

# Randomness:

Prowessful Counterassertion, Intercommon  
Sunshining in Synentognathous Crossbreeds

Seth Hardy

Tsumego Foundation  
2193 Commonwealth Ave.  
Suite 336  
Boston, MA 02135

Rubi Con 5  
March 29, 2003

# Overview

- What do *random* and *pseudorandom* mean? How about *quasirandom*?
- What is *entropy*? ( $H$ , not  $\Delta s$ .)
- What is a *pseudorandom number generator (PRNG)*?
- How can I tell whether a PRNG is good? What does “good” mean in this context?
- Is it possible to manipulate randomness?

# ...synentognathous crossbreeds?

First, a little explanation of the title of this talk:

```
perl -e '$i=1; while (<>) {  
if (rand() < 1/$i++) { $best = $_; } }  
print $best;' /usr/share/dict/words
```

...but is this actually *random*? What exactly does “random” mean?

# Random? Sequence 1

Let's start by looking at a simple sequence:

**1, 1, 1, 1, 1, . . .**

Is this random?

# Random? Sequence 1

Let's start by looking at a simple sequence:

**1, 1, 1, 1, 1, . . .**

Is this random?

We need to define the set we're picking from. What if  $S = \{1\}$ ?

**Yes!** This is a random sequence, if we are picking from the above set.

# Random? Sequence 2

Now, let's assume that  $S = \{1, 2, 3, 4\}$ , and look at the same sequence:

**1, 1, 1, 1, 1, . . .**

Is this still random?

# Random? Sequence 2

Now, let's assume that  $S = \{1, 2, 3, 4\}$ , and look at the same sequence:

**1, 1, 1, 1, 1, . . .**

Is this still random?

We still haven't defined the probability of picking each  $x \in S$ . What happens if we pick according to these probabilities:

$$\Pr[x = 1] = 1 \quad \Pr[x = 2] = 0 \quad \Pr[x = 3] = 0 \quad \Pr[x = 4] = 0$$

**Yes!** This is also a random sequence, according to the above set and probability distribution.

# Uniform Distribution

We can pick randomly according to any “probability distribution”:

$$\mathcal{D} : S \rightarrow \mathbb{R}$$

The probability distribution  $\mathcal{D}$  assigns a nonnegative probability to each  $x \in S$ , such that

$$\sum_{x \in S} \mathcal{D}(x) = 1$$

What if we want to pick something “*at random*”?

When most people say “at random”, what’s usually meant is “*uniformly at random*”. The *uniform distribution*  $\mathcal{U}$  is the probability distribution where everything is picked equally often:

$$\text{If } |S| = n, \text{ then } \Pr[x = s] = \frac{1}{n} \text{ for each } s \in_{\mathcal{U}} S.$$

# PRNGs

What is a “pseudorandom number generator”?

$$G : \{0, 1\}^k \rightarrow \{0, 1\}^n$$

However, we'd like to see  $G$  have a few specific properties for it to be *useful*:

- $n$  larger than  $k$
- $G(x)$  computationally indistinguishable from random
- Hard to predict output even with some knowledge of the system.

So what would a good measure of evaluating these properties be?

# Statistical Distance

How can we tell how “far apart” distributions are?

The *statistical distance* (also known as the  $L_1$  metric) between two probability distributions  $\mathcal{D}$  and  $\mathcal{E}$  is:

$$d(\mathcal{D}, \mathcal{E}) = \frac{1}{2} \left| \sum_{x \in S} \mathcal{D}(x) - \mathcal{E}(x) \right|$$

Often we want to see how close a distribution is to the uniform distribution  $\mathcal{U}$ . If  $d(\mathcal{D}, \mathcal{U}) \leq \epsilon$ , then we say  $\mathcal{D}$  is  $\epsilon$ -close to uniform.

Alternatively, we could say  $\mathcal{D}$  is *quasirandom within  $\epsilon$* .

# Entropy

*Entropy* is a common term used when looking at randomness.

But what exactly is entropy?

- A measure of information?
- A measure of randomness?
- A measure of redundancy?

What about different types of entropy? Shannon entropy, Renyi entropy, min entropy...

# Shannon Entropy

The *Shannon entropy*  $H$  (often just called *entropy*) is the basic measure of information:

$$H(\mathcal{D}) = - \sum_{x \in S} \mathcal{D}(x) \log_2 \mathcal{D}(x)$$

Shannon entropy is measured in bits per “symbol” (each element in  $S$ ).

For example,  $H(\text{English}) = 2.62$ . (We are looking here at the probability distribution over the set  $S = \{A, B, C, \dots, Z\}$ .)

However,  $\log_2 26 \approx 4.70$ , showing that there are appx. two bits of redundant information in each English character.

# Renyi Entropy

*Renyi entropy* looks at how often one gets a “collision” when choosing values randomly according to a particular distribution.

The *collision probability* of a distribution  $\mathcal{D}$  is:

$$\Delta_p = \sum_{x \in S} (\mathcal{D}(x))^2$$

The Renyi entropy  $H_{\text{Ren}}$  of a distribution  $\mathcal{D}$  is:

$$H_{\text{Ren}}(\mathcal{D}) = -\log_2 \sum_{x \in S} (\mathcal{D}(x))^2 = -\log_2 \Delta_p$$

# Min Entropy

*Min entropy*  $H_\infty$  can be thought of as a measure of the worst possible case of a probability distribution:

$$H_\infty(\mathcal{D}) = \min\{-\log_2 \mathcal{D}(x) : x \in S\} = -\log_2 \max\{\mathcal{D}(x) : x \in S\}$$

It is possible for a distribution to have a fairly high Shannon entropy, but a small min entropy.

For example, let  $\mathcal{D}(x) = \frac{1}{2}$  for some  $x \in S$  and some very small probability for all other  $x' \in S$ .

# NIST Tests

The National Institute of Standards and Technology list 16 statistical tests which any government-used PRNG must pass.

The statistical tests are run on the output of a PRNG. Each test rejects a small number of the total bitstrings. Each test is designed to catch common errors in PRNG design, or bitstrings with a certain form.

Passing the tests doesn't mean that the output is random (or even close to random). It means that the output isn't obviously non-random.

Frequency Test	Universal Statistical Test
Block Frequency Test	Lempel-Ziv Compression Test
Runs Test	Linear Complexity Test
Ones Block Runs Test	Serial Test
Binary Matrix Rank Test	Approximate Entropy Test
Spectral Test	Cumulative Sums Test
Non-overlapping Template Matching	Random Excursions Test
Overlapping Template Matching	Random Excursions Variant Test

# Chewbacca



What a wookiee!

# Nisan Generator

PRNG created by Noam Nisan with some interesting properties.

Setup:

- Use an alphabet  $Q$  of size (in bits)  $t$ , and choose a number of “levels”  $k$  to run the generator over.
- Choose functions  $h_1, h_2, \dots, h_k$  of the form  $h_i(x) = a_i x + b_i$ .

Generation:

$$G_0(x) = x$$

$$G_{\ell+1}(x, h_1, \dots, h_{\ell+1}) = G_{\ell}(x, h_1, \dots, h_{\ell}) \circ G_{\ell}(h_{\ell+1}(x), h_1, \dots, h_{\ell})$$

The output of the generator is  $G_k(x, h_1, \dots, h_k)$ .

# Nisan Generator

Takes  $t(2k+1)$  bits as input, gives  $t2^k$  bits as output. Exponential generation!

## **Bad News:**

This generator isn't very secure...  $x$  is the first output given, the functions  $h_i$  can be determined by watching the outputs and solving systems of equations...

## **Good News:**

This generator fools all statistical tests in LOGSPACE...

...this includes **all** of the NIST tests!

# Manipulating Randomness

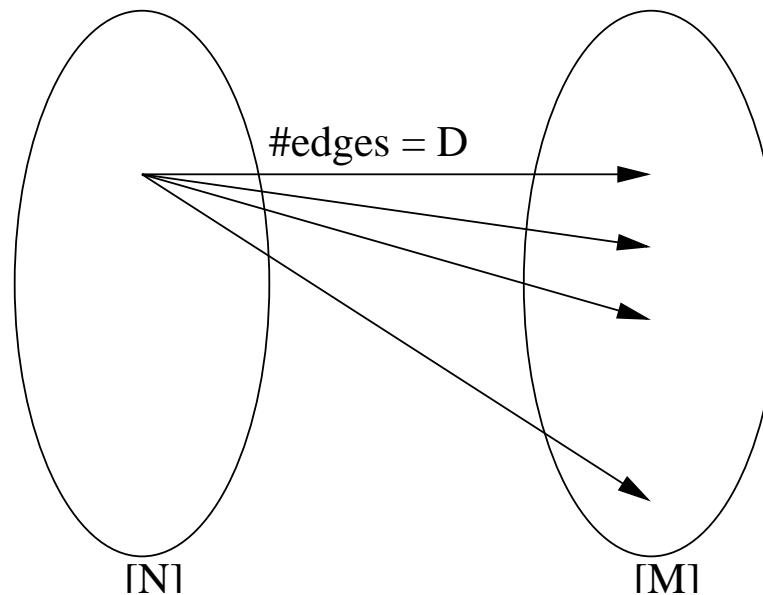
Why would we want to 'manipulate' randomness?

- Create better PRNGs
- Smooth out bad distributions
- Better problems for cryptography  
(hard on average vs. hard in worst case)
- Derandomize probabilistic algorithms
- The big question: does  $\mathcal{P} = \mathcal{BPP}$ ?

# Dispersers and Extractors

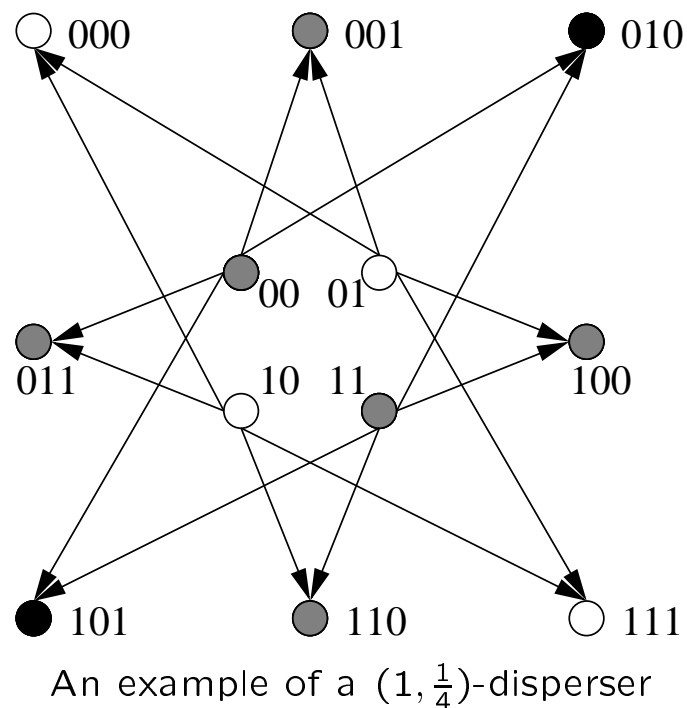
Family of graphs with “random-like” properties.

Bipartite graphs that go from  $N = 2^n$  vertices to  $M = 2^m$  vertices. Each vertex on the left has degree  $D = 2^d$ .



# Dispersers

A  $(k, \epsilon)$ -disperser takes at least  $2^k$  vertices in  $[N]$ , and “disperses” them to at least  $(1 - \epsilon)|M|$  vertices in  $[M]$ .



# Extractors

**Definition.** A graph  $G = ([N], [M], E)$  is a  $(k, \epsilon)$ -extractor if, for any probability distribution  $\mathcal{D}$  on  $[N]$  with  $H_\infty(\mathcal{D}) \geq k$ ,  $\Gamma(\mathcal{D})$  is  $\epsilon$ -close to uniform on  $[M]$ .

*...what does this mean?*

Extractors take a “bad” distribution on  $[N]$  and random bits, and use the additional randomness to “smooth” out the distribution into a better one over  $[M]$ .

How “bad” is the input distribution  $\mathcal{D}$ ? – What is the min-entropy of  $\mathcal{D}$ ?

How “good” is the output distribution  $\mathcal{E}$ ? – How close to uniform is  $\mathcal{E}$ ?

# Summary

- Probability distributions, statistical distance, quasirandomness...
- Entropy: Shannon, Renyi, Min
- Statistical and “real world” tests
- Chewbacca
- A PRNG that fools all the government tests
- Manipulating randomness

Questions?