

Pseudorandomness

You can support Wikipedia by making a tax-deductible donation.

From Wikipedia, the free encyclopedia
(Redirected from Pseudorandom)

A **pseudorandom** process is a process that appears random but is not. Pseudorandom sequences typically exhibit statistical randomness while being generated by an entirely deterministic causal process. Such a process is easier to produce than a genuine random one, and has the benefit that it can be used again and again to produce exactly the same numbers, useful for testing and fixing software.

To date there is no known method to produce true randomness, because due to the very nature of randomness, any factor determining the outcome would mean that it is not random at all. The random number generation functions provided in all software packages are therefore pseudorandom.

Contents

- 1 History
- 2 Almost random
- 3 Complexity-based pseudorandomness
 - 3.1 Definition
- 4 Cryptography
- 5 Monte Carlo method simulations
- 6 See also
- 7 External links

History

The generation of random numbers has many uses (mostly in statistics, for random sampling, and simulation); originally researchers needing random numbers would generate them through various means - dice, cards, roulette wheels ... or use existing random number tables.

The first attempt to provide researchers with a ready supply of random digits was in 1927, when the Cambridge University Press published a table of 41,600 digits developed by Leonard H.C. Tippet. In 1947, the RAND Corporation generated numbers by the electronic simulation of a roulette wheel; the results were eventually published in 1955 as *A Million Random Digits with 100,000 Normal Deviates*.

John von Neumann was a pioneer in computer-based random number generators. In 1951, Derrick Henry Lehmer invented the linear congruential generator, used in most pseudorandom number generators today. With the spread of the use of computers, algorithmic pseudorandom number generators replaced random number tables, and "true" random number generators (Hardware random number generators) are only used in a few cases.

Almost random

Anyone who considers arithmetical methods of producing random digits is, of course, in a state of sin. — John von Neumann (1951)

A pseudo-random variable is a variable which is created by a deterministic procedure (often a computer program or subroutine) which (generally) takes random bits as input. The pseudo-random string will typically be longer than the original random string, but less random (less entropic, in the information theory sense). This can be useful for randomized algorithms.

Pseudorandom number generators are widely used in such applications as computer modeling (e.g., Markov chains), statistics, experimental design, etc. Some of them are sufficiently random to be useful in these applications. Many are not, and considerable sophistication is required to correctly determine the difference for any particular purpose. Incautious use of readily available random number generators has caused considerable, and long sustained, damage to the worth of large numbers of research projects for many years. The RANDU generator routine available on many large mainframe computers for decades had considerable, widely unappreciated, faults.

Complexity-based pseudorandomness

A distribution is considered to be pseudorandom if it cannot be distinguished from the truly (Uniform) random distribution by any efficient (polynomial time) procedure or circuit. The **uniform distribution**, for a length parameter n assigns each n -bit string $x \in \{0, 1\}^n$ with equal probability of 2^{-n} . Pseudorandom distributions can be generated deterministically from short random seeds, which are much shorter than the length of the pseudorandom output.

A method for distinguishing two distributions from each other is by taking their statistical distance. If two distributions $p = [p_1, \dots, p_N]$ and $q = [q_1, \dots, q_N]$ have a very small statistical distance $\|p - q\|_1/2$, there exists no circuit, S , such that S can distinguish them well, even with no restriction on the size of S . Also, if the size of S is too small then it may not be able to distinguish some distributions that are very different.

Definition

A distribution ensemble D_n is $(S(n), \epsilon(n))$ pseudorandom if, for any circuit C of size $\leq S(n)$, with U_n as the uniform random distribution, then

$$|\Pr_{x \in U_n}[C(x) = 1] - \Pr_{x \in D_n}[C(x) = 1]| \leq \epsilon(n)$$

D_n is called pseudorandom if it is pseudorandom for all $S(n) = poly(n)$ and $\epsilon(n) = 1/O(poly(n))$. This definition of pseudorandomness is used primarily in the study of pseudorandom generators.

Cryptography

See also: Cryptographically secure pseudorandom number generator

For such applications as cryptography, the use of pseudorandom number generators (whether hardware or software or some combination) is insecure. When random values are required in cryptography, the goal is to make a message as hard to crack as possible, by eliminating or obscuring the parameters used to encrypt the message (the key) from the message itself or from the context in which it is carried. Pseudorandom sequences are deterministic and reproducible; all that is required to discover and reproduce a pseudorandom sequence is the algorithm used to generate it and the initial seed. So the entire sequence of numbers is only as powerful as the randomly chosen parts - sometimes the algorithm and the seed, but usually only the seed.

There are many examples in cryptographic history of cyphers, otherwise excellent, in which random choices were not random enough and security was lost in direct consequence. The World War II Japanese PURPLE cypher machine used for diplomatic communications is a good example. It was consistently broken throughout WWII, and indeed from somewhat before the United States entered that war, mostly because the "key values" used were insufficiently random. They had patterns, and those patterns made any intercepted traffic readily decryptable. Had keys (ie, the initial settings of the stepping switches in the machine) been made unpredictably (ie, randomly), that traffic would have been much harder to break, and perhaps even secure in practice.

Users and designers of cryptography are strongly cautioned to treat their randomness needs with the utmost care. Absolutely nothing has changed with the era of computerized cryptography, except that patterns in pseudorandom data are easier to discover than ever before. Randomness is, if anything, more important than ever.

Monte Carlo method simulations

Computers are used to simulate everything from nuclear reactions to the economy. These are examples of Monte Carlo method Simulations. A Monte Carlo method Simulation is defined as any method that utilizes sequences of random numbers to perform the simulation. Other simulations include quantum chromo dynamics, radiation cancer therapy, traffic flow, stellar evolution and VLSI design. All these simulations require the use of random numbers and therefore pseudorandom number generators, which makes creating random like generators very important.

An easy example of how a computer would do a Monte Carlo method Simulation is the calculation of π . If a square enclosed a circle and a point were randomly chosen inside the square the point would either lie inside the circle or outside it. If the process were repeated many times, you can see that the ratio of the random points that lie inside the circle to outside it is proportional to ratio of the circle area to the square area. From this we can estimate π .

See also

- Disperser
- Expander graph
- Extractor
- Random variable
- PN Sequences
- Pseudo-random binary sequence
- Pseudorandom number generator
- List of random number generators

External links

- Random number history
- Using and Creating Cryptographic-Quality Random Numbers
- In RFC 1750, the use of pseudo-random number sequences in cryptography is discussed at length.
- In Donald E. Knuth, *The Art of Computer Programming, Volume 2: Seminumerical Algorithms* (3rd edition), 1997. Addison-Wesley Professional, ISBN 0-201-89684-2

Retrieved from "http://en.wikipedia.org/wiki/Pseudorandomness"

Categories: Cleanup from May 2007 | All pages needing cleanup | All articles with unsourced statements | Articles with unsourced statements since July 2007 | Pseudorandomness

- This page was last modified 08:18, 21 September 2007.
- All text is available under the terms of the GNU Free Documentation License. (See **Copyrights** for details.) Wikipedia® is a registered trademark of the Wikimedia Foundation, Inc., a U.S. registered 501(c)(3) tax-deductible nonprofit charity.