LE TRAITEMENT DES ERREURS

A) Notions d'erreurs

Les rayonnements électromagnétiques, les perturbations propres au système (distorsion, bruit) peuvent entacher d'erreurs les informations transmises (bits erronés) Une liaison est qualifiée par son ***taux d'erreurs*** appelé BERT. Le taux d'erreurs est exprimé par le rapport entre le nombre d'informations (bits) erronées et le nombre d'informations (bits) transmises.

Te = Nb d'info. (ou bits) erronées / Nb d'info. (ou bits) transmises

Par exemple soit la transmission de la suite: "0110 0100 1100 1001 0100 1010".

qui est reçue "0110 0110 1100 1011 0100 0010".

Quel est le taux d'erreurs de ce canal ?

Le message reçu diffère de 3 bits du message émis. Le nombre de bits émis est de 24 bits. Le taux d'erreurs est:

$$Te = 3 / 24 = 0,125$$

Le Te varie en pratique de 10 (Liaisons RTC) à $1O^{-9}$ (réseaux locaux). Il exprime un taux statistique, l'erreur affecte n bits et non I bit tous les x bits.

Si Te est la probabilité pour qu'un bit soit erroné, la probabilité de recevoir un bit correct est de (1 - Te), pour qu'un bloc de N bits soit reçu correctement la probabilité est de $(1 - Te)^N$.

Supposons une transaction de 100 caractères (CCITT $N^0 5$, 7 bits) émis sur une liaison en mode synchrone 4 800 bits / s avec un Te de 10^4. Les erreurs sont supposées être distribuées aléatoirement. Déterminons la probabilité pour qu'un message reçu soit correct:

Le message de 100 caractères correspond à un bloc de:

$$100 \times 7 = 700 \text{ bits}$$

La probabilité de réception d'un bloc correct (Pb) est de:

$$Pb = (1 - 0{,}0001)^{700} = (0{,}9999)^{700} = 0{,}932$$

La probabilité de recevoir un message erroné (Pe)

$$Pe = 1 - 0{,}932 = 0{,}068$$

Quatre techniques sont mises en œuvre pour détecter et corriger les erreurs

➲ **La détection par écho**:
le récepteur renvoie le message reçu, si le message est différent de celui émis, l'émetteur retransmet le message. Cette technique est peu utilisée.

➲ **La détection par répétition**:
chaque message émis est suivi de sa réplique. Si les deux messages sont différents, le récepteur demande une retransmission. Cette technique est très utilisée dans les milieux sécurisés et très perturbés.

➲ **La détection d'erreurs par code**:
une information supplémentaire au niveau du caractère (bit de parité) ou au niveau d'un groupe de caractères (clé) est ajoutée à l'information transmise. Le récepteur contrôle le bit de parité ou la clé, s'il détecte une erreur, il ignore les données reçues et en demande la retransmission.

➲ **La détection et correction d'erreurs par code**:
cette technique consiste à substituer au code des caractères à transmettre, par exemple le code ASCII, par un codage particulier qui autorise la détection et l'autocorrection d'erreurs (code auto-correcteur).

B) Détection d'erreurs par bit de parité

Dans cette technique, on introduit une information complémentaire, un bit ou un caractère, dépendant du contenu binaire du message à protéger, tel que le nombre de bits, à 1 ou à 0, à transmettre soit pair (bit de parité) ou impair (bit d'imparité).

Par exemple sur la figure suivante, pour protéger un caractère de 7 bits (code ASCII), on introduit un 8ème bit, dit bit de parité:

Caractère	O	S	I
Bit 0	1	1	1
Bit 1	0	0	0
Bit 2	0	1	0
Bit 3	1	1	1
Bit 4	1	1	0
Bit 5	1	1	0
Bit 6	1	1	1
Bit de parité	1	0	1
Bit d'imparité	0	1	0

Cette technique porte le nom de VRC (*Vertical Redundancy Check*), vérification par redondance verticale. Elle est utilisée essentiellement dans les transmissions asynchrones. Dans les transmissions synchrones, les caractères sont envoyés en blocs.

Caractère à transmettre	Bit de parité	Caractère à transmettre	Bit de parité		Caractère LRC	Bit de parité

Dans ce cas, la technique utilisée protège tous les bits d'un caractère (VRC) et tous les bits de chaque caractère de même rang (tous les 1er bit de chaque caractère, tous les 2ème bit . . .). Le caractère constitué, avec ces différents bits, est ajouté au message. Ce caractère, appelé LRC (*Longitudinal Redundancy Check)*, est lui-même protégé par un bit de parité (VRC).

Exemple:

Transmission du mot « HELLO ».

	H	**E**	**L**	**L**	**O**	**LRC→**
Bit 1	0	1	0	0	1	0
Bit 2	0	0	0	0	1	1
Bit 3	0	1	1	1	1	0
Bit 4	1	0	1	1	1	0
Bit 5	0	0	0	0	0	0
Bit 6	0	0	0	0	0	0
Bit 7	1	1	1	1	1	1
VRC ↓	0	1	1	1	1	0

0001001	0	1010001	1	0011001	1	0011001	1	1111100	1	0100001	0
H		E		L		L		O		LRC	

C) Détection par clé calculée

1) Principe

La détection d'erreurs par clé calculée s'applique aux procédures de transmission dites orientées bits (on ne transmet pas des caractères mais une suite quelconque de bits). Dans ces transmissions, les données sont groupées en blocs appelés trames (*Frame*). Une clé, constituée d'un nombre de bits prédéterminé, déduite d'une opération mathématique appliquée au bloc de données est ajoutée au message, fugure suivante. En principe, la clé se nomme *CRC (Cyclic Redundancy Check*) si elle fait 16 bits, et FCS (*Frame Check Sequence*) si elle est sur 32 bits.

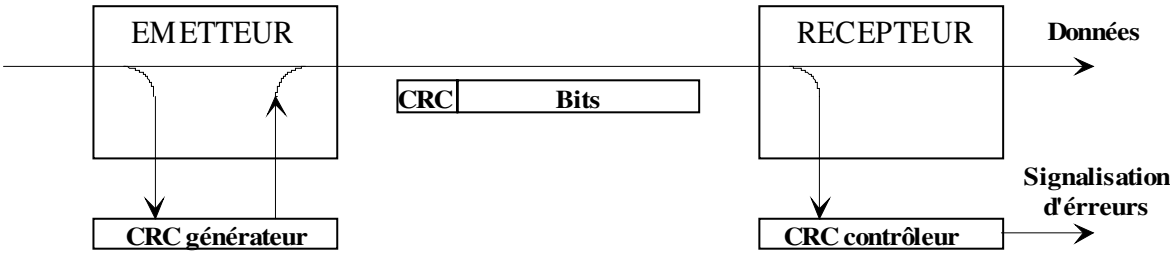
Exemple de structure d'un bloc de bits protégé par clé calculée.

Données suite de bits quelconque.	Clé ou CRC 16 bits FCS 32 bits
Bloc ou TRAME à Transmettre	

Le bloc de N+1 bits à transmettre est considéré comme un polynôme de degré N, la clé est le reste de la division de ce polynôme par un autre polynôme normalisé, appelé polynôme générateur.

Le CRC (ou FCS) est calculé au vol sur les bits émis et ajouté en fin de bloc (trame). En réception, on recalcule au vol le CRC sur le bloc reçu. Si le résultat est différent de celui trouvé par l'émetteur, le récepteur ignore les données reçues et signale à l'émetteur qu'il a détecté une erreur, voir figure suivante.

Mise en œuvre de la détection d'erreurs par clé calculée.



En arithmétique booléenne figure suivante, l'addition et la soustraction sont la même opération. Si le diviseur, appelé polynôme générateur, est de degré N, le reste de la division booléenne est un polynôme de degré N-I.

Addition

	0	**1**
0	0	1
1	1	0

Soustraction

	0	**1**
0	0	1
1	1	0

Multiplication

	0	**1**
0	0	0
1	0	1

Exemple:

On désire protéger le message « 110111» par une clé calculée à l'aide du polynôme générateur $x^2 + x + 1$.

Au message	1	1	0	1	1	1
on fait correspondre le polynôme	x^5	$+x^4$	$+x^3$	$+x^2$	$+x^1$	$+1$

Le dividende doit être, au moins, de même degré que le diviseur. Pour réaliser cette condition, on multiplie le polynôme représentatif du message par x^m où m est le degré du polynôme générateur. Le dividende devient:

$$(x^5 + x^4 + 0 + x^2 + x^1 + 1) \times x^2 = x^7 + x^6 + 0 + x^4 + x^3 + x^2 + 0 + 0$$

								$x^2 + x^1 + 1$
x^7	$+x^6$	$+0$	$+x^4$	$+x^3$	$+x^2$	$+0$	$+0$	$x^5\ x^3\ 1$
x^7	x^6	x^5	↓	↓	↓			
		x^5	x^4	x^3	↓			
		x^5	x^4	x^3	↓			
					x^2	0	0	
					x^2	x	1	
			RESTE		$\Rightarrow$	x	1	

Le reste de la division polynomiale est de degré inférieur à celui du diviseur, la division est terminée.

La division est réalisée par des systèmes «hardware» qui effectuent des «ou exclusif».

Appliquons la division par «ou exclusif» au polynôme 1010010111. Si le polynôme générateur est $x^4 + x^2 + x + 1$, il lui correspond la séquence binaire:

$$1\times(x^4) + 1\times (x^2)\pm 1\times (x^1)+1(x^0) \text{ soit } 10111$$

Multiplier par x^n, le polynôme représentatif du message, revient à ajouter n bits à 0 au message (voir exemple précédent). Le degré du polynôme générateur étant de 4, on ajoute 4 zéros à la trame de données.

On obtient la division ci-dessous:

														10111
1	0	1	0	0	1	0	1	1	1	**0**	**0**	**0**	**0**	100 110 0100
1	0	1	1	1										ce quotient est sans intérêt
0	0	0	1	1	1	0	1							
			1	0	1	1	1							
			0	1	0	1	0	1						
				1	0	1	1	1						
				0	0	0	1	0	1	**0**	**0**			
							1	0	1	**1**	**1**			
							0	0	0	**1**	**1**	**0**	**0**	

Le reste (clé) comporte 4 chiffres, il est de degré -1 par rapport au polynôme générateur. En algèbre booléenne, si le reste est de même degré, une division est encore possible. Le reste est 1100, le CRC4 est donc 1100. Le message à transmettre est

10100101111**1100**

A la réception, l'ensemble message et clé subit la même opération, si le reste de la division est égal à zéro, on suppose que le message n'a pas été affecté par une erreur de transmission.

Vérification à la réception:

message										reste				
1	0	1	0	0	1	0	1	1	1	**1**	**1**	**0**	**0**	10111
1	0	1	1	1										
0	0	0	1	1	1	0	1							
			1	0	1	1	1							
			0	1	0	1	0	1						
				1	0	1	1	1						
				0	0	0	1	0	1	**1**	**1**			
							1	0	1	**1**	**1**			
							0	0	0	**0**	**0**	**0**	**0**	

Le message a été correctement transmis, le reste de la division (message + reste) est nul.

D) Les polynômes générateurs

Les polynômes générateurs utilisés font l'objet de normalisation. Le degré du polynôme est d'autant plus important que la probabilité d'apparition d'une erreur l'est, ou que la longueur du bloc à protéger est importante. Les principaux polynômes employés sont:

➲ Protection de l'en-tête des cellules ATM.

$$x^8 + x^2 + x + 1$$

➲ Détection d'erreur couche AAL type 3 et 4 d'ATM,

$$x^{10} + x^9 + x^5 + x^4 + x + 1.$$

➲ Avis du CCITT $N^0 41$.

$x^{16} + x^{12} + x^5 + 1$.permet de détecter:

- toutes les séquences d'erreurs de longueur égale ou inférieure à 16 bits,
- toutes les séquences erronées comportant un nombre impair de bits,

-99,99% des erreurs de longueur supérieure à 16 bits,
-utilisé dans X25 (HDLC).

➲ Comité IEEE 802

$$x^{32} + x^{26} + x^{23} + x^{22} + x^{16} + x^{12} + x^{10} + x^8 + x^7 + x^5 + x^4 + x^2 + 1$$

- utilisé dans les réseaux locaux.

E) Brouillage et débrouillage

Une technique, s'apparentant à la cryptographie, consiste à transmettre le quotient (Q_x) et le reste (R_x) d'une division polynomiale. Cette opération porte le nom de brouillage. Le récepteur ayant connaissance du diviseur (d_x) retrouve les données (le dividende D_x) en effectuant l'opération (débrouillage):

$$D_x = Q_x \times d_x + R_x$$

Les polynômes diviseurs peuvent être obtenus par une séquence pseudoaléatoire (la génération de cette séquence devant être la même par l'émetteur et le récepteur) ou faire l'objet d'un avis du CCITT.

L'avis V22 (Modem V22) met en œuvre un circuit brouilleur / débrouilleur utilisant le polynôme:

$$1 + x^{14} + x^{17}$$

L'utilisation de la technique de brouillage et débrouillage permet, aussi, d'améliorer la résistance aux erreurs des transmissions multisymbole en supprimant certaines pointes d'énergie.

F) Les codes auto-correcteurs

Dans les systèmes auto-correcteurs, on fait correspondre à chaque mot à transmettre un nouveau mot (***mot code***) tel que 2 mots successifs diffèrent de n bits (n est appelé distance de Hamming). Si n est la distance de Hamming on peut:

- ➲ détecter toute erreur portant sur (n-1) bits,
- ➲ corriger toute erreur portant sur (n-1) / 2 bits.

Dans la technique du bit de parité la distance de Hamming est de 2, on peut détecter toute erreur portant sur 1 bit.

Supposons le code ci-dessous:

Mots	Mots Code
00	10011
01	10100
10	01001
11	01110

Dans ce code de Hamming, il y a toujours, au moins, trois bits qui diffèrent d'un mot code à un autre, la distance de Hamming est de trois. Ce code permet de détecter toutes les erreurs portant sur 2 bits et de corriger toutes les erreurs ne portant que sur un seul bit.

Soit le mot 00, on transmet 10011, une erreur sur un bit correspond à la réception d'un des mots suivants:

10010 10001 10111 00011

Le mot reçu ne correspondant à aucun des mots du code, on recherche celui qui s'en approche le plus (distance de Hamming la plus petite). Dans notre cas, nous retrouvons bien 10011 soit le mot origine 00.