Cryptography

Corso di Laurea Magistrale in Informatica

Theoretical Constructions of Pseudorandom Objects and Hash Functions

Ugo Dal Lago

ALMA MATER STUDIORUM UNIVERSITÀ DI BOLOGNA

Inria informatiques mathématiques

Academic Year 2023-2024

Theoretical Constructions of Pseudorandom Objects and Hash Functions

- As already mentioned, pseudorandom objects and hash functions can be constructed:
  - **Practically**, as we did in the previous chapter.
  - **Theoretically**, as we will do in this chapter, but only for pseudorandom objects.

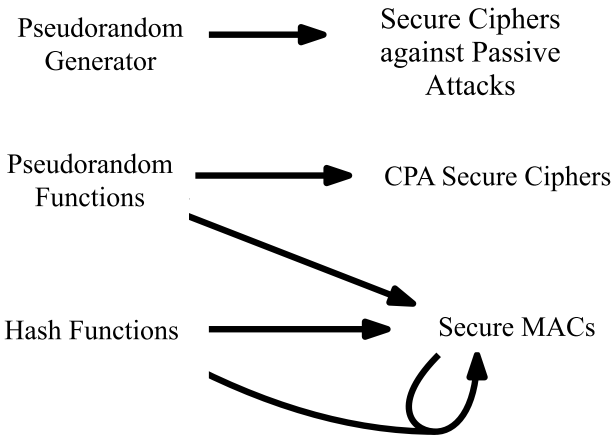
Theoretical Constructions of Pseudorandom Objects and Hash Functions

- As already mentioned, pseudorandom objects and hash functions can be constructed:
  - **Practically**, as we did in the previous chapter.
  - **Theoretically**, as we will do in this chapter, but only for pseudorandom objects.
- In the theoretical approach, we will show that pseudorandom objects can be constructed **from other objects** whose existence, although not certain, is considered to be highly probable.
  - Cryptography, pseudorandomness and (partly) computational complexity deal with the **non-existence** of certain polytime algorithms with specific properties.
  - The work we did in the first part of the course, and which we will continue to do, consists in inferring this non-existence from gradually weaker hypothesis.

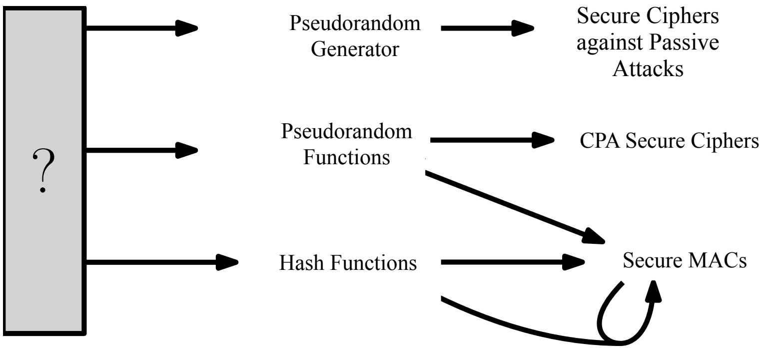
Theoretical Constructions of Pseudorandom Objects and Hash Functions

- As already mentioned, pseudorandom objects and hash functions can be constructed:
  - **Practically**, as we did in the previous chapter.
  - **Theoretically**, as we will do in this chapter, but only for pseudorandom objects.
- In the theoretical approach, we will show that pseudorandom objects can be constructed **from other objects** whose existence, although not certain, is considered to be highly probable.
  - Cryptography, pseudorandomness and (partly) computational complexity deal with the **non-existence** of certain polytime algorithms with specific properties.
  - The work we did in the first part of the course, and which we will continue to do, consists in inferring this non-existence from gradually weaker hypothesis.
- The interest in the constructions we will give is theoretical, but for purely efficiency reasons.

The Situation, in Brief



Where We Want to Go



One-Way Functions

- *Informally*, a one-way function is a function that is easy to to compute, but which is hard to invert.

One-Way Functions

- *Informally*, a one-way function is a function that is easy to to compute, but which is hard to invert.
- How to *formalise* this idea? We will proceed, in line with this course, by giving an experiment.

$\mathsf{Invert}_{A,f}(n)$:
$x \leftarrow \{0,1\}^n$;
$y \leftarrow f(x)$;
$z \leftarrow A(1^n, y)$;
Result: $(f(z) = y)$

One-Way Functions

- *Informally*, a one-way function is a function that is easy to to compute, but which is hard to invert.
- How to *formalise* this idea? We will proceed, in line with this course, by giving an experiment.

$\mathsf{Invert}_{A,f}(n)$:
$x \leftarrow \{0,1\}^n$;
$y \leftarrow f(x)$;
$z \leftarrow A(1^n, y)$;
Result: $(f(z) = y)$

Definition

A function $f : \{0,1\}^* \to \{0,1\}^*$ is a **one-way function** iff there exists a polytime and deterministic algorithm which computes f and furthermore for every PPT A there exists a negligible ε such that

$$Pr(\mathsf{Invert}_{A,f}(n) = 1) \leq \varepsilon(n)$$

One-Way Functions

- **One-way permutations** are length-preserving one-way functions with the interesting property that $y \in \{0, 1\}^*$ uniquely determines x such that $f(x) = y$.

One-Way Functions

- **One-way permutations** are length-preserving one-way functions with the interesting property that $y \in \{0,1\}^*$ uniquely determines x such that $f(x) = y$.
- *Examples of (Assumed) One-Way Functions*:
  - **Multiplication Between Natural Numbers**
    - Consider f_{MULT} defined by $f_{MULT}(x, y) = x \cdot y$: given two strings that we interpret as natural numbers, we return their product.
    - If we do not put any other constraints on x and y, the function f_{MULT} is easily invertible.
    - We will study this function in the next chapter.
  - **Subset-Sum Problem**
    - Consider instead the function f_{SS} defined by $f_{SS}(x_1, \ldots, x_n, J) = (x_1, \ldots, x_n, \sum_{j \in J} x_j)$, where $|x_j| = n$ and J is interpreted as a subset of $\{1, \ldots, n\}$.
    - The inverse of f_{SS} corresponds to the so-called subset-sum problem.

Hard-Core Predicates

- A one-way function f is such that $f(x)$ does not *entirely* reveal x
  - This does not imply that the same applies to **parts of** x, for example a single bit.
  - Consider for example a one-way function f , and construct $g(x, y) = (x, f(y))$. g is also a one-way function (if we could invert g, we can also invert f). But $g(x, y)$ reveals an important part of its input, namely x.

Definition

A predicate $hc : \{0,1\}^* \rightarrow \{0,1\}$ is called *hard-core predicate* of a function f if and only if hc is polynomial time computable and for every adversary PPT A it holds that

$$Pr(A(f(x)) = hc(x)) \leq \frac{1}{2} + \varepsilon(n)$$

where ε is negligible.

Hard-Core Predicates

- It may seem at first sight that hc defined by $hc(x_1 \cdots x_n) = \oplus_{i=1}^{n} x_i$ is a hard-core predicate *for each* function f.
  - Given g one-way, the function f defined by $f(x) = (g(x), hc(x))$ is also a one-way function, but certainly hc is not hard-core for f, because its value is easily retrievable from the output.
- For some (non-one-way) functions, it is possible to construct trivial hard-core predicates. For example the function $f : \{0,1\}^* \to \{0,1\}^*$ defined by $f(\epsilon) = \epsilon$ and $f(b \cdot s) = s$ for every $b \in \{0,1\}$ and $s \in \{0,1\}^*$.
  - The result $f(s)$ does not depend on the first bit of s, which can then become a hard-core predicate.

The Goldreich-Levin Theorem

Theorem

If there is a one-way function (respectively, a one-way permutation) f, then there exists a one-way function (respectively a one-way permutation) g and a hard-core predicate hc for g.

The Goldreich-Levin Theorem

Theorem

If there is a one-way function (respectively, a one-way permutation) f, then there exists a one-way function (respectively a one-way permutation) g and a hard-core predicate hc for g.

- This is one of the most important results in the theory of one-way functions, with crucial implications in cryptography.
- The function g is constructed from f by setting $g(x, r) = (f(x), r)$, while hc is defined by $hc(x, r) = \oplus_{i=1}^{n} x_i \cdot r_i$.

From One-Way Permutations to Pseudorandom Generators

Theorem

Let f be a one-way permutation and let hc be a hard-core predicate for f. Then G defined by $G(s) = (f(s), hc(s))$ is a pseudorandom generator with expansion factor $\ell(n) = n + 1$.

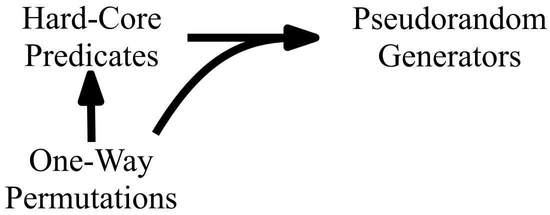
From One-Way Permutations to Pseudorandom Generators

Theorem

Let f be a one-way permutation and let hc be a hard-core predicate for f. Then G defined by $G(s) = (f(s), hc(s))$ is a pseudorandom generator with expansion factor $\ell(n) = n + 1$.

- This is another crucial result, linking the theory of one-way functions to pseudorandomness.
- Intuitively, the first $|s|$ bits of G's output are pseudorandom due to the properties of f, while the last bit is pseudorandom due to the properties of hc.

One-Way Permutations and Pseudorandom Generators



Arbitrary Expansion Factor

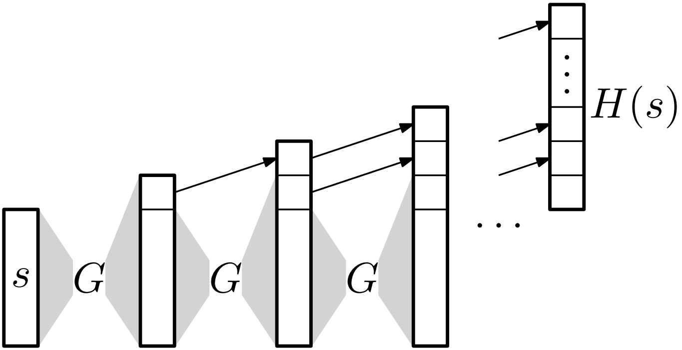
Theorem

If there exists a pseudorandom generator G with expansion factor $\ell(n) = n + 1$, then there exists another other pseudorandom generator H, with an arbitrary expansion factor, as long as it is polynomial.

Arbitrary Expansion Factor

Theorem

If there exists a pseudorandom generator G with expansion factor $\ell(n) = n + 1$, then there exists another other pseudorandom generator H, with an arbitrary expansion factor, as long as it is polynomial.



From Generators to Functions

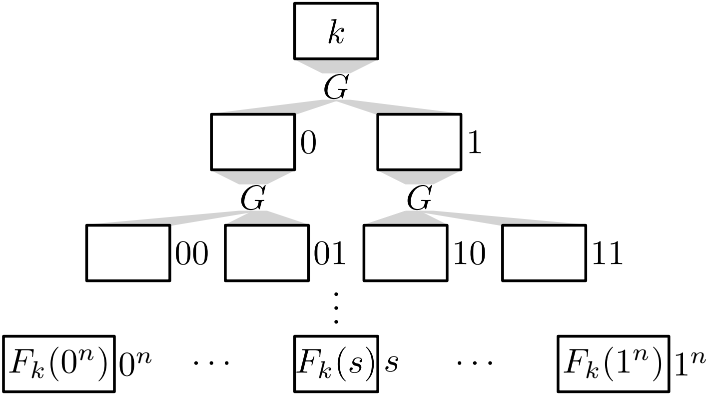
Theorem

If there exists a pseudorandom generator G with expansion factor $\ell(n) = 2n$, then there exists a pseudorandom function.

From Generators to Functions

Theorem

If there exists a pseudorandom generator G with expansion factor $\ell(n) = 2n$, then there exists a pseudorandom function.



Coming Full Circle

Theorem

If a pseudorandom function exists, then there exists a strong pseudorandom permutation.

Coming Full Circle

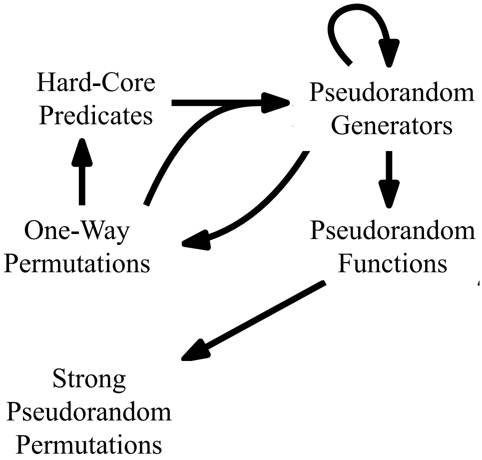
Theorem

If a pseudorandom function exists, then there exists a strong pseudorandom permutation.

Theorem

If there is a pseudorandom generator, then there is a one-way function.

One-Way Functions and Pseudorandomness



The Overall Picture