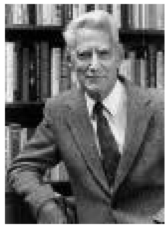
Lecture 2: Perfect Ciphers (in Theory, not Practice)

Shannon was the person who saw that the binary digit was the fundamental element in all of communication. That was really his discovery, and from it the whole communications revolution has sprung.
R G Gallager



Claude Shannon, 1916-2001

I just wondered how things were put together.
Claude Shannon

CS588: Cryptology
University of Virginia
Computer Science

David Evans
http://www.cs.virginia.edu/~evans

Menu

- Survey Results
- Perfect Ciphers
- Entropy and Unicity

Survey Results

- Forged email: 7 out of 34
- Broken into systems: 5 out of 34
  - All socially responsible, of course
- Victim: 10 out of 34

  "hopefully not, but if they did a good job I probably would never have noticed it."
- Movies/Books: Sneakers (10 – "should be required for the course"), Cryptonomicon (5), Hackers (3), Matrix (2), Mercury Rising (2), Crypto (2), Takedown, Enemy of the State, Cuckoo's Egg, Maximum Security, The Net, Dr. Strangelove, Office Space, "Swordfish was really really bad"

Survey: Requested Topics

- SSL, PGP, RSA
- Network and web security, e-commerce
- Quantum Computing
- Banks, ATMs
- "All the NSA secrets"

Last Time

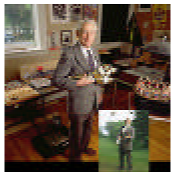
- Big keyspace is not necessarily a strong cipher
- Claim: One-Time Pad is **perfect cipher**
  - In theory: depends on perfectly random key, secure key distribution, no reuse
  - In practice: usually ineffective (VENONA, Lorenz Machine)
- Today: what does is mean to be a perfect cipher?

Ways to Convince

- "I tried really hard to break my cipher, but couldn't. I'm a genius, so I'm sure no one else can break it either."
- "Lots of really smart people tried to break it, and couldn't."
- Mathematical arguments – key size (dangerous!), statistical properties of ciphertext, depends on some provably (or believed) hard problem
- Invulnerability to known cryptanalysis techniques (but what about undiscovered techniques?)
- Show that ciphertext could match multiple reasonable plaintexts without knowing key
  - Simple monoalphabetic secure for about 10 letters of English:

```
XBCF CF FWPHGW
This is secure
Spat at troner
```

Claude Shannon



- Master's Thesis [1938] – boolean algebra in electronic circuits
- "Mathematical Theory of Communication" [1948] – established information theory
- "Communication Theory of Secrecy Systems" [1945/1949] (linked from manifest)
- Invented rocket-powered Frisbee, could juggle four balls while riding unicycle

Entropy

Amount of information in a message

$H(M) = - \Sigma P(M_i) \log P(M_i)$

over all possible messages M_i

If there are n equally probable messages,

$H(M) = - \Sigma 1/n \log 1/n$

$= - (n * (1/n \log 1/n))$

$= - (1 \log 1/n) = \log n$

Base of log is alphabet size, so for binary:

$H(M) = \log_2 n$

where n is the number of possible meanings

Entropy Example

M = months of the year

H(M) =

$= \log_2 12 \approx 3.6$ (need 4 bits to encode a year)

Rate

- Absolute rate: how much information can be encoded

  $R = \log_2 Z$ (Z=size of alphabet)

  $R_{English} = \log_2 26 \approx 4.7$ bits / letter
- Actual rate of a language:

  $r = H(M) / N$

  M is an N-letter message.

  r of months spelled out using ASCII:

  $= \log_2 12 / (8 \text{ letters} * 8 \text{ bits/letter}) \approx 0.06$

Rate of English

- r(English) is about .28 letters/letter (1.3 bits/letter)
  - How do we get this?
- How many meaningful 20-letter messages in English?

  $r = H(M) / N$

  $.28 = H(M)/20$

  $H(M) = 5.6 = \log_{26} n$

  $n = 26^{5.6} \sim 83$ million (of $2*10^{28}$ possible)

  Probability that 20-letters are sensible English is

  About 1 in $2 * 10^{20}$

Redundancy

- Redundancy (D) is defined:

  D = R – r
- Redundancy in English:

  D = 1 - .28 = .72 letters/letter

  D = 4.7 – 1.3 = 3.4 bits/letter

  Each letter is 1.3 bits of content, and 3.4 bits of redundancy. (~72%)
- 7-bit ASCII

  D = 7 – 1.3 = 5.7

  81% redundancy, 19% information

Unicity Distance

- Entropy of cryptosystem: (K = number of possible keys)

  $H(K) = \log_{\text{Alphabet Size}} K$

  if all keys equally likely

  $H(\text{64-bit key}) = \log_2 2^{64} = 64$

- Unicity distance is defined as:

  $U = H(K)/D$

  Expected *minimum* amount of ciphertext needed for *brute-force attack* to succeed.

Unicity Examples

- One-Time Pad

  H(K) = infinite

  U = H(K)/D = infinite

- Monoalphabetic Substitution

  $H(K) = \log_2 26!$ » 87

  D = 3.4 (redundancy in English)

  U = H(K)/D » 25.5

  Intuition: if you have 25 letters, probably only matches one possible plaintext.

  D = 0 (random bit stream)

  U = H(K)/D = infinite

Unicity Distance

- Probabilistic measure of how much ciphertext is needed to determine a unique plaintext
- Does **not** indicate how much ciphertext is needed for cryptanalysis
- If you have less than unicity distance ciphertext, can't tell if guess is right.
- As redundancy approaches 0, hard to cryptanalyze even simple cipher.

Shannon's Theory [1945]

Message space: $\{ M_1, M_2, ..., M_n \}$

Assume finite number of messages

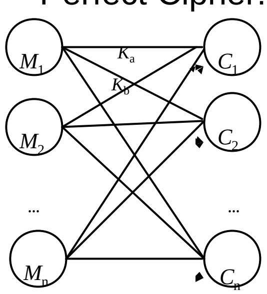
Each message has probability

$p(M_1) + p(M_2) + ... + p(M_n) = 1$

Key space: $\{ K_1, K_2, ..., K_l \}$

(Based on Eli Biham's notes)

Perfect Cipher: Definition



A perfect cipher: there is some key that maps any message to any ciphertext with equal probability.

For any i, j: $p(M_i|C_j) = p(M_i)$

Conditional Probability

$P(B | A)$ = The probability of B, given that A occurs

P (coin flip is tails) = ½

P (coin flip is tails | last coin flip was heads) = ½

P (today is Monday | yesterday was Sunday) = 1

P (today is a weekend day | yesterday was a workday) = 1/5

Calculating Conditional Probability

$$P(B \mid A) = \frac{P(A \cap B)}{P(A)}$$

$$\text{P (coin flip is tails | last coin flip was heads)} = \frac{\text{P (coin flip is tails \textbf{and} last coin flip was heads)}}{\text{P(last coin flip was heads)}}$$

$$= (½ * ½) / ½ = ½$$

P (today is a weekend day | yesterday was a workday)
= P (today is a weekend day **and** yesterday was a workday) / P (yesterday was a workday)
= P (today is a weekend day) * P(yesterday was a workday) / P (yesterday was a workday)
= 2/7 * 5/7 / 5/7 = 2/7

Wrong!

$P(A \cap B) = P(A) * P(B)$
only if A and B are *independent events*

Perfect Cipher

Definition: $\forall$ i, j: $P(M_i \mid C_j) = P(M_i)$

A cipher is perfect iff:

$$\forall M, C \quad P(C \mid M) = P(C)$$

Or, equivalently:

$$\forall M, C \quad P(M \mid C) = P(M)$$

Perfect Cipher

$$\forall M, C \quad P(C \mid M) = P(C)$$
$$\forall M, C \quad P(C) = \sum_{K,\ E_K(M) = C} P(K)$$

Or:
$\forall C \quad \sum_{K,\ E_K(M) = C} P(K)$ is independent of M

Without knowing anything about the key, any ciphertext is equally likely to match and plaintext.

Example: Monoalphabetic

- Random monoalphabetic substitution for one letter message:

  $\forall$ C,M: $p(C) = p(C \mid M) = 1/26$.

Example: One-Time Pad

For each bit:
$p(C_i = 0) = p(C_i = 0 \mid M_i = 0) = p(C_i = 0 \mid M_i = 1) = ½$
since $C_i = K_i \oplus M_i$
$p(K_i \oplus M_i = 0) = p(K_i = 1) * p(M_i = 1) + p(K_i = 0) * p(M_i = 0)$
Truly random K means $p(K_i = 1) = p(K_i = 0) = ½$
$= ½ * p(M_i = 1) + ½ * p(M_i = 0)$
$= ½ * (p(M_i = 1) + p(M_i = 0)) = ½$
All key bits are independent, so:
$p(C) = p(C \mid M)$ QED.

Perfect Cipher Keyspace Theorem

Theorem: If a cipher is perfect, there must be at least as many keys (l) are there are possible messages (n).

Proof by Contradiction

Suppose there is a perfect cipher with $l < n$. (More messages than keys.)

Let C_0 be some ciphertext with $p(C_0) > 0$.

There exist

m messages M such that $M = D_K(C_0)$

$n - m$ messages M_0 such that $M_0 \neq D_K(C_0)$

We know $1 \leq m \leq l < n$ so $n - m > 0$ and there is at least one message M_0.

Proof, cont.

Consider the message M_0 where $M_0 \neq D_K(C_0)$ for any K.

So,

$p\ (C_0 \mid M_0) = 0$.

In a perfect cipher,

$p\ (C_0 \mid M_0) = p\ (C_0) > 0$.

Contradiction! It isn't a perfect cipher.

Hence, all perfect ciphers must have $l \geq n$.

Example: Monoalphabetic

Random monoalphabetic substitution is **not** a perfect cipher for messages of up to 20 letters:

$l = 26!$ $n = 26^{20}$

$l < n$ its not a perfect cipher.

In previous proof, could choose C_0 = "AB" and M_0 = "ee" and $p\ (C_0 \mid M_0) = 0$.

Example: Monoalphabetic

Is random monoalphabetic substitution a perfect cipher for messages of up to 2 letters?

$l = 26!$ $n = 26^2$

$l \geq n$.

No! Showing $l \geq n$ does not prove its perfect.

Summary

- Cipher is perfect: $\forall$ i, j: $p\ (M_i|C_j) = p\ (M_i)$
  Given any ciphertext, the probability that it matches any particular message is the same.
- Equivalently, $\forall$ i, j: $p\ (C_i|M_j) = p\ (C_i)$
  Given any plaintext, the probability that it matches any particular ciphertext is the same.

Imperfect Cipher

- To prove a cipher is imperfect:
  - Find a ciphertext that is more likely to be one message than another
  - Show that there are more messages than keys
    - Implies there is some ciphertext more likely to be one message than another even if you can't find it.

Charge

- Problem Set 1: due next Monday
  - Next lecture will help with Question 5a,b
  - All other questions covered (as much as we will cover them in class) already
- Next time:
  - Project Kickoff
  - Enigma