

LA TRANSFORMÉE MOJETTE : UN CODE CORRECTEUR D'ERREURS

Benoît Souhard, Christian Chatellier, Christian Olivier et Anne-Marie Poussard

Laboratoire IRCOM-SIC, UMR CNRS 6615, Bât. SP2MI – Téléport2 – Bvd M. & P. Curie, BP 30179 – F, 86962 Futuroscope-Chasseneuil Cedex France
souhard@sic.sp2mi.univ-poitiers.fr

Résumé :

La Transformée Mojette (MT) incluse dans une chaîne de transmission de données numériques doit être considérée comme un codage canal. Appliquée sur des données binaires, la MT nous permet de détecter et de corriger des erreurs isolées, tandis que la MT appliquée à une technique de corrélation corrige des erreurs par paquets (bursts). Dans cet article nous mettons en œuvre ces deux types de transformées, puis nous comparons leurs performances par rapport aux codeurs canaux usuels : les codes en blocs et les codes cycliques.

Mots clef :

Transformée Mojette, détection et correction d'erreurs.

1 Introduction

Un intérêt en transmissions numériques est d'utiliser en même temps un codage source et un codage canal.

Le codage canal qui nous intéresse plus particulièrement, consiste à effectuer diverses opérations mathématiques avec les informations issues du codage source. A la réception, le décodeur vérifie si ces lois sont toujours respectées, détecte ainsi les erreurs dues à la transmission et les corrige s'il en a la possibilité. Rappelons que les différents codage canaux utilisés en transmission sont : les codes en blocs linéaires [1], les codes cycliques [2], les codes convolutifs et les turbo codes.

Nous rappelons dans la seconde partie les définitions du codage en blocs linéaires. La troisième partie traite de l'application du codage canal via la MT [3]. Les composantes du support de la MT peuvent être des bits ou des nombre dits *q-aires* ($q=2^m$). Dans le premier cas nous parlons de Transformée Mojette Binaire (MTB) et dans le second, appliquée à une technique de corrélation, nous parlons de Transformée Mojette Spline (MTS). Dans chaque cas nous donnerons des résultats de simulations.

2 Les codes en blocs linéaires

A un bloc de k éléments binaires (0 ou 1), appelé mot d'informations \mathbf{m} , on fait correspondre un bloc de n éléments binaires ($n > k$), appelé mot du code \mathbf{c} . Ces codes notés $C(n,k)$ sont caractérisés par leur taux de correction t et leur rendement défini par $\tau = k/n$.

L'obtention des mots du code \mathbf{c} s'effectue en appliquant la relation matricielle :

$$\mathbf{c} = \mathbf{m}\mathbf{G} \quad (1)$$

où \mathbf{G} représente la matrice génératrice permettant de coder le mot \mathbf{m} . Cette relation matricielle peut aussi s'écrire sous la forme :

$$\mathbf{c} = [\mathbf{c}_N, \mathbf{c}_C] = \mathbf{m}[\mathbf{G}_N, \mathbf{G}_C] \quad (2)$$

\mathbf{c}_N est un vecteur dont les composantes sont non contraintes, c'est à dire qu'elles sont indispensables pour la reconstruction de l'information transmise. \mathbf{c}_C est un vecteur de composantes contraintes, il porte l'information redondante utilisée pour la reconstruction des erreurs. Les matrices \mathbf{G}_N et \mathbf{G}_C sont les matrices génératrices des vecteurs correspondants.

Lors d'une transmission de données nous supposons que le mot reçu \mathbf{r} est entaché d'erreurs additives. Si \mathbf{c} est le mot du code émis alors le mot \mathbf{r} s'écrit :

$$\mathbf{r} = \mathbf{c} + \mathbf{e} \quad (3)$$

où \mathbf{e} est le vecteur d'erreurs dans lequel une composante égale à 1 indique la présence d'erreurs.

La détection des erreurs se fait en calculant le syndrome \mathbf{s} par la relation suivante :

$$\mathbf{s} = \mathbf{r}\mathbf{H}^T = (\mathbf{c} + \mathbf{e})\mathbf{H}^T = \mathbf{e}\mathbf{H}^T \quad (4)$$

avec \mathbf{H} la matrice duale qui vérifie l'expression $\mathbf{G}\mathbf{H}^T = \mathbf{0}$.

Un syndrome \mathbf{s} différent du vecteur nul implique la présence d'erreur(s) de transmission. Si le syndrome est nul, cela signifie que le mot reçu \mathbf{r} est égal à un mot du code, ce qui amène deux possibilités : une première dans laquelle $\mathbf{e}=\mathbf{0}$ c'est à dire qu'il n'y a pas d'erreurs de transmission et une deuxième dans laquelle $\mathbf{e}\neq\mathbf{0}$ mais $\mathbf{e}+\mathbf{c}$ est un autre mot du code. On est alors dans une configuration d'erreurs indétectables.

Pour caractériser de tels codes, on dispose de plusieurs outils comme la distance minimale (d_{min}) au sens de Hamming, qui est égale au plus petit nombre de colonne linéairement dépendante de la matrice \mathbf{H} , le pouvoir de détection, qui est égale à $d_{min}-1$, et enfin le pouvoir de correction t :

$$t = \left\lfloor \frac{d_{min}-1}{2} \right\rfloor \quad (5)$$

où $\lfloor x \rfloor$ désigne la partie inférieure par valeur entière de x [1].

Ainsi pour un code noté $C(7,4)$, d_{min} est égale à 3 et son pouvoir de correction t est égale à une erreur corrigéable sur 7 bits transmis et le rendement $\tau=0,571$.

3 La Transformée Mojette

La Transformée Mojette (MT) est basée sur le principe de la transformée de Radon discrétisée. L'intérêt principal d'une telle transformée étant sa linéarité et surtout son inversibilité. Elle permet de projeter un signal bidimensionnel en un ensemble de signaux monodimensionnels (Figure 1)

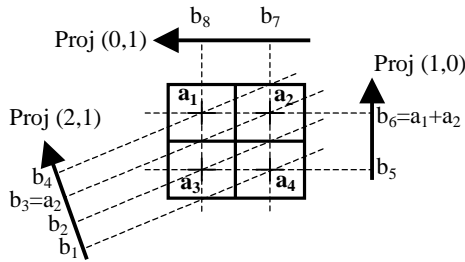


Figure 1 : Exemple de Transformée Mojette

Elle se calcule par la relation suivante :

$$f(x, y) = \sum_{k=-\infty}^{+\infty} \sum_{l=-\infty}^{+\infty} f(k, l) \delta(x-k) \delta(y-l) \quad (6)$$

où $f(k, l)$ représente les coefficients du support original au point (k, l) (les a_j de la Figure 1) et δ le symbole de Kronecker. Le résultat de l'équation (6) noté $f(x, y)$ est appelé bin (b_j de la Figure 1) et correspond à la somme des éléments rencontrés le long d'une projection.

La MT étant linéaire et inversible nous pouvons écrire la transformée avec l'écriture des codes en blocs linéaires.

$$\mathbf{c} = \mathbf{mG} \quad (7)$$

où \mathbf{G} est la représentation matricielle de la MT.

Avec un ensemble suffisant de projection, en respectant les équations de Katz [4], il est possible de reconstruire les composantes du support. Il existe alors des bins non contraints notés \mathbf{c}_N servant à la reconstruction et des bins contraints notés \mathbf{c}_C servant à la reconstruction en cas d'erreurs sur les bins non contraints. L'écriture (2) est alors respectées :

$$\mathbf{c} = [\mathbf{c}_N, \mathbf{c}_C] = \mathbf{m}[\mathbf{G}_N, \mathbf{G}_C]$$

\mathbf{G}_N est la représentation des bins non contraint de la MT et \mathbf{G}_C celle des bins contraints. Sa matrice duale associée sera de la forme :

$$\mathbf{H} = [\mathbf{G}_C \mathbf{G}_N^T, \mathbf{Id}_C] \quad (8)$$

qui vérifie que le produit de la matrice génératrice par sa matrice duale transposée est une matrice nulle.

3.1 La Transformée Mojette Binaire

La Transformée Mojette Binaire (MTB) est une application de la MT sur des données binaires. Si nous appliquons une MTB sur un support carré de dimension 2 avec comme jeu de projections $\{(2,1); (1,0); (0,1)\}$ (cf Figure 1), à partir des 4 bits d'informations qui représentent le mot \mathbf{m} (a_1 à a_4) nous obtenons après la MTB un mot du code de 8 bits. Le code peut alors s'écrire $C(8,4)$ avec comme distance minimale $d_{min}=3$ et donc un pouvoir de correction $t=1$ erreur.

L'intérêt de la MTB est qu'elle permet également de corriger un certain nombre d'erreurs consécutives. Pour connaître ce nombre, nous proposons la méthode suivante : En reprenant l'équation (3) et en supposant qu'il y ait un bit erroné sur 8, les huit configurations d'erreur sont alors les suivantes :

$$\begin{cases} \mathbf{e}_1 = [1 & 0 & 0 & \dots & 0]^T \\ \mathbf{e}_2 = [0 & 1 & 0 & \dots & 0]^T \\ & \vdots \\ \mathbf{e}_8 = [0 & 0 & 0 & \dots & 1]^T \end{cases}$$

Il en résulte un calcul de huit syndromes, qui sont convertis en décimal puis rangés dans un vecteur \mathbf{ES}_i . Pour le jeu de projection précédent, nous obtenons :

$$\mathbf{ES}_1 = [10 \ 9 \ -9 \ -10 \ 8 \ -11 \ -13 \ -14]$$

Tous les éléments du vecteur \mathbf{ES}_1 sont différents, ce qui permet de dire que le jeu de projections peut corriger une erreur parmi 8. Supposons maintenant qu'il y est deux erreurs consécutives parmi 8 bits, nous avons alors 7 configurations d'erreurs possibles. Suivant le même principe nous obtenons un nouveau vecteur de syndrome :

$$\mathbf{ES}_2 = [-12 \ 15 \ -12 \ 13 \ 12 \ -9 \ -12]$$

On constate que la composante -12 se retrouve deux fois dans ce vecteur, cela indique l'impossibilité de corriger 2 erreurs consécutives puisque à deux configurations d'erreurs différentes correspond une seule valeur de syndrome. La condition nécessaire pour corriger deux erreurs consécutives est donc d'avoir les éléments de \mathbf{ES}_2 différents entre eux et différents de \mathbf{ES}_1 .

Il existe au moins une configuration qui permet de corriger autant d'erreurs consécutives que l'on souhaite à partir d'un support carré de taille N avec $N > 4$ et comme projections : $\{(N,1); (2,1); (1,1)\}$. Nous pouvons corriger toutes les erreurs consécutives avec un taux de redondance qui diminue lorsque N augmente.

Nous avons effectué des simulations qui nous permettent de comparer les performances de la MTB avec les codes de Hamming (noté H) et les

codes BCH. Ces codes sont de la forme $C(n,k,t)$ où C représente le code utilisé. Nous avons alors le code MTB(8,4,1) calculé sur un support carré de taille 2 avec les projections $\{(2,1);(1,0);(0,1)\}$ puis le code MTB(47,25,1) calculé sur un support carré de dimension 5 avec comme jeu de projections $\{(5,1);(2,1);(1,1)\}$. La Figure 2, représente les caractéristiques de ces codes ainsi que celles des codes H(15,11,1), H(7,4,1), BCH(31,16,3) et BCH(15,7,2), ces derniers étant choisis pour leurs propriétés proches de celles des MTB utilisées.

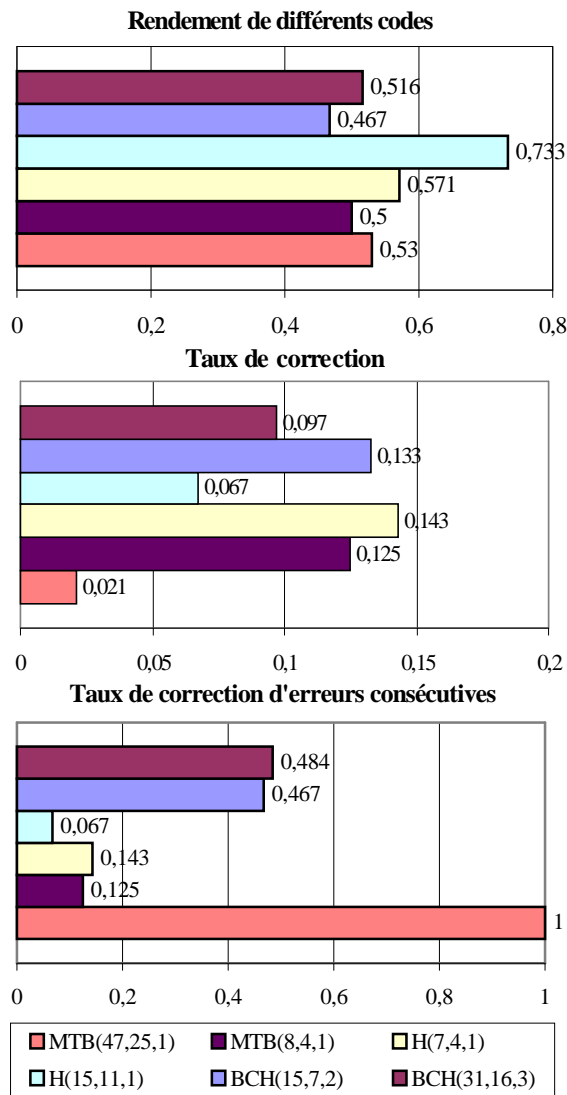


Figure 2 : Comparaison de la MTB avec les codes BCH et de Hamming.

Pour le code BCH(15,7,2) les deux premières données de la figure permettent de dire que ce code a un rendement de 0,467 pour un taux de correction de 0,133 c'est à dire qu'il corrige 2 erreurs isolées sur 15 bits transmis. La troisième information est le pouvoir de correction d'erreurs consécutives dans ce cas 0,467 (7 erreurs consécutives sur 15 bits). Nous pouvons alors remarquer qu'à rendement approximativement égal la MTB(8,4,1) se comporte bien, en terme de correction d'erreur

isolée, par rapport aux autres codes. La MTB(47,25,1), malgré un taux de correction plus faible, corrige toutes les erreurs consécutives que tous les autres codes, elle peut alors s'avérer utiles dans les canaux de transmissions provoquant des erreurs en rafales.

3.2 La Transformée Mojette Spline

Dans une projection MT, une composante sur support ne participe qu'à un bin et sa contribution est unitaire. En choisissant le modèle de Haar [3][5], la projection d'une composante se repartie sur plusieurs bins, sa contribution à un bin spline est alors proportionnelle à la longueur de l'intersection de cette composante avec la projection issue du bin spline.

La trace d'un pixel sur la projection est trapézoïdale et est définie par le double produit de convolution:

$$t(p,q) = R(p) * R(q) * P(p,q) \quad (9)$$

où $R(p)$ (respectivement $R(q)$) représente une fonction rectangle de largeur p (respectivement q). Quant à $P(p,q)$, elle représente une fonction rectangle de largeur deux seulement si p ou q sont paires sinon elle définit une fonction unitaire.

La MTS, notée $Proj_s(p,q)$ est donc définie par :

$$Proj_s(p,q) = Proj(p,q) * t(p,q) \quad (10)$$

L'intérêt de la MTS est de travailler sur des éléments q -aires comme pour les codes RS. La technique de corrélation appliquée sur des éléments binaires n'aurait aucun intérêt excepté celui d'augmenter le rendement par rapport à la MTB.

3.2.1 Principe de détection et de correction

Pour une matrice $P \times Q$ le nombre de bins par projection est égale à :

- $|P \times p| + |Q \times q|$ si p ou q sont paires
 - $|P \times p| + |Q \times q| - 1$ si p et q sont impaires
- et un élément du support sera présent :
- $p + q$ fois si p ou q sont paires.
 - $p + q - 1$ fois si p et q sont impaires.

L'algorithme de correction explicité par la suite n'est valable que pour le jeu de projections ayant pour forme $\{(Q,1);(0,1);(1,0)\}$. La détection des erreurs intervient seulement sur la première projection, celle ci étant la seule où le modèle de Haar est réellement appliqué.

Etant donné que la deuxième projection contient Q bins si une erreur par paquet intervient sur plus de Q bins splines, alors il y a un risque pour que les trois projections soient fausses ce qui rend impossible la correction. De ce fait, on suppose que seules deux des trois projections peuvent être erronées.

Afin de détecter une erreur, il suffit de décorréler la première projection spline dans les deux sens et de comparer les deux vecteurs bins obtenus. Pour plus de précision, le premier vecteur bins (B_1) est obtenu en décorrélant la projection spline du premier au dernier terme, le second

vecteur bins (\mathbf{B}_2) est obtenu en décorrélant du dernier au premier terme.

Si les deux vecteurs \mathbf{B}_1 et \mathbf{B}_2 sont égaux cela implique qu'aucune erreur de transmission est intervenue sur la première projection. La reconstruction des composantes du support original est alors possible sans avoir besoin des deux autres projections. Dans le cas inverse, la troisième projection sert de projection détectrice d'erreurs et est supposée correcte. Une des propriétés fondamentale de la MT est alors utilisée. Cette propriété est donnée par l'expression suivante :

$$S_C = \sum_{i=1}^{i=n=P} \mathbf{B}(i) = \sum_{j=1}^{j=k=P \times Q} \mathbf{A}(j) \quad (11)$$

où \mathbf{B} représente le vecteur bins des différentes projections de la MT (ici les bins de la troisième projection) et \mathbf{A} représente le vecteur des composantes du support original de dimension $P \times Q$.

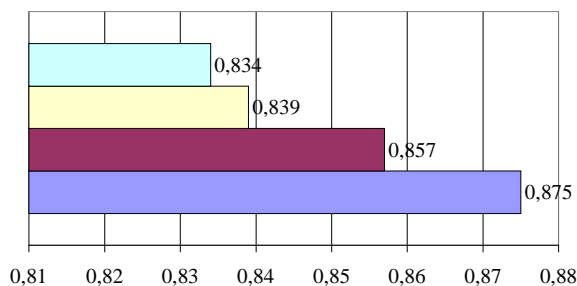
La correction s'effectue en comparant itérativement le vecteur \mathbf{B}_1 à la somme correcte S_C des composantes du support original. Si le vecteur \mathbf{B}_1 ne vérifie pas (11) alors nous remplaçons la dernière valeur non modifiée des composantes de \mathbf{B}_1 par la dernière valeur des composantes de \mathbf{B}_2 et ceci jusqu'à ce que la somme de \mathbf{B}_1 soit égale à S_C . Dès que ce test est vérifié nous appliquons la Transformée Mojette Inverse [3]. Si après reconstruction, un des termes supposé non erroné des deux dernières projections est non nul, alors il faut continuer notre procédure itérative; sinon si nos termes supposé vrais sont nuls cela entraîne que nous avons correctement reconstruit nos composantes originales.

3.2.2 Résultats de simulation

Sur la Figure 3, nous comparons la MTS à un code RS. La notation utilisée est la même que pour les résultats de la MTB excepté le dernier terme qui est égale à la puissance "m" définissant la taille des éléments q-aires.

Toujours pour un rendement approximativement égal (à 0,04 près), nous pouvons remarquer que le pouvoir de correction de nos codes MTS sont très nettement supérieur à celui du code RS utilisé le plus couramment, en moyenne environ 5% d'erreurs corrigés en plus pour une perte d'environ 2% en terme de rendement.

Rendement de différents codes



Taux de correction

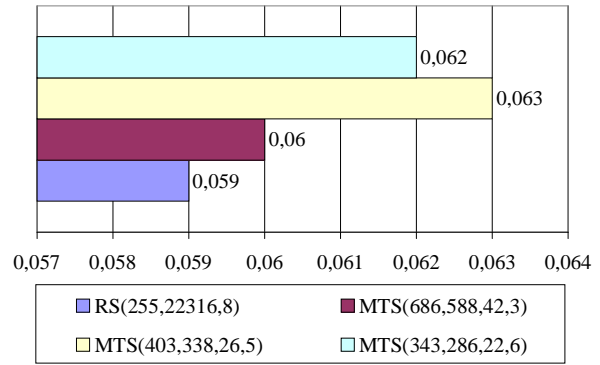


Figure 3 : Comparaison de la MTS avec les codes correcteurs de bursts.

4 Conclusion

La Transformée Mojette, que ce soit la MT appliquée sur des éléments binaires (MTB) ou la MTS, a des performances équivalentes avec les codes linéaires les plus courants. Si la MTB n'apporte pas un grand intérêt par rapport au code en blocs excepté celui de corriger une infinité d'erreurs consécutives, à l'inverse la MTS s'avère nettement plus performante en terme de correction que les codes RS. Cependant pour ce qui est de la MTB, les propriétés de la MT ne sont pas entièrement exploitées, le principe de correction restant à approfondir. Une perspective est de travailler sur la forme du support original car la corrélation suivant certaines projections reste insuffisante.

Références :

- [1] O.Papini and J.Wolfmann. Algèbre Discrète et Codes Correcteurs. *Springer-Verlag*, 1995.
- [2] W.W Peterson and E.J. Weldon Jr. Error-Correcting Codes. *M.I.T Press*, 1972.
- [3] N. Normand and J.P. Guédon. Transformation Mojette : une représentation redondante pour l'image. *Science Academy Reports (Paris)*, Series 1, theoretical computer science, pp 123-126, January 1998.
- [4] J.H.Wu and C.Olivier. Redundant transformation on the binary data and its applications to image coding and transmission. *2000 IEEE Asia-Pacific Conference on Circuits and Systems*, Tianjin, Chine, pp 763-766, Dec 2000.
- [5] J.P. Guédon and Y. Bizais. Band-limited and Haar filtering backprojection reconstruction. *IEEE Trans. Medical Imaging*, 13, pp. 430-440, September 1994.