n^2}\right)^2$$

- Multiplying all:

$$\begin{aligned}\Pr[\text{no collision}] &\geqslant \left(1 - \frac{1}{n^2}\right)^2 \cdots \left(1 - \frac{n-1}{n^2}\right)^2 \\ &\geqslant \frac{1}{n}\end{aligned}$$

- Combining the above with Claim 1,

$$\Pr[M \text{ contains a permutation } n \times n \text{ submatrix }] \geqslant \frac{1}{3n^2}$$

Analysis of Q (contd.)

Proof of Claim 3: Want to bound the probability that M contains an $n \times n$ cycle sub-matrix

- Observe:

$$\Pr[n \times n \text{ permutation matrix is a cycle matrix}] = \frac{1}{n}$$

- Combining the above with Claim 2,

$$\Pr[M \text{ contains a cycle } n \times n \text{ submatrix }] \geqslant \frac{1}{3n^3}$$

Step II: Details

Construction of $(\mathsf{K}_2, \mathsf{P}_2, \mathsf{V}_2)$ **for** L_{H}**:**

$\mathsf{K}_2(1^n)$: Output $r \leftarrow \{0,1\}^L$ where $L = \lceil 3 \log n \rceil \cdot n^8$

$\mathsf{P}_2(r, x, w)$: Parse $r = r_1, \ldots, r_{n^4}$ s.t. for every $i \in [n^4]$, $|r_i| = \lceil 3 \log n \rceil \cdot n^4$. For every $i \in [n^4]$:

- If $Q(r_i) = (M^i, \perp)$, set $I_i = [|r_i|]$ (i.e., reveal the entire r_i), and $\pi_i = \emptyset$
- Else, let $(M^i, M_c^i) \leftarrow Q(r_i)$. Compute $(I_i', \varphi_i) \leftarrow \mathsf{P}_1(M_c^i, x, w)$. Set $I_i = I_i' \cup J_i$ where J_i is the set of indices s.t. r_i restricted to J_i yields the residual M^i after removing M_c^i, and $\pi_i = \varphi_i$

Output $(I = \{I_i\}, \pi = \{\pi_i\})$

Step II: Details (contd.)

Construction of $(\mathsf{K}_2, \mathsf{P}_2, \mathsf{V}_2)$ **for** L_{H}**:**

$\mathsf{V}_2(I, r_I, \pi)$: Parse $I = I_1, \ldots, I_{n^4}$, $r_I = s_1, \ldots, s_{n^4}$, and $\pi = \pi_1, \ldots, \pi_{n^4}$. For every $i \in [n^4]$:

- If I_i is the complete set, then check that $Q(s_i) = (\cdot, \perp)$
- Otherwise, parse $I_i = I_i' \cup J_i$. Parse $s_i = s_i^1, s_i^2$ and check that s_i^2 is the all 0 string. Also, check that $\mathsf{V}_1(I_i', s_i^1, \pi_i) = 1$

If all the checks succeed, then output 1 and 0 otherwise

Step II: Security

Completeness: Follows from completeness of the construction in Step I

Soundness: For random $r = r_1, \ldots, r_{n^4}$, $Q(r_i) \in \textsc{Good}$ for at least one r_i with high probability. Soundness then follows from the soundness of the construction in Step I

Zero-Knowledge: For i s.t. $Q(r_i) \in \textsc{Good}$, V does not learn any information from the zero-knowledge property of the construction in Step I. For i s.t. $Q(r_i) \in \textsc{Bad}$, V does not see anything besides r_i.