Une technique de correction d'erreurs basée sur le CRC pour des systèmes codés en treillis contenant des bits de bourrage

Raoul PRÉVOST[1,2], Martial COULON[1], David BONACCI[2], Julia LEMAITRE[3], Jean-Pierre MILLERIOUX[3] et Jean-Yves TOURNERET[1]

[1]Université de Toulouse, INP-ENSEEIHT/IRIT, 2 rue Charles Camichel, BP 7122, 31071 Toulouse cedex 7, France

[2]TéSA, 14-16 Port Saint-Étienne, 31000 Toulouse, France

[3]CNES, 18 Avenue Edouard Belin, 31400 Toulouse, France

{raoul.prevost, david.bonacci}@tesa.prd.fr,
{martial.coulon, jean-yves.tourneret}@enseeiht.fr,
{julia.lemaitre, jean-pierre.millerioux}@cnes.fr

Résumé – Cet article présente une nouvelle stratégie de correction d'erreurs basée sur le contrôle de redondance cyclique (CRC) pour des systèmes codés en treillis contenant des bits de bourrage. Le récepteur proposé est conçu pour simultanément démoduler, décoder, et corriger les messages reçus en présence de bits de bourrage. Il s'appuie sur un algorithme de Viterbi utilisant des transitions conditionnelles et un treillis étendu approprié. Les performances de ce récepteur sont mesurées avec des messages du système d'identification automatique (AIS) incluant un CRC de 16 bits et utilisant la modulation à déplacement minimum gaussien (GMSK). Comme le définit la recommandation AIS, les bits de bourrage sont insérés après chaque séquence de cinq bits 1 consécutifs. Les résultats de simulation illustrant les performances de l'algorithme en terme de taux d'erreurs de paquets montrent un gain de plus de $2{,}5$ dB par rapport au récepteur GMSK conventionnel.

Abstract – This paper introduces a new error correction strategy using cyclic redundancy checks (CRC) for a trellis coded system in the presence of bit stuffing. The proposed receiver is designed to simultaneously demodulate, decode and correct the received message in the presence of bit stuffing. It is based on a Viterbi algorithm exploiting the conditional transitions of an appropriate extended trellis. The receiver is evaluated with automatic identification system (AIS) messages constructed with a 16 bit CRC and a Gaussian Minimum Shift Keying (GMSK) modulation. The stuffed bits are inserted after any sequence of five consecutive bits 1 as requested by the AIS recommendation. Simulation results illustrate the algorithm performance in terms of packet error rate. A gain of more than 2.5dB is obtained when compared to the conventional GMSK receiver.

1 Introduction

On s'intéresse dans cet article à la détection de messages de type AIS (*Automatic Identification System*) [1]. Le système AIS est un système de communications automatisées entre bateaux pour leur identification et leur localisation. On considère ici le cas où ces signaux sont captés par un satellite. Étant donnés les hauts niveaux de bruit et d'interférences, de nouvelles méthodes de correction d'erreurs doivent être développées afin d'obtenir des taux d'erreurs de paquets acceptables. Les contrôles de redondance cyclique (CRC) ont été initialement développés pour la détection d'erreurs. Cependant, plusieurs études ont proposé de les utiliser à des fins de correction. Notamment, la différence entre le CRC reçu et celui recalculé à partir des données est utilisée pour localiser une unique erreur de bit [2], ou en localiser deux [3]. Une méthode de correction de plusieurs bits erronés est proposée dans [4], tandis qu'une technique basée sur les probabilités d'erreurs de bits est présentée dans [5]. Une technique basée sur les codes convolutifs assistés d'un CRC est également étudiée dans [6]. Toutes les méthodes venant d'être citées ne fonctionnent pas lorsque les données contiennent potentiellement des bits de bourrage, dont la présence est aléatoire, car dépendant des bits d'information.

Cet article s'intéresse à la correction d'erreurs basée sur le CRC pour un signal contenant des bits de bourrage. Ces derniers servent à éviter la confusion entre les données utiles et les drapeaux de fin de trame, ou bien à créer des transitions supplémentaires. On retrouve ce mécanisme dans les bus USB (*universal serial bus*), le protocole HDLC (*high level data link control*), dans les systèmes X.25, ainsi que dans les systèmes RNIS et AIS. La méthode proposée ici n'est cependant restreinte à aucune de ces applications, et peut s'envisager pour tout système incluant un CRC et des bits de bourrage. Celle-ci est basée sur une utilisation particulière de l'algorithme de Viterbi, en développant un treillis étendu et en utilisant des transitions conditionnelles. Notons que cette méthode a fait l'objet de la soumission de deux brevets [7, 8].

L'article est organisé de la façon suivante : les sections 2 et 3 présentent les concepts du CRC et des bits de bourrage. Le récepteur proposé est développé en section 4, suivi par les résultats de simulation en section 5. Enfin, les conclusions sont données en section 6.

*Les auteurs souhaitent remercier la DGA et le CNES pour le financement.

2 Contrôle de Redondance Cyclique

Le CRC est une fonction de hachage qui est généralement utilisée pour détecter les erreurs dans les communications de données. Il s'agit d'une séquence binaire de longueur fixe calculée à partir des données à transmettre et envoyée jointe à ces dernières. Le récepteur peut calculer le CRC des données reçues et le comparer au CRC reçu afin de détecter des erreurs de transmission.

Il est bien connu que le CRC est défini comme le reste de la division par un polynôme générateur d'un polynôme engendré par les données. Il est important de noter que le calcul du CRC peut s'effectuer de façon itérative, à chaque nouveau bit considéré : ce calcul itératif est à la base du treillis utilisé dans la méthode proposée et est illustré sur la figure 1.

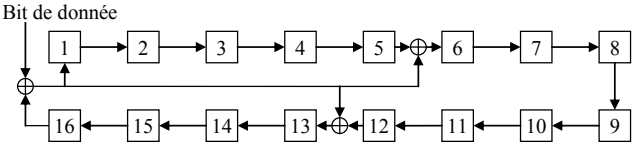

FIGURE 1 – Exemple de calcul de CRC iteratif de polynôme générateur $G(x) = x^{16} + x^{12} + x^5 + 1$. Les $\oplus$ représentent des *ou exclusifs* et sont placés en fonction du polynôme générateur. Les cases numérotées contiennent les bits du CRC dérivé et doivent être initialisées d'après le standard de CRC utilisé.

De plus, plutôt que de déterminer le CRC sur les données reçues, et de le comparer au CRC transmis, une approche équivalente classique consiste à calculer un CRC joint sur un message formé par les données et le CRC de ces données. Aucune erreur n'est détectée lorsque ce CRC joint est nul, c.-à-d., lorsque

$$\mathrm{CRC}([\text{Données}, \mathrm{CRC}(\text{Données})]) = 0. \qquad (1)$$

3 Bits de bourrage

Dans certains systèmes de transmission, des bits de bourrage sont insérés dans les bits de données afin de limiter le nombre de bits consécutifs de même valeur. Les nouvelles transitions ainsi générées permettent de resynchroniser l'horloge du récepteur, ou d'éviter des séquences binaires spécifiques. Ainsi, dans le protocole HDLC, un bit 0 (appelé bit de bourrage) est inséré après une séquence de cinq bits 1 afin que les données utiles ne soient pas confondues avec le drapeau de fin, constitué de deux bits 0 de part et d'autre de six bits 1 (comme cela est illustré sur la figure 2). On supposera ici, pour simplifier la présentation, que les bits de bourrage sont toujours des bits 0, comme dans les standards HDLC et AIS.

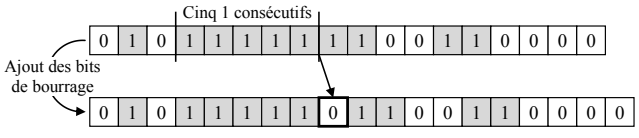

FIGURE 2 – Principe des bits de bourrage dans HDLC : un bit 0 est inséré après chaque séquence de cinq bits 1 consécutifs.

4 Méthode de correction proposée

4.1 Principe général

La méthode de correction proposée s'appuie sur un algorithme de Viterbi modifié. Celui-ci utilise un treillis étendu spécialement conçu pour prendre en considération le CRC. Notons que l'approche présentée ici peut être considérée comme une technique particulière de décodage source/canal conjoint [9], où le CRC constituerait le codage source, et la partie treillis de codage constituerait le codage canal. Des transitions conditionnelles sont ajoutées au treillis afin de prendre en compte la présence de bits de bourrage. Ainsi, tout comme l'algorithme de Viterbi, l'algorithme proposé tente de minimiser la distance entre la suite de symboles estimée et le signal reçu, mais ceci en respectant deux contraintes : 1) le CRC joint satisfait (1) et 2) le nombre de bits 1 consécutifs ne peut excéder une limite $\overline{P}$ définie par le système.

4.2 Construction du treillis étendu

Le récepteur proposé utilise donc un treillis constitué d'états étendus, formés chacun par un état de CRC et un état de codage treillis. Le CRC considéré est le CRC joint mentionné dans la partie 2. En effet, celui-ci pouvant se calculer de façon itérative, il est possible d'initialiser le CRC selon la spécification du standard (par exemple, que des 1 dans le cas de l'AIS), et de le remettre à jour à chaque nouveau bit reçu. Les valeurs intermédiaires du CRC peuvent alors être considérées comme les états du treillis. Un état de CRC est alors couplé à un état du codage treillis pour former un état étendu du nouveau treillis, appelé alors treillis étendu. Par exemple, si un état α du treillis de codage (TC) est suivi d'un état β (resp., d'un état γ) quand un bit 0 (resp. un bit 1) est transmis, et qu'un état de CRC A est suivi d'un état de CRC B (resp. C) quand le bit 0 (resp. 1) est transmis, alors un état étendu $(A;\alpha)$ est suivi d'un état étendu $(B;\beta)$ (resp., $(C;\gamma)$) lors de la transmission d'un bit 0 (resp. 1). Ce principe d'état étendu est illustré ci-dessous, où l'entier k représente le numéro du symbole reçu :

État de CRC		État du TC		État étendu
$A \xrightarrow{0}_{k}{}^{k+1} B$	&	$\alpha \xrightarrow{0}_{k}{}^{k+1} \beta$	$\Rightarrow$	$(A;\alpha) \xrightarrow{0}_{k}{}^{k+1} (B;\beta)$
$A \xrightarrow{1} C$		$\alpha \xrightarrow{1} \gamma$		$(A;\alpha) \xrightarrow{1} (C;\gamma)$

On note $\Gamma[k,(A;\alpha)]$ la distance entre le signal reçu et la suite de k symboles menant à l'état étendu $(A;\alpha)$ à l'instant k. D'autre part, la variable de transition $\Gamma_{\textit{trans}}[k,(A;\alpha),t]$ est définie comme la somme de $\Gamma[k,(A;\alpha)]$ et de la distance entre le symbole reçu à l'instant $k+1$ et le symbole venant de l'état étendu $(A;\alpha)$ contenant le bit t.

4.3 Prise en considération des bits de bourrage

Pour prendre en compte la présence possible de bits de bourrage, des transitions particulières sont introduites dans le treillis étendu, qui ne sont utilisées que lorsqu'un bit de bourrage est reçu. Ce bit est alors pris en compte dans l'évolution de l'état

du codage en treillis, mais sans modifier l'état du CRC. Ces transitions sont illustrées ci-dessous, où un bit de bourrage 0 est noté ***BB*** :

$$\begin{array}{cccc} \text{État de CRC} & & \text{État du TC} & \text{État étendu} \\ {\scriptstyle k}\quad{\scriptstyle k+1} & & {\scriptstyle k}\quad{\scriptstyle k+1} & {\scriptstyle k}\quad\quad{\scriptstyle k+1} \\ A \xrightarrow{0} B & & \alpha \xrightarrow{0} \beta & (A;\alpha) \xrightarrow{0} (B;\beta) \\ A \xrightarrow{1} C & \& & \alpha \xrightarrow{1} \gamma & \Rightarrow \quad (A;\alpha) \xrightarrow{1} (C;\gamma) \\ A \xrightarrow{BB} A & & \alpha \xrightarrow{BB} \beta & (A;\alpha) \xrightarrow{BB} (A;\beta) \end{array}$$

Pour déterminer si un bit reçu est un bit de bourrage, une variable $P[k,(A;\alpha)]$ est attachée à chaque état, qui indique le nombre de bits 1 consécutifs reçus avant d'atteindre l'état étendu $(A;\alpha)$ à l'instant k. Si $P[k,(A;\alpha)] = \overline{P}$, où $\overline{P}$ est une valeur maximale spécifiée par le standard ($\overline{P} = 5$ pour HDLC et AIS), un bit de bourrage est détecté, et la seule transition possible à partir de $(A;\alpha)$ est la transition $(A;\alpha) \xrightarrow{BB} (A;\beta)$. Après celle-ci, $P[k+1,(A;\beta)]$ prend la valeur 0, puisque le bit reçu est alors le bit de bourrage, égal à 0. Cette procédure est illustrée ci-dessous, où $\not\rightarrow$ représente une transition impossible (un bit d'information ne peut être un bit de bourrage, et vice-versa), ce qui s'obtient en attribuant une valeur infinie à la distance correspondant à cette transition.

$$\begin{array}{cc} \text{Bit d'information} & \text{Bit de bourrage} \\ {\scriptstyle k}\quad\quad{\scriptstyle k+1} & {\scriptstyle k}\quad\quad{\scriptstyle k+1} \\ (A;\alpha)_{P=3} \xrightarrow{0} (B;\beta)_{P=0} & (A;\alpha)_{P=5} \not\xrightarrow{0} (B;\beta) \\ (A;\alpha)_{P=3} \xrightarrow{1} (C;\gamma)_{P=4} & (A;\alpha)_{P=5} \not\xrightarrow{1} (C;\gamma) \\ (A;\alpha)_{P=3} \not\xrightarrow{BB} (A;\beta) & (A;\alpha)_{P=5} \xrightarrow{BB} (A;\beta)_{P=0} \end{array}$$

On associe également à chaque état une variable $S[k,(A;\alpha)]$ qui représente le nombre de bits de bourrage reçus avant d'atteindre l'état à l'instant k. Cette variable permet de déterminer le nombre de bits d'information contenus dans la trame reçue. De plus, tout comme la variable $\Gamma_{\textit{trans}}[k,(A;\alpha),t]$, on associe à chaque transition des variables $P_{\text{trans}}[k,(A;\alpha),t]$ et $S_{\text{trans}}[k,(A;\alpha),t]$, qui représentent l'évolution de $P[k,(A;\alpha)]$ et de $S[k,(A;\alpha)]$ lorsque le bit t est reçu (t peut être un bit d'information ou de bourrage).

4.4 Choix de l'état final

Le chemin sélectionné dans le treillis doit correspondre à un CRC final nul, conformément à (1). Cependant, l'état final du TC, noté $\boldsymbol{\theta}^f_{TC}$, est inconnu. De plus, le nombre K de symboles d'information effectivement reçus dans le signal est également inconnu à cause de la présence d'éventuels bits de bourrage. Il faut donc estimer conjointement K et $\boldsymbol{\theta}^f_{TC}$, ce qui s'obtient en minimisant la distance globale sur le treillis étendu $\Gamma[K,(0;\boldsymbol{\theta}^f_{TC})]$ pour toutes les valeurs possibles de K et de $\boldsymbol{\theta}^f_{TC}$, c'est-à-dire

$$\left(\widehat{K},\widehat{\boldsymbol{\theta}}^f_{TC}\right) = \underset{K,\boldsymbol{\theta}^f_{TC}}{\arg\min}\, \Gamma[K,(0;\boldsymbol{\theta}^f_{TC})]$$

sous les contraintes :

$$S_{\min} \leq S[K,(0;\boldsymbol{\theta}^f_{TC})] \leq S_{\max}$$
$$N_{\min} \leq K - S[K,(0;\boldsymbol{\theta}^f_{TC})] \leq N_{\max},$$

où les bornes $N_{\min}$ et $N_{\max}$ (resp. $S_{\min}$ et $S_{\max}$) sont les valeurs minimale et maximale du nombre de bits d'information (resp. de bits de bourrage). Par ex., $N_{\min} = N_{\max} = 184$, $S_{\min} = 0$ et $S_{\max} = 4$ pour le système AIS. $\boldsymbol{\theta}^f_{TC}$ peut prendre toute valeur des états du treillis de codage, tandis que K prend ses valeurs dans l'ensemble $\{N_{\min}+S_{\min},\ldots,N_{\max}+S_{\max}\}$. Les contraintes ci-dessus assurent que les nombres de bits d'information et de bourrage respectent ces bornes.

5 Simulations

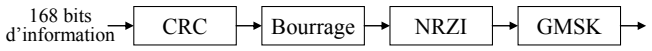

FIGURE 3 – Modèle d'émetteur.

Cette section présente quelques résultats de simulation obtenus pour le système AIS dont le modèle d'émetteur est illustré sur la figure 3. Les détails de l'algorithme présenté dans la section 4 sont explicités dans [10]. Les données des messages ont une longueur fixe de 168 bits auxquels est concaténé le CRC de 16 bits. Après l'insertion des bits de bourrage, la trame est encodée en NRZI (*Non Return to Zero Inverted*) puis modulée en GMSK avec le paramètre $BT = 0{,}4$ et une valeur de troncature de 3 (on note que ces paramètres sont connus du récepteur). Ces simulations considèrent une démodulation parfaite avec une récupération de la porteuse et une synchronisation idéales. Dans ce modèle, le codage en treillis est constitué du codage NRZI suivi de la modulation GMSK. Le polynôme générateur utilisé pour le calcul du CRC est $G(x) = x^{16} + x^{12} + x^5 + 1$ (spécifié par la recommandation de l'AIS) et un canal à bruit blanc gaussien additif (AWGN) est utilisé pour simuler une communication par satellite. On note que les méthodes [2]–[6] mentionnées en introduction ne peuvent être appliquées en présence de bits de bourrage. La méthode proposée ne peut donc pas être comparée à celles-ci. À la place, on la compare à un récepteur conventionnel basé sur un démodulateur GMSK cohérent utilisant l'algorithme de Viterbi. Son décodage NRZI est réalisé comme suit : deux valeurs consécutives identiques donnent un bit 1 alors que deux valeurs consécutives différentes donnent un bit 0. De plus, le bit suivant chaque séquence de cinq bits 1 est retiré afin de supprimer les bits de bourrage.

Dans le système AIS, le critère de performance le plus important est le taux d'erreurs de paquets (TEP). La figure 4 fait apparaître un gain supérieur à 2,5 dB entre l'algorithme proposé et le récepteur de référence pour un TEP ciblé de 0,1. Ce gain est encore plus important lorsque le TEP diminue. Les résultats présentés dans la figure 5 montrent que le récepteur proposé est également meilleur que le récepteur GMSK conventionnel en terme de taux d'erreurs de bits (TEB).

Le TEB des paquets erronés, noté TEBE, peut être déterminé en utilisant la relation suivante

$$\textit{TEBE} = \frac{\textit{TEB}}{\textit{TEP}}. \qquad (2)$$

Comme le montre la figure 6, le TEBE du récepteur proposé est bien supérieur à celui du récepteur GMSK conventionnel.

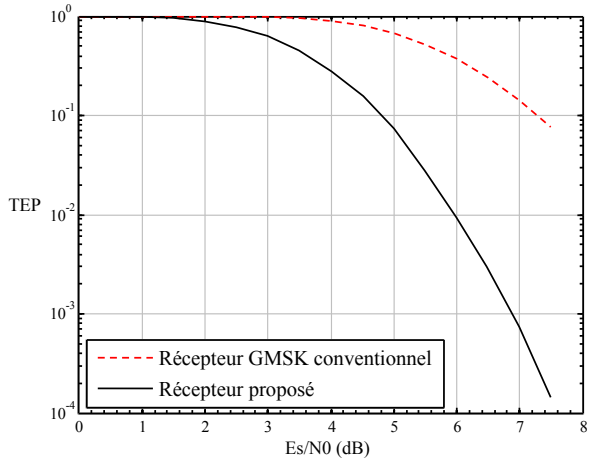

FIGURE 4 – Taux d'erreurs de paquets pour l'algorithme proposé et le récepteur GMSK conventionnel.

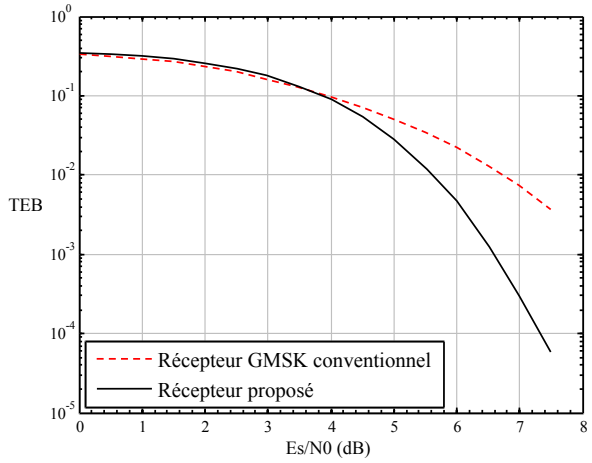

FIGURE 5 – Taux d'erreurs de bits pour l'algorithme proposé et le récepteur GMSK conventionnel.

Il est même proche de $0{,}5$, ce qui indique que les bits erronés sont concentrés dans un faible nombre de paquets : les paquets contenant moins d'erreurs de bits sont pour la plupart corrigés par l'algorithme proposé, ce qui n'est pas le cas pour le récepteur GMSK conventionnel. Cette concentration rend possible l'utilisation des méthodes basées sur la cohérence des données pour déterminer si un paquet contient ou non des erreurs.

6 Conclusion

Cet article a présenté une nouvelle stratégie de correction d'erreurs utilisant les contrôles de redondance cyclique (CRC) pour des systèmes codés en treillis. Ce système de correction permet de prendre en considération toute la redondance présente dans chaque message. Un nouveau type de méthode de correction a également été introduit permettant de compenser la présence des bits de bourrage qui ont été insérés entre le calcul du CRC et codage en treillis. Les résultats de simulation illustrent les performances de l'algorithme en terme de taux

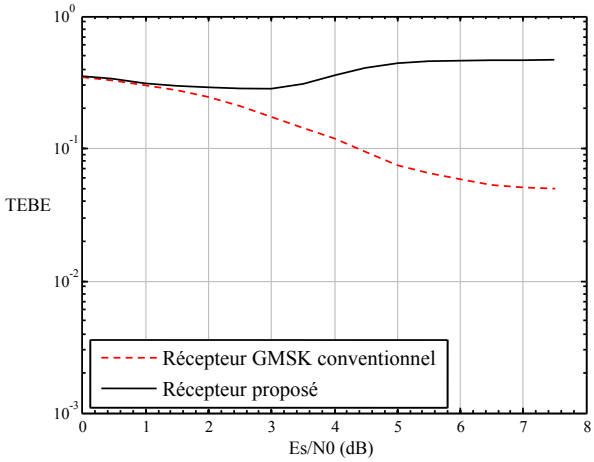

FIGURE 6 – Taux d'erreurs de bits des paquets erronés pour l'algorithme proposé et le récepteur GMSK conventionnel.

d'erreurs de paquets. Un gain supérieur à $2{,}5$ dB a été obtenu en comparaison du récepteur GMSK conventionnel pour un taux d'erreurs de paquets de $TEP = 0{,}1$. Une extension de l'approche proposée pour le cas multiutilisateur [11], ainsi que la poursuite de phase lorsque celle-ci est supposée inconnue sont actuellement en cours d'étude.

Références

[1] RECOMMANDATION ITU-R M.1371, « Technical characteristics for a universal automatic identification system using time division multiple access in the VHF maritime mobile band », ITU, 2001.

[2] B. MCDANIEL, « An algorithm for error correcting cyclic redundance checks », *C/C++ Users Journal*, p. 6, 2003.

[3] S. BABAIE, A. K. ZADEH, S. H. ES-HAGI et N. J. NAVIMIPOUR, « Double bits error correction using CRC method », *in Proc. Int. Conf. Semantics, Knowledge and Grid, Zhuhai, Chine*, no. 5, p. 254–257, 2009.

[4] C. SHI-YI et L. YU-BAI, « Error correcting cyclic redundancy checks based on confidence declaration », *in Proc. ITS Telecommunications, Chengdu, Chine*, no. 6, p. 511–514, 2006.

[5] Y. ZHANG et Q. YUAN, « A multiple bits error correction method based on cyclic redundancy check codes », *ICSP Signal Processing*, no. 9, p. 1808–1810, 2008.

[6] R. WANG, W. ZHAO et G. B. GIANNAKIS, « CRC-assisted error correction in a convolutionally coded system », *IEEE Trans. Comm.*, vol. 56, no. 11, p. 1807–1815, 2008.

[7] R. PRÉVOST, D. BONACCI, M. COULON, J. LEMAITRE, J.-P. MILLERIOUX et J.-Y. TOURNERET, « Multi-encodage error correction with extended trellis ». Brevet déposé.

[8] R. PRÉVOST, D. BONACCI, M. COULON, J. LEMAITRE, J.-P. MILLERIOUX et J.-Y. TOURNERET, « A Viterbi algorithm with conditional transitions ». Brevet déposé.

[9] H. JÉGOU, S. MALINOWSKI et C. GUILLEMOT, « Décodage conjoint source/canal par aggrégation d'états et décodage multi-treillis », *in Proc. CORESA Compression et REprésentation des Signaux Audiovisuels, Rennes, France*, 2005.

[10] R. PRÉVOST, D. BONACCI, M. COULON, J. LEMAITRE, J.-P. MILLERIOUX et J.-Y. TOURNERET, « CRC-Assisted Error Correction in a Trellis Coded System with Bit Stuffing », *in Proc. IEEE Workshop on Stat. Signal Processing, Nice, France*, juin 2011.

[11] R. PRÉVOST, M. COULON, D. BONACCI, J. LEMAITRE, J.-P. MILLERIOUX et J.-Y. TOURNERET, « An interference mitigation and error correction method based on cyclic redundancy check for trellis coded system involving bit stuffing », *in Proc. IEEE Global Comm. – Sat & Space Comm.*, 2011. Soumis.