WIKIPEDIA

One-way function

In computer science, a **one-way function** is a function that is easy to compute on every input, but hard to invert given the image of a random input. Here, "easy" and "hard" are to be understood in the sense of computational complexity theory, specifically the theory of polynomial time problems. Not being one-to-one is not considered sufficient of a function for it to be called one-way (see Theoretical definition, below).

Unsolved problem in computer science:

? *Do one-way functions exist?*

(more unsolved problems in computer science)

The existence of such one-way functions is still an open conjecture. In fact, their existence would prove that the complexity classes P and NP are not equal, thus resolving the foremost unsolved question of theoretical computer science.[1]:ex. 2.2, page 70 The converse is not known to be true, i.e. the existence of a proof that P and NP are not equal would not directly imply the existence of one-way functions.[2]

In applied contexts, the terms "easy" and "hard" are usually interpreted relative to some specific computing entity; typically "cheap enough for the legitimate users" and "prohibitively expensive for any malicious agents". One-way functions, in this sense, are fundamental tools for cryptography, personal identification, authentication, and other data security applications. While the existence of one-way functions in this sense is also an open conjecture, there are several candidates that have withstood decades of intense scrutiny. Some of them are essential ingredients of most telecommunications, e-commerce, and e-banking systems around the world.

Contents

Theoretical definition

A function $f : \{0,1\}^* \to \{0,1\}^*$ is **one-way** if f can be computed by a polynomial time algorithm, but any polynomial time randomized algorithm F that attempts to compute a pseudo-inverse for f succeeds with negligible probability.[3] That is, for all randomized algorithms F , all positive integers c and all sufficiently large $n = \text{length}(x)$,

$$\Pr[f(F(f(x))) = f(x)] < n^{-c},$$

where the probability is over the choice of x from the discrete uniform distribution on $\{0,1\}^n$, and the randomness of F.[4]

Note that, by this definition, the function must be "hard to invert" in the average-case, rather than worst-case sense. This is different from much of complexity theory (e.g., NP-hardness), where the term "hard" is meant in the worst-case. That is why even if some candidates for one-way functions (described below) are known to be NP-complete, it does not imply their one-wayness. The latter property is only based on the lack of known algorithm to solve the problem.

It is not sufficient to make a function "lossy" (not one-to-one) to have a one-way function. In particular, the function that outputs the string of n zeros on any input of length n is *not* a one-way function because it is easy to come up with an input that will result in the same output. More precisely: For such a function that simply outputs a string of zeroes, an algorithm F that just outputs any string of length n on input $f(x)$ will "find" a proper preimage of the output, even if it is not the input which was originally used to find the output string.

Related concepts

A **one-way permutation** is a one-way function that is also a permutation—that is, a one-way function that is bijective. One-way permutations are an important cryptographic primitive, and it is not known if their existence is implied by the existence of one-way functions.

A trapdoor one-way function or trapdoor permutation is a special kind of one-way function. Such a function is hard to invert unless some secret information, called the *trapdoor*, is known.

A **collision-free hash function** f is a one-way function that is also *collision-resistant*; that is, no randomized polynomial time algorithm can find a collision—distinct values x, y such that $f(x) = f(y)$—with non-negligible probability.[5]

Theoretical implications of one-way functions

If f is a one-way function, then the inversion of f would be a problem whose output is hard to compute (by definition) but easy to check (just by computing f on it). Thus, the existence of a one-way function implies that FP≠FNP, which in turn implies that P≠NP. However, it is not known whether P≠NP implies the existence of one-way functions.

The existence of a one-way function implies the existence of many other useful concepts, including:

- Pseudorandom generators
- Pseudorandom function families
- Bit commitment schemes
- Private-key encryption schemes secure against adaptive chosen-ciphertext attack
- Message authentication codes
- Digital signature schemes (secure against adaptive chosen-message attack)

The existence of one-way functions also implies that there is no natural proof for P≠NP.

Candidates for one-way functions

The following are several candidates for one-way functions (as of April 2009). Clearly, it is not known whether these functions are indeed one-way; but extensive research has so far failed to produce an efficient inverting algorithm for any of them.

Multiplication and factoring

The function *f* takes as inputs two prime numbers p and q in binary notation and returns their product. This function can be "easily" computed in $O(b^2)$ time, where b is the total number of bits of the inputs. Inverting this function requires finding the factors of a given integer N. The best factoring algorithms known run in $O\left(\exp \sqrt[3]{\frac{64}{9} b (\log b)^2}\right)$ time, where b is the number of bits needed to represent N.

This function can be generalized by allowing p and q to range over a suitable set of semiprimes. Note that *f* is not one-way for randomly selected integers $p,q>1$, since the product will have 2 as a factor with probability 3/4 (because the probability that an arbitrary p is odd is 1/2, and likewise for q, so if they're chosen independently, the probability that both are odd is therefore 1/4; hence the probability that p or q is even is 1 - 1/4 = 3/4).

The Rabin function (modular squaring)

The **Rabin function**,[1]:57 or squaring modulo $N = pq$, where p and q are primes is believed to be a collection of one-way functions. We write

$$\mathrm{Rabin}_N(x) \triangleq x^2 \bmod N$$

to denote squaring modulo N: a specific member of the **Rabin collection**. It can be shown that extracting square roots, i.e. inverting the Rabin function, is computationally equivalent to factoring N (in the sense of polynomial-time reduction). Hence it can be proven that the Rabin collection is one-way if and only if factoring is hard. This also holds for the special case in which p and q are of the same bit length. The Rabin cryptosystem is based on the assumption that this Rabin function is one-way.

Discrete exponential and logarithm

Modular exponentiation can be done in polynomial time. Inverting this function requires computing the discrete logarithm. Currently there are several popular groups for which no known algorithm to calculate the underlying discrete logarithm in polynomial time is known. These groups are all finite abelian groups and the general discrete logarithm problem can be described as thus.

Let G be a finite abelian group of cardinality n. Denote its group operation by multiplication. Consider a primitive element $\alpha \in G$ and another element $\beta \in G$. The discrete logarithm problem is to find the positive integer k, where $1 \le k \le n$, such that:

$$\alpha^k = \underbrace{\alpha \cdot \alpha \cdot \ldots \cdot \alpha}_{k \text{ times}} = \beta$$

The integer k that solves the equation $\alpha^k = \beta$ is termed the **discrete logarithm** of β to the base α. One writes $k = \log_\alpha \beta$.

Popular choices for the group G in discrete logarithm cryptography are the cyclic groups $(\mathbf{Z}_p)^\times$ (e.g. ElGamal encryption, Diffie–Hellman key exchange, and the Digital Signature Algorithm) and cyclic subgroups of elliptic curves over finite fields (*see* elliptic curve cryptography).

An elliptic curve is a set of pairs of elements of a field satisfying $y^2 = x^3 + ax + b$. The elements of the curve form a group under an operation called "point addition" (which is not the same as the addition operation of the field). Multiplication kP of a point P by an integer k (*i.e.*, a group action of the additive group of the integers) is defined as repeated addition of the point to itself. If k and P are known, it is easy to compute $R = kP$, but if only R and P are known, it is assumed to be hard to compute k.

Cryptographically secure hash functions

There are a number of cryptographic hash functions that are fast to compute, such as SHA 256. Some of the simpler versions have fallen to sophisticated analysis, but the strongest versions continue to offer fast, practical solutions for one-way computation. Most of the theoretical support for the functions are more techniques for thwarting some of the previously successful attacks.

Other candidates

Other candidates for one-way functions have been based on the hardness of the decoding of random linear codes, the subset sum problem (Naccache-Stern knapsack cryptosystem), or other problems.

Universal one-way function

There is an explicit function f that has been proved to be one-way, if and only if one-way functions exist.[6] In other words, if any function is one-way, then so is f. Since this function was the first combinatorial complete one-way function to be demonstrated, it is known as the "universal one-way function". The problem of finding a one way function is thus reduced to proving that one such function exists.

See also

- One-way compression function
- Cryptographic hash function
- Geometric cryptography
- Trapdoor function

References

1. Oded Goldreich (2001). Foundations of Cryptography: Volume 1, Basic Tools, (draft available (http://www.wisdom.weizmann.ac.il/~oded/PSBookFrag/part2N.ps) from author's site). Cambridge University Press. ISBN 0-521-79172-3. (see also wisdom.weizmann.ac.il (http://www.wisdom.weizmann.ac.il/~oded/foc-book.html))
2. Goldwasser, S. and Bellare, M. "Lecture Notes on Cryptography" (http://cseweb.ucsd.edu/~mihir/papers/gb.html). Summer course on cryptography, MIT, 1996–2001
3. For the meaning of $\{0,1\}^*$ see Kleene star.
4. Many authors view this definition as strong one-way function. Weak one-way function can be defined similarly except that the probability that every adversarial F fails to invert f is noticeable. However, one may construct strong one-way functions based on weak ones. Loosely speaking, strong and weak versions of one-way function are equivalent theoretically. See Goldreich's Foundations of Cryptography, vol. 1, ch 2.1–2.3.
5. Russell, A. (1995). "Necessary and Sufficient Conditions for Collision-Free Hashing". *Journal of Cryptology*. **8** (2): 87–99. doi:10.1007/BF00190757 (https://doi.org/10.1007%2FBF00190757).
6. Leonid Levin (2003). "The Tale of One-Way Functions". ACM. arXiv:cs.CR/0012023 (https://arxiv.org/abs/cs.CR/0012023).

Further reading

- Jonathan Katz and Yehuda Lindell (2007). *Introduction to Modern Cryptography*. CRC Press. ISBN 1-58488-551-3.
- Michael Sipser (1997). *Introduction to the Theory of Computation*. PWS Publishing. ISBN 978-0-534-94728-6. Section 10.6.3: One-way functions, pp. 374–376.
- Christos Papadimitriou (1993). *Computational Complexity* (1st ed.). Addison Wesley. ISBN 978-0-201-53082-7. Section 12.1: One-way functions, pp. 279–298.