Codes correcteurs d'erreurs

- I. Modélisation, notations, distance, propriétés
  - Exemples simples de codes de parité
  - Code de Hamming
- II. Codes linéaires
  - Codage et décodage simple et efficace
- III. Codes cycliques et Reed-Solomon
  - Construction de codes de taux de correction arbitraire
- IV. Exemples d'application des codes correcteurs

Codes équivalents, étendus raccourcis

- C : code (n,k,d) sur V
- Def 1: code **équivalent** : C' obtenu par
  - Permutation de positions dans tous les mots de C
  - Permutation de symboles de V dans tous les mots de C

  C' a même distance et rendement que C
- Def 2: code **étendu** : C' obtenu par ajout d'un chiffre de parité:

  $C' = \{c_1 \dots c_n\ c_{n+1}$ tq $c_1 \dots c_n \in C$ et $c_1 + \dots + c_{n+1} = 0\}$
- Def 3: code **poinçonné** : C' obtenu en supprimant une position:

  $C' = \text{poinçonné}(C,i) = \{c_1 \dots c_{i-1}\ c_{i+1} \dots c_n$ tq $c_1 \dots c_n \in C\}$

  si on poinçonne m positions: $d' \geq d-m$
- Def 4 : code **raccourci** : soit $s \in V$ et $1 \leq i \leq n$ fixés:

  $C' = \text{raccourci}(C,i,s) = \{c_1 \dots c_{i-1}\ c_{i+1} \dots c_n$ tq $c_1 \dots c_{i-1}\ s\ c_{i+1} \dots c_n \in C\}$

  on a $d' \geq d$

Code de Hamming binaire

- Code binaire parfait 1-correcteur
- Si il y a 0 ou 1 erreur de bits => n+1 possibilités
- Au moins $r = \log_2 (n+1)$ bits pour coder une erreur possible
- Code de Hamming: code (2^r-1, k=2^r-r-1, d=3)

  $c_1 \ldots c_n$ défini par :

  - Si $i \neq 2^j$ alors c_i est un bit de source
  - Si $i = 2^j$ alors c_i est un bit de contrôle de parité = somme des c_k tel que k écrit en binaire a un 1 en position j
  - Exemple code de Hamming (7,4)

- Détection: si au moins un des bits de parité est erroné
- Correction: on change le bit d'indice la somme des indices des bits de parité erronés

II. Codes linéaires

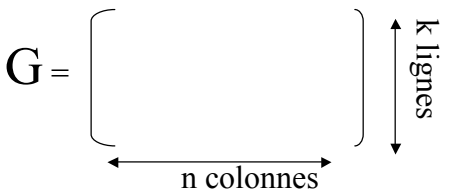
- Hypothèse sur V = vocabulaire source : *corps*
- Code linéaire - caractérisation
- Codage et décodage d'un code linéaire

Hypothèse sur le vocabulaire V du canal

- V est muni d'une structure de **corps fini** :
  soient a,b ∈ V : e_0, e_1, a+b, -a, a*b, a/b ∈ V
- Possible ssi **|V| = p^m** avec **p** premier
  Soit q = p^m : V est alors isomorphe à GF_q
  - Exemple: V= {0,1} ; V={mots de 32 bits}
- Implantation
  - GF_q isomorphe à (Z/pZ[x])/Q(x) avec Q polynôme irréductible de degré m à coefficients dans Z/pZ
  - NB avec p= 2: facile à implanter avec des registres à décalage !

Code linéaire

- V corps => V^n est un espace vectoriel
- **Définition** : code linéaire ⇔ sev de V^n de dim k
  - Si $x=[x_0, \ldots, x_{n-1}] \in C$, $y=[y_0, \ldots, y_{n-1}] \in C$
    => $x+y =[x_0+y_0, \ldots, x_{n-1}+y_{n-1}] \in C$
- C est engendré par une **matrice génératrice** G
- C=Im(G)



Exemple 1

- V = {a=00, b=01, c=10, d=11}
- Code (6,4) engendré par G

G =

01	10	11	10	11	00
10	11	10	00	01	10
10	01	11	10	11	10
01	10	11	11	10	01

Exemple 2: code de Hamming (7,4)

- C'est aussi un code linéaire sur V = {0,1}

engendré par G =

1	1	1	0	0	0	0
1	0	0	1	1	0	0
0	1	0	1	0	1	0
1	1	0	1	0	0	1

- Remarque: il est équivalent aux codes de matrices génératrices G'=L.G.P avec L inversible et P permutation. Par exemple:

G'=

1	0	0	0	1	1	0
0	1	0	0	1	0	1
0	0	1	0	0	1	1
0	0	0	1	1	1	1

G''=

1	0	1	0	1	0	0
0	1	0	0	1	0	1
0	0	1	0	0	1	1
0	0	0	1	1	1	1

Distance d'un code linéaire

- $d = \text{Min} \{ d_H(x, y) ; \forall x,y \in C \}$
  $= \text{Min} \{ w_H(x - y) ; \forall x,y \in C \}$
  $= \text{Min} \{ w_H(z) ; \forall z \in C \}$

- Borne de Singleton : $d \leq n-k+1 = r+1$

- On peut donc corriger :
  - jusqu'à (d-1)/2 erreurs quelconques
  - Jusqu'à (d-1) erreurs localisées (effacements)

- Code MDS:
  - distance maximale: atteint la borne de Singleton: d = r+1

Décodage d'un code linéaire

- Un code linéaire est équivalent à un code de matrice génératrice $G = [Id_k \mid A]$

- *code*($[s_0, \ldots, s_{k-1}]$) = $s.G = [c_0, \ldots, c_{k-1}, c_k \ldots, c_{n-1}]$ où:
  $[c_0 \ldots, c_{k-1}] = [s_0, \ldots, s_{k-1}]$
  et $[c_k \ldots, c_{n-1}] = [s_0, \ldots, s_{k-1}].A = [c_0, \ldots, c_{k-1}].A$
  Intérêt: coût de codage : calcul de c=s.G => O(k.r) opérations

- De plus: $[c_k \ldots, c_{n-1}].Id_{n-k} - [c_0, \ldots, c_{k-1}].A = [0]_{n-k}$
- Soit (en transposant) $[-A^t \mid Id_{n-k}]$. $c^t = 0$, i.e. **H** $.c^t = 0$

  On a aussi: $H.G^t = 0$

  H = Matrice de contrôle de dimension (n-k, n)
- Rem: d = nbre minimal de cols indép. de H

Détection d'erreurs

- On émet x et on reçoit y;
  y est un mot de code $\Leftrightarrow H.y^t = 0$
- **Syndrome** d'erreur : $\mathbf{s = H .y^t}$
  $s \neq 0$ => il y eu erreur de transmission
- Exemple: matrice de contrôle du code de Hamming.

Correction d'erreurs

- $x = [x_1 .. x_n]$ émis ; $y = [y_1 .. y_n]$ reçu
- Corriger y $\Leftrightarrow$trouver x tq x=y- e appartient au code avec e = vecteur de correction (=erreur) de poids minimum
- Or: $s = H .y^t = H.x^t + H.e^t = H.e^t$
  Le syndrome donne toute l'information pour corriger :
  e = vecteur d'erreur de poids minimal tel que $H.e^t = s$
- Deux méthodes possibles :
  - Localisation des erreurs puis résolution du système linéaire $H.e^t = s$
  - Par tabulation de la correction e associé à chaque syndrome possible
    - on stocke dans un tableau Cor[s] := e pour chaque syndrome s
    - Ex: V= {0,1}, n=64 k=52 :
      On peut recevoir un mot parmi 2^{64} mots possibles, dont $2^{64} - 2^{52}$ mots erronés
      mais seulement 2^{12}=4096 syndromes possibles !
      donc table des corrections de taille 4096

Bons codes

- Facile et efficace à implémenter
  - Codage et décodage (détection/correction) peu coûteux
  - logiciel et/ou matériel
  - Exemple: codes linéaires cycliques
- Etant donné un taux de correction « t/n » donné, pouvoir facilement construire un code (n,k,d) qui permet ce taux de correction
  - Exemple: codes de Reed-Solomon