Code

1. Introduction

- A Alphabet fini
- A^* Ensemble des mots sur A
- w:
    - le mot vide ϵ
    - $w = a_1 ... a_n$ avec $a_1 ... a_n \in \mathrm{A}$
- $\mid w \mid = m$ longueur de w
- $\mid \epsilon \mid = 0$
- $u \in \mathrm{A}^*, v \in \mathrm{A}^*$
    - $u = a_1 ... a_l$ et $v = b_1 ... b_k$
    - $u.v = a_1 ... a_l . b_1 ... b_k$ **concatenation**
- on as $u.\epsilon = u = \epsilon.u$
- $u, v, t \in \mathrm{A}^*$
    - $(u.v).t = u.(v.t)$
- $\mathrm{A}^* = \mathrm{A}^* \setminus \{\epsilon\}$

1.1. Code

$\mathrm{A} = \{a, b, c, d, e\}$

$\mathrm{B} = \{0, 1\}$

Codage : $\Phi : \mathrm{A} \rightarrow \mathrm{B}^+$
$\Phi : a \rightarrow 00$
$\Phi : b \rightarrow 01$
$\Phi : c \rightarrow 10$
$\Phi : d \rightarrow 110$
$\Phi : e \rightarrow 111$

$\Phi(abca) = \Phi(a)\Phi(b)\Phi(c)\Phi(a)$

Remarque

On a :

- $\Phi(u, v) = \Phi(u).\Phi(v)$
- $\Phi(\epsilon) = \epsilon$

$\Phi(abca) = 00011000$

1.2. Question

Φ^{-1} Bien definie ?

$\Phi^{-1}(00011000) = abca$ est bien definie (**injective**)

1.3. Definition :

Φ est une fonction de codage **si** Φ est **injective**

($\Phi : \mathrm{A} \to \mathrm{B}^+$ etendu à A)

Soit $\mathrm{G} = \Phi(\mathrm{A})$ (ensemble de mot):

$\mathrm{C} = \{00, 01, 10, 110, 111\}$ G est un code **si et seulement si** tout mot de C^+ a une décomposition unique en mot de G.

G est un code **si et seulement si** pour tout mot de $u \in \mathrm{G}^+$:

- si $u = u_1 ... u_m$ avec $u_1 ... u_m \in \mathrm{G}$
- et $u = u'_1 ... u'_k$ avec $u'_1 ... u'_k \in \mathrm{G}$
- alors $m = k$ et $u_1 = u'_1$ ... $u_k = u'_k$

$Soit \mathrm{G} = \{00, 01, 10, 110, 111\}$ est un code.

Supposons que $u_1 ... u_k = u'_1 ... u'_m$

$u_1 ... u_k \in \mathrm{G}$
$u'_1 ... u'_m \in \mathrm{G}$

En plus avec k minimal

$u_1 ... u_k = u'_1 ... u'_m$ si $u_1 = 00$ alors $u'_1 = 00$ $u_2 ... u_k = u'_2 ... u'_m$

Contredit minimalité de k.

2. Ensemble préfix

E est un préfix:

- Si $u \in \mathrm{E}$ et $u.v \in \mathrm{E}$
- Alors $v = \epsilon$

Aucun mot de E n'est le préfix d'un autre mot de E

2.1. Proposition

Si E est un préfix alors E est un code.

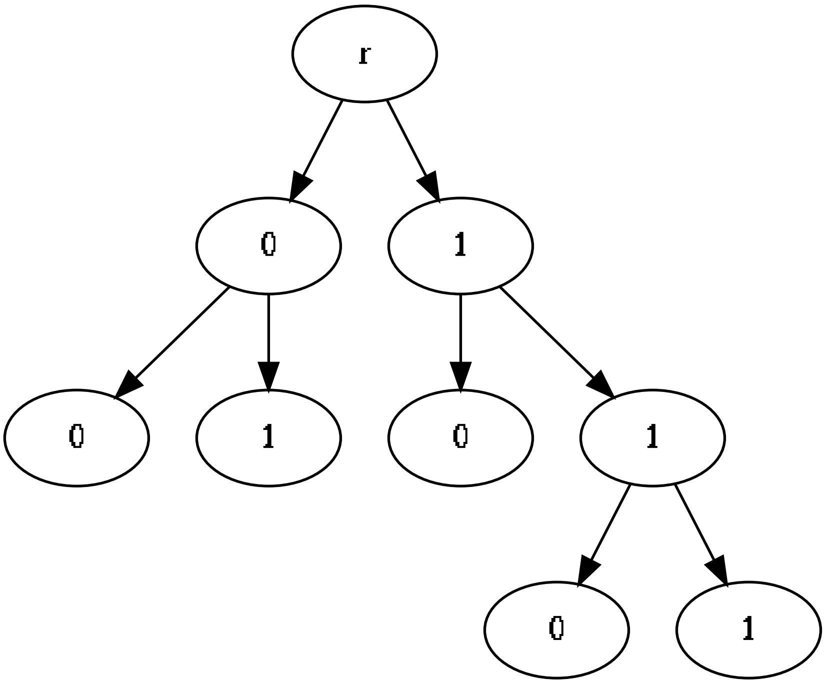
- $x \in \mathrm{E}, u \in \mathrm{E}^*$
- $y \in \mathrm{E}, v \in \mathrm{E}^*$

$x.u = y.v$ alors $x = y$

2.2. Exercice :

Trouver un code qui n'est ni préfix ni suffix.

Préfix: sur $\{0, 1\}$



Φ : Codage de taille fixe

$\forall a, a' \in \mathrm{A} \mid \Phi(a) \mid = \mid \Phi(a') \mid$

G code $\forall w, w' \in textG \mid w \mid = \mid w' \mid$

Code correcteurs d'erreur

1. Généralité

On considère uniquement des codes binaires à longueur fixe

$\{0, 1\}$ la longueur l

m un mot : $< u_1,, u_e > ui \in \{0,1\}$

vecteur sur $\langle \mathbb{Z}/2\mathbb{Z} \rangle^l$

$\langle \mathbb{Z}/2\mathbb{Z} \rangle$

$$\begin{cases} 0+0 & = & 0 \\ 0+1 & = & 1 & = & 1+0 \\ 1+1 & = & 0 \end{cases}$$

espace vectoriel $\langle \mathbb{Z}/2\mathbb{Z} \rangle^l$

$< u_1, ..., u_l > + < u'_1, ..., u'_l > = < u_1 + u'_1, ..., u_l + u'_l >$

$< 00010 > + < 11010 > = < 11000 >$

$< u > + < v > = \vec{0}$
$0. < u > = \vec{0}$
$1. < u > = < u >$

2. Distance de Hamming

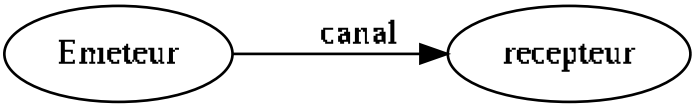
$a, b \in \langle \mathbb{Z}/2\mathbb{Z} \rangle$

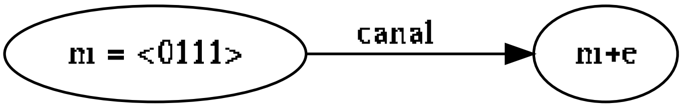
$d(a,b) = 0$ si $a = b$
$d(a,b) = 1$ sinon
$d(< u_1, ..., u_l >, < u'_1, ..., u'_l > = < u_1 + u'_1, ..., u_l + u'_l >) = \sum_{i=1}^{l} d(u_i + u'_i)$

$u = 00001110$
$v = 01011010$
$d(u,v) = 3$

2.1. Remarque :

- d est une distance
- $d(u,u) = 0$
- $d(u,v) = d(v,u)$
- $d(u,u) \leq d(u,x) + d(x,v)$
- $d(u+x, v+x) = d(u,v)$





ou e est l'erreur

Le canal peur inverser 1 bit ou plus
$e = \langle 0,0,0,0 \rangle$

$e = \langle 1,0,0,0 \rangle$
$e = \langle 0,1,0,0 \rangle$
$e = \langle 0,0,1,0 \rangle$
$e = \langle 0,0,0,1 \rangle$

G un ensemble (de vecteur) $d(\mathrm{G}) = \min \mathrm{d}(x,y)\ x,y \in \mathrm{G} x \neq y$

$\mathrm{C} =< 0001, 1100, 1001 > d(\mathrm{C}) = 1$

Corriger des erreur ?

recu $m' = m + e$

à partir de m' (avec des info sur e)
Trouver m

Exemple Checksum : Détecte une erreur

$x =< u_1, ..., u_l > m =< u_1, ..., u_l, \sum u_i >$

$l = 4\ x = 1010 \Rightarrow m =< 10100 >$

$< 0,0,0,0,0 >$
$< 1,0,0,0,0 >$
$< 0,1,0,0,0 >$
$< 0,0,1,0,0 >$
$< 0,0,0,1,0 >$
$< 0,0,0,0,1 >$

Impossible que ce soit le massage d'origine

m' reçu tel que $d(m,m') \leq 1$

Verification : les seuls messages sans erreur sont $< u_1, ..., u_l, u_{l+1} >$ tel que $\sum\limits_{i=1}^{l} u_i = u_{l+1}$

$\sum\limits_{i=1}^{l} u_i = u_{l+1} \Leftrightarrow \sum\limits_{i=1}^{l} u_i - u_{l+1} = 0 \Leftrightarrow \sum\limits_{i=1}^{l} u_i + u_{l+1} = 0$

Avec au plus une erreur si reçu m' $m' =< m'_1, ..., m'_l$ * si m' (message reçu) est tel que $\sum\limits_{i=1}^{l+1} m'_i = 0$ Alors c'est que m qui a été emis

$$\begin{pmatrix} a_1^1 & \cdots & a_l^1 & a_{l+1}^1 \\ a_1^2 & \cdots & a_l^2 & a_{l+1}^2 \\ \vdots & \vdots & \ddots & \vdots \\ a_1^k & \cdots & a_l^k & a_{l+1}^k \\ a_1^{k+1} & \cdots & a_l^{k+1} & a_{l+1}^{k+1} \end{pmatrix}$$

$u_{l+1}^j = \sum\limits_{i=1}^{l} u_i$ parité par ligne $u_j^{k+1} = \sum\limits_{i=1}^{k} u_j$ parité par colonne

$$\begin{array}{cccc|c} 0 & 0 & 1 & 1 & 0 \\ 0 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 0 & 1 \\ \hline 1 & 0 & 1 & 0 & 0 \end{array}$$

$0011.0111.1110 \longrightarrow 00110.01111.11101.10100$