Pseudorandom Functions & Permutations

Iftach Haitner, Tel Aviv University

Bar Ilan Winter School

January 27, 2014

Today's Plan

1. One-way functions and hardcore predicates
2. Pseudorandom generators
3. Pseudorandom functions and permutations
4. Symmetric encryption and MACs.

Reminder: Repeated Sampling From Pseudorandom Distributions

Claim 1

Let $G\colon \{0,1\}^n \mapsto \{0,1\}^{m(n)}$ be a pseudorandom generator and let $t \in \mathrm{poly}$, then $G^t\colon (\{0,1\}^n)^{t(n)} \mapsto (\{0,1\})^{t(n)}$, defined by

$$G^t(x_1,\ldots,x_{t(n)}) = G(x_1),\ldots,G(x_{t(n)})$$

is a pseudorandom generator.

Reminder: Repeated Sampling From Pseudorandom Distributions

Claim 1

Let $G\colon \{0,1\}^n \mapsto \{0,1\}^{m(n)}$ be a pseudorandom generator and let $t \in \mathrm{poly}$, then $G^t\colon (\{0,1\}^n)^{t(n)} \mapsto (\{0,1\})^{t(n)}$, defined by

$$G^t(x_1,\ldots,x_{t(n)}) = G(x_1),\ldots,G(x_{t(n)})$$

is a pseudorandom generator.

Proof: ? via hybrid

Part I

Pseudorandom Functions

Motivation Discussion

1. We've seen a small set of objects: $\{G(x)\}_{x\in\{0,1\}^n}$, that "looks like" a larger set of objects: $\{x\}_{x\in\{0,1\}^{2n}}$.

Motivation Discussion

1. We've seen a small set of objects: $\{G(x)\}_{x\in\{0,1\}^n}$, that "looks like" a larger set of objects: $\{x\}_{x\in\{0,1\}^{2n}}$.
2. We want small set of objects: efficient function families, that looks like a huge set of objects: the set of <u>all</u> functions.

Motivation Discussion

1. We've seen a small set of objects: $\{G(x)\}_{x\in\{0,1\}^n}$, that "looks like" a larger set of objects: $\{x\}_{x\in\{0,1\}^{2n}}$.
2. We want small set of objects: efficient function families, that looks like a huge set of objects: the set of all functions.

But

Motivation Discussion

1. We've seen a small set of objects: $\{G(x)\}_{x \in \{0,1\}^n}$, that "looks like" a larger set of objects: $\{x\}_{x \in \{0,1\}^{2n}}$.
2. We want small set of objects: efficient function families, that looks like a huge set of objects: the set of all functions.

Solution

Random Functions

Definition 2 (random functions)

For $n, k \in \mathbb{N}$, let $\Pi_{n,k}$ be the family of all functions from $\{0,1\}^n$ to $\{0,1\}^k$.
Let $\Pi_n = \Pi_{n,n}$.

Random Functions

Definition 2 (random functions)

For $n, k \in \mathbb{N}$, let $\Pi_{n,k}$ be the family of all functions from $\{0,1\}^n$ to $\{0,1\}^k$.
Let $\Pi_n = \Pi_{n,n}$.

- $\pi \stackrel{R}{\leftarrow} \Pi_n$ is a “random access" source of randomness

Random Functions

Definition 2 (random functions)

For $n, k \in \mathbb{N}$, let $\Pi_{n,k}$ be the family of all functions from $\{0,1\}^n$ to $\{0,1\}^k$.
Let $\Pi_n = \Pi_{n,n}$.

- $\pi \xleftarrow{R} \Pi_n$ is a “random access" source of randomness
- Parties with access to a common $\pi \xleftarrow{R} \Pi_n$ can do a lot

Random Functions

Definition 2 (random functions)

For $n, k \in \mathbb{N}$, let $\Pi_{n,k}$ be the family of all functions from $\{0,1\}^n$ to $\{0,1\}^k$.
Let $\Pi_n = \Pi_{n,n}$.

- $\pi \stackrel{R}{\leftarrow} \Pi_n$ is a "random access" source of randomness
- Parties with access to a common $\pi \stackrel{R}{\leftarrow} \Pi_n$ can do a lot
- How long does it take to describe $\pi \in \Pi_n$?

Random Functions

Definition 2 (random functions)

For $n, k \in \mathbb{N}$, let $\Pi_{n,k}$ be the family of all functions from $\{0,1\}^n$ to $\{0,1\}^k$.
Let $\Pi_n = \Pi_{n,n}$.

- $\pi \stackrel{R}{\leftarrow} \Pi_n$ is a “random access" source of randomness
- Parties with access to a common $\pi \stackrel{R}{\leftarrow} \Pi_n$ can do a lot
- How long does it take to describe $\pi \in \Pi_n$?

Random Functions

Definition 2 (random functions)

For $n, k \in \mathbb{N}$, let $\Pi_{n,k}$ be the family of all functions from $\{0,1\}^n$ to $\{0,1\}^k$.
Let $\Pi_n = \Pi_{n,n}$.

- $\pi \xleftarrow{R} \Pi_n$ is a "random access" source of randomness
- Parties with access to a common $\pi \xleftarrow{R} \Pi_n$ can do a lot
- How long does it take to describe $\pi \in \Pi_n$? $2^n \cdot n$ bits

Random Functions

Definition 2 (random functions)

For $n, k \in \mathbb{N}$, let $\Pi_{n,k}$ be the family of all functions from $\{0,1\}^n$ to $\{0,1\}^k$.
Let $\Pi_n = \Pi_{n,n}$.

- $\pi \stackrel{R}{\leftarrow} \Pi_n$ is a “random access" source of randomness
- Parties with access to a common $\pi \stackrel{R}{\leftarrow} \Pi_n$ can do a lot
- How long does it take to describe $\pi \in \Pi_n$? $2^n \cdot n$ bits
- The truth table of $\pi \stackrel{R}{\leftarrow} \Pi_n$ is a uniform string of length $2^n \cdot n$

Ensembles of Function Families

- $\mathcal{F} = \{\mathcal{F}_n\}_{n\in\mathbb{N}}$, where $\mathcal{F}_n = \{f\colon \{0,1\}^{m(n)} \mapsto \{0,1\}^{\ell(n)}\}$
  (we simply write $\mathcal{F} = \{\mathcal{F}_n\colon \{0,1\}^{m(n)} \mapsto \{0,1\}^{\ell(n)}\}$)

Ensembles of Function Families

- $\mathcal{F} = \{\mathcal{F}_n\}_{n \in \mathbb{N}}$, where $\mathcal{F}_n = \{f \colon \{0,1\}^{m(n)} \mapsto \{0,1\}^{\ell(n)}\}$
  (we simply write $\mathcal{F} = \{\mathcal{F}_n \colon \{0,1\}^{m(n)} \mapsto \{0,1\}^{\ell(n)}\}$)
- We identify functions with their description

Ensembles of Function Families

- $\mathcal{F} = \{\mathcal{F}_n\}_{n\in\mathbb{N}}$, where $\mathcal{F}_n = \{f\colon \{0,1\}^{m(n)} \mapsto \{0,1\}^{\ell(n)}\}$

  (we simply write $\mathcal{F} = \{\mathcal{F}_n\colon \{0,1\}^{m(n)} \mapsto \{0,1\}^{\ell(n)}\}$)
- We identify functions with their description

Ensembles of Function Families

- $\mathcal{F} = \{\mathcal{F}_n\}_{n\in\mathbb{N}}$, where $\mathcal{F}_n = \{f\colon \{0,1\}^{m(n)} \mapsto \{0,1\}^{\ell(n)}\}$
  (we simply write $\mathcal{F} = \{\mathcal{F}_n\colon \{0,1\}^{m(n)} \mapsto \{0,1\}^{\ell(n)}\}$)
- We identify functions with their description

Definition 3 (efficient function family)

An ensemble of function families $\mathcal{F} = \{\mathcal{F}_n\}_{n\in\mathbb{N}}$ is efficient, if:

Ensembles of Function Families

- $\mathcal{F} = \{\mathcal{F}_n\}_{n \in \mathbb{N}}$, where $\mathcal{F}_n = \{f \colon \{0,1\}^{m(n)} \mapsto \{0,1\}^{\ell(n)}\}$
  (we simply write $\mathcal{F} = \{\mathcal{F}_n \colon \{0,1\}^{m(n)} \mapsto \{0,1\}^{\ell(n)}\}$)
- We identify functions with their description

Definition 3 (efficient function family)

An ensemble of function families $\mathcal{F} = \{\mathcal{F}_n\}_{n \in \mathbb{N}}$ is efficient, if:

Samplable. $\exists$ PPT that given 1^n, outputs (the description of) a uniform element in $\mathcal{F}_n$.

Ensembles of Function Families

- $\mathcal{F} = \{\mathcal{F}_n\}_{n\in\mathbb{N}}$, where $\mathcal{F}_n = \{f\colon \{0,1\}^{m(n)} \mapsto \{0,1\}^{\ell(n)}\}$

  (we simply write $\mathcal{F} = \{\mathcal{F}_n\colon \{0,1\}^{m(n)} \mapsto \{0,1\}^{\ell(n)}\}$)
- We identify functions with their description

Definition 3 (efficient function family)

An ensemble of function families $\mathcal{F} = \{\mathcal{F}_n\}_{n\in\mathbb{N}}$ is efficient, if:

Samplable. $\exists$ PPT that given 1^n, outputs (the description of) a uniform element in $\mathcal{F}_n$.

Efficient. $\exists$ poly-time algorithm that given $x \in \{0,1\}^{m(n)}$ and (a description of) $f \in \mathcal{F}_n$, outputs $f(x)$.

Pseudorandom Functions

Definition 4 (pseudorandom functions (PRFs))

An efficient function family ensemble $\mathcal{F} = \{\mathcal{F}_n \colon \{0,1\}^{m(n)} \mapsto \{0,1\}^{\ell(n)}\}$ is pseudorandom, if

$$\left|\Pr[\mathsf{D}^{\mathcal{F}_n}(1^n) = 1] - \Pr[\mathsf{D}^{\Pi_{m(n),\ell(n)}}(1^n) = 1\right| = \operatorname{neg}(n),$$

for any oracle-aided PPT D.

Pseudorandom Functions

Definition 4 (pseudorandom functions (PRFs))

An efficient function family ensemble $\mathcal{F} = \{\mathcal{F}_n \colon \{0,1\}^{m(n)} \mapsto \{0,1\}^{\ell(n)}\}$ is pseudorandom, if

$$\left|\Pr[\mathsf{D}^{\mathcal{F}_n}(1^n) = 1] - \Pr[\mathsf{D}^{\Pi_{m(n),\ell(n)}}(1^n) = 1\right| = \operatorname{neg}(n),$$

for any oracle-aided PPT D.

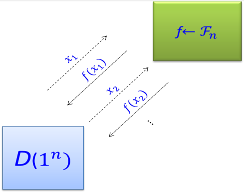
Pseudorandom Functions

Definition 4 (pseudorandom functions (PRFs))

An efficient function family ensemble $\mathcal{F} = \{\mathcal{F}_n \colon \{0,1\}^{m(n)} \mapsto \{0,1\}^{\ell(n)}\}$ is pseudorandom, if

$$\left|\Pr[\mathsf{D}^{\mathcal{F}_n}(1^n) = 1] - \Pr[\mathsf{D}^{\Pi_{m(n),\ell(n)}}(1^n) = 1\right| = \operatorname{neg}(n),$$

for any oracle-aided PPT D.



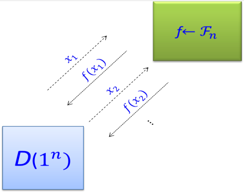
Pseudorandom Functions

Definition 4 (pseudorandom functions (PRFs))

An efficient function family ensemble $\mathcal{F} = \{\mathcal{F}_n \colon \{0,1\}^{m(n)} \mapsto \{0,1\}^{\ell(n)}\}$ is pseudorandom, if

$$\left|\Pr[\mathsf{D}^{\mathcal{F}_n}(1^n) = 1] - \Pr[\mathsf{D}^{\Pi_{m(n),\ell(n)}}(1^n) = 1\right| = \operatorname{neg}(n),$$

for any oracle-aided PPT D.



$\approx_C$

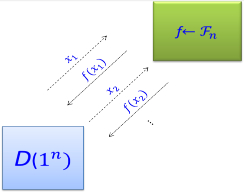
Pseudorandom Functions

Definition 4 (pseudorandom functions (PRFs))

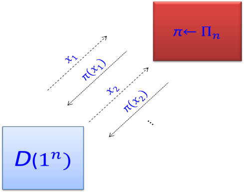
An efficient function family ensemble $\mathcal{F} = \{\mathcal{F}_n \colon \{0,1\}^{m(n)} \mapsto \{0,1\}^{\ell(n)}\}$ is pseudorandom, if

$$\left|\Pr[\mathsf{D}^{\mathcal{F}_n}(1^n) = 1] - \Pr[\mathsf{D}^{\Pi_{m(n),\ell(n)}}(1^n) = 1\right| = \operatorname{neg}(n),$$

for any oracle-aided PPT D.



$\approx_C$



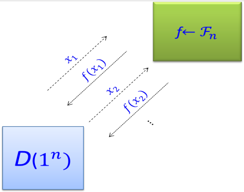
Pseudorandom Functions

Definition 4 (pseudorandom functions (PRFs))

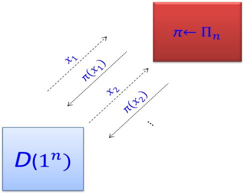
An efficient function family ensemble $\mathcal{F} = \{\mathcal{F}_n \colon \{0,1\}^{m(n)} \mapsto \{0,1\}^{\ell(n)}\}$ is pseudorandom, if

$$\left|\Pr[\mathsf{D}^{\mathcal{F}_n}(1^n) = 1] - \Pr[\mathsf{D}^{\Pi_{m(n),\ell(n)}}(1^n) = 1\right| = \operatorname{neg}(n),$$

for any oracle-aided PPT D.



$\approx_C$



- Why "oracle-aided"?

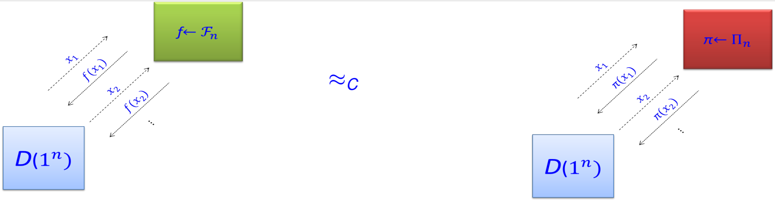
Pseudorandom Functions

Definition 4 (pseudorandom functions (PRFs))

An efficient function family ensemble $\mathcal{F} = \{\mathcal{F}_n \colon \{0,1\}^{m(n)} \mapsto \{0,1\}^{\ell(n)}\}$ is pseudorandom, if

$$\left|\Pr[\mathsf{D}^{\mathcal{F}_n}(1^n) = 1] - \Pr[\mathsf{D}^{\Pi_{m(n),\ell(n)}}(1^n) = 1\right| = \operatorname{neg}(n),$$

for any oracle-aided PPT D.



- Why "oracle-aided"?
- Easy to construct (no assumption!) with logarithmic input length

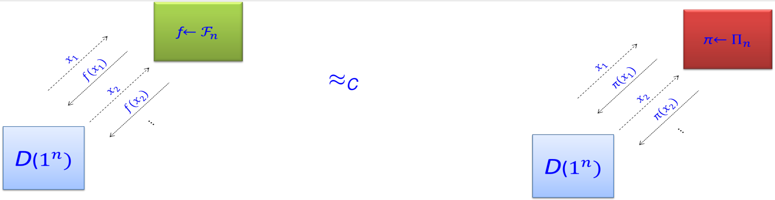
Pseudorandom Functions

Definition 4 (pseudorandom functions (PRFs))

An efficient function family ensemble $\mathcal{F} = \{\mathcal{F}_n \colon \{0,1\}^{m(n)} \mapsto \{0,1\}^{\ell(n)}\}$ is pseudorandom, if

$$\left|\Pr[\mathsf{D}^{\mathcal{F}_n}(1^n) = 1] - \Pr[\mathsf{D}^{\Pi_{m(n),\ell(n)}}(1^n) = 1\right| = \operatorname{neg}(n),$$

for any oracle-aided PPT D.



- Why "oracle-aided"?
- Easy to construct (no assumption!) with logarithmic input length
- PRFs of super logarithmic input length, which is the interesting case, imply PRGs

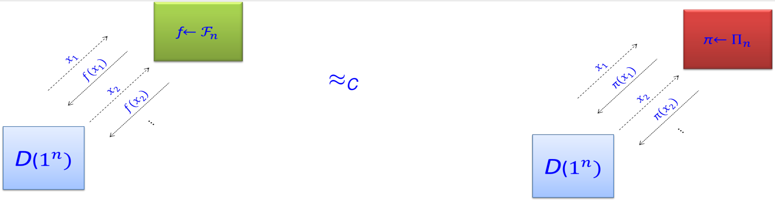
Pseudorandom Functions

Definition 4 (pseudorandom functions (PRFs))

An efficient function family ensemble $\mathcal{F} = \{\mathcal{F}_n \colon \{0,1\}^{m(n)} \mapsto \{0,1\}^{\ell(n)}\}$ is pseudorandom, if

$$\left|\Pr[\mathsf{D}^{\mathcal{F}_n}(1^n) = 1] - \Pr[\mathsf{D}^{\Pi_{m(n),\ell(n)}}(1^n) = 1\right| = \operatorname{neg}(n),$$

for any oracle-aided PPT D.



- Why “oracle-aided"?
- Easy to construct (no assumption!) with logarithmic input length
- PRFs of super logarithmic input length, which is the interesting case, imply PRGs
- We will mainly focus on the case $m(n) = \ell(n) = n$

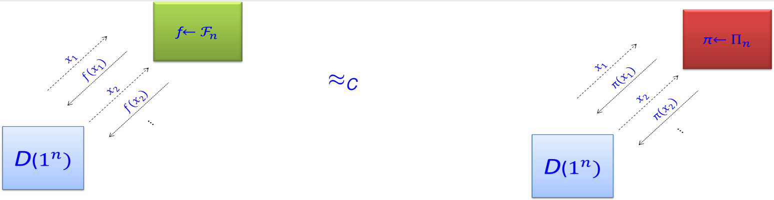
Pseudorandom Functions

Definition 4 (pseudorandom functions (PRFs))

An efficient function family ensemble $\mathcal{F} = \{\mathcal{F}_n \colon \{0,1\}^{m(n)} \mapsto \{0,1\}^{\ell(n)}\}$ is pseudorandom, if

$$\left|\Pr[\mathsf{D}^{\mathcal{F}_n}(1^n) = 1] - \Pr[\mathsf{D}^{\Pi_{m(n),\ell(n)}}(1^n) = 1\right| = \operatorname{neg}(n),$$

for any oracle-aided PPT D.



- Why “oracle-aided"?
- Easy to construct (no assumption!) with logarithmic input length
- PRFs of super logarithmic input length, which is the interesting case, imply PRGs
- We will mainly focus on the case $m(n) = \ell(n) = n$
- Main application: design a scheme assuming that you have random functions, and the realize them using PRFs.

Section 3

Pseudorandom Functions from One-Way Functions

Naive Construction

Let $G\colon \{0,1\}^n \mapsto \{0,1\}^{2n}$, and for $s \in \{0,1\}^n$ define $f_s\colon \{0,1\} \mapsto \{0,1\}^n$ by

- $f_s(0) = G(s)_{1,\ldots,n}$
- $f_s(1) = G(s)_{n_1,\ldots,2n}$.

Naive Construction

Let $G\colon \{0,1\}^n \mapsto \{0,1\}^{2n}$, and for $s \in \{0,1\}^n$ define $f_s\colon \{0,1\} \mapsto \{0,1\}^n$ by

- $f_s(0) = G(s)_{1,\ldots,n}$
- $f_s(1) = G(s)_{n_1,\ldots,2n}$.

Claim 5

Assume G is a PRG, then $\{\mathcal{F}_n = \{f_s\}_{s\in\{0,1\}^n}\}_{n\in\mathbb{N}}$ is a PRF.

Naive Construction

Let $G\colon \{0,1\}^n \mapsto \{0,1\}^{2n}$, and for $s \in \{0,1\}^n$ define $f_s\colon \{0,1\} \mapsto \{0,1\}^n$ by

- $f_s(0) = G(s)_{1,\ldots,n}$
- $f_s(1) = G(s)_{n_1,\ldots,2n}$.

Claim 5

Assume G is a PRG, then $\{\mathcal{F}_n = \{f_s\}_{s\in\{0,1\}^n}\}_{n\in\mathbb{N}}$ is a PRF.

Proof:

Naive Construction

Let $G\colon \{0,1\}^n \mapsto \{0,1\}^{2n}$, and for $s \in \{0,1\}^n$ define $f_s\colon \{0,1\} \mapsto \{0,1\}^n$ by

- $f_s(0) = G(s)_{1,\ldots,n}$
- $f_s(1) = G(s)_{n_1,\ldots,2n}$.

Claim 5

Assume G is a PRG, then $\{\mathcal{F}_n = \{f_s\}_{s\in\{0,1\}^n}\}_{n\in\mathbb{N}}$ is a PRF.

Proof: The truth table of $f \stackrel{R}{\leftarrow} \mathcal{F}_n$ is $G(U_n)$, where the truth table of $\pi \stackrel{R}{\leftarrow} \Pi_{1,n}$ is $U_{2n}\Box$

Naive Construction

Let $G\colon \{0,1\}^n \mapsto \{0,1\}^{2n}$, and for $s \in \{0,1\}^n$ define $f_s\colon \{0,1\} \mapsto \{0,1\}^n$ by

- $f_s(0) = G(s)_{1,\ldots,n}$
- $f_s(1) = G(s)_{n_1,\ldots,2n}$.

Claim 5

Assume G is a PRG, then $\{\mathcal{F}_n = \{f_s\}_{s\in\{0,1\}^n}\}_{n\in\mathbb{N}}$ is a PRF.

Proof: The truth table of $f \xleftarrow{R} \mathcal{F}_n$ is $G(U_n)$, where the truth table of $\pi \xleftarrow{R} \Pi_{1,n}$ is $U_{2n}\square$

- Naturally extends to input of length $O(\log n)$:-)

Naive Construction

Let $G\colon \{0,1\}^n \mapsto \{0,1\}^{2n}$, and for $s \in \{0,1\}^n$ define $f_s\colon \{0,1\} \mapsto \{0,1\}^n$ by

- $f_s(0) = G(s)_{1,\dots,n}$
- $f_s(1) = G(s)_{n_1,\dots,2n}$.

Claim 5

Assume G is a PRG, then $\{\mathcal{F}_n = \{f_s\}_{s\in\{0,1\}^n}\}_{n\in\mathbb{N}}$ is a PRF.

Proof: The truth table of $f \stackrel{R}{\leftarrow} \mathcal{F}_n$ is $G(U_n)$, where the truth table of $\pi \stackrel{R}{\leftarrow} \Pi_{1,n}$ is U_{2n} $\square$

- Naturally extends to input of length $O(\log n)$:-)
- Miserably fails for longer length (which is the only interesting case) :-(

Naive Construction

Let $G\colon \{0,1\}^n \mapsto \{0,1\}^{2n}$, and for $s \in \{0,1\}^n$ define $f_s\colon \{0,1\} \mapsto \{0,1\}^n$ by

- $f_s(0) = G(s)_{1,\ldots,n}$
- $f_s(1) = G(s)_{n_1,\ldots,2n}$.

Claim 5

Assume G is a PRG, then $\{\mathcal{F}_n = \{f_s\}_{s\in\{0,1\}^n}\}_{n\in\mathbb{N}}$ is a PRF.

Proof: The truth table of $f \stackrel{R}{\leftarrow} \mathcal{F}_n$ is $G(U_n)$, where the truth table of $\pi \stackrel{R}{\leftarrow} \Pi_{1,n}$ is $U_{2n}\square$

- Naturally extends to input of length $O(\log n)$:-)
- Miserably fails for longer length (which is the only interesting case) :-(
- Problem, we are constructing the whole truth table, even to compute a single output

The GGM Construction

Construction 6 (GGM)

For $G\colon \{0,1\}^n \mapsto \{0,1\}^{2n}$ and $s \in \{0,1\}^n$,

- $G_0(s) = G(s)_{1,\ldots,n}$
- $G_1(s) = G(s)_{n+1,\ldots,2n}$

For $x \in \{0,1\}^k$ let $f_s(x) = G_{x_k}(f_s(x_{1,\ldots,k-1}))$, letting $f_s() = s$.

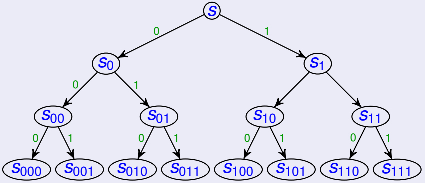
The GGM Construction

Construction 6 (GGM)

For $G\colon \{0,1\}^n \mapsto \{0,1\}^{2n}$ and $s \in \{0,1\}^n$,

- $G_0(s) = G(s)_{1,\ldots,n}$
- $G_1(s) = G(s)_{n+1,\ldots,2n}$

For $x \in \{0,1\}^k$ let $f_s(x) = G_{x_k}(f_s(x_{1,\ldots,k-1}))$, letting $f_s() = s$.



$s_x = f_s(x)$

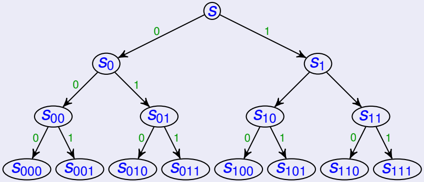
The GGM Construction

Construction 6 (GGM)

For $G\colon \{0,1\}^n \mapsto \{0,1\}^{2n}$ and $s \in \{0,1\}^n$,

- $G_0(s) = G(s)_{1,\ldots,n}$
- $G_1(s) = G(s)_{n+1,\ldots,2n}$

For $x \in \{0,1\}^k$ let $f_s(x) = G_{x_k}(f_s(x_{1,\ldots,k-1}))$, letting $f_s() = s$.



$s_x = f_s(x)$

- Example: $f_s(001) = s_{001} = G_1(s_{00}) = G_1(G_0(s_0)) = G_1(G_0(G_0(s)))$

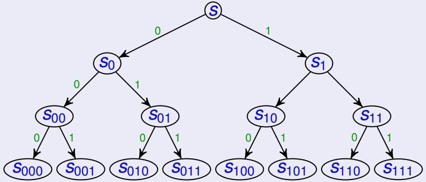
The GGM Construction

Construction 6 (GGM)

For $G\colon \{0,1\}^n \mapsto \{0,1\}^{2n}$ and $s \in \{0,1\}^n$,

- $G_0(s) = G(s)_{1,\ldots,n}$
- $G_1(s) = G(s)_{n+1,\ldots,2n}$

For $x \in \{0,1\}^k$ let $f_s(x) = G_{x_k}(f_s(x_{1,\ldots,k-1}))$, letting $f_s() = s$.



$s_x = f_s(x)$

- Example: $f_s(001) = s_{001} = G_1(s_{00}) = G_1(G_0(s_0)) = G_1(G_0(G_0(s)))$
- G is poly-time $\implies \mathcal{F} := \{\mathcal{F}_n = \{f_s\colon s \in \{0,1\}^n\}\}$ is efficient

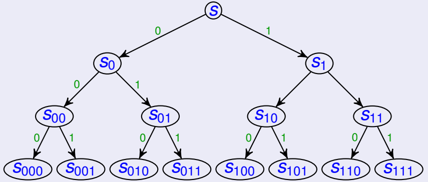
The GGM Construction

Construction 6 (GGM)

For $G\colon \{0,1\}^n \mapsto \{0,1\}^{2n}$ and $s \in \{0,1\}^n$,

- $G_0(s) = G(s)_{1,\ldots,n}$
- $G_1(s) = G(s)_{n+1,\ldots,2n}$

For $x \in \{0,1\}^k$ let $f_s(x) = G_{x_k}(f_s(x_{1,\ldots,k-1}))$, letting $f_s() = s$.



$s_x = f_s(x)$

- Example: $f_s(001) = s_{001} = G_1(s_{00}) = G_1(G_0(s_0)) = G_1(G_0(G_0(s)))$
- G is poly-time $\implies \mathcal{F} := \{\mathcal{F}_n = \{f_s \colon s \in \{0,1\}^n\}\}$ is efficient

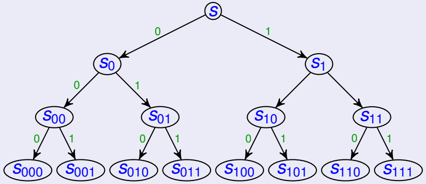
The GGM Construction

Construction 6 (GGM)

For $G\colon \{0,1\}^n \mapsto \{0,1\}^{2n}$ and $s \in \{0,1\}^n$,

- $G_0(s) = G(s)_{1,\ldots,n}$
- $G_1(s) = G(s)_{n+1,\ldots,2n}$

For $x \in \{0,1\}^k$ let $f_s(x) = G_{x_k}(f_s(x_{1,\ldots,k-1}))$, letting $f_s() = s$.



$s_x = f_s(x)$

- Example: $f_s(001) = s_{001} = G_1(s_{00}) = G_1(G_0(s_0)) = G_1(G_0(G_0(s)))$
- G is poly-time $\implies \mathcal{F} := \{\mathcal{F}_n = \{f_s\colon s \in \{0,1\}^n\}\}$ is efficient

Theorem 7 (Goldreich-Goldwasser-Micali (GGM))

If G is a PRG then $\mathcal{F}$ is a PRF.

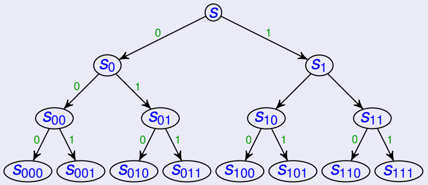
The GGM Construction

Construction 6 (GGM)

For $G\colon \{0,1\}^n \mapsto \{0,1\}^{2n}$ and $s \in \{0,1\}^n$,

- $G_0(s) = G(s)_{1,\ldots,n}$
- $G_1(s) = G(s)_{n+1,\ldots,2n}$

For $x \in \{0,1\}^k$ let $f_s(x) = G_{x_k}(f_s(x_{1,\ldots,k-1}))$, letting $f_s() = s$.



$s_x = f_s(x)$

- Example: $f_s(001) = s_{001} = G_1(s_{00}) = G_1(G_0(s_0)) = G_1(G_0(G_0(s)))$
- G is poly-time $\implies \mathcal{F} := \{\mathcal{F}_n = \{f_s\colon s \in \{0,1\}^n\}\}$ is efficient

Theorem 7 (Goldreich-Goldwasser-Micali (GGM))

If G is a PRG then $\mathcal{F}$ is a PRF.

Corollary 8

OWFs imply PRFs.

Proof Idea

Assume $\exists$ PPT D, $p \in$ poly and infinite set $\mathcal{I} \subseteq \mathbb{N}$ with

$$\left|\Pr[\mathsf{D}^{F_n}(1^n) = 1] - \Pr[\mathsf{D}^{\Pi_n}(1^n) = 1]\right| \geq \frac{1}{p(n)}, \tag{1}$$

for any $n \in \mathcal{I}$.

Proof Idea

Assume $\exists$ PPT D, $p \in \text{poly}$ and infinite set $\mathcal{I} \subseteq \mathbb{N}$ with

$$\left|\Pr[\mathsf{D}^{F_n}(1^n) = 1] - \Pr[\mathsf{D}^{\Pi_n}(1^n) = 1]\right| \geq \frac{1}{p(n)}, \tag{1}$$

for any $n \in \mathcal{I}$.

Fix $n \in \mathbb{N}$ and let $t = t(n)$ be a bound on the running time of $\mathsf{D}(1^n)$.

Proof Idea

Assume $\exists$ PPT D, $p \in$ poly and infinite set $\mathcal{I} \subseteq \mathbb{N}$ with

$$\left|\Pr[\mathsf{D}^{F_n}(1^n) = 1] - \Pr[\mathsf{D}^{\Pi_n}(1^n) = 1]\right| \geq \frac{1}{p(n)}, \tag{1}$$

for any $n \in \mathcal{I}$.

Fix $n \in \mathbb{N}$ and let $t = t(n)$ be a bound on the running time of $\mathsf{D}(1^n)$. We use D to construct a PPT D′ such that

$$\left|\Pr[\mathsf{D}'((U_{2n})^t) = 1] - \Pr[\mathsf{D}'(G(U_n))^t) = 1\right| > \frac{1}{np(n)},$$

where $(U_{2n})^t = U_{2n}^{(1)}, \ldots, U_{2n}^{(t)}$ and $G(U_n)^t = G(U_n^{(1)}), \ldots, G(U_n^{(t)})$.

Proof Idea

Assume $\exists$ PPT D, $p \in$ poly and infinite set $\mathcal{I} \subseteq \mathbb{N}$ with

$$\left|\Pr[\mathsf{D}^{F_n}(1^n) = 1] - \Pr[\mathsf{D}^{\Pi_n}(1^n) = 1]\right| \geq \frac{1}{p(n)}, \tag{1}$$

for any $n \in \mathcal{I}$.

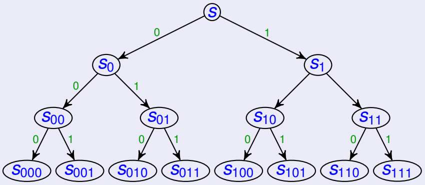
Fix $n \in \mathbb{N}$ and let $t = t(n)$ be a bound on the running time of $\mathsf{D}(1^n)$. We use D to construct a PPT D′ such that

$$\left|\Pr[\mathsf{D}'((U_{2n})^t) = 1] - \Pr[\mathsf{D}'(G(U_n))^t) = 1\right| > \frac{1}{np(n)},$$

where $(U_{2n})^t = U_{2n}^{(1)}, \ldots, U_{2n}^{(t)}$ and $G(U_n)^t = G(U_n^{(1)}), \ldots, G(U_n^{(t)})$.

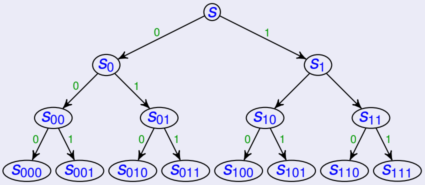
Hence, D′ violates the security of G

The Hybrid



$s_x = f_s(x)$

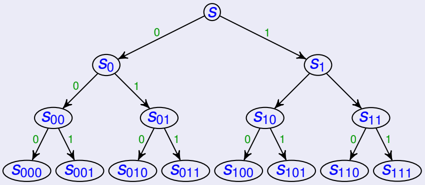
The Hybrid



$s_x = f_s(x)$

- Let $\mathcal{T}_i$ be the set of all possible trees, in which the $i+1,\ldots,n$ levels are obtained by "applying GGM" to the i'th level.

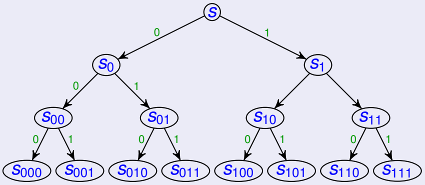
The Hybrid



$s_x = f_s(x)$

- Let $\mathcal{T}_i$ be the set of all possible trees, in which the $i+1,\ldots,n$ levels are obtained by "applying GGM" to the i'th level.
- Given a tree t, let $h_t(x)$ return the x'th leaf of t.

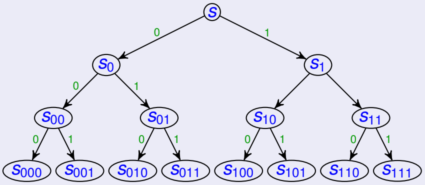
The Hybrid



$s_x = f_s(x)$

- Let $\mathcal{T}_i$ be the set of all possible trees, in which the $i+1,\ldots,n$ levels are obtained by "applying GGM" to the i'th level.
- Given a tree t, let $h_t(x)$ return the x'th leaf of t.
- What family is $\mathcal{H}_1 = \{h_t\}_{t \in \mathcal{T}_1}$?

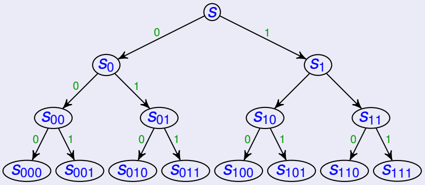
The Hybrid



$s_x = f_s(x)$

- Let $\mathcal{T}_i$ be the set of all possible trees, in which the $i+1,\ldots,n$ levels are obtained by “applying GGM" to the i'th level.
- Given a tree t, let $h_t(x)$ return the x'th leaf of t.
- What family is $\mathcal{H}_1 = \{h_t\}_{t \in \mathcal{T}_1}$?

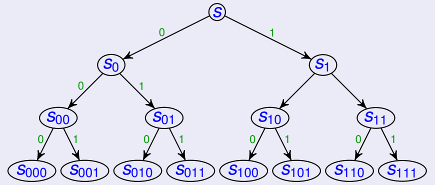
The Hybrid



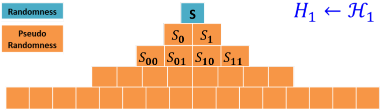
$s_x = f_s(x)$

- Let $\mathcal{T}_i$ be the set of all possible trees, in which the $i+1,\ldots,n$ levels are obtained by "applying GGM" to the i'th level.
- Given a tree t, let $h_t(x)$ return the x'th leaf of t.
- What family is $\mathcal{H}_1 = \{h_t\}_{t\in\mathcal{T}_1}$? $\mathcal{F}_n$.

The Hybrid



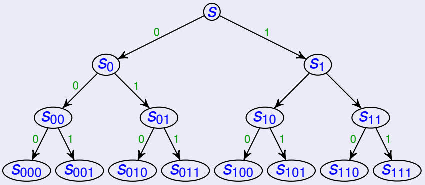
$s_x = f_s(x)$



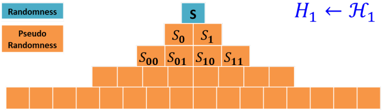
$H_1 \leftarrow \mathcal{H}_1$

- Let $\mathcal{T}_i$ be the set of all possible trees, in which the $i+1, \ldots, n$ levels are obtained by "applying GGM" to the i'th level.
- Given a tree t, let $h_t(x)$ return the x'th leaf of t.
- What family is $\mathcal{H}_1 = \{h_t\}_{t \in \mathcal{T}_1}$? $\mathcal{F}_n$.

The Hybrid

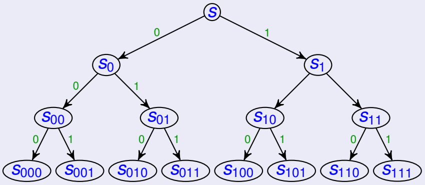


$s_x = f_s(x)$

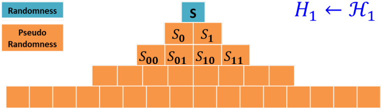


- Let $\mathcal{T}_i$ be the set of all possible trees, in which the $i+1,\ldots,n$ levels are obtained by "applying GGM" to the i'th level.
- Given a tree t, let $h_t(x)$ return the x'th leaf of t.
- What family is $\mathcal{H}_1 = \{h_t\}_{t \in \mathcal{T}_1}$? $\mathcal{F}_n$. What is $\mathcal{H}_n$?

The Hybrid



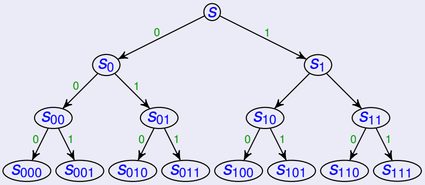
$s_x = f_s(x)$



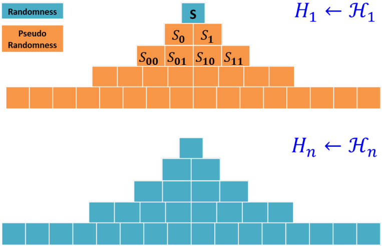
$H_1 \leftarrow \mathcal{H}_1$

- Let $\mathcal{T}_i$ be the set of all possible trees, in which the $i+1,\ldots,n$ levels are obtained by "applying GGM" to the i'th level.
- Given a tree t, let $h_t(x)$ return the x'th leaf of t.
- What family is $\mathcal{H}_1 = \{h_t\}_{t \in \mathcal{T}_1}$? $\mathcal{F}_n$. What is $\mathcal{H}_n$? Π_n.

The Hybrid

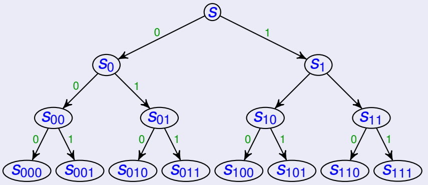


$s_x = f_s(x)$

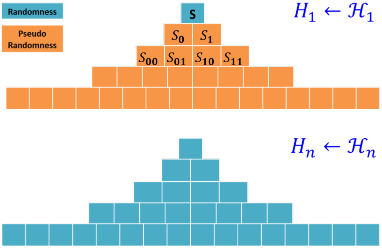


- Let $\mathcal{T}_i$ be the set of all possible trees, in which the $i+1,\ldots,n$ levels are obtained by "applying GGM" to the i'th level.
- Given a tree t, let $h_t(x)$ return the x'th leaf of t.
- What family is $\mathcal{H}_1 = \{h_t\}_{t \in \mathcal{T}_1}$? $\mathcal{F}_n$. What is $\mathcal{H}_n$? Π_n.

The Hybrid



$s_x = f_s(x)$



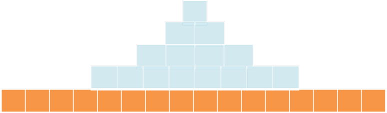
- Let $\mathcal{T}_i$ be the set of all possible trees, in which the $i+1,\ldots,n$ levels are obtained by “applying GGM" to the i’th level.
- Given a tree t, let $h_t(x)$ return the x’th leaf of t.
- What family is $\mathcal{H}_1 = \{h_t\}_{t\in\mathcal{T}_1}$? $\mathcal{F}_n$. What is $\mathcal{H}_n$? Π_n.
- For some $i \in \{1,\ldots,i-1\}$, algorithm D distinguishes $\mathcal{H}_i$ from $\mathcal{H}_{i+1}$ by $\frac{1}{np(n)}$

The Hybrid cont.

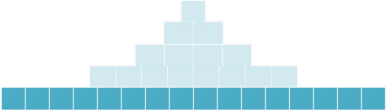
We assume wlg. that D distinguishes between $\mathcal{H}_{n-1}$ and $\mathcal{H}_n$ (can we?)

The Hybrid cont.

We assume wlg. that D distinguishes between $\mathcal{H}_{n-1}$ and $\mathcal{H}_n$ (can we?)



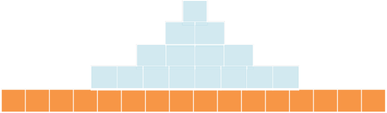
$\not\approx$



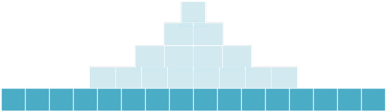
- D distinguishes (via t samples) between
  - R – a uniform string of length $2^n \cdot n$, and
  - P - a string generated by 2^{n-1} independent calls to G

The Hybrid cont.

We assume wlg. that D distinguishes between $\mathcal{H}_{n-1}$ and $\mathcal{H}_n$ (can we?)



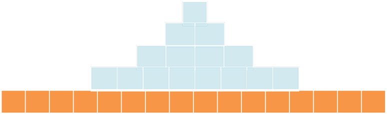
$\not\approx$



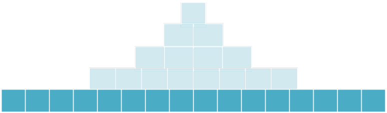
- D distinguishes (via t samples) between
  - R – a uniform string of length $2^n \cdot n$, and
  - P - a string generated by 2^{n-1} independent calls to G
- We would like to use D for breaking the security of G,

The Hybrid cont.

We assume wlg. that D distinguishes between $\mathcal{H}_{n-1}$ and $\mathcal{H}_n$ (can we?)



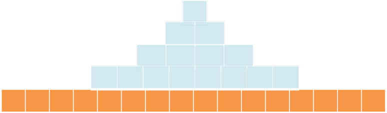
$\not\approx$



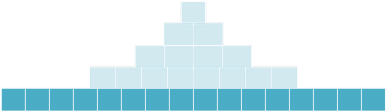
- D distinguishes (via t samples) between
  - R – a uniform string of length $2^n \cdot n$, and
  - P - a string generated by 2^{n-1} independent calls to G
- We would like to use D for breaking the security of G, but R and P seem too long :-(

The Hybrid cont.

We assume wlg. that D distinguishes between $\mathcal{H}_{n-1}$ and $\mathcal{H}_n$ (can we?)



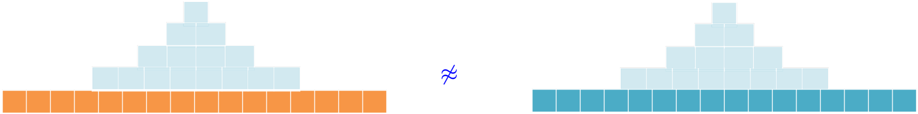
$\not\approx$



- D distinguishes (via t samples) between
  - R – a uniform string of length $2^n \cdot n$, and
  - P - a string generated by 2^{n-1} independent calls to G
- We would like to use D for breaking the security of G, but R and P seem too long :-(
- Solution: focus on the part (i.e., cells) that D *sees*

The Hybrid cont.

We assume wlg. that D distinguishes between $\mathcal{H}_{n-1}$ and $\mathcal{H}_n$ (can we?)



- D distinguishes (via t samples) between
  - R – a uniform string of length $2^n \cdot n$, and
  - P - a string generated by 2^{n-1} independent calls to G
- We would like to use D for breaking the security of G, but R and P seem too long :-(
- Solution: focus on the part (i.e., cells) that D *sees*

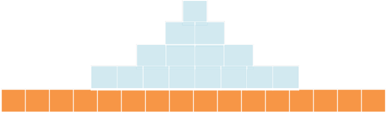
Algorithm 9 (D′ on $y_1, \ldots, y_t \in (\{0,1\}^{2n})^t$)

Emulate D. On the i'th query q_i made by D:

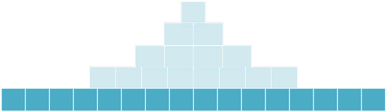
- If the cell queries by q_i'th is empty, fill it with the next y
- Answer with the content of the q_i'th cell.

The Hybrid cont.

We assume wlg. that D distinguishes between $\mathcal{H}_{n-1}$ and $\mathcal{H}_n$ (can we?)



$\not\approx$

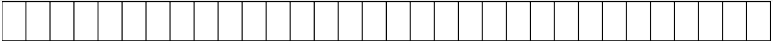


- D distinguishes (via t samples) between
  - R – a uniform string of length $2^n \cdot n$, and
  - P - a string generated by 2^{n-1} independent calls to G
- We would like to use D for breaking the security of G, but R and P seem too long :-(
- Solution: focus on the part (i.e., cells) that D *sees*

Algorithm 9 (D′ on $y_1, \ldots, y_t \in (\{0,1\}^{2n})^t$)

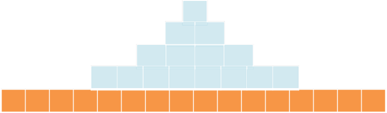
Emulate D. On the i'th query q_i made by D:

- If the cell queries by q_i'th is empty, fill it with the next y
- Answer with the content of the q_i'th cell.

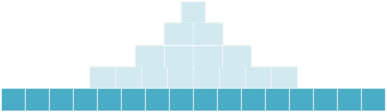


The Hybrid cont.

We assume wlg. that D distinguishes between $\mathcal{H}_{n-1}$ and $\mathcal{H}_n$ (can we?)



$\not\approx$

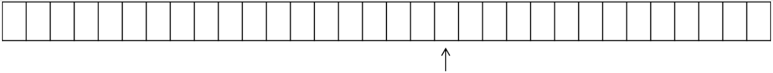


- D distinguishes (via t samples) between
  - R – a uniform string of length $2^n \cdot n$, and
  - P - a string generated by 2^{n-1} independent calls to G
- We would like to use D for breaking the security of G, but R and P seem too long :-(
- Solution: focus on the part (i.e., cells) that D *sees*

Algorithm 9 (D′ on $y_1, \ldots, y_t \in (\{0,1\}^{2n})^t$)

Emulate D. On the i'th query q_i made by D:

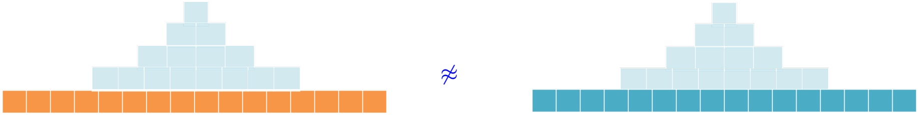
- If the cell queries by q_i'th is empty, fill it with the next y
- Answer with the content of the q_i'th cell.



q_1

The Hybrid cont.

We assume wlg. that D distinguishes between $\mathcal{H}_{n-1}$ and $\mathcal{H}_n$ (can we?)

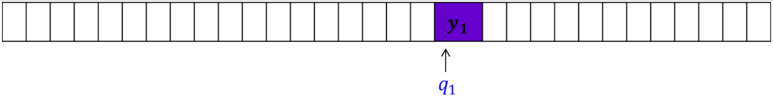


- D distinguishes (via t samples) between
  - R – a uniform string of length $2^n \cdot n$, and
  - P - a string generated by 2^{n-1} independent calls to G
- We would like to use D for breaking the security of G, but R and P seem too long :-(
- Solution: focus on the part (i.e., cells) that D *sees*

Algorithm 9 (D′ on $y_1, \ldots, y_t \in (\{0,1\}^{2n})^t$)

Emulate D. On the i'th query q_i made by D:

- If the cell queries by q_i'th is empty, fill it with the next y
- Answer with the content of the q_i'th cell.



The Hybrid cont.

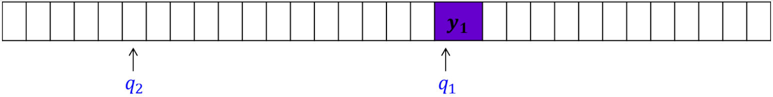
We assume wlg. that D distinguishes between $\mathcal{H}_{n-1}$ and $\mathcal{H}_n$ (can we?)

- D distinguishes (via t samples) between
  - R – a uniform string of length $2^n \cdot n$, and
  - P - a string generated by 2^{n-1} independent calls to G
- We would like to use D for breaking the security of G, but R and P seem too long :-(
- Solution: focus on the part (i.e., cells) that D *sees*

Algorithm 9 (D′ on $y_1, \ldots, y_t \in (\{0,1\}^{2n})^t$)

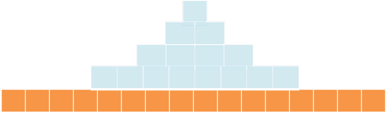
Emulate D. On the i'th query q_i made by D:

- If the cell queries by q_i'th is empty, fill it with the next y
- Answer with the content of the q_i'th cell.

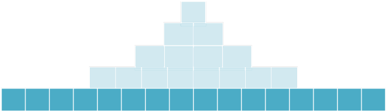


The Hybrid cont.

We assume wlg. that D distinguishes between $\mathcal{H}_{n-1}$ and $\mathcal{H}_n$ (can we?)



$\not\approx$

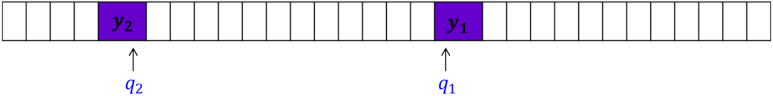


- D distinguishes (via t samples) between
  - R – a uniform string of length $2^n \cdot n$, and
  - P - a string generated by 2^{n-1} independent calls to G
- We would like to use D for breaking the security of G, but R and P seem too long :-(
- Solution: focus on the part (i.e., cells) that D *sees*

Algorithm 9 (D′ on $y_1, \ldots, y_t \in (\{0,1\}^{2n})^t$)

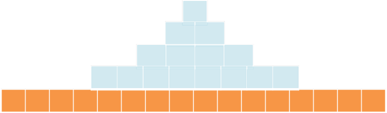
Emulate D. On the i'th query q_i made by D:

- If the cell queries by q_i'th is empty, fill it with the next y
- Answer with the content of the q_i'th cell.

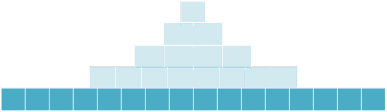


The Hybrid cont.

We assume wlg. that D distinguishes between $\mathcal{H}_{n-1}$ and $\mathcal{H}_n$ (can we?)



$\not\approx$

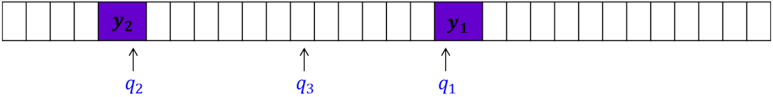


- D distinguishes (via t samples) between
  - R – a uniform string of length $2^n \cdot n$, and
  - P - a string generated by 2^{n-1} independent calls to G
- We would like to use D for breaking the security of G, but R and P seem too long :-(
- Solution: focus on the part (i.e., cells) that D *sees*

Algorithm 9 (D′ on $y_1, \ldots, y_t \in (\{0,1\}^{2n})^t$)

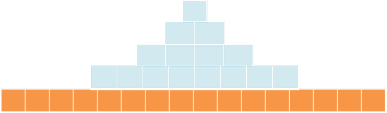
Emulate D.   On the i'th query q_i made by D:

- If the cell queries by q_i'th is empty, fill it with the next y
- Answer with the content of the q_i'th cell.

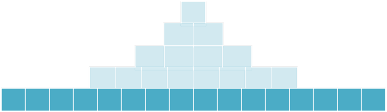


The Hybrid cont.

We assume wlg. that D distinguishes between $\mathcal{H}_{n-1}$ and $\mathcal{H}_n$ (can we?)



$\not\approx$

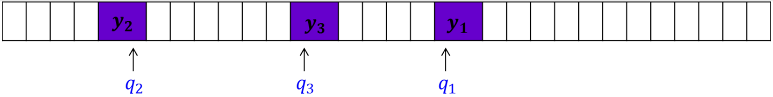


- D distinguishes (via t samples) between
  - R – a uniform string of length $2^n \cdot n$, and
  - P - a string generated by 2^{n-1} independent calls to G
- We would like to use D for breaking the security of G, but R and P seem too long :-(
- Solution: focus on the part (i.e., cells) that D *sees*

Algorithm 9 (D′ on $y_1, \ldots, y_t \in (\{0,1\}^{2n})^t$)

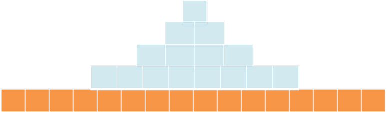
Emulate D. On the i'th query q_i made by D:

- If the cell queries by q_i'th is empty, fill it with the next y
- Answer with the content of the q_i'th cell.

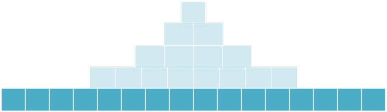


The Hybrid cont.

We assume wlg. that D distinguishes between $\mathcal{H}_{n-1}$ and $\mathcal{H}_n$ (can we?)



$\not\approx$

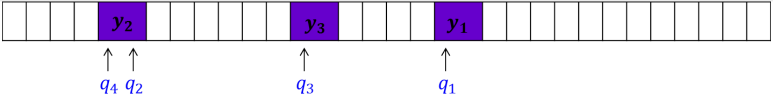


- D distinguishes (via t samples) between
  - R – a uniform string of length $2^n \cdot n$, and
  - P - a string generated by 2^{n-1} independent calls to G
- We would like to use D for breaking the security of G, but R and P seem too long :-(
- Solution: focus on the part (i.e., cells) that D *sees*

Algorithm 9 (D′ on $y_1, \ldots, y_t \in (\{0,1\}^{2n})^t$)

Emulate D. On the i'th query q_i made by D:

- If the cell queries by q_i'th is empty, fill it with the next y
- Answer with the content of the q_i'th cell.



The Hybrid cont.

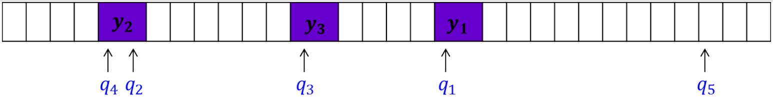
We assume wlg. that D distinguishes between $\mathcal{H}_{n-1}$ and $\mathcal{H}_n$ (can we?)

- D distinguishes (via t samples) between
  - R – a uniform string of length $2^n \cdot n$, and
  - P - a string generated by 2^{n-1} independent calls to G
- We would like to use D for breaking the security of G, but R and P seem too long :-(
- Solution: focus on the part (i.e., cells) that D *sees*

Algorithm 9 (D′ on $y_1, \ldots, y_t \in (\{0,1\}^{2n})^t$)

Emulate D. On the i'th query q_i made by D:

- If the cell queries by q_i'th is empty, fill it with the next y
- Answer with the content of the q_i'th cell.



The Hybrid cont.

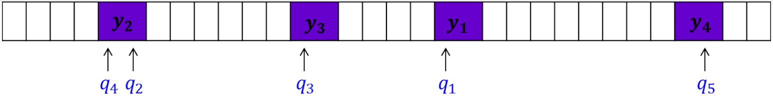
We assume wlg. that D distinguishes between $\mathcal{H}_{n-1}$ and $\mathcal{H}_n$ (can we?)

- D distinguishes (via t samples) between
  - R – a uniform string of length $2^n \cdot n$, and
  - P - a string generated by 2^{n-1} independent calls to G
- We would like to use D for breaking the security of G, but R and P seem too long :-(
- Solution: focus on the part (i.e., cells) that D *sees*

Algorithm 9 (D′ on $y_1, \ldots, y_t \in (\{0,1\}^{2n})^t$)

Emulate D. On the i'th query q_i made by D:

- If the cell queries by q_i'th is empty, fill it with the next y
- Answer with the content of the q_i'th cell.



The Hybrid cont.

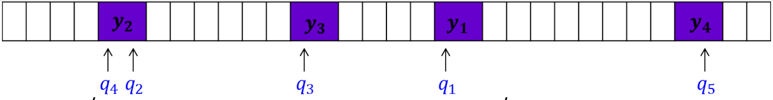
We assume wlg. that D distinguishes between $\mathcal{H}_{n-1}$ and $\mathcal{H}_n$ (can we?)

- D distinguishes (via t samples) between
  - R – a uniform string of length $2^n \cdot n$, and
  - P - a string generated by 2^{n-1} independent calls to G
- We would like to use D for breaking the security of G, but R and P seem too long :-(
- Solution: focus on the part (i.e., cells) that D *sees*

Algorithm 9 (D′ on $y_1, \ldots, y_t \in (\{0,1\}^{2n})^t$)

Emulate D. On the i'th query q_i made by D:

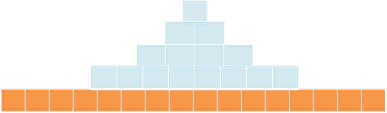
- If the cell queries by q_i'th is empty, fill it with the next y
- Answer with the content of the q_i'th cell.



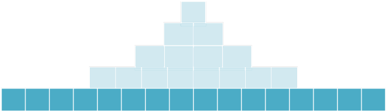
- $\mathsf{D}'((U_{2n})^t)$ / $\mathsf{D}'(G(U_n))^t)$ emulates D with access to R / P

The Hybrid cont.

We assume wlg. that D distinguishes between $\mathcal{H}_{n-1}$ and $\mathcal{H}_n$ (can we?)



$\not\approx$

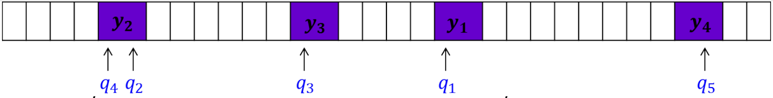


- D distinguishes (via t samples) between
  - R – a uniform string of length $2^n \cdot n$, and
  - P - a string generated by 2^{n-1} independent calls to G
- We would like to use D for breaking the security of G, but R and P seem too long :-(
- Solution: focus on the part (i.e., cells) that D *sees*

Algorithm 9 (D′ on $y_1,\ldots,y_t \in (\{0,1\}^{2n})^t$)

Emulate D. On the i'th query q_i made by D:

- If the cell queries by q_i'th is empty, fill it with the next y
- Answer with the content of the q_i'th cell.



- $D'((U_{2n})^t)$ / $D'(G(U_n))^t)$ emulates D with access to R / P
- Hence, $\left|\Pr[D'((U_{2n})^t) = 1] - \Pr[D'(G(U_n))^t) = 1\right| > \frac{1}{np(n)}$

Part II

Pseudorandom Permutations

Formal Definition

Let $\widetilde{\Pi}_n$ be the set of all permutations over $\{0,1\}^n$.

Definition 10 (pseudorandom permutations (PRPs))

A permutation ensemble $\mathcal{F} = \{\mathcal{F}_n : \{0,1\}^n \mapsto \{0,1\}^n\}$ is a pseudorandom permutation, if

$$\left|\Pr[\mathsf{D}^{\mathcal{F}_n}(1^n) = 1] - \Pr[\mathsf{D}^{\widetilde{\Pi}_n}(1^n) = 1\right| = \operatorname{neg}(n), \tag{2}$$

for any oracle-aided PPT D

Formal Definition

Let $\widetilde{\Pi}_n$ be the set of all permutations over $\{0,1\}^n$.

Definition 10 (pseudorandom permutations (PRPs))

A permutation ensemble $\mathcal{F} = \{\mathcal{F}_n : \{0,1\}^n \mapsto \{0,1\}^n\}$ is a pseudorandom permutation, if

$$\left|\Pr[\mathsf{D}^{\mathcal{F}_n}(1^n) = 1] - \Pr[\mathsf{D}^{\widetilde{\Pi}_n}(1^n) = 1\right| = \operatorname{neg}(n), \tag{2}$$

for any oracle-aided PPT D

- Eq 2 holds for any PRF (taking the role of $\mathcal{F}$)

Formal Definition

Let $\widetilde{\Pi}_n$ be the set of all permutations over $\{0,1\}^n$.

Definition 10 (pseudorandom permutations (PRPs))

A permutation ensemble $\mathcal{F} = \{\mathcal{F}_n : \{0,1\}^n \mapsto \{0,1\}^n\}$ is a pseudorandom permutation, if

$$\left|\Pr[\mathsf{D}^{\mathcal{F}_n}(1^n) = 1] - \Pr[\mathsf{D}^{\widetilde{\Pi}_n}(1^n) = 1\right| = \operatorname{neg}(n), \tag{2}$$

for any oracle-aided PPT D

- Eq 2 holds for any PRF (taking the role of $\mathcal{F}$)
- Hence, PRPs are indistinguishable from PRFs...

Formal Definition

Let $\widetilde{\Pi}_n$ be the set of all permutations over $\{0,1\}^n$.

Definition 10 (pseudorandom permutations (PRPs))

A permutation ensemble $\mathcal{F} = \{\mathcal{F}_n \colon \{0,1\}^n \mapsto \{0,1\}^n\}$ is a pseudorandom permutation, if

$$\left|\Pr[\mathsf{D}^{\mathcal{F}_n}(1^n) = 1] - \Pr[\mathsf{D}^{\widetilde{\Pi}_n}(1^n) = 1\right| = \operatorname{neg}(n), \qquad (2)$$

for any oracle-aided PPT D

- Eq 2 holds for any PRF (taking the role of $\mathcal{F}$)
- Hence, PRPs are indistinguishable from PRFs...
- If no one can distinguish between PRFs and PRPs, let's use PRFs

Formal Definition

Let $\widetilde{\Pi}_n$ be the set of all permutations over $\{0,1\}^n$.

Definition 10 (pseudorandom permutations (PRPs))

A permutation ensemble $\mathcal{F} = \{\mathcal{F}_n : \{0,1\}^n \mapsto \{0,1\}^n\}$ is a pseudorandom permutation, if

$$\left|\Pr[\mathsf{D}^{\mathcal{F}_n}(1^n) = 1] - \Pr[\mathsf{D}^{\widetilde{\Pi}_n}(1^n) = 1\right| = \operatorname{neg}(n), \tag{2}$$

for any oracle-aided PPT D

- Eq 2 holds for any PRF (taking the role of $\mathcal{F}$)
- Hence, PRPs are indistinguishable from PRFs...
- If no one can distinguish between PRFs and PRPs, let's use PRFs
  - (partial) Perfect "security"

Formal Definition

Let $\widetilde{\Pi}_n$ be the set of all permutations over $\{0,1\}^n$.

Definition 10 (pseudorandom permutations (PRPs))

A permutation ensemble $\mathcal{F} = \{\mathcal{F}_n \colon \{0,1\}^n \mapsto \{0,1\}^n\}$ is a pseudorandom permutation, if

$$\left|\Pr[\mathsf{D}^{\mathcal{F}_n}(1^n) = 1] - \Pr[\mathsf{D}^{\widetilde{\Pi}_n}(1^n) = 1\right| = \operatorname{neg}(n), \tag{2}$$

for any oracle-aided PPT D

- Eq 2 holds for any PRF (taking the role of $\mathcal{F}$)
- Hence, PRPs are indistinguishable from PRFs...
- If no one can distinguish between PRFs and PRPs, let's use PRFs
  - (partial) Perfect “security"
  - Inversion

Section 4

PRP from PRF

Feistel Permutation

How does one turn a function into a permutation?

Feistel Permutation

How does one turn a function into a permutation?

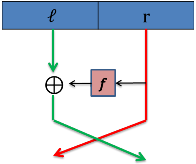
Definition 11 (LR)

For $f \colon \{0,1\}^n \mapsto \{0,1\}^n$, let $\mathsf{LR}_f \colon \{0,1\}^{2n} \mapsto \{0,1\}^{2n}$ be defined by

$$\mathsf{LR}_f(\ell, r) = (r, f(r) \oplus \ell).$$

Feistel Permutation

How does one turn a function into a permutation?



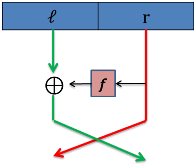
Definition 11 (LR)

For $f\colon \{0,1\}^n \mapsto \{0,1\}^n$, let $\mathrm{LR}_f\colon \{0,1\}^{2n} \mapsto \{0,1\}^{2n}$ be defined by

$$\mathrm{LR}_f(\ell, r) = (r, f(r) \oplus \ell).$$

Feistel Permutation

How does one turn a function into a permutation?



Definition 11 (LR)

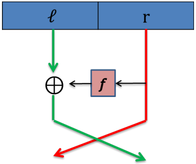
For $f\colon \{0,1\}^n \mapsto \{0,1\}^n$, let $\mathrm{LR}_f\colon \{0,1\}^{2n} \mapsto \{0,1\}^{2n}$ be defined by

$$\mathrm{LR}_f(\ell, r) = (r, f(r) \oplus \ell).$$

- LR_f is a permutation: $\mathrm{LR}_f^{-1}(z, w) = (f(z) \oplus w, z)$

Feistel Permutation

How does one turn a function into a permutation?



Definition 11 (LR)

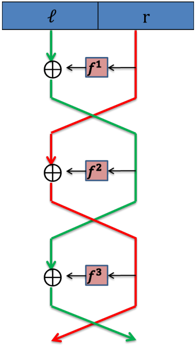
For $f\colon \{0,1\}^n \mapsto \{0,1\}^n$, let $\mathrm{LR}_f\colon \{0,1\}^{2n} \mapsto \{0,1\}^{2n}$ be defined by

$$\mathrm{LR}_f(\ell, r) = (r, f(r) \oplus \ell).$$

- LR_f is a permutation: $\mathrm{LR}_f^{-1}(z, w) = (f(z) \oplus w, z)$
- LR_f is efficiently computable and invertible given oracle access to f

Feistel Permutation

How does one turn a function into a permutation?



Definition 11 (LR)

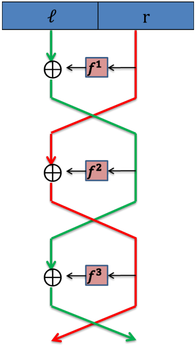
For $f\colon \{0,1\}^n \mapsto \{0,1\}^n$, let $\mathrm{LR}_f\colon \{0,1\}^{2n} \mapsto \{0,1\}^{2n}$ be defined by

$$\mathrm{LR}_f(\ell, r) = (r, f(r) \oplus \ell).$$

- LR_f is a permutation: $\mathrm{LR}_f^{-1}(z, w) = (f(z) \oplus w, z)$
- LR_f is efficiently computable and invertible given oracle access to f

Feistel Permutation

How does one turn a function into a permutation?



Definition 11 (LR)

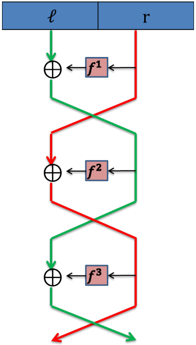
For $f\colon \{0,1\}^n \mapsto \{0,1\}^n$, let $\mathrm{LR}_f\colon \{0,1\}^{2n} \mapsto \{0,1\}^{2n}$ be defined by

$$\mathrm{LR}_f(\ell, r) = (r, f(r) \oplus \ell).$$

- LR_f is a permutation: $\mathrm{LR}_f^{-1}(z, w) = (f(z) \oplus w, z)$
- LR_f is efficiently computable and invertible given oracle access to f
- For $i \in \mathbb{N}$ and $f^1, \ldots, f^i$, define $\mathrm{LR}_{f^1,\ldots,f^i}\colon \{0,1\}^{2n} \mapsto \{0,1\}^{2n}$ by

Feistel Permutation

How does one turn a function into a permutation?



Definition 11 (LR)

For $f\colon \{0,1\}^n \mapsto \{0,1\}^n$, let $\mathrm{LR}_f\colon \{0,1\}^{2n} \mapsto \{0,1\}^{2n}$ be defined by

$$\mathrm{LR}_f(\ell, r) = (r, f(r) \oplus \ell).$$

- LR_f is a permutation: $\mathrm{LR}_f^{-1}(z, w) = (f(z) \oplus w, z)$
- LR_f is efficiently computable and invertible given oracle access to f

- For $i \in \mathbb{N}$ and $f^1, \ldots, f^i$, define $\mathrm{LR}_{f^1,\ldots,f^i}\colon \{0,1\}^{2n} \mapsto \{0,1\}^{2n}$ by

  $\mathrm{LR}_{f^1,\ldots,f^i}(\ell, r) = (r^{i-1}, f^i(r^{i-1}) \oplus \ell^{i-1})$, for $(\ell^{i-1}, r^{i-1}) = \mathrm{LR}_{f^1,\ldots,f^{i-1}}(\ell, r)$.
  (letting LR_γ be the identity function)

Luby-Rackoff Thm.

Recall $\mathrm{LR}_f(\ell, r) = (r, f(r) \oplus \ell)$.

Luby-Rackoff Thm.

Recall $\mathrm{LR}_f(\ell, r) = (r, f(r) \oplus \ell)$.

Definition 12

Given a function family $\mathcal{F} = \{\mathcal{F}_n \colon \{0,1\}^n \mapsto \{0,1\}^n\}$, let
$\mathrm{LR}^i(\mathcal{F}) = \{\mathrm{LR}^i_{\mathcal{F}_n} = \{\mathrm{LR}_{f^1,\ldots,f^i} \colon f^1, \ldots, f^i \in \mathcal{F}_n\}\}$,

Luby-Rackoff Thm.

Recall $\mathrm{LR}_f(\ell, r) = (r, f(r) \oplus \ell)$.

Definition 12

Given a function family $\mathcal{F} = \{\mathcal{F}_n \colon \{0,1\}^n \mapsto \{0,1\}^n\}$, let $\mathrm{LR}^i(\mathcal{F}) = \{\mathrm{LR}^i_{\mathcal{F}_n} = \{\mathrm{LR}_{f^1,\ldots,f^i} \colon f^1, \ldots, f^i \in \mathcal{F}_n\}\}$,

- $\mathrm{LR}^i_{\mathcal{F}}$ is always a permutation family, and is efficient if $\mathcal{F}$ is.

Luby-Rackoff Thm.

Recall $\mathrm{LR}_f(\ell, r) = (r, f(r) \oplus \ell)$.

Definition 12

Given a function family $\mathcal{F} = \{\mathcal{F}_n \colon \{0,1\}^n \mapsto \{0,1\}^n\}$, let $\mathrm{LR}^i(\mathcal{F}) = \{\mathrm{LR}^i_{\mathcal{F}_n} = \{\mathrm{LR}_{f^1,\ldots,f^i} \colon f^1, \ldots, f^i \in \mathcal{F}_n\}\}$,

- $\mathrm{LR}^i_{\mathcal{F}}$ is always a permutation family, and is efficient if $\mathcal{F}$ is.
- Is $\mathrm{LR}^1_{\mathcal{F}}$ pseudorandom?

Luby-Rackoff Thm.

Recall $\mathrm{LR}_f(\ell, r) = (r, f(r) \oplus \ell)$.

Definition 12

Given a function family $\mathcal{F} = \{\mathcal{F}_n \colon \{0,1\}^n \mapsto \{0,1\}^n\}$, let $\mathrm{LR}^i(\mathcal{F}) = \{\mathrm{LR}^i_{\mathcal{F}_n} = \{\mathrm{LR}_{f^1,\ldots,f^i} \colon f^1, \ldots, f^i \in \mathcal{F}_n\}\}$,

- $\mathrm{LR}^i_{\mathcal{F}}$ is always a permutation family, and is efficient if $\mathcal{F}$ is.
- Is $\mathrm{LR}^1_{\mathcal{F}}$ pseudorandom?
- $\mathrm{LR}^2_{\mathcal{F}}$?

Luby-Rackoff Thm.

Recall $\mathsf{LR}_f(\ell, r) = (r, f(r) \oplus \ell)$.

Definition 12

Given a function family $\mathcal{F} = \{\mathcal{F}_n \colon \{0,1\}^n \mapsto \{0,1\}^n\}$, let
$\mathsf{LR}^i(\mathcal{F}) = \{\mathsf{LR}^i_{\mathcal{F}_n} = \{\mathsf{LR}_{f^1,\ldots,f^i} \colon f^1, \ldots, f^i \in \mathcal{F}_n\}\}$,

- $\mathsf{LR}^i_{\mathcal{F}}$ is always a permutation family, and is efficient if $\mathcal{F}$ is.
- Is $\mathsf{LR}^1_{\mathcal{F}}$ pseudorandom?
- $\mathsf{LR}^2_{\mathcal{F}}$?

Luby-Rackoff Thm.

Recall $\mathrm{LR}_f(\ell, r) = (r, f(r) \oplus \ell)$.

Definition 12

Given a function family $\mathcal{F} = \{\mathcal{F}_n \colon \{0,1\}^n \mapsto \{0,1\}^n\}$, let $\mathrm{LR}^i(\mathcal{F}) = \{\mathrm{LR}^i_{\mathcal{F}_n} = \{\mathrm{LR}_{f^1,\ldots,f^i} \colon f^1, \ldots, f^i \in \mathcal{F}_n\}\}$,

- $\mathrm{LR}^i_{\mathcal{F}}$ is always a permutation family, and is efficient if $\mathcal{F}$ is.
- Is $\mathrm{LR}^1_{\mathcal{F}}$ pseudorandom?
- $\mathrm{LR}^2_{\mathcal{F}}$? $\mathrm{LR}_{f^1,f^2}(0^n, 0^n) = \mathrm{LR}_{f^2}(0^n, f^1(0^n)) = (f^1(0^n), \cdot)$
  and $\mathrm{LR}_{f^1,f^2} = \mathrm{LR}_{f^2}(0^n, f^1(0^n) \oplus 1^n) = (f^1(0^n) \oplus 1^n, \cdot)$

Luby-Rackoff Thm.

Recall $\mathrm{LR}_f(\ell, r) = (r, f(r) \oplus \ell)$.

Definition 12

Given a function family $\mathcal{F} = \{\mathcal{F}_n \colon \{0,1\}^n \mapsto \{0,1\}^n\}$, let $\mathrm{LR}^i(\mathcal{F}) = \{\mathrm{LR}^i_{\mathcal{F}_n} = \{\mathrm{LR}_{f^1,\ldots,f^i} \colon f^1, \ldots, f^i \in \mathcal{F}_n\}\}$,

- $\mathrm{LR}^i_{\mathcal{F}}$ is always a permutation family, and is efficient if $\mathcal{F}$ is.
- Is $\mathrm{LR}^1_{\mathcal{F}}$ pseudorandom?
- $\mathrm{LR}^2_{\mathcal{F}}$? $\mathrm{LR}_{f^1,f^2}(0^n, 0^n) = \mathrm{LR}_{f^2}(0^n, f^1(0^n)) = (f^1(0^n), \cdot)$
  and $\mathrm{LR}_{f^1,f^2} = \mathrm{LR}_{f^2}(0^n, f^1(0^n) \oplus 1^n) = (f^1(0^n) \oplus 1^n, \cdot)$
- $\mathrm{LR}^3_{\mathcal{F}}$?

Luby-Rackoff Thm.

Recall $\mathrm{LR}_f(\ell, r) = (r, f(r) \oplus \ell)$.

Definition 12

Given a function family $\mathcal{F} = \{\mathcal{F}_n \colon \{0,1\}^n \mapsto \{0,1\}^n\}$, let $\mathrm{LR}^i(\mathcal{F}) = \{\mathrm{LR}^i_{\mathcal{F}_n} = \{\mathrm{LR}_{f^1,\ldots,f^i} \colon f^1, \ldots, f^i \in \mathcal{F}_n\}\}$,

- $\mathrm{LR}^i_{\mathcal{F}}$ is always a permutation family, and is efficient if $\mathcal{F}$ is.
- Is $\mathrm{LR}^1_{\mathcal{F}}$ pseudorandom?
- $\mathrm{LR}^2_{\mathcal{F}}$? $\mathrm{LR}_{f^1,f^2}(0^n, 0^n) = \mathrm{LR}_{f^2}(0^n, f^1(0^n)) = (f^1(0^n), \cdot)$
  and $\mathrm{LR}_{f^1,f^2} = \mathrm{LR}_{f^2}(0^n, f^1(0^n) \oplus 1^n) = (f^1(0^n) \oplus 1^n, \cdot)$
- $\mathrm{LR}^3_{\mathcal{F}}$?

Luby-Rackoff Thm.

Recall $\mathsf{LR}_f(\ell, r) = (r, f(r) \oplus \ell)$.

Definition 12

Given a function family $\mathcal{F} = \{\mathcal{F}_n \colon \{0,1\}^n \mapsto \{0,1\}^n\}$, let $\mathsf{LR}^i(\mathcal{F}) = \{\mathsf{LR}^i_{\mathcal{F}_n} = \{\mathsf{LR}_{f^1,\ldots,f^i} \colon f^1, \ldots, f^i \in \mathcal{F}_n\}\}$,

- $\mathsf{LR}^i_{\mathcal{F}}$ is always a permutation family, and is efficient if $\mathcal{F}$ is.
- Is $\mathsf{LR}^1_{\mathcal{F}}$ pseudorandom?
- $\mathsf{LR}^2_{\mathcal{F}}$? $\mathsf{LR}_{f^1,f^2}(0^n, 0^n) = \mathsf{LR}_{f^2}(0^n, f^1(0^n)) = (f^1(0^n), \cdot)$
  and $\mathsf{LR}_{f^1,f^2} = \mathsf{LR}_{f^2}(0^n, f^1(0^n) \oplus 1^n) = (f^1(0^n) \oplus 1^n, \cdot)$
- $\mathsf{LR}^3_{\mathcal{F}}$?

Theorem 13 (Luby-Rackoff)

Assuming that $\mathcal{F}$ is a PRF, then $\mathsf{LR}^3_{\mathcal{F}}$ is a PRP

Luby-Rackoff Thm.

Recall $\mathrm{LR}_f(\ell, r) = (r, f(r) \oplus \ell)$.

Definition 12

Given a function family $\mathcal{F} = \{\mathcal{F}_n \colon \{0,1\}^n \mapsto \{0,1\}^n\}$, let
$\mathrm{LR}^i(\mathcal{F}) = \{\mathrm{LR}^i_{\mathcal{F}_n} = \{\mathrm{LR}_{f^1,\ldots,f^i} \colon f^1, \ldots, f^i \in \mathcal{F}_n\}\}$,

- $\mathrm{LR}^i_{\mathcal{F}}$ is always a permutation family, and is efficient if $\mathcal{F}$ is.
- Is $\mathrm{LR}^1_{\mathcal{F}}$ pseudorandom?
- $\mathrm{LR}^2_{\mathcal{F}}$? $\mathrm{LR}_{f^1,f^2}(0^n, 0^n) = \mathrm{LR}_{f^2}(0^n, f^1(0^n)) = (f^1(0^n), \cdot)$
  and $\mathrm{LR}_{f^1,f^2} = \mathrm{LR}_{f^2}(0^n, f^1(0^n) \oplus 1^n) = (f^1(0^n) \oplus 1^n, \cdot)$
- $\mathrm{LR}^3_{\mathcal{F}}$?

Theorem 13 (Luby-Rackoff)

Assuming that $\mathcal{F}$ is a PRF, then $\mathrm{LR}^3_{\mathcal{F}}$ is a PRP

- $\mathrm{LR}^4(\mathcal{F})$ is pseudorandom even if inversion queries are allowed

Proving Luby-Rackoff

Proving Luby-Rackoff

It suffices to prove that $\mathrm{LR}^3_{\Pi_n}$ is pseudorandom (why ?)

Proving Luby-Rackoff

It suffices to prove that $\mathrm{LR}^3_{\Pi_n}$ is pseudorandom (why ?)

- How would you prove that?

Proving Luby-Rackoff

It suffices to prove that $\mathrm{LR}^3_{\Pi_n}$ is pseudorandom (why ?)

- How would you prove that?
- Maybe $\mathrm{LR}^3(\Pi_n) \equiv \widetilde{\Pi}_{2n}$?

Proving Luby-Rackoff

It suffices to prove that $\mathsf{LR}^3_{\Pi_n}$ is pseudorandom (why ?)

- How would you prove that?
- Maybe $\mathsf{LR}^3(\Pi_n) \equiv \widetilde{\Pi}_{2n}$?

Proving Luby-Rackoff

It suffices to prove that $\mathsf{LR}^3_{\Pi_n}$ is pseudorandom (why ?)

- How would you prove that?
- Maybe $\mathsf{LR}^3(\Pi_n) \equiv \widetilde{\Pi}_{2n}$? description length of element in $\mathsf{LR}^3(\Pi_n)$ is $2^n \cdot n$, where that of element in $\widetilde{\Pi}_{2n}$ is

Proving Luby-Rackoff

It suffices to prove that $\mathsf{LR}^3_{\Pi_n}$ is pseudorandom (why ?)

- How would you prove that?
- Maybe $\mathsf{LR}^3(\Pi_n) \equiv \widetilde{\Pi}_{2n}$? description length of element in $\mathsf{LR}^3(\Pi_n)$ is $2^n \cdot n$, where that of element in $\widetilde{\Pi}_{2n}$ is $\log(2^{2n}!) > \log\left((\frac{2^{2n}}{e})^{2^{2n}}\right) > 2^{2n} \cdot n$

Proving Luby-Rackoff

It suffices to prove that $\mathsf{LR}^3_{\Pi_n}$ is pseudorandom (why ?)

- How would you prove that?
- Maybe $\mathsf{LR}^3(\Pi_n) \equiv \widetilde{\Pi}_{2n}$? description length of element in $\mathsf{LR}^3(\Pi_n)$ is $2^n \cdot n$, where that of element in $\widetilde{\Pi}_{2n}$ is $\log(2^{2n}!) > \log\left((\frac{2^{2n}}{e})^{2^{2n}}\right) > 2^{2n} \cdot n$

Claim 14

For any q-query D,

$$|\Pr[\mathsf{D}^{\mathsf{LR}^3(\Pi_n)}(1^n) = 1] - \Pr[\mathsf{D}^{\widetilde{\Pi}_{2n}}(1^n)| = 1] \in O(q^2/2^n).$$

Proving Luby-Rackoff

It suffices to prove that $\mathrm{LR}^3_{\Pi_n}$ is pseudorandom (why ?)

- How would you prove that?
- Maybe $\mathrm{LR}^3(\Pi_n) \equiv \widetilde{\Pi}_{2n}$? description length of element in $\mathrm{LR}^3(\Pi_n)$ is $2^n \cdot n$, where that of element in $\widetilde{\Pi}_{2n}$ is $\log(2^{2n}!) > \log\left((\frac{2^{2n}}{e})^{2^{2n}}\right) > 2^{2n} \cdot n$

Claim 14

For any q-query D,

$$|\Pr[\mathrm{D}^{\mathrm{LR}^3(\Pi_n)}(1^n) = 1] - \Pr[\mathrm{D}^{\widetilde{\Pi}_{2n}}(1^n)| = 1] \in O(q^2/2^n).$$

- We assume for simplicity that D is *deterministic*, *non-repeating* and *non-adaptive*.

Proving Luby-Rackoff

It suffices to prove that $\mathrm{LR}^3_{\Pi_n}$ is pseudorandom (why ?)

- How would you prove that?
- Maybe $\mathrm{LR}^3(\Pi_n) \equiv \widetilde{\Pi}_{2n}$? description length of element in $\mathrm{LR}^3(\Pi_n)$ is $2^n \cdot n$, where that of element in $\widetilde{\Pi}_{2n}$ is $\log(2^{2n}!) > \log\left((\frac{2^{2n}}{e})^{2^{2n}}\right) > 2^{2n} \cdot n$

Claim 14

For any q-query D,

$$|\Pr[\mathrm{D}^{\mathrm{LR}^3(\Pi_n)}(1^n) = 1] - \Pr[\mathrm{D}^{\widetilde{\Pi}_{2n}}(1^n)| = 1] \in O(q^2/2^n).$$

- We assume for simplicity that D is *deterministic*, *non-repeating* and *non-adaptive*.
- Let $x_0, x_1, \ldots, x_q$ be D's queries.

Proving Luby-Rackoff

It suffices to prove that $\mathrm{LR}^3_{\Pi_n}$ is pseudorandom (why ?)

- How would you prove that?
- Maybe $\mathrm{LR}^3(\Pi_n) \equiv \widetilde{\Pi}_{2n}$? description length of element in $\mathrm{LR}^3(\Pi_n)$ is $2^n \cdot n$, where that of element in $\widetilde{\Pi}_{2n}$ is $\log(2^{2n}!) > \log\left((\frac{2^{2n}}{e})^{2^{2n}}\right) > 2^{2n} \cdot n$

Claim 14

For any q-query D,

$$|\Pr[\mathrm{D}^{\mathrm{LR}^3(\Pi_n)}(1^n) = 1] - \Pr[\mathrm{D}^{\widetilde{\Pi}_{2n}}(1^n)| = 1] \in O(q^2/2^n).$$

- We assume for simplicity that D is *deterministic*, *non-repeating* and *non-adaptive*.
- Let $x_0, x_1, \ldots, x_q$ be D's queries.
- We show $(f(x_0), \ldots, f(x_q))_{f \stackrel{R}{\leftarrow} \mathrm{LR}^3(\Pi_n)}$ is $O(q^2/2^n)$ close (i.e., in statistical distance) to $(f(x_0), \ldots, f(x_q))_{f \stackrel{R}{\leftarrow} \widetilde{\Pi}}$

Proving Luby-Rackoff

It suffices to prove that $\mathrm{LR}^3_{\Pi_n}$ is pseudorandom (why ?)

- How would you prove that?
- Maybe $\mathrm{LR}^3(\Pi_n) \equiv \widetilde{\Pi}_{2n}$? description length of element in $\mathrm{LR}^3(\Pi_n)$ is $2^n \cdot n$, where that of element in $\widetilde{\Pi}_{2n}$ is $\log(2^{2n}!) > \log\left((\frac{2^{2n}}{e})^{2^{2n}}\right) > 2^{2n} \cdot n$

Claim 14

For any q-query D,

$$|\Pr[\mathrm{D}^{\mathrm{LR}^3(\Pi_n)}(1^n) = 1] - \Pr[\mathrm{D}^{\widetilde{\Pi}_{2n}}(1^n)| = 1] \in O(q^2/2^n).$$

- We assume for simplicity that D is *deterministic*, *non-repeating* and *non-adaptive*.
- Let $x_0, x_1, \ldots, x_q$ be D's queries.
- We show $(f(x_0), \ldots, f(x_q))_{f \xleftarrow{R} \mathrm{LR}^3(\Pi_n)}$ is $O(q^2/2^n)$ close (i.e., in statistical distance) to $(f(x_0), \ldots, f(x_q))_{f \xleftarrow{R} \widetilde{\Pi}}$
- To do that, we show both distributions are $O(q^2/2^n)$ close to $\mathit{Distinct} := \left((z_1, \ldots z_q) \xleftarrow{R} (\{0,1\}^{2n})^q \mid \forall i \neq j \colon (z_i)_0 \neq (z_j)_0\right)$.

Reminder: Statistical Distance

Definition 15

The statistical distance between distributions P and Q, is defined by

$$\mathrm{SD}(P, Q) = \frac{1}{2} \cdot \sum_{u \in \mathcal{U}} |P(u) - Q(u)|$$

Reminder: Statistical Distance

Definition 15

The statistical distance between distributions P and Q, is defined by

$$\mathsf{SD}(P, Q) = \frac{1}{2} \cdot \sum_{u \in \mathcal{U}} |P(u) - Q(u)|$$

Fact 16

$\mathsf{SD}(P, Q) = \max_{\mathsf{A}}\{\Pr_{u \xleftarrow{R} P}[\mathsf{A}(u) = 1] - \Pr_{u \xleftarrow{R} Q}[\mathsf{A}(u) = 1]\}$, *where* $\max$ *is over all possible algorithms.*

Reminder: Statistical Distance

Definition 15

The statistical distance between distributions P and Q, is defined by

$$\mathsf{SD}(P,Q) = \frac{1}{2} \cdot \sum_{u \in \mathcal{U}} |P(u) - Q(u)|$$

Fact 16

$\mathsf{SD}(P,Q) = \max_{\mathsf{A}}\{\Pr_{u \xleftarrow{R} P}[\mathsf{A}(u) = 1] - \Pr_{u \xleftarrow{R} Q}[\mathsf{A}(u) = 1]\}$, *where* max *is over* *all possible algorithms.*

Namely, statistical distance is the analogue of computational distance, when we remove the efficiency restriction from the distinguisher.

Reminder: Statistical Distance

Definition 15

The statistical distance between distributions P and Q, is defined by

$$\mathrm{SD}(P,Q) = \frac{1}{2} \cdot \sum_{u \in \mathcal{U}} |P(u) - Q(u)|$$

Fact 16

$\mathrm{SD}(P,Q) = \max_{\mathsf{A}}\{\Pr_{u \xleftarrow{R} P}[\mathsf{A}(u) = 1] - \Pr_{u \xleftarrow{R} Q}[\mathsf{A}(u) = 1]\}$, *where* max *is over all possible algorithms.*

Namely, statistical distance is the analogue of computational distance, when we remove the efficiency restriction from the distinguisher.

In case $\mathrm{SD}(P,Q) \leq \varepsilon$, we say that P and Q are ε close.

Reminder: Statistical Distance

Definition 15

The statistical distance between distributions P and Q, is defined by

$$\mathsf{SD}(P,Q) = \frac{1}{2} \cdot \sum_{u \in \mathcal{U}} |P(u) - Q(u)|$$

Fact 16

$\mathsf{SD}(P,Q) = \max_{\mathsf{A}}\{\Pr_{u \xleftarrow{R} P}[\mathsf{A}(u) = 1] - \Pr_{u \xleftarrow{R} Q}[\mathsf{A}(u) = 1]\}$, *where* $\max$ *is over all possible algorithms.*

Namely, statistical distance is the analogue of computational distance, when we remove the efficiency restriction from the distinguisher.

In case $\mathsf{SD}(P,Q) \leq \varepsilon$, we say that P and Q are ε close.

Fact 17

Assume $\mathsf{SD}(P|\neg E, Q) \leq \delta_1$ *and* $\Pr_P[E] \leq \delta_2$, *then* $\mathsf{SD}(P,Q) \leq \delta_1 + \delta_2$

$(f(x_0), \ldots, f(x_q))_{f \xleftarrow{R} \widetilde{\Pi}}$ is close to *Distinct*

$(f(x_0),\ldots,f(x_q))_{f \stackrel{R}{\leftarrow} \widetilde{\Pi}}$ is close to *Distinct*

Recall *Distinct* $:= \left((z_1,\ldots z_q) \stackrel{R}{\leftarrow} (\{0,1\}^{2n})^q \mid \forall i \neq j \colon (z_i)_0 \neq (z_j)_0\right)$.

$(f(x_0),\ldots,f(x_q))_{f \xleftarrow{R} \widetilde{\Pi}}$ is close to *Distinct*

Recall *Distinct* $:= \left((z_1,\ldots z_q) \xleftarrow{R} (\{0,1\}^{2n})^q \mid \forall i \neq j \colon (z_i)_0 \neq (z_j)_0\right)$.

For $f \in \widetilde{\Pi}$, let $Bad(f) := \exists i \neq j \colon f(x_i)_0 = f(x_j)_0$.

$(f(x_0),\ldots,f(x_q))_{f \xleftarrow{R} \widetilde{\Pi}}$ is close to *Distinct*

Recall *Distinct* $:= \left((z_1,\ldots z_q) \xleftarrow{R} (\{0,1\}^{2n})^q \mid \forall i \neq j \colon (z_i)_0 \neq (z_j)_0\right)$.

For $f \in \widetilde{\Pi}$, let $Bad(f) := \exists i \neq j \colon f(x_i)_0 = f(x_j)_0$.

Claim 18

$\Pr_{f \xleftarrow{R} \widetilde{\Pi}}[Bad(f)] \leq \frac{\binom{q}{2}}{2^n} \leq \frac{q^2}{2^n}$

$\left(f(x_0),\ldots,f(x_q)\right)_{f \xleftarrow{R} \widetilde{\Pi}}$ is close to *Distinct*

Recall *Distinct* $:= \left((z_1,\ldots z_q) \xleftarrow{R} (\{0,1\}^{2n})^q \mid \forall i \neq j \colon (z_i)_0 \neq (z_j)_0\right)$.

For $f \in \widetilde{\Pi}$, let $Bad(f) := \exists i \neq j \colon f(x_i)_0 = f(x_j)_0$.

Claim 18

$\Pr_{f \xleftarrow{R} \widetilde{\Pi}}\left[Bad(f)\right] \leq \frac{\binom{q}{2}}{2^n} \leq \frac{q^2}{2^n}$

Proof: ?

$(f(x_0),\ldots,f(x_q))_{f \xleftarrow{R} \widetilde{\Pi}}$ is close to *Distinct*

Recall *Distinct* $:= \left((z_1,\ldots z_q) \xleftarrow{R} (\{0,1\}^{2n})^q \mid \forall i \neq j \colon (z_i)_0 \neq (z_j)_0\right)$.

For $f \in \widetilde{\Pi}$, let $Bad(f) := \exists i \neq j \colon f(x_i)_0 = f(x_j)_0$.

Claim 18

$\Pr_{f \xleftarrow{R} \widetilde{\Pi}}[Bad(f)] \leq \frac{\binom{q}{2}}{2^n} \leq \frac{q^2}{2^n}$

Proof: ?

Claim 19

$\left((f(x_0),\ldots,f(x_q)); f \xleftarrow{R} \widetilde{\Pi} \mid \neg\mathrm{Bad}(f)\right) \equiv$ *Distinct*

$\left(f(x_0),\ldots,f(x_q)\right)_{f \stackrel{R}{\leftarrow} \widetilde{\Pi}}$ is close to *Distinct*

Recall *Distinct* $:= \left((z_1,\ldots z_q) \stackrel{R}{\leftarrow} (\{0,1\}^{2n})^q \mid \forall i \neq j\colon (z_i)_0 \neq (z_j)_0\right)$.

For $f \in \widetilde{\Pi}$, let $Bad(f) := \exists i \neq j\colon f(x_i)_0 = f(x_j)_0$.

Claim 18

$\Pr_{f \stackrel{R}{\leftarrow} \widetilde{\Pi}}[Bad(f)] \leq \frac{\binom{q}{2}}{2^n} \leq \frac{q^2}{2^n}$

Proof: ?

Claim 19

$\left((f(x_0),\ldots,f(x_q)); f \stackrel{R}{\leftarrow} \widetilde{\Pi} \mid \neg \mathrm{Bad}(f)\right) \equiv Distinct$

Proof: ?

$\left(f(x_0),\ldots,f(x_q)\right)_{f \stackrel{R}{\leftarrow} \widetilde{\Pi}}$ is close to *Distinct*

Recall *Distinct* $:= \left((z_1,\ldots z_q) \stackrel{R}{\leftarrow} (\{0,1\}^{2n})^q \mid \forall i \neq j \colon (z_i)_0 \neq (z_j)_0\right)$.

For $f \in \widetilde{\Pi}$, let $Bad(f) := \exists i \neq j \colon f(x_i)_0 = f(x_j)_0$.

Claim 18

$\Pr_{f \stackrel{R}{\leftarrow} \widetilde{\Pi}}[Bad(f)] \leq \frac{\binom{q}{2}}{2^n} \leq \frac{q^2}{2^n}$

Proof: ?

Claim 19

$\left((f(x_0),\ldots,f(x_q)); f \stackrel{R}{\leftarrow} \widetilde{\Pi} \mid \neg \mathrm{Bad}(f)\right) \equiv Distinct$

Proof: ?

By Fact 17, $\left(f(x_0),\ldots,f(x_q)\right)_{f \stackrel{R}{\leftarrow} \widetilde{\Pi}}$ is $\frac{q^2}{2^n}$ close to *Distinct*

$(f(x_0),\ldots,f(x_q))_{f \xleftarrow{R} \mathrm{LR}^3(\Pi_n)}$ is close to *Distinct*

$(f(x_0), \ldots, f(x_q))_{f \stackrel{R}{\leftarrow} \mathrm{LR}^3(\Pi_n)}$ is close to *Distinct*

Let $(\ell_1^0, r_1^0), \ldots, (\ell_q^0, r_q^0) = (x_1, \ldots, x_k)$.

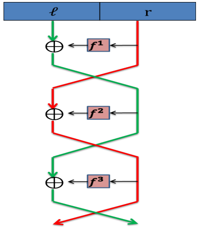
$(f(x_0),\ldots,f(x_q))_{f \xleftarrow{\mathrm{R}} \mathrm{LR}^3(\Pi_n)}$ is close to *Distinct*

Let $(\ell_1^0, r_1^0),\ldots,(\ell_q^0, r_q^0) = (x_1,\ldots,x_k)$.

The following rv's are defined w.r.t. $(f^1, f^2, f^3) \xleftarrow{\mathrm{R}} \Pi_n^3$.

ℓ_1^0	r_1^0	ℓ_2^0	r_2^0	$\ldots$	ℓ_q^0	r_q^0
ℓ_1^1	r_1^1	ℓ_2^1	r_2^1	$\ldots$	ℓ_q^1	r_q^1
ℓ_1^2	r_1^2	ℓ_2^2	r_2^0	$\ldots$	ℓ_q^2	r_q^2
ℓ_1^3	r_1^3	ℓ_2^3	r_2^0	$\ldots$	ℓ_q^3	r_q^3

where $\ell_b^j = r_b^{j-1}$ and $r_b^j = f^j(r_b^{j-1}) \oplus \ell_b^{j-1}$.



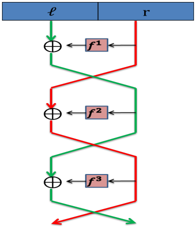
$(f(x_0), \ldots, f(x_q))_{f \xleftarrow{R} \mathrm{LR}^3(\Pi_n)}$ is close to *Distinct*

Let $(\ell_1^0, r_1^0), \ldots, (\ell_q^0, r_q^0) = (x_1, \ldots, x_k)$.

The following rv's are defined w.r.t. $(f^1, f^2, f^3) \xleftarrow{R} \Pi_n^3$.

ℓ_1^0	r_1^0	ℓ_2^0	r_2^0	$\ldots$	ℓ_q^0	r_q^0
ℓ_1^1	r_1^1	ℓ_2^1	r_2^1	$\ldots$	ℓ_q^1	r_q^1
ℓ_1^2	r_1^2	ℓ_2^2	r_2^0	$\ldots$	ℓ_q^2	r_q^2
ℓ_1^3	r_1^3	ℓ_2^3	r_2^0	$\ldots$	ℓ_q^3	r_q^3

where $\ell_b^j = r_b^{j-1}$ and $r_b^j = f^j(r_b^{j-1}) \oplus \ell_b^{j-1}$.



Claim 20

$$\Pr_{f^1 \xleftarrow{R} \Pi_n}\left[\mathrm{Bad}^1 := \exists i \neq j \colon r_i^1 = r_j^1\right] \leq \frac{\binom{q}{2}}{2^n}$$

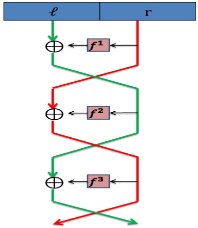
$\left(f(x_0),\ldots,f(x_q)\right)_{f \xleftarrow{R} \mathrm{LR}^3(\Pi_n)}$ is close to *Distinct*

Let $(\ell_1^0, r_1^0), \ldots, (\ell_q^0, r_q^0) = (x_1, \ldots, x_k)$.

The following rv's are defined w.r.t. $(f^1, f^2, f^3) \xleftarrow{R} \Pi_n^3$.

ℓ_1^0	r_1^0	ℓ_2^0	r_2^0	$\ldots$	ℓ_q^0	r_q^0
ℓ_1^1	r_1^1	ℓ_2^1	r_2^1	$\ldots$	ℓ_q^1	r_q^1
ℓ_1^2	r_1^2	ℓ_2^2	r_2^0	$\ldots$	ℓ_q^2	r_q^2
ℓ_1^3	r_1^3	ℓ_2^3	r_2^0	$\ldots$	ℓ_q^3	r_q^3

where $\ell_b^j = r_b^{j-1}$ and $r_b^j = f^j(r_b^{j-1}) \oplus \ell_b^{j-1}$.



Claim 20

$$\Pr_{f^1 \xleftarrow{R} \Pi_n}\left[\mathrm{Bad}^1 := \exists i \neq j \colon r_i^1 = r_j^1\right] \leq \frac{\binom{q}{2}}{2^n}$$

Proof:

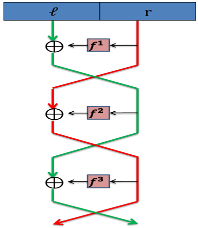
$(f(x_0),\ldots,f(x_q))_{f\xleftarrow{R}\mathrm{LR}^3(\Pi_n)}$ is close to *Distinct*

Let $(\ell_1^0, r_1^0),\ldots,(\ell_q^0, r_q^0) = (x_1,\ldots,x_k)$.

The following rv's are defined w.r.t. $(f^1, f^2, f^3) \xleftarrow{R} \Pi_n^3$.

ℓ_1^0	r_1^0	ℓ_2^0	r_2^0	$\ldots$	ℓ_q^0	r_q^0
ℓ_1^1	r_1^1	ℓ_2^1	r_2^1	$\ldots$	ℓ_q^1	r_q^1
ℓ_1^2	r_1^2	ℓ_2^2	r_2^0	$\ldots$	ℓ_q^2	r_q^2
ℓ_1^3	r_1^3	ℓ_2^3	r_2^0	$\ldots$	ℓ_q^3	r_q^3

where $\ell_b^j = r_b^{j-1}$ and $r_b^j = f^j(r_b^{j-1}) \oplus \ell_b^{j-1}$.



Claim 20

$$\Pr_{f^1\xleftarrow{R}\Pi_n}\left[\mathrm{Bad}^1 := \exists i \neq j\colon r_i^1 = r_j^1\right] \leq \frac{\binom{q}{2}}{2^n}$$

Proof: $r_i^0 = r_j^0 \implies r_i^1 \neq r_j^1$

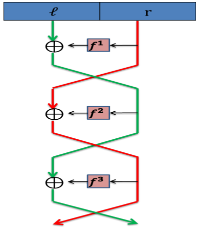
$(f(x_0),\ldots,f(x_q))_{f \xleftarrow{R} \mathrm{LR}^3(\Pi_n)}$ is close to *Distinct*

Let $(\ell_1^0, r_1^0), \ldots, (\ell_q^0, r_q^0) = (x_1, \ldots, x_k)$.

The following rv's are defined w.r.t. $(f^1, f^2, f^3) \xleftarrow{R} \Pi_n^3$.

ℓ_1^0	r_1^0	ℓ_2^0	r_2^0	$\ldots$	ℓ_q^0	r_q^0
ℓ_1^1	r_1^1	ℓ_2^1	r_2^1	$\ldots$	ℓ_q^1	r_q^1
ℓ_1^2	r_1^2	ℓ_2^2	r_2^0	$\ldots$	ℓ_q^2	r_q^2
ℓ_1^3	r_1^3	ℓ_2^3	r_2^0	$\ldots$	ℓ_q^3	r_q^3

where $\ell_b^j = r_b^{j-1}$ and $r_b^j = f^j(r_b^{j-1}) \oplus \ell_b^{j-1}$.



Claim 20

$$\Pr_{f^1 \xleftarrow{R} \Pi_n}\left[\mathrm{Bad}^1 := \exists i \neq j \colon r_i^1 = r_j^1\right] \leq \frac{\binom{q}{2}}{2^n}$$

Proof: $r_i^0 = r_j^0 \implies r_i^1 \neq r_j^1$ and $r_i^0 \neq r_j^0 \implies \Pr_{f^1}\left[r_i^1 = r_j^1\right] = 2^{-n}$ $\square$

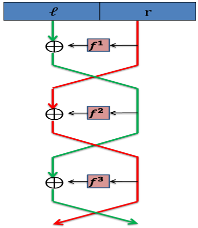
$(f(x_0),\ldots,f(x_q))_{f \xleftarrow{R} \mathrm{LR}^3(\Pi_n)}$ is close to *Distinct*

Let $(\ell_1^0, r_1^0),\ldots,(\ell_q^0, r_q^0) = (x_1,\ldots,x_k)$.

The following rv's are defined w.r.t. $(f^1, f^2, f^3) \xleftarrow{R} \Pi_n^3$.

ℓ_1^0	r_1^0	ℓ_2^0	r_2^0	$\ldots$	ℓ_q^0	r_q^0
ℓ_1^1	r_1^1	ℓ_2^1	r_2^1	$\ldots$	ℓ_q^1	r_q^1
ℓ_1^2	r_1^2	ℓ_2^2	r_2^0	$\ldots$	ℓ_q^2	r_q^2
ℓ_1^3	r_1^3	ℓ_2^3	r_2^0	$\ldots$	ℓ_q^3	r_q^3

where $\ell_b^j = r_b^{j-1}$ and $r_b^j = f^j(r_b^{j-1}) \oplus \ell_b^{j-1}$.



Claim 20

$$\Pr_{f^1 \xleftarrow{R} \Pi_n}\left[\mathrm{Bad}^1 := \exists i \neq j \colon r_i^1 = r_j^1\right] \leq \frac{\binom{q}{2}}{2^n}$$

Proof: $r_i^0 = r_j^0 \implies r_i^1 \neq r_j^1$ and $r_i^0 \neq r_j^0 \implies \Pr_{f^1}\left[r_i^1 = r_j^1\right] = 2^{-n}$ $\square$

Claim 21

$$\Pr_{(f^1,f^2) \xleftarrow{R} \Pi_n^2}\left[\mathrm{Bad}^2 := \exists i \neq j \colon r_i^2 = r_j^2\right] \leq 2 \cdot \frac{\binom{q}{2}}{2^n} \in O\left(\frac{q^2}{2^n}\right)$$

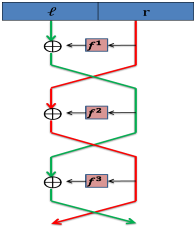
$(f(x_0),\ldots,f(x_q))_{f \xleftarrow{R} LR^3(\Pi_n)}$ is close to *Distinct*

Let $(\ell_1^0, r_1^0),\ldots,(\ell_q^0, r_q^0) = (x_1,\ldots,x_k)$.

The following rv's are defined w.r.t. $(f^1, f^2, f^3) \xleftarrow{R} \Pi_n^3$.

ℓ_1^0	r_1^0	ℓ_2^0	r_2^0	$\ldots$	ℓ_q^0	r_q^0
ℓ_1^1	r_1^1	ℓ_2^1	r_2^1	$\ldots$	ℓ_q^1	r_q^1
ℓ_1^2	r_1^2	ℓ_2^2	r_2^0	$\ldots$	ℓ_q^2	r_q^2
ℓ_1^3	r_1^3	ℓ_2^3	r_2^0	$\ldots$	ℓ_q^3	r_q^3

where $\ell_b^j = r_b^{j-1}$ and $r_b^j = f^j(r_b^{j-1}) \oplus \ell_b^{j-1}$.



Claim 20

$$\Pr_{f^1 \xleftarrow{R} \Pi_n}\left[\mathrm{Bad}^1 := \exists i \neq j \colon r_i^1 = r_j^1\right] \leq \frac{\binom{q}{2}}{2^n}$$

Proof: $r_i^0 = r_j^0 \implies r_i^1 \neq r_j^1$ and $r_i^0 \neq r_j^0 \implies \Pr_{f^1}\left[r_i^1 = r_j^1\right] = 2^{-n}$ $\square$

Claim 21

$$\Pr_{(f^1,f^2) \xleftarrow{R} \Pi_n^2}\left[\mathrm{Bad}^2 := \exists i \neq j \colon r_i^2 = r_j^2\right] \leq 2 \cdot \frac{\binom{q}{2}}{2^n} \in O(\tfrac{q^2}{2^n})$$

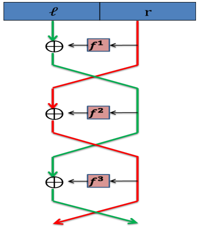
Proof:

$(f(x_0), \ldots, f(x_q))_{f \overset{R}{\leftarrow} \mathrm{LR}^3(\Pi_n)}$ is close to *Distinct*

Let $(\ell_1^0, r_1^0), \ldots, (\ell_q^0, r_q^0) = (x_1, \ldots, x_k)$.

The following rv's are defined w.r.t. $(f^1, f^2, f^3) \overset{R}{\leftarrow} \Pi_n^3$.

ℓ_1^0	r_1^0	ℓ_2^0	r_2^0	$\ldots$	ℓ_q^0	r_q^0
ℓ_1^1	r_1^1	ℓ_2^1	r_2^1	$\ldots$	ℓ_q^1	r_q^1
ℓ_1^2	r_1^2	ℓ_2^2	r_2^0	$\ldots$	ℓ_q^2	r_q^2
ℓ_1^3	r_1^3	ℓ_2^3	r_2^0	$\ldots$	ℓ_q^3	r_q^3



where $\ell_b^j = r_b^{j-1}$ and $r_b^j = f^j(r_b^{j-1}) \oplus \ell_b^{j-1}$.

Claim 20

$$\Pr_{f^1 \overset{R}{\leftarrow} \Pi_n}\left[\mathrm{Bad}^1 := \exists i \neq j \colon r_i^1 = r_j^1\right] \leq \frac{\binom{q}{2}}{2^n}$$

Proof: $r_i^0 = r_j^0 \implies r_i^1 \neq r_j^1$ and $r_i^0 \neq r_j^0 \implies \Pr_{f^1}\left[r_i^1 = r_j^1\right] = 2^{-n}$ $\square$

Claim 21

$$\Pr_{(f^1, f^2) \overset{R}{\leftarrow} \Pi_n^2}\left[\mathrm{Bad}^2 := \exists i \neq j \colon r_i^2 = r_j^2\right] \leq 2 \cdot \frac{\binom{q}{2}}{2^n} \in O\left(\frac{q^2}{2^n}\right)$$

Proof: similar to the above

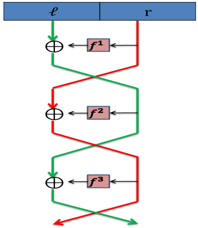
$(f(x_0),\ldots,f(x_q))_{f \stackrel{R}{\leftarrow} \mathrm{LR}^3(\Pi_n)}$ is close to *Distinct*

Let $(\ell_1^0, r_1^0),\ldots,(\ell_q^0, r_q^0) = (x_1,\ldots,x_k)$.

The following rv's are defined w.r.t. $(f^1, f^2, f^3) \stackrel{R}{\leftarrow} \Pi_n^3$.

ℓ_1^0	r_1^0	ℓ_2^0	r_2^0	$\ldots$	ℓ_q^0	r_q^0
ℓ_1^1	r_1^1	ℓ_2^1	r_2^1	$\ldots$	ℓ_q^1	r_q^1
ℓ_1^2	r_1^2	ℓ_2^2	r_2^0	$\ldots$	ℓ_q^2	r_q^2
ℓ_1^3	r_1^3	ℓ_2^3	r_2^0	$\ldots$	ℓ_q^3	r_q^3

where $\ell_b^j = r_b^{j-1}$ and $r_b^j = f^j(r_b^{j-1}) \oplus \ell_b^{j-1}$.



Claim 20

$$\Pr_{f^1 \stackrel{R}{\leftarrow} \Pi_n}\left[\mathrm{Bad}^1 := \exists i \neq j \colon r_i^1 = r_j^1\right] \leq \frac{\binom{q}{2}}{2^n}$$

Proof: $r_i^0 = r_j^0 \implies r_i^1 \neq r_j^1$ and $r_i^0 \neq r_j^0 \implies \Pr_{f^1}\left[r_i^1 = r_j^1\right] = 2^{-n}$ $\square$

Claim 21

$$\Pr_{(f^1,f^2) \stackrel{R}{\leftarrow} \Pi_n^2}\left[\mathrm{Bad}^2 := \exists i \neq j \colon r_i^2 = r_j^2\right] \leq 2 \cdot \frac{\binom{q}{2}}{2^n} \in O\left(\frac{q^2}{2^n}\right)$$

Proof: similar to the above

Claim 22

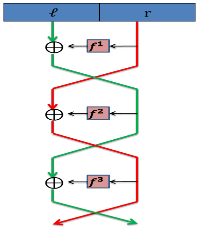
$$\left((\ell_1^3, r_1^3),\ldots,(\ell_q^3, r_q^3) \mid \neg\mathrm{Bad}^2\right) \equiv \textit{Distinct}$$

$\left(f(x_0),\ldots,f(x_q)\right)_{f \xleftarrow{R} \mathrm{LR}^3(\Pi_n)}$ is close to *Distinct*

Let $(\ell_1^0, r_1^0),\ldots,(\ell_q^0, r_q^0) = (x_1,\ldots,x_k)$.

The following rv's are defined w.r.t. $(f^1, f^2, f^3) \xleftarrow{R} \Pi_n^3$.

ℓ_1^0	r_1^0	ℓ_2^0	r_2^0	$\ldots$	ℓ_q^0	r_q^0
ℓ_1^1	r_1^1	ℓ_2^1	r_2^1	$\ldots$	ℓ_q^1	r_q^1
ℓ_1^2	r_1^2	ℓ_2^2	r_2^0	$\ldots$	ℓ_q^2	r_q^2
ℓ_1^3	r_1^3	ℓ_2^3	r_2^0	$\ldots$	ℓ_q^3	r_q^3



where $\ell_b^j = r_b^{j-1}$ and $r_b^j = f^j(r_b^{j-1}) \oplus \ell_b^{j-1}$.

Claim 20

$$\Pr_{f^1 \xleftarrow{R} \Pi_n}\left[\mathrm{Bad}^1 := \exists i \neq j \colon r_i^1 = r_j^1\right] \leq \frac{\binom{q}{2}}{2^n}$$

Proof: $r_i^0 = r_j^0 \implies r_i^1 \neq r_j^1$ and $r_i^0 \neq r_j^0 \implies \Pr_{f^1}\left[r_i^1 = r_j^1\right] = 2^{-n}$ $\square$

Claim 21

$$\Pr_{(f^1,f^2) \xleftarrow{R} \Pi_n^2}\left[\mathrm{Bad}^2 := \exists i \neq j \colon r_i^2 = r_j^2\right] \leq 2 \cdot \frac{\binom{q}{2}}{2^n} \in O\left(\frac{q^2}{2^n}\right)$$

Proof: similar to the above

Claim 22

$$\left((\ell_1^3, r_1^3),\ldots,(\ell_q^3, r_q^3) \mid \neg\,\mathrm{Bad}^2\right) \equiv \textit{Distinct}$$

Proof: ?

Conclusion

- We constructed PRFs and PRPs from length-doubling PRG (and thus from one-way functions)

Conclusion

- We constructed PRFs and PRPs from length-doubling PRG (and thus from one-way functions)
- Main question: find a simpler, more efficient construction

Conclusion

- We constructed PRFs and PRPs from length-doubling PRG (and thus from one-way functions)
- Main question: find a simpler, more efficient construction

  or at least, a less adaptive one