

# Cryptanalyse de la cryptographie classique

MOHAMED MEJRI

*Groupe LSFM*

*Département d'Informatique et de Génie Logiciel*

*l'Université LAVAL*

*Québec, Canada*

## Plan

- ⇒ Introduction
- ⇒ Cryptanalyse par recherche exhaustive des clés :
  - Cryptanalyse du chiffrement affine
- ⇒ Cryptanalyse linéaire :
  - Cryptanalyse du chiffrement de Hill
- ⇒ Cryptanalyse par analyse de fréquences :
  - Idée
  - Cryptanalyse du chiffrement affine
  - Cryptanalyse du chiffrement de Vigenere
  - Mots probables
- ⇒ Vers un système cryptographique parfait

## Introduction

- ⇒ **Types d'attaques** : On peut classer les attaques selon les informations disponibles aux cryptanalystes.
- **Attaque à texte chiffré** : L'analyste se dispose de textes chiffrés  $c_1, \dots, c_n$  et cherche à trouver leurs correspondants en clair.
  - **Attaque à texte clair** : L'analyste se dispose de textes en clair  $m_1, \dots, m_n$  et de leurs chiffréments  $c_1, \dots, c_n$  respectifs et essaye de trouver la clé du cryptage ou de décrypter d'autres textes.
  - **Attaque à texte clair choisi** : L'analyste peut choisir des textes clairs et obtenir leurs textes chiffrés correspondants. En ayant ces connaissances, il essaye de trouver la clé du cryptage ou de décrypter d'autres textes.
  - Etc.

## Introduction

- ⇒ **Types de sécurité d'un cryptosystème** : La sécurité d'un cryptosystème peut se mesurer par la difficulté de le casser.
- ◆ **Inconditionnellement sûr** : Si un cryptanalyste ne peut pas trouver le texte en clair, quelle que soit les ressources dont il dispose.
  - ◆ **Algorithmiquement sûr** : S'il ne peut être cassé avec les ressources disponibles dans un temps raisonnable.
  - ◆ **Autre** : Les idées sur lesquelles se base la sécurité du cryptosystème semblent raisonnables. Mais, il n'y a aucune preuve qui garantit sa sécurité.

## Introduction

⇒ **Analyse de système cryptographique** : Préciser les hypothèses et l'objectif.

- Modèle de l'intrus : ses éventuelles connaissances (texte clair seulement, textes chiffrés et leurs correspondants en claires) et sa capacité de calcul (limité ou illimité), passif ou actif, etc.
- Objectif de l'intrus :
  - décrypter des messages sans connaître la clé.
  - encrypter des messages sans connaître la clé.
  - trouver la clé.
  - etc.

Un système cryptographique peut être attaqué en dehors des hypothèses que nous fixons

## Introduction

⇒ Qu'est-ce qu'un "bon" système cryptographique ? : Besoin d'une définition plus précise.

- Ce n'est pas suffisant de dire qu'un cryptosystème est sécuritaire si l'intrus ne trouve pas les clés. Le système  $E_k(m) = m$ , ne permet jamais à l'intrus de trouver la clé à partir de  $m$ .
- Ce n'est pas suffisant de dire qu'un cryptosystème est sécuritaire si l'intrus ne trouve pas les messages clairs. C'est mauvais si l'intrus trouve une partie d'un message.
- Ce n'est pas suffisant de dire qu'un cryptosystème est sécuritaire si l'intrus ne trouve aucun bit des messages clairs. Si les messages sont les salaires des employés, c'est mauvais si l'intrus ne trouve aucun chiffre d'un salaire, mais il peut dire dans quel intervalle il se trouve.

Intuitivement, un système cryptographique est "bon" s'il ne permet pas à l'intrus de déduire des informations sur des messages clairs à partir des messages chiffrés.

- Définition d'un système parfait :  $Pr(m/c) = Pr(c)$
- Définition d'un bon système :  $Pr(m/c) = Pr(c) \pm \epsilon$  Si l'intrus nous envoie deux messages  $m_1$  et  $m_2$  et on lui retourne  $c_1$  et  $c_2$  leurs correspondants chiffrés, alors il ne lui sera pas possible de dire lequel qui correspond à  $m_1$  avec une probabilité égale  $\frac{1}{2} + \epsilon$  avec  $\epsilon$  non négligeable ( $\epsilon < 2^{-80}$ ).

## Introduction

### ⇒ Techniques :

- Bien comprendre le système cryptographique en question.
- Dégager ses propriétés.
- Exploiter ses propriétés pour en déduire ses faiblesses.

## Plan

- ⇒ Introduction
- ⇒ Cryptanalyse par recherche exhaustive des clés :
  - Cryptanalyse du chiffrement affine
- ⇒ Cryptanalyse linéaire :
  - Cryptanalyse du chiffrement de Hill
- ⇒ Cryptanalyse par analyse de fréquences :
  - Idée
  - Cryptanalyse du chiffrement affine
  - Cryptanalyse du chiffrement de Vigenere
  - Mots probables
- ⇒ Vers un système cryptographique parfait



## Recherche exhaustive de la clé (Brut force Attack)

### ⇒ Idées :

- Un système cryptographique manipule un ensemble fini de clés (espace de clés)
- Si l'espace de clés est petit alors un adversaire peut les essayer une par une, jusqu'à ce qu'il trouve la bonne.

$(0,0,0); (0,0,1); (0,0,2); \dots$



J'essaie toutes les possibilités  
et je vais l'avoir... 😊

## Cryptanalyse du chiffrement par décalage

⇒ **Rappel :**  $e_k(x) = x + k \bmod 26$  et  $d_k(x) = x - k \bmod 26$ .

⇒ **Remarque :** Il y a 26 clés possibles  $\Rightarrow$  on peut rapidement les parcourir.

⇒ **Exemple :** Le message crypté est  $C = JZCBM\ NWZKM$

$K = 0 \Rightarrow D_0(C) = JZCBM\ NWZKM$

$K = 1 \Rightarrow D_1(C) = IYBAL\ MVYJL$

$K = 2 \Rightarrow D_2(C) = HXAZK\ LUXIK$

$K = 3 \Rightarrow D_3(C) = GWZYJ\ KTW HJ$

$K = 4 \Rightarrow D_4(C) = FVYXI\ JSVGI$

$K = 5 \Rightarrow D_5(C) = EUXWH\ IRUFH$

$K = 6 \Rightarrow D_6(C) = DTWVG\ HQTEG$

$K = 7 \Rightarrow D_7(C) = CSVUF\ GPSDF$

**$K = 8 \Rightarrow D_8(C) = BRUTE\ FORCE$**

Aha ! j'ai trouvé la clé  $K = 8$

## Recherche exhaustive de la clé

- ➡ **Limites :** Pour que cette technique soit réalisable, il faut que l'espace de clé ait une taille raisonnable.
- ➡ **Question :** Peut-on appliquer cette technique sur le chiffrement du Vigenere avec un clé de taille 20
  - ➡ **Réponse :** Non, il y a trop de clé à explorer  $26^{20}$
  - ➡ **Question :** C'est peut être trop pour moi, mais est ce que c'est trop pour mon ordinateur qui peut faire 2 milliard d'opérations par secondes ( $2Ghz = 2 * 10^9 o.p.s$ ) ?

## Recherche exhaustive de la clé

### ⇒ Limites (suite) :

- ♦ Réponse : Oui c'est trop pour ton ordinateur également. En effet :
- Avec ton ordinateur, il te prend

$$\begin{aligned}\frac{26^{20}}{2 \times 10^9} &\approx 9964074447604704576 && \text{secondes} \\ &\approx 166067907460078409 && \text{minutes} \\ &\approx 2767798457667973 && \text{heures} \\ &\approx 115324935736165 && \text{jours} \\ &\approx 315095452831 && \text{années} \\ &\approx 315 && \text{milliard d'années}\end{aligned}$$

♦ Remarque :  $26^{20} > \underbrace{(2^4)}_{16}^{20} = 2^{80}$

## Recherche exhaustive de la clé

⇒ **Limites (suite)** : Supposons qu'on a des capacités infinies (des ordinateurs ultra-puissants et qu'on est éternel !).

• Question : La recherche exhaustive nous permet-elle de casser n'importe quel système qui utilise un nombre fini de clés ?

• Réponse : Pas de tout ! En effet :

⇒ cette technique repose sur le fait que seule la bonne clé qui déchiffre le message crypté en un message qui a un "sens".

– Exemple : Le message crypté est  $C = JZCBM\ NWZKM$

$$K = 0 \Rightarrow D_0(C) = JZCBM\ NWZKM$$

$$K = 1 \Rightarrow D_1(C) = IYBAL\ M VYJL$$

$$K = 2 \Rightarrow D_2(C) = HXAZK\ LUXIK$$

$$K = 3 \Rightarrow D_3(C) = GWZYJ\ K TWHJ$$

$$K = 4 \Rightarrow D_4(C) = FVYXI\ JSVGI$$

$$K = 5 \Rightarrow D_5(C) = EUXWH\ IRUFH$$

$$K = 6 \Rightarrow D_6(C) = DTWVG\ HQTEG$$

$$K = 7 \Rightarrow D_7(C) = CSVUF\ GPSDF$$

$$K = 8 \Rightarrow D_8(C) = BRUTE\ FORCE$$

Aha ! j'ai trouvé la clé  $K = 8$

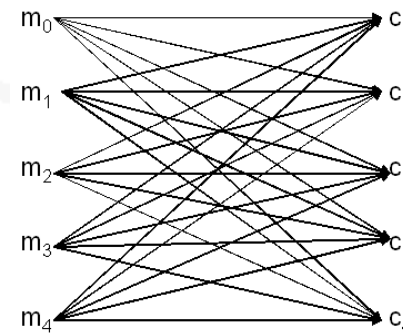
## Recherche exhaustive de la clé

- ⇒ **Attaque à texte chiffré** Si on connaît juste le message chiffré.
- ⇒ Qu'advient-il si toutes les clés ou une grande partie d'entre elles donnent des textes qui ont un "sens". Laquelle de ces clés est la bonne ?
  - Exemple 1 : Le message crypté avec le chiffrement par décalage est  $C = WNAJW$ .

$$\begin{aligned} K = 5 &\Rightarrow D_5(C) &= & river \\ K = 22 &\Rightarrow D_{10}(C) &= & arena \end{aligned}$$

- Exemple 2 : Soient  $\{m_0, \dots, m_4\}$  cinq messages qui ont un "sens" et  $S = (\mathcal{P}, \mathcal{C}, \mathcal{K}, \mathcal{E}, \mathcal{D})$  le système suivant :

$$\begin{aligned} \mathcal{P} &= \{m_0, \dots, m_4\} \\ \mathcal{C} &= \{c_0, \dots, c_4\} \\ \mathcal{K} &= \{k_0, \dots, k_4\} \\ e_{k_i}(m_j) &= c_{i+j \bmod 5} \end{aligned}$$

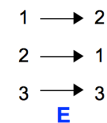


## Recherche exhaustive de la clé

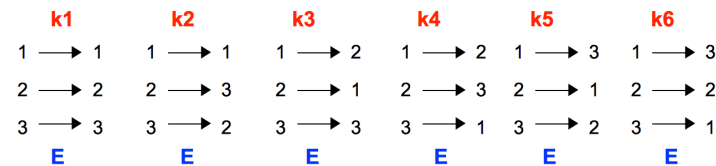
⇒ **Attaque à texte clair** Si on connaît des messages et leurs correspondants chiffrés

⇒ Ayant  $(c, m)$ , combien de clés peuvent chiffrer  $m$  en  $c$  ?

- Un système réversible de  $n$  bits vers  $n$  bits attribue aux valeurs  $(0, 1, \dots, 2^n - 1)$  une permutation de  $(0, 1, \dots, 2^n - 1)$



- Un système "parfait" attribue aléatoirement une permutation différente à chaque clé.



- Pour  $(m, c)$ , on aura  $n$  permutation ( $n$  clés) parmi  $n!$  qui donne  $E_k(m) = c$ .
- Pour DES, la probabilité que deux clés différentes produisent le même résultat sur le même message ( $E_{k_1}(m) = E_{k_2}(m)$ ) est inférieur à  $\frac{1}{2}^{71}$ , pour AES c'est  $\frac{1}{2}^{128}$
- Si on a deux couples  $(m_1, c_1)$  et  $(m_2, c_2)$  et on trouve une clé telle que  $DES_k(m_1) = c_1$  et  $DES_k(m_2) = c_2$ , alors la probabilité que  $k$  soit unique est supérieure à  $1 - \frac{1}{2}^{71}$

## Recherche exhaustive de la clé

- ⇒ **Attaque à texte clair** La clé DES de 56 bits était sécuritaire il y a 20 ans, mais susceptible aux attaques par recherche exhaustive avec la technologie actuelle.
- ⇒ **Défi lancé par RSA (prix 10 000US\$)** : message = "The unknown messages is : XXXX...."
- $m_1$ ="The unkn" et  $DES(k, m_1) = c_1$   
 $m_2$ ="own mess" et  $DES(k, m_2) = c_2$   
 $m_3$ ="ages is :" et  $DES(k, m_3) = c_3$   
 $c_4, \dots, c_n$  sont aussi donnés mais leurs parties claires ne sont pas connues
- ⇒ **Objectif** trouver  $k \in \{0, 1\}^{56}$  telle que  $DES(k, m_i) = c_i$  avec  $i \in \{1, 2, 3\}$
- ⇒ **1997** : recherche en utilisant le réseau Internet : **3 mois**
- ⇒ **1998** : machine EFF (deep crack) : **3 jours** (250K \$)
- ⇒ **2003** : **4 heures**, (120 K Euro)
- ⇒ **2006** : **7 jours** projet COPACOBANA : 120 FPGAs (10K\$)

**Ne plus utiliser DES simple : si vous oubliez votre clé DES, vous pouvez la retrouver dans 7 j**

**Pour AES-128 : 128 bits de clé  $\implies$  plus que  $2^{72}(2^{128-56})$  jour**

Et RSA 1024bits ?



## Plan

### ⇒ Introduction

### ⇒ Cryptanalyse par recherche exhaustive des clés :

- Cryptanalyse du chiffrement affine

### ⇒ Cryptanalyse linéaire :

- Cryptanalyse du chiffrement de Hill

### ⇒ Cryptanalyse par analyse de fréquences :

- Idée
- Cryptanalyse du chiffrement affine
- Cryptanalyse du chiffrement de Vigenere
- Mots probables

### ⇒ Vers un système cryptographique parfait

## Cryptanalyse du chiffrement de Hill

### ⇒ Rappel :

- La clé est une matrice inversible dans  $\mathbb{Z}_{26}$
- Encrypter un message  $x$  de longueur  $m$  :  $e_K(x) = xK$
- Décrypter un message  $y$  de longueur  $m$  :  $d_K(y) = yK^{-1}$
- Tout le calcul se fait dans  $\mathbb{Z}_{26}$ .

### ⇒ Recherche de la clé :

- Supposons qu'on connaît  $m$ , la taille de la matrice clé.
- Si la valeur de  $m$  n'est pas connue, on fait plusieurs essais  $m = 1, m = 2$ , etc.
- Supposons qu'on a  $m$  pairs  $(x_1, y_1), \dots, (x_m, y_m)$  telles que  $e_K(x_i) = y_i, 1 \leq i \leq m$ .
- $x_i, 1 \leq i \leq m$ , est une chaîne de  $m$  caractères :  $x_i = x_{i1} \dots x_{im}$
- $y_i, 1 \leq i \leq m$ , est une chaîne de  $m$  caractères :  $y_i = y_{i1} \dots y_{im}$

## Cryptanalyse du chiffrement de Hill

⇒ Recherche de la clé (suite) :

– On a :

$$\begin{pmatrix} x_{11} & \dots & x_{1m} \\ \vdots & \vdots & \vdots \\ x_{m1} & \dots & x_{mm} \end{pmatrix} K = \begin{pmatrix} y_{11} & \dots & y_{1m} \\ \vdots & \vdots & \vdots \\ y_{m1} & \dots & y_{mm} \end{pmatrix}$$

– Donc

$$K = \begin{pmatrix} x_{11} & \dots & x_{1m} \\ \vdots & \vdots & \vdots \\ x_{m1} & \dots & x_{mm} \end{pmatrix}^{-1} \times \begin{pmatrix} y_{11} & \dots & y_{1m} \\ \vdots & \vdots & \vdots \\ y_{m1} & \dots & y_{mm} \end{pmatrix}$$

## Cryptanalyse du chiffrement de Hill

### ⇒ Recherche de la clé (suite) :

- On déduit que pour trouver  $K$ , il suffit de trouver

$$\begin{pmatrix} x_{11} & \dots & x_{1m} \\ \vdots & \vdots & \vdots \\ x_{m1} & \dots & x_{mm} \end{pmatrix}^{-1}$$

- Si cette matrice n'est pas inversible, on essaie avec un autre ensemble de couples  $\{(x'_1, y'_1), \dots, (x'_m, y'_m)\}$ .

## Cryptanalyse du chiffrement de Hill

### ⇒ Exemple :

- Supposons que  $m = 2$  et que  $e_K(\text{FRIDAY}) = e_K(\text{FR})e_K(\text{ID})e_K(\text{AY}) = \text{PQCFKU}$
- Puisque  $\text{FRIDAY} = 5 \ 17 \ 8 \ 3 \ 0 \ 24$  et  $\text{PQCFKU} = 15 \ 16 \ 2 \ 5 \ 10 \ 20$
- Donc :

$$\begin{pmatrix} 15 & 16 \\ 2 & 5 \end{pmatrix} = \begin{pmatrix} 5 & 17 \\ 8 & 3 \end{pmatrix} K$$

- Or :

$$\begin{pmatrix} 5 & 17 \\ 8 & 3 \end{pmatrix}^{-1} = \begin{pmatrix} 9 & 1 \\ 2 & 15 \end{pmatrix}$$

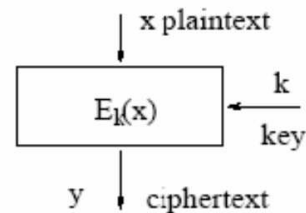
- On conclut que :

$$K = \begin{pmatrix} 9 & 1 \\ 2 & 15 \end{pmatrix} \begin{pmatrix} 15 & 16 \\ 2 & 5 \end{pmatrix} = \begin{pmatrix} 7 & 19 \\ 8 & 3 \end{pmatrix}$$

## Cryptanalyse du chiffrement de Hill

### ⇒ Fonction linéaire

- $f(x + y) = f(x) + f(y)$
- $f(a.x) = a.f(x)$
- en binaire :  $f(x \oplus y) = f(x) \oplus f(y)$



$$\begin{aligned}
 y_1 &= x_1 + x_2 + x_3 + x_4 + k_1 + k_2 + k_3 + k_4 \\
 y_2 &= x_2 + x_3 + x_4 + x_5 + k_2 + k_3 + k_4 + k_5 \\
 y_3 &= x_3 + x_4 + x_5 + x_6 + k_3 + k_4 + k_5 + k_6 \\
 y_4 &= x_4 + x_5 + x_6 + x_7 + k_4 + k_5 + k_6 + k_7 \\
 y_5 &= x_5 + x_6 + x_7 + x_8 + k_5 + k_6 + k_7 + k_8 \\
 y_6 &= x_6 + x_7 + x_8 + x_1 + k_6 + k_7 + k_8 + k_1 \\
 y_7 &= x_7 + x_8 + x_1 + x_2 + k_7 + k_8 + k_1 + k_2 \\
 y_8 &= x_8 + x_1 + x_2 + x_3 + k_8 + k_1 + k_2 + k_3
 \end{aligned}$$

## Cryptanalyse du chiffrement de Hill

⇒ **Non linéarité d'une fonction** : C'est la distance de hamming à la fonction linéaire la plus proche

| x1 | x2 | f(x1,x2)=<br>x1x2 | g1(x1,x2)=<br>0 | g2(x1,x2)=<br>1 | g3(x1,x2)=<br>x1 | g4(x1,x2)=<br>x2 | g4(x1,x2)=<br>x1⊕x2 | ... |
|----|----|-------------------|-----------------|-----------------|------------------|------------------|---------------------|-----|
| 0  | 0  | 0                 | 0               | 1               | 0                | 0                | 0                   | ... |
| 0  | 1  | 0                 | 0               | 1               | 0                | 1                | 1                   | ... |
| 1  | 0  | 0                 | 0               | 1               | 1                | 0                | 1                   | ... |
| 1  | 1  | 1                 | 0               | 1               | 1                | 1                | 0                   | ... |

Les fonctions affines de  $(x1,x2) \rightarrow \{0,1\}$  sont

|                |                         |
|----------------|-------------------------|
| 0              | 1                       |
| x1             | x2                      |
| $x1 \oplus 1$  | $x2 \oplus 1$           |
| $x1 \oplus x2$ | $x1 \oplus x2 \oplus 1$ |

La non-linéarité de f est 1 : si f est approximée par g1 ou g3 alors notre approximation nous donne le bon résultat trois fois sur quatre

## Cryptanalyse du chiffrement de Hill

⇒ **Non linéarité d'une fonction** : C'est la distance de hamming à la fonction linéaire la plus proche

- La fonction  $f(x_1, x_2, x_3, x_4) = x_1 \oplus x_2 \oplus x_3 \oplus x_4 \oplus x_1.x_2.x_3.x_4$  a une mauvaise non linéarité.
- Soit  $g(x_1, x_2, x_3, x_4) = x_1 \oplus x_2 \oplus x_3 \oplus x_4$
- $g$  est linéaire
- $f$  et  $g$  produisent toujours les mêmes résultats sauf lorsque  $x_1 = x_2 = x_3 = x_4 = 1$



## Plan

### ⇒ Introduction

### ⇒ Cryptanalyse par recherche exhaustive des clés :

- Cryptanalyse du chiffrement affine

### ⇒ Cryptanalyse linéaire :

- Cryptanalyse du chiffrement de Hill

### ⇒ Cryptanalyse par analyse de fréquences :

- Idée
- Cryptanalyse du chiffrement affine
- Cryptanalyse du chiffrement de Vigenere
- Mots probables

### ⇒ Vers un système cryptographique parfait

## Analyse de fréquences

⇒ **Origine :** Approche introduite par Abu Youssif Al-Kindi (IXe siècle)



⇒ **Idée :**

- établir la fréquence de chaque lettre de l'alphabet. En français la lettre la plus fréquente est 'e' suivie par 'a' puis par 's', puis par ...

( référence <http://www.apprendre-en-ligne.net/crypto/menu/index.html>)

- Examiner les fréquences des caractères dans le texte chiffré.
- Remplacer les caractères les plus fréquents du texte chiffré par les caractères les plus fréquents du langage.
- Si par exemple la lettre la plus fréquente du texte chiffré est 'j', suivie par 'm', suivie par 'k' alors on fait un premier essai en remplaçant 'j' par 'e', 'm' par 'a' et 'k' par 's'.

## Cryptanalyse du chiffrement affine

### ⇒ Cryptanalyse du chiffrement affine

#### ◆ Texte à décrypter :

FMNVEDKAPHFERBNDKRX  
RSREFMORUDSDKDVSHVU  
FEDKAPRKDLYEVLRRHHRH

#### ◆ Comment le décrypter ? l'idée est d'utiliser la méthode de fréquences.

| Letter | Frequency |
|--------|-----------|
| E      | .127      |
| T      | .091      |
| A      | .082      |
| O      | .075      |
| I      | .070      |
| N      | .067      |
| S      | .063      |
| H      | .061      |

Fréquences moyennes des lettres

| Letter   | # of Occurrences |
|----------|------------------|
| <i>R</i> | 8                |
| <i>D</i> | 6                |
| <i>E</i> | 5                |
| <i>H</i> | 5                |
| <i>K</i> | 5                |
| <i>V</i> | 4                |
| <i>F</i> | 4                |

Occurrences des lettres dans le texte crypté

## Cryptanalyse du chiffrement affine

### ⇒ Cryptanalyse de la chiffrement affine (suite)

- Connu : On sait que l'algorithme de cryptage est :

$$E_k(x) = a * x + b \text{ mod } 26$$

- À trouver : Les deux variables  $a$  et  $b$ .
- 1er essai :  $(R \mapsto E)$  et  $(D \mapsto T)$ .
  - $(E_k(E) = R)$  et  $(E_k(T) = D)$ .
  - $(E_k(4) = 17)$  et  $(E_k(19) = 5)$ .
  - $(4 * a + b = 17 \text{ mod } 26)$  et  $(19 * a + b = 5 \text{ mod } 26)$ .
  - $(a = 20)$  et  $(b = 15)$ .
  - La solution n'est pas acceptable car  $\text{pgcd}(20, 26) > 1$ .

## Cryptanalyse du chiffrement affine

### ⇒ Cryptanalyse de la chiffrement affine (suite)

- ◆ 2ème essai :  $(R \mapsto E)$  et  $(E \mapsto T)$ .
  - $(E_k(E) = R)$  et  $(E_k(T) = E)$ .
  - $(E_k(4) = 17)$  et  $(E_k(19) = 4)$ .
  - $(4 * a + b = 17 \text{ mod } 26)$  et  $(19 * a + b = 4 \text{ mod } 26)$ .
  - $(a = 13)$  et  $(b = 17)$ .
  - La solution n'est pas acceptable car  $\text{pgcd}(13, 26) > 1$ .

## Cryptanalyse du chiffrement affine

### ⇒ Cryptanalyse de la chiffrement affine (suite)

- ◆ 3ème essai :  $(R \mapsto E)$  et  $(H \mapsto T)$ .
  - $(E_k(E) = R)$  et  $(E_k(T) = H)$ .
  - $(E_k(4) = 17)$  et  $(E_k(19) = 7)$ .
  - $(4 * a + b = 17 \text{ mod } 26)$  et  $(19 * a + b = 7 \text{ mod } 26)$ .
  - $(a = 8)$  et  $(b = 11)$ .
  - La solution n'est pas acceptable car  $\text{pgcd}(8, 26) > 1$ .

## Cryptanalyse du chiffrement affine

### ⇒ Cryptanalyse de la chiffrement affine (suite)

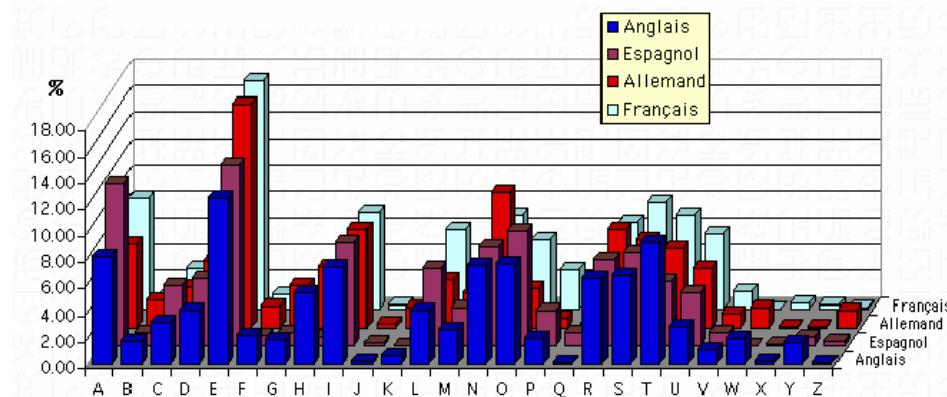
❖ 4ème essai :  $(R \mapsto E)$  et  $(K \mapsto T)$ .

- $(E_k(E) = R)$  et  $(E_k(T) = K)$ .
- $(E_k(4) = 17)$  et  $(E_k(19) = 10)$ .
- $(4 * a + b = 17 \text{ mod } 26)$  et  $(19 * a + b = 10 \text{ mod } 26)$ .
- $(a = 3)$  et  $(b = 5)$ .
- La solution est acceptable car  $\text{pgcd}(3, 26) = 1$ .

❖ Conclusion : La technique du chiffrement affine succombe facilement aux attaques à textes chiffrés seulement.

## Analyse de fréquences

- ⇒ **Idée (suite) :** D'une manière plus générale, l'idée est :
- d'étudier l'effet de l'utilisation de certains systèmes cryptographiques sur les statistiques (fréquences de monogramme, fréquence de bigrammes, etc.).
  - et de déduire par la suite des informations sur le système cryptographiques utilisé, la clé, etc.
- ⇒ **Fréquences des lettres dans différentes langues :**

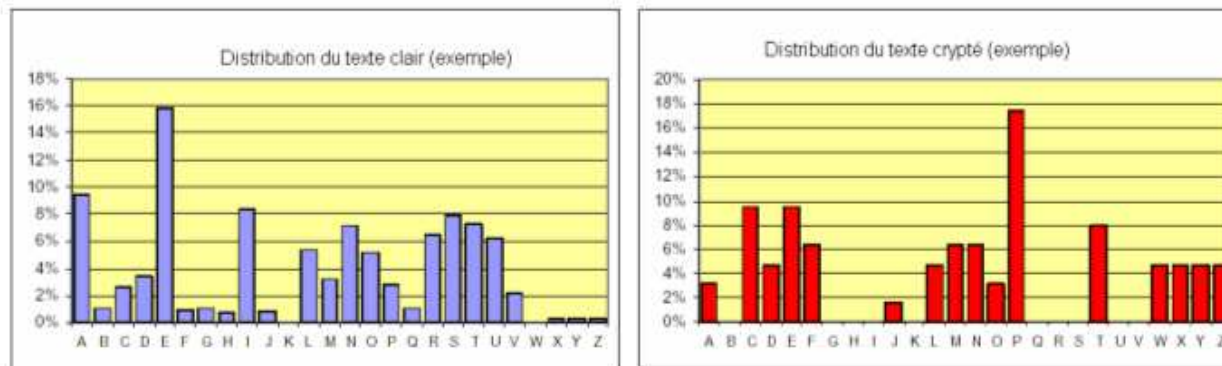




## Analyse de fréquences

### Remarques :

- Dans un texte quelconque chaque lettre a une fréquence d'apparition.
- Les probabilités de toutes les lettres définissent la distribution de ce texte.
- Soit  $D_C$  la distribution d'un texte clair.
- Soit  $D_E$  la distribution de son correspondant crypté.
- Le lien entre  $D_E$  et  $D_C$  dépend du cryptosystème.
- Un bon cryptosystème doit détruire le lien entre  $D_E$  et  $D_C$ .



## Analyse de fréquences

### ⇒ Observations :

- Si le cryptosystème est le chiffrement par permutation alors  $D_E = D_C$ .

|   |   |   |   |  |   |  |   |   |   |   |  |   |   |   |
|---|---|---|---|--|---|--|---|---|---|---|--|---|---|---|
| t | h | e | t |  | E |  |   |   |   | E |  |   |   |   |
| r | u | t | h |  | H |  |   |   |   | H |  |   |   |   |
| i | s | o | u |  | I |  |   |   |   | I |  |   |   |   |
| t | t | h | e |  | O |  |   |   |   | O |  |   |   |   |
| r | e | x | x |  | R |  |   |   |   | R |  |   |   |   |
|   |   |   |   |  | S |  |   |   |   | S |  |   |   |   |
|   |   |   |   |  | T |  | + | + | + | T |  | + | + | + |
|   |   |   |   |  | U |  |   |   |   | U |  |   |   |   |
|   |   |   |   |  | X |  |   |   |   | X |  |   |   |   |

⇒ TRITR THUEX HUSTE ETOHX

- Si le cryptosystème est une substitution monoalphabétique alors  $D_E$  est une permutation  $D_C$ .

|     |     |     |   |     |     |     |
|-----|-----|-----|---|-----|-----|-----|
| one | one | two | ↦ | RQH | RQH | WZR |
| O   |     |     |   | R   |     |     |
| E   |     |     |   | H   |     |     |
| N   |     |     |   | Q   |     |     |
| T   |     |     |   | W   |     |     |
| W   |     |     |   | Z   |     |     |

## Analyse de fréquences

### ⇒ Observations (suite) :

- Si le cryptosystème est une substitution polyalphabétique alors  $D_E$  est différent de  $D_C$ .

cluecluecluecl  
teleconference  $\mapsto$  VPFIEZHJGCRYEP

|   |  |
|---|--|
| E |  |
| C |  |
| N |  |
| F |  |
| L |  |
| O |  |
| T |  |

|   |  |
|---|--|
| E |  |
| P |  |
| C |  |
| F |  |
| G |  |
| H |  |
| I |  |
| J |  |
| R |  |
| V |  |
| Y |  |
| Z |  |

⇒ Question : Y a-t-il un moyen qui permet de savoir si  $D_E$  est une permutation  $D_C$  ?

⇒ Réponse : Oui, en utilisant l'indice de coïncidence.

## Indice de coïncidence

⇒ **Définition :** Défini par FRIEDMAN en 1920, comme suit :

- Soit  $x = x_1x_2 \dots x_n$  une chaîne de caractères.
- L'indice de coïncidence de  $x$ , noté par  $I_c(x)$ , est la probabilité que :
  - deux caractères  $x_i$  et  $x_j$  ( $i \neq j$ ) de  $x$ ,
  - choisis aléatoirement,
  - sont identiques.

## Indice de coïncidence

### ⇒ Calcul de $I_c(x)$ :

• Soient :

- $x = x_1 x_2 \dots x_n$  une chaîne de caractères (un texte) qui ne contient que les lettres  $a, b, \dots, z$ .
- $n_0, \dots, n_{25}$  sont respectivement les nombres d'occurrences des lettres  $a, b, \dots, z$  dans la chaîne  $x$ .
- $p_0, \dots, p_{25}$  est la distribution de la langue (français, anglais, etc.) dans laquelle le texte  $x$  est écrit.

• Alors  $I_c(x) = \frac{\sum_{i=0}^{25} n_i(n_i-1)}{n(n-1)} \xrightarrow{n \rightarrow \infty} \approx \sum_{i=0}^{25} p_i^2$ . En effet :

$$\begin{aligned} I_c(x) &= \frac{\sum_{i=0}^{25} n_i(n_i-1)}{n(n-1)} \\ &\xrightarrow{n \rightarrow \infty} \sum_{i=0}^{25} \left(\frac{n_i}{n}\right)^2 \\ &\approx \sum_{i=0}^{25} p_i^2 \end{aligned}$$

## Indice de coïncidence

⇒ Quelques valeurs de  $I_c(x)$  :

| Langue   | Indice de coïncidence |
|----------|-----------------------|
| Français | $\approx 0.078$       |
| Anglais  | $\approx 0.066$       |
| Espagnol | $\approx 0.078$       |

⇒ **Remarques** : Plus la distribution est plate, plus la valeur de  $I_c$  diminue. Le minimum est obtenu lorsque la distribution est complètement plate.

◆ Distribution de la langue anglaise :

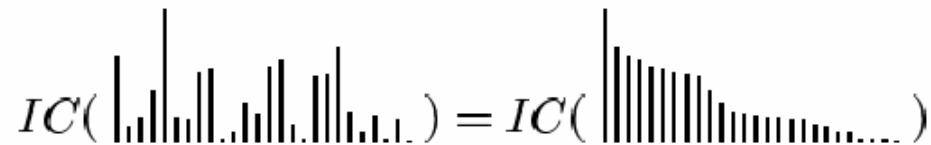


◆ Distribution plate :

$$IC(\text{Distribution plate}) = 1/26 = 0.038$$

## Indice de coïncidence

⇒ **Effet d'une permutation** : La permutation des fréquences d'une distribution n'a pas d'influence sur  $I_c$ .


$$IC(\text{distribution 1}) = IC(\text{distribution 2})$$

⇒ **Lien avec les cryptosystèmes** : Si le cryptosystème est monoalphabétique alors le  $I_c$  du texte clair est le même que celui du texte crypté.

⇒ **Exemple d'utilisation** : L'indice de coïncidence joue un rôle important dans la cryptanalyse du chiffrement de Vigenere (voir plus loin).

## Exercice

⇒ **Hypothèses :** Supposons que :

◆ Vous avez intercepté le texte suivant :

|       |       |       |       |       |       |       |       |       |       |
|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|
| HZPKY | OMEML | HXSBN | JWSXD | RIDBS | PERBC | VQLEL | WLTGG | VWSXD | RIDCU |
| VXENR | QQPHN | HZPGT | KSFZH | PCWBF | HFPYO | UISTS | EIPGT | UERBC | QSHBK |
| QSHFY | OSGXF | RVSXR | JSPLO | Q     |       |       |       |       |       |

◆ Le texte clair correspondant à ce texte chiffré était écrit en anglais.

◆ Le chiffrement était effectué en utilisant l'un des cryptosystèmes classiques vus dans ce cours.

⇒ **Travail demandé :** Déterminer le cryptosystème utilisé pour chiffrer le message original.



## Cryptanalyse du chiffrement de Vigenere

- ⇒ **Objectif** : À travers un exemple, on donnera des techniques qui permettent de déchiffrer un message crypté à l'aide du chiffrement de Vigenere.
- ⇒ **L'exemple à étudier** : Supposons que :
  - **Texte crypté** : Vous avez intercepté le texte suivant :  
NLPKMVGNSOXYPTGCEMQYHGD TG YWGPWGEHGD SRTRK  
ZRUPWVFRFPWFCSKEWNPWRWYUAVGNMGFBFPPJZQOP  
XQFXETXQJIPAIWEHQYGR LVNPVGNVKCIKXTTTQGCP  
KMVGX IPEWCFJCCIRZRFCIFPPCMYUOIEPXVPPKMIT  
EIFLRUWIUNEUOIVPVOTRGDTCCPCWSK
  - **Langue** : Le texte clair correspondant a été écrit en français.
  - **Cryptosystème** : Il était produit par le chiffrement de Vigenere.
  - **Clé** : La longueur de la clé est 3 (plus tard on verra des techniques qui permettent de déterminer la longueur de la clé)

## Cryptanalyse du chiffrement de Vigenere

### ⇒ Cryptanalyse :

- Par hypothèse, on sait que la longueur de la clé=3. En quoi cela peut nous aider ?
- C'est simple ! puisque le chiffrement est de Vigenere alors, on déduit que :
  - Les lettres qui ont la position  $3*i$  ( $i \geq 0$ ) ont été cryptées par la simple technique de décalage (monoalphabétique) :

NKGOPCQGGGGGRKUVFFKNRUGGFJOQEQPW

QRNGKKTGKGPCCRFFCUEVKTFUUUVOGCCK

- La même chose pour lettres qui ont la position  $3*i + 1$  ( $i \geq 0$ ) :

LMNXTEYDYPEDTZPFPCEPWANFPZPFTJAE

YLPNCXTCMXEFCZCPMOPPMELWNOPTDCW

- Et la même chose pour lettres qui ont la position  $3*i + 2$  ( $i \geq 0$ ) :

PVSYGMHTWWHSRRWRWSWWYVMBPQXXXIIH

GVVVITQPVIWJIRIPYIXPIIRIEIVRTPS

## Cryptanalyse du chiffrement de Vigenere

### ⇒ Cryptanalyse (suite) :

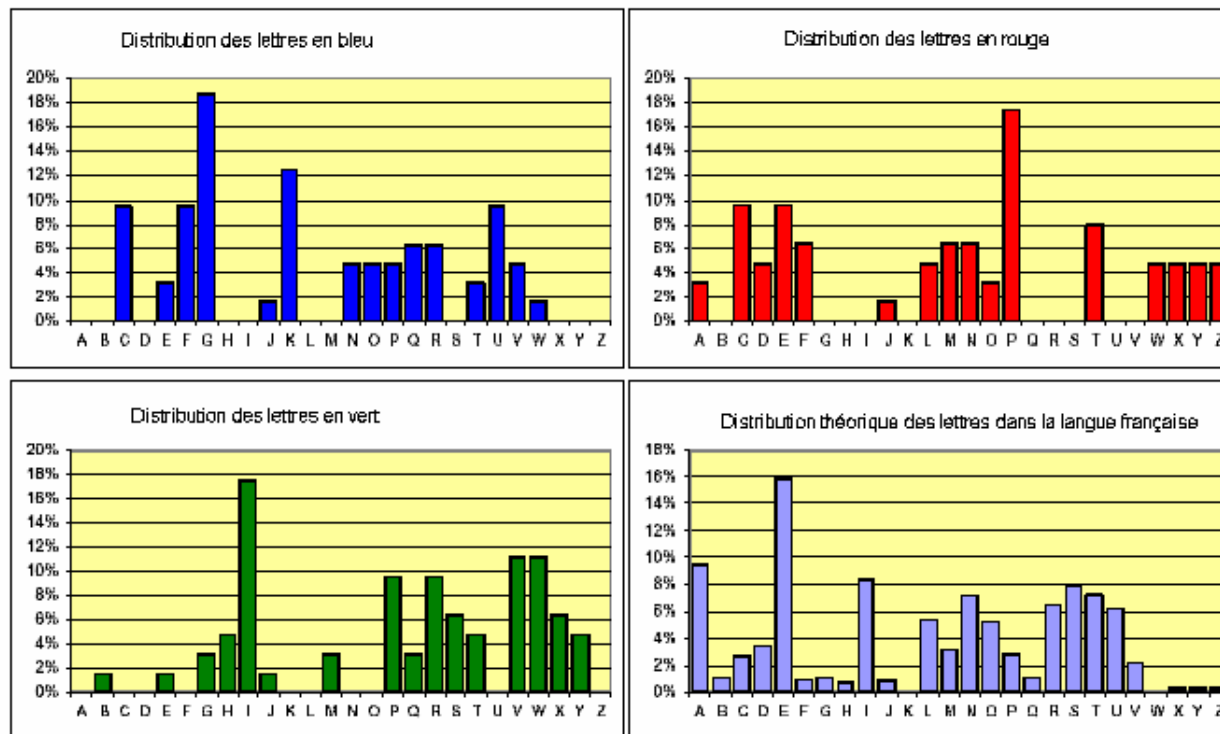
#### ◆ Faits :

- D'après l'acétate précédente, on a : La technique utilisée pour crypter les lettres  $3 * i$  (en rouge),  $3 * i + 1$  (en bleu) et  $3 * i + 2$  (en vert) est le chiffrement par décalage.
- Par hypothèse on a : Le texte original est écrit en français.

◆ **Déduction :** On déduit que les distributions des lettres  $3 * i$  (en rouge),  $3 * i + 1$  (en bleu) et  $3 * i + 2$  (en vert) ne sont un décalage de la distribution théorique de la langue française.

## Cryptanalyse du chiffrement de Vigenere

⇒ Cryptanalyse (suite) :



## Cryptanalyse du chiffrement de Vigenere

### ⇒ Cryptanalyse (suite) :

- La lettre G dans le texte  $3 * i$  (en bleu) était probablement E ⇒ Décalage de 2.
- La lettre P dans le texte  $3 * i + 1$  (en rouge) était probablement E ⇒ Décalage de 11.
- La lettre I dans le texte  $3 * i + 2$  (en vert) était probablement E ⇒ Décalage de 4.

| 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 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| C | D | E | F | G | H | I | J | K | L | M | N | O | P | Q | R | S | T | U | V | W | X | Y | Z | A | B |
| L | M | N | O | P | Q | R | S | T | U | V | W | X | Y | Z | A | B | C | D | E | F | G | H | I | J | K |
| E | F | G | H | I | J | K | L | M | N | O | P | Q | R | S | T | U | V | W | X | Y | Z | A | B | C | D |

### ⇒ Conclusion : Le texte original est :

*La libre communication des pensées et des opinions est un des droits les plus précieux de l'homme ; tout citoyen peut donc parler, écrire, imprimer librement, sauf à répondre de l'abus de cette liberté dans les cas déterminés par la loi.*

## Cryptanalyse du chiffrement de Vigenere

### ⇒ problème :

- D'une manière générale, il n'est pas évident de trouver les bons décalages surtout si on veut faire cela par un programme (absence de l'effet visuel).
- Comment donc déterminer les décalages les plus plausibles ?

### ⇒ Solution :

- Soit  $p_0, \dots, p_{25}$  sont les probabilités d'avoir les lettres  $a, \dots, z$  (respectivement) dans un texte écrit dans une langue donnée (français, anglais, etc.)
- La distribution des lettres dans cette langue peut être vue comme un vecteur  $u$  tel que :

$$u = (p_0, p_1, \dots, p_{25})$$

- Si une autre distribution  $v$  est un décalage de  $u$  de  $l$  positions alors on aura

$$v = (p_{(0-l) \bmod 26}, p_{(1-l) \bmod 26}, \dots, p_{(25-l) \bmod 26})$$

## Cryptanalyse du chiffrement de Vigenere

⇒ Solution (suite) :

• Le produit scalaire de  $u$  et  $v$  est :

$$\begin{aligned} u \times v &= p_0 p_{(0-l) \bmod 26} + \dots + p_{25} p_{(25-l) \bmod 26} \\ &= \sum_{h=0}^{25} p_h p_{(h-l) \bmod 26} \end{aligned}$$

• le produit scalaire  $u \times v$  prend sa valeur maximale lorsque  $u$  et  $v$  sont parallèles :

$$u \times v = ||u|| * ||v|| \cos(u, v) = ||u||^2 \cos(u, v)$$

avec  $||u||^2 = u \times u$

• Autrement dit, le maximum aura lieu lorsque  $u \approx v$  et c'est ce qu'on cherche.

## Cryptanalyse du chiffrement de Vigenere

⇒ Solution (suite) :

➤ Résumé :

- Soient :
  - $u = (u_0, \dots, u_{25})$  la distribution du langage source.
  - $v = (v_0, \dots, v_{25})$  une distribution décalée de  $u$ .
- Pour trouver le décalage de  $v$  par rapport à  $u$ , on calcule  $l$  tel que :

$$u \times v^l = \max_{0 \leq i \leq 25} (u \times v^i)$$

avec  $v^i = (v_{(i+0) \bmod 25}, v_{(i+1) \bmod 25}, \dots, v_{(i+25) \bmod 25})$



## Cryptanalyse du chiffrement de Vigenere

### ⇒ Remarque :

• Soient :

- $x = x_1, \dots, x_n$  et  $y = y_1, \dots, y_n$  deux chaînes de caractères.
- $u$  est la distribution de  $x$  et  $v$  de  $y$ .

• Alors, le produit  $u \times v$  est également la probabilité que deux caractères pris au hasard, l'un de  $x$  et l'autre de  $y$ , soient identiques.

• Cette probabilité est appelée indice de coïncidence mutuel de  $x$  et  $y$  et on la note généralement par :

$$MI_c(x, y)$$

## Cryptanalyse du chiffrement de Vigenere

⇒ **problème** : Jusqu'à présent, on a supposé que la longueur de la clé est connue. Le problème est maintenant de savoir comment la trouver.

⇒ **Solutions** : Deux techniques en sont disponibles :

❖ **Le test de Kasiski** : L'idée est la suivante :

- Supposons qu'on a une chaîne de caractère  $x$  qu'on veut crypter avec le chiffrement de Vigenere.
- Supposons que  $x$  contient des sous-chaînes que se répètent (e.g.  $x = \dots \underbrace{abc}_{d} \dots abc \dots$ ).
- Question : Quand est-ce que le cryptage de cette même sous-chaîne ( $abc$ ) donnera la même sous-chaîne dans le texte crypté ?
- Réponse : Une condition suffisante est que les deux sous-chaînes identiques soient séparées d'une distance  $d$  telle que  $d = 0 \bmod m$  avec  $m$  est la longueur de la clé.
- Inversement : Si on trouve dans un texte crypté par le chiffrement de Vigenere une même sous chaîne séparée par une distance  $d$ , alors il y a des chances qu'il existe  $k$  tel que  $d = k * m$ .

## Cryptanalyse du chiffrement de Vigenere

### ♦ Le test de Kasiski (suite) :

- Pour estimer la longueur de la clé, on procède ainsi :
  - Chercher des sous-chaines (de préférence contenant plus que deux caractères) qui apparaissent plus qu'une fois dans le texte crypté.
  - Noter les distances qui séparent les mêmes séquences.
  - Il y a une forte chance que la taille de la clé soit un diviseur du plus grand commun diviseur de toutes ces distances.

## Cryptanalyse du chiffrement de Vigenere

### ➤ Le test de Kasiski (suite) :

- **Exemple** : Il y a une forte chance que la longueur de la clé utilisée pour crypter le message suivant est 3.

NLPKMVGNSOXYPTGCEMQY**HGD**TGYWGPWGE**HGD**SRTRK  
 ZRUP**PW**VFR**FPW**FCSKEWNP**PW**RWYUAVGNMGFB**FP**PJZQOP  
 XQFXETXQJIPAIWEHQYGR LVNPVGNVKCIKXTTTQGCP  
 KMGX IPEWCFJCCIRZRFCIFPPCMYUOIEPXVPPKMIT  
 EIFLRUWIUNEUOIVPVOTRGDTCCPCWSK

En effet :

| Séquence   | distance | facteurs premiers |
|------------|----------|-------------------|
| <b>HGD</b> | 12       | 2,3               |
| <b>PW</b>  | 6 & 9    | 2,3 & 3           |
| <b>FP</b>  | 24       | 2,3               |

## Cryptanalyse du chiffrement de Vigenere

➤ **Le test de Friedman :** L'idée est la suivante :

- Soit  $x = x_1 \dots x_n$  le texte à déchiffré.
- Si la longueur de la clé est  $m$  alors, les sous-chaînes  $Y^i$  suivantes ont été cryptées par une simple technique de décalage :

$$Y_{1 \leq i \leq m}^i = x_i, x_{m+i}, x_{2m+i}, \dots$$

- Autrement dit, les sous chaînes  $Y^i$  ont des distributions qui a des caractéristiques semblables à celle de la langue d'origine.
- Par conséquent, les indices de coïncidence des chaînes  $Y^i$  est proche de celui de la langue utilisée pour écrire le texte original.

## Cryptanalyse du chiffrement de Vigenere

### ➤ Le test de Friedman (suite) :

- Comment faire ?
  - Supposer que  $m = 1, 2, 3, \dots$
  - Pour chaque valeur de  $m$ , calculer les indices de coïncidence des  $Y^i$ .
  - Retenir la valeur de  $m$  qui donne des indices de coïncidence proches de celui de la langue d'origine.

## Cryptanalyse du chiffrement de Vigenere

### ➤ Le test de Friedman (suite) :

#### – Exemple

- Le texte clair correspondant au texte chiffré suivant a été écrit en français.

NLPKMVGNSOXYPTGCEMQYHGDTGYWGPWGEHGDSRTRK  
 ZRUPWVFRFPWFCSKEWNPWRWYUAVGNMGFBFPPJZQOP  
 XQFXETXQJIPAIWEHQYGR LVNPNVGNVKCIKXTTTQGCP  
 KMGXIP EWCFJCCIRZRFCIFPPCMYUOIEPXVPPKMIT  
 EIFLRUWIUNEUOIVPVOTRGDTCCPCWSK

- L'indice de coïncidence de la langue française 0.078

| m | Ic(x) |       |       |       |       |
|---|-------|-------|-------|-------|-------|
| 1 | 0.051 |       |       |       |       |
| 2 | 0.053 | 0.056 |       |       |       |
| 3 | 0.096 | 0.082 | 0.091 |       |       |
| 4 | 0.052 | 0.069 | 0.079 | 0.070 |       |
| 5 | 0.062 | 0.056 | 0.059 | 0.076 | 0.078 |





## Plan

### ⇒ Introduction

### ⇒ Cryptanalyse par recherche exhaustive des clés :

- Cryptanalyse du chiffrement affine

### ⇒ Cryptanalyse linéaire :

- Cryptanalyse du chiffrement de Hill

### ⇒ Cryptanalyse par analyse de fréquences :

- Idée
- Cryptanalyse du chiffrement affine
- Cryptanalyse du chiffrement de Vigenere
- Mots probables

### ⇒ Vers un système cryptographique parfait

## Vers un système cryptographique parfait

- Intuitivement un système cryptographique est sécuritaire si aucun cryptanalyste n'est capable de le casser.
- Pour mieux clarifier cette définition, on a besoin de préciser les moyens dont ce cryptanalyste se dispose.
  - ◆ Capacité calculatoire :
    - Est-ce qu'on suppose que les capacités de calcul et le temps dont le cryptanalyste se dispose sont limitées (sécurité calculatoire) ?
    - Ou bien qu'on suppose qu'il se dispose d'une capacité de calcul infinie (sécurité inconditionnelle) ?
  - ◆ Types d'information :
    - Est-ce qu'on suppose qu'il se dispose seulement des textes chiffrés ? (attaque à texte chiffrer seulement)
    - Des textes chiffrés et leurs correspondant en clair ?
    - Etc.
- Dans ce qui suit, on suppose que le cryptanalyste a des capacités de calcul illimité, mais qu'il ne se dispose que des textes chiffrés.

## Vers un système cryptographique parfait

### ⇒ Confidentialité parfaite (Théorie de Shannon)

- Définition : Un système cryptographique  $(\mathcal{P}, \mathcal{C}, \mathcal{K}, \mathcal{E}, \mathcal{D})$  assure une confidentialité parfaite (inconditionnellement sécuritaire) si :

$$\forall m \in \mathcal{P}, c \in \mathcal{C} : P(m|c) = P(m)$$

- ⇒ aucun message crypté ne divulgue des informations sur correspondant en clair.
- ⇒ en d'autres termes, les messages cryptés que l'intrus ramasse ne l'aide pas à attaquer le système
- Théorème : Un système cryptographique  $(\mathcal{P}, \mathcal{C}, \mathcal{K}, \mathcal{E}, \mathcal{D})$  tel que  $|\mathcal{P}| = |\mathcal{C}| = |\mathcal{K}|$  assure une confidentialité parfaite si :
  - ⇒ Chaque clé est utilisée avec la même probabilité  $1/|\mathcal{K}|$ .
  - ⇒  $\forall x \in \mathcal{P}, y \in \mathcal{C} : \exists! k \in \mathcal{K} \mid e_k(x) = y$ .
- Conséquence : Le chiffrement de Verman assure une confidentialité parfaite ssi :
  - ⇒ La taille de la clé est identique à celle du message à encrypter.
  - ⇒ La clé est utilisée une seule fois.
  - ⇒ La clé est choisie d'une manière aléatoire.

## Vers un système cryptographique parfait

⇒ **Confidentialité parfaite** **Théorème :** One-Time-Pad ( $\mathcal{M} = \{0, 1\}^n$ ,  $\mathcal{C} = \{0, 1\}^n$ ,  $\mathcal{K} = \{0, 1\}^n$ ,  $E_k(m) = k \oplus m$ ,  $D_k(c) = k \oplus c$ ) assure une confidentialité parfaite si le générateur des clés est uniformément réparti.

⇒ **Preuve :** Montrons que  $Pr(M = m|C = c) = Pr(M = m)$   
 $Pr(M = m|C = c) = \frac{Pr(C=c|M=m) \times Pr(M=m)}{Pr(C=c)}$  (d'après le Théorème de Bayes)

$$\begin{aligned} Pr(C = c) &= \sum_{m' \in M} (Pr(C = c|M = m') \times Pr(M = m')) \text{ (théorème des probabilités totales)} \\ &= \sum_{m' \in M} (Pr(k = c \oplus m') \times Pr(M = m')) \text{ (} c = k \oplus m' \text{)} \\ &= \sum_{m' \in M} \left( \frac{1}{2}^n \times Pr(M = m') \right) \text{ (génération uniforme de clés)} \\ &= \frac{1}{2}^n \times \underbrace{\sum_{m' \in M} Pr(M = m')}_{=1} \\ &= \frac{1}{2}^n = Pr(C = c|M = m') \end{aligned}$$

$$Pr(M = m|C = c) = \frac{Pr(C=c|M=m) \times Pr(M=m)}{Pr(C=c)} = \frac{\frac{1}{2}^n \times Pr(M=m)}{\frac{1}{2}^n} = Pr(M = m)$$

## Vers un système cryptographique parfait

### ⇒ Confidentialité parfaite (suite) (Théorie de Shannon)

- ❖ **Question :** Est-il toujours le cas qu'un cryptanalyste ne peut jamais décrypter un message encrypté par un système cryptographique qui assure une confidentialité parfaite ?
- ❖ **Réponse :** Seulement lorsque le nombre de clés est important. En effet, le système cryptographique  $S = (\{m\}, \{c\}, \{k\}, \{e_k(m) = c\}, \{d_k(c) = m\})$  est parfait sauf qu'un cryptanalyste peut décrypter le message  $c$  à coup sûr.
- ❖ **Remarques :** La confidentialité parfaite impose des conditions (à propos des clés) difficiles à réaliser en pratique.
  - Avoir un système cryptographique parfait est une tâche compliqué voire même impossible.
  - Avoir un "bon" système cryptographique est une tâche raisonnable.
  - Pour ce faire, essayer de concevoir un système qui assure une bonne **diffusion** et bonne **confusion**.

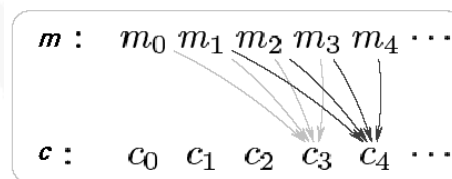
## Vers un système cryptographique parfait

⇒ **Diffusion** (Théorie de Shannon) Répartir la redondance de  $m$  sur  $c$  : Le message crypté  $c$  ne doit révéler aucune caractéristique statistique du message  $m$ .

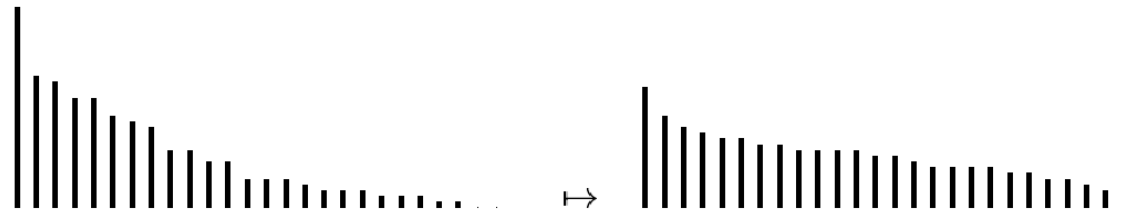
$$E_k(\text{distribution de } m) = (\text{distribution uniforme})$$

Pour ce faire :

- intégrer des permutations dans le système.
- chaque bit de  $c$  doit dépendre de plusieurs bits de  $m$ . Exemple  $c_i = \sum_{k=0}^3 m_{i-k}$



Cette technique contribue à l'aplatissement de la distribution de  $m$



## Vers un système cryptographique parfait

⇒ **Confusion** (Théorie de Shannon) Rendre la relation entre  $m$ ,  $c$  et  $k$  aussi complexe que possible.

Pour ce faire :

- Utiliser des fonctions non linéaires : si  $e_k(m) = c$  avec  $m = m_1 \dots, m_n$ ,  $c = c_1 \dots c_n$  et  $k = k_1 \dots k_n$ , alors il existe  $n$  fonctions  $f_1, \dots, f_n$  telle que :

$$\begin{cases} c_1 &= f_1(m_1, \dots, m_n, k_1, \dots, k_n) \\ &\vdots \\ c_n &= f_n(m_1, \dots, m_n, k_1, \dots, k_n) \end{cases}$$

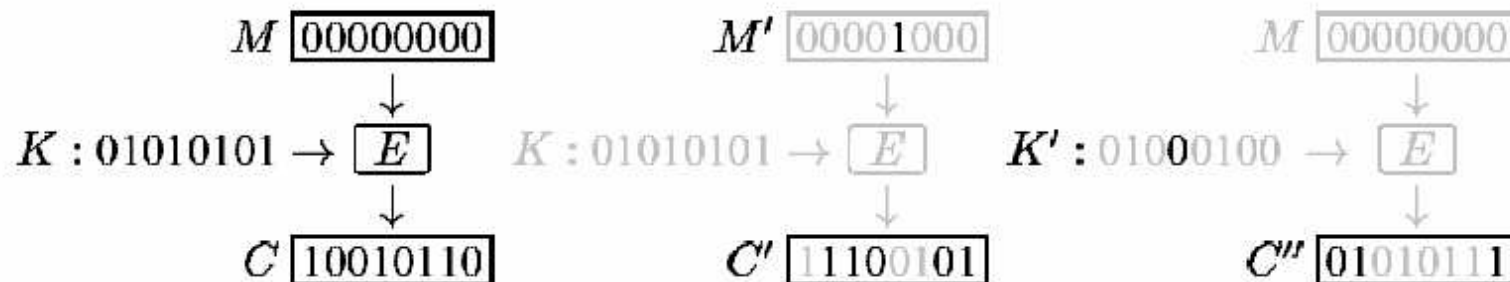
Donc, si  $e_k$  n'utilise que des fonctions linéaires alors toutes le  $f_i$ ,  $1 \leq i \leq n$ , sont linéaires et il est facile de résoudre le système ci-dessus et de trouver la clé si on se dispose de plusieurs messages claires et leurs correspondants cryptés.

- Chaque bit de  $c$  doit dépendre de plusieurs bits de  $k$ .

## Vers un système cryptographique parfait

### ⇒ Diffusion/confusion : Effet d'avalanche (Théorie de Shannon)

- Un petit changement au niveau de  $M$  ou  $K$  entraîne un changement énorme et imprévisible au niveau de  $C$ .
- En terme de bits : Le changement d'un bit au niveau de  $M$  ou  $K$  devrait entraîner le changement de la moitié des bits de  $C$ .
- Remarque :** Si tous les bits changent de valeurs alors cela n'est pas bien car le changement ne devient plus imprévisible.



Des **permutations** combinées avec des **substitutions** contribuent considérablement à la réalisation d'un "bon" système cryptographique.