Problem Set 8

Jaan Tollander de Balsch

March 16, 2020

H8.1

Zero-knowledge interactive proof system for graph isomorphism problem.

RSA cryptosystem: Key pair (p, q, e, d), message x and encrypted message y.

Graph isomorphism: Graphs $G = (V, E)$ and $G' = (V', E')$ are isomorphic if and only if there exists a bijection

$$\chi : V \to V'$$

such that for all $(i, j) \in E$ there exists $(\chi(i), \chi(j)) \in E'$.

We say that χ is a *certificate of graph isomorphism*.

Alice wants to convince **Bob** with high probability that she has certificate of graph isomorphism without revealing any other information about the certificate.

Alice and **Bob** know the input $x = (G, G')$ which is a pair of graphs $G = (V, E)$ and $G' = (V', E')$ such that $|V| = |V'|$ and $|E| = |E'|$.

Alice claims to have a certificate χ of graph isomorphism of input x.

Protocol: In each round

1) **Alice**
   - Creates a random permutation $\pi : V' \to V'$
   - For each vertex $i \in V$: Generate RSA key pair (p_i, q_i, e_i, d_i) and compute y_i, a randomized RSA coding of $\pi(\chi(i))$ and reveals $(e_i, p_i q_i, y_i)$ to Bob.
   - Reveals the permutation of the edges $E'_\pi = \{(\pi(i'), \pi(j')) \mid (i', j') \in E'\}$ to Bob.

(I am not sure if this needs to be encrypted as well? Bob might be able to infer knowledge from E'_π)

2) **Bob** picks two random vertices $i, j \in V$ and Alice reveals values d_i, d_j.

3) **Bob** decodes y_i, y_j to obtain $i'_\pi = \pi(\chi(i)), j'_\pi = \pi(\chi(j))$ and checks

   - Bijectivity: If $i \neq j$ then $i'_\pi \neq j'_\pi$.
   - Adjacency: If $(i, j) \in E$ then $(i'_\pi, j'_\pi) \in E'_\pi$ otherwise $(i'_\pi, j'_\pi) \notin E'_\pi$.

Alice must send the whole encrypted certificate so that she cannot fake the certificate for vertices $i, j \in V$ after Bob asks for them.

Bob must pick the vertices at random so that Alice cannot predict these vertices and fake the certificate for those vertices.

H8.2

$$\mathbf{P}^{\mathbf{PP}} = \mathbf{P}^{\#\mathbf{P}}$$

If a polynomial time Turing machine with **#P** oracle does a query and receive answer x, then a polynomial time Turing machine with **PP** oracle can do $|x|$ queries to obtain x, such that first one obtains the most significant bit of x, second one the second most significant bit of x and so forth.

Since the difference between the amount of queries is linear, the classes are equal.

H8.3