REPUBLIQUE ALGERIENNE DEMOCRATIQUE ET POPULAIRE

MINISTERE DE L'ENSEIGNEMENT SUPERIEUR ET DE LA RECHERCHE SCIENTIFIQUE

Université de Constantine

Faculté des Sciences de l'Ingénieur

Laboratoire d'Electromagnétisme et de Télécommunication

MAGISTERE MICROONDES

Protocoles de Communications Satellitaires avec Sécurisation de la transmission et Récupération des Données

Président	M. BENNIA	Professeur Université Constantine
Rapporteur	M. BENSLAMA	Professeur Université Constantine
Examinateurs	D.SOLTANI	Professeur Université Constantine
	T.LAROUSSI	Maître de Conférences Université Constantine

Année Universitaire 2007 /2008

Remerciements

Je remercie très particulièrement mon encadreur du mémoire : le professeur **Malek BENSLAMA**, et ce pour de multiples raisons. Tout d'abord, je suis lui très reconnaissant d'avoir dirigé mon travail de recherche tout en me laissant libre d'explorer des pistes à ma guise. De plus, je le remercie pour ses conseils avisés tout au long de ce mémoire ainsi pour avoir été présent et encourageant. Mercie infiniment monsieur pour la confiance que vous m'avez accordé.

Je suis reconnaissant au Professeur **Abdelhak BENNIA** pour avoir bien voulu présider le jury soutenance.

J'exprime ma gratitude au Professeur **Fouzi Soltani** qui a bien voulu examiner ce travail et faire partie du jury.

Mes remerciements au Docteur **Tewfik Laroussi** qui a bien voulu examiner ce mémoire et faire partie du jury.

Je tiens à remercier les membres de mon équipe au laboratoire LET de Constantine pour leurs encouragements pendant le travail, en particulier: **M**erabtine, **L**ottfi et **Sk**ander.

Je remercie très chaleureusement monsieur **Hakiki Khalid** pour son grand soutien, ainsi que son collègue.

Je voudrais aussi remercier tous les collègues et amis: Attef, Samir, Ahmed, Fayçal, Khaled, Issam, Tahar, Bassem, Lyes, Djamel, Yacine, Azdine, groupe ABB. Sans oublier mes amis de la cité, que je connais récemment: Messaoud, Mohamed, Mohamed, Imed et Houcine. Et mes frères du Scout _Elmanar_Barika: Zitouni, Abdelkarim, Samir-k-, Talibou, Hichem, Houceme, Lyes, Belbel, Samir-s-, Abdelkarim-r-, mouhelkheir……..

Mon affection particulière à : mon grand-père **Hamou** et ma grand-mère **Manouba**, ainsi que toute la **famille.**

Je souhaite maintenant remercier du fond du cœur, ma petite famille pour son soutien de tous les instants, mes frères: **Ali** et **Abdelhamid** et mes sœurs: **Omelkheir** et **Wafia**.

Finalement, je pense que je ne suis arrivé à ce stade que grâce aux encouragements de mes parents, mon **père Abdelmadjid** et ma **mère Farida**, qui m'ont encouragé tout au long de mes études et qui ont toujours été là pour moi : je ne saurais être qu'infiniment reconnaissant quant aux sacrifices qu'ils ont consentis. ***À eux je dédie ce travail et je leur dis mille mercis.***

Introduction Générale

Les communications satellitaires doivent utiliser la largeur des bandes de fréquences allouées de manière optimale. Il s'agit en effet, d'une part de transmettre un maximum de données utiles par unités de temps entre la source et le destinataire, mais également de fixer les règles permettant à tous les émetteurs de communiquer de façon optimale [1,2]. Il sera donc nécessaire de définir les principes de communication à l'intérieur du médium pour que les utilisateurs puissent se partager le canal. Ces principes basés sur le partage de la ressource sont appelés techniques ou méthode d'accès. Le choix des paramètres correspondants à un type d'accès multiple particulier constitue le facteur critique lorsque l'on cherche à optimiser l'efficacité spectrale, c'est-à-dire le débit d'information véhiculé dans une unité de largeur de bande. De nombreuses recherches ont d'ailleurs été entreprises afin de trouver le meilleur compromis possible entre l'accès multiple par répartition de temps (TDMA), de fréquences (FDMA) et de codes (CDMA). Une solution envisageable serait d'utiliser une technique hybride entre les trois méthodes d'accès multiple intégrant les avantages de chaque technique. Néanmoins la conception d'un appareil utilisant ce procédé constituerait un réel défit technologique en raison de sa complexité. Les techniques à accès aléatoire sont conçues pour permettre une approche décentralisée pour l'accès au canal de communication, lorsqu'une source utilise le canal seulement en cas de besoin : quand elle a vraiment des informations à transmettre. Contrairement au FDMA et TDMA, quand le canal n'est pas utilisé par une source, il est complètement disponible pour d'autres sources. Dans le cas où un grand nombre de sources sont souvent inactives, l'accès aléatoire permet une utilisation beaucoup plus efficace du canal. En ajoutant à cela la simplicité de la mise en œuvre due à la décentralisation des protocoles d'accès aléatoire. Les techniques à accès aléatoire ont été adaptées aux réseaux de satellite en tenant compte du délai important de propagation. Le principe est simple: on émet un paquet quand on veut mais si l'on ne reçoit pas d'acquittement, on considère qu'il y a eu collision et on attend un temps aléatoire avant de réémettre le paquet.

L'inconvénient majeur de ces protocoles de base est le faible débit, à cause des phénomènes de collision des paquets. Les protocoles Aloha et Csma souffrent de la collision des paquets qui influe sur la fiabilité du système et conduit ainsi à un affaiblissement grave de débit [3]. Nous avons donc besoin d'augmenter le débit de transmission tout en gardant ou en améliorant la qualité de ceux-ci, mais sans souci de fiabilité tous les efforts d'amélioration

seraient vains car cela impliquerait forcément à ce que certaines données soient retransmises. Pour cela, l'introduction des codes correcteurs permettra d'améliorer de manière concomitante le débit et la fiabilité. Cet aspect sera abordé dans ce mémoire.

Notre étude a été concentrée sur les méthodes d'accès au médium de communication, en particulier les méthodes d'accès fixe (TDMA, FDMA et CDMA) et les techniques à accès aléatoire (ALOHA et CSMA), où on a montré les caractéristiques ainsi que les avantages et les inconvénients pour chaque technique pour pouvoir tirer celle qui offre la meilleur qualité de transmission. Ce mémoire est organisé en deux parties principales:

Partie I: ***Présentation des Techniques à accès Aléatoire dans les communications satellitaires.*** **C**ette partie comporte un chapitre;

- Dans ce chapitre on effectue une description générale des techniques à accès aléatoire pour la résolution de collision, on parle de la technique Aloha et ses dérivées et la technique Csma et ses dérivées. Les protocoles d'accès aléatoires présentent en générale de bien meilleurs délais que le temps partagé périodique, sous réserve bien entendu d'une résolution efficace de collision. Les techniques d'accès aléatoire sont conçues pour permettre une approche décentralisée pour l'accès au canal de communication, lorsqu'une source utilise le canal seulement en cas de besoin : quand elle a vraiment des informations à transmettre. Contrairement au FDMA et TDMA, quand le canal n'est pas utilisé par une source, il est complètement disponible pour d'autres sources. Dans le cas où un grand nombre de sources sont souvent inactives, l'accès aléatoire permet une utilisation beaucoup plus efficace du canal. En ajoutant à cela la simplicité de la mise en œuvre due à la décentralisation des protocoles d'accès aléatoire. Dans les réseaux satellitaires, on peut utiliser les techniques d'accès aléatoire soit directement pour transmettre des informations, soit pour faire des réservations, c'est à dire pour demander l'allocation d'une bande de fréquence fixe.

Dans ce chapitre on fait une comparaison de débit entre les différentes techniques d'accès aléatoire afin d'examiner l'efficacité de transmission et la fiabilité de signal que peut apporter chaque technique.

Partie II: *Présentation de la technique* ***Erasure*** *pour récupérer les paquets et améliorer le débit de la transmission des techniques à accès aléatoire.* Sur deux chapitres.

- Un code correcteur d'erreur permet de corriger une ou plusieurs erreurs dans un mot code en ajoutant aux informations des symboles redondants, autrement dit, des symboles de contrôle. Différents codes possibles existent mais dans cette partie on traitera seulement les codes de Reed–Solomon car pour le moment, ils représentent le meilleur

compromis entre efficacité (symboles de parité ajoutés aux informations) et complexité (difficulté de codage).

Ø Dans la deuxième partie, nous proposons d'utiliser le codage Erasure à base des codes RS afin d'améliorer le débit de transmission, en corrigeant les erreurs et récupérant ainsi des paquets perdus pendant les communications. Avec le codage Erasure, un nombre K des paquets originaux sont codés en (N,K) mots Erasures, qui consiste à N (>K) paquets codés. À la réception, on peut récupérer un nombre K de paquets originaux si on reçoit un nombre K entre N paquets codés. Alors le codage Erasure augmente la charge offerte du trafic, et la probabilité de succès des paquets de transmission peut être aussi augmentée si le codage est bien implémenté.

Ø Le travail effectué dans ce mémoire a fait l'objet d'un dépôt d'un article sur HAL, il est actuellement en ligne. Un autre article plus élaboré a fait l'objet d'une soumission à l' IGARSS (International Geoscience and Remote Sensing) 2008 de Boston USA.

I.1 Introduction:

Les protocoles d'accès multiples en particulier le TDMA, FDMA et CDMA qui consistent à partager le canal de communication entre plusieurs utilisateurs ont été discutés dans [3], où on peut voir que le TDMA est en général peu efficace et induit de lourds délais d'accès au canal. En effet, les délais sont proportionnels au nombre d'utilisateurs. Dans ce chapitre nous examinons en particulier des protocoles à accès aléatoire à résolution de collusions. Quand deux utilisateurs transmettent en même temps, un seul paquet est susceptible de passer, et les paquets émis simultanément se détruisent mutuellement: c'est une collision. Les protocoles d'accès aléatoires présentent en général de bien meilleurs délais que le temps partagé périodique, sous réserve bien entendu d'une résolution efficace de collisions [4].

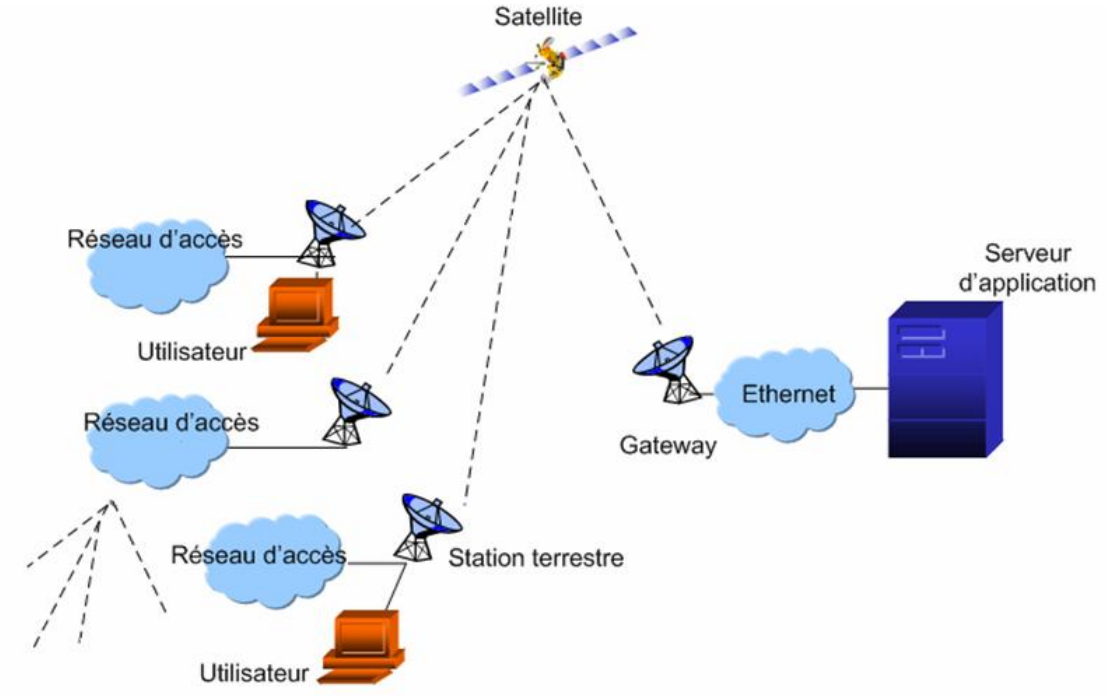


Figure 1.1 Les protocoles de communications.

Les techniques d'accès aléatoire sont conçues pour permettre une approche décentralisée pour l'accès au canal de communication, lorsqu'une source utilise le canal seulement en cas de besoin : quand elle a vraiment des informations à transmettre. Contrairement au FDMA et TDMA, quand le canal n'est pas utilisé par une source, il est complètement disponible pour d'autres sources. Dans le cas où un grand nombre de sources sont souvent inactives, l'accès aléatoire permet une utilisation beaucoup plus efficace du canal. En ajoutant à cela la simplicité de la mise en œuvre due à la décentralisation des protocoles d'accès aléatoire, on peut

comprendre l'importance de ces techniques, aussi bien dans les réseaux locaux terrestres, que dans les réseaux satellitaires.

Dans les réseaux satellitaires, on peut utiliser les techniques d'accès aléatoire soit directement pour transmettre des informations, soit pour faire des réservations, c'est à dire pour demander l'allocation d'une bande de fréquence fixe [5].

I.2 L'Aloha: *(Aloha Classique)*

Le nom de cette méthode provient des expériences faites à l'université de Hawaï pour relier les centres informatiques dispersés sur plusieurs îles. Les stations émettent, de façon inconditionnelle, des paquets dés qu'ils sont en leur possession. Il n'y a pas d'écoute du support avant la transmission. De plus, le temps de propagation des signaux sur le canal satellite est un facteur contraignant car les stations sont averties d'une collision seulement 270 ms après l'émission des données. Dans le cas où la transmission de données ne s'est pas bien passée, la station va retransmettre les paquets après un délai aléatoire. Cette méthode d'accès a donc un taux d'utilisation du canal satellite faible, approchant les 20%, d'où l'apparition des techniques similaires mais avec des modifications qui apportent de meilleures performances [5,6].

La transmission dans ce protocole est complètement décentralisée. A la fin de la transmission de chaque paquet de chaque source, la source reçoit l'information si

- ü le paquet est bien reçu ou
- ü il y a eu une collision, et dans ce cas il y aura une retransmission à un instant ultérieur [7].

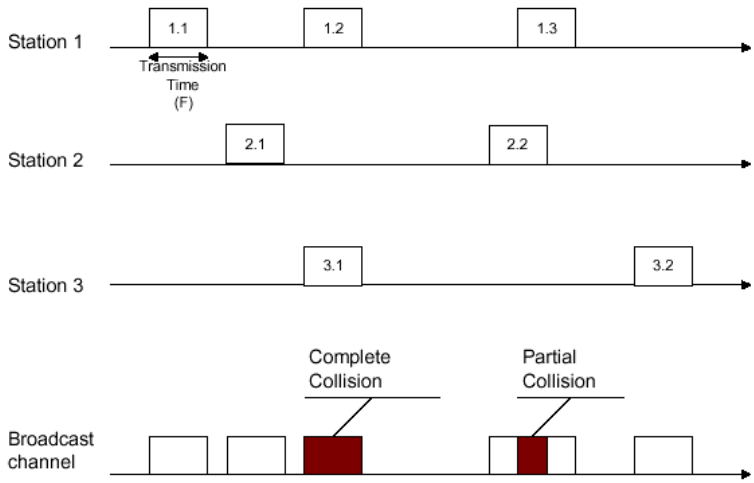


Figure 1.2 La Collision dans le protocole Aloha.

I.2.1 La Loi de Poisson dans la modélisation du trafic:

L'étude des canaux à diffusion et à accès multiples nécessite une nouvelle composante théorique qui est la modélisation de trafic. Le trafic est le concept qui décrit l'ensemble des sources d'informations connectées au réseau et la répartition dans le temps de leur activité. La loi de poisson est aux modèles de trafic ce que la loi de Bernoulli est aux modèles des sources d'informations: le modèle le plus simple dont la connaissance est un préalable indispensable. La loi de poisson est la manière la plus simple de décrire le processus de génération de paquets d'informations issus de sources nombreuses et indépendantes. À l'origine, le mathématicien Poisson avait introduit sa loi pour expliquer la statistique des chutes de chevaux dans la grande armée. Récemment la loi de Poisson a été utilisée pour modéliser les émissions de particules en radioactivité, les atomes remplaçant les chevaux et les neutrons les cavaliers [7]. Revenant à la description de la loi de Poisson (processus de comptage) dans les réseaux de télécommunication. Supposons que le taux de génération de paquet d'une source soit de l paquets par unité de temps, et considérons un intervalle de temps arbitraire de longueur Δt. La loi de Poisson établit que la probabilité pour que k paquets, soient générés durant l'intervalle de temps en question est [4] :

$$P = \frac{(l.\Delta t)^k}{k!} e^{-l.\Delta t} \qquad (1.1)$$

Une autre manière de décrire la loi de Poisson est de caractériser la durée aléatoire I séparant deux événements consécutifs. La probabilité $\Pr\{ I \geq t \}$, pour que cette durée soit plus grande q'un réel positif t donné a pour expression: $\Pr\{ I \geq t \} = e^{-lt}$

Ou à l'échelle infinitésimale:

$$\Pr\{ I \in [t, t+dt] \} = e^{-lt}\, l\, dt \qquad (1.2)$$

I.2.2 Le Débit de canal dans Aloha:

Le débit d'un canal n'est pas le seul paramètre important qui indique la stabilité d'une méthode d'accès, mais il est essentiel. Dans le système Aloha, le délai est une conséquence de l'occurrence de collision [8].

La probabilité que n paquets arrivent en deux temps différents de paquets est donnée par [9-11]:

$$P(n) = \frac{(2l)^n e^{-2l}}{n!} \tag{1.3}$$

Où l est la charge du trafic.

La probabilité $P(0)$, pour qu'un paquet soit reçu avec succès et sans collision est:

$$P(0) = e^{-2l} \tag{1.4}$$

Alors le débit S est donné comme suit:

$$S = l.P(0) = l.e^{-2l} \tag{1.5}$$

Et donc le débit max de cette technique est:

$$S_{\max} = \frac{1}{2.e} \approx 0.184 \tag{1.6}$$

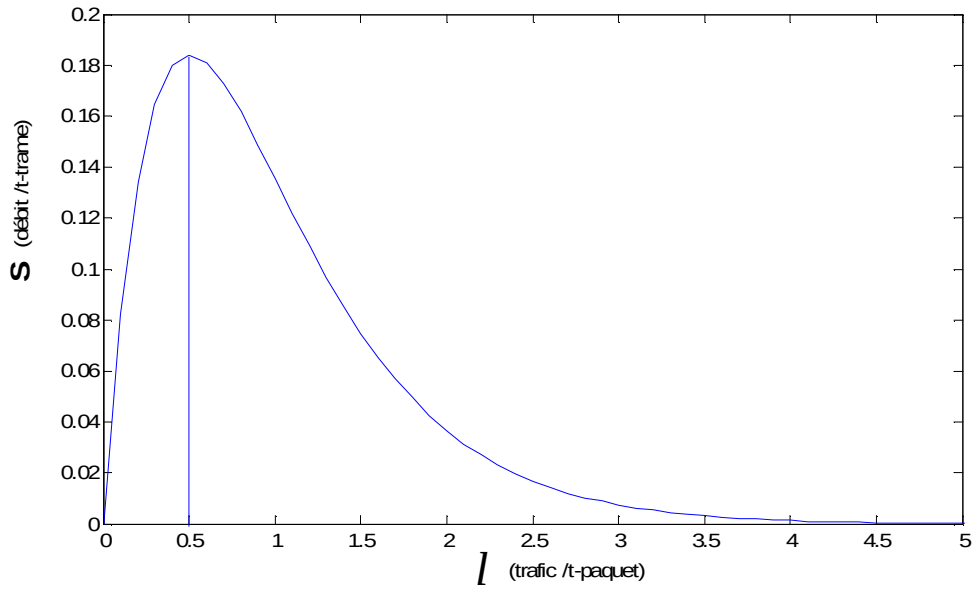


Figure 1.3 Le Débit d'Aloha.

I.3 La Technique Slotted-Aloha: *(Aloha en tranches)*

On suppose que le temps est partagé en tranches appelées slots. Sur chacun des slots les utilisateurs transmettent ou retransmettent indépendamment leur paquet avec chacun une probabilité P spécifiée à l'avance. Si le nombre d'utilisateurs susceptibles de transmettre est grand on estime que les transmissions effectives par slot suivent une loi de Poisson de paramètre l [7].

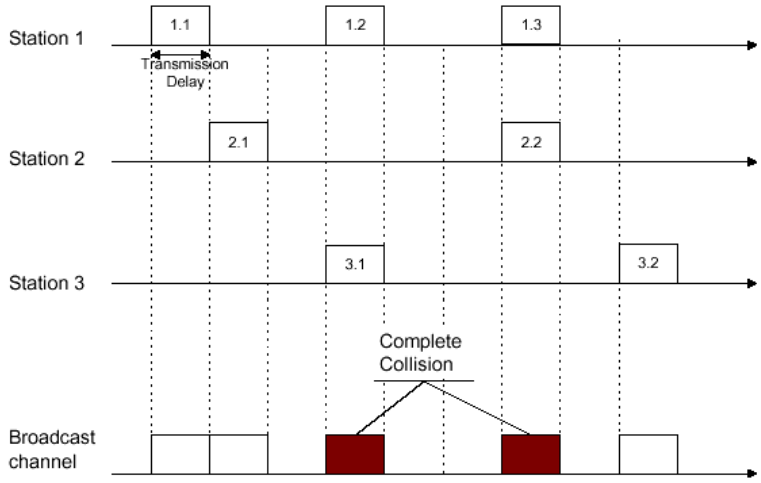


Figure 1.4 La Collision dans le protocole S-Aloha.

Le principe de cette technique consiste à discrétiser le temps en tranches de tailles égales appelées slots. Toutes les stations sont synchronisées et chaque station transmet un paquet au début d'un slot.

I.3.1 Le Débit de canal dans S-Aloha:

n : le nombre d'utilisateurs,

P_i : la probabilité qu'un utilisateur émette un paquet/ i : pour le i^{éme} paquet,

S_i : la probabilité qu'un paquet soit reçu avec succès.

S'il n'y a pas un autre utilisateur en état de transmission, au début du time slot, la fonction de probabilité S_i sera égale à [8]:

$$S_i = \frac{P_i}{(1-P_i)} \prod_{i=1}^{n} (1-P_i) \tag{1.7}$$

Et lorsque le débit de trafic est S et le trafic est λ, on peut écrire:

$$S_i = \frac{S}{n} \quad \text{et} \quad P_i = \frac{l}{n} \tag{1.8}$$

$$\frac{S}{n} = \frac{l}{n} \cdot \left[\frac{1}{(1-\frac{l}{n})} \right] \cdot \prod_{i=1}^{n} (1-\frac{l}{n}) \tag{1.9}$$

$$S = l.e^{-l} \tag{1.10}$$

Ce qui fait que le maximum débit du Slotted Aloha est de:

$$S_{\max} = \frac{1}{e} \approx 0.37 \tag{1.11}$$

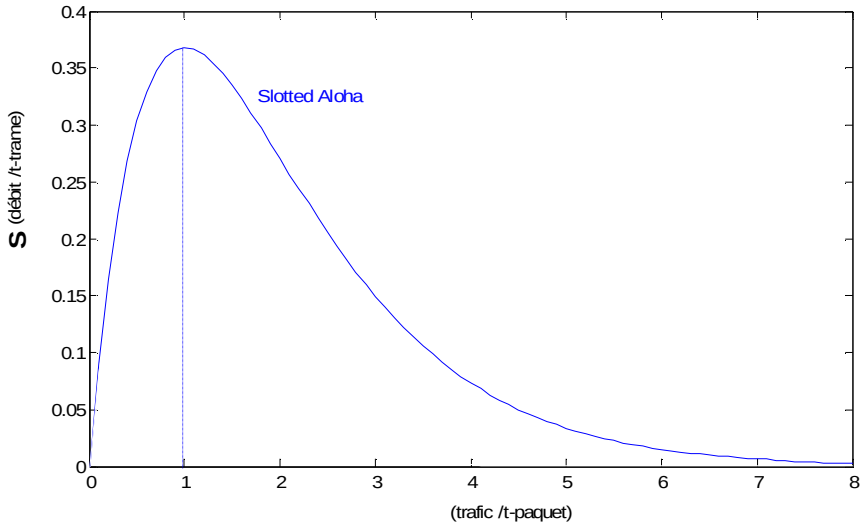


Figure 1.5 Le Débit de S-Aloha.

I.3.2 Aloha et Slotted-Aloha:

L'efficacité de la transmission et la stabilité du système sont en général les paramètres les plus importants à examiner pour un protocole de communication satellitaire. La technique Aloha pure (ou non slottée) présente une faible immunité face aux problèmes de collisions, ainsi qu'une efficacité de transmission de bas niveau ($S_{\max} \approx 0.185$) et aussi un régime de communication instable. Cependant la S-Aloha offre un débit plus grand ($S_{\max} \approx 0.37$), une meilleure synchronisation entre les différentes sources de communication et plus précisément une plus grande capacité pour éviter les collisions.

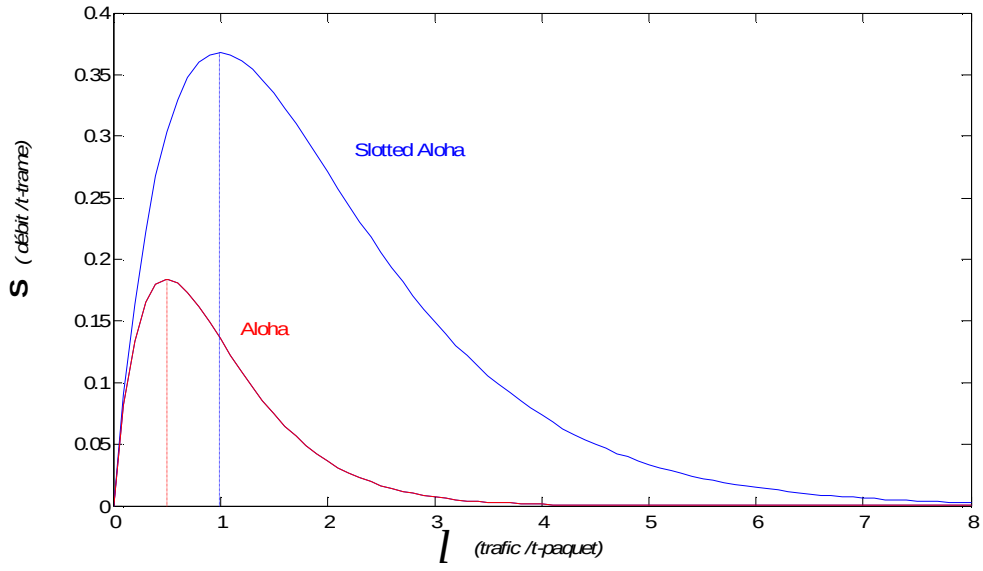


Figure 1.6 Comparaison des Débits entre Aloha et S-Aloha.

I.4 La Multi-copy Aloha:

Lorsqu'on envoie m copies d'un paquet dans la technique Slotted Aloha (Multi-copy), la probabilité de succès de la transmission pour ce paquet, ou la probabilité qu'un paquet parmi les m copies envoyés ne tombe pas en collision, va être plus élevée par rapport à celle où une seule copie est envoyée (S-Aloha). Cela est vraie seulement lorsque les autres paquets sont envoyés en une seule copie ou quand le trafic du canal demeure stable sans perturbations. Pour maximiser la probabilité de succès de transmission, on va supposer que tous les utilisateurs transmettent le même nombre de copies (m) [10].

La Multi-copy Aloha est généralement conçue pour les systèmes suivants:

- Ø Les systèmes satellitaires; offrant une meilleur probabilité de succès de transmission,
- Ø Les systèmes Aloha Multicanaux,
- Ø Les systèmes Aloha à réservations, avec une probabilité de succès très élevée.

I.4.1 Le Débit de canal avec Multi-copy Aloha:

On considère que l'arrivée des paquets est un processus de Poisson. Pour la simple Slotted Aloha, on suppose que le délai moyen de la retransmission est plus grand que 5 slots, alors que la valeur moyenne du délai pour les M-Copies en incluant la première transmission doit être aussi grande que 5 slots.

Pour le trafic on a d'après [10]:

$$\lambda=\lambda 1+\lambda 2+\ldots..\lambda n=\sum_{i=1}^{n} l i \qquad (1.12)$$

Le nombre moyen des copies (N) par paquet égale à:

$$N = l^{-1}.\sum_{i=1}^{n} i.l_i \qquad (1.13)$$

La probabilité P_i pour que le paquet i soit reçu avec succès est:

$$P_i = 1 - prob[tous.les.copies.sont.en.collision] = 1-(1-e^{-N\,l})^i$$

Le débit $S_i = l_i.P_i$ (1.14)

Et pour un nombre k copies, on aura:

$$S_k = l.P_k = l\left[1-(1-e^{-k.l})^k\right] \qquad (1.15)$$

[10] :Pour k =1,2,3,…., et $k = \dfrac{\ln 2}{l}$ (1.16)

Donc Le débit:

$$S = \frac{\ln 2}{k}\left[1-\left(\frac{1}{2}\right)^{k}\right] = S_k \qquad (1.17)$$

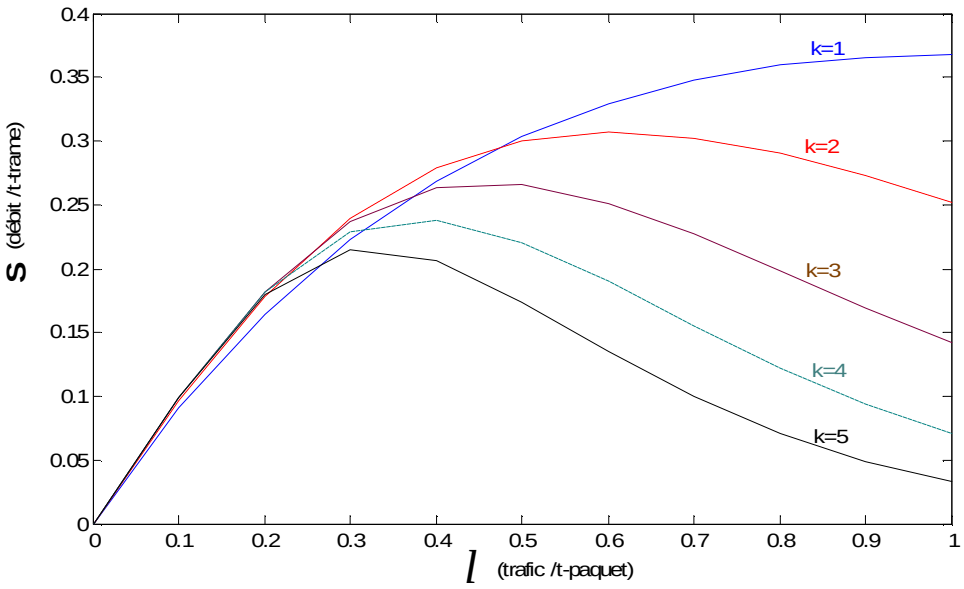


Figure 1.7 Le Débit de Multi-Copy Aloha.

La figure 1.7 montre la variation de la probabilité de succès P par apport au trafic l avec$1 \le k \le 5$. On voit que pour maximiser la probabilité de succès de transmission, une seule copie doit être transmise lorsque le trafic du canal est supérieur à 0.48, et quand $l \in [0.28,0.48]$, on peut transmettre que deux copies. La même figure montre, que lorsque l est petit, la probabilité P peut atteindre la valeur max (p=1) en utilisant un k supérieur.

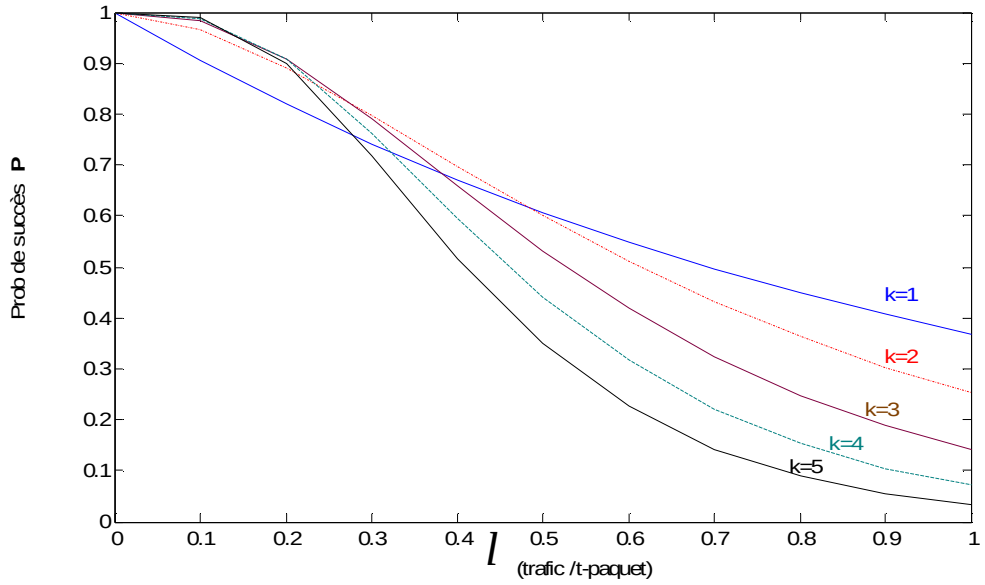


Figure 1.8 Probabilité de succès Avec Multi-copy Aloha.

I.5 CSMA (Accès Multiple avec écoute de Porteuse):

La CSMA, pour Carrier Sense Multiple Access est utilisée si les stations sont capables de sentir la porteuse, et si elles transmettent seulement lorsque le canal est libre, alors dans ce cas on peut bien éviter plusieurs collisions.

Le principe du CSMA peut être expliqué de la manière suivante. Une station (utilisateur) qui veut transmettre dans un canal, écoute d'abord le milieu de communication (cas d'un maillage avec bus). Si le milieu est libre elle transmet, si non la station attend pendant un temps spécifique. Si le transmetteur n'a pas reçu l'information après un moment donné, il suppose qu'il y a une collision. Après la collision la station attend pendant une période aléatoire puis elle retransmet [11].

Il existe plusieurs variants du CSMA, chaque type de CSMA spécifie comment se comporter devant un milieu occupé:

- CSMA non Persistent,
- CSMA Persistent et
- CSMA P-Persistent.

I.5.1 CSMA Non-Persistent:

1-Si le milieu est libre, envoie immédiatement

2-Si le milieu est occupé, attend un instant de temps variable puis répète ***étape1***. Ceci permet de réduire la probabilité de collisions.

I.5.2 CSMA Persistent:

1-Si le milieu est libre, transmet immédiatement,

2-Si le milieu est occupé, alors il continue d'écouter le canal jusqu'à se qu'il devienne libre, alors il transmet immédiatement.

Il y aura toujours des collisions si deux stations veulent retransmettre.

I.5.3 CSMA P-Persistent:

1-Si le milieu est libre, transmet avec une probabilité P, et attend un délai avec une probabilité (1-P),

2-Si le milieu est occupé, il continue d'écouter jusqu'à ce qu'il devienne libre, et répète l'***étape 1 (écouter le canal)***,

3-Si la transmission a pris un délai très long, alors il faut reprendre l'***écoute(écouter le canal)***. Ceci permet de diminuer la probabilité de collision par apport au CSMA persistant [11].

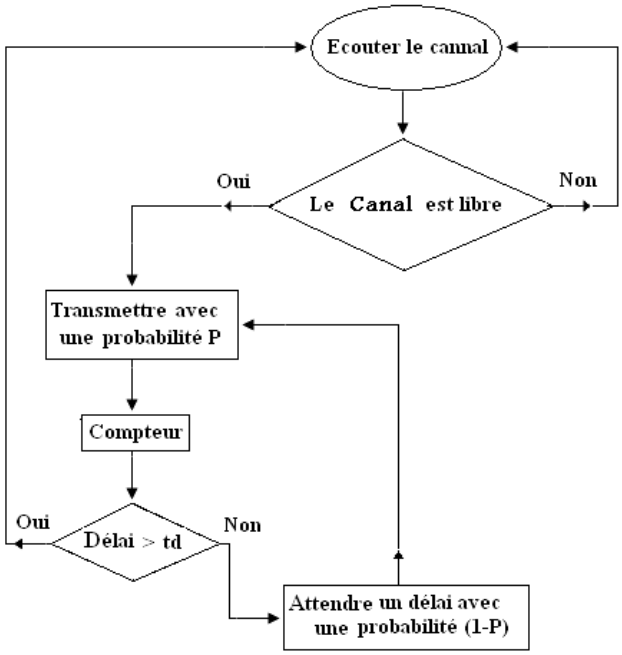


Figure 1.9 CSMA P-Persistent.

I.5.3.1 Comment sélectionner la probabilité *P*:

On suppose qu'il y a ***N*** stations qui veulent transmettre un paquet, et que le milieu est occupé. On estime qu'il y aura N_p stations qui peuvent transmettre juste lorsque le milieu sera libre. Si N_p est supérieur à 1, alors la collision sera faite.

C'est pour cette raison qu'il faut bien s'assurer que N_p est inférieur à 1, pour éviter les collisions, sachant que ***N*** est le nombre max des stations qui peuvent activer simultanément.

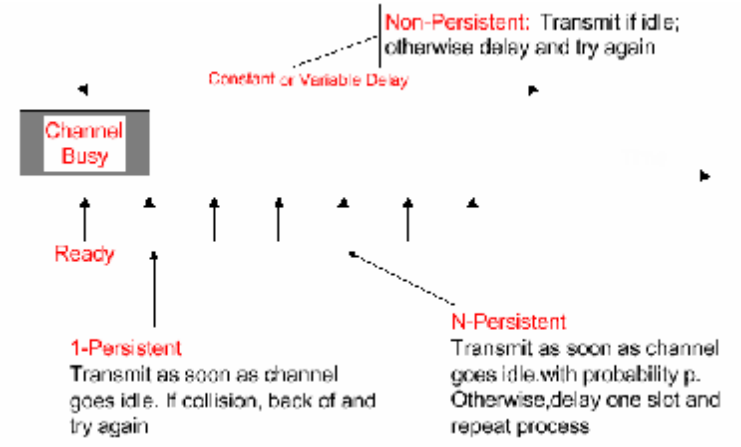
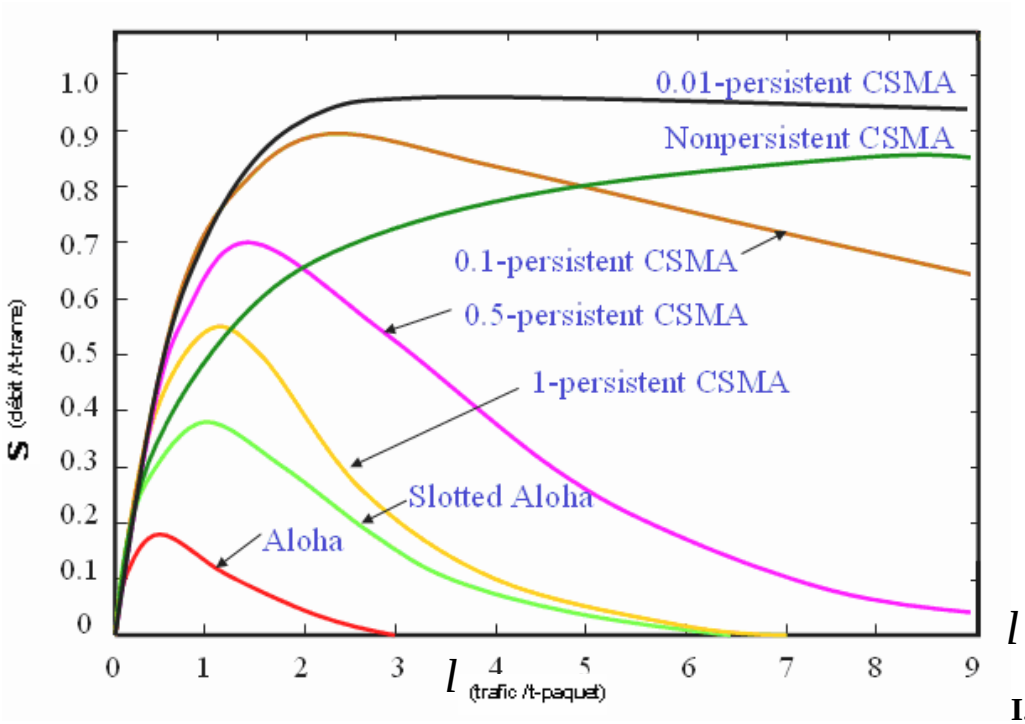


Figure 1.10 Comparaison entre dans différentes stratégies du CSMA.



I.6

Figure 1.11 Comparaison entre les différents protocoles Aléatoires [7].

CSMA/CD (CSMA avec détection de collisions):

Avec le CSMA, une fois qu'elle a commencé à émettre, la source n'écoute plus le canal et envoie la totalité de la trame. Cependant avec CSMA/CD, la source écoute le canal pendant qu'elle émet et si elle détecte une collision, elle stoppe son émission.
Après une collision, le temps est divisé en slots de 2τ secondes (le double du plus long temps de propagation: si on arrive à transmettre pendant ce temps, on est certain que la trame ne subira pas de collision).
Nous allons envisager deux modes de fonctionnement en cas de collision:
***P* Constant:** On a une probabilité constante *P* (≠ du p de P-Persistent) de retransmission dans un slot pour chaque station. Cette probabilité garantit que chaque station restera en attente pendant un temps aléatoire avant de réessayer d'émettre [11].

I.6.1 Le Principe général :

L'idée du protocole CSMA ou plus exactement sa dérivée (CSMA /CD) est « listen before talk ; écouter avant de parler », alors avant de commencer la transmission, la station dite X doit d'abord sentir le support de transmission; s'il y a une porteuse présente, cela signifie qu'il y a une autre station dite Y, qui est en train de transmettre, et dans ce cas X retransmettra après un temps aléatoire. Ceci est avec Non-persistent CSMA. Mais avec 1-Persistent CSMA, la station X continue à écouter, et elle transmet directement après que Y termine sa transmission. Mais le problème avec 1-persistent CSMA c'est la troisième station Z qui veut aussi transmettre en même temps avec X, et dans ce cas, les paquets de X et Z vont se heurtés juste après que Y termine. Ainsi en P-persistent CSMA, l'une des stations X et Z doit générer des nombres aléatoires avec une probabilité p, et transmettre juste après que Y termine sa transmission. Si le nombre aléatoire indique à la station d'attendre, elle attend (*backs off*) pendant une période aléatoire, ensuite elle reprend l'essai, l'avantage c'est que la probabilité de la collision dans ce cas est seulement p^2, maintenant s'il indique que seulement la station Z doit attendre, cette dernière doit faire le *back off*. Cependant si la station portant le message écoute qu'il n'y a rien, la collision peut encore se produire, parce qu'une autre station peut être actuellement en transmission, mais à cause du délai de propagation sa trame peut ne pas arriver à la première station. Pour cette raison, le mécanisme CD est ajouté au CSMA; pendant qu'une station transmet, elle continue de surveiller le canal, pour vérifier si sa trame est sur le canal intact.

Avec ce mécanisme la collision sera détectée beaucoup plus tôt, qu'attendre le signal ACK comme dans le régime Aloha [12].

On pose *I* et *F* des variables aléatoires qui représentent la période occupée et la période libre lorsque le canal est respectivement occupé et libre . La durée moyenne du cycle Occupé/Libre sera égale à :

$$E(F)+E(I) \tag{1.18}$$

Pendant cette période, on a *T* le temps occupé pour transmettre un message avec succès, dans chaque cycle Occupé/Libre, on peut avoir soit une transmission réussie soit l'inverse (collision dans *F*), d'après la définition de la période occupée, il y aura au moins une station qui a des messages à émettre.

t_0 est le temps de départ, le temps de transmission sera égale à $(t_0, t_0+\Delta t)$. La probabilité qu'il n'y a pas une autre station en état d'envoie est [13]: e^{-la}, alors

$$E(T)=1e^{-la}+0.(1-e^{-la})=e^{-la} \tag{1.19}$$

L'utilisation du canal est (*taux d'utilisation*):

$$u=\frac{E(T)}{E(F)+E(I)} \tag{1.20}$$

Où a est le rapport de délai de propagation du paquet par trame.

On note que s'il n'y a pas de collision, *F* sera égale à la somme des (temps de transmission du paquet + le délai de propagation), et en moyenne égale à :

$$F=1+\frac{3}{4}a \tag{1.21}$$

S'il y a de collisions :

$$F=1+\frac{3}{4}a+M \tag{1.22}$$

Où *M* est défini comme suit : Quand une station commence à transmettre à t_0, alors *M* sera la durée de temps écoulée et perdu par cette station lors de sa transmission sur une période occupée. Donc la dernière station commence à t_0+M, on a besoin d'identifier $E(F)$ et $E(M)$. On note que $M \le a$, en rappelant que lorsqu'une collision se produit dans la première place « la station pense que le canal est libre, mais elle n'a pas reçu la transmission déjà en progression, à cause du délai de propagation. La probabilité que $M \le x$ « il n'y a plus de transmission pendant l'intervalle de temps (t_0+x, t_0+a) » est : $e^{l(a-x)}$ et

$$P(M \le x) = e^{-l(a-x)} \tag{1.23}$$

Alors, la fonction de densité de probabilité *M* est:

$$\frac{d}{dx} p(M \le x) = \frac{d}{dx} e^{-l(a-x)} = l e^{-l(a-x)} \tag{1.24}$$

Pour $0 < x < a$;

$$E(M) = \int_0^a x . l e^{-l(a-x)} = a - \frac{1}{l}(1 - e^{-la}) \tag{1.25}$$

Donc;

$$E(F) = 1 + 0.75a + a - \frac{1}{l}(1 - e^{-la}) \tag{1.26}$$

Finalement la quantité $E(I)$ est facilement déterminée, *I* est le temps jusqu'à la première transmission, après certain délai $(t_0 + 1 + a)$, il possède une distribution exponentielle de moyenne $1/G$, Donc, en utilisant les résultats de [12,13] ;

$$u = \frac{l e^{-al}}{l(1 + 1.75a) + e^{-al}} \tag{1.27}$$

L'utilisation du canal est fonction du délai de transmission des paquets et sa variation est inversement proportionnelle au délai de propagation entre les différentes stations du réseau. Dans les réseaux satellitaires les délais de transmission sont importants, et le facteur temps devient de plus en plus influant sur le rendement du système de communication, en rendant l'exploitation du réseau très difficile.

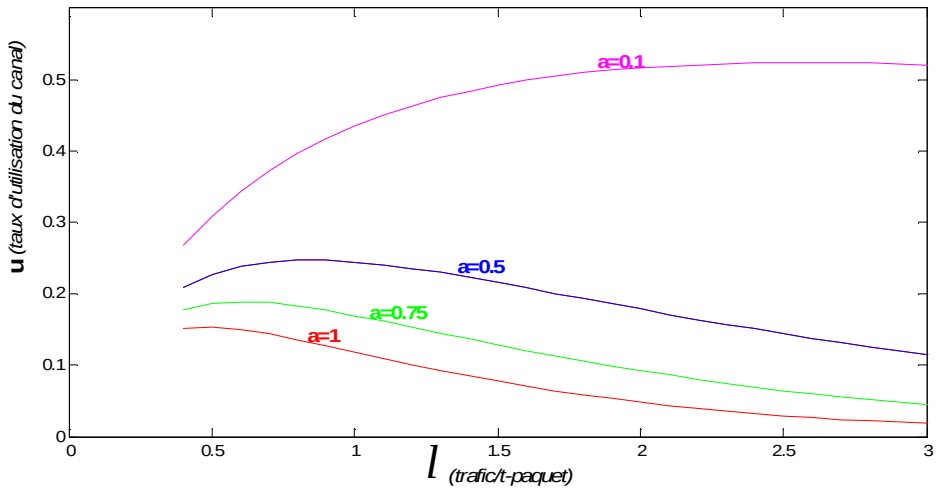


Figure 1.12 Taux d'utilisation du canal de transmission

Pour améliorer la situation, plusieurs solutions sont proposées. Elles sont actuellement les sujets de recherches de plusieurs laboratoires dans le monde. Une solution consiste

à minimiser le taux de collisions, et augmenter le débit effectif de transmission, pour avoir une probabilité de succès meilleure que l'ancienne, est utilisée dans ce mémoire.

I.7 Conclusion:

On distingue deux grandes familles à accès multiple aléatoire : l'Aloha et ses dérivés et le CSMA et ses dérivés que la plupart des réseaux sans fils commencent à utiliser. L'inconvénient majeur de ces techniques est le faible débit, à cause des phénomènes de collision des paquets. Dans ce chapitre on a expliqué le principe de fonctionnement de chaque méthode, suivi par une comparaison des débits obtenus par chacune de ces techniques afin d'évaluer le rendement, et voir l'efficacité de transmission et la fiabilité de signal que pourrait apporter chaque technique.

Nous avons donc besoin d'augmenter le débit de transmission tout en gardant ou en améliorant la qualité de ceux-ci, mais sans souci de fiabilité, tous les efforts d'amélioration seraient vains car cela impliquerait forcément à ce que certaines données soient retransmises. Pour cela, l'introduction des codes correcteurs permettra d'améliorer de manière concomitante le débit et la fiabilité. Cet aspect sera abordé dans ce mémoire.

II. Introduction :

L'histoire du codage de canal ou du codage *FEC* remonte au travail pilote de Shannon en 1948, prévoyant qu'arbitrairement les communications fiables sont réalisables à l'aide du codage de canal, où en ajoutant des informations redondantes au message transmis [14].

Dans Ce chapitre on introduit les notions fondamentales relatives à la protection des données contre les erreurs de transmission, en expliquant le principe général du mécanisme codage/décodage. La partie la plus importante sera une étude bien détaillée sur le codeur/décodeur Reed-Solomon.

La Section II.1 résume l'article "pionnier" de Shannon qui fixe les bases de la théorie de l'information, en stipulant qu'une transmission avec un taux d'erreurs contrôlable est possible à travers un canal bruité.

La Section II.2 décrit deux modes de contrôle des erreurs par codage, soit le '*Automatic Repeat Request* '(***ARQ***) et le '*Forward Error Correction* '(***FEC***). Ces notions sont suivies par la section ***II.3*** qui introduit les principes du codage en bloc linéaire et du codage convolutif. Les caractéristiques de ces codages ainsi que les capacités potentielles de correction d'erreurs seront ensuite discutées.

La Section II.4 présente un rappel des principes du contrôle des erreurs par codage enchaîné. L'intérêt de l'enchaînement de plusieurs opérations de codage de canal –soit en série soit en parallèle- y est brièvement présenté.

II.1 La Théorie de C. E. Shannon :

La théorie de l'information est née en 1948 avec l'article de C. E. Shannon [15]. Cette théorie détermine les limites de performance des systèmes de communication numérique, et anticipe notablement les besoins pratiques de ce type de communication. En généralisant le schéma d'un système de communication (Figure 2.1), Shannon présente des notions, qui s'avèreront être à la base de la théorie de l'information. Dans ce schéma, Shannon identifie cinq éléments principaux [16]:

- ***La source d'informations***: Cet élément génère le message à transmettre, message qui appartient à un groupe prédéfini de messages possibles.
- ***Le transmetteur***: Cet élément est responsable de la préparation du message de manière à permettre sa transmission.

- ***Le canal***: Il entraîne des modifications du signal selon les caractéristiques physiques du media de communication. La modélisation du canal se base sur plusieurs paramètres et éléments, dont certains possèdent des propriétés non prédictives.
- ***Le récepteur***: La tâche de cet élément est la reconstruction et/ou l'estimation du message original à partir du signal reçu.
- ***Le destinataire***: Cet élément constitue l'entité à qui le message est adressé.

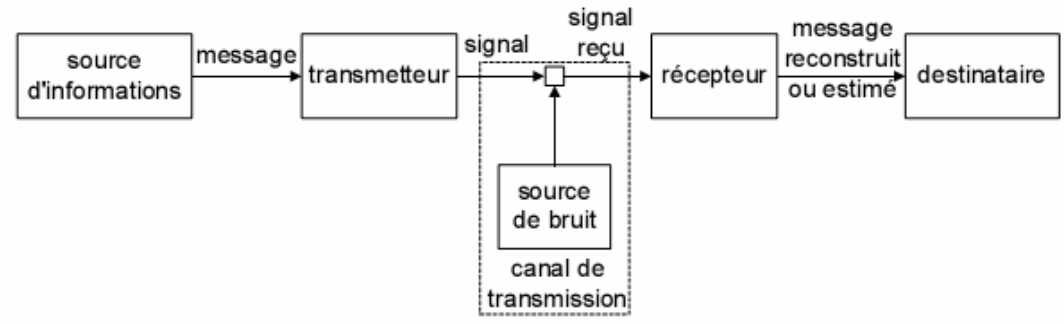


Figure 2.1 Modèle générique des systèmes de communication.

Parmi les informations présentées dans l'article de Shannon, il convient de citer l'importance du second théorème qui montre qu'une communication numérique fiable est possible via un canal de transmission bruité, en recourrant à un système d'encodage suffisamment complexe [16]. Toutefois, Shannon n'indique pas la méthode pour atteindre ce type de communication.

II.2 Contrôle des Erreurs par Codage :

Suite à la publication des articles de Shannon, une stratégie de codage basée sur la génération du signal numérique à transmettre en deux étapes s'est imposée (Figure 2.2). La première étape consiste principalement en l'élimination de la redondance de l'information dans le message ou du moins en sa réduction. Cette étape est nommée codage de source. Dans une deuxième étape, le codage de canal insère une redondance contrôlée dans le message afin de permettre la gestion des erreurs de transmission par codage (*Error Control Coding*).

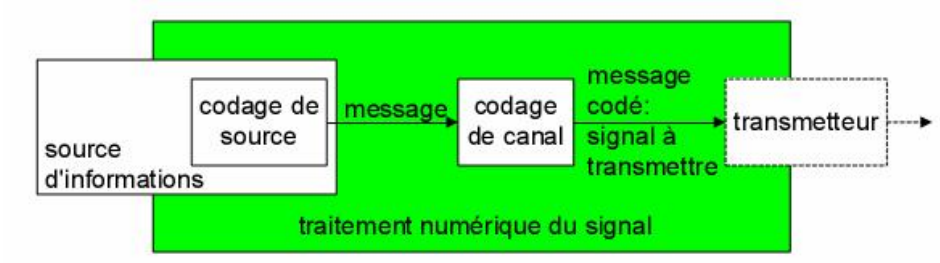


Figure 2.2 Principe du contrôle des erreurs par codage.

L'objectif du codage de canal est d'établir un système de contrôle des erreurs par un nouveau codage du message. Ceci se réalise en créant et en insérant une certaine redondance au message. Cette redondance permet au récepteur de détecter, voire de corriger les erreurs de transmission (Figure 2.3) [17]. Naturellement, la procédure de génération de la redondance doit être adaptée aux caractéristiques du support de transmission et doit être connue par le système de décodage.

Les méthodes de contrôle des erreurs se regroupent principalement en deux modes d'utilisation: le '*Automatic Repeat Request'* (**ARQ**) et le '*Forward Error Correction'* (**FEC**). L'objectif du mode ARQ est l'ajout d'une petite quantité de redondance au message, de manière à permettre la détection d'éventuelles erreurs de transmission. Dans le cas d'une détection d'erreurs, le décodeur demande la retransmission du message erroné. Par contre, dans le cas du mode FEC, la redondance introduite permet de détecter et corriger au niveau du décodeur un nombre fini d'erreurs. La quantité de redondance nécessaire est naturellement plus grande pour le mode FEC que pour le mode ARQ.

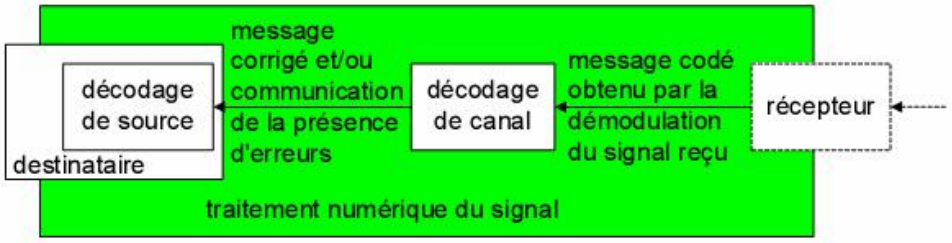


Figure 2.3 Principe du décodage d'un message protégé par codage de canal.

Le principal désavantage du mode FEC est l'utilisation constante d'une plus large bande passante, même en l'absence d'erreurs. Cette méthode s'applique surtout pour des systèmes de communication où le retard de retransmission n'est pas acceptable. Généralement, pour des systèmes ayant des taux d'erreurs raisonnables. Les coûts liés aux demandes et aux retransmissions des blocs de signaux erronés (mode ARQ) sont normalement moins importants que ceux causés par l'usage d'une bande passante plus large (mode FEC). On observe que pour des applications supportant un retard de transmission, un mode mixte (hybrid ARQ-FEC) permettant de bénéficier des avantages des deux approches est couramment utilisé. Grâce à la redondance fournie par le mode FEC, le système cherche d'éventuelles erreurs qui sont ensuite corrigées. Si le taux d'erreurs est supérieur à celui supportable par la méthode FEC, la méthode ARQ intervient en exigeant la retransmission du message [18,19].

II.3 Les Types de Codage de Canal :

II.3.1 Les Codes en Bloc Linéaires :

Le mérite de la découverte du premier code permettant la correction des erreurs est attribué à **Hamming**; en 1946, alors qu'il travaillait dans les laboratoires Bell, il était frustré par la non fiabilité des ordinateurs de ce temps-là; les ordinateurs équipés de systèmes de détection d'erreurs arrêtaient prématurément l'exécution des programmes en présence d'erreurs. Ainsi, Il chercha un moyen pour coder les données d'entrées de manière à ce que les ordinateurs puissent non seulement détecter les erreurs mais également les corriger [20].

Les premiers codes en bloc montraient des limites importantes: la correction d'une erreur unique demandait un nombre conséquent de bits supplémentaires. Cet aspect indésirable poussa Golay dans le perfectionnement de la génération de codes en bloc. Ce travail permit la définition de codes dont les capacités de correction étaient supérieures à celles des codes initiaux de Hamming [20]. A partir de ce résultat prometteur, d'autres codes ont été ensuite développés et raffinés, élargissant la gamme des codes en bloc et le nombre de contributions au domaine de la théorie du codage. En particulier [21,22]:

- **Les codes Reed-Muller ('RM') :** qui, par rapport aux codes de Hamming et Golay, permettent d'obtenir une vitesse de décodage supérieure.
- **Les codes Cycliques :** des propriétés mathématiques particulières,qu'ils sont principalement utilisés pour la détection d'erreurs.
- **Les codes Bose-Chaudhuri-Hocquenghem ('BCH') :** représentant une large classe des codes cycliques. L'utilisation des codes dans un champ de Galois 'Galois fields' *GF(q)* plus étendu *(q>2)* permet d'améliorer la qualité de protection contre les concentrations temporelles d'erreurs 'Burst errors'. L'exemple le plus représentatif est la famille de codes ***Reed Solomon***.

II.3.1.1 Principe des Codes en Bloc : Chaque famille de codes possède des caractéristiques et potentialités propres, bien qu'elles aient été développées en suivant le même principe: le regroupement des symboles d'entrée en blocs afin de leur ajouter une quantité contrôlée de redondance (Figure 2.4).

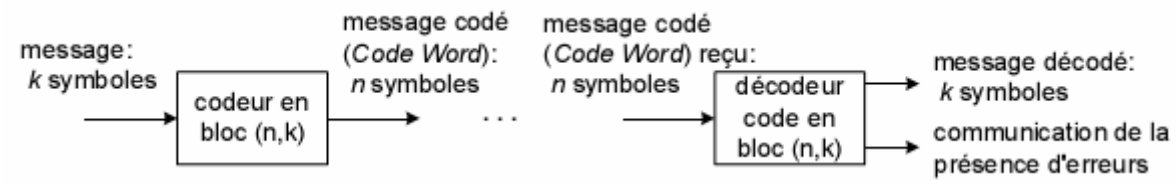


Figure 2.4 Signaux d'entrée et de sortie du codage et du décodage des codes en blocs.

En considérant le cas d'un code binaire linéaire (n,k), la stratégie commune des codes en blocs est la modification de la représentation numérique des 2^k messages possibles d'entrées. L'utilisation d'un espace de codage plus grand de celui utilisé pour représenter le message, introduit la redondance nécessaire au codage de canal (Figure 1.5). Si la transformation d'un espace de codage à l'autre se passe de manière linéaire, le code est appelé code linéaire (linear code) [23].

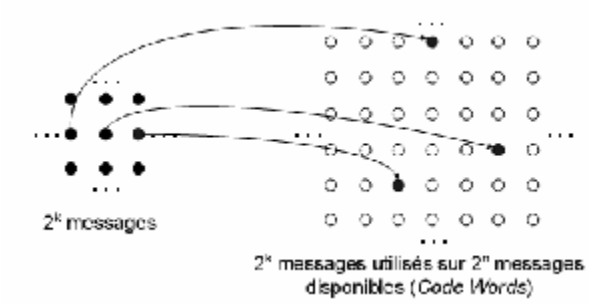


Figure 2.5 Principe de l'insertion de redondance par une nouvelle représentation du message utilisant un plus grand espace de codage.

II.3.1.2 Distance de Hamming et poids d'un message : Un important paramètre, caractérisant le code en bloc, est la distance minimale existant entre les divers messages codés (Minimal distance of the code) [21], qui se mesure par la distance de Hamming (Hamming Distance). La distance de Hamming d_{ij} , entre deux messages i et j, indique le nombre de cas où les symboles correspondants des deux messages sont différents. En utilisant cette notion, la distance minimale $d_{\min}$ du code est définie comme la distance d_{ij} la plus petite existant entre les messages générés par le code en bloc:

$$d_{\min} = \min_{i \neq j} \{d_{ij}\} \tag{2.1}$$

Dans le cas de codes en bloc linéaires, la recherche de la distance minimale $d_{\min}$ peut être simplifiée. Si les messages sont générés par des fonctions linéaires, la différence entre deux messages est également un message [21]. Soit w_i le poids (weigth) d'un message codé i, identifiant le nombre de ses éléments non-nuls, la distance minimale $d_{\min}$ peut être définie ainsi: $d_{\min} = \min_{i, i \neq i_0} \{w_i\}$ (2.2)

Où i_0 est le message codé 'zéro', (message formé uniquement de zéros). En utilisant la relation (I.4), la recherche de la distance minimale se réduit au calcul du poids de 2^{k-1} messages au lieu de déterminer $2^{k-1} \cdot (2^k - 1)$ distance de Hamming [21].

II.3.1.3 Les Codes en bloc CRC : Un code linéaire (n,k*)* est nommé code cyclique si tous les décalages cycliques (cyclic shift) d'un message codé sont également des messages codés [21,24]. Dans le domaine du codage cyclique, on associe au message codé $V = [v_{n-1} \cdots v_2 v_1 v_0]$ le polynôme$v(x)$ de degré inférieur ou égal à n-1, tel que:

$$v(x) = v_{n-1}x^{n-1} + \cdots + v_2 x^2 + v_1 x + v_0 \tag{2.3}$$

Pour les codes binaires :$v_i \in GF(2)$ pour $i = 0,1,\cdots,n-1$

La méthode de codage *CRC* se base sur la génération du message codé $v(x)$ par la multiplication polynomiale (*modulo 2* pour les codes binaires) du message $u(x)$ avec le générateur polynomial $g(x)$, donné par:

$$\begin{aligned} v(x) &= u(x)\cdot g(x) = (u_{k-1}x^{k-1} + \cdots + u_1 x + u_0)\cdot(g_{n-k}x^{n-k} + \cdots + g_1 x + g_0) \\ &= (v_{n-1}x^{n-1} + \cdots + v_2 x_2 + v_1 x_1 + v_0) \end{aligned} \tag{2.4}$$

Si le polynôme générateur$g(x)$ de degré (n-k) est un facteur du polynôme$X^n + 1$, ce polynôme génère un code en bloc cyclique (n,k) [21,24,25,26].

Etant donné le générateur polynomial$g(x)$ peut demander que le message codé $v(x)$ contienne directement le message$u(x)$, tout en respectant les caractéristiques des codes en bloc cycliques. **U**ne solution est la formation du message codé $v(x)$ par la multiplication du message $u(x)$ avec le polynômeX^{n-k}, en additionnant successivement le polynôme $b(x)$ qui garanti la divisibilité du polynôme $v(x)$ par$g(x)$.

$$v(x) = X^{n-k}\cdot u(x) + b(x) = a(x)\cdot g(x) \tag{2.5}$$

La tâche de l'encodeur systématique est ainsi de déterminer le polynôme$b(x)$, qui correspond au reste de la division du terme$X^{n-k}\cdot u(x)$ par le générateur polynomial$g(x)$, tel que:

$$b(x) = reste\left\{\frac{X^{n-k}\cdot u(x)}{g(x)}\right\} \tag{2.6}$$

Cette tâche est réalisable matériellement en utilisant un registre à décalage avec une boucle de contre-réaction utilisant des additions modulo 2.

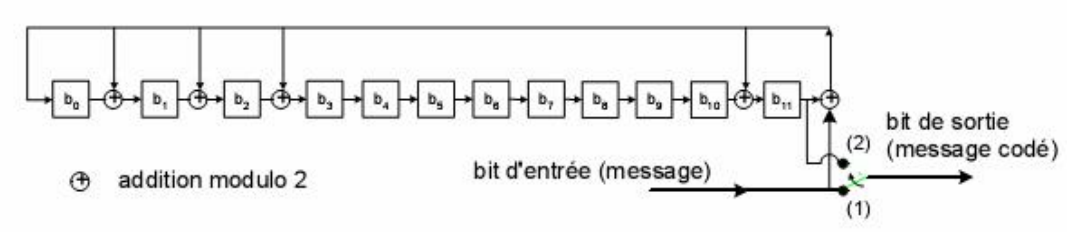


Figure 2.6 Schéma en bloc de l'encodeur systématique CRC à 12 bits avec générateur polynomial $g(x) = x^{12} + x^{11} + x^3 + x^2 + x + 1$**.**

II.3.2 Les Codes Convolutifs :

Présenté par Elias en 1955 [21], ce type de codage ajoute systématiquement de la redondance au message codé au fur et à mesure que les symboles du message (chacun formés de *b* bits) sont livrés au codeur. Le message codé se forme ainsi itérativement en utilisant un registre à décalage. Ce registre est dimensionné pour accueillir les *K* symboles les plus récents du message et la génération du message codé utilise *n* fonctions linéaires algébriques. Ces fonctions sont appelées fonctions génératrices (linear algebraic function generators). Un exemple de codeur convolutif est illustré à la Figure 2.7

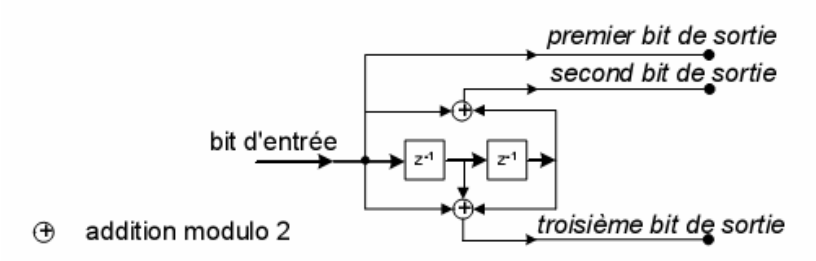


Figure 2.7 Exemple de codeur convolutif avec une longueur de contrainte K=3. Ce codeur génère 3 bits de sortie (n=3) pour chaque bit d'entrée (b=1).

A chaque instant t, le codeur de la Figure 2.7 produit trois bits de sortie en fonction des trois bits d'entrée les plus récents, selon les fonctions algébriques suivantes:

$$\begin{aligned} &Premier\,bit(t) = bit\,d\,'entrée(t) \\ &Second\,bit(t) = bit\,d\,'entrée \oplus bit\,d\,'entrée(t-2) \\ &troisième\,bit = bit\,d\,'entrée(t) \oplus bit\,d\,'entrée(t-1) \oplus bit\,d\,'entrée(t-2) \end{aligned} \qquad (2.7)$$

Le signal de sortie se forme itérativement en enchaînant les n bits de sortie du codeur convolutif (Figure 2.8) ;

Figure II.8 Représentation graphique du signal de sortie du codeur convolutif de la Figure II.7.

II.3.2.1 Principe de protection (*formation de la redondance*) : Analytiquement, le codeur convolutif protège le symbole d'entrée, en tenant compte des *K-1* derniers symboles d'entrée. Pratiquement, il s'agit d'une machine d'états, qui produit une sortie en fonction de l'entrée (nouveau symbole) et de l'état de la mémoire (formée par les *K-1* anciens symboles d'entrée). Du point de vue de la protection de canal, la redondance est générée principalement par l'augmentation de la taille du message et renforcée ensuite par la participation des derniers

K-1 symboles à la procédure de codage. Dans le cas idéal du codage d'une suite infinie de symboles, chaque symbole d'entrée prend part *K* fois à la génération des *n* bits de sortie. L'influence de chaque symbole s'étale ainsi sur un nombre important de bits de sortie ($k \cdot n$) bits. Il en résulte qu'en augmentant ces deux paramètres (la longueur de contrainte du code *K* et le nombre de bits de sortie *n*), on renforce la redondance du codage convolutif, ce qui permet de disposer d'un système de protection plus efficace contre les erreurs de transmission. Il faut aussi souligner que le changement du nombre de bits de sortie *n*, entraîne la modification du rendement du code R_c .

II.3.2.2 Représentation du déroulement du décodage : Trois méthodes sont communément utilisées pour la représentation graphique du déroulement du codage convolutif: le diagramme en arbre (Tree Diagram, Figure 2.9), le diagramme d'états (State Diagram) et le diagramme en treillis (Trellis Diagram) [21]. **E**n utilisant à titre d'exemple le code de la Figure 2.7, la Figure 2.9 illustre le début de la procédure de codage par un diagramme en arbre. Le critère à la base de cette représentation est l'illustration du comportement de la procédure de codage, à chaque instant *t*, en fonction des symboles potentiellement déjà codés. Toutefois, la répétition (persistante) de la même structure rend cette représentation rapidement trop compliquée en pratique.

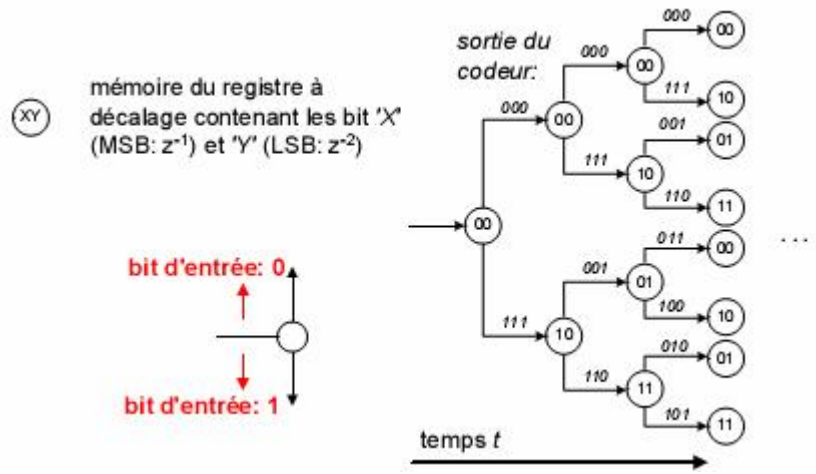


Figure 2.9 Diagramme en arbre du codeur de la Figure II.7, étant donné la remise à zéro de la mémoire du registre à décalage avant le début du codage.

II.3.3 Les Codes Enchaînés :

Si l'on désire créer un code encore plus performant, alors la concaténation de deux types de codes permet de bénéficier des avantages de chaque type de code [21]. La complémentarité des codes en bloc et Convolutifs suggère leur mise en cascade, de manière à

mieux profiter des qualités des deux familles de codes (Figure 2.10). Le codeur extérieur (*Outer Encoder*) chargé du premier codage du message, est communément un code en bloc. Un code convolutif exécute le second codage et fournit le message codé final (code intérieur, "Inner Encoder") [27]. Le décodage de cette double protection est normalement exécuté de manière séquentielle, soit décodant le message reçu, puis en vérifiant et/ou corrigeant le message obtenu.

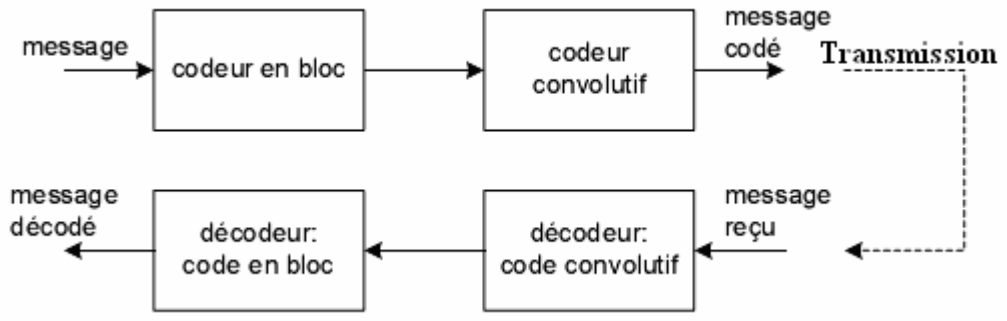


Figure 2.10 Structure classique de la concaténation entre un code en bloc et un code convolutif.

Une approche plus récente consiste en l'exécution parallèle de plusieurs opérations de codage des mêmes symboles du message. Après la transmission, les messages codés protégeant les mêmes symboles d'information montrent des distorsions non corrélées entre eux: ce codage multiple et indépendant permet au décodeur d'utiliser (itérativement) les informations de décodage de ces messages pour améliorer la qualité de protection contre les erreurs. Le codage *Turbo* utilise cette approche de codage parallèle [28]. La séquence d'informations sera codée deux (02) fois entre les deux encodeurs afin de former deux séquences data codées l'une indépendante de l'autre [14]. Il a été adopté par toutes les agences spatiales mondiales, et sera utilisé dans la transmission des données du nouveau standard de téléphonie mobile qui va succéder au GSM. Toutefois, tous les résultats concernant ces codes n'ont été établis pour le moment que de manière expérimentale. C'est pourquoi l'on se contentera de constater leur efficacité, sans pouvoir la démontrer [29].

- Les codes de **Reed–Solomon** sont des codes correcteurs d'erreurs utilisés dans tous les domaines requérant des données fiables. Typiquement, dans les communications spatiales, télévision numérique et stockage de données. Les codes RS permettent de corriger des erreurs et des effacements grâce à des symboles de contrôle ajoutés après l'information.
- Dans Cette partie on présente les différentes étapes nécessaires pour le codage/ décodage de Reed-Solomon.

II.4 définition:

Les codes de Reed-Solomon ont été développés par Reed et Solomon dans les années 50, mais avaient déjà été construits par Bush un peu avant [30]. Ces codes sont certainement les codes par blocs les plus utilisés pour la correction d'erreurs en étant présent dans les CD, les DVD et la plupart des supports de données numériques. Ils sont très utilisés car ils sont puissants du point de vue de la capacité de protection.

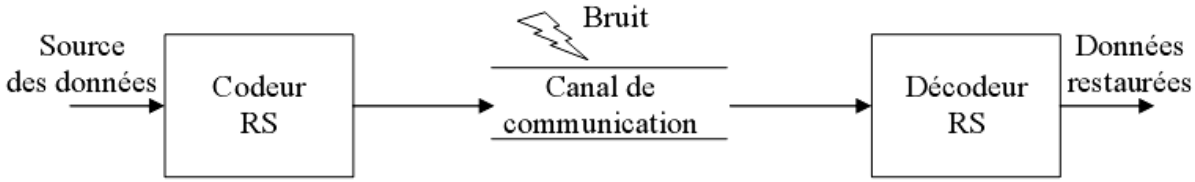


Figure 2.11 Schéma Général de la Transmission des données dans un canal.

Les messages sont divisés en blocs et on a ajouté des informations redondantes à chaque bloc permettant ainsi de diminuer la possibilité de retransmission. La longueur du bloc dépend de la capacité du codeur. La transmission des données dans un canal est effectuée ainsi (Figure 2.11) [30]. Mais avant de passer au procédures de codage/ décodage, on doit commencer d'abord par quelques définitions essentielles.

v Champs de Galois :

Les «champs de Galois» font partie d'une branche particulière des mathématiques qui modélise les fonctions du monde numérique. Ils sont très utilisés dans la cryptographie ainsi que pour la reconstruction des données comme on le verra dans le chapitre III. Un corps ou champ est un domaine d'intégrité (anneau muni d'autres caractéristiques) [31] dans lequel tous les éléments non nuls sont inversibles [32]. Il y a deux types de champs, les champs finis et les champs infinis. Les «champs de Galois» finis sont des ensembles d'éléments fermés sur eux-mêmes. L'addition et la multiplication de deux éléments du champ donnent toujours un élément du champ fini.

v **Eléments de champ de Galois :** Un « champ de Galois » consiste en un ensemble de nombres, ces nombres sont constitués à l'aide de l'élément base a comme suit :

$$0, 1, a, a^2, a^3, \ldots, a^{N-1}$$

En prenant $N = 2^m - 1$, on forme un ensemble de 2^m éléments. Le champ est alors noté $GF(2^m)$. Ce dernier est formé à partir du champ de base $GF(2)$ et contiendra des multiples des éléments simples de $GF(2)$. En additionnant les puissances de a, chaque élément du champ peut être représenté par une expressions polynomiale :

$$a^{m-1}x^{m-1} + a^{m-2}x^{m-2} + \ldots + a\,x + a^0$$

Avec : $a^{m-1}, a^{m-2}, \ldots, a^0$: Éléments base du $GF(2)$.

Sur les « champs de Galois », on peut effectuer toutes les opérations de base. L'addition dans un champ fini $GF(2)$ correspond à faire une addition modulo 2, donc l'addition de tous les éléments d'un « champ de Galois » dérivés du champ de base sera une addition modulo 2 (XOR). La soustraction effectuera la même opération qu'une addition, c'est-à-dire, la fonction logique «XOR». La multiplication et la division seront des opérations modulo «grandeur du champ», donc $\mathrm{mod}(2^m - 1)$ [33].

v **Polynôme primitif :** Ce polynôme permet de construire le « champ de Galois » souhaité. Tous les éléments non nuls du champ peuvent êtres construits en utilisant l'élément a comme racine du polynôme primitif. Chaque *m* a peut être plusieurs polynômes primitifs $p(x)$.

II.4.1 Propriétés des Codes Reed-Solomon:

Ces codes ont une propriété importante. Ils sont linéaires et font partie des codes BCH. Le codeur prend k symboles de données (chaque symbole contenant m bits) et calcule les informations de contrôle pour constituer n symboles, ce qui donne n-k symboles de contrôle. Le décodeur peut corriger au maximum t symboles, où 2t=n-k. Le diagramme ci-dessous montre une trame constituée avec le codeur RS:

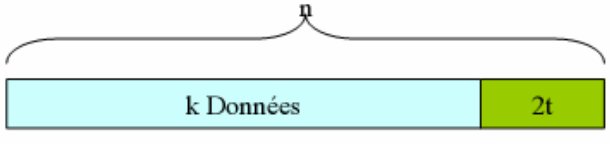


Figure 2.12 Mot-code de Reed-Solomon.

La longueur maximale d'un code RS est définie comme:

$$n = k + 2t = 2^m - 1 \quad (2.8)$$

k : nombre de symboles d'information,

2t : nombre de symboles de contrôle et

m : nombre de bits par symbole.

La distance minimale d'un code Reed-Solomon est:

$$d_{\min} = 2t + 1 \quad (2.9)$$

II.4.2 Le Codage:

L'équation définissant le codage systématique de RS (n, k) est:

$$c(x) = i(x)x^{n-k} + \left[i(x).x^{n-k} \right] \bmod(g(x))$$

(2,10)

C(x): polynôme du Mot-code, degré n-1,

I(x): polynôme d'information, degré k-1, et

$\left[i(x).x^{n-k} \right] \bmod(g(x))$: Polynôme de contrôle de degré n-k-1

Le codage systématique signifie que l'information est codée dans le degré élevé du mot-code et que les symboles de contrôles sont introduits après les mots d'informations.

II.4.2.1 Le Polynôme Générateur:

Les symboles de contrôles sont générés à l'aide des polynômes particuliers, appelés polynômes générateurs. Tous les codes RS sont valables si et seulement si ils sont divisibles par leur polynôme générateur, c(x) doit être divisible par g(x) [30].

Pour la génération d'un correcteur d'erreurs de t symboles, on devrait avoir un polynôme générateur de puissance a^{2t}. La puissance maximale du polynôme est déterminée grâce à la distance minimale qui est $d_{min}=2t+1$. On devrait avoir 2t+1 termes du polynôme générateur.

Le polynôme générateur est sous la forme:

$$g(x) = (x - a)(x - a^2)....(x - a^{2t}) \quad (2.11)$$

$$g(x) = g_{2t}x^{2t} + g_{2t-1}x^{2t-1} + ... + g_1x + g_0 \quad (2.12)$$

II.4.2.2 Implémentation physique du codeur:

Pour comprendre comment la partie hardware fonctionne, on doit comprendre la définition mathématique du codage et les différentes opérations effectuées. Le codage est systématique, on doit effectuer une opération de décalage pour placer les informations dans le

degré élevé du mot-code de sortie. Mathématiquement le décalage est effectué selon la fonction:

$$i(x)x^{n-k} \tag{2.13}$$

i(x) : polynôme d'information.

x^{n-k} : décalage du polynôme d'information de n-k positions vers la gauche. Le deuxième terme de l'équation (2,10) est le reste de la division de $\frac{i(x)x^{n-k}}{g(x)}$ .cette division donnera les symboles de contrôle. L'implémentation du codeur demandera deux opérations: un décalage et une division. Ces deux opérations peuvent êtres effectuées grâce à des registres à décalage et à des multiplieurs [30].

II.4.2.3 Schéma: Schéma général du codage:

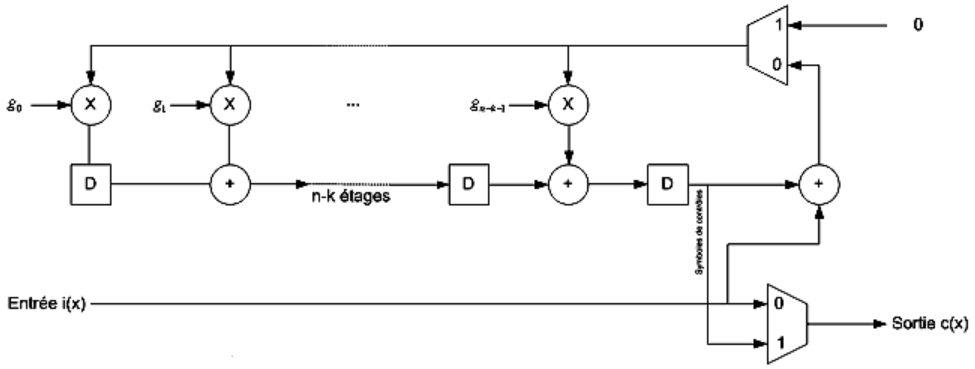


Figure 2.13 Schéma de Codage.

Les éléments utilisés pour le codage sont:

- Additionneurs dans le champ de Galois,
- Multiplieurs dans le champ de Galois,
- Multiplexeurs dans le champ de Galois,
- Registres à m bits.

L'Addition : Les additions sont définies dans le champ de Galois, $GF(2^m)$, donc pour additionner deux éléments, on prendra la notation binaire de chaque élément et on les additionnera modulo2. L'addition modulo2 est une opération logique définie par l'opérateur logique « XOR » bit à bit.

La Multiplication : Les multiplications utilisées dans les codes de Reed-Solomon sont des multiplications dans le champ de Galois $GF(2^m)$, La multiplication dans le champ de Galois est une opération modulaire, c'est-à-dire que la multiplication entre deux éléments d'un champ fini donnera toujours un élément dans le même champ.

II.4.3 Le Décodage:

L'idée de base du décodeur RS est de détecter une séquence erronée avec peu de termes, qui est sommée aux données reçues, donne lieu à un mot-code valable.

Plusieurs étapes sont nécessaires pour le décodage de ces codes:

- **F** Calcul du syndrome,
- **F** Calcul des polynômes de localisation des erreurs et d'amplitudes,
- **F** Calcul des racines, évaluations des deux polynômes, et
- **F** Sommation du polynôme constitué et du polynôme reçu pour reconstituer l'information de départ sans erreur [30].

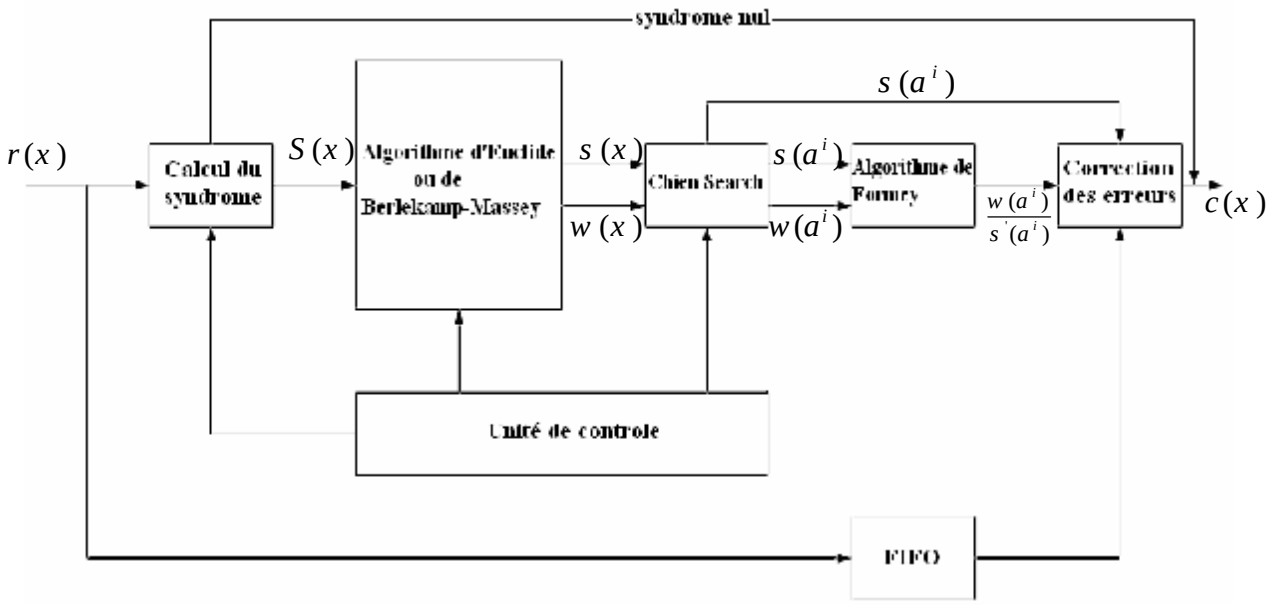


Figure 2.14 Schéma du décodage

$r(x)$: Mot-code reçu

$S(x)$: Syndrome calculé

$w(x)$: Polynôme d'amplitude des erreurs

$w(a^i)$: Polynôme d'amplitude des erreurs, pour tous les éléments compris dans $GF(2^m)$

$s(x)$: Polynôme de localisation des erreurs

$s(a^i)$: Polynôme de localisation des erreurs, pour tous les éléments dans $GF(2^m)$

$s'(a^i)$: Dérivée du $s(a^i)$

$\frac{w(a^i)}{s'(a^i)}$: Division entre le polynôme d'amplitude et $s'(a^i)$, et $c(x)$: Mot-code reconstitué

II.4.3.1 Généralité du décodage :

Considérons un code de Reed-Solomon $c(x)$ correspond au code transmis et soit $r(x)$ le code que l'on reçoit. Le polynome d'erreurs introduit par le canal est défini comme :

$$e(x) = r(x) - c(x) \quad = r(x) + c(x)$$
$$= e_0 + e_1 x + ... + e_{n-1} x^{n-1} \tag{2.14}$$

Supposant que le polynome des erreurs contienne v erreurs aux positions $x^{j_1}, x^{j_2}, ..., x^{j_v}$ avec $0 \le j_1 < j_2 < ... < j_v \le n-1$. On peut donc redéfinir le polynome des erreurs comme :

$$e(x) = e_{j_1} x^{j_1} + e_{j_2} x^{j_2} + ... + e_{j_v} x^{j_v} \tag{2.15}$$

Avec :

$e_{j_1}, e_{j_2}, ...e_{j_v}$: Valeurs d'amplitude des erreurs

$x^{j_1}, x^{j_2}, ..., x^{j_v}$: Emplacements des erreurs

A partir du polynôme $r(x)$ reçu, on peut calculer le polynôme du syndrome $S(x)$ qui nous indiquera la présence d'éventuelles erreurs. Si tous les coefficients du syndrome sont nuls, alors les étapes suivantes du décodage n'ont pas lieu d'être car le mot-code reçu ne contiendra pas d'erreurs. Par contre, si le syndrome est non nul, on devra calculer le polynôme de localisation des erreurs et le polynôme d'amplitude des erreurs. Il y a plusieurs méthodes de calcul de ces deux polynômes, dans le cadre de ce projet on ne traitera qu'une seule méthode, le décodage selon l'algorithme d'Euclide. Une fois les polynômes calculés en utilisant l'algorithme de Forney, on calculera les valeurs à soustraire pour obtenir le mot-code sans erreur.

II.4.3.2 Calcul du Syndrome :

Le calcul du syndrome est défini comme le reste de la division entre le polynôme reçu $r(x)$ et le polynôme générateur $g(x)$. Le reste indiquera la présence d'erreurs. Comme l'opération division est toujours une opération complexe par rapport à des sommes et des additions, on est amené à chercher une autre méthode pour le calcul du syndrome [33]. Le calcul du syndrome peut aussi être effectué par un processus itératif. Avant de pouvoir calculer le polynôme du syndrome, on doit attendre que l'on ait reçu tous les éléments du polynôme $r(x)$. Comme :

$$S_i = r(a^i) = c(a^i) + e(a^i) = e(a^i) \tag{2.16}$$

A partir de cette relation on peut définir les différentes équations :

$$\begin{aligned} S_1 &= e_{j_1} a^{j_1} + e_{j_2} a^{j_2} + ... + e_{j_v} a^{j_v} \\ S_2 &= e_{j_1} a^{2j_1} + e_{j_2} a^{2j_2} + ... + e_{j_v} a^{2j_v} \\ &... \\ S_{2t} &= e_{j_1} a^{2tj_1} + e_{j_2} a^{2tj_2} + ... + e_{j_v} a^{2tj_v} \end{aligned} \quad (2.17)$$

Le syndrome sous forme polynomiale sera :

$$S(x) = ... + S_{2t+1} x^{2t} + S_{2t} x^{2t-1} + ... + S_2 x + S_1 \quad (2.18)$$

Seuls les premiers $2t$ symboles du syndrome sont connus. Si le code $r(x)$ n'est pas affecté par des erreurs alors tous les cœfficients du syndrome seront nuls ($r(x) = c(x)$).

Schéma du calcul du syndrome : Le schéma ci-dessous calcule de façon itérative :

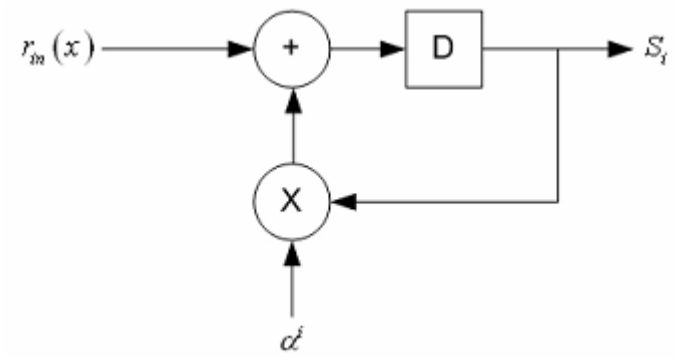


Figure 2.15 Schéma pour le calcul du syndrome

II.4.3.3 Algorithme d'Euclide :

II.4.3.3.1 Généralités du théorème d'Euclide :

L'algorithme d'Euclide [35] est un algorithme récursif qui permet de trouver le plus grand diviseur commun de deux polynômes $r_0(x)$ et $r_1(x)$ dans le champ de Galois $GF(q)$. Il existe deux polynômes $a(x)$ et $b(x)$ en $GF(q)$ tels que :

$$MCD(r_0(x), r_1(x)) = a(x) r_0(x) + b(x) r_1(x) \quad (2.19)$$

$a(x)$ et $b(x)$ peuvent êtres calculés selon l'algorithme d'Euclide.

En donnant deux polynôme non nuls $a(x)$ et $b(x)$ en $GF(q)$, l'algorithme d'Euclide fonctionne de la façon suivante :

$$\begin{aligned} &\deg(r_1(x)) \le \deg(r_0(x)) \\ &a_0(x) = 1, b_0(x) = 0 \\ &a_1(x) = 0, b_1(x) = 1 \end{aligned} \quad (2.20)$$

Avec : $\deg(r_1(x))$: Degré du polynôme $r_1(x)$ et $\deg(r_0(x))$: Degré du polynôme $r_0(x)$

Pour $i \geq 2$, on calcule le quotient $q_1(x)$ et le polynôme restant $r_i(x)$, a division est effectuée sur $r_{i-2}(x)$ et $r_{i-1}(x)$

$$r_{i-2}(x) = q_i(x) r_{i-1}(x) + r_i(x) \tag{2.21}$$

Avec,

$$0 \leq \deg(r_i(x)) < \deg(r_{i-1}(x))$$
$$a_i(x) = a_{i-2}(x) - q_i(x) a_{i-1}(x)$$
$$b_i(x) = b_{i-2}(x) - q_i(x) b_{i-1}(x)$$

Les calculs se terminent lorsque; $\deg(r_i) = 0$ et le dernier polynôme non nul indique le plus grand diviseur commun.

II.4.3.3.2 Correction d'erreurs avec Euclide [33-35]:

Le polynôme de localisation d'erreurs est défini comme :

$$s(x) = \prod_{k=1}^{v} (1 - a^{i_k} x)$$

$$= s_v x^v + s_{v-1} x^{v-1} + \ldots + s_1 x + 1 \tag{2.22}$$

Ø Le polynôme d'amplitude des erreurs se calculera de la façon suivante :

$$w(x) = S(x) s(x) \tag{2.23}$$

$s(x)$: Polynôme de localisation des erreurs, inconnu à ce stade,

$w(x)$: Polynôme d'amplitude, inconnu à ce stade,

$S(x)$: Polynôme syndrome, connu.

Comme on connaît seulement $2t$ symboles du polynôme du syndrome $(x^0 \ldots x^{2t-1})$, on devrait limiter le résultat à $2t$:

$$S(x) s(x) = w(x) \bmod(x^{2t}) \tag{2.24}$$

Cette expression est l'équation clé pour les codes de Reed-Solomon. Si le nombre d'erreurs v dans le mot-code transmis $c(x)$ est plus petit ou égale à t, l'équation clé a une seule paire de solution $s(x)$ et $w(x)$. Les deux degrés des polynômes doivent respecter la contrainte qui suit :

$$\deg(w(x)) < \deg(s(x)) \leq t \tag{2.25}$$

L'équation clé peut être résolue selon l'algorithme d'Euclide en appliquant $r_0(x) = x^{2t}$ et $r_1(x) = S(x)$. Le calcul du théorème d'Euclide nous donnera comme solution le polynôme de localisation des erreurs et le polynôme d'amplitude. Figure 2.16

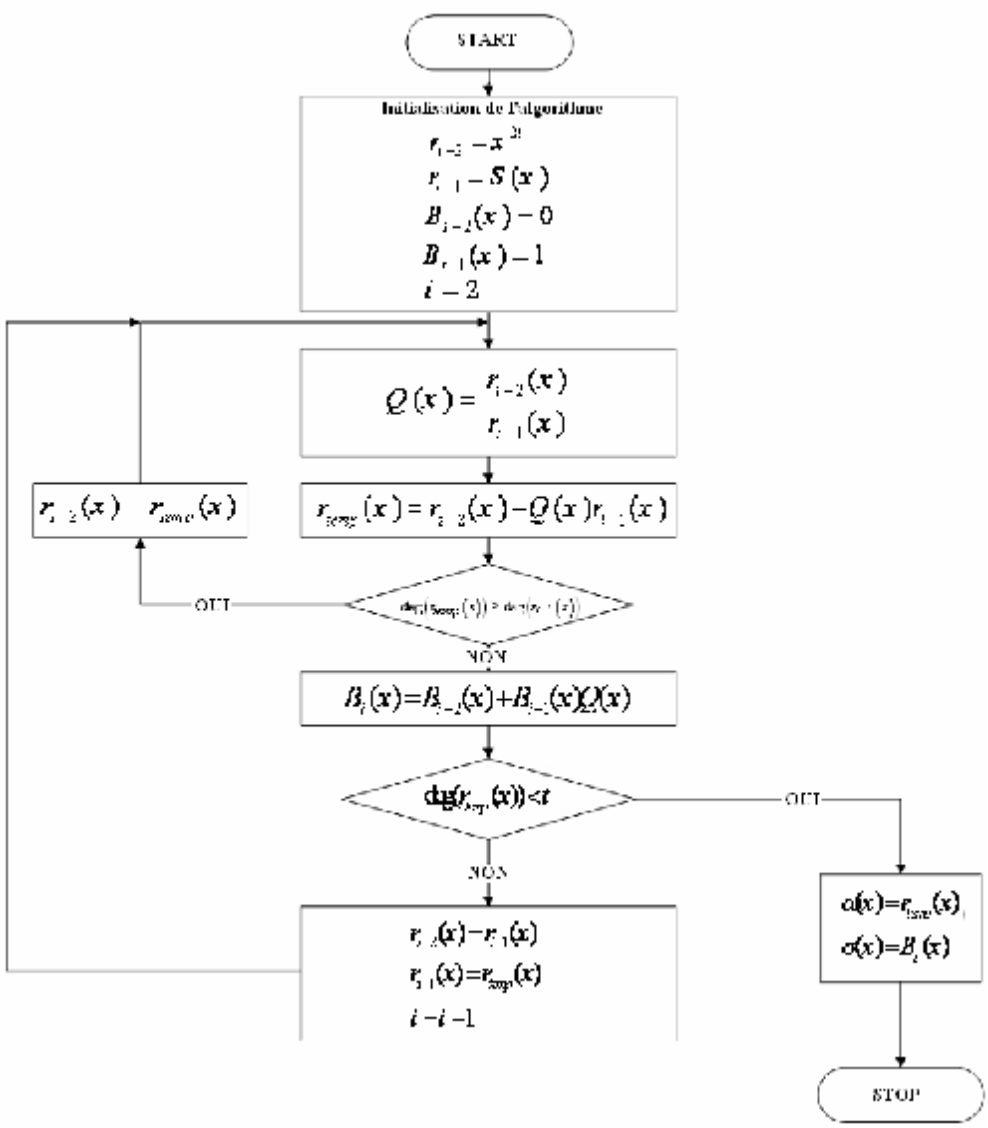


Figure 2.16 Algorithme d'Euclide pour le calcul du polynôme de localisation et pour le polynôme d'amplitude

Le dernier reste de la division nous donnera le polynôme d'amplitude. Le polynôme de localisation des erreurs est donné selon la relation :

$$s_i(x) = s_{i-2}(x) + s_{i-1}(x) Q_i(x) \tag{2.26}$$

Tel que ; $s_i(x) = B_i(x)$

La théorie montre que l'on est obligé d'avoir deux blocs dans l'implémentation hardware.

Un bloc qui effectue la division et qui donnera le polynôme d'amplitude des erreurs, et également un bloc de multiplication qui donnera le polynôme de localisation des erreurs.

II.4.3.4 Chien Search :

Une fois le polynôme de localisation des erreurs calculé, on doit évaluer ses racines et sa dérivée. L'évaluation des racines est effectuée avec l'algorithme appelé « Chien Search »

qui est du type « brute force », c'est-à-dire, qu'il évalue toutes les possibilités. A la sortie de ce bloc, on obtiendra une séquence de symboles. Lorsque les symboles sont nuls, ceux-ci nous indiqueront qu'une racine a été détectée.

II.4.3.5 L'Algorithme de Forney :

Cet algorithme permet de construire le polynôme d'erreurs $e(x)$ à additionner avec le polynôme reçu $r(x)$ pour reconstituer le polynôme $c(x)$. Pour le calcul du polynôme $e(x)$, les polynômes $s(a^i), s'(a^i), w(a^i)$ sont nécessaires. Le polynôme de localisation des erreurs et sa dérivée sont déjà évalués pour les différentes valeurs de α, il nous reste à évaluer $w(a^i)$. Une fois les différentes valeurs de $w(a^i)$ calculées, on applique l'algorithme de Forney. Cet algorithme est défini comme :

$$e_i = \frac{w(a^i)}{s'(a^i)} \tag{2.27}$$

$w(a^i)$: Polynôme d'amplitude évalué pour les valeurs de $GF(2^4)$.

$s'(a^i)$: Dérivée du polynôme de localisation des erreurs pour les valeurs de $GF(2^4)$

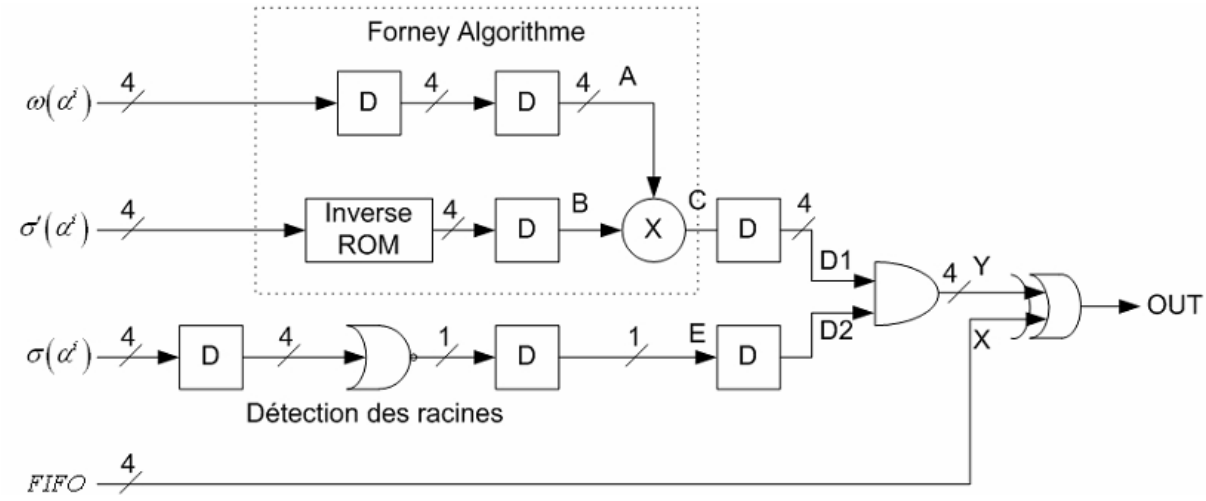


Figure 2.17 Schéma de l'algorithme de Forney.

Remarque : Les codes de Reed–Solomon sont non seulement utilisés pour la correction des erreurs, mais permettent aussi de corriger les effacements. Un effacement suit le même principe que lorsqu'on efface une lettre dans un mot à l'aide d'un effaceur. La lettre effacée dans le mot n'est pas connue, mais la position de celle-ci l'est. Les codes de Reed – Solomon permettent de corriger deux fois plus d'effacements que d'erreurs. La séquence de décodage est presque la même que celle utilisée pour la correction des erreurs, la seule différence est

qu'avant de calculer les syndromes, on doit substituer dans le polynôme reçu $r(x)$ les effacements avec des '0' avant de procéder au calcul du syndrome lui même. La première opération à effectuer pour le décodage des erreurs et des effacements, est l'évaluation du polynôme de localisation des effacements.

- Ø **Note :** La capacité de correction dans ce cas sera plus grande qu'avant « sans effacements 'erasures' » car on devrait ajouter f effacements aux v erreurs.

II.4.3.6 Résumé des opérations de décodage selon Euclide [34]:

On peut résumer les étapes précédentes comme suit :

1. Evaluation du polynôme des effacements
2. Modification du polynôme reçu $r(x)$ en substituant les valeurs effacées par des zéros et calculer le syndrome,
3. Evaluation du syndrome modifié $y(x) = [S(x)b(x)] \bmod (x^{2t})$
4. Application du théorème d'Euclide avec $r_0(x) = x^{2t}$ et $r_1(x) = y(x)$ pour le calcul du Polynôme de localisation des erreurs $s(x)$ et pour le polynôme d'amplitude $w(x)$
5. Calcul des racines selon « Chien Search » du polynôme des erreurs et des effacements $g(x) = s(x)b(x)$
6. Evaluation des amplitudes des erreurs et des effacements associés avec le polynôme de localisation des erreurs et des effacements. Ce calcul donnera le polynôme d'erreurs $e(x)$
7. Soustraction du polynôme $e(x)$ au polynôme reçu $r(x)$ pour la correction des erreurs et des effacements.

II.4.4 Applications : Les principaux domaines d'utilisations des codes RS sont [30]:

- Dans la communication mobile et les réseaux sans fils (wireless, etc…),
- Dans les communications satellitaires,
- Dans la télévision numérique et la radio diffusion numérique (DVB),
- Dans les modems ADSL et VDSL,
- Dans la sauvegarde de données (sauvegarde magnétique, optique, etc…).

II.5 Conclusion :

Ce chapitre a brièvement introduit les notions fondamentales de la protection des données contre les erreurs de communication. A partir des contributions pionnières de Shannon, les deux modes principaux de contrôle des erreurs FEC et ARQ ont été présentés. Puis, ce chapitre a décrit les principes ainsi que les propriétés du codage en bloc, codage convolutif et brièvement le codage enchaîné. Les codes de Reed–Solomon sont aujourd'hui dans toute application multimédia et dans toute application nécessitant une correction d'erreurs. Selon l'application on aura différents types de codes utilisés, le dernier paragraphe de ce chapitre nous a permis d'expliquer les codes de Reed-Solomon, en décrivant les étapes essentielles durant les deux opérations (codage et décodage), ces étapes seront les éléments de base pour le chapitre suivant, où on exploitera les avantages des codes RS pour pouvoir améliorer la transmission dans les protocoles aléatoires.

III.1 Introduction:

Le protocole de contrôle à accès médium (MAC) est un élément essentiel pour les réseaux de communication sans fils, alors qu'il y a deux grandes techniques d'accès aléatoire; Aloha et ses dérivées et CSMA, (pour accès multiple avec détection de la porteuse), et ses dérivées que la plupart de ces réseaux commencent à les utiliser. L'inconvénient majeur de ces protocoles de base est le faible débit, à cause des phénomènes de collision des paquets; quand plus qu'un utilisateur envoient des paquets dans le même temps, alors leurs paquets vont se heurtes et aucun d'eux ne peut être bien reçu. Les protocoles d'accès aléatoire comme Aloha et CSMA souffrent de la collision des paquets qui influe sur la fiabilité du système et conduit ainsi à un affaiblissement grave de débit, ceci a été le sujet de recherches pendant ces dernières années, où plusieurs chercheurs ont essayé de trouver de meilleures solutions pour ce système [37]. Dans ce cadre, nous proposons dans ce chapitre d'intégrer le codage Erasure avec les protocoles MAC afin de récupérer les paquets heurtés et améliorer ainsi la performance de débit. Pour démontrer l'efficacité de cette approche, on a commencé par combiner la technique Erasure avec les protocoles d'accès aléatoire Slotted Aloha et Slotted non-persistent CSMA respectivement, après on a évalué le protocole résultant par un modèle analytique suivi par une simulation sur Matlab. Cette dernière a donné des résultats similaires avec celles d'autres analyses, avec cependant une amélioration du débit de transmission.

Avec le codage Erasure, un nombre K de paquets originaux sont codés en (N,K) mots erasures, qui consiste en N (>K) paquets codés. À la réception, on peut récupérer un nombre K de paquets originaux si on reçoit un nombre K entre N paquets codés. Alors le codage Erasure augmente la charge offerte du trafic, et la probabilité de succès des paquets de transmission peut être aussi augmentée si ce codage est bien implémenté [37]. On va commencer par expliquer le codage Erasure, en donnant deux exemples, par la suite on intègre le codage Erasure dans le (Slotted Aloha) ainsi que dans la technique CSMA respectivement.

III.2 Le Principe du Codage Erasure:

Le but du codage Erasure est de récupérer les paquets perdus, si la position de ces derniers est connue. Dans (N,K) codes erasures, un (N,K) mots de codes consiste en N paquets codés, lorsqu'il y a K paquets originaux , on peut considérer que (N-K)

sont des paquets redondants. Les K paquets originaux peuvent êtres récupérés avec succès si on reçoit K paquets à partir de N paquets codés (Fig 3. 3). Les (N-K) paquets redondants sont générés suivant certaines fonctions. Il est bien connu que le code Reed-Solomon est un bon candidat pour le codage Erasure [38]. On donne un exemple de génération de deux paquets redondants.

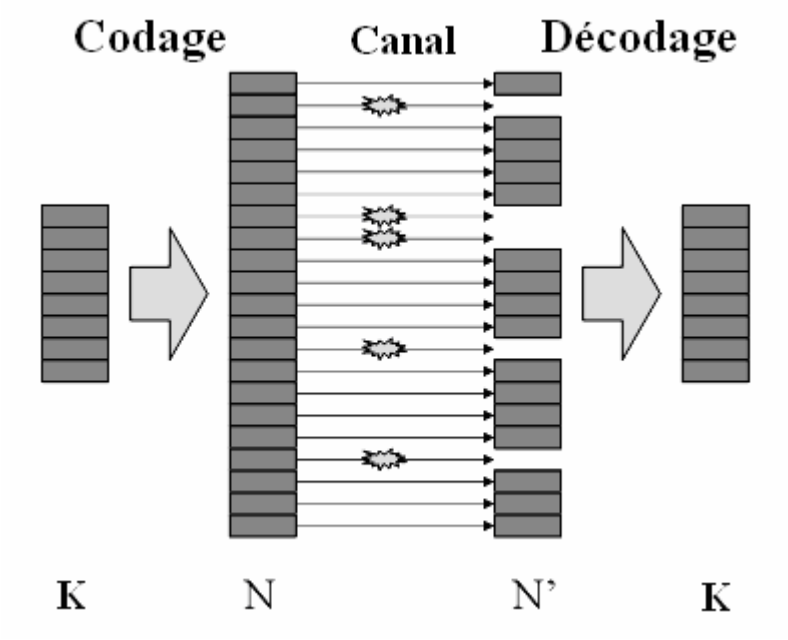


Figure 3.1 Mécanisme du codage Erasure.

Ø On suppose qu'on a K paquets originaux, de 'm' bits chaqu'un. Le premier paquet redondant est généré par l'équation suivante [39]:

$$C_{k+1,i} = \left(\sum_{j=1}^{k} C_{j,i} \right) \bmod 2 \tag{3.1}$$

Où $C_{i,j}$ est le i^{ème} bit du j^{ème} paquet.

Le deuxième redondant paquet peut être généré selon l'équation suivante:

$$C_{k+2,i} = \begin{cases} \left(\sum_{j=1}^{i} C_{j,i+1-j} \right) \bmod 2, \ldots\ldots\ldots\ldots\ldots 1 \le i \le k+1 \\ \left(\sum_{j=1}^{k+1} C_{j,i+1-j} \right) \bmod 2, \ldots\ldots\ldots\ldots\ldots k+2 \le i \le m \\ \left(\sum_{j=1}^{k+m-i+1} C_{i-m+j,m+1-j} \right) \bmod 2, \ldots\ldots\ldots m+1 \le i \le m+k \end{cases} \tag{3.2}$$

* On doit noter ici que la taille du deuxième paquet redondant est m+k, cependant la taille entre paquets peut être ignorée lorsque m>>k. Dans ce chapitre on suppose que tous les paquets ont la même taille pour simplifier les calculs. Cela est vérifié dans [39] où il est montré que c'est possible de récupérer K paquets originaux si K paquets parmi K+2 paquets codés sont reçus.

Ø Dans ce qui suit, on donne un exemple qui utilise le codage montré précédent, en supposant que K=3 et m=5. Le premier paquet redondant peut être facilement construit à l'aide des équations précédentes. On montre comment générer le second paquet redondant, huit bits du deuxième paquet redondant peuvent êtres construits comme suit:

$C_{5,1} = C_{1,1}$,

$C_{5,2} = C_{1,2} + C_{2,1}$,

$C_{5,3} = C_{1,3} + C_{2,2} + C_{3,1}$,

$C_{5,4} = C_{1,4} + C_{2,3} + C_{3,2} + C_{4,1}$,

$C_{5,5} = C_{1,5} + C_{2,4} + C_{3,3} + C_{4,2}$,

$C_{5,5} = C_{1,5} + C_{2,4} + C_{3,3} + C_{4,2}$,

$C_{5,6} = C_{2,5} + C_{3,4} + C_{4,3}$,

$C_{5,7} = C_{3,5} + C_{4,4}$,

$C_{5,8} = C_{4,5}$.

Figure 3.2 Génération du second paquet redondant.

III.3 Slotted-Aloha avec le Codage Erasure:

III.3.1 Description du Schéma:

La technique Slotted Aloha est très populaire dans les réseaux mobiles et satellitaires, ceci est démontré dans [40,41], l'utilisation de la technique Multi-Copy Aloha peut améliorer la performance du débit du system Slotted Aloha, par l'envoie de multiples copies du paquet lorsque le trafic du système est inférieur à 0.48. On peut traiter Multi-Copy Aloha comme un schéma de (m, 1) codes erasures, où chaque paquet est codé en un mot de m copies de paquets, le paquet original est reçu avec succès, si l'une des m copies est correctement reçue, notant que ce schéma va

augmenter 'm' fois le débit du système. Nous proposons d'adopter plus de (N,K) schémas de codage, par exemple les codes de Reed-Solomon, pour améliorer la performance du Slotted Aloha. À l'arrivée des K paquets, il y a (N,K) mots de code erasures. Lorsqu'il y a k paquets de N paquets reçus, on a K paquets originaux qui peuvent êtres récupérer. Cela permet d'atteindre une tolérance de (N-K) des paquets perdus ou bruités. Le débit du système augmente de N/K fois, ce qui rend notre système plus performant que le Multi Copies Aloha.

III.3.2 Procédure de Fonctionnement :

Le schéma que nous proposons fonctionne comme suit [40,42,43,44]:

1. On code les K paquets en un (N,K) mots de code erasures, où on assigne un identificateur pour chaque paquet que le récepteur utilisera pour l'identification des paquets.

2. Les N paquets codés sont transmis aléatoirement et indépendamment dans les M time-slots prochains (M généralement plus grand que N). Pour chaque paquet original envoyé, le temporisateur est activé (Si l'information de ce paquet n'a pas été reçue avant que le temporisateur expire, ce paquet sera retransmis).

3. Au récepteur, seulement les paquets correctement reçus (en incluant ceux récupérés avec succès par le codage Erasure), qui sont reconnus. Les paquets non reconnus dans le temps sont considérés comme perdus, il seront retransmis suivant le schéma du système Slotted Aloha.

III.3.3 Modèle Analytique:

Construisons, maintenant le modèle analytique pour l'étude de la performance du débit du Slotted Aloha avec le codage Erasure, on suppose que:

- Le trafic et les data retransmises constituent un processus de Poisson de moyenne **λ** paquets par time-slot (***P/Ts***).
- Le comportement de la transmission des données peut conduire à l'équilibre.
- Le schéma des (N,K) codes erasures est employé, i.e., pour un groupe de K paquets originaux on ajoute (N-K) paquets redondants.

Après la procédure de codage, le trafic actuel a un débit de Nl/K comme dans le Slotted Aloha conventionnel, la probabilité $\boldsymbol{P}$ pour qu'un paquet codé soit reçu avec succès sans collisions est:

$$p = e^{-Nl/K} \tag{3.3}$$

La probabilité P_K pour qu'un nombre K des paquets codés soit reçus avec succès est:

$$P_K = \sum_{i=K}^{N} \binom{N}{i} p^i (1-p)^{N-i}$$

(3.4)

On pose $P_{n,m}$ la probabilité qu'un n (n<k) paquets codés soient reçus, sachant que m parmi n paquets reçus sont des paquets originaux, on aura:

$$P_{n,m} = \binom{k}{m}\binom{N-k}{n-m} p^n (1-p)^{N-n}$$

(3.5)

* On peut voir qu'un paquet original peut être reçu avec succès dans les deux conditions:

- Au moins k, parmi N paquets codés en code Erasure qui contiennent le paquet original, reçu avec succès
- On reçoit seulement n parmi N paquets codés, où n>k et m entre n paquets reçus sont des paquets originaux, et le paquet original qui est en considération est l'un des m paquets (avec la probabilité m/k).

Donc la probabilité p_S qu'un paquet original est reçu avec succès (ou récupéré) est égale à:

$$P_s = P_k + \sum_{n=1}^{k-1}\sum_{m=1}^{n} \frac{m}{k} P_{n,m} \tag{3.6}$$

Le débit S du système est donc donné par l'équation:

$$S = l.P_s \tag{3.7}$$

* Quand $K = 1$, notre système devient Multi-Copy Aloha, et le débit sera $S_{multi} = l.P_1$ quand $P_1 = P_{k(k=1)}$. Lorsque $N = K = 1$, le système devient semblable à celui du Slotted Aloha, et le débit sera égal à $S = l.P_s = l.e^{-l}$.

Charge du trafic λ	Erasure (N, K)	Charge du trafic λ	Multi-copy $(N, 1)$
0.01–0.07	(20,5)	0.01–0.12	(6,1)
0.08–0.16	(20,6)	0.13–0.15	(5,1)
0.17–0.26	(20,7)	0.16–0.19	(4,1)
0.27–0.35	(20,8)	0.2–0.28	(3,1)
0.36–0.48	(2,1)	0.29–0.48	(2,1)
0.49–	(1,1)	0.49–	(1,1)

Tableau 3.1 Paramètres utilisés pour les différentes charges du trafic dans S.Aloha

Le **Tableau 3.1** montre les paramètres qu'on a utilisé pour ces résultats numériques, on voit clairement que lorsque la charge du trafic λ augmente, il y a le taux de codage qui augmente aussi; cela signifie que la partie du trafic superflu est réduite pour éviter que la probabilité de collision empire. Il est aussi facile de vérifier que, quand on donne les mêmes taux (N/k), le débit s'élève lorsque N augmente (et k augmente après), en raison du mécanisme de récupération des paquets du codage Erasure. Cependant l'amélioration de débit sera négligeable lorsque N est assez grand.

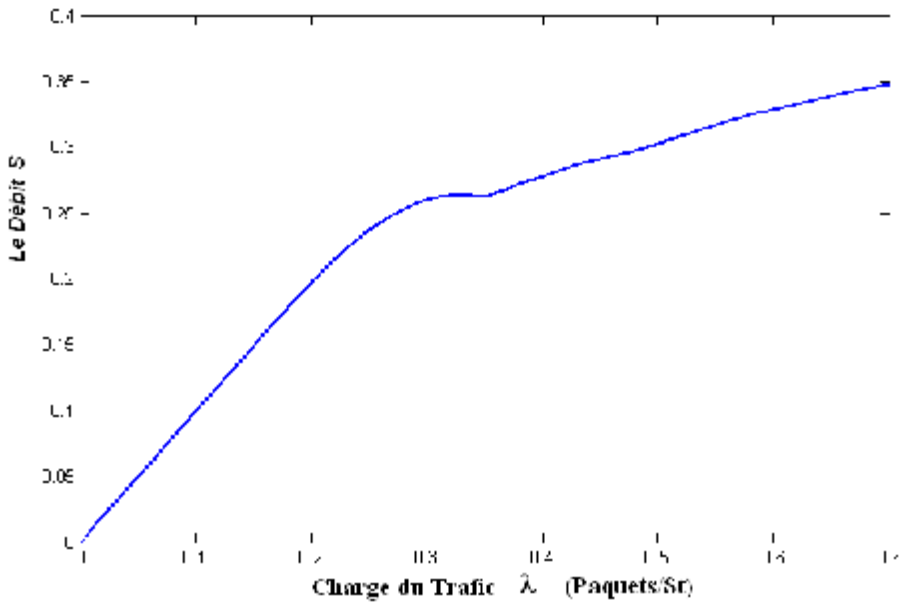


Figure 3.3 Le Débit du Slotted Aloha avec le Codage Erasure.

La **Figure 3.5** montre la performance du system obtenu, qu'il sera après comparé avec celui du S.Aloha, et M-C.Aloha (**Figure 3.6**).

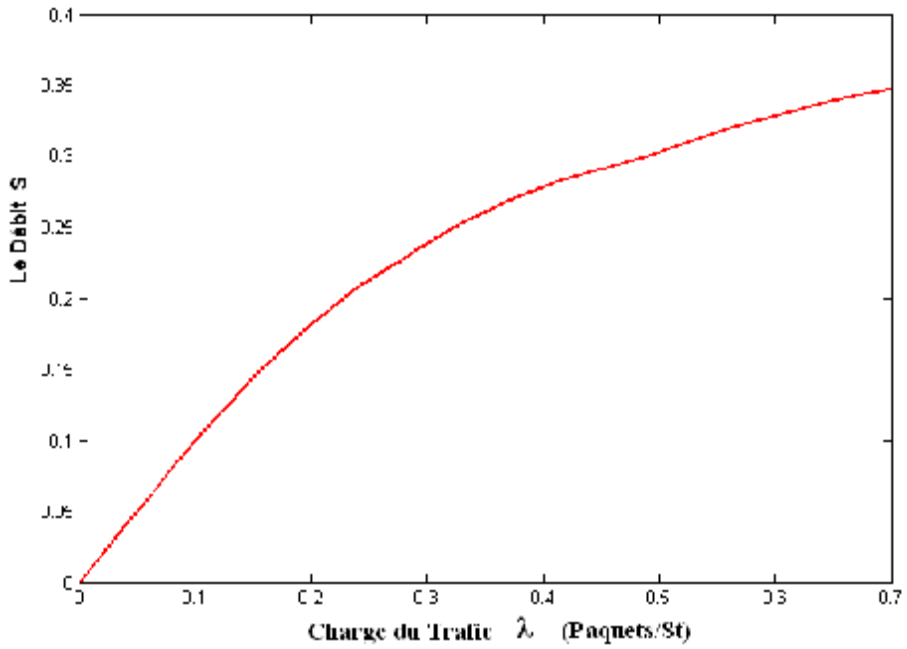


Figure 3.4 Le Débit de Multi-Copy Aloha avec le Codage Erasure.

Il est clair que, lorsque la charge du trafic est entre 0.1 et 0.35, notre courbe ressemble à celui du Multi Copy Aloha. Quand la charge du trafic est très faible les trois courbe sont semblables, comme la probabilité de la collision est aussi très faible.

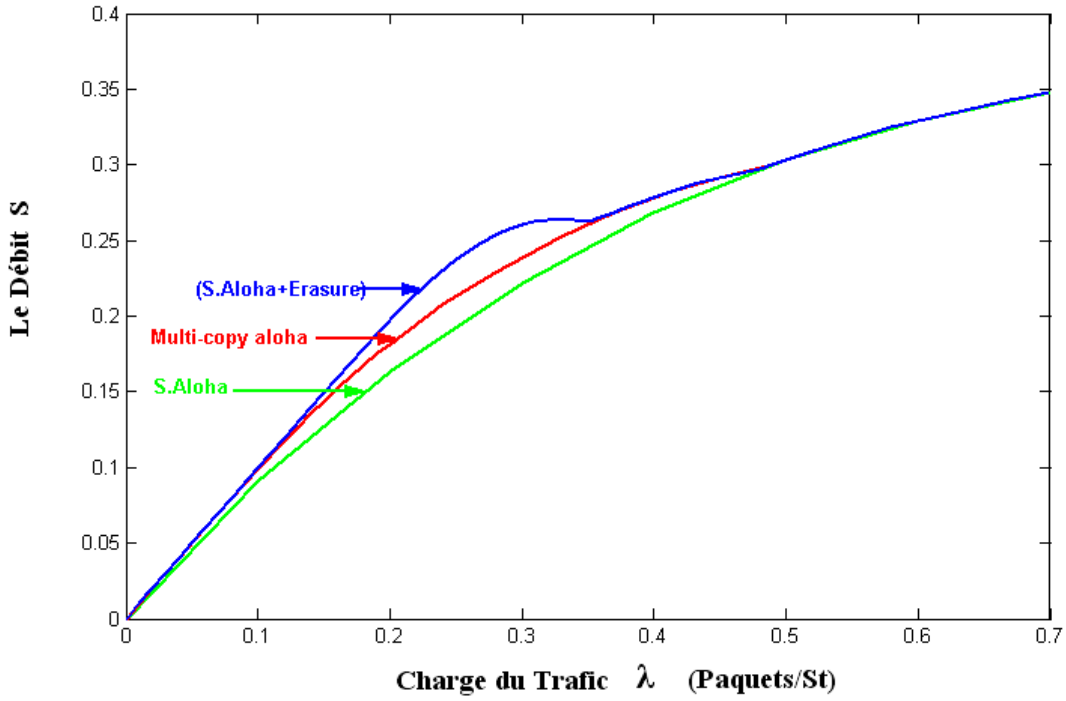


Figure 3.5 L'effet du Codage Erasure sur les techniques d'Accès Aléatoires.

III.4 Slotted-non-persistent CSMA avec le codage Erasure:

III.4.1 Description du Schéma:

En CSMA, quand un nœud veut transmettre un paquet, il écoute d'abord le canal, si le canal est libre alors il transmette immédiatement, mais si le canal est occupé (autres noeuds en transmission), le nœud attend jusqu'à ce qu'il devienne libre. Cela permet de diminuer la probabilité de collision et prouve la performance de cette technique. Plusieurs variations de Csma sont proposées, comme P-persistent Csma, non-persistent Csma, Slotted non-persistent Csma, Csma/Cd et Csma/Ca ...etc. **C**ependant i pour récupérer les paquets perdus [37,40]. Dans ce paragraphe, on intègre le codageon n'a pas pu éviter la collision dans le Csma, et le codage Erasure est employé ic Erasure dans Slotted non-persistent Csma, comme un exemple pour montrer, comment on peut faire évoluer la performance de cette technique.

-On suppose que (N, K) codes erasures sont choisis pour la récupération d'erreur.

III.4.2 Procédure de Fonctionnement:

Notre modèle sera fonctionnel selon [37,40]:

1. Quand il y a K paquets originaux, on les encode en (N,K) mots de code et on met au niveau de chaque récepteur un identificateur de code.

2. Chaque paquet codé est alors retardé pour un intervalle aléatoire de temps avant que le nœud essaye de l'envoyer. Quand ce temps de retard est expiré, le nœud écoute le canal; Si le canal est libre, alors il transmette le paquet dans le slot prochain. Dés qu'un paquet original sera envoyé, un temporisateur Ts est lancé.

3. Si le canal est occupé, le paquet passera un autre retard de temps aléatoire avant que le nœud essaye de l'envoyer encore.

4. Au niveau du récepteur, quand un paquet appartenant à un nouveau mot de code, est reçu, un temporisateur Tr est lacé pour recevoir d'autres paquets de ce mot de code. Si au moins K paquets codés par N sont correctement reçus avant que Tr expire, une information est envoyée pour reconnaître tous les paquets originaux de K. Quand Tr expire, si seulement n (n<N) paquets codés sont reçus et m par n sont les paquets originaux, alors seulement ces paquets originaux sont reconnus.

5. Quand un paquet original est reconnu avant que sont temporisateur expire, ce paquet est enlevé de la file d'attente de l'expéditeur. Autrement, ce paquet est traité comme paquet retransmis, comme dans la technique Aloha.

III.4.3 Modèle analytique:

Comme dans [45,46,47], on considère τ comme la durée de propagation entre deux nœuds (c'est le temps nécessaire pour le nœud pour sentir la transmission des autres nœuds), et T comme le temps nécessaire pour transmettre un paquet. On pose $a = t/T$. On suppose aussi que le trafic des nouveaux ou retransmis data, constitue un processus de Poisson, avec une valeur moyenne l (paquets/seconde), et un codage Erasure de (N,K) est utilisé.

Après le codage du trafic chargé au réseau qui est $G = Nl/K$. Ici nous suivons la dérivation semblable que [41]. La **Figure 3.8** montre les rapports de plusieurs variables utilisés dans l'analyse. Dans cet exemple, a=1/3. L'axe de temps est divisé en cycles, chaque cycle se compose d'une période d'activité B d'une moyenne $\overline{B}$, et une période de vide I d'une moyenne $\overline{I}$. On pose U le temps (d'une moyenne $\overline{U}$) nécessaire pendant un cycle pour qu'un canal délivre un paquet original avec succès.

Alors le débit aura pour expression:

$$S = \frac{\overline{U}}{\overline{B} + \overline{I}} \tag{3.8}$$

Comme dans [41] on peut avoir $\overline{B} = T + t$.

La longueur de I est toujours en nombre entier des slots, puisque un paquet peut seulement être transmis au début d'un slot. En outre, pour qu'un slot de vide soit suivi par une transmission, un ou plusieurs arrivées doivent se produire dans le slot de vide. On pose J la longueur de la période de vide dans le slot. On aura

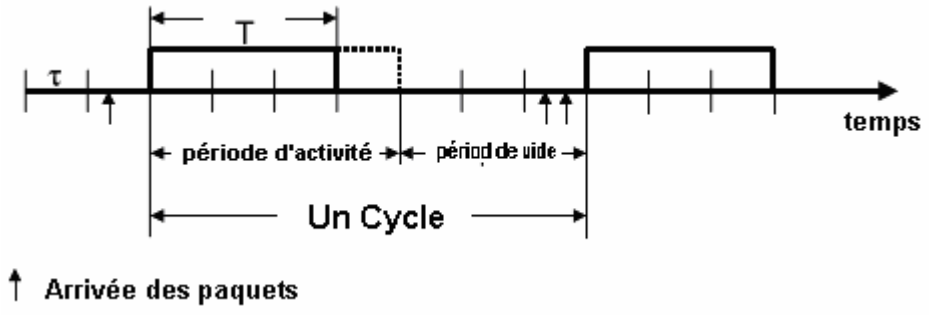


Figure 3.6 Slotted non-persistent Csma.

$P\{J = 0\} = P$ (Au moins une arrivée par τ)

$$= 1 - e^{-Gt/T} = 1 - e^{-aG} \tag{3.9}$$

On met $Pa = P\{J = 0\}$ la probabilité qu'au moins une arrivée par slot. Il est facile de vérifier que:

$$P\{J = k\} = (1 - Pa)^k Pa \tag{3.10}$$

Cela veut dire:

$$\bar{J} = \frac{1 - Pa}{Pa} = \frac{e^{-aG}}{1 - e^{-aG}} \tag{3.11}$$

La période moyenne d'un slot de vide $\bar{I}$ est égale à τ fois $\bar{I}$, et:

$$\bar{I} = \frac{t.e^{-aG}}{1 - e^{-aG}} \tag{3.12}$$

Maintenant on dérive $\bar{U}$. D'abord on a besoin d'identifier deux variables; on a $\bar{U}s$ qui est la probabilité pour qu'une transmission soit réussie, sachant qu'il existe quelques transmissions pendant un time slot T, et Ps qui présente la probabilité qu'un paquet original soit reçu ou peut être récupéré avec succès, en donnant une transmission réussie sans collisions. Aussi on a:

$$\bar{U} = UsPsT \tag{3.13}$$

Comme dans [41], Us peut être exprimé comme suit:

$Us = P$ {un seul arrive pendant l'intervalle τ /quelques arrivées produits} (3.14)

On utilise les statistiques de Poisson, on aura:

$$\text{P\{une seule arrivée pendant } \tau\} = G(t/T)e^{-G(t/T)} \tag{3.15}$$

$$= aGe^{-aG} \tag{3.16}$$

Et

$$\text{P\{quelques arrivées se produisent\}} = 1 - e^{-G(t/T)} = 1 - e^{-aG} \tag{3.17}$$

A partir de (3.12), (3.16) et (3.17) Us peut être définit comme suit:

$$Us = \frac{aGe^{-aG}}{1 - e^{-aG}} \tag{3.18}$$

Maintenant on détermine l'expression de Ps, pour cela on considère un paquet encodé, ce paquet est transmis avec succès si on aura un seul arrivée dans le time slot précédent, comme c'est montré dans la **Fig 8**. La valeur moyenne du débit des arrivées pendant ce time slot est égale à aG, ce qui fait que la probabilité que ce paquet soit transmis avec succès est:

$$P = e^{-aG} \tag{3.19}$$

Aussi, Pk la probabilité qu'au moins k paquets encodés peuvent êtres reçus avec succès, est égale à:

$$Pk = \sum_{i=k}^{N} \binom{N}{i} p^i (1-p)^{N-i} \tag{3.4}$$

On pose $P_{n,m}$ la probabilité qu'on reçoit seulement n $(n \le k)$ paquets encodés, où il existe m paquets originaux entre les n reçus, cela nous permet d'écrire:

$$P_{n,m} = \binom{k}{m}\binom{N-K}{n-m} p^i (1-p)^{N-n} \tag{3.5}$$

* Comme on a mentionné précédemment, un paquet original peut être reçu avec succès (ou récupéré) si:

1. Au moins k parmi N paquets encodés sont correctement reçus; ou

2. n $(n \le k)$ parmi N paquets encodés sont reçus; et m entre n reçus sont originaux; et le paquet sous étude est parmi ces m paquets (avec probabilité de m/k).

Ensuite on a:

$$Ps = \frac{Pk + \sum_{n=1}^{K-1}\sum_{m=1}^{n}(m/K)P_{n,m}}{\sum_{i=i}^{N}\binom{N}{i}P^i(1-P)^{N-i}\,i} \tag{3.20}$$

$$= \frac{Pk + \sum_{n=1}^{K-1}\sum_{m=1}^{n}(m/K)P_{n,m}}{Np} \tag{3.21}$$

Donc :

$$\overline{U} = UsPsT = \frac{P_s aGe^{-aG}T}{1-e^{-aG}} \tag{3.22}$$

Alors le débit de notre Slotted non-persistent Csma avec le codage Erasure est égale:

$$S = \frac{\overline{U}}{\overline{B}+1} = \frac{aGPse^{-aG}}{1+a-e^{-aG}} \tag{3.23}$$

On peut noter que si N=K=1, notre modèle devient le conventionnel Slotted non-persistent Csma, et dans ce cas on aura Ps=1, et le débit sera égale à :

$$S = a\,l\,e^{-a\,l}/(1+a-e^{-a\,l}) \tag{3.24}$$

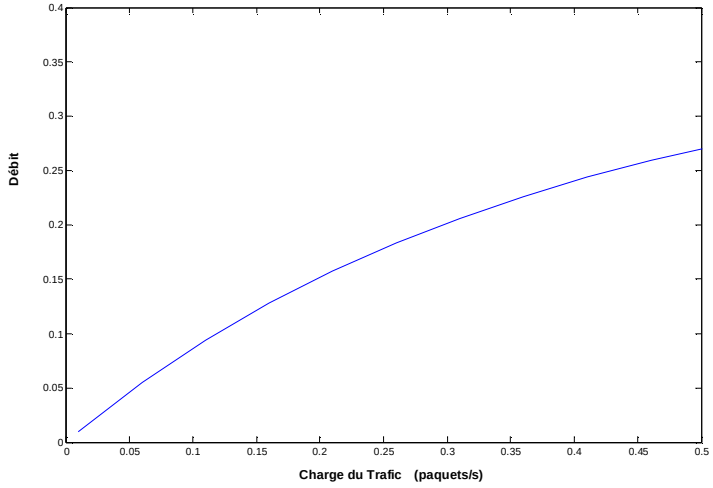


Figure 3.7 Le Débit du Slotted non-persistent CSMA Conventionnel

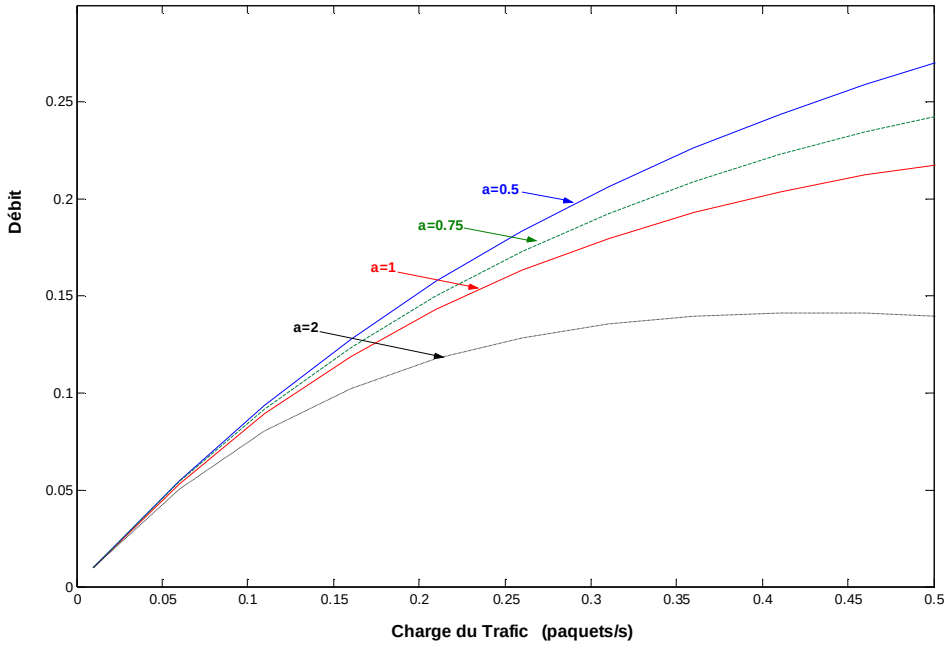


Figure 3.8 Le Débit du Slotted non-persistent CSMA Conventionnel avec a différentes a=0.5, 0.75, 1 et 2 .

- **Tableau .2**. montre les paramètres qu'on a utilisé pour tracer nos courbes. D'autre part, on a limité $N \leq 20$, comme dans le cas de Aloha.

Charge du trafic λ	Erasure (N, K)	Charge du trafic λ	Erasure (N, K)
0.01–0.06	(20,18)	0.29–0.36	(20,14)
0.07–0.13	(20,17)	0.37–0.45	(20,13)
0.14–0.2	(20,16)	0.46–0.47	(20,12)
0.21–0.28	(20,15)	0.48–	(1,1)

Tableau 3.2 Paramètres utilisés pour les différentes charges du trafic dans le Modèle Csma

- La **Fig.9**. montre une comparaison entre notre modèle et le conventionnel Slotted non-persistent Csma, étant donné que a=0.5.

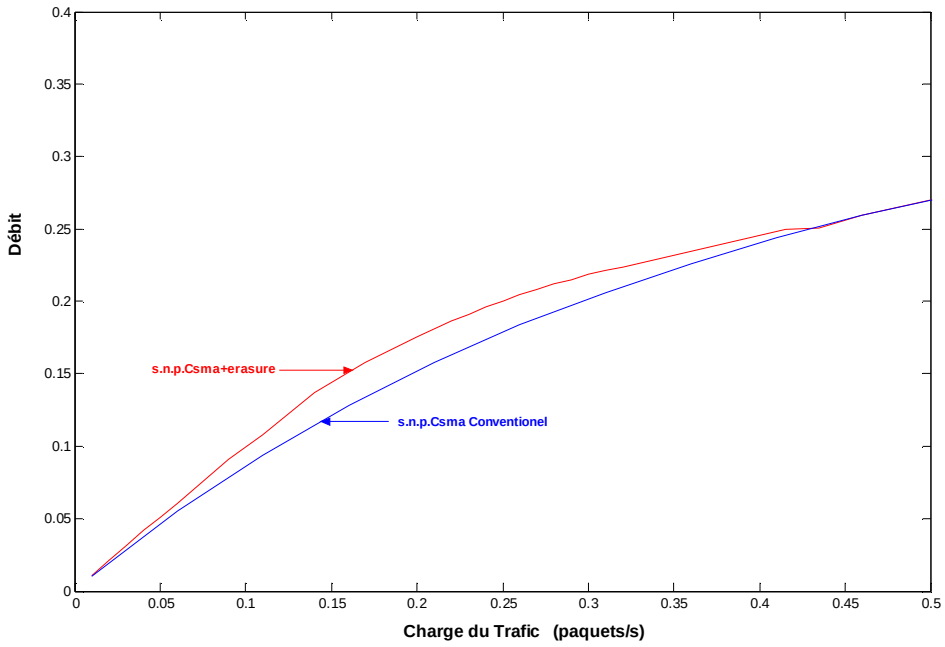


Figure 3.9 Le Débit du Slotted non-persistent CSMA+le codage Erasure

On remarque que lorsque λ varie entre 0.1 et 0.45, notre modèle se comporte mieux que le modèle conventionnel. Cependant lorsque λ est très fort ou très faible, la performance de notre modèle est similaire avec celle du conventionnel. Il est également évident que pour le CSMA, notre modèle sera moins efficace avec des petites valeurs de 'a' car moins de collision se produira. Aussi il doit être vrai que pour le même λ, les paramètres peuvent êtres différents quand "a" est différent. Une comparaison entre **Tableau 3.2** et **Tableau 3.1**, démontre qu'il y a une seule différence entre notre modèle et le CSMA; cette différence apparaît dans la

diminution du débit lorsque λ augmente, jusqu'à ce que notre modèle soit similaire au CSMA conventionnel après que le λ soit plus grand que 0.48. La raison de cette diminution réside dans le fait que la probabilité de collision est moins significative après l'intégration de l'écoute de la porteuse, et par conséquence on peut encore bénéficier de la possibilité d'augmenter la récupération des paquets par l'injection de plus de paquets aléatoires lorsque le trafic augmente.

III.5 Conclusion:

Dans ce chapitre, on vous propose l'utilisation du codage Erasure avec la technique d'accès aléatoire Aloha et CSMA respectivement, afin de montrer la performance de ces protocoles d'accès mélieu. On remarque que quand le codage Erasure est implémenté il sera facile de récupérer les k paquets originaux quand k parmi N paquets codés sont reçus, et par voie de conséquence les paquets heurtés peuvent êtres ainsi récupérés. D'autre part on voit que l'utilisation du codage Erasure est bénéfique seulement quand le débit de trafic est élevé au niveau de mélieu de communication, puisque quand le débit est assez faible la probabilité de collisions est très faible aussi, ce qui implique que ce n'est pas nécessaire de récupérer les paquets heurtés. Par contre quand le débit est élevé, la collision est déjà très sérieuse, et de ce fait l'utilisation de toutes les techniques de contrôle d'erreur devient très nécessaire pour améliorer la situation. A partir de la simulation, on observe que si on accorde seulement des temps de retransmission limités, notre modèle peut de manière significative améliorer la performance du système, en choisissant un k petit on peut ainsi réduire le retard, ce qui aboutit à diminuer l'exécution optimale de débit.

Conclusion Générale

L'objectif de ce travail était d'évaluer la performance des protocoles MAC dans les réseaux de communications satellitaires. L'interface radio, et plus particulièrement l'accès multiple constituent assurément l'un des points les plus sensibles de la chaîne de transmission. Il s'agit en effet, d'une part de transmettre un maximum de données utiles par unités de temps entre la source et le destinataire, mais également de fixer les règles permettant à tous les émetteurs de communiquer de façon optimale. Il sera donc nécessaire de définir les principes de communication à l'intérieur du milieu pour que les utilisateurs puissent se partager le canal. Ces principes basés sur le partage de la ressource sont appelés protocoles d'accès. Pour atteindre nos objectifs, en premier lieu, nous avons essayé de présenter les techniques d'accès multiple. Ici, une station qui a besoin d'émettre va tenter de le faire immédiatement sans avoir besoin « d'attendre son tour ». Il se peut donc qu'il y ait des collisions, et le but des protocoles à accès aléatoire va être de les minimiser. Le principe de base de ces protocoles est simple: laisser les utilisateurs accéder librement au canal lorsqu'ils ont des données à transmettre. Dans ces conditions il est clair que des collisions se produisent. Afin d'améliorer la capacité d'Aloha, d'autres protocoles ont étés proposés, comme par exemple Slotted Aloha, CSMA, CSMA-CA et CSMA-CD, la différence entre ces protocoles réside dans leur réaction face aux collisions. L'inconvénient majeur de ces protocoles est le faible débit, à cause des phénomènes de collision des paquets.

Dans ce travail on a essayé d'exposer les capacités de transmission et présenter ainsi le débit de trafic que peut offrir chaque technique. Finalement la technique CSMA et ses dérivées présentent de meilleurs débits que celles des techniques Aloha. Cela est du à son efficacité face aux problèmes de collisions. Dans ce même contexte, nous proposons dans le dernier chapitre d'intégrer le codage Erasure avec les protocoles à accès aléatoire pour récupérer les paquets heurtés et améliorer ainsi la performance de débit.

Avec le codage Erasure, un nombre k des paquets originaux sont codés en (N,k) mots erasures, qui consiste à N (>k) paquets codés. À la réception, on peut récupérer un nombre k de paquets originaux si on reçoit un nombre k parmi N paquets codés. Alors le codage Erasure augmente la charge offerte du trafic, et la probabilité de succès des paquets de transmission peut être aussi augmentée si ce codage est bien implémenté. On remarque que lorsque le codage Erasure est implémenté il est facile de récupérer les k paquets originaux parmi N

paquets codés reçus, et par voie de conséquence les paquets heurtés peuvent êtres ainsi récupérés.

Nous avons constaté que l'utilisation du codage Erasure est bénéfique seulement si le débit de trafic est élevé au niveau du milieu de communication, puisque lorsque le débit est assez faible, la probabilité de collisions est très faible aussi, et donc ce n'est pas nécessaire de récupérer les paquets heurtés. Par contre lorsque le débit est élevé, la collision est déjà très sérieuse, et de ce fait l'utilisation de toutes les techniques de contrôle d'erreur devient nécessaire pour améliorer la situation.

Dans ce travail préliminaire sur l'utilisation des techniques de codages évolués, nous avons constaté qu'il serait raisonnable d'évoluer vers une approche plus formelle du problème du codage et de la correction des erreurs. Nous pensons approfondir notre travail par une étude plus poussée de l'Algèbre des Nombres et plus précisément de la théorie des Groupes de Galois. Dans un souci de recherche plus évolué, et étant donné les développements survenus dans la littérature, nous comptons approfondir nos recherches avec la problématique suivante : comment introduire les Codes de Clifford pour améliorer la stabilité des corrections sur les protocoles envisagées. Il a été en effet démontré qu'une généralisation de la théorie des Groupes de Galois conduirait inévitablement à une Algèbre évoluée appelée Algèbre de Kac Moody.

Table des Matières

Abréviations

La signification d'une abréviation ou d'un acronyme n'est souvent indiquée qu'à sa première apparition dans le texte. Il existe dans la plupart des cas une abréviation en français et une abréviation en anglais. Toutes les deux sont indiquées une première fois puis nous employons l'abréviation la plus usuelle, qui est le plus souvent l'abréviation en anglais.

Acronymes & Abréviations

ABR	**A**vailable **B**it **R**ate
BER	**B**it **E**rror **R**atio
BCH	Bose Chaudhuri Hocquenghem
ALOHA	Random access technique
CAC	**C**onnexion **A**dmission **C**ontrol
CDMA	**C**ode **D**ivision **M**ultiple **A**ccess
CNES	**C**entre **N**ational français d'**E**tudes **S**patiales
CSMA	**C**arrier **S**ense **M**ultiple **A**ccess
CSMA/CD	**CSMA/ C**ollision **D**etection
DS/ CDMA	**D**irect **S**equencing **C**ode **D**ivision **M**ultiple **A**ccess
ETSI	**E**uropean **T**elecommunications **S**tandards **I**stitute
FDD	**F**requency **D**ivision **D**uplex
FDMA	**F**requency **D**ivision **M**ultiple **A**ccess
FEC	**F**orward **E**rror **C**orrection
FH/CDMA	**F**requency **H**opping **C**ode **D**ivision **M**ultiple **A**ccess
GEO	**G**eostationary **E**arth **O**rbit
GH	**G**aranteed **H**andover
GSM	**G**lobal **S**ystem **M**obile
ISL	**I**nter**S**atellite **L**ink
LAN	**L**ocal **A**rea **N**etwork
LEO	**L**ow **E**arth **O**rbit
M.C.ALOHA	**M**ulti **C**opy ALOHA
MAC	**M**edium **A**ccess **C**ontrol

MAI	**M**ultiple **A**ccess **I**nterference
MEO	**M**edium **E**arth **O**rbit
P-PERSISTENT	Persistent With probability **P**
PRMA	**P**acket **R**eservation **M**ultiple **A**ccess
QoS	**Q**uality **o**f **S**ervice
RLC	**R**etransmission **L**ink **C**ontrol
RNIS	**R**éseau **N**umérique à **I**ntégration de **S**ervices
RS	**R**eed-**S**olomon codes
S.ALOHA	**S**lotted ALOHA
S-CDMA	**S**atellite **C**ode **D**ivision **M**ultiple **A**ccess
S-CDMA/PRMA	**S**atellite CDMA/PRMA
SF	**S**preading **F**actor
SGA	**S**tandard **G**aussian **A**pproximation
SNR	**S**ignal to **N**oise **R**atio
STC	**S**low **T**ransfer **C**apability
TDD	**T**ime **D**ivision **D**uplex
TD/CDMA	**T**ime **D**ivision CDMA
TDMA	**T**ime **D**ivision **M**ultiple **A**ccess
UMTS	**U**niversal **M**obile **T**elecommunication **S**ystem
UTRA	**U**MTS **T**errestrial **R**adio **A**ccess
W-CDMA	**W**ideband CDMA