Codage et cryptographie : quand $1 + 1 = 0$

Daniel Augot

INRIA Saclay-Île de France et École polytechnique

Journées Math-INRIA
Rocquencourt, juin 2010

Plan

Jeu de Marienbad

Addition modulo 2

Applications : codes binaires

Applications : stéganographie

Application : chiffrement à flot

coder, décoder

Plan

Jeu de Marienbad

Addition modulo 2

Applications : codes binaires

Applications : stéganographie

Application : chiffrement à flot

coder, décoder, chiffrer, déchiffrer

Plan

Jeu de Marienbad

Addition modulo 2

Applications : codes binaires

Applications : stéganographie

Application : chiffrement à flot

coder, décoder, chiffrer, déchiffrer, décrypter

Plan

Jeu de Marienbad

Addition modulo 2

Applications : codes binaires

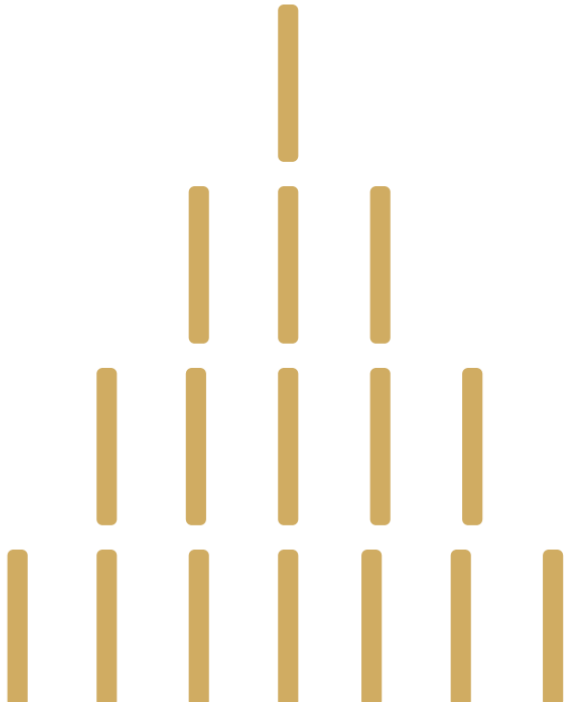
Applications : stéganographie

Application : chiffrement à flot

coder, *décoder*, *chiffrer*, *déchiffrer*, *décrypter* mais pas crypter !

Plan

Le jeu de Marienbad



Règle

- À tour de rôle, chacun retire des allumettes dans un même tas.
- Version misère : celui qui retire la dernière allumette a perdu.
- Version normale : celui qui retire la dernière allumette a gagné.

Codage de l'état stratégique d'un jeu

1. On écrit les tailles des tas en base 2 ;
2. On fait la somme, sans tenir compte des retenues. C'est la *somme-nim*.

Exemple

Tas	Taille	2^2	2^1	2^0
1	1	0	0	1
2	3	0	1	1
3	5	1	0	1
4	7	1	1	1
		0	0	0

somme-nim

Lemme

Si la somme-nim est nulle, quelque soit le mouvement suivant, la somme-nim devient non nulle.

Théorème (Version normale)

Si la somme-nim initiale est non nulle, le premier joueur a une stratégie gagnante.

Exemple

$$S = 3 \oplus 4 \oplus 5 = 2 \neq 0.$$

- On calcule

$$\begin{aligned} 3 \oplus S &= 1 \\ 4 \oplus S &= 6 \\ 5 \oplus S &= 7 \end{aligned}$$

  Le seul tas dont la somme a diminué est le tas 1.
- J'enlève deux allumettes dans le tas 1.

$$\begin{array}{lcl} S & = & T_1 \oplus T_2 \oplus T_3 \\ S \oplus S & = & 0 \\ & = & S \oplus (T_1 \oplus T_2 \oplus T_3) \\ & = & (T_1 \oplus S) \oplus T_2 \oplus T_3 \end{array}$$

Lemme

Il existe forcément un tas dont la taille diminue.

Pseudo-code et invariant de boucle

tant que Il reste une allumette faire
- *Au tour de la machine* // La somme de nim est non nulle

Calculer la somme de nim S du jeu
pour chaque tas i faire
$T_i \leftarrow$ la taille du tas i
si Si $T_i \oplus S < T_i$ alors
$T_i \leftarrow T_i \oplus S$
Interrompre la boucle
fin si
fin pour
- *Au tour de l'adversaire* // La somme de nim est nulle

// Quelque soit son mouvement,
// la somme de Nim deviendra non nulle
fin tant que

Plan

Construction par induction d'un corps sur $\mathbb{N}$

Définition (Mex)

Pour $I \subset \mathbb{N}$, $I \neq \mathbb{N}$ *:* $\mathsf{mex}(I) = \min(\mathbb{N} \setminus I)$.

Définition (Somme de Nim, $\oplus$)

$$i \oplus j = \mathsf{mex}\ \left(\left\{i' \oplus j;\ i' < i\right\} \cup \left\{i \oplus j'\ j' < j\right\}\right)$$

Construction par induction d'un corps sur $\mathbb{N}$

Définition (Mex)

Pour $I \subset \mathbb{N}$, $I \neq \mathbb{N}$ *:* $\mathsf{mex}(I) = \min(\mathbb{N} \setminus I)$.

Définition (Somme de Nim, $\oplus$)

$$i \oplus j = \mathsf{mex}\ \left(\left\{i' \oplus j;\ i' < i\right\} \cup \left\{i \oplus j'\ j' < j\right\}\right)$$

Définition (Produit de Nim, $\otimes$)

$$i \otimes j = \mathsf{mex}\left\{(i' \otimes j) \oplus (i \otimes j') \oplus (i' \otimes j');\ i' < i, j' < j\right\}$$

Construction par induction d'un corps sur $\mathbb{N}$

Définition (Mex)

Pour $I \subset \mathbb{N}$, $I \neq \mathbb{N}$ *:* $\mathsf{mex}(I) = \min(\mathbb{N} \setminus I)$.

Définition (Somme de Nim, $\oplus$)

$$i \oplus j = \mathsf{mex}\ \left(\left\{i' \oplus j;\ i' < i\right\} \cup \left\{i \oplus j'\ j' < j\right\}\right)$$

Définition (Produit de Nim, $\otimes$)

$$i \otimes j = \mathsf{mex}\left\{(i' \otimes j) \oplus (i \otimes j') \oplus (i' \otimes j');\ i' < i, j' < j\right\}$$

Justification :

$$(i - i') \otimes (j - j') > 0 \implies i \otimes j > (i' \otimes j) \oplus (i \otimes j') \ominus (i' \otimes j')$$

Table

$\oplus$ et $\otimes$

$$i \oplus j = \mathsf{mex}\ \left(\left\{i' \oplus j;\ i' < i\right\} \cup \left\{i \oplus j'\ j' < j\right\}\right)$$
$$i \otimes j = \mathsf{mex}\left\{(i' \otimes j) \oplus (i \otimes j') \oplus (i' \otimes j');\ i' < i, j' < j\right\}$$

Table

$\oplus$ et $\otimes$

$$i \oplus j = \mathrm{mex}\ \left(\left\{i' \oplus j;\ i' < i\right\} \cup \left\{i \oplus j'\ j' < j\right\}\right)$$
$$i \otimes j = \mathrm{mex}\left\{(i' \otimes j) \oplus (i \otimes j') \oplus (i' \otimes j');\ i' < i, j' < j\right\}$$

$\oplus$	0	1	2	3	...
0	0	1	2	3	
1	1	0	3	2	
2	2	3	0	1	
3	3	2	1	0	
$\vdots$					

Table

$\oplus$ et $\otimes$

$$i \oplus j = \mathrm{mex}\ \left(\left\{i' \oplus j;\ i' < i\right\} \cup \left\{i \oplus j'\ j' < j\right\}\right)$$
$$i \otimes j = \mathrm{mex}\left\{(i' \otimes j) \oplus (i \otimes j') \oplus (i' \otimes j');\ i' < i, j' < j\right\}$$

$\oplus$	0	1	2	3	...
0	0	1	2	3	
1	1	0	3	2	
2	2	3	0	1	
3	3	2	1	0	
$\vdots$					

$\otimes$	0	1	2	3	...
0	0	0	0	0	
1	0	1	2	3	
2	0	2	3	1	
3	0	3	1	2	
$\vdots$					

Résultat

Théorème

On obtient un corps infini de caractéristique 2.

Théorème

Les sous corps finis sont les $\mathbb{F}_{2^{2^n}}$.

Théorème

Tout polynôme de degré 2^{2^n} *à coefficients dans ce corps y est scindé.*

Résultat

Théorème

On obtient un corps infini de caractéristique 2.

Théorème

Les sous corps finis sont les $\mathbb{F}_{2^{2^n}}$ *.*

Théorème

Tout polynôme de degré 2^{2^n} *à coefficients dans ce corps y est scindé.*

- Ce corps est la $\ll$ clôture quadratique de $\mathbb{F}_2$ $\gg$.

Résultat

Théorème

On obtient un corps infini de caractéristique 2.

Théorème

Les sous corps finis sont les $\mathbb{F}_{2^{2^n}}$.

Théorème

Tout polynôme de degré 2^{2^n} *à coefficients dans ce corps y est scindé.*

- Ce corps est la « clôture quadratique de $\mathbb{F}_2$ ».
- Définition « canonique » de $\mathbb{F}_{2^{2^n}}$.

Résultat

Théorème

On obtient un corps infini de caractéristique 2.

Théorème

Les sous corps finis sont les $\mathbb{F}_{2^{2^n}}$.

Théorème

Tout polynôme de degré 2^{2^n} *à coefficients dans ce corps y est scindé.*

- Ce corps est la « clôture quadratique de $\mathbb{F}_2$ ».
- Définition « canonique » de $\mathbb{F}_{2^{2^n}}$.
- Par exemple $\mathbb{F}_{256}$ (les octets).

Résultat

Théorème
On obtient un corps infini de caractéristique 2.

Théorème
Les sous corps finis sont les $\mathbb{F}_{2^{2^n}}$.

Théorème
Tout polynôme de degré 2^{2^n} à coefficients dans ce corps y est scindé.

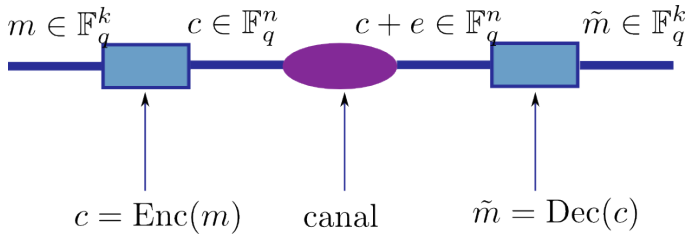
- Ce corps est la « clôture quadratique de $\mathbb{F}_2$ ».
- Définition « canonique » de $\mathbb{F}_{2^{2^n}}$.
- Par exemple $\mathbb{F}_{256}$ (les octets).

John H. Conway. *On Numbers and Games*. 2nd edition, 2000

H.W. Lenstra Jr. Nim multiplication. Technical report, Institut des Hautes Études Scientifiques, 1978

Plan

Codage dans un canal de transmission



Définition

Le code correspondant à la fonction d'encodage Enc est l'ensemble des mots codés, i.e.

$$C = \{c = Enc(m); \quad m \in \mathbb{F}_q^k\}$$

Objectifs

- On veut que $R = \frac{k}{n} \in [0, 1]$ soit grand.
- On veut que la probabilité $\Pr(\tilde{m} \neq m)$ soit petite.
- On veut que l'algorithme de décodage Dec soit rapide.

Deux extrêmes triviaux, code de parité, code de répétition

Code à parité

Encodage Enc

$$(x_1, \ldots, x_n) \mapsto (x_1, \ldots, x_n, x_{n+1})$$

avec $x_{n+1} = \bigoplus_{i=1}^{n} x_i$.

Deux extrêmes triviaux, code de parité, code de répétition

Code à parité

Encodage Enc

$$(x_1, \ldots, x_n) \mapsto (x_1, \ldots, x_n, x_{n+1})$$

avec $x_{n+1} = \bigoplus_{i=1}^{n} x_i$.

- redondance minimale, i.e. taux de transmission maximal ;

Deux extrêmes triviaux, code de parité, code de répétition

Code à parité

Encodage Enc

$$(x_1, \ldots, x_n) \mapsto (x_1, \ldots, x_n, x_{n+1})$$

avec $x_{n+1} = \bigoplus_{i=1}^{n} x_i$.

- redondance minimale, i.e. taux de transmission maximal ;
- détecte une erreur, pas de correction.

Deux extrêmes triviaux, code de parité, code de répétition

Code à parité

Encodage Enc

$$(x_1, \ldots, x_n) \mapsto (x_1, \ldots, x_n, x_{n+1})$$

avec $x_{n+1} = \bigoplus_{i=1}^{n} x_i$.

- redondance minimale, i.e. taux de transmission maximal ;
- détecte une erreur, pas de correction.

Code de répétition

Encodage Enc

$$(x_1) \mapsto \underbrace{(x_1, \ldots, x_1)}_{n \text{ fois}}$$

Deux extrêmes triviaux, code de parité, code de répétition

Code à parité

Encodage Enc

$$(x_1, \ldots, x_n) \mapsto (x_1, \ldots, x_n, x_{n+1})$$

avec $x_{n+1} = \bigoplus_{i=1}^{n} x_i$.

- redondance minimale, i.e. taux de transmission maximal ;
- détecte une erreur, pas de correction.

Code de répétition

Encodage Enc

$$(x_1) \mapsto \underbrace{(x_1, \ldots, x_1)}_{n \text{ fois}}$$

- redondance maximale, i.e. taux de transmission minimal.

Deux extrêmes triviaux, code de parité, code de répétition

Code à parité

Encodage Enc

$$(x_1, \ldots, x_n) \mapsto (x_1, \ldots, x_n, x_{n+1})$$

avec $x_{n+1} = \bigoplus_{i=1}^{n} x_i$.

- redondance minimale, i.e. taux de transmission maximal ;
- détecte une erreur, pas de correction.

Code de répétition

Encodage Enc

$$(x_1) \mapsto \underbrace{(x_1, \ldots, x_1)}_{n \text{ fois}}$$

- redondance maximale, i.e. taux de transmission minimal.
- taux de correction maximal (vote majoritaire).

Distance de Hamming

Définition (Distance de Hamming)

La distance de Hamming *entre* x *et* y *est*

$$d(x,y) = \left|\{i;\quad x_i \neq y_i\}\right|.$$

Le poids de Hamming *de* x *est* $d(x,0)$.

Définition

Soit $C \subset \mathbb{F}_2^n$, *la* distance minimale *de* C *est*

$$d(C) = \min_{x,y \in C, x \neq y} d(x,y).$$

Si le code est $\mathbb{F}_2$*-linéaire*[1], *alors*

$$d(C) = \min_{x \neq 0 \in C} w(x).$$

1. $x, y \in C \implies x \oplus y \in C$

Hamming 1950

- longueur 7, trois bits de redondance ;
- trois *équations de « parité »* :

$$\begin{aligned} x_5 &= x_1 \oplus x_2 \oplus x_4 \\ x_6 &= x_1 \oplus x_3 \oplus x_4 \\ x_7 &= x_2 \oplus x_3 \oplus x_4 \end{aligned}$$

- *matrice de contrôle, de parité* :

$$\begin{bmatrix} 1 & 1 & 0 & 1 & 1 & 0 & 0 \\ 1 & 0 & 1 & 1 & 0 & 1 & 0 \\ 0 & 1 & 1 & 1 & 0 & 0 & 1 \end{bmatrix}$$

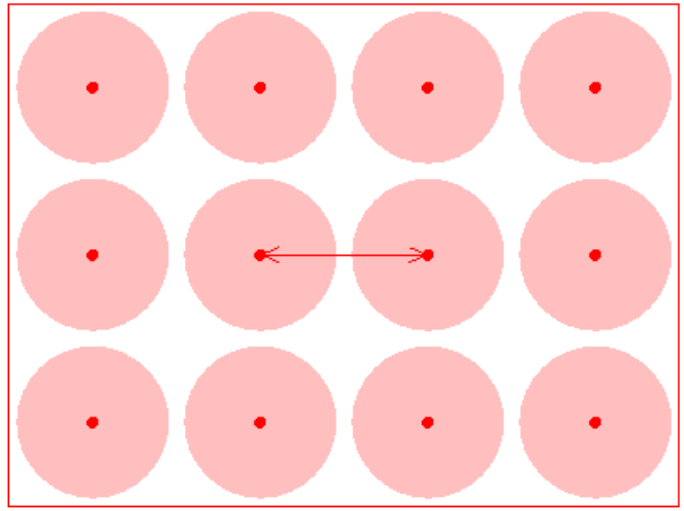
et

$$c \in C \iff H \cdot c^T = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix} ;$$

- les colonnes sont deux à deux distantes $\Longrightarrow$ pas de mots de poids 2 dans le code ; le poids minimal est trois.

Distance minimale et capacité de correction

- Si un code a pour distance minimale d, alors les boules de rayon $t = \lfloor \frac{d-1}{2} \rfloor$ sont disjointes.



- On note alors $t = \lfloor \frac{d-1}{2} \rfloor$, c'est la *capacité de correction du code*.

Décodage par syndrome

- Soit c le mot de code, et y le mot reçu, on écrit

$$y = c \oplus e;$$

où e est l'erreur. Elle correspond aux bits inversés.

- *Décoder revient à retrouver à trouver l'erreur*.
- Si H est la matrice de parité du code, alors

$$Hy^T = Hc^T \oplus He^T = He^T.$$

- On dit que $S = Hy^T$ est le *syndrome de l'erreur* e.
- *Si le poids de e est inférieur à t, le syndrome S caractérise e*.

Décodage

- Le code de Hamming est de distance minimale 3 ;
- il est donc 1-correcteur ;

Principe :

- pour une erreur e de poids 1, par exemple

$$e = (0, 0, 0, 1, 0, 0, 0)$$

$$\begin{bmatrix} 1 & 1 & 0 & 1 & 1 & 0 & 0 \\ 1 & 0 & 1 & 1 & 0 & 1 & 0 \\ 0 & 1 & 1 & 1 & 0 & 0 & 1 \end{bmatrix} e^T = \begin{bmatrix} 1 \\ 1 \\ 1 \end{bmatrix}$$

- le ≪ syndrome ≫ est la colonne correspondante de la matrice de contrôle.

Algorithme :

calculer $S = Hy^T$;
identifier S comme une colonne de la matrice de parité ;
l'indice de cette colonne donne la position du bit à inverser.
// *Algorithme propre au code de Hamming*

Généralisation

Une infinité de codes de Hamming :

- Pour m donné, on liste tous les entiers non nuls entre 1 et $2^m - 1$;
- on met en colonne les représentations binaires de ces entiers ;
- c'est la matrice de contrôle du code de Hamming de longueur $2^m - 1$;
- on obtient un code de distance minimale trois ;
- les symboles de parité correspondent aux entiers de poids 1 (puissance de 2).

Note

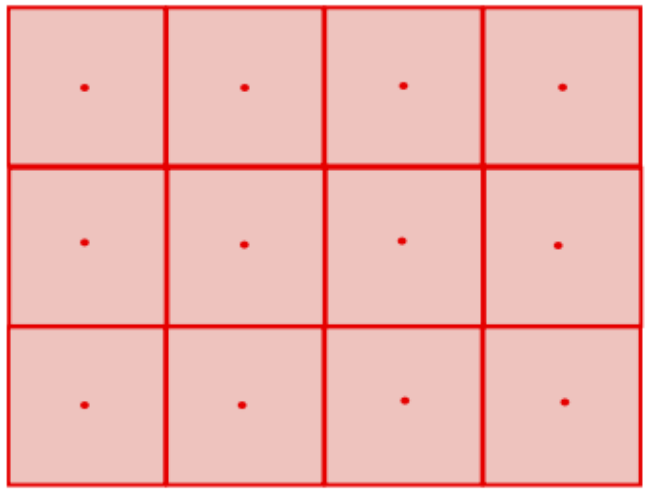
La redondance relative vérifie

$$\lim_{n \to \infty} \frac{m}{2^m} = 0.$$

Le code de Hamming est « parfait »

$$\{\text{Les boules de rayon } 1\} \times \{\text{les mots de code}\} = \text{tout l'espace}$$

$$(1 + (2^m - 1))\, 2^{2^m - 1 - m} = 2^{2^m - 1}$$



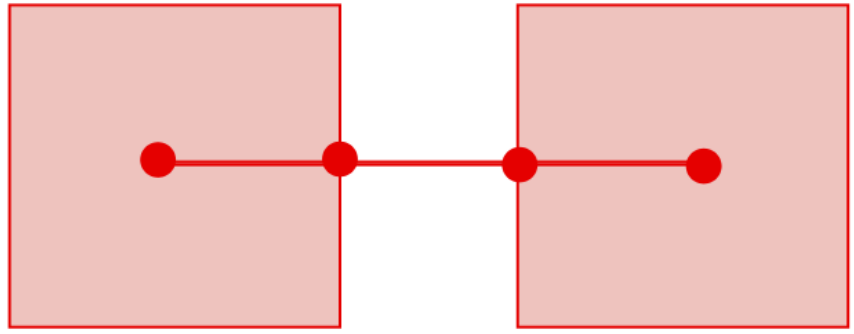
La distance de Hamming est non intuitive !

NVidia Fermi (2009)

- NVidia propose ses cartes pour du calcul scientifique. Dans ce contexte, les erreurs sont cruciales.
- La carte Fermi implémente des codes *SECDED* ≪ Single-Error Correct Double-Error Detect ≫.
- Construction à base des codes de Hamming :
  1. pn considère un code de Hamming de longueur $127 = 2^7 - 1$, de dimension $127 - 7$, distance minimale 3 ;
  2. on l'étend en ajoutant un ≪ bit de parité ≫ : code de longueur 128, distance minimale 4 ; redondance 8 bits.
  3. on le raccourcit pour obtenir un code de longueur 72, de dimension $72 - 8$, de distance minimale 4.
  4. même redondance que 8 codes de parité (8 bits), mais correction d'erreur.
- Sont protégés :
  1. les registres ;
  2. la mémoire partagée ;
  3. les caches L1 et L2.

Pourquoi distance 4 ?

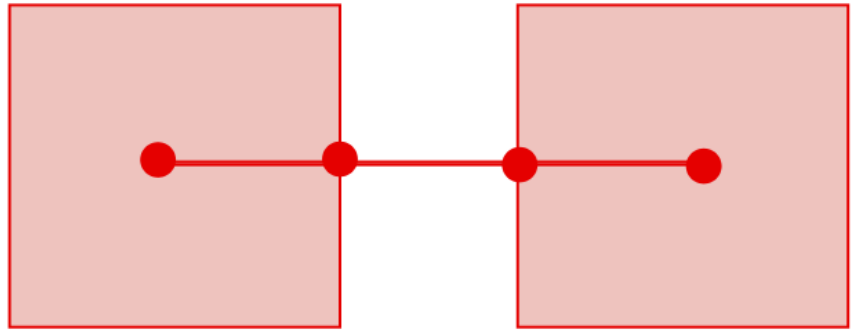
- Hamming standard ($d = 3$) :



Une erreur de poids 2 est mal corrigée.

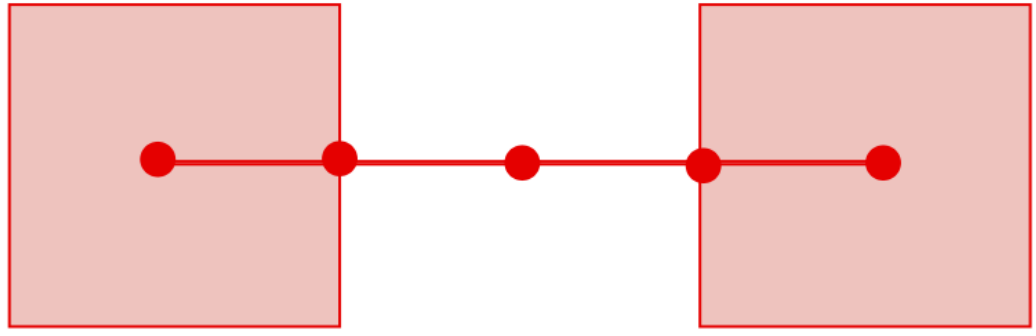
Pourquoi distance 4 ?

- Hamming standard ($d = 3$) :



Une erreur de poids 2 est mal corrigée.

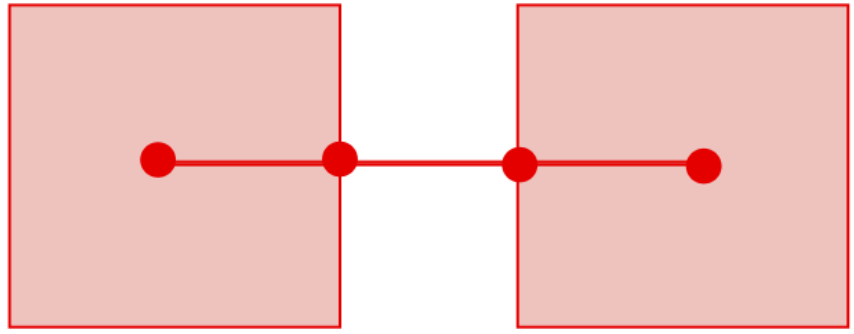
- Hamming étendu ($d = 4$) :



Une erreur de poids 2 n'est pas corrigée et est détectée !

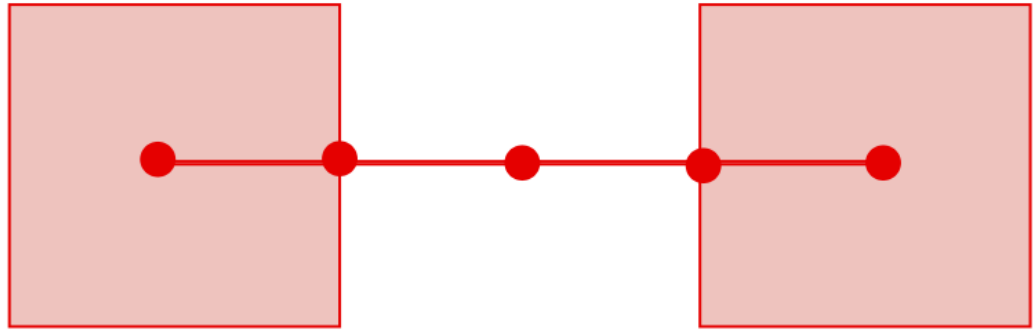
Pourquoi distance 4 ?

- Hamming standard ($d = 3$) :



Une erreur de poids 2 est mal corrigée.

- Hamming étendu ($d = 4$) :



Une erreur de poids 2 n'est pas corrigée et est détectée !

Seulement en ajoutant un bit de parité !

Plan

Stéganographie (F5)

- But : cacher un secret s dans une $\ll$ couverture $\gg$ x (image, vidéo, son).
- La plupart des médias perceptifs tolèrent des modifications mineures dans leur représentation binaire, sans que ce soit perceptible.
- Exemple naïf : les bits les moins significatifs d'un représentation binaire d'une image, par octets par exemple (format `ppm`, en `RGB`).
- Exemple plus sophistiqué : les bits de poids faible de la discrétisation de la transformée en cosinus de l'image (`JPEG`).
- Algorithme célèbre : `F5`.

Modélisation discrète

- passage d'un modèle « signal », ou « perceptif », à un modèle « discret » ;
- la *couverture* est $(x_1, \dots x_n) \in \mathbb{F}_2^n$;
- le *secret* est $(s_1, \dots, s_r) \in \mathbb{F}_2^r$; et $r \ll n$;
- On cherche un schéma *plongement-extraction* (E, R)[2] :

$$E \ : \ \mathbb{F}_2^r \times \mathbb{F}_2^n \ \to \ \mathbb{F}_2^n \ \text{ et } \ R \ : \ \mathbb{F}_2^n \ \to \ \mathbb{F}_2^r$$

tel que

$$R(E(s, x)) = s \quad \text{correction de la méthode}$$

et

$$\rho = \max(d(x, E(s, x))$$

soit minimal (furtivité).

2. Embedding-Recovery

Stéganographie (F5)

Soit H la matrice de contrôle d'un code $[n, n-r]$;

- *Codage par syndrome* : soit $s \in \mathbb{F}_2^r$ le message à cacher, et $x \in \mathbb{F}_2^n$ la couverture ;
- on cherche e de poids minimal tel que

$$He^T = s \oplus Hx^T;$$

  C'est un problème de décodage par syndrome, où $s \oplus Hx^T$ est le syndrome.
- alors $\hat{x} = x \oplus e$ vérifie

$$H\hat{x}^T = Hx^T \oplus He^T = Hx^T \oplus s \oplus Hx^T = s.$$

Schéma

- La fonction de plongement est $E : (s, x) \mapsto \hat{x} = x \oplus e$, où e est le décodage de $s \oplus Hx^T$.
- La fonction d'extraction est donc $R : \hat{x} \mapsto H\hat{x}^T$.

Stéganographie (F5 et code de Hamming)

- Le code de Hamming $[n = 2^r - 1, 2^r - 1 - r, 3]_2$ est *parfait* :
- Pour tout secret $s \in \mathbb{F}_2^r$, il existe un mot de poids 1 tel que

$$He^T = s.$$

- Donc si la couverture est de longueur $n = 2^r - 1$, on peut y plonger r bits, en modifiant 1 bit de la couverture.

Stéganographie (F5 et code de Hamming)

- Le code de Hamming $[n = 2^r - 1, 2^r - 1 - r, 3]_2$ est *parfait* :
- Pour tout secret $s \in \mathbb{F}_2^r$, il existe un mot de poids 1 tel que

$$He^T = s.$$

- Donc si la couverture est de longueur $n = 2^r - 1$, on peut y plonger r bits, en modifiant 1 bit de la couverture.
- Logique ! Un mot de poids 1 de longueur $2^r - 1$ peut être vu comme un entier $i \in [0, 2^r - 1]$, donc comme r bits.

Stéganographie (F5 et code de Hamming)

- Le code de Hamming $[n = 2^r - 1, 2^r - 1 - r, 3]_2$ est *parfait* :
- Pour tout secret $s \in \mathbb{F}_2^r$, il existe un mot de poids 1 tel que

$$He^T = s.$$

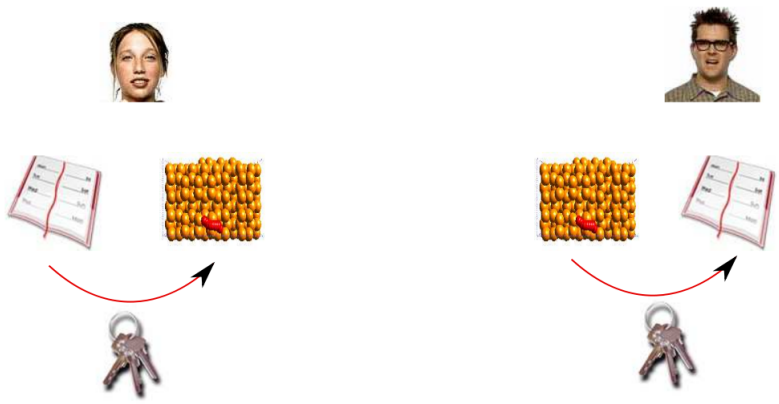
- Donc si la couverture est de longueur $n = 2^r - 1$, on peut y plonger r bits, en modifiant 1 bit de la couverture.
- Logique ! Un mot de poids 1 de longueur $2^r - 1$ peut être vu comme un entier $i \in [0, 2^r - 1]$, donc comme r bits.

Exemple

On peut cacher 10 bits dans $2^{10} = 1024$ bits en ne modifiant qu'un bit de ceux-ci.

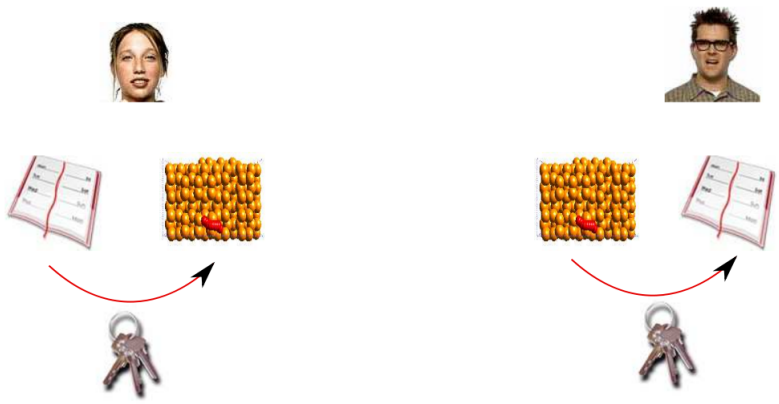
Plan

Chiffrement symétrique



La clé K est supposée constante suffisamment longtemps.

Chiffrement symétrique



La clé K est supposée constante suffisamment longtemps.

On veut donc un algorithme de chiffrement :

$$\begin{array}{rcccc} f_C & : & \mathbb{F}_2^k \times \mathbb{F}_2^n & \to & \mathbb{F}_2^n \\ & & (K, M) & \mapsto & C = f_c(K, M) \end{array}$$

et un algorithme de déchiffrement

$$\begin{array}{rcccc} f_D & : & \mathbb{F}_2^k \times \mathbb{F}_2^n & \to & \mathbb{F}_2^n \\ & & (K, M) & \mapsto & M = f_D(K, C) \end{array}$$

Chiffrement de César, Masque jetable, chiffre de Vernam, etc

- La clé est $K = (K_1, \ldots, K_n) \in \mathbb{F}_2^n$;
- Le message est $M = (M_1, \ldots, M_n) \in \mathbb{F}_2^n$;
- Le chiffré est $C = (C_1, \ldots, C_n) \in \mathbb{F}_2^n$, avec

$$C_i = M_i \oplus K_i, \quad i \in \{1, \ldots, n\}$$

- Pour déchiffrer, on fait

$$C_i \oplus K_i = M_i, \quad i \in \{1, \ldots, n\}$$

Théorème

Si la clé est aléatoire, c'est-à-dire si :

$$p(K_i = 1) = \frac{1}{2},$$

alors la connaissance de C ne donne aucune information sur M.

Chiffrement de César, Masque jetable, chiffre de Vernam

Théorème

Si la clé est aléatoire, c'est-à-dire si :

$$p(K_i = 1) = \frac{1}{2},$$

alors la connaissance de C ne donne aucune information (!) *sur M.*

Démonstration.

Supposons $C_i = 0$, alors $p(M_i = 0) = p(K_i = 0) = \frac{1}{2}$. □

Chiffrement de César, Masque jetable, chiffre de Vernam

Théorème

Si la clé est aléatoire, c'est-à-dire si :

$$p(K_i = 1) = \frac{1}{2},$$

alors la connaissance de C ne donne aucune information (!) *sur M.*

Démonstration.

Supposons $C_i = 0$, alors $p(M_i = 0) = p(K_i = 0) = \frac{1}{2}$. □

À condition que la clé soit aussi longue que le message.

Chiffrement de César, Masque jetable, chiffre de Vernam

Théorème

Si la clé est aléatoire, c'est-à-dire si :

$$p(K_i = 1) = \frac{1}{2},$$

alors la connaissance de C ne donne aucune information (!) *sur M.*

Démonstration.

Supposons $C_i = 0$, alors $p(M_i = 0) = p(K_i = 0) = \frac{1}{2}$. □

À condition que la clé soit aussi longue que le message.

Par exemple, si on chiffre deux messages $M^{(1)}$ et $M^{(2)}$ avec la même clé K, alors, pour $i \in \{1, \ldots, n\}$:

1. $C_i^{(1)} = M_i^{(1)} \oplus K_i$
2. $C_i^{(2)} = M_i^{(2)} \oplus K_i$
3. $\Longrightarrow C_i^{(1)} \oplus C_i^{(2)} = M_i^{(1)} \oplus M_i^{(2)}$

On obtient de l'information sur les deux messages clairs.

Générateur pseudo-aléatoire

Il est donné par :

- une « graîne » courte, $s_0 \in \mathbb{F}_2^m$;
- une fonction de mise à jour : $U : \mathbb{F}_{2^m} \to \mathbb{F}_{2^m}$;
- une fonction d'extraction : $E : \mathbb{F}_{2^m} \to \mathbb{F}_2$.

Pour chaque bit M_i à chiffrer :

1. extraire un bit $K_i = E(s_i)$, s_i étant l'état à l'étape i ;
2. chiffrer : $C_i = M_i \oplus K_i$;
3. mettre à jour l'état : $s_{i+1} = U(s_i)$.

Le générateur pseudo-aléatoire doit

1. passer un certains nombres de tests statistiques (NIST) ;
2. être difficile à inverser (impossible de retrouver s_0) ;
3. impossible à prédire, même après observation.

Registre à décalage à rétroaction linéaire

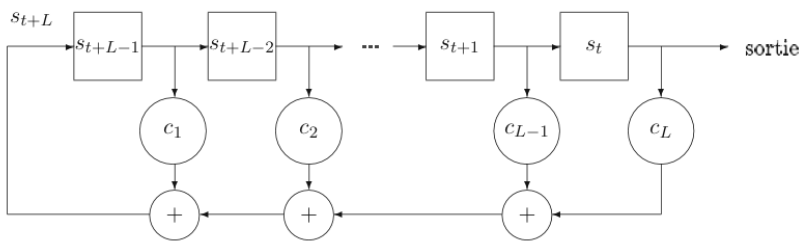
- l'état initial est $S_1, \ldots S_L$;
- on fixe des constantes $(c_1, \ldots, c_L) \in \mathbb{F}_2^L$

$$S_{t+L} = c_1 S_{t+L-1} \oplus \cdots \oplus c_L S_t; \quad t \geq L+1$$

- fonction de mise à jour $U(S_{t+L-1}, \ldots, S_t)$:

$$\begin{bmatrix} S_{t+L} \\ \vdots \\ S_{t+1} \end{bmatrix} = \begin{bmatrix} c_1 & & & c_L \\ 1 & 0 & & 0 \\ 0 & 1 & \ddots & 0 \\ & \ddots & \ddots & 0 \\ & & 0 & 1 \end{bmatrix} \begin{bmatrix} S_{t+L-1} \\ \vdots \\ S_t \end{bmatrix}$$

- extraction : $E(S_{t+L-1}, \ldots, S_t) = S_t$;
- implantation matérielle très simple :



m-séquences

Pour une relation de récurrence bien choisie, pour un registre à m bits :

1. La suite est de période maximale $2^m - 1$;
2. si on fait glisser une fenêtre de m bits le long de la séquence, on voit tous les entiers de $\{1, \ldots, 2^m - 1\}$, *une fois et une seule*.
3. très bonnes propriétés statistiques ;

m-séquences

Pour une relation de récurrence bien choisie, pour un registre à m bits :

1. La suite est de période maximale $2^m - 1$;
2. si on fait glisser une fenêtre de m bits le long de la séquence, on voit tous les entiers de $\{1, \ldots, 2^m - 1\}$, *une fois et une seule*.
3. très bonnes propriétés statistiques ;

Mais facile à prédire :

Proposition

L'observation de $2t$ bits de la séquence produite permet de retrouver la rétroaction et l'initialisation*.*

Conclusion

- Une définition naturelle pour la somme de Nim des entiers ;
- $1 \oplus 1 = 0$ est fondamental en informatique, parce que c'est simple !
- force de la simplicité du code de Hamming et son décodage ;
- application en stéganographie ;
- chiffrement à flot : niveau recherche.