Building Hard Problems by Combining Easy Ones: Revisited

Yael Eisenberg[*] Christopher Havens[†] Alexis Korb[‡] Amit Sahai[§]

Abstract

We establish the following theorem:

Let $\mathsf{O}_0, \mathsf{O}_1, \mathsf{R}$ be random functions from $\{0,1\}^n$ to $\{0,1\}^n$, $n \in \mathbb{N}$. For all polynomial-query-bounded distinguishers D making at most $q = \mathsf{poly}(n)$ queries to each oracle, there exists a poly-time oracle simulator $\mathsf{Sim}^{(\cdot)}$ and a constant $c > 0$ such that the probability is negligible, that is

$$\left|\Pr\left[\mathsf{D}^{(\mathsf{O}_0+\mathsf{O}_1),(\mathsf{O}_0,\mathsf{O}_1,\mathsf{O}_0^{-1},\mathsf{O}_1^{-1})}(1^n) = 1\right] - \Pr\left[\mathsf{D}^{\mathsf{R},\mathsf{Sim}^{\mathsf{R}}}(1^n) = 1\right]\right| = negl(n).$$

[*]Cornell University. Email: ye45@cornell.edu.

[†]Department of Computer Science, UCLA; Washington Corrections Center; and Prison Mathematics Project. Email: chavens28@gmail.com.

[‡]UCLA. Email: alexiskorb@cs.ucla.edu. ORCID: 0000-0001-6888-5296.

[§]UCLA. Email: sahai@cs.ucla.edu. ORCID: 0000-0003-2216-9600.

Contents

1 Introduction

In Theorem 3 of [GS23a, GS23b], Theorem 1.1 (below) was established, where the inverse oracles are formally defined as $\mathsf{O}_1^{-1} : \{0,1\}^m \to \{0,1\}^n \cup \{\perp\}$, and $\mathsf{O}_2^{-1} : \{0,1\}^m \to \{0,1\}^n \cup \{\perp\}$:

Theorem 1.1. *Let $\mathsf{O}_1, \mathsf{O}_2, \mathsf{R}$ are random functions from $\{0,1\}^n$ to $\{0,1\}^m$, $m = n+t(n)$, $m, n, t \in \mathbb{N}$. For all polynomial-query-bounded distinguishers D making at most $q = \mathsf{poly}(n)$ queries to each oracle, there exists a poly-time oracle simulator $\mathsf{Sim}^{(\cdot)}$ and a constant $c > 0$ such that*

$$\left|\Pr\left[\mathsf{D}^{(\mathsf{O}_1+\mathsf{O}_2),(\mathsf{O}_1,\mathsf{O}_2,\mathsf{O}_1^{-1},\mathsf{O}_2^{-1})}(1^n) = 1\right] - \Pr\left[\mathsf{D}^{\mathsf{R},\mathsf{Sim}^{\mathsf{R}}}(1^n) = 1\right]\right| \leq \frac{cq}{2^t} + \frac{6q^2}{2^{n+t}}.$$

Note that D's advantage becomes negligible when $t = \Omega(\log^{1+\varepsilon} n), \varepsilon > 0$

The theorem above trivializes if $t = m-n$ is too small, since we have a term 2^t in the denominator on the right hand side of the bound. Thus, the work of [GS23a] left open whether a meaningful similar theorem could be established for the case where t was small.

In this paper, we establish that a similar theorem holds even when $t = 0$, that is, when the domain and the codomain of the oracles is $\{0,1\}^n$.

2 Preliminaries

Definition 2.1 (Polynomial-Query-Bounded Oracle Turing Machine)**.** *We say that an oracle Turing machine $T^{(\cdot)}$ is* polynomial-query-bounded *if there exists a polynomial $p : \mathbb{N} \to \mathbb{N}$ such that for any input $x \in \{0,1\}^*$ and for any oracle O, the execution of $T^O(x)$ makes at most $p(|x|)$ many queries to O.* ♢

Definition 2.2 (Indifferentiability [MRH04, DKT16])**.** *] Let $\mathbf{C} : \{0,1\}^n \to \{0,1\}^{m(n)}$ be a construction having access to an ideal primitive $\mathsf{F} : \{0,1\}^{p(n)} \to \{0,1\}^{q(n)}$ and implements a functionality based on F, where $p(n), q(n), m(n) = poly(n)$. We say that $\mathbf{C}$ is* indifferentiable *from a random function $RO : \{0,1\}^n \to \{0,1\}^m$, if there is a poly-time simulator Sim with oracle access to RO such that for all polynomial-query-bounded distinguishers D, we have*

$$\left| \Pr\left[D^{\mathbf{C}^F,F}(1^n)\right] - \Pr\left[D^{RO,\mathsf{Sim}^{RO}}(1^n)\right] \right| \tag{1}$$

is negligible. ♢

3 Our Result

Formally, the inverse oracles are defined as $\mathsf{O}_0^{-1} : \{0,1\}^n \to \mathcal{P}(\{0,1\}^n)$, and $\mathsf{O}_1^{-1} : \{0,1\}^n \to \mathcal{P}(\{0,1\}^n)$. However we will denote the empty set of inverses by $\perp$. Below is our main result.

Theorem 3.1. *Let $\mathsf{O}_0, \mathsf{O}_1, \mathsf{R}$ be random functions from $\{0,1\}^n$ to $\{0,1\}^n$, $n \in \mathbb{N}$. There exists a negligible function $negl(n)$ such that: for all polynomial-query-bounded distinguishers D making at most $q = \mathsf{poly}(n)$ queries to each oracle, there exists a poly-time oracle simulator $\mathsf{Sim}^{(\cdot)}$ and a constant $c > 0$ such that:*

$$\left|\Pr\left[\mathsf{D}^{(\mathsf{O}_0+\mathsf{O}_1),(\mathsf{O}_0,\mathsf{O}_1,\mathsf{O}_0^{-1},\mathsf{O}_1^{-1})}(1^n) = 1\right] - \Pr\left[\mathsf{D}^{\mathsf{R},\mathsf{Sim}^{\mathsf{R}}}(1^n) = 1\right]\right| = negl(n).$$

We prove the result using a sequence of intermediate indifferentiable hybrids. To help present these hybrids, we define several working sets and registries, as well as some corresponding notations used throughout the paper.

Within our security proof, we will use the following registers:

- Reg_i will contain (domain, codomain) pairs (a, b_i) for oracle O_i. We write $\mathsf{Reg}_i(a) = b_i$ to mean that $(a, b_i) \in \mathsf{Reg}_i$.
- FReg will be the registry of determined pairs $(a, f(a))$, where $f(a) = \mathsf{O}_0(a) + \mathsf{O}_1(a)$.
- RList_i will be a set of some vectors $\boldsymbol{v}$. Each vector in RList_i has the form $\boldsymbol{v} = (\perp, b)$ if $\mathsf{O}_i^{-1}(b)$ is empty, or $\boldsymbol{v} = (a_1, a_2, ..., a_k, b)$, if $\{a_1, \ldots, a_k\}$ are the preimages of b under O_i.
- $\mathsf{DomList}_i = \{a \mid \exists b \text{ s.t. } \mathsf{Reg}_i(a) = b\}$. We will use α_i to denote the number of elements in $\mathsf{DomList}_i$: $\alpha_i := |\mathsf{DomList}_i|$.
- ImList_i will be a set of some codomain values $b \in \{0,1\}^n$ whose preimages have been completely determined in our experiments. Symbolically, we will maintain the invariant that $\mathsf{ImList}_i = \{b | \exists a \text{ s.t. } \mathsf{RList}_i(a) = b\}$. We will use β_i to denote the number of elements in ImList_i: $\beta_i := |\mathsf{ImList}_i|$.
- $\mathsf{CodList}_i = \{b \mid \exists a \text{ s.t. } \mathsf{Reg}_i(a) = b\}$.

Proof of Theorem 3.1. We present the proof using a hybrid argument. For notational simplicity, we use $(\mathsf{LOra}_i, \mathsf{ROra}_i)$ to denote the two oracles accessed by D in Hybrid i.

- **Hybrid H_1**
  This represents the case where D interacts with oracles $(\mathsf{LOra}_1, \mathsf{ROra}_1) = (\mathsf{O}_0 + \mathsf{O}_1, (\mathsf{O}_0, \mathsf{O}_0^{-1}, \mathsf{O}_1, \mathsf{O}_1^{-1}))$.
  - $\mathsf{LOra}_1(a)$: Returns $\mathsf{O}_0(a) + \mathsf{O}_1(a)$, where O_0 and O_1 are the real world random oracles.
  - $\mathsf{ROra}_1(a)$: Answers queries of the form $\mathsf{O}_0(a)$, $\mathsf{O}_0^{-1}(b)$, $\mathsf{O}_1(a)$, $\mathsf{O}_1^{-1}(b)$ according to the corresponding real world random oracles O_0 and O_1.
- **Hybrid H_2**
  This is the same as the previous hybrid except that we record oracle queries in the corresponding registers.
  - $\mathsf{LOra}_2(a)$:
    1. Stores $(a, \mathsf{O}_0(a))$ in Reg_0 and $(a, \mathsf{O}_1(a))$ in Reg_1.
    2. Returns $\mathsf{O}_0(a) + \mathsf{O}_1(a)$.
  - $\mathsf{ROra}_2(a)$:
    * On queries of the form $O_i(a)$:
      1. Set $b_i = \mathsf{O}_i(a)$ and store (a, b_i) in Reg_i
      2. Run $\text{FReact}_2(i, (a, b_i))$
      3. Return b_i.
    * On queries of the form $\mathsf{O}_i^{-1}(b)$:
      1. Compute the set S of all $a \in \{0,1\}^n$ such that $\mathsf{O}_i(a) = b$
      2. $\mathsf{ImList}_i = \mathsf{ImList}_i \cup \{b\}$
      3. Return the set S. (Note that if $S = \emptyset$ then $\perp$ is returned.)

  We define:

FReact$_2(i, (a, b_i))$:

1. If $(a, b_{1-i}) \notin \mathsf{Reg}_{1-i}$ for any b_{1-i}.
   (i.e. No query of the form $\mathsf{O}_{1-i}(a)$ has been made),
   (a) Set $b_{1-i} = \mathsf{O}_{1-i}(a)$ and store (a, b_{1-i}) in Reg_{1-i}

Lemma 3.2. *For all $n \in \mathbb{N}$ and all polynomial-query-bounded oracle Turing Machines D,*

$$\Pr[\mathsf{D}^{\mathsf{LOra}_1,\mathsf{ROra}_1}(1^n) = 1] = \Pr[\mathsf{D}^{\mathsf{LOra}_2,\mathsf{ROra}_2}(1^n) = 1]$$

Proof. It's straightforward to see that H_2 and H_1 are identical since we are simply recording additional information that does not affect the output of the left and right oracles. □

- **Hybrid** H_3
  In this hybrid, we begin perfect but inefficient simulations of O_0 and O_1. Thus, in this hybrid, there are no longer any oracles O_0 and O_1, rather they are simulated as described below.

  – $\mathsf{LOra}_3(a)$:
    1. Below, all calls to O_0 and O_1 are implemented as shown in ROra_3 below.
    2. Stores $(a, \mathsf{O}_0(a))$ in Reg_0 and $(a, \mathsf{O}_1(a))$ in Reg_1.
    3. Returns $\mathsf{O}_0(a) + \mathsf{O}_1(a)$.

  – ROra_3 begins a simulation and introduces several registries and sets:
    ∗ On queries of the form $\mathsf{O}_i(a)$:
      1. If $a \in \mathsf{DomList}_i$:
         (a) BEGIN FReact$_3(i, (a, \mathsf{Reg}_i(a)))$
         (b) Return $b_i = \mathsf{Reg}_i(a)$
      2. Else:
         (a) $b_i \xleftarrow{\$} \{0,1\}^n \setminus \mathsf{ImList}_i$
         (b) Set $\mathsf{Reg}_i(a) = b_i$
         (c) BEGIN FReact$_3(i, (a, b_i))$
         (d) Return b_i
    ∗ On queries of the form $\mathsf{O}_i^{-1}(b)$
      1. If $b \in \mathsf{ImList}_i$:
         (a) Return $\boldsymbol{v} \in \mathsf{RList}_i$ with b as a final component
      2. Else if $b \in \mathsf{CodList}_i \setminus \mathsf{ImList}_i$:
         (a) Let $\{c_1, c_2, ..., c_d\}$ be the set of all elements x such that $\mathsf{Reg}_i(x) = b$
         (b) Choose k from $\mathrm{Bin}(2^n - \alpha_i, (2^n - \beta_i)^{-1})$
         (c) Choose k additional domain elements $a_\ell \in \{0,1\}^n \setminus \mathsf{DomList}_i$ uniformly at random, without replacement
         (d) Set $\mathsf{Reg}_i(a_\ell) = b$ for $1 \leqslant \ell \leqslant k$.
         (e) $\mathsf{RList}_i = \mathsf{RList}_i \cup \{(c_1, c_2, ..., c_d, a_1, a_2, ..., a_k, b)\}$
         (f) $\mathsf{ImList}_i = \mathsf{ImList}_i \cup \{b\}$
         (g) Return $(c_1, c_2, ..., c_d, a_1, a_2, ..., a_k, b)$
      3. Else (if b is neither in ImList_i nor in $\mathsf{CodList}_i$):
         (a) Choose k from $\mathrm{Bin}(2^n - \alpha_i, (2^n - \beta_i)^{-1})$

(b) If $k = 0$:

i. $\mathsf{RList}_i = \mathsf{RList}_i \cup \{(\perp, b)\}$

ii. $\mathsf{ImList}_i = \mathsf{ImList}_i \cup \{b\}$

iii. Return $(\perp, b)$

(c) Else: (if $k > 0$)

i. Choose k domain elements $a_\ell \in \{0,1\}^n \setminus \mathsf{DomList}_i$ uniformly at random, without replacement

ii. Set $\mathsf{Reg}_i(a_\ell) = b$ for $1 \leqslant \ell \leqslant k$

iii. $\mathsf{RList}_i = \mathsf{RList}_i \cup \{(a_1, a_2, ..., a_k, b)\}$

iv. $\mathsf{ImList}_i = \mathsf{ImList}_i \cup \{b\}$

v. Return $(a_1, a_2, ..., a_k, b)$

We define:

FReact$_3(i, (a, b_i))$: Assume that $\mathsf{O}_i(a)$ has already been established and stored in Reg_i.

1. If $(a, c) \notin \mathsf{FReg}$, for any c:

   (a) If $\mathsf{Reg}_{1-i}(a)$ exists:

   i. Set $b_{1-i} = \mathsf{Reg}_{1-i}(a)$

   (b) Else:

   i. Set $\mathsf{Reg}_{1-i}(a)$ to a random element from $\{0,1\}^n \setminus \mathsf{ImList}_{1-i}$

   ii. Set $b_{1-i} = \mathsf{Reg}_{1-i}(a)$

2. Set $c = b_0 + b_1$

3. $\mathsf{FReg} = \mathsf{FReg} \cup \{(a, c)\}$

We will use the following lemmas to argue that this hybrid is a perfect simulation of the previous hybrid.

Lemma 3.3. *The number of oracles with domain* $\{0,1\}^n \setminus \mathsf{DomList}$ *and range* $\{0,1\} \setminus \mathsf{ImList}$ *is* $(2^n - \beta)^{2^n - \alpha}$.

Proof. There are 2^n codomain elements of which β have complete preimages, therefore there are $2^n - \beta$ remaining codomain elements. Similarly, there are $2^n - \alpha$ remaining domain elements. Therefore, the $2^n - \alpha$ remaining domain elements must be mapped to the $2^n - \beta$ remaining codomain values, which can be done in $(2^n - \beta)^{2^n - \alpha}$ ways. □

Lemma 3.4. *Suppose that in* H_2*, no previous query* O_i *or* O_{1-i} *on* x *has been made. On a query* $\mathsf{O}_i(x)$ *or* $\mathsf{O}_{1-i}(x)$*, the output will be one of the values* $y_i \in \{0,1\}^n \setminus \mathsf{ImList}_i$ *or* $y_{1-i} \in \{0,1\}^n \setminus \mathsf{ImList}_{1-i}$*. The probability that* $\mathsf{O}_i(x) = y_i$ *and* $\mathsf{O}_{1-i}(x) = y_{1-i}$ *is equal to* $\dfrac{1}{(2^n - \beta_i)(2^n - \beta_{1-i})}$.

Proof. By Lemma 3.3, there are $(2^n - \beta_i)^{2^n - \alpha_i}$ remaining oracles that are consistent with all queries already made.

Now set $\mathsf{O}_i(x) = y_i$ for some fixed value $y_i \in \{0,1\}^n \setminus \mathsf{ImList}_i$. Of the $(2^n - \beta_i)^{2^n - \alpha_i}$ possible oracles, there are still $(2^n - \beta_i)^{2^n-\alpha_i-1}$ remaining oracles after choosing y_i. Thus, the probability that $\mathsf{O}_i(x) = y_i$ is

$$\frac{(2^n - \beta_i)^{2^n-\alpha_i-1}}{(2^n - \beta_i)^{2^n-\alpha_i}} = \frac{1}{2^n - \beta_i}.$$

For O_{1-i}, there are a total of $(2^n - \beta_{1-i})^{2^n-\alpha_{1-i}}$ possible oracles (by Lemma 3.3). Since no previous query O_i on x has been made, $\mathsf{Reg}_{1-i}(x)$ is set to $\mathsf{O}_{1-i}(x)$ from the $\text{FReact}_3(x, y_i)$ process. After setting $\mathsf{O}_{1-i}(x) = y_{1-i}$, there are $(2^n - \beta_{1-i})^{2^n-\alpha_{1-i}-1}$ remaining oracles. Thus, the probability that $\mathsf{O}_{1-i}(x) = y_{1-i}$ is

$$\frac{(2^n - \beta_{1-i})^{2^n-\alpha_{1-i}-1}}{(2^n - \beta_{1-i})^{2^n-\alpha_{1-i}}} = \frac{1}{2^n - \beta_{1-i}}.$$

Finally, we have

$$\begin{aligned} Pr[\mathsf{O}_i(x) = y_i] \text{ and } Pr[\mathsf{O}_{1-i}(x) = y_{1-i}] &= \frac{1}{2^n - \beta_i} \times \frac{1}{2^n - \beta_{1-i}} \\ &= \frac{1}{(2^n - \beta_i)(2^n - \beta_{1-i})} \end{aligned}$$

□

Lemma 3.5. *Let $\mathsf{O}_i^{-1}(y)$ be a backwards query in H_2 where no such query has yet been made. Suppose that there exists a set $\{x_1, x_2, \ldots x_d\}$ such that $\mathsf{Reg}_i(x_\ell) = y$ for $1 \leqslant \ell \leqslant d$.*

The conditional probability that y has $\ell + d$ preimages given that $\alpha_i = |\mathsf{Reg}_i|$ and $\beta_i = |\mathsf{ImList}_i|$ is equal to

$$\binom{2^n - \alpha_i}{\ell}\left(\frac{1}{2^n - \beta_i}\right)^{\ell}\left(1 - \frac{1}{2^n - \beta_i}\right)^{2^n-\alpha_i-\ell}.$$

Proof. Observe that there are $\binom{2^n-\alpha_i}{\ell}$ subsets of exactly ℓ elements taken from a domain of size $2^n - \alpha_i$. There are now $2^n - \alpha_i - \ell$ remaining domain values which can be mapped to the remaining $2^n - \beta_i - 1$ codomain values. This can be done in $(2^n - \beta_i - 1)^{2^n-\alpha_i-\ell}$ ways, making

$$\binom{2^n - \alpha_i}{\ell}(2^n - \beta_i - 1)^{2^n-\alpha_i-\ell}$$

oracles for which there are exactly ℓ preimages for $\mathsf{O}_i^{-1}(y)$.
The probability that $\{x_1, x_2, \ldots, x_d, x_{1+d}, \ldots, x_{\ell+d}\}$ is exactly equal to $\mathsf{O}_i^{-1}(y)$, is then

$$\begin{aligned} \frac{\binom{2^n-\alpha_i}{\ell}(2^n - \beta_i - 1)^{2^n-\alpha_i-\ell}}{(2^n - \beta_i)^{2^n-\alpha_i}} &= \frac{\binom{2^n-\alpha_i}{\ell}(2^n - \beta_i - 1)^{2^n-\alpha_i-\ell}}{(2^n - \beta_i)^{\ell}(2^n - \beta_i)^{2^n-\alpha_i-\ell}} \\ &= \binom{2^n - \alpha_i}{\ell}\left(\frac{1}{2^n - \beta_i}\right)^{\ell}\left(1 - \frac{1}{2^n - \beta_i}\right)^{2^n-\alpha_i-\ell}. \end{aligned}$$

□

Without loss of generality, assume that whenever the adversary A makes a forward query, A immediately makes the corresponding backwards query.

Our goal is to show that:

1) All probabilities and conditional probabilities are exactly the same in H_2 as in H_3.

2) The number of remaining choices for oracles O_0 and O_1 is given by Lemma 3.3.

We will induct on the number q of queries.

Let $q = 0$. In H_2 on a forward query, we apply Lemma 3.4 to obtain the probability

$$\Pr\left[\, \mathsf{O}_i(x) = y_i \text{ and } \mathsf{O}_{1-i}(x) = y_{1-i} \,\right] \;=\; \frac{1}{\left(2^n - \beta_i\right)\left(2^n - \beta_{1-i}\right)} \;=\; \frac{1}{2^{2n}}.$$

In H_3 on a forward query, the probability that $y_i \xleftarrow{\$} \{0,1\}^n \setminus \mathsf{ImList}_i$ is equal to

$$\Pr\left[\, y_i \,\right] \;=\; \frac{1}{2^n - |\mathsf{ImList}_i|} \;=\; \frac{1}{2^n}.$$

The $\mathrm{FReact}_3(x, y_i)$ process then simulates the query $\mathsf{O}_{1-i}(x)$, which gives the value y_{1-i} with probability

$$\Pr\left[\, y_{1-i} \,\right] \;=\; \frac{1}{2^n - |\mathsf{ImList}_{1-i}|} \;=\; \frac{1}{2^n}.$$

Thus,

$$\Pr\left[\, y_i \,\right] \cdot \Pr\left[\, y_{1-i} \,\right] \;=\; \frac{1}{\left(2^n\right)^2} \;=\; \frac{1}{2^{2n}}.$$

In other words, the distributions in H_2 and H_3 on a forward query when $q = 0$ is identical.

On a backwards query $\mathsf{O}^{-1}(y)$ in H_2, we wish to find the probability that y has $k \geqslant 0$ preimages. There are a total of $\left(2^n\right)^{2^n}$ oracles. There are $\binom{2^n}{k}$ ways for which we can choose k domain elements. This leaves a remaining $2^n - k$ domain elements to be mapped to the remaining $2^n - 1$ codomain elements, for which there are $(2^n - 1)^{2^n - k}$ such oracles. Thus, there are

$$\binom{2^n}{k}\left(2^n - 1\right)^{2^n - k}$$

oracles for which y has k preimages, and therefore

$$\begin{aligned}
\Pr[y \text{ has } k \text{ preimages}] \;&=\; \frac{\binom{2^n}{k}\left(2^n - 1\right)^{2^n - k}}{\left(2^n\right)^{2^n}} \\
&=\; \frac{\binom{2^n}{k}\left(2^n - 1\right)^{2^n - k}}{\left(2^n\right)^{2^n - k}\left(2^n\right)^{k}} \\
&=\; \binom{2^n}{k}\left(\frac{1}{2^n}\right)^{k}\left(1 - \frac{1}{2^n}\right)^{2^n - k}.
\end{aligned}$$

In H_3, by construction and the definition of the Binomial distribution, since $\alpha_i = \beta_i = 0$, we have

$$Pr[y \text{ has } k \text{ preimages}] \;=\; \binom{2^n}{k}\left(\frac{1}{2^n}\right)^{k}\left(1 - \frac{1}{2^n}\right)^{2^n - k}.$$

Thus, when $q = 0$, the distributions in H_2 and H_3 are identical.

Now suppose that after $q = \ell - 1$ queries, 1) is satisfied so that H_2 and H_3 have the same probabilities and conditional probabilities.

By Lemma 3.4, on a forward query in H_2, we have

$$\Pr[\mathsf{O}_i(x) = y_i \text{ and } \mathsf{O}_{1-i}(x) = y_{1-i}] = \frac{1}{\big(2^n - \beta_i\big)\big(2^n - \beta_{1-i}\big)}$$

In H_3, the probability that $y_i \xleftarrow{\$} \{0,1\}^n \setminus \mathsf{ImList}_i$ is equal to

$$\Pr(y_i) = \frac{1}{2^n - |\mathsf{ImList}_i|} = \frac{1}{2^n - \beta_i}.$$

The $\mathrm{FReact}_3(x, y_i)$ process then simulates the query $\mathsf{O}_{1-i}(x)$, which gives, in a similar way, the value y_{1-i} with probability

$$\Pr(y_{1-i}) = \frac{1}{2^n - |\mathsf{ImList}_{1-i}|} = \frac{1}{2^n - \beta_{1-i}}.$$

Thus,

$$\begin{aligned}\Pr[\mathsf{O}_i(x) = y_i \text{ and } \mathsf{O}_{1-i}(x) = y_{1-i}] &= \Pr(y_i)\Pr(y_{1-i}) \\ &= \frac{1}{(2^n - \beta_i)(2^n - \beta_{1-i})},\end{aligned}$$

showing that the distributions in H_2 and H_3 on a forward query after q queries are identical.

On a backwards query $\mathsf{O}^{-1}(y)$ in H_2, for $1 \leqslant \ell \leqslant k$, denote by k_q the number of added domain values in Reg_i in the q-th query. Then we have

$$\alpha = k_1 + k_2 + \ldots + k_{\ell-1}$$

elements taken from the domain after $q = \ell - 1$ queries. Since there $\ell - 1$ queries, then there are at most $\beta = \ell - 1$ elements in ImList. By Lemma 3.3, the number of oracles with our current α and β is $(2^n - \beta)^{2^n-\alpha}$, for the given y. Since there are a total of $2^n - \alpha$ domain elements, then choosing an additional k_q domain elements from $\{0,1\}^n \setminus \mathsf{DomList}$ can be done in $\binom{2^n-\alpha}{k_q}$ ways. The remaining $2^n - \alpha - k_q$ domain elements must then be mapped to the remaining $2^n - \beta - 1$ codomain elements, for which there are $(2^n - \beta - 1)^{2^n-\alpha-k_q}$ such oracles. Thus, there are

$$\binom{2^n - \alpha}{k_q}(2^n - \beta - 1)^{2^n-\alpha-k_q}$$

oracles for which y has k_q preimages. The probability that y has k preimages is:

$$\begin{aligned}&\frac{\binom{2^n-\alpha}{k_q}(2^n - \beta - 1)^{2^n-\alpha-k_q}}{(2^n - \beta)^{2^n-\alpha}} \\ &= \frac{\binom{2^n-\alpha}{k_q}(2^n - \beta - 1)^{2^n-\alpha-k_q}}{(2^n - \beta)^{2^n-\alpha-k_q}(2^n - \beta)^{k_q}} \\ &= \binom{2^n - \alpha}{k_q}\left(\frac{1}{2^n - \beta}\right)^{k_q}\left(1 - \frac{1}{2^n - \beta}\right)^{2^n-\alpha-k_q}.\end{aligned}$$

In H_3, we have the identical probability by construction using the definition of the Bionomial distribution. Therefore, the distributions in H_2 and H_3 are identical, i.e.

$$\Pr[\mathsf{D}^{\mathsf{LOra}_2,\mathsf{ROra}_2}(1^n) = 1] = \Pr[\mathsf{D}^{\mathsf{LOra}_3,\mathsf{ROra}_3}(1^n) = 1].$$

In the following hybrid we will make our simulation efficient, at the cost of a small probability of error.

- **Hybrid** H_4
  - LOra_4 remains identical to LOra_3.
  - ROra_4 remains identical to ROra_3 with the difference being that abort conditions are introduced into O and O^{-1}:
    * On queries of the form $\mathsf{O}_i(a)$:
      1. If $a \in \mathsf{DomList}_i$, then
         (a) BEGIN $\mathrm{FReact}_4(a, \mathsf{Reg}_i(a))$
         (b) Return $b_i = \mathsf{Reg}_i(a)$
      2. Else:
         (a) $b_i \xleftarrow{\$} \{0,1\}^n \setminus \mathsf{ImList}_i$
         (b) If $b_i \in \mathsf{CodList}_i$, then ABORT
         (c) Set $\mathsf{Reg}_i(a) = b_i$
         (d) BEGIN $\mathrm{FReact}_4(a, b_i)$
         (e) Return b_i
    * On queries of the form $\mathsf{O}_i^{-1}(b)$
      1. If $b \in \mathsf{ImList}_i$:
         (a) Return $v \in \mathsf{RList}_i$ with b as a final component.
      2. Else if $b \in \mathsf{CodList}_i \setminus \mathsf{ImList}_i$:
         (a) Let $\{c_1, c_2, ..., c_d\}$ be the set of all elements x such that $\mathsf{Reg}_i(x) = b$
         (b) Choose k from $\mathrm{Bin}(2^n - \alpha_i, (2^n - \beta_i)^{-1})$
         (c) If $k > n$: then ABORT.
         (d) Choose k additional domain elements $a_\ell \in \{0,1\}^n \setminus \mathsf{DomList}_i$ uniformly at random, without replacement
         (e) Set $\mathsf{Reg}_i(a_\ell) = b$ for $1 \leqslant \ell \leqslant k$.
         (f) $\mathsf{RList}_i = \mathsf{RList}_i \cup \{(c_1, c_2, ..., c_d, a_1, a_2, ..., a_k, b)\}$
         (g) $\mathsf{ImList}_i = \mathsf{ImList}_i \cup \{b\}$
         (h) Return $(c_1, c_2, ..., c_d, a_1, a_2, ..., a_k, b)$
    * Else:
      1. Choose k from $\mathrm{Bin}(2^n - \alpha_i, (2^n - \beta_i)^{-1})$
      2. If $k = 0$, then:
         (a) $\mathsf{RList}_i = \mathsf{RList}_i \cup \{(\perp, b)\}$
         (b) $\mathsf{ImList}_i = \mathsf{ImList}_i \cup \{b\}$
         (c) Return $(\perp, b)$
      3. If $k > n$: then ABORT
      4. Else:

(a) Choose k domain elements $a_\ell \in \{0,1\}^n/\mathsf{DomList}_i$ uniformly at random, without replacement
(b) Set $\mathsf{Reg}_i(a_\ell) = b$ for $1 \leqslant \ell \leqslant k$
(c) $\mathsf{RList}_i = \mathsf{RList}_i \cup \{(a_1, a_2, ..., a_k, b)\}$
(d) $\mathsf{ImList}_i = \mathsf{ImList}_i \cup \{b\}$
(e) Return $(a_1, a_2, ..., a_k, b)$

In the algorithm above, FReact_4 remains identical to FReact_3. To calculate the transitional probability, we must only show that in H_4, a forward query $\mathsf{O}_i(x)$ aborts with negligible probability, as hybrid H_4 is otherwise identical to H_3.

The abort condition is triggered when $b_i \in \mathsf{CodList}_i$ after being randomly sampled from $\{0,1\}^n \setminus \mathsf{ImList}_i$. Note that $|\mathsf{CodList}_i|$ is bounded above by the number q of queries. Thus, $|\mathsf{CodList}_i| \leqslant q$ so that

$$\Pr[b_i \in \mathsf{CodList}_i] \leqslant \frac{q}{2^n}.$$

Next, we want to show that, on a backwards query in H_3 and H_4, the indistinguishability advantage is negligible. In H_3, we have a perfect simulation, but in H_4, we introduce abort conditions, which affect the distribution. It will suffice to show that the sum of probabilities in H_2 for $n+1 \leqslant k \leqslant 2^n-\alpha$ is negligible as this is the indistinguishability advantage in H_2 and H_3.

For $n+1 \leqslant k \leqslant 2^n - \alpha$, we have:

$$\begin{aligned}
&\binom{2^n-\alpha}{k}\left(\frac{1}{2^n-\beta}\right)^k\left(1-\frac{1}{2^n-\beta}\right)^{2^n-\alpha-k} \qquad (2)\\
&< \binom{2^n-\alpha}{k}\left(\frac{1}{2^n-\beta}\right)^k
\end{aligned}$$

In general, we have $\alpha \geqslant \beta$, so that by using Sterling's formula, we have

$$\begin{aligned}
&\leqslant \binom{2^n-\beta}{k}\left(\frac{1}{2^n-\beta}\right)^k\\
&< \left(\frac{e\big(2^n-\beta\big)}{k}\right)^k\left(\frac{1}{2^n-\beta}\right)^k = \left(\frac{e}{k}\right)^k \qquad (3)\\
&< \left(\frac{1}{2}\right)^k = \frac{1}{2^k}. \qquad (4)
\end{aligned}$$

Summing (2) over all possible k, we obtain by (4)

$$\sum_{k=n+1}^{2^n-\alpha} \binom{2^n-\alpha}{k} \left(\frac{1}{2^n-\beta}\right)^k \left(1-\frac{1}{2^n-\beta}\right)^{2^n-\alpha-k} \tag{5}$$

$$< \sum_{k=n+1}^{2^n-\alpha} \frac{1}{2^k} \tag{6}$$

$$= \frac{(1/2)^{2^n-\alpha+1}-(1/2)^{n+1}}{(1/2)-1}$$

$$= \frac{1}{2^n} - \frac{1}{2^{2^n-\alpha}} = \frac{1}{2^n}\left(1-\frac{1}{2^{2^n-\alpha-n}}\right)$$

$$< \frac{1}{2^n},$$

showing that the indistinguishability advantage is negligible, i.e.

$$\left|\Pr[\mathsf{D}^{\mathsf{LOra}_3,\mathsf{ROra}_3}(1^n)=1]-\Pr[\mathsf{D}^{\mathsf{LOra}_4,\mathsf{ROra}_4}(1^n)=1]\right| \leq \mathsf{negl}(n).$$

Finally, we will link to the random function R through the construction of Hybrid H_5.

- **Hybrid** H_5
  - LOra_5 remains identical to LOra_4.
  - ROra_5 introduces two separate abort conditions, one in the FReact_5 process and the other when $\mathsf{O}_i(a) = b \in \mathsf{ImList}_i$.
    - ∗ On queries of the form $\mathsf{O}_i(a)$:
      1. If $a \in \mathsf{DomList}_i$:
         (a) BEGIN $\mathrm{FReact}_4(a, \mathsf{Reg}_i(a))$
         (b) Return $b_i = \mathsf{Reg}_i(a)$
      2. Else:
         (a) $b_i \xleftarrow{\$} \{0,1\}^n$
         (b) If $b_i \in \mathsf{ImList}_i$:
            i. ABORT
         (c) Else:
            i. Set $\mathsf{Reg}_i(a) = b_i$
            ii. BEGIN $\mathrm{FReact}_4(a, b_i)$
            iii. Return b_i

  We define:

FReact$_5$: Assume that $\mathsf{O}_i(a)$ has already been established and stored in Reg_i.

1. If for some $c \in \{0,1\}^n$, we have $(a, c) \in \mathsf{FReg}$:
   (a) Set $\mathsf{Reg}_{1-i}(a) := R(a) - \mathsf{Reg}_i(a)$
   (b) Return TRUE

2. Else:

   (a) Send a as a query to the random function R.

   (b) If $R(a) - \mathsf{Reg}_i(a) \in \mathsf{CodList}_{1-i}$:

      i. CABORT

   (c) Else:

      i. Set $\mathsf{Reg}_{1-i} := R(a) - \mathsf{Reg}_i(a)$

      ii. $\mathsf{FReg} = \mathsf{FReg} \cup \{(a, R(a))\}$

Let x be the distribution for H_2 and H_3:

$$x = \sum_{k=n+1}^{2^n-\alpha} \binom{2^n-\alpha}{k} \left(\frac{1}{2^n-\beta}\right)^k \left(1 - \frac{1}{2^n-\beta}\right)^{2^n-\alpha-k}$$

Recall that the distribution for H_4 is less than $\left|x - \frac{1}{2^n}\right|$.
For H_5, we incorporate two abort conditions:

- FReact$_5$ triggers a CABORT, which happens with probability $\frac{q}{2^n}$.
- if $b_i \in \mathsf{ImList}_i$, then we ABORT which happens with probability $\frac{\beta_i}{2^n}$.

The sum of the probabilities is $\frac{q+\beta_i}{2^n}$.
Next, to find the difference in distribution between H_4 and H_5, we have

$$\left|\left(x - \frac{1}{2^n}\right) - \left(x - \frac{q+\beta_i}{2^n}\right)\right| = \frac{q+\beta_i-1}{2^n} \leqslant \frac{2q-1}{2^n}$$

since $\beta_i \leqslant q$. This indicates that the indistuinguishablility advantage between H_4 and H_5 is negligible, i.e.

$$\left|\Pr[\mathsf{D}^{\mathsf{LOra}_4,\mathsf{ROra}_4}(1^n) = 1] - \Pr[\mathsf{D}^{\mathsf{LOra}_5,\mathsf{ROra}_5}(1^n) = 1]\right| \leq \mathsf{negl}(n).$$

Combining all the hybrids completes the proof.

□

4 References

[DKT16] Dana Dachman-Soled, Jonathan Katz, and Aishwarya Thiruvengadam. 10-round Feistel is indifferentiable from an ideal cipher. In Marc Fischlin and Jean-Sébastien Coron, editors, *EUROCRYPT 2016, Part II*, volume 9666 of *LNCS*, pages 649–678, May 2016.

[GS23a] Riddhi Ghosal and Amit Sahai. Building hard problems by combining easy ones. In *IEEE International Symposium on Information Theory, ISIT 2023, Taipei, Taiwan, June 25-30, 2023*, pages 1770–1775. IEEE, 2023.

[GS23b] Riddhi Ghosal and Amit Sahai. Building hard problems by combining easy ones. Cryptology ePrint Archive, Paper 2023/1088, 2023.

[MRH04] Ueli M. Maurer, Renato Renner, and Clemens Holenstein. Indifferentiability, impossibility results on reductions, and applications to the random oracle methodology. In Moni Naor, editor, *TCC 2004*, volume 2951 of *LNCS*, pages 21–39, February 2004.