

## Encryption Algorithms

A. Transposition ciphers rearrange characters or bits.

**Example:** Write rows, read columns of a matrix.

There is a fixed period,  $d$ , say. If we assume all  $d!$  permutations to be equally likely, then the unicity distance is

$$N = \frac{H(K)}{D} = \frac{\log_2(d!)}{D} \approx \frac{d \log_2(d/e)}{3.2}$$

or  $N \approx 0.3d \log_2(d/e)$ .

**Example:** With a  $3 \times 9$  matrix we have  $d = 27$  and  $N = 27.9$ .

Use digrams to crack transposition ciphers. The process is called *anagramming*.

Use frequency counts to distinguish Transposition ciphers from Substitution ciphers.

## B. Substitution Ciphers

Replace (blocks of) characters by other characters. Four types:

1. Simple: Replace  $m_i$  by  $c_i$ .
2. Homophonic: Replace  $m_i$  by a random one of many possible  $c_i$ .
3. Polyalphabetic: Use multiple maps from the plaintext alphabet to the ciphertext alphabet.
4. Polygram: Make arbitrary substitution for groups of characters.

1. Simple: Replace  $m_i$  by  $c_i$ . Write  $f(m) = c$ .

**Example.** Caesar cipher—rotate the alphabet:  $f(m) = (m + k) \bmod n$ , where  $n$  is the alphabet size. The unicity distance is  $H(K)/D = (\log_2 26)/3.2 \approx 1.5$  letters.

If all  $n!$  permutations of the alphabet are equally likely (the best case) in a simple substitution cipher, then the unicity distance is  $\log_2(n!)/D$ . For English,  $n = 26$  and the unicity distance would be  $\log(26!)/3.2 \approx 27.6$ . This is the case for the Cryptoquote in the *Exponent*.

These ciphers may be broken with frequency analysis and trial and error.

An *affine cipher*,  $f(m) = (am + b) \bmod n$ , guess some two-letter pairs and solve two congruences in two unknowns  $a$  and  $b$ .

2. Homophonic: Replace  $m_i$  by a random one of many possible  $c_i$ .

To confound the frequency analysis which succeeds so well for simple substitution ciphers, one might use a ciphertext alphabet larger than the plaintext alphabet and assign each plaintext letter  $a$  to a subset (*homophone*)  $f(a)$  of the ciphertext alphabet. To permit deciphering, require  $f(a) \cap f(b) = \emptyset$  when  $a \neq b$ . Encipher each  $m_i$  in the plaintext as a randomly chosen  $c_i \in f(m_i)$ .

Usually, the ciphertext alphabet is much larger than the plaintext alphabet and the size of  $f(a)$  is proportional to the frequency of occurrence of  $a$  in English. Then the letters of the ciphertext alphabet have a uniform distribution in the ciphertext. Use digrams to break.

One can define  $f$  via a standard text using the number of an instance of the letter as its cipher. Example: the Beale ciphers.

One can encipher two plaintext messages of equal length together using a  $26 \times 26$  matrix of ciphers. One cannot tell which message it is without the key.

3. Polyalphabetic: Use multiple maps  $f_i$  from the plaintext alphabet to the ciphertext alphabet. Encipher  $M = m_1m_2\dots$  as  $C = f(m_1)f(m_2)\dots$ .

Let  $n$  be the length of the alphabet. The sequence  $\{f_i\}$  may be periodic, perhaps defined by a *keyword*  $K = k_1\dots k_d$ .

**Example:** Vigenère cipher:  $f_i(a) = (a + k_i) \bmod n$ .

M: RENA ISSA NCE

K: BAND BAND BAN

C: SEAD JSFD OCR

**Example:** Beaufort cipher:  $f_i(a) = (k_i - a) \bmod n$ .

If the period of the key is  $d$ , then the unicity distance is  $H(K)/D = \log_2(n^d)/D = (d/D) \log_2 n$ . For English, this is  $d \log_2(26)/3.2 \approx 1.47d$ .

There are two basic methods to break periodic polyalphabetic substitution ciphers.

**Kasiski Method:** Look for repetitions in cipher text. They might occur at multiples of the period  $d$ . Look at the gcd of some of the differences.

**The Index of Coincidence Method:** Measure frequency variations of letters to guess the period  $d$ . Let  $\{a_0, a_1, \dots, a_{n-1}\}$  be the (plain or ciphertext) alphabet. Let  $F_i$  be the frequency of occurrence of  $a_i$  in a ciphertext of length  $N$ . Define the Index of Coincidence as

$$IC = \left( \sum_{i=0}^{n-1} \frac{F_i(F_i - 1)}{2} \right) / \frac{N(N - 1)}{2}.$$

Then  $IC$  represents the probability that two letters chosen at random in the ciphertext are the same.

One can estimate  $IC$  theoretically in terms of the period  $d$ . For English and a polyalphabetic cipher with period  $d$ , the expected value of  $IC$  is

$$\frac{1}{d} \frac{N-d}{N-1} (0.066) + \frac{d-1}{d} \frac{N}{N-1} (0.038).$$

Guess  $d$  by comparing the  $IC$  with this table:  $(d, IC)$ : (1, 0.066), (2, 0.052), (3, 0.047), (4, 0.045), (5, 0.044), (10, 0.041), (infinity, 0.038).