

TD 3: Codes correcteurs

10/05/06 6

1. Codes cycliques

1.1 $(a_0 \dots a_{n-1})$ - Notation polynomiale

$$p(x) = a_0 x^0 + a_1 x^1 + a_2 x^2 + a_3 x^3 + \dots + a_{n-1} x^{n-1}$$

$$g(x) = x^3 + x + 1$$

S. $p(x)$ n'est pas multiple de $g(x)$, il y a une erreur.
Il faut coder pour que $p(x) = g(x)q(x)$

Base canonique :

$$0001 \rightarrow m_1(x) = 1 \xrightarrow{x^0 n(x)} x^3$$

$$0010 \rightarrow m_2(x) = x \xrightarrow{x^1 n(x)} x^4$$

$$0100 \rightarrow m_3(x) = x^2 \xrightarrow{x^2 n(x)} x^5$$

$$1000 \rightarrow m_4(x) = x^3 \xrightarrow{x^3 n(x)} x^6$$

$$g(x) = x^3 + x + 1 \Rightarrow \text{degré } 3 \Rightarrow R = 3$$

$$p(x) = x^R m(x) + r(x)$$

Pour calculer $r(x)$, on va faire la division polynomiale $x^R n(x) \div g(x)$

$n(x)$

Division longue

$$\begin{array}{r} x^3 \\ -x^3 + x + 1 \\ \hline x + 1 \end{array} \quad \begin{array}{r} x^3 + x + 1 \\ \hline 1 \end{array}$$

$$P_1(x) = x^3 + x + 1$$

$$\begin{array}{r} x^4 + x^2 + x \\ -x^4 \\ \hline x^2 + x \\ \hline x \end{array} \quad \begin{array}{r} x^4 + x^2 + x \\ -x^4 \\ \hline x^2 + x + 1 \\ \hline x \end{array}$$

$$P_2(x) = x^4 + x^2 + x$$

$$\begin{array}{r} x^5 + x^3 + x^2 \\ -x^5 \\ \hline x^3 + x^2 \\ \hline x^3 + x + 1 \\ \hline x^2 + x + 1 \end{array} \quad \begin{array}{r} x^5 + x^3 + x^2 \\ -x^5 \\ \hline x^3 + x^2 \\ \hline x^3 + x + 1 \\ \hline x^2 + x + 1 \end{array}$$

$$P_3(x) = x^5 + x^2 + x + 1$$

$$\begin{array}{r} x^6 + x^4 + x^3 \\ -x^6 \\ \hline x^4 + x^3 \\ \hline x^4 + x^2 + x \\ \hline x^3 + x^2 + x \\ \hline x^3 + x + 1 \\ \hline x^2 + x + 1 \end{array} \quad \begin{array}{r} x^6 + x^4 + x^3 \\ -x^6 \\ \hline x^4 + x^3 \\ \hline x^4 + x^2 + x \\ \hline x^3 + x^2 + x \\ \hline x^3 + x + 1 \\ \hline x^2 + x + 1 \end{array}$$

$$P_4(x) = x^6 + x^2 + x + 1$$

						x^6	x^5	x^4	x^3	x^2	x^1	x^0
	b_1	0	0	0	1	→	0	0	0	1	0	1
→	b_2	0	0	1	0	→	0	0	1	0	1	0
	b_3	0	1	0	0	→	0	1	0	0	1	1
→	b_4	1	0	0	0	→	1	0	0	0	1	0
							Message				Code	

Cette base canonique va nous permettre de retrouver les messages sur 4 bits.

Ex: 1010

(on additionne un 0 à la fin)

$$\begin{array}{r}
 1010 \quad 1000101 \\
 \oplus \quad 0010110 \\
 \hline
 1010011
 \end{array}$$

1.2 On divise le polynôme qu'on a à l'arrivée par $g(x)$.
 $p'(x) = p(x) + x^i$ - x^i est l'erreur
 $p'(x)$ est-il divisible par $g(x)$?
 $p'(x) = g(x) q'(x) + r'(x)$
 $= x^k m(x) + r(x) + x^i$
 $= g(x) q(x) + x^i$

Si $i < k$ - degré de $g(x)$

$$r'(x) = x^i$$

L'erreur se trouve sur les bits de contrôle
message valide

Si $i \geq k$

$$p'(x) = g(x) q(x) + x^i$$

on divise par $g(x)$

$$\begin{aligned}
 p'(x) &= g(x) q(x) + g(x) q_i(x) + r_i(x) \\
 &= g(x) q'(x) + r_i(x)
 \end{aligned}$$

$$r_i(x) = r(x)$$

Le reste donne la valeur $r_i(x)$ qui est le reste de la division de x^i par $g(x)$.
On compare ensuite la valeur du reste et de voir à quel élément il est associé de la base canonique.

Après le test d'un mot reçu, on divise ce mot par $g(x)$. Il y a 3 cas : ① le reste est nul : Pas d'erreur.

② le reste a 1 seul bit : l'erreur est sur le champ de contrôle : message valide

③ le reste a plus qu'un bit : l'erreur est dans le message. Le reste associé à un élément de la base canonique pointe l'erreur.

1.3 $111\ 00\ 11 \Rightarrow x^6 + x^5 + x^4 + x + 1$

Division longue

$$\begin{array}{r}
 x^6 + x^5 + x^4 + x + 1 \\
 \underline{x^6 + x^5 + x^3} \\
 x^5 + x^3 + x + 1 \\
 \underline{x^5 + x^3 + x^2} \\
 x^2 + x + 1
 \end{array}
 \quad
 \begin{array}{r}
 x^3 + x + 1 \\
 \underline{x^3 + x^2} \\
 x^2 + x + 1
 \end{array}$$

$$x^2 + x + 1 \Rightarrow 1\ 1\ 1$$

$\Rightarrow$ Erreur sur x^5 . Voir base canonique

$\Rightarrow$ Vrai message : $101\ 00\ 11$

$$1010111 \Rightarrow x^6 + x^4 + x^2 + x^1 + x^0$$

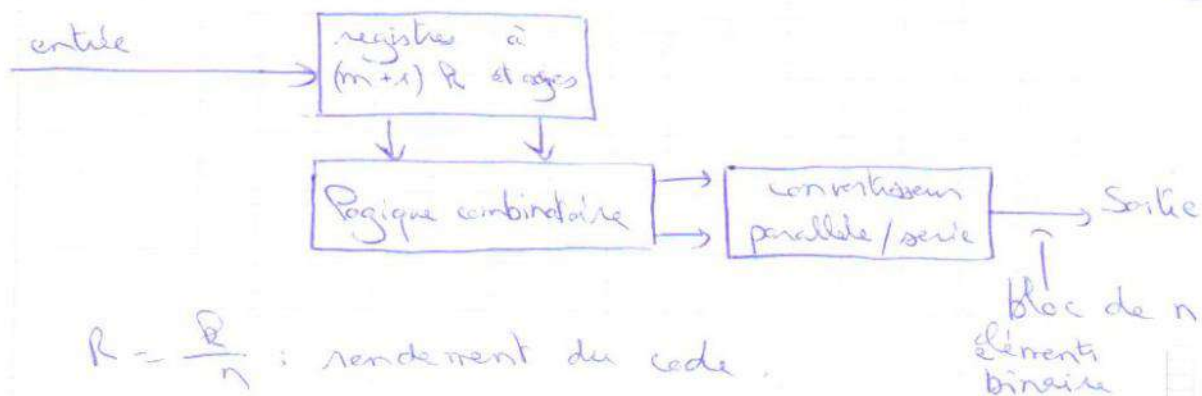
$$\begin{array}{r|l} x^6 + x^4 + x^2 + x + 1 & x^3 + x + 1 \\ \hline x^6 + x^4 + x^3 & x^3 + 1 \\ \hline x^3 + x^2 + x + 1 & \\ x^3 + x + 1 & \\ \hline x^2 & \end{array}$$

Erreur sur les bits de contrôle.
Message valide

2. Codes convolutifs

Chaque bloc de n éléments binaires en sortie du codeur dépend non seulement du bloc de k éléments binaires pris à son entrée mais aussi des m blocs précédents. Effet mémoire d'ordre m .

La qte $(m+1)$ s'appelle la longueur de contrainte du code.



$$R = \frac{k}{n} : \text{rendement du code}$$

$$R = \frac{1}{2} \begin{array}{l} \leftarrow \text{entrée} \\ \leftarrow \text{sortie} \end{array} \quad \begin{array}{l} 1 \text{ entrée} \\ 2 \text{ sorties} \end{array}$$

ici $n = m$

2 sorties

contrainte: $(m+1) = 3$

17/05/06

$\Rightarrow (m+1) = 3$ étages de registre

2 polynômes générateurs

$g_1 = x^2 + x + 1$

$g_2 = x^2 + 1$

On va construire le bloc diagramme du code.

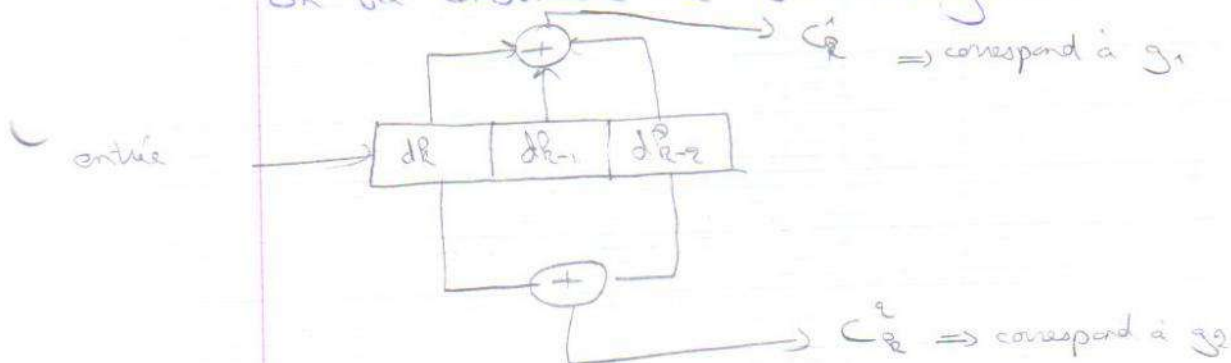
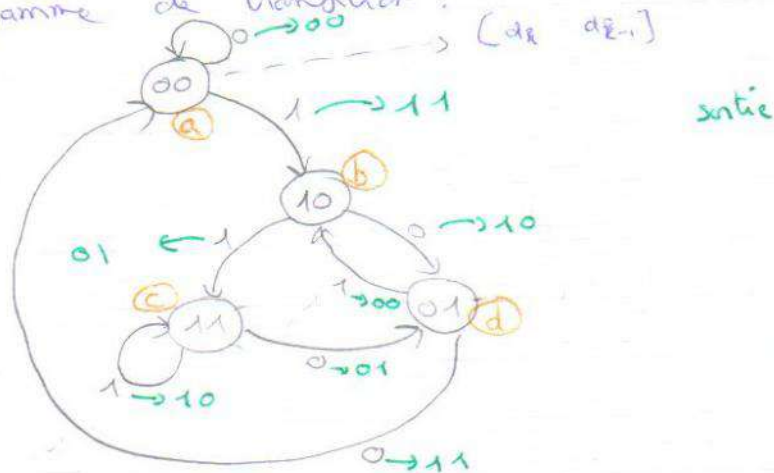


Diagramme de transition: $(dk, dk-1)$

Registre à décalage



2.2. On rajoute 00 en queue de message (que des 0)
J'appelle les états a b c et d

Je suppose au départ que je suis dans l'état a.

message:	0	1	0	1	1	1	0	0
état (dr dr-1)	a	b	d	b	c	c	d	a
codage binaire	00	11	10	00	01	10	01	11

2.3 00 ~~1~~ 1 ~~0~~ 00 ~~1~~ 1 ~~0~~ 1 ~~1~~ (10 bits)
 $\Rightarrow$ 00 1 1 0 0 1 1 0 1 1 (11 bits)

$$R = \frac{8}{11} \rightarrow \begin{array}{l} \text{nb bits du message de départ} \\ \text{nb bits du message envoyé} \end{array}$$

le rendement a augmenté

$$\frac{8}{11} > \frac{1}{2}$$

2.4 Message reçu: 00 1 1 0 1 1 1 0 1 0

1^{er} étape: remplir les positions pointillées. Ici on va les remplir par des 0 mais on peut aussi les remplir par des 1.

$\Rightarrow$ 00 0 1 1 0 0 1 0 1 1 0 0 1 0 0

2^e étape: je regroupe par bloc de 2 éléments binaires.

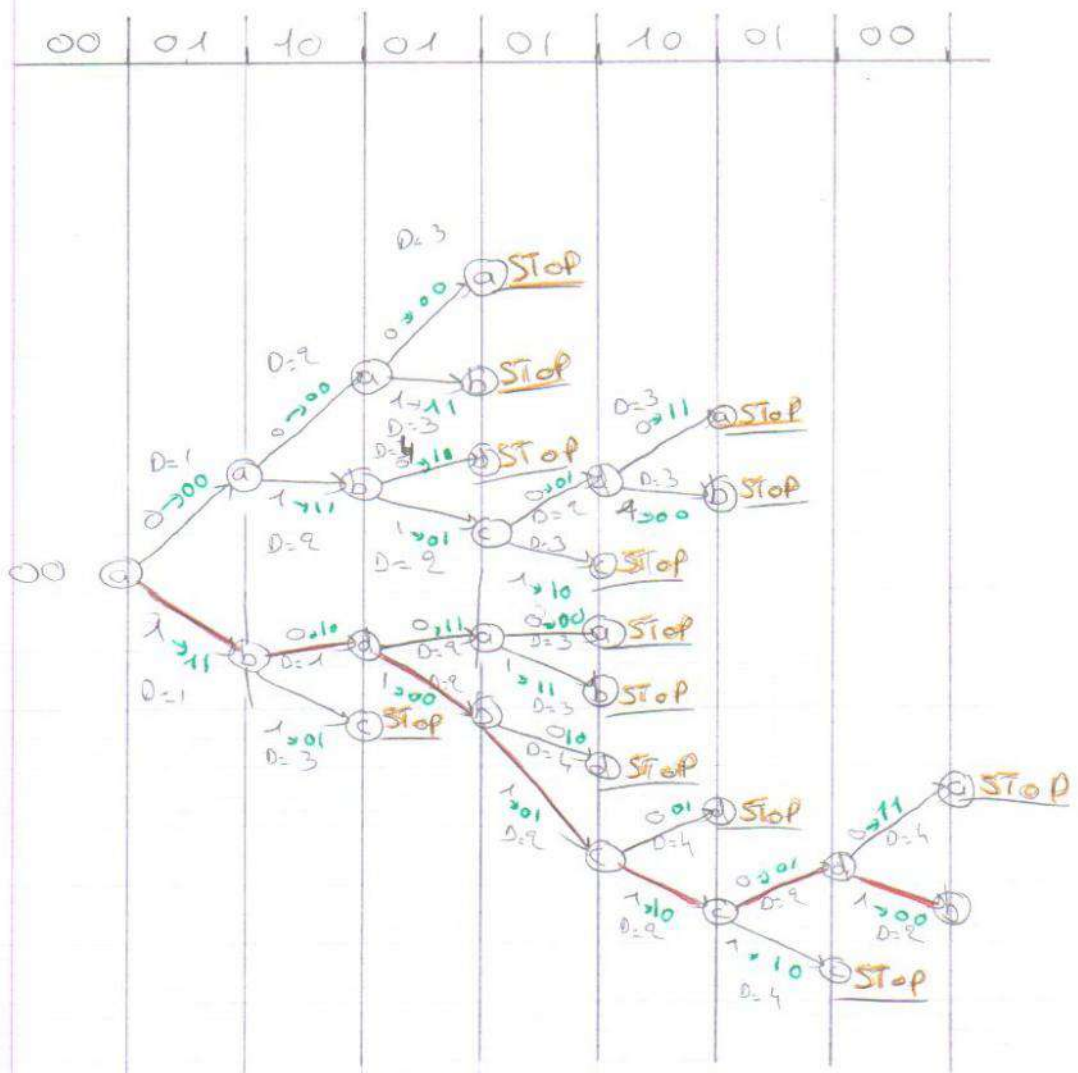
$\Rightarrow$ 00 01 10 01 01 10 01 00

3^e étape: Algo de Viterbi.

Elaboré à partir du diagramme de transition
 Choix: On s'arrête pour $D=3$

comm num

17/05/06



On remonte l'arbre :

0 1 0 1 1 1 | 0 1

message réellement envoyé : (on avait rajouté en queue

0 1 0 1 1 1