Chapter 07

Mingjia Huo

Problem 7.2. Proof by contradiction: if $g = (f(x_1), x_2)$ is not a one-way function, there is an `PPT` $\mathcal{A}$, such that:

$$\Pr[\text{Invert}_{\mathcal{A},g}(n) = 1] > \text{negl}(n)].$$

Construct $\mathcal{A}'$ based on $\mathcal{A}$:

1. When $\mathcal{A}'$ is given $y(= f(x_1))$, he uniformly chooses a $x_2 \in \{0,1\}^n$, and give (y, x_2) to $\mathcal{A}$.
2. When $\mathcal{A}$ return (x_1', x_2'), then $\mathcal{A}'$ output x_1'.

The input of $\mathcal{A}$ and $\mathcal{A}'$ are both poly($|x_1|$), so $\mathcal{A}$ is `PPT`. If $\mathcal{A}$ can invert g correctly, then $x_1' \in f^{-1}(f(x_1))$, thus $\mathcal{A}'$ can invert f correctly. So:

$$\Pr[\text{Invert}_{\mathcal{A}',f}(n) = 1] \geq \Pr[\text{Invert}_{\mathcal{A},g}(n) = 1] > \text{negl}(n),$$

a contradiction.

Thus, g is a one-way function.

Problem 7.3. Let f be a one-way function and let $p(\cdot)$ be a polynomial such that $|f(x)| < p(|x|)$. (If p doesn't exist, then there is no algorithm which can compute $f(x)$ in poly($|x|$).) Without loss of generality, p is increasing with n.

Let function $q(n)$ denotes the largest value len such that $p(len) \leq n$. So we have $p(q(n) + 1) > n$, and $n = \text{poly}(q(n) + 1) = \text{poly}(q(n))$.

Then given $x \in \{0,1\}^n$, x_q denotes the first $q(n)$ bits of x. (That is: if $x = x_1 \cdots x_n$, then $x_q = x_1 \cdots x_{q(n)}$.)

Finally, define $f' : \{0,1\}^* \to \{0,1\}^*$ as followed:

$$f'(x) = f(x_q) \| 10^{|x| - |f(x_q)| - 1}.$$

- f' is length-preserving:

  $$\begin{aligned} &|f(x_q) \| 10^{|x|-|f(x_q)|-1}| \\ =&|f(x_q)| + 1 + |x| - |f(x_q)| - 1 \\ =&|x|. \end{aligned}$$

  And $|f(x_q)| < p(|x_q|) \leq |x|$, thus $|x| - |f(x_q)| - 1 \geq 0$.

  So it's length-preserving.

- f' is one-way:

  If f' is not one-way, assume there is an `PPT` $\mathcal{A}', n'$, such that

  $$\Pr[\text{Invert}_{\mathcal{A}',f'}(n') = 1] > \text{negl}(n')].$$

  Assume $q(n') = n$, construct $\mathcal{A}$ of f based on $\mathcal{A}'$ when $\mathcal{A}'$ is given 1^n:

  1. Given $y, 1^n$, $\mathcal{A}$ constructs $y \| 10^{n'-|y|-1}$, and gives it and $1^{n'}$ to $\mathcal{A}'$.
  2. When $\mathcal{A}'$ outputs a value $x' = x_1 \cdots x_{n'}$, get the first n bits($x_1 \cdots x_n = x$) and output x.

  By the definition of f', if $f'(x') = y \| 10^{n'-|y|-1}$, then $f(x) = y$. Thus

  $$\Pr[\text{Invert}_{\mathcal{A},f}(n) = 1] \geq \Pr[\text{Invert}_{\mathcal{A}',f'}(n') = 1] > \text{negl}(n')$$

  We have proved that $n' = \text{poly}(n)$, thus $\mathcal{A}$ is `PPT` and

  $$\Pr[\text{Invert}_{\mathcal{A},f}(n) = 1] > \text{negl}(n).$$

  That's a contradiction.

  Thus f' is one-way.

Problem 7.6.

No.

Let $f' : \{0,1\}^{n-1} \to \{0,1\}^{n-1}$ be a length-preserving one-way function, construct $f : \{0,1\}^n \to \{0,1\}^n$:

$$f(x) = f'(x_1 \cdots x_{n-1}) \| 0, x = x_1 \cdots x_n.$$

First prove f is a length-preserving one-way function.

Obviously, it's length-preserving. Then $\forall$ algorithm $\mathcal{A}$ for f, construct an $\mathcal{A}'$ for f':

1. When $\mathcal{A}'$ is given y, construct $y\|0$ and give it to $\mathcal{A}$.
2. When $\mathcal{A}$ output $x_1 \cdots x_n$, output $x = x_1 \cdots x_{n-1}$

If $\mathcal{A}$ can invert f with non negligible probability, then

$$\Pr[\mathrm{Invert}_{\mathcal{A},f}(n) = 1] > \mathrm{negl}(n).$$

And

$$\begin{aligned}\Pr[\mathrm{Invert}_{\mathcal{A}',f'}(n-1) = 1] =& \Pr[f'(x_1 \cdots x_{n-1}) = y] \\ =& \Pr[f(x_1 \cdots x_n) = y\|0] \\ =& \Pr[\mathrm{Invert}_{\mathcal{A},f}(n) = 1]\end{aligned}$$

Thus,

$$\Pr[\mathrm{Invert}_{\mathcal{A}',f'}(n-1) = 1] > \mathrm{negl}(n),$$

a contradiction. So f is a length-preserving one-way function.

Use $G(x) = f(x)\|\mathrm{hc}(x) = f'(x_1 \cdots x_{n-1})\|0\|\mathrm{hc}(x)$, here we have $x \in \{0,1\}^n$. And construct D, when the input is $s \in \{0,1\}^{n+1}$:

1. if $s_n = 0$, output 0;
2. if $s_n = 1$, output 1.

So $\Pr[D(G(x)) = 0] = 1$. But if we uniformly draw $r \in \{0,1\}^{n+1}$, $\Pr[D(r) = 0] = \frac{1}{2}$. So it's not a pseudorandom generator.

Problem 7.8. Part1: $g = f(f(x))$ is not necessarily a one-way function.

By problem 7.3, we have if there is a one-way function, there is also a length-preserving one-way function, denoted as f.

Given $x = x_1, \cdots x_n$, we prove $f'(x) = f(x_1 \cdots x_{n-1})$ is also a one-way function.(Specifically, if $n = 1$, then $f'(x) = f(x)$.)

Assume we have $\mathcal{A}'$ to invert f', construct $\mathcal{A}$ to invert f:

1. Given y and 1^n, $\mathcal{A}$ give $y, 1^{n+1}$ to $\mathcal{A}'$.
2. When $\mathcal{A}'$ outputs $x' = x_1 \cdots x_{n+1}$, output $x = x_1 \cdots x_n$.

So,

$$\begin{aligned}\Pr[\mathrm{Invert}_{\mathcal{A}',f'}(n+1) = 1] =& \Pr[f'(x_1 \cdots x_n \| x_{n+1}) = y] \\ =& \Pr[f(x_1 \cdots x_n) = y] \\ =& \Pr[\mathrm{Invert}_{\mathcal{A},f}(n) = 1]\end{aligned}$$

Since f is a one-way function, $\Pr[\mathrm{Invert}_{\mathcal{A},f}(n) = 1] \leq \mathrm{negl}(n)$. Thus, $\forall \mathcal{A}', \Pr[\mathrm{Invert}_{\mathcal{A}',f'}(n+1) = 1] \leq \mathrm{negl}(n)$, that is

$$\Pr[\mathrm{Invert}_{\mathcal{A}',f'}(n) = 1] \leq \mathrm{negl}(n).$$

So $f' : \{0,1\}^n \to \{0,1\}^{n-1} (n > 1)$ is also a one-way function.

And if $f'(f'(x))$ is a one-way function, then $f'^{(2^t)}(x)$ is a one-way function. Note that $f'^{(2^t)} : \{0,1\}^n \to \{0,1\}^{n-2^t}$.

Define $n = 2^k + 1$, then if we set $t = k$, given $x = x_1 \cdots x_n$, $f'^{(2^t)}(x) = f'(x_1) = b, b \in \{0,1\}$. However, an algorithm $\mathcal{A}$ could compute $f(x_1) = b$ and get x_1 in constant time, then randomly choose $x_2 \cdots x_n \in \{0,1\}^{n-1}$, and output $x_1 \| x_2 \cdots x_n$, which is a valid answer. That is

$$\Pr[\text{Invert}_{\mathcal{A}, f'^{(2^t)}}(n+1) = 1] = 1.$$

So $g = f(f(x))$ is not necessarily a one-way function.

Part2: $g' = f(x) \| f(f(x))$ **is a one-way function.**
If g' is not, then

$$\exists \mathcal{A}, \Pr[\text{Invert}_{\mathcal{A}, g'}(n) = 1] > \text{negl}(n).$$

Construct A_f for f:

1. Given y, compute $z = f(y)$, and give $y \| z$ to $\mathcal{A}$.
2. When $\mathcal{A}$ output x, output x.

Then if $f(x) \| f(f(x)) = y \| z$, then $f(x) = y$. So

$$\Pr[\text{Invert}_{\mathcal{A}_f, f}(n) = 1] \geq \Pr[\text{Invert}_{\mathcal{A}, g'}(n) = 1] > \text{negl}(n),$$

a contradiction.
Thus, $g'f = f(x) \| f(f(x))$ is also a one-way function.

Problem 7.11. (a).
First, to invert a function of $\{0,1\}^n \to \{0,1\}^{p(n)}$ is in NP. That is, given $y (= f(x))$, we can guess a value for each $x_i, i = 1, 2, \cdots, n$, which can be done by a non deterministic turing machine.

Second, if $one-way$ function exists, then there is no PPT algorithm $\mathcal{A}$ which can invert it(except with negligible probability). Since a deterministic algorithm is also a PPT algorithm. Thus there is no deterministic algorithm can invert it in polynomial time. So it's not in P.

To sum up, $P \neq NP$.

(b).
Assume the parameter is n.

If $P \neq NP$, we have a language L and there is a non deterministic Turing Machine M such that: If $l \in L$, M accepts it in polynomial time (bounded by $t(n)$). But there is no deterministic Turing Machine which can do that.

Since M is non-determinism, it can take multiple paths and branch into multiple copies, each of which tries a different path. Define the path as $p \in \{0,1\}^{t(n)}$. Then we define,

$$f(w, p, flag) = \begin{cases} (1, w), & M(p) = accept \wedge flag = 0^n \\ (0, w), & otherwise \end{cases}$$

Here $flag$ is uniformly drawn from $\{0,1\}^n$.

(1).Since there are at most $t(n)$ steps along the path, $f(w, p)$ can be computed in polynomial time.

(2).If $\exists \text{PPT} \mathcal{A}, s.t. \Pr[f(\mathcal{A}(f(x))) = f(x)] = 1$, then we can replace the randomness by a type and get a deterministic algorithm $\mathcal{A}'$, such that $\mathcal{A}'(1, w) = (w, p)$.

Construct M'. When the input is w, run $(1, w)$ on $\mathcal{A}'$ and get (w, p). Then run M following the path p. And M' accept w if and only if M accepts. Thus $M' \in P$, a contradiction.

So f does not have a polynomial time computable right inverse and f is a hard to invert.

(3).Just construct $\mathcal{A}$:

- when $\mathcal{A}$ is given $(0, w)$, then randomly choose p, output $(w, p, 1^n)$.
- When $\mathcal{A}$ is given $(1, w)$, output an arbitrary value.

So $\mathcal{A}$ can invert when $flag \neq 0^n$. Since $\Pr[flag = 0^n] = 2^{-n}$, thus

$$\begin{aligned} & \Pr[f(\mathcal{A}(f(w, p, flag))) = f(w, p, flag)] \\ > & \Pr[f(\mathcal{A}(f(w, p, flag))) = f(w, p, flag) \wedge flag \neq 0^n] \\ = & \Pr[flag \neq 0^n] = 1 - 2^{-n}. \end{aligned}$$

So f is not one-way.
And since 2^{-n} is negligible for n, so f is not weakly one-way.

Problem 7.16. Construct D with oracle access to $\mathcal{O}(\cdot)$(Given (L_0, R_0), the oracle returns (L_2, R_2)):

1. Run 1^n. Randomly choose L_0, R_0 in $\{0,1\}^n$.
2. Get $\mathcal{O}(L_0, R_0)) = (L_2, R_2)$. Then compute $L_0' = L_2 \oplus L_0$.
3. Get $\mathcal{O}(L_0', R_0) = (L_2', R_2')$.
4. If $L_2' = 0^n$, output 1; otherwise, output 0.

If $\mathcal{O} = \pi$ which is truly random, then L_2' is a random string, thus

$$\Pr[D^{\pi(\cdot)}(1^n) = 1] = 2^{-n}.$$

If $\mathcal{O} = \text{Feistel}_{f_1,f_2}$, then

$$\begin{aligned} L_2' &= L_0' \oplus f_1(R_0) \\ &= L_2 \oplus L_0 \oplus f_1(R_0) \\ &= L_2 \oplus L_2 = 0^n \end{aligned}$$

Thus

$$\Pr[D^{\text{Feistel}_{f_1,f_2}(\cdot)}(1^n) = 1] = 1.$$

And

$$\Pr[D^{\text{Feistel}_{f_1,f_2}(\cdot)}(1^n) = 1] - \Pr[D^{\pi(\cdot)}(1^n) = 1] = 1 - 2^{-n}.$$

So it's not a pseudorandom permutation.

Problem 7.17. Construct D with oracle access to $\mathcal{O}(\cdot), \mathcal{O}^{-1}(\cdot)$:

1. Run 1^n. Randomly choose L_0, R_0 in $\{0,1\}^n$.
2. Get $\mathcal{O}(L_0, R_0) = (L_3, R_3)$.
3. Randomly choose $R_3' \neq R_3$ in $\{0,1\}^n$, ask $\mathcal{O}^{-1}(L_3, R_3') = (L_0', R_0')$.
4. Compute $L_0'' = R_3 \oplus R_3' \oplus L_0$, ask $\mathcal{O}(L_0'', R_0) = (L_3'', R_3'')$.
5. If $L_3'' = L_3 \oplus R_0 \oplus R_0'$, output 1; otherwise, output 0.

If $\mathcal{O} = \pi$ which is truly random, then L_3'' is a random string, thus

$$\Pr[D^{\pi(\cdot),\pi^{-1}(\cdot)}(1^n) = 1] = 2^{-n}.$$

If $\mathcal{O} = \text{Feistel}_{f_1,f_2,f_3}$,

1. After step 2, we have $L_3 = R_0 \oplus f_2(L_0 \oplus f_1(R_0)), R_3 = L_0 \oplus f_1(R_0) \oplus f_3(L_3)$.
2. After step 3, we have $L_3 = R_0' \oplus f_2(L_0' \oplus f_1(R_0')), R_3' = L_0' \oplus f_1(R_0') \oplus f_3(L_3)$.
3. So $L_0'' = R_3 \oplus R_3' \oplus L_0 = (L_0 \oplus f_1(R_0) \oplus f_3(L_3)) \oplus (L_0' \oplus f_1(R_0') \oplus f_3(L_3)) \oplus L_0 = L_0' \oplus f_1(R_0') \oplus f_1(R_0)$.
4. And in step 4, $L_3'' = R_0 \oplus f_2(L_0'' \oplus f_1(R_0)) = R_0 \oplus f_2(L_0' \oplus f_1(R_0') \oplus f_1(R_0) \oplus f_1(R_0)) = R_0 \oplus f_2(L_0' \oplus f_1(R_0')) = R_0 \oplus L_3 \oplus R_0'$.

Thus

$$\Pr[D^{\text{Feistel}_{f_1,f_2,f_3}(\cdot),\text{Feistel}^{-1}_{f_1,f_2,f_3}(\cdot)}(1^n) = 1] = 1.$$

And

$$\Pr[D^{\text{Feistel}_{f_1,f_2,f_3}(\cdot),\text{Feistel}^{-1}_{f_1,f_2,f_3}(\cdot)}(1^n) = 1] - \Pr[D^{\pi(\cdot),\pi^{-1}(\cdot)}(1^n) = 1] = 1 - 2^{-n}.$$

So it's not a strong pseudorandom permutation.

Problem 7.19. Let $\mathcal{A}$ be an arbitrary probabilistic polynomial-time algorithm. We show that $\Pr[\text{Invert}_{\mathcal{A},G}(n) = 1]$ is negligible.

To see this, consider the following PPT distinguisher D: on input a string $w \in \{0,1\}^{n+1}$, run $\mathcal{A}(w)$ to obtain output s. If $G(s) = w$ then output 1; otherwise, output 0.

Denote $W_0 = \{w \mid \exists s \in \{0,1\}^n, s.t. G(s) = w\}$.

If w is chosen by $G(s)$, then

$$\begin{aligned}\Pr[D(G(s)) = 1] =& \Pr[\text{Invert}_{\mathcal{A},G}(n) = 1] \\ =& \sum_{w \in W_0} \Pr[G(\mathcal{A}(w)) = w \mid W = w] \Pr[W = w] \\ \geq& 2^{-n} \sum_{w \in W_0} \Pr[G(\mathcal{A}(w)) = w \mid W = w]\end{aligned}$$

The last inequality holds because if $w \in W_0$, there exists at least one s, such that $G(s) = w$. So $\Pr[W = w] \geq \Pr[S = s] = 2^{-n}$.

If w is uniformly chosen from $\{0,1\}^{n+1}$, then

$$\begin{aligned}\Pr[D(w) = 1] =& \sum_{w \in W_0} \Pr[G(\mathcal{A}(w)) = w \mid W = w] \Pr[W = w] \\ =& 2^{-(n+1)} \sum_{w \in W_0} \Pr[G(\mathcal{A}(w)) = w \mid W = w]\end{aligned}$$

Let's denote r as a uniform string, so

$$\Pr[D(G(s)) = 1] \geq 2 \times 2^{-(n+1)} \sum_{w \in W_0} \Pr[G(\mathcal{A}(w)) = w \mid W = w] = 2 \times \Pr[D(r) = 1].$$

If $\Pr[\text{Invert}_{\mathcal{A},G}(n) = 1] > \frac{1}{p(n)}$, where $p(n) \in \text{poly}(n)$, then

$$\Pr[D(G(s)) = 1] - \Pr[D(r) = 1] \geq \frac{1}{2} \Pr[D(G(s)) = 1] = \frac{1}{2} \Pr[\text{Invert}_{\mathcal{A},G}(n) = 1] > \frac{1}{2p(n)},$$

which means G is not a pseudorandom generator, a contradiction.

Thus, G is a one-way function.

Problem 7.20. For arbitrary $D \in \texttt{PPT}$, since $\mathcal{X} \stackrel{c}{\equiv} \mathcal{Y}$, we have

$$\left| \Pr_{x \leftarrow X_n}[D(1^n, x) = 1] - \Pr_{y \leftarrow Y_n}[D(1^n, y) = 1] \right| \leq \text{negl}(n).$$

Similarly,

$$\left| \Pr_{y \leftarrow Y_n}[D(1^n, y) = 1] - \Pr_{z \leftarrow Z_n}[D(1^n, z) = 1] \right| \leq \text{negl}(n).$$

To sum up,

$$\left| \Pr_{x \leftarrow X_n}[D(1^n, x) = 1] - \Pr_{z \leftarrow Z_n}[D(1^n, z) = 1] \right| \leq \text{negl}(n).$$

So $\mathcal{X} \stackrel{c}{\equiv} \mathcal{Z}$.

Problem 7.22. For any $D \in \texttt{PPT}$ for $\{\mathcal{A}(X_n)\}_{n \in N}$ and $\{\mathcal{A}(Y_n)\}_{n \in N}$, construct D':

1. When given x, y, compute $\mathcal{A}(x), \mathcal{A}(y)$, and give them and 1^n to D.
2. Output the same as what D outputs.

Thus

$$\begin{aligned}\Pr_{x \leftarrow X_n}[D(1^n, x) = 1] &= \Pr_{\mathcal{A}(x) \leftarrow \mathcal{A}(X_n)}[D(1^n, \mathcal{A}(x)) = 1] \\ \Pr_{y \leftarrow Y_n}[D(1^n, y) = 1] &= \Pr_{\mathcal{A}(y) \leftarrow \mathcal{A}(Y_n)}[D(1^n, \mathcal{A}(y)) = 1]\end{aligned}$$

Since $\mathcal{X} \overset{c}{\equiv} \mathcal{Z}$, we have

$$\left| \Pr_{x \leftarrow X_n}[D(1^n, x) = 1] - \Pr_{y \leftarrow Y_n}[D(1^n, y) = 1] \right| \leq \mathrm{negl}(n).$$

So

$$\left| \Pr_{\mathcal{A}(x) \leftarrow \mathcal{A}(X_n)}[D(1^n, \mathcal{A}(x)) = 1] - \Pr_{\mathcal{A}(y) \leftarrow \mathcal{A}(Y_n)}[D(1^n, \mathcal{A}(y)) = 1] \right| \leq \mathrm{negl}(n).$$

That is $\{\mathcal{A}(X_n)\}_{n \in N} \overset{c}{\equiv} \{\mathcal{A}(Y_n)\}_{n \in N}$