Cryptography

Alec Zabel-Mena

February 9, 2022

Contents

Chapter 1

Classical Cryptography.

1.1 Simple Cryptosystems and Classical Ciphers.

Definition. We define a **cryptosystem** to be a triple $(\mathcal{P}, \mathcal{C}, \mathcal{K})$ where $\mathcal{P}$ and $\mathcal{C}$ are called the **plain text space** and **cipher text space**; and $\mathcal{K}$, called the **key space** is such that, for any $K \in \mathcal{K}$, there exist maps $e_K : \mathcal{P} \to \mathcal{C}$, and $d_K : \mathcal{C} \to \mathcal{P}$ such that $d_K e_K(x) = x$ for every $x \in \mathcal{P}$. We call the elements of $\mathcal{P}$ **plain texts**, the elements of $\mathcal{C}$ **cipher texts**, and the elements of $\mathcal{K}$ **keys**. We call e_K and d_K the **encryption rule** and **decryption rule**, respectively. We call the pait (e_K, d_K) the **cipher**.

Remark. It is important to note, that in this definition, we take $\mathcal{P}$ and $\mathcal{C}$ to be arbitrary sets. However, in practice, they will usually result to be vector spaces of fields like $\mathbb{F}_2$. Recall that with computers, if we encrypt a message like `Hello, world!`, we are encrypting a string of bits.

Remark. The property that $d_K e_K(x) = x$ just implies that e_K is the right inverse of d_K. This howver, does not assert that $e_K = d_K^{-1}$.

The figure below outlines a communications channel between two parties Alice and Bob. Here, Alice and Bob both agree on a protocol that uses a specified cyrptosystem. They take a key K from the set $\mathcal{K}$ of the system. This key then used by the ecnryptor, and by the decryptor (through a secure channel) to encrypt and decrypt messages.

Oscar, who can intercept their encrypted communications cannot read them without the key. For this reason, it is important to choose the key K in a secure setting. Here, the

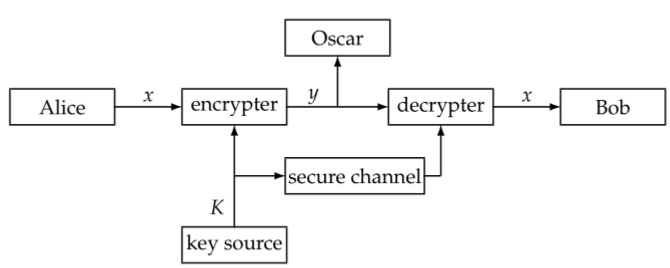

Figure 1.1: Encrypted Communication Channel between Alice and Bob.

A	B	C	D	E	F	G	H	I	J	K	L	M
0	1	2	3	4	5	6	7	8	9	10	11	12

N	O	P	Q	R	S	T	U	V	W	X	Y	Z
13	14	15	16	17	18	19	20	21	22	23	24	25

Figure 1.2: The mapping of Q onto $\mathbb{Z}/26\mathbb{Z}$.

encyrptor and the decryptor are just simply the maps e_K and d_K of the system. We call this kind of scheme a **symmetric key encryption protocol**, and we call the key K in this setting the **symmetric key**, or the **secret key**.

Lemma 1.1.1. *In any cryptosystem with any key K, the map e_K is $1-1$.*

Proof. Let $e = e_K$, and let $x, y \in \mathcal{P}$ be such that $e(x) = e(y) = z$. Then let $d = d_K$. Then $d(z) = d(e(x)) = x$ and $d(z) = d(e(y)) = y$, implying that $x = y$. ∎

Lemma 1.1.2. *In any cryptosystem, if $\mathcal{P} = \mathcal{C}$, then e_K is a permutation on $\mathcal{P}$.*

Proof. We have that e_K is $1-1$. We have that $e_K(\mathcal{P}) \subseteq \mathcal{P}$. Now, let $x \in \mathcal{P}$, and consider $d_K(x) = y$. By definition, we get that $x = e_K(y)$ for some $y \in \mathcal{P}$. This makes $e_K(\mathcal{P}) = \mathcal{P}$, and hence onto. Therefore, e_K is a permutation. ∎

We now finish the section by discribing the shift cipher.

Definition. Let $\mathcal{P} = \mathcal{C} = \mathcal{K} = \mathbb{Z}/n\mathbb{Z}$. We define the **shift cipher** to be the pair (e_K, d_K), defined by the rules $e_K : x \to (x + K) \mod 26$, and $d_K : y \to y - K \mod n$.

Remark. When $K = 3$, we call the shift cipher the **Caesar cipher**.

Example 1.1. Let $Q = \{A, B, C, \ldots, Z\}$ be the entire english alphabet. We implement the shift cipher on Q by first taking the map $Q \to \mathbb{Z}/26\mathbb{Z}$ defined by $A \to 0, B \to 1, C \to 2 \ldots Z \to 25$; see figure 1.2 Then if x is any English plain text message (without spaces), we compute the cipher text on the image of x, and then take the inverse map to get our cipher y in English plaintext. To decrypt, it is the same process, except computing d_K.

Example 1.2. Using the shift cipher for $n = 26$, choose the key $K = 11$, and the plaintext message `wewillmeetatmidnight`. Using the map $Q \to \mathbb{Z}/26\mathbb{Z}$, we get the following:

22	4	8	22	11	11	12	4	4	19
0	19	12	8	3	13	8	6	7	19

Using the shift cipher, $\mod 11$, we get the following:

7	15	7	19	22	22	23	15	15	4
11	4	23	19	14	24	19	17	18	4

So taking $\mathbb{Z}/26\mathbb{Z} \to Q$, we get our cipher text to be: `HPHTWWXPPELEXTOYTRSE`. To decrypt, we simply just reverse the process.

Normally, it is necessary to provide an intermediary map, such as the map $Q \to \mathbb{Z}/_{26\mathbb{Z}}$ so that our plaintext can be computed to sipher text. We call these maps **encodings**.

Theorem 1.1.3. *Let $\mathcal{P} = \mathcal{C} = \mathcal{K} = \mathbb{Z}/_{n\mathbb{Z}}$, for any $K \in \mathcal{K}$, the shift cipher defines a cryptosystem.*

Proof. let (e_K, d_K) be the encryption and decryption rules defined by the shift cipher for any $K \in \mathcal{K}$. That is $e_K(x) = x + K \mod n$ and $d_K(y) = y - K \mod n$, for any $x, y \in \mathcal{P}$. Thus, letting $y = e_K(x)$, we get $d_K(y) = y - K \mod n = (x + K) - K \mod 26 = x + (K - K) \mod n = x \mod n$. Therefore, the pair (e_K, d_K) define a cryptosystem. ■

Definition. We define a cyptosystem to be of **practical use** if:

(1) The ecnryption and decryption rules e_K and d_K are computationally feasable; i.e. a anyone should be able to efficiently compute them.

(2) An adversary obtaining a cipher text y cannot determine the plaintext x, nor the key K in any computationally feasable ammount of time.

Example 1.3. The shift cipher is not of practical use. Notice that if we have any cipher text, we can simply try the following sequence $\{d_i\}_{i=0}^{25}$ of decryption rules until we successfully decrypt the cipher. For example, if we have the cipher text `JBCRCLQRWCRVNBJENBWRWN` we dectypt it with the above sequence to obtain:

```
jbcrclqrwcrvnbjenbwrwn
iabqbkpqvbqumaidmavqvm
hzapajopuaptlzhclzupul
gyzozinotzoskygbkytotk
fxynyhmnsynrjxfajxsnsj
ewxmxglmrxmqiweziwrmri
dvwlwfklqwlphvdyhvqlqh
cuvkvejkpvkogucxgupkpg
btujudijoujnftbwftojof
astitchintimesavesnine
```

to obtain the plain text `astitchintimesavesnine`. We notice that this was done in precisely 9 computations. On average, we can compute the plaintext in $\frac{26}{2} = 13$ computations.

Definition. Let $\mathcal{P} = \mathcal{C} = \mathbb{Z}/_{n\mathbb{Z}}$ and let $\mathcal{K} = S_n$ the permutation group on 26 elements. For $\pi \in \mathcal{K}$, we define the **substitution cipher** on π to be the pair (e_π, d_π) such that $e_\pi : x \to \pi(x)$ and $d_\pi : y \to \pi^{-1}(y)$.

Theorem 1.1.4. *For $\mathcal{P} = \mathcal{C} = \mathbb{Z}/_{n\mathbb{Z}}$ and any $\pi \in S_{26}$, the substitution cipher defines a cryptosystem.*

Proof. For any $x, y \in \mathbb{Z}/_{n\mathbb{Z}}$, and $\pi \in S_n$, let $y = e_\pi(x)$. Then $d_\pi(y) = \pi^{-1}(e(x)) = \pi^{-1}\pi(x) = x$. ■

Example 1.4. Let π be the permutation defining the encryption rule $e = \pi$ by:

```
a b c d e f g h i j k l m n o p q r s t u v w x y z
X N Y A H P O G Z Q W B T S F L R C V M U E K J D I
```

and the decryption rule $d = \pi^{-1}$ by:

```
A B C D E F G H I J K L M N O P Q R S T U V W X Y Z
d l r y v o h e z x w p t b g f j q n m u s k a c i
```

Where, for e, the first row represents the elements $x \in \mathbb{Z}/26\mathbb{Z}$, and the the second row represents the elements $e(x) \in \mathbb{Z}/26\mathbb{Z}$. Likewise, for d, the first row represents all y and the second all $d(y)$.

Now take the cipher text:

```
MGZVYZLGHCMHJMYXSSFMNHAHYCDLMHA
```

we get the plain text:

```
thisciphertextcannotbedecrypted
```

Lemma 1.1.5. *Let $|\mathcal{P}| = \mathcal{C}| = n$. There are $n!$ possible keys for the substitution cipher.*

Proof. Notice a key in the substitution cipher is any permutation $\pi : \mathcal{P} \to \mathcal{C}$, hence there are $n!$ of them. ∎

Remark. With the number of possible keys for the substitution cipher being $n!$, for n sufficiently large, the permutations become dificult to count. For example, for $n = 26$, there are already 26! possible keys, which makes it infeasible to guess by brute force. This provides more secruity than the shift cipher, however there are other methods of breaking the substitution cipher.

Example 1.5. Shift ciphers are substitution ciphers.

Now, before defining the next cipher, lets state and prove a theorem from number theory.

Theorem 1.1.6. *Let $a, b \in \mathbb{Z}/n\mathbb{Z}$. The congruence $ax \equiv b \mod n$ has unique solution for $x \in \mathbb{Z}/n\mathbb{Z}$ if, and only if $(a, n) = 1$.*

Proof. First suppose that $(a, n) = d > 1$. Then the congruence $ax \equiv 0 \mod n$ has two solutions $x = 0$ and $x = \frac{n}{d}$. This proves the first direction.

Now suppose $(a, n) = 1$. Then there exist $u, v \in \mathbb{Z}/n\mathbb{Z}$ for which $au + vn \equiv au \equiv 1 \mod n$, thus $x = (au)x \equiv bu \mod n$. So $x = bu$ is a solution, that x is unique follows from the fact that u and v are uniquely determined. ∎

Corollary. *The congruence $ax + b \equiv y \mod n$ has unique solution if, and only if $(a, n) = 1$.*

Remark. Here, the pair (a, n) is taken to mean the greated common divisor of a and n. Also note that if we consider this theorem group theoretically, we have $a \in U(\mathbb{Z}/_{n\mathbb{Z}})$, and the theorem reduces to the cancellation law for this group.

Definition. Let $\mathcal{P} = \mathcal{C} = \mathbb{Z}/_{n\mathbb{Z}}$ and let $K = U(\mathbb{Z}/_{n\mathbb{Z}})$. We define the **affine cipher** to be a substitution cipher defined by the pair (e, d) where $e : x \to ax + b \mod n$ and $d : y \to a^{-1}(y - b) \mod n$.

Example 1.6. (1) For the pair $(1, b)$, the affine cipher is equivalent to the shift cipher whos key is b.

(2) For $n = 26$ and the pair $K = (7, 3)$ take $e(x) = 7x + 3 \mod 26$ and $d(y) = 15(y - 3) \mod n \equiv 15y - 19 \mod 26$. Then if $y = e(x)$, $d(y) = 15(7x+3) - 19 \equiv x + (19 - 19) \equiv x \mod n$.

Lemma 1.1.7. *The affine cipher defines a cryptosystem.*

All these cryptosystem presented so far have one thing in common.

Definition. We call a cryptosystem is said to be **monoalphabetic** if for any given key K, the encryption and decryption rules e_K and e_K map each element of $\mathcal{P}$ to a unique element of $\mathcal{C}$ and viceversa.

Example 1.7. (1) The shift cipher is monoalphabetic.

(2) They cryptosystem defined by the substitution cipher is monoalphabetic; indeed since the key π is a permutation, then it is $1 - 1$ and onto, since $e = \pi$, $d = \pi^{-1}$, this establishes the result.

(3) Affine ciphers are monoalphabetic, since they are substitution ciphers.

Definition. Let $m \in \mathbb{Z}^+$, and let $\mathcal{P} = \mathcal{C} = \mathcal{K} = (\mathbb{Z}/_{n\mathbb{Z}})^m$. Then for any $K = (k_1, \ldots, k_m) \in (\mathbb{Z}/_{n\mathbb{Z}})^m$, define the pair (e_K, d_K) by the maps $e_K : x \to x + K \mod m$ and $d_k K : y \to y - K \mod n$, for $x = (x_1, \ldots, x_m), y = (y_1, \ldots, y_m) \in (\mathbb{Z}/_{n\mathbb{Z}})^m$. We call this cipher **Vigenère's cipher**.

Theorem 1.1.8. *Vigenère's cipher defines a cryptosystem.*

Proof. Let $x = (x_1, \ldots, x_m)$ and $y = (y_1, \ldots, y_m)$. Notice that $y = x + K \mod n$ if, and only if $y_i = x_i + k_i \mod n$. Thus each k_i defines the key for a shift cipher, so we can see that when $y = e(x)$, then $d(y) = x$. ■

Example 1.8. Vigenère's cipher is not monoalphabetic. Notice that the map $x + K = (x_i + k_i, \ldots, x_m + k_m)$. Since each $x_i \in \mathbb{Z}/_{n\mathbb{Z}}$, and each k_i is the key for a shift cipher, the map $e(x) = x + K \mod n$ does not map elements of $\mathcal{P}$ uniquely.

Example 1.9. Let $m = 6$, and choose the key K to be the word `CIPHER`, so that with the map $Q \to \mathbb{Z}/_{26\mathbb{Z}}$, $K = (2, 8, 15, 7, 4, 17)$. If we have the plain text:

```
thiscr yptosy stemis notsec ure
CIPHER CIPHER CIPHER CIPHER CIP
------ ------ ------ ------ ----
VPXZGI AXIVWP UBTTMJ PWIZIT WZT

VPXZGI AXIVWP UBTTMJ PWIZIT WZT
YSLTWP YSLTWP YSLTWP YSLTWP YSL
------ ------ ------ ------ ----
thiscr yptosy stemis notsec ure
```

Figure 1.3: Encryption and decryption with Vigenère's cipher. See example (1.9).

`thiscryptosystemisnotsecure`

then we divide the plaintext into blocks of 6 to get:

`thiscr yptosy stemis notsec ure`

treating each block as a 6-tuple, i.e. `thiscr`$= (19, 7, 8, 18, 2, 17)$ (and `ure`$= (20, 17, 4, 0, 0, 0)$ where we discard the remaining 3 components), we can apply Vigenère's cipher on each individual block to reviece the cipher blocks:

`VPXZGI AXIVWP UBTTMJ PWIZIT WZT`

We then concatenate each block to obtain the ciphertext:

`VPXZGIAXIVWPUBTTMJPWIZITWZT`

We can visualize this process in figure 1.3. Decryption is the same process.

The cipher of Vigenère has the property that given a keyword of length m, an alphabetic character can be mapped onto one of m possible characters.

Definition. Let $m \in \mathbb{Z}^+$. We call a cryptosystem **polyalphabetic** if given a keyword of length m, of m distinct characters, then a given element of $\mathcal{P}$ can be mapped to any one of m possible elements of $\mathcal{C}$.

We now describe a cryptosystem that takes as plaintext a string of characters and outputs a permutation on those characters.

Definition. Let $\mathcal{P} = \mathcal{C} = (\mathbb{Z}/n\mathbb{Z})^m$ and let $\mathcal{K} = S_m$ We define the **transpostion cipher** (or **permutation cipher**) to be a pair (e_π, d_π), for $\pi \in S_m$ such that for any $x, y \in (\mathbb{Z}/n\mathbb{Z})^m$, $e_\pi(x) = (x_{\pi(1)}, \dots x_{\pi(m)})$ and $d(y) = (y_{\pi^{-1}(1)}, \dots, y_{\pi^{-1}(m)})$.

Theorem 1.1.9. *The transposition cipher defines a cryptosystem.*

Proof. Let $\pi \in S_m$ and $y = e_\pi(x) = (x_{\pi(1)}, \ldots x_{\pi(m)})$. Then $d(y) = (x_{\pi^{-1}(\pi(1))}, \ldots x_{\pi^{-1}(\pi(m))}) = (x_1, \ldots, x_m) = x$. ■

Example 1.10. Let $m = 6$, $n = 26$ and define $\pi \in S_m$ by the permutation $\pi = (1\ 3)(2\ 5\ 4\ 6)$. Then $\pi^{-1} = (1\ 3)(6\ 4\ 5\ 2)$. Given the plaintext:

`shesellseashellsbytheseashore`

Encryption is proceeds similarly as in Vigenère's cipher. We partition the message into blocks of 6:

`shesel lsseas hellsb ythese ashore`

Then apply the encryption rule $e_\pi(x) = (x_{\pi(1)}, \ldots x_{\pi(m)})$ to get the cipher blocks:

`EESLSH SALSES LSHBLE HSYEET HRAEOS`

and we then concatenate the blocks to obtain the ciphertext:

`EESLSHSALSESLSHBLEHSYEETHRAEOS`

Decryption is the same, except we use d_π instead of e_π on the ciphertext.

So far, the ciphers presented all (with exception of Vigenère's cipher) use one key to ecnrypt the whole message. These kind of ciphers encrypt the message in "blocks". However, this isn't the only way to excrypt messages.

Definition. We call the cipher (e, d) of a given cryptosystem a **block cipher** if successive plaintext elements $x \in \mathcal{P}$ are encrypted using the same key $K \in \mathcal{K}$; that is, for any cipher text string $y = y_1 \ldots y_m$ of length m, $y = e(x_1) \ldots e(x_m)$ where $x = x_1 \ldots x_m$ is the associated plaintext string.

Remark. When we say string, we simply mean an m-tuple of elements. We can then alternatively write $x = x_1 x_2 \ldots x_m$ to denote $x = (x_1, x_2, \ldots, x_m)$. We will often write tuples this way when we want to emphasize them as being strings, or some other symbol stream.

Example 1.11. The shift, substitution, and transposition ciphers are all block ciphers.

We now define a type of cipher that is not a block cipher.

Definition. Let $\mathcal{P}$, and $\mathcal{C}$ be finite plaintext and ciphertext spaces, respectively, and let $\mathcal{K}$ be a finite key space. let $\mathcal{L}$ be a finite set called the **key stream alphabet**. We define the **synchronous stream cipher** as follows: define the map $g : \mathcal{K} \to \mathcal{L}^m$, where $m \in \mathbb{Z}^+$, called the **key stream generator**, such that $g : K \to z = z_1 z_2 \ldots z_m$. Then define the pair (e_z, d_z) for each $z \in \mathcal{L}$ such that $e_z : \mathcal{P} \to \mathcal{C}$ and $d_z : \mathcal{C} \to \mathcal{P}$ with $d_z(e_z(x)) = x$. We call elements of $\mathcal{L}^m$ **keystreams**.

Remark. When $m \in \mathbb{Z}^+$, we call the stream cipher **finite**, and the keystream z a **finite keystream**. Often however, we want infite keystreams. To do this, we just take $m > M$ for some arbitrarily large $M \in \mathbb{Z}^+$; we then call z an **infinite keystream** and we call the stream cipher infinite.

Example 1.12. (1) Vigenère's cipher is a finite synchronous stream cipher. Let $m \in \mathbb{Z}^+$ be the length of a given keyword $K = (k_1, \ldots, k_m)$. Let $\mathcal{P} = \mathcal{C} = \mathcal{L} = \mathbb{Z}/26\mathbb{Z}$ and $\mathcal{K} = (\mathbb{Z}/26\mathbb{Z})^m$. Define then the pair (e_z, d_z) such that $e_z : x \to x + z \mod 26$ and $d_z : y \to y - z \mod 26$, where $z = z_1 \ldots z_m$ and each $z_i = \begin{cases} k_i, & 1 \leq i \leq m \\ z_i - m, & i \geq m+1 \end{cases}$.

(2) Block ciphers are finite stream ciphers where the keystream consist of characters $z_i = K$ for all $1 \leq i \leq m$; i.e. the keystream is constant.

Definition. We call a stream cipher **periodic** with period d if for some keystream z, $z_{i+d} = z_i$ for all $i \geq 1$. We write $\operatorname{ord} z = d$.

Example 1.13. Defining Vigenère's cipher as an infinite stream cipher, it is periodic with period $\operatorname{ord} z = m$.

Often with stream ciphers, we have that $\mathcal{P} = \mathcal{C} = \mathcal{L} = \mathbb{Z}/2\mathbb{Z}$. This motivates us to define a method for creating keystreams.

Definition. Let $m \in \mathbb{Z}^+$, and let $z = (z_1, \ldots, z_m) \in (\mathbb{Z}/n\mathbb{Z})$. We call z a **linear recurrence** of degree $\deg z = m$ if

$$z = \sum_{j=0}^{m-1} c_j z_{i+j} \mod n \tag{1.1}$$

for all $i \geq 1$, and where $c_j \in \mathbb{Z}/n\mathbb{Z}$ for $1 \leq j \leq m-1$ are called the **constants** of the linear recurrence.

Remark. Often we use linear recurrences when working with $n = 2$.

One appealing aspect of using linear recurrence with binary messages (i.e. in $\mathbb{Z}/2\mathbb{Z}$) is that they can be easily implemented in harware using linear feedback shift registers; which we define abstractly as:

Definition. We define a **linear feedback shift register** of m **steps** to be a set of rules: for some keystream $k = (k_1, \ldots k_m) \in (\mathbb{Z}/2\mathbb{Z})^m$,

(1) k_1 is the next keystream bit.

(2) $k_2, \ldots k_m$ is shifted one stage left.

(3) $k_m = \sum_{j=0}^{m-1} c_j k_{j+1}$

Definition. We define an **asynchronous stream cipher** to be a stream cipher whos keystream elements z_i depend on the plaintext elements as well as the key K.

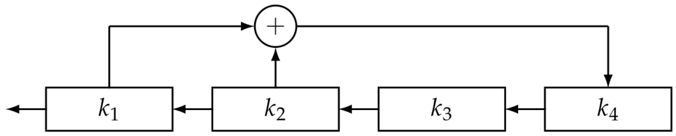

Figure 1.4: A linear feedback shift register.

Definition. Let $\mathcal{P} = \mathcal{C} = \mathcal{K} = \mathbb{Z}/n\mathbb{Z}$. Let $z_1 = K \in \mathcal{K}$ and define $z_i = x_{i-1}$ for all $i \geq 2$. Then for $0 \leq z \leq n$, define the **auto cipher** to be the pair (e_z, d_z) such that $e_z : x \to x + z \mod n$ and $d_z : y \to y - z \mod n$.

Theorem 1.1.10. *The auto cipher defines a cryptosystem.*

Corollary. *The auto cipher is an asynchronous stream cipher.*

Proof. Notice that given out key K, $z_1 = K$, and $z_i = x_{i-1}$ for all $i \geq 2$. Thus taking $z = z_1 z_2 \dots$ to be our keystream, we see that z depends on K and the elements x_i. ∎

We end the section with an example on the auto cipher.

Example 1.14. Let $K = 8 \mod 26$ with plaintext `rendezvous`. We first encode the plaintext with the map $q \to \mathbb{Z}/26\mathbb{Z}$, $q_i \to i$ to get:

17 4 13 3 4 25 21 14 20 18

Then our keystream is

8 17 4 13 3 4 25 21 14 20

Applying e_z, we get

25 21 17 16 7 3 20 9 8 12

We then decode with the rule $\mathbb{Z}/26\mathbb{Z} \to Q$ to get the ciphertext:

`ZVRQHDUJIM`

To decrypt, we take the encoded ciphertext and compute $x_1 = d_8(25) \equiv 17 \mod 26, x_2 = d_{17}(21) \equiv 4 \mod 26, \dots, x_{10} = d_{20}(12) \equiv 18 \mod 26$ which gives us the encoding: 17 4 13 3 4 25 21 14 20 18 which gives us our original plaintext:

`rendezvous`

1.2 Classical Cryptanalysis.

Chapter 2

Perfect Secrecy.

2.1 Perfect Secrecy and The One-Time Pad.

We assume a cryptosystem $(\mathcal{P}, \mathcal{C}, \mathcal{K})$ with cipher (e, d) is used, where a key K is used for only one encryption. Let $\mathcal{P}$ have a probability distribution represented by the random variable X, and assume the key $k \in \mathcal{K}$ is chosen according to a probaility distribution represented by K. We define the set of all possible ciphertexts encrypted with k to be:

$$C(k) = \{e_k(x) : x \in \mathcal{P}\} \tag{2.1}$$

Then we can also define a probaility distribution on $\mathcal{C}$ represented by the random variable Y, such that for every $y \in \mathcal{C}$, $P(Y = y) = \sum_{y \in C(K)} P(X = x)P(K = k)$ and $P(Y = y|X = x) = \sum_{y \in C(K)} P(K = k)$. Then by Baye's theorem, we have:

$$P(X = x|Y = x) = \frac{P(X = x)P(Y = y|X = y)}{p(Y = y)} = \frac{P(X = x)\sum P(K = k)}{\sum P(X = x)P(K = k)} \tag{2.2}$$

Example 2.1. Let $\mathcal{P} = \{a, b\}$ where $P(a) = \frac{1}{4}$ and $P(b) = 1 - P(a) = \frac{3}{4}$. Let $\mathcal{K} = \{K_1, K_2, K_3, K_4\}$ with $P(K_1) = \frac{1}{2}$, $P(K_2) = P(K_3) = \frac{1}{4}$, and let $\mathcal{C} = \{1, 2, 3, 4\}$. Definee the encryption rules $e_{K_1} : a \to 1, b \to 2$, $e_{K_2} : a \to 2, b \to 3$, and $e_{K_3} : a \to 3, b \to 4$ we get the following matrix whose $(P(K_i, X))$

$$\begin{pmatrix} 1 & 2 \\ 2 & 3 \\ 3 & 4 \end{pmatrix}$$

We find the probability distribution on $\mathcal{C}$ to be $P(1) = \frac{1}{8}$, $P(2) = \frac{3}{8} + \frac{1}{16}$, $P(3) = \frac{1}{4}$, and $P(4) = \frac{1}{16}$. We find the conditional probaility distrobution to be:

$$\begin{aligned} P(a|1) &= 1 & P(b|1) &= 0 \\ P(a|2) &= \frac{1}{7} & P(b|1) &= \frac{6}{7} \\ P(a|3) &= \frac{1}{7} & P(b|1) &= \frac{3}{4} \\ P(a|4) &= 0 & P(b|1) &= 1 \end{aligned}$$

Definition. We say a cryptosystem ha **perfect secrecy** if $P(x|y) = P(x)$ for all possible plaintext elements x and all possible plaintext elements y.

Example 2.2. In the above example, the cryptosystem $(\mathcal{P}, \mathcal{C}, \mathcal{K})$ has perfect secrecy only when $y = 3$; $P(a|3) = P(a) = \frac{1}{4}$.

Theorem 2.1.1. *Let $\mathcal{P} = \mathcal{C} = \mathcal{K} = \mathbb{Z}/_{n\mathbb{Z}}$ and suppose $P(K) = \frac{1}{n}$ fo all $K \in \mathbb{Z}/_{n\mathbb{Z}}$. Then for any plaintex distribution, the shift cipher has perfect secrecy.*

Proof. Given the encryption rule $e : x \to x + K \mod n$, computing the probaility distribution on $\mathcal{C} = \mathbb{Z}/_{n\mathbb{Z}}$, we have $P(Y = y) = \sum_K P(K)P(d_K(y)) = \frac{1}{n} \sum P(x = y - K)$. Now if x and y are given plaintext and ciphertext elements, then $d_k(y) = x - K \mod n$ is a permutation on $\mathbb{Z}/_{n\mathbb{Z}}$, hebce we get $\sum P(x = y - K) = \sum P(X = x) = 1$; consequently, $p(y) = \frac{1}{n}$. Now $p(y|x) = P(K = y - x \mod n) = \frac{1}{n}$. Therefore, by Baye's theorem, we get $P(x|y) = P(x)$. ∎

Remark. This theorem says that the shift cipher is unbreakable provided a new random key is used to encrypt each plaintext element. This is computationally inefficient.

Theorem 2.1.2. *Let $(\mathcal{P}, \mathcal{C}, \mathcal{K})$ be a cryptosystem where $|\mathcal{K}| = |\mathcal{C}| = |\mathcal{P}|$. Then $(\mathcal{P}, \mathcal{C}, \mathcal{K})$ has perfect secrecy if, and only if every secret key is used with equal probailitiy $\frac{1}{|\mathcal{K}|}$ and for all $x \in \mathcal{P}$, $y \in \mathcal{C}$, there is a unique $K \in \mathcal{K}$ for which $e_K(x) = y$.*

Proof. Suppose $(\mathcal{P}, \mathcal{C}, \mathcal{K})$ has perfect secrecy. There is atleast one key K with $e_K(x) = y$, so we get $|\mathcal{C}| \leq |C(x)| \leq |\mathcal{K}|$, where $C(x) = \{e_K(x) : K \in \mathcal{K}\}$. Then by assumption $|\mathcal{C}| = |C(x)| = |\mathcal{K}|$ which implies that $e_{K_1}(x) = e_{K_2}(x) = y$ only when $K_1 = K_2$. Now, again by assumption, since $P(x|y) = P(x)$, it follows that $P(K_i) = P(y)$ which implies the keys are used with equal probability.

COnversely, let $n = |\mathcal{K}|$ and $\mathcal{P} = \{x_1, \ldots, x_n\}$. Let $y \in \mathcal{C}$ be a ciphertext element, and suppose that $e_{K_i}(x_i) = y$ for unique K_i, $1 \leq i \leq n$. Then by Baye's theorem, $P(x_i|y) = \frac{P(K=K_i)P(x_i)}{P(y)}$. Now, since every K_i is chosen with probability $\frac{1}{n}$, we see $P(K = K_i) = P(y) = \frac{1}{n}$. Thuse $P(x_i|y) = P(x_i)$. ∎

We now define the one-time pad.

Definition. Let $n \in \mathbb{Z}^+$ and let $\mathcal{P} = \mathcal{C} = \mathcal{K} = (\mathbb{Z}/_{2\mathbb{Z}})^n$. For $K \in (\mathbb{Z}/_{2\mathbb{Z}})^n$, define the pair (e, d) by the rules $e : x \to x + K \mod 2 = (x_1 + K_1, \cdots x_2 + K_2) \mod 2$ and $d : y \to y + K \mod 2 = (y_1 + K_1, \ldots, y_n + K_n) \mod 2$. We call the cipher (e, d) the **one-time pad**.

Theorem 2.1.3. *The one-time pad defines a perfectly secure cryptosystem.*

2.2 Entropy

Definition. Let X be a discrete random variable. We define the **entropy** of X to be:

$$H(X) = -\sum_x P(x) \log P(x) \tag{2.3}$$

wher log is the logarithm base 2. When $P(x) = 0$, we define $P(x) \log P(x) = 0$.

Example 2.3. (1) Let X be a random variable with sample size n. If $P(X = x) = \frac{1}{n}$, then $H(X) = \log n$.

(2) Let $(\mathcal{P}, \mathcal{C}, \mathcal{K})$ be the cryptosystem defined in example (2.1), then $H(\mathcal{P}) = -\frac{1}{4}\log\frac{1}{4} - \frac{3}{4}\log\frac{3}{4} = 2 - \frac{3}{4}\log 3$. So $\mathcal{P}$ gives about 0.81 bits of uncertainty. Similarly, $H(\mathcal{C}) = 1.85$ and $H(\mathcal{K}) = 1.5$.

Definition. We define a real-valued function f to be **concave** on an interval I if

$$f(\frac{x+y}{2}) \geq \frac{f(x)+f(y)}{2} \tag{2.4}$$

We say f is **strictly concave** on I if

$$f(\frac{x+y}{2}) > \frac{f(x)+f(y)}{2} \tag{2.5}$$

for all $x, y \in I$.

Theorem 2.2.1 (Jensen's Inequality)**.** *Let f be a continuous real-valued function on an interval I, and suppose $\sum a_i = 1$ for some sequence $\{a_i\}_{i=1}^n$ where $a_i > 0$ for all i. Then:*

$$\sum f(a_i x_i) \leq f(\sum a_i x_i) \tag{2.6}$$

given a sequence $\{x_i\}_{i=1}^n \subseteq I$.

Corollary. *Equality holds when $x_1 = \cdots = x_n$.*

Theorem 2.2.2. *For any random variable X with probaility distribution $\{p_i\}_{i=1}^n$, we have $0 \leq H(X) \leq \log n$.*

Proof. First notice that since $\{p_i\}$ is a probability distribution, $p_i > 0$ for all i and $\sum p_i = 1$. Now, that $0 \leq H(X)$ follows from definition. Then, by Jensen's inequality, $H(X) = -\sum p_i \log p_i = \sum p_i \log p_i^{-1} \leq \log \sum p_i p_i^{-1} = \log n$. ∎

Corollary. *$H(X) = 0$ when atleast one $p_i = 0$, and $H(X) = \log n$ if $p_i = \frac{1}{n}$ for all i.*

Theorem 2.2.3. *$H(X, Y) \leq H(X) + H(Y)$, with equality if, and only if X and Y are independent random variables.*

Definition. Let X and Y be random variables. We define the **conditional entropy** of X given $Y = y$, and X given Y to be:

$$H(X|y) = -\sum_x P(x|y)\log P(x|y) \tag{2.7}$$

and

$$H(X|Y) = \sum_y P(y)H(X|y) = -\sum_y \sum_x P(y)P(x|y)\log P(x|y) \tag{2.8}$$

Theorem 2.2.4. $H(X, Y) = H(Y) + H(X|Y)$

Corollary. *$H(X|Y) \leq H(X)$ with equality if, and only if X and Y are independent.*

2.3 Spurious Keys.

The goal of cryptanalysis is to recover the key from a sufficiently large enough body of ciphertext. Supposing that an adversary launches a cyrptanalytic attack, we make the following definition.

Definition. In a cryptanalytic attex, we call incorrectly determined, but possible keys **spurious keys**.

Example 2.4. Suppose an adversary obtaines the ciphertext `WNAJW` and determines a shift cipher has been used. Then there are two possible keys, $F = 5$ giving the plaintext `river` and $W = 22$ giving the plaintext `arena`. However, only one of these keys is the correct key, and the other is spurious.

Definition. Let $(\mathcal{P}, \mathcal{C}, \mathcal{K})$ be a cryptosystem. We define $H(K, C)$ to be the **key equivocation** wich measures the uncertainty of the key K given a ciphertext C.

Theorem 2.3.1. *Let $(\mathcal{P}, \mathcal{C}, \mathcal{K})$ be a cryptosystem with cipher (e, d), then $H(K|C) = H(K) + H(P) - H(C)$.*

Proof. Notice that $H(K, P, C) = H(C|K, P) + H(K, P)$. Now K and P uniquely determine C, given $y = e(x)$; so $H(C|K, P) = 0$. So $H(K, P, C) = H(K, P) = H(K) + H(P)$ (since K and P are independent).

Similarly, C and K uniquely determine P, since $x = d(y)$, so $H(P|K, C) = 0$, and $H(K, P, C) = H(K, C) = H(K) + H(K|C)$. Now, rearranging terms and substituting, we get $H(K|C) = H(K) + H(P) - H(C)$. ■

Example 2.5. Again, considering example (2.1), $H(K|C)$ is about $1.5 + 0.81 - 1.85 = 0.46$ bits of uncertainty. Computing with conditional entropy, we compute the probability matrix $(P(K = K_i|y = j))$ for $1 \leq i \leq 3$ and $1 \leq j \leq 4$ to obtain:

$$\begin{pmatrix} 1 & 0 & 0 \\ \frac{6}{7} & \frac{1}{7} & 0 \\ 0 & \frac{3}{4} & \frac{1}{4} \\ 0 & 0 & 1 \end{pmatrix}$$

then $H(K) = 0.46$

Definition. Let L be a natural language and P^n the random variable with probability distribution all n-grams of plaintext. We define the **entropy** of L to be:

$$H_L = \lim_{n\to\infty} \frac{H(P^n)}{n} \tag{2.9}$$

and the **redundancy** of L to be:

$$R_L = 1 - \frac{H_L}{\log |\mathcal{P}|} \tag{2.10}$$

Where $\mathcal{P}$ is the plaintex space.

Theorem 2.3.2. *Let $(\mathcal{P}, \mathcal{C}, \mathcal{K})$ be a cryptosystem with $|\mathcal{C}| = |\mathcal{P}|$ and keys chose equiprobably. Let L be the underlying natural language, then given a string of ciphertext of length n, with n sufficiently large, the expected number of spurious keys s:*

$$\overline{s}_n \geq \frac{|\mathcal{K}|}{|\mathcal{P}|^{nR_L}} \tag{2.11}$$

Remark. $\overline{s}_n \to 0$ exponentially quickly as $n \to \infty$.

Definition. The **unicity distance**, n_0, of a cryptosystem is define to be the value of n for which $\overline{s}_n \to 0$. I.e. it is the average ammount of ciphertext an adversary needs to uniquely determine the correct key, given enough time and resources.

Lemma 2.3.3. *As $n \to \infty$,*

$$n_0 = \frac{\log |\mathcal{K}|}{R_L \log |\mathcal{P}|} \tag{2.12}$$

Chapter 3

Discrete Memoryless Channels and Cost.

3.1 The Capacity and Cost of a DMC.

We now define in a much more precise way what we mean by a discrete memoryless channel.

Definition. We define **discrete memoryless channel** (DMC) to be a triple (A_X, A_Y, Q) where A_X and A_Y are finite sets of size $|A_X| = r$ and $|A_Y| = s$ called the **input alphabet** and **output alphabet** resepctively, and Q is an $r \times s$ stochastic mmatrix called the **transitional probability matrix** whose entries are $Q = (p(y|x))$. Additionally, define $b : A_X \to \mathbb{R}$ by the rule $x \to b(x)$. We call $b(x)$ the **cost** of x. We call b a **cost function**.

Remark. That is, a DMC is completely described by its input space, output space, and the matrix Q of all transitional probailities. Additionally, associating to each x the cost $b(x)$ serves to characterize what it "costs" to send the input x over the channel.

Example 3.1. (1) Let $A_X = A_Y = \mathbb{F}_2$ with transitional matrix $Q = \begin{pmatrix} 1-p & p \\ p & 1-p \end{pmatrix}$, where $0 \leq p \leq \frac{1}{2}$. Let $b(0) = 0$ and $b(1) = 1$. We call this DMC the **binary symmetric channel**.

(2) LEt $A_X = \{0, \frac{1}{2}, 1\}$, and $A_Y = \mathbb{F}_2$. Let $Q = \begin{pmatrix} 1 & 0 \\ \frac{1}{2} & \frac{1}{2} \\ 0 & 1 \end{pmatrix}$ with $b(0) = b(1) = 1$ and $b(\frac{1}{2}) = 0$. This defines a DMC.

(3) Let $A_X = A_Y = \mathbb{F}_3$, with $Q = I_{3\times 3}$ and $b(0) = b(1) = 1$, $b(2) = 4$.

Definition. Suppose $D = (A_X, A_Y, Q)$ is a DMC with cost function $b(x)$. Assume D is used n consecutive times for $n \in \mathbb{Z}^+$, with input and ouptut sequences $x = \{x_i\}_{i=1}^n$ and $y = \{y_i\}_{i=1}^n$. The **memoryless assumption** that y_i is a function only of x_i for each i is the product $\prod_{i=1}^n p(x_i|x_i)$. We then define the **cost** of x to be;

$$b(x) = \sum_{i=1}^{n} b(x_i) \tag{3.1}$$

We define the **average cost** of x to be:

$$\bar{b}(x) = E(b(x_i)) = \sum_{i=1}^{n} p(x_i)b(x_i) \tag{3.2}$$

where p is the probaility distribution on x treated as a random variable.

Definition. For a $D = (A_X, Q)$ with cost function $b(x)$ and $n \in \mathbb{Z}^+$; we define the ***n*-th capacity-cost function** of D to be

$$C_n(\beta) = \max_{(X,Y)\in A_X^n \times A_Y^n} \{I(X,Y) : \bar{b}(x) \leq n\beta\} \tag{3.3}$$

where $P(Y|X) = \prod p(y_i x_i)$.. We call $x = \{x_i\}$ the **test source** and we say it is **β-admissable** if $\bar{b}(X) \leq n\beta$.

Lemma 3.1.1. *For $n \in \mathbb{Z}^+$ and any test source $x = \{x_i\}$, if $\beta_{\min} = \min_{x_i \in A_X} b(x_i)$, then $n\beta_{\min} \leq \bar{b}x$.*

Proof. Notice $\beta_{\min} \leq \frac{\sum p(x_i)x_i}{n} = \frac{\bar{b}(x)}{n}$. ∎

Corollary. *$C_n(\beta)$ is defined for all $\beta \geq \beta_{\min}$.*

Corollary. *If $\beta_{\min} \leq \beta_1 < \beta_2$, then $C_n(\beta_1) \leq C_n(\beta_2)$. That is, $C_n(\beta)$ is an increasing function of β.*

Proof. Let $C_i = \{I(X,Y) : \bar{b}(x) \leq n\beta_i\}$ for $1 \leq i \leq 2$. Then if X is a test source achieving $C_n(\beta_1)$, then $\bar{b}(x) \leq n\beta_1 \leq n\beta_2$. This makes $C_1 \subseteq C_2$, thus $C_n(\beta_1) \leq C_n(\beta_2)$. ∎

Definition. LEt $D = (A_X, A_Y, Q)$ be a DMC. We define the **capacity cost function** of D to be:

$$C(\beta) = \sup \frac{C_n(\beta)}{n} \tag{3.4}$$

Theorem 3.1.2. *For any discrete memoryless channel D, with n-th capacity cost function $C_n(\beta)$, we have $C_n(\beta)$ is concave down for all $\beta\beta_{\min}$.*

Proof. Let $\alpha_1, \alpha_2 > 0$ with $\alpha_1 + \alpha_2 = 1$; and let $\beta_1, \beta_2 \geq \beta$. Now, let x_1 and x_2 be test sources, with probailities $p_1(x)$ and $p_2(x)$, achieving $C_n(\beta_1)$ and $C_n(\beta_2)$. That is, $I(X_i, Y_i) = C_n(\beta_i)$ and $\overline{x_i} \leq n\beta_i$ for $1 \leq i \leq n$ and random variables Y_1 and Y_2 associated with the outputs of X_1 and X_2. Now, define X to be the test source with probaility distribution $p(x) = 1p(x) + \alpha_2 p(x)$. Then:

$$\bar{b}(x) = \sum_x p(x)x = \alpha_1 \sum p_1(x)b(x_1) + \alpha_2 \sum p_2(x)b(x_2) \leq \alpha_1 \bar{b}(x_1) + \alpha_2 \bar{b}(x_2) \leq n(\alpha_1\beta_1 + \alpha_2\beta_2)$$

Additionally, since $I(X,Y)$ is concave down, we get

$$C_n(\alpha_1\beta_1 + \alpha_2\beta_2) \geq I(X,Y) \geq \alpha_1 I(X_1,Y_1) + \alpha_2 I(X_2,Y_2) = \alpha_1 C_n(\beta_1) + \alpha_2 C_n(\beta_2)$$

∎

Theorem 3.1.3. *For any DMC, $C_n(\beta) = nC_1(\beta)$.*

Proof. Let $x = \{x_i\}$ be a β-admissable test source achieving $C_n(\beta)$. Then $I(X,Y) \leq \sum I(X_i, Y_i)$. Now, defining $\beta_i = \bar{b}(x_i)$, we get $\sum \beta_i = \sum \bar{b}(x_i) = \sum\sum p(x_i)b(x_i) = \bar{b}(x) \leq n\beta$. We also have $I(X_i, Y_i) \leq C_1(\beta_i)$, ∎