MIT 6.875 & Berkeley CS276

Foundations of Cryptography
Lecture 3

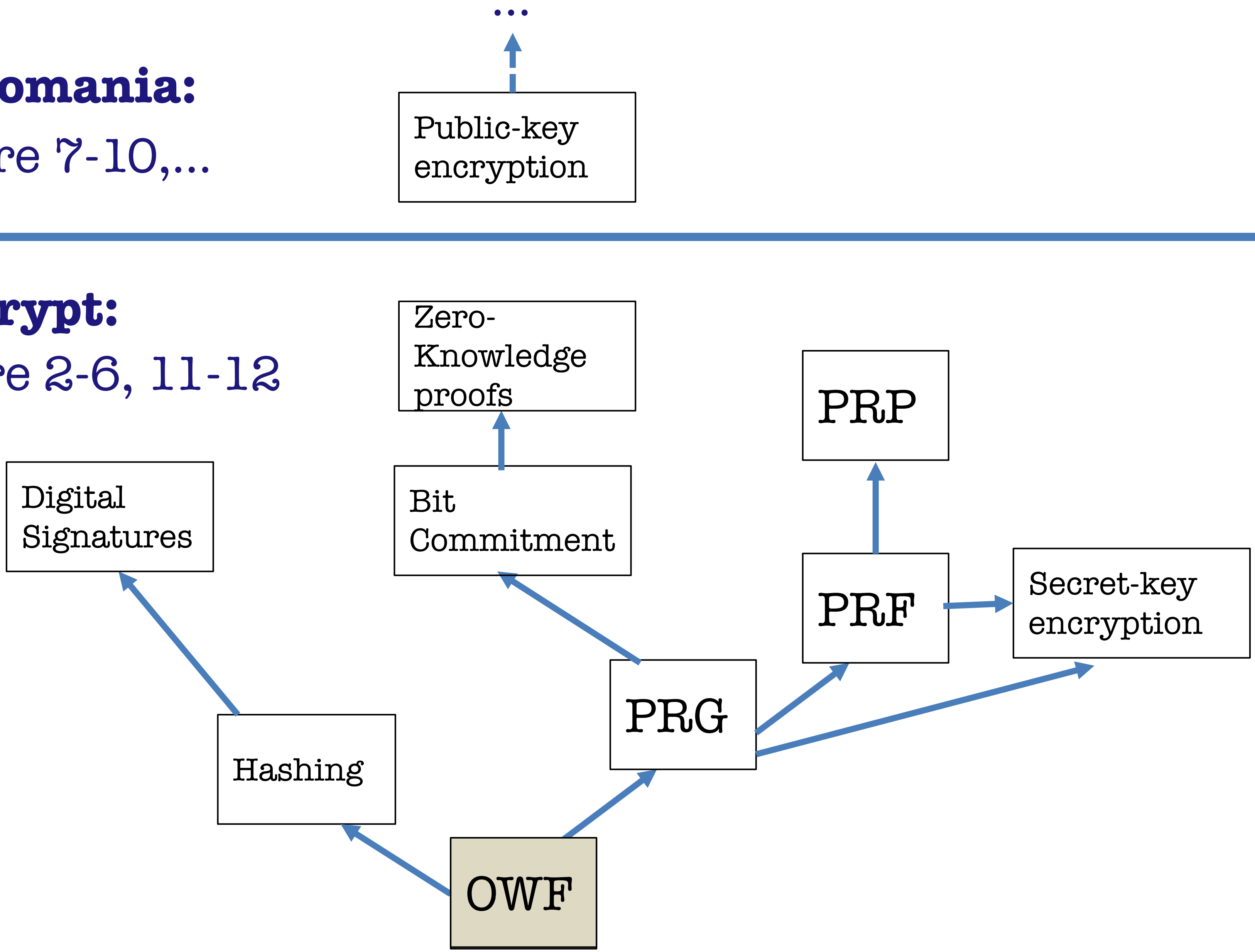
Roadmap of the Course: Worlds in Crypto

Cryptomania:

Lecture 7-10,...

Minicrypt:

Lecture 2-6, 11-12



…

Public-key encryption

Zero-Knowledge proofs

PRP

Digital Signatures

Bit Commitment

PRF

Secret-key encryption

PRG

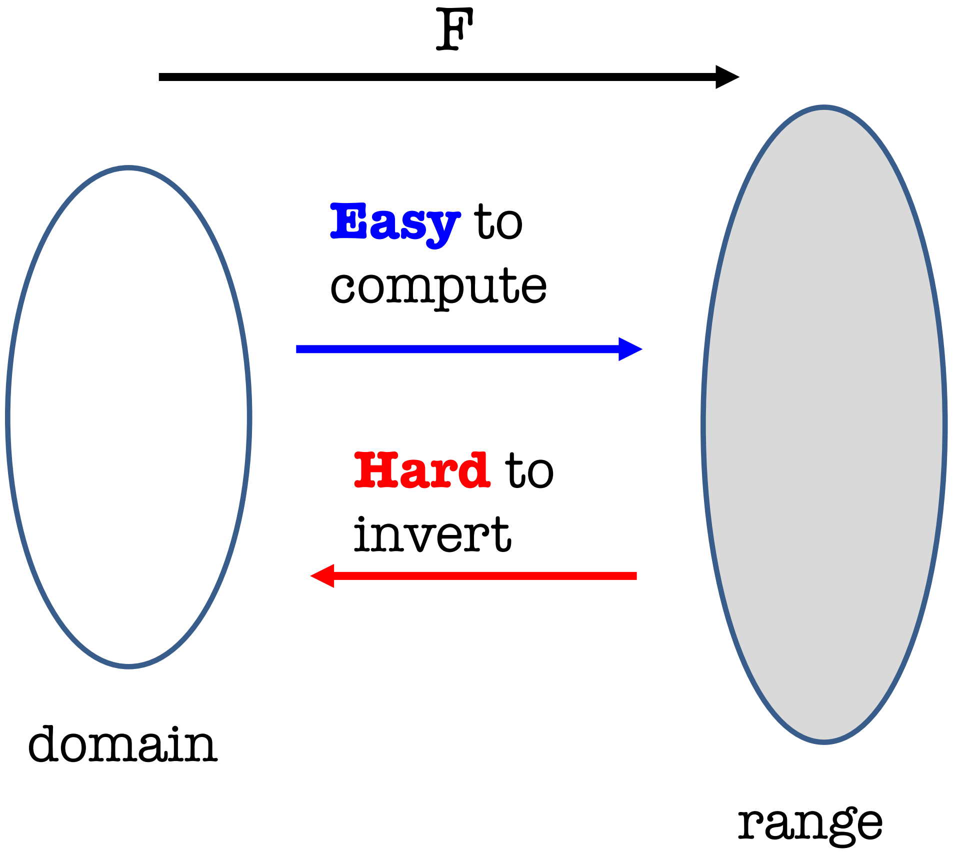
Hashing

OWF

Today

1. Define one-way functions (OWF).
2. Define Hardcore bits (HCB).
3. Show that one-way functions* + HCB ⇒ PRG
4. Goldreich-Levin Theorem: every OWF has a HCB.

One-way Functions (Informally)



One-way Functions (Take 1)

A function (family) $\{F_n\}_{n\in\mathbb{N}}$ where $F_n: \{0,1\}^n \rightarrow \{0,1\}^{m(n)}$ is one-way if for every p.p.t. adversary A, there is a negligible function μ s.t.

$$\Pr[x \leftarrow \{0,1\}^n; y = F_n(x): A(1^n, y) = x] \leq \mu(n)$$

Consider $\boldsymbol{F_n(x) = 0}$ for all x.

This is one-way according to the above definition.
In fact, impossible to find *the* inverse even if A has unbounded time.

Conclusion: not a useful/meaningful definition.

One-way Functions (Take 1)

A function (family) $\{F_n\}_{n\in\mathbb{N}}$ where $F_n\colon \{0,1\}^n \to \{0,1\}^{m(n)}$ is one-way if for every p.p.t. adversary A, there is a negligible function μ s.t.

$$\Pr[x \leftarrow \{0,1\}^n; y = F_n(x) \colon A(1^n, y) = x] \leq \mu(n)$$

The Right Definition: Impossible to find ***an*** inverse in p.p.t.

One-way Functions: The Definition

A function (family) $\{F_n\}_{n\in\mathbb{N}}$ where $F_n\colon \{0,1\}^n \rightarrow \{0,1\}^{m(n)}$ is one-way if for every p.p.t. adversary A, there is a negligible function μ s.t.

$$\Pr[x \leftarrow \{0,1\}^n; y = F_n(x); A(1^n, y) = \boldsymbol{x'\colon y = F_n(x')}] \le \mu(n)$$

- Can always find *an* inverse with unbounded time
- … but should be hard with probabilistic polynomial time

One-way Permutations:

One-to-one one-way functions with $m(n) = n$.

One-way Functions: Candidates

Subset sum:

$G(a_1, \dots, a_n, x_1, \dots, x_n) = (a_1, \dots, a_n, \sum_{i=1}^{n} x_i a_i \bmod 2^{n+1})$

where a_i are random n-bit numbers, and x_i are random bits.

One-way functions candidates are abundant in nature.

We will see many other candidates from number theory, coding theory, combinatorics later in class.

Today

1. Define one-way functions (OWF).
2. Define Hardcore bits (HCB).
3. Show that one-way *permutations* (OWP) ⇒ PRG
4. Goldreich-Levin Theorem: every OWF has a HCB.

Hardcore Bits

If F is a one-way function, we know it's hard to compute a pre-image of $F(x)$ for a randomly chosen x.

How about computing partial information about an inverse?

Exercise: There are one-way functions for which it is easy to compute the first half of the bits of the inverse.

Hardcore Bits

If F is a one-way function, we know it's hard to compute a pre-image of $F(x)$ for a randomly chosen x.

HARDCORE BIT (Take 1)

Nevertheless, there has to be a hardcore set of hard to invert inputs. Concretely: Does there exist some bit of x that is hard to compute?

Does there necessarily exist some bit of x to guess with probability non-negligibly better than 1/2?

- Any bit can be guessed correctly w.p. 1/2
- So, "hard to compute" → "hard to guess with probability non-negligibly better than 1/2"

Hardcore Bits

If F is a one-way function, we know it's hard to compute a pre-image of $F(x)$ for a randomly chosen x.

HARDCORE BIT (Take 1)

For any function (family) $F: \{0,1\}^n \rightarrow \{0,1\}^m$, a bit $i = i(n)$ is hardcore if for every p.p.t. adversary A, there is a negligible function μ s.t.

$$\Pr[x \leftarrow \{0,1\}^n; y = F(x): A(y) = x_i] \leq \frac{1}{2} + \mu(n)$$

Does every one-way function have a hardcore bit?

(Hard) Exercise: There are functions that are one-way, yet *every* bit is somewhat easy to predict (say, with probability $\frac{1}{2} + 1/n$).

So, we will generalize the notion of a hardcore "bit".

Hardcore Bits

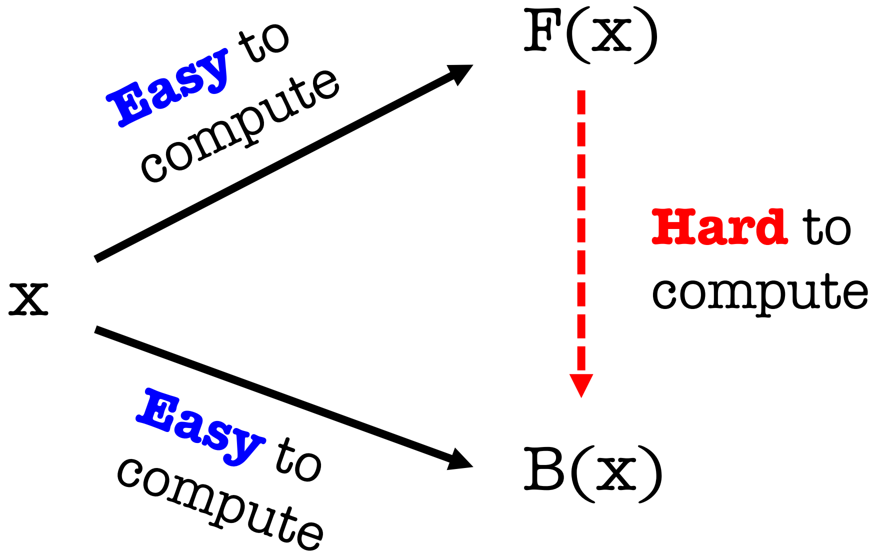
HARDCORE PREDICATE (Definition)

For any function (family) $F: \{0,1\}^n \rightarrow \{0,1\}^m$, a function $B: \{0,1\}^n \rightarrow \{0,1\}$ is a hardcore **predicate** if for every p.p.t. adversary A, there is a negligible function μ s.t.

$$\Pr[x \leftarrow \{0,1\}^n; y = F(x): A(y) = B(x)] \leq \frac{1}{2} + \mu(n)$$

For us, henceforth, a hardcore bit will mean a hardcore predicate.

Hardcore Predicate (in pictures)



Discussion on the Definition

HARDCORE PREDICATE (Definition)

For any function (family) $F: \{0,1\}^n \rightarrow \{0,1\}^m$, a bit $B: \{0,1\}^n \rightarrow \{0,1\}$ is a hardcore **predicate** (HCP) if for every p.p.t. adversary A, there is a negligible function μ s.t.

$$\Pr[x \leftarrow \{0,1\}^n; y = F(x): A(y) = B(x)] \leq \frac{1}{2} + \mu(n)$$

1. Definition of HCP makes sense for *any* function family, not just one-way functions.

2. Some functions can have information-theoretically hard to guess predicates (e.g., compressing functions)

3. We'll be interested in settings where x is uniquely determined given $\mathrm{F}(x)$, yet $\mathrm{B}(x)$is hard to predict given $\mathrm{F}(x)$

Today

1. Define one-way functions (OWF).

2. Define Hardcore bits (HCB).

3. Show that one-way *permutations* (OWP) ⇒ PRG

4. Goldreich-Levin Theorem: every OWF has a HCB.

OWP $\Rightarrow$ PRG

CONSTRUCTION

Let F be a one-way permutation, and B an associated hardcore predicate for F.

Then, define $G(x) = \mathrm{F}(x) \mid \mathrm{B}(x)$.

Theorem: G is a PRG assuming F is a one-way permutation.

(Note that G stretches by one bit. Shafi will tell you how to extend the stretch of G to any poly number of bits.)

OWP $\Rightarrow$ PRG

CONSTRUCTION

Let F be a one-way permutation, and B an associated hardcore predicate for F.

Then, define $G(x) = \mathrm{F}(x) \mid \mathrm{B}(x)$.

Theorem: G is a PRG assuming F is a one-way permutation.

Proof (next slide): From Distinguishing to Predicting.

OWP $\Rightarrow$ PRG

Theorem: G is a PRG assuming F is a one-way permutation.

Proof: Assume for contradiction that G is not a PRG. Therefore, there is a p.p.t. distinguisher D and a polynomial function p such that

$$\Pr[x \leftarrow \{0,1\}^n; y = G(x) : D(y) = 1] - \Pr[y \leftarrow \{0,1\}^{n+1} : D(y) = 1] \geq 1/p(n)$$

Think: D outputs "1" $=$ D thinks its input is pseudorandom

OWP $\Rightarrow$ PRG

Theorem: G is a PRG assuming F is a one-way permutation and B is its hardcore predicate .

Proof: Assume for contradiction that G is not a PRG. Therefore, there is a p.p.t. distinguisher D and a polynomial function p such that

$$\Pr[x \leftarrow \{0,1\}^n; y = G(x) : D(y) = 1] - \Pr[y \leftarrow \{0,1\}^{n+1} : D(y) = 1] \geq 1/p(n)$$

We will construct a hardcore predictor A and show:

$$\Pr[x \leftarrow \{0,1\}^n : A(F(x)) = B(x)] \geq \frac{1}{2} + 1/p'(n)$$

OWP $\Rightarrow$ PRG

Theorem: G is a PRG assuming F is a one-way permutation and B is its hardcore predicate .

Proof: Assume for contradiction that G is not a PRG. Therefore, there is a p.p.t. distinguisher D and a polynomial function p such that

$$\Pr[x \leftarrow \{0,1\}^n; y = G(x) : D(y) = 1] - \Pr[y \leftarrow \{0,1\}^{n+1} : D(y) = 1] \geq 1/p(n)$$

We will construct a hardcore predictor A and show:

$$\Pr[x \leftarrow \{0,1\}^n : A(F(x)) = B(x)] \geq \frac{1}{2} + 1/p'(n)$$

OWP ⇒ PRG

Let's look closely at D.

$$\Pr[x \leftarrow \{0,1\}^n; y = G(x) \colon D(y) = 1] - \Pr[y \leftarrow \{0,1\}^{n+1} : D(y) = 1] \geq 1/p(n)$$

By definition:

$$\Pr[x \leftarrow \{0,1\}^n; y = F(x) | B(x) \colon D(y) = 1] - \Pr[y \leftarrow \{0,1\}^{n+1} : D(y) = 1] \geq 1/p(n)$$

OWP ⇒ PRG

Let’s look closely at D.

$$\Pr[x \leftarrow \{0,1\}^n; y = G(x): D(y) = 1] - \Pr[y \leftarrow \{0,1\}^{n+1} : D(y) = 1] \geq 1/p(n)$$

A syntactic change:

$$\Pr[x \leftarrow \{0,1\}^n; y = F(x)|B(x): D(y) = 1] - \Pr[y_0 \leftarrow \{0,1\}^n , y_1 \leftarrow \{0,1\}, y = y_0|y_1: D(y) = 1] \geq 1/p(n)$$

OWP ⇒ PRG

Let's look closely at D.

$$\begin{aligned}&\Pr[x \leftarrow \{0,1\}^n; y = G(x) \colon D(y) = 1] - \\ &\qquad \Pr[y \leftarrow \{0,1\}^{n+1} : D(y) = 1] \geq 1/p(n)\end{aligned}$$

Rewriting the second term:

$$\begin{aligned}&\Pr[x \leftarrow \{0,1\}^n; y = F(x)|B(x) \colon D(y) = 1] - \\ &\Pr[x \leftarrow \{0,1\}^n, y_1 \leftarrow \{0,1\}, y = F(x)|y_1 \colon D(y) = 1] \geq 1/p(n)\end{aligned}$$

$$= \frac{\Pr[x \leftarrow \{0,1\}^n, y = F(x)|\mathbf{0} \colon D(y) = 1] + \Pr[x \leftarrow \{0,1\}^n, y = F(x)|\mathbf{1} \colon D(y) = 1]}{2}$$

OWP ⇒ PRG

Let’s look closely at D.

$$\Pr[x \leftarrow \{0,1\}^n; y = G(x): D(y) = 1] - \Pr[y \leftarrow \{0,1\}^{n+1} : D(y) = 1] \geq 1/p(n)$$

Rewriting the second term (again):

$$\Pr[x \leftarrow \{0,1\}^n; y = F(x)|B(x): D(y) = 1] - \Pr[x \leftarrow \{0,1\}^n, y_1 \leftarrow \{0,1\}, y = F(x)|y_1: D(y) = 1] \geq 1/p(n)$$

$$=$$

$$\frac{\Pr[x \leftarrow \{0,1\}^n, y = F(x)|\boldsymbol{B(x)}: D(y) = 1] + \Pr[x \leftarrow \{0,1\}^n, y = F(x)|\overline{\boldsymbol{B(x)}}: D(y) = 1]}{2}$$

OWP ⇒ PRG

Let's look closely at D.

$$\Pr[x \leftarrow \{0,1\}^n; y = G(x): D(y) = 1] - \Pr[y \leftarrow \{0,1\}^{n+1} : D(y) = 1] \geq 1/p(n)$$

Putting things together:

$$\frac{1}{2}(\Pr[x \leftarrow \{0,1\}^n; y = F(x)|\boldsymbol{B(x)}: D(y) = 1] - \Pr\left[x \leftarrow \{0,1\}^n, y = F(x)|\overline{\boldsymbol{B(x)}}: D(y) = 1\right]) \geq 1/p(n)$$

In English: D says "1" more often when fed with the "right bit" than the "wrong bit".

OWP ⇒ PRG

Let's look closely at D.

$$\Pr[x \leftarrow \{0,1\}^n; y = G(x): D(y) = 1] - \Pr[y \leftarrow \{0,1\}^{n+1} : D(y) = 1] \geq 1/p(n)$$

Putting things together:

$$(*) \quad \frac{1}{2}(\Pr[x \leftarrow \{0,1\}^n; y = F(x)|\boldsymbol{B(x)}: D(y) = 1] - \Pr\left[x \leftarrow \{0,1\}^n , y = F(x)|\overline{\boldsymbol{B(x)}}: D(y) = 1\right]) \geq 1/p(n)$$

Now, let's use D to *predict* the right bit.

OWP $\Rightarrow$ PRG

The Predictor A works as follows:

Get as input $z = F(x)$; Pick a random bit b; and feed D with input $z|\mathrm{b}$.

If D says “1”, output b as the prediction for the hardcore bit and if D says “0”, output $\bar{b}$.

Analysis of the Predictor A

$$\Pr[x \leftarrow \{0,1\}^n : A(F(x)) = B(x)]$$

$$= \Pr[x \leftarrow \{0,1\}^n : D(F(x)|b) = 1 |\ b = B(x)]\Pr[b = B(x)] +$$
$$\Pr[x \leftarrow \{0,1\}^n : D(F(x)|b) = 0 |\ b \neq B(x)]\Pr[b \neq B(x)]$$

$$= \frac{1}{2}(\Pr[x \leftarrow \{0,1\}^n : D(F(x)|b) = 1 |\ b = B(x)] +$$
$$\Pr[x \leftarrow \{0,1\}^n : D(F(x)|b) = 0 |\ b \neq B(x)])$$

$$= \frac{1}{2}(\Pr[x \leftarrow \{0,1\}^n : D(F(x)|B(x)) = 1] +$$
$$\Pr\left[x \leftarrow \{0,1\}^n : D\left(F(x)|\overline{B(x)}\right) = 0\right])$$

$$= \frac{1}{2}(\Pr[x \leftarrow \{0,1\}^n : D(F(x)|B(x)) = 1] +$$
$$1 - \Pr\left[x \leftarrow \{0,1\}^n : D\left(F(x)|\overline{B(x)}\right) = 1\right])$$

$$= \frac{\mathbf{1}}{\mathbf{2}}(\mathbf{1} + (*)) \quad \geq \frac{\mathbf{1}}{\mathbf{2}} + \mathbf{1}/\boldsymbol{p}(\boldsymbol{n})$$

$\blacksquare$

Today

1. Define one-way functions (OWF).
2. Define Hardcore bits (HCB).
3. Show that one-way *permutations* (OWP) ⇒ PRG
4. Goldreich-Levin Theorem: every OWF has a HCB.

A Hardcore Predicate for all OWF

Let’s shoot for a *universal* hardcore predicate.

i.e., a single predicate B where it is hard to guess $B(x)$ given $\mathrm{F}(x)$

Is this possible?

Turns out the answer is “no”. Pick your favorite amazing B. I claim that you can construct a one-way function F for which B is not hard-core. I will leave it to you as an exercise.

So, what is one to do?

Goldreich-Levin (GL) Theorem

Let $\{B_r: \{0,1\}^n \to \{0,1\}\}$ where

$$B_r(x) = \langle r, x\rangle = \sum_{i=1}^{n} r_i x_i \bmod 2$$

be a collection of predicates (one for each r). Then, a ***random*** B_r is hardcore for ***every*** one-way function F. That is, for every one-way function F, every PPT A, there is a negligible function μ s.t.

$$\Pr[x \leftarrow \{0,1\}^n; r \leftarrow \{0,1\}^n: A(F(x), r) = B_r(x)] \leq \frac{1}{2} + \mu(n)$$

Alternative Interpretation 1: For every one-way function F, there is a related one-way function $F'(x,r) = (F(x), r)$ which has a *deterministic* hardcore predicate.

Goldreich-Levin (GL) Theorem

Let $\{B_r: \{0,1\}^n \rightarrow \{0,1\}\}$ where

$$B_r(x) = \langle r, x\rangle = \sum_{i=1}^{n} r_i x_i \bmod 2$$

be a collection of predicates (one for each r). Then, a ***random*** B_r is hardcore for ***every*** one-way function F. That is, for every one-way function F, every PPT A, there is a negligible function μ s.t.

$$\Pr[x \leftarrow \{0,1\}^n; r \leftarrow \{0,1\}^n: A(F(x), r) = B_r(x)] \le \frac{1}{2} + \mu(n)$$

Alternative Interpretation 2: For every one-way function F, there ***exists*** (non-uniformly) a (possibly different) hardcore predicate $\langle r_F, x\rangle$. **(Cool open problem: remove the non-uniformity)**

Proof of GL Theorem

Let's make our lives easier: assume a perfect predictor P

~~Assume for contradiction there is a predictor P~~

$$\Pr[x \leftarrow \{0,1\}^n; r \leftarrow \{0,1\}^n : P(F(x), r) = \langle r, x \rangle] \geq \frac{1}{2} + 1/p(n)$$

$$\Pr[x \leftarrow \{0,1\}^n; r \leftarrow \{0,1\}^n : P(F(x), r) = \langle r, x \rangle] = 1$$

We will need to show an inverter A for F

$$\Pr\left[x \leftarrow \{0,1\}^n : A\left(F(x)\right) = x' : F(x') = F(x)\right] \geq 1/p'(n)$$

Proof of GL Theorem

Let's make our lives easier: assume a perfect predictor P

~~Assume for contradiction there is a predictor P~~

$$\Pr[x \leftarrow \{0,1\}^n; r \leftarrow \{0,1\}^n: P(F(x), r) = \langle r, x \rangle] = 1$$

The inverter A works as follows:

On input $\mathrm{y} = F(x)$, A runs the predictor P n times, on inputs $(y, e_1), (y, e_2), \dots,$ and (y, e_n) where $e_1 = 100..0, e_2 = 010 \dots 0, \dots$ are the unit vectors.

Since A is perfect, it returns $\langle e_i, x \rangle = x_i$, the i^{th} bit of x on the i^{th} invocation.

Proof of GL Theorem

OK, now let's assume less: assume a pretty good predictor P

~~Assume for contradiction there is a predictor P~~

$$\Pr[x \leftarrow \{0,1\}^n; r \leftarrow \{0,1\}^n : P(F(x), r) = \langle r, x \rangle] \geq \frac{3}{4} + 1/p(n)$$

First, we need an **averaging argument**.

Claim: For at least a $1/2p(n)$ fraction of the x,

$$\Pr[r \leftarrow \{0,1\}^n : P(F(x), r) = \langle r, x \rangle] \geq \frac{3}{4} + 1/2p(n)$$

Proof: Exercise in counting.

Call these the good x.

Proof of GL Theorem

For at least a $1/2p(n)$ fraction of the x,

$$\Pr[r \leftarrow \{0,1\}^n : P(F(x), r) = \langle r, x \rangle] \geq \frac{3}{4} + 1/2p(n)$$

Key Idea: Linearity

Pick a random r and ask P to tells us $\langle r, x \rangle$ and $\langle r + e_i, x \rangle$.
Subtract the two answers to get $\langle e_i, x \rangle = x_i$.

Proof: $\Pr[\text{we compute } x_i \text{ correctly}]$

$$\geq \Pr[\text{P predicts } \langle r, x \rangle \text{ and } \langle r + e_i, x \rangle \text{ correctly}]$$

$$= 1 - \Pr[\text{P predicts} \langle r, x \rangle \text{ or } \langle r + e_i, x \rangle \text{ wrong}]$$

$$\geq 1 - (\Pr[\text{P predicts} \langle r, x \rangle \text{ wrong}] + \Pr[\text{P predicts} \langle r + e_i, x \rangle \text{ wrong}])$$ *(by union bound)*

$$\geq 1 - 2 \cdot \left(\frac{1}{4} - \frac{1}{2p(n)} \right) = \frac{1}{2} + 1/p(n)$$

Proof of GL Theorem

For at least a $1/2p(n)$ fraction of the x,

$$\Pr[r \leftarrow \{0,1\}^n : P(F(x), r) = \langle r, x \rangle] \geq \frac{3}{4} + 1/2p(n)$$

Inverter A:

```
Repeat for each $i \in \{1,2,\ldots,n\}$:
    Repeat $\log n / p(n)$ times:
        Pick a random $r$ and ask $P$ to tells us $\langle r, x \rangle$ and $\langle r + e_i, x \rangle$.
        Subtract the two answers to get a guess for $x_i$.
    Compute the majority of all such guesses and set the bit as $x_i$
Output the concatenation of all $x_i$ as $x$.
```

Analysis: Chernoff + Union Bound

Real Proof (will not do in class)

Assume (after averaging) that for $\geq 1/2p(n)$ fraction of the x,

$$\Pr[r \leftarrow \{0,1\}^n : P(F(x), r) = \langle r, x \rangle] \geq \mathbf{\frac{1}{2}} + 1/2p(n)$$

Key Idea: Pairwise independence

Reference: Goldreich Book Part 1, Section 2.5.2.

http://www.wisdom.weizmann.ac.il/~oded/PSBookFrag/part2N.ps

The Coding-Theoretic View of GL

$x \to (\langle x, r\rangle)_{r \in \{0,1\}^n}$ can be viewed as a highly redundant, exponentially long encoding of x = **the Hadamard code**.

$P(F(x), r)$ can be thought of as providing access to a **noisy** codeword.

What we proved = **unique decoding** algorithm for Hadamard code with error rate $\frac{1}{4} - 1/p(n)$.

The real proof = **list-decoding algorithm** for Hadamard code with error rate $\frac{1}{2} - 1/p(n)$.

Recap

1. Defined one-way functions (OWF).

2. Defined Hardcore bits (HCB).

3. Goldreich-Levin Theorem: every OWF has a HCB.

   (showed proof for an important special case)

4. Show that one-way *permutations* (OWP) ⇒ PRG

   (*in fact, one-way functions* ⇒ PRG, but that's a much harder theorem)

Next Lecture: Back to PRGs