COLLOQUE NATIONAL SUR LE TRAITEMENT DU SIGNAL
ET SES APPLICATIONS

NICE du 16 au 21 JUIN 75

REINTERPRETATION ET GENERALISATIONS DU DECODAGE A SEUIL PONDERE DES CODES RECURRENTS

Gérard BATTAIL et Martine DECOUVELAERE

E.N.S.T. - 46 rue Barrault - 75634 PARIS CEDEX 13

RESUME

Les principaux procédés de décodage des codes récurrents : l'algorithme de Viterbi et le décodage séquentiel, sont universels, reposent sur une description graphique du code (treillis ou arbre) et se prêtent à une pondération tenant compte des probabilités a priori des symboles reçus. Le décodage à seuil, d'une réalisation simple, permet aussi une pondération, mais n'est applicable qu'à une classe particulière de codes et ses performances sont limitées. Sur des voies où les erreurs sont groupées d'une manière mal définie, il permet de combiner codage et entrelacement sans accroissement de complexité, à la différence des algorithmes précités.

On interprète le décodage à seuil à l'aide de la notion de "répliques" : le code permet de calculer chacun des symboles utiles en fonction de plusieurs ensembles disjoints de symboles reçus. Dans le cas binaire, la décision à vraisemblance maximale à partir de toutes les répliques indépendantes disponibles s'exprime simplement. On retrouve le décodage à seuil pondéré, mais quelques extensions : pondération du décodage à seuil réfléchi et itération sont introcuits et améliorent le résultat. La règle de décision s'applique en outre à la répétition et la diversité ainsi qu'à leur combinaison avec le décodage à seuil (exemples).

Le manque de généralité du décodage à seuil et surtout la contradiction inhérente à l'emploi dans l'itération de la règle de décision inchangée, alors que les décisions introduisent une certaine dépendance, conduisent à s'intéresser aux répliques non indépendantes. La règle de décision devient complexe dans le cas général, mais des approximations peuvent être employées.

SUMMARY

The main probabilistic decoding processes of convolutional codes, namely Viterbi algorithm and sequential decoding, apply to any code, rely on a graph (treillis or tree) describing the code and enable weighting the received symbols in term of their a priori probability. On the other hand threshold decoding is easy to implement, but is restricted to a particular class of codes and its performance is limited. When used on channels where the errors are clustered as ill-defined bursts, it enables combining encoding and interleaving without increased complexity, at variance with the aforementioned algorithms.

Threshold decoding is interpreted using the concept of "replica" : the code enables computing each information symbol in term of several disjoint sets of received symbols. In the binary case, the maximum likelihood decision rule, given all available independent replicas, is very simple. Conventional weighted threshold decoding results, but extensions : weighted-feedback decoding and iteration, are introduced and improve the performance. Moreover, the decision rule also applies to repetition or diversity, and to their joint use with threshold decoding (examples).

The lack of generality of threshold decoding and, moreover, the contradiction germane to using the decision rule without change, in the iteration, whereas decisions themselves result in some dependence, prompted us to consider the case of non-independent replicas. The decision rule becomes quite complex in the general case, but useful approximations can be derived.

REINTERPRETATION ET GENERALISATIONS DU DECODAGE A SEUIL PONDERE DES CODES RECURRENTS

1. INTRODUCTION

Cet exposé est consacré à des procédés de décodage pondéré de codes récurrents. Rappelons que l'on désigne par code récurrent (appelé le plus souvent en anglais "convolutional") un procédé qui associe à une séquence numérique d'entrée (dite "d'information") une séquence numérique redondante telle que, la séquence d'information étant découpée en mots de k_o symboles, il correspond à chacun d'eux dans la séquence de sortie un mot de n_o symboles ($n_o > k_o$), fonction du mot d'information considéré et des m_o mots qui l'ont précédé. L'entier m_o mesure donc la mémoire du codeur ; le cas particulier $m_o = 0$ définit la classe des codes "en blocs". Nous supposerons dans toute la suite que m_o est différent de 0.

La modulation transforme les symboles de la séquence codée en signaux continus pouvant être transmis. Les perturbations se manifestent aussi de manière continue, par exemple, dans le cas le plus classique, par addition de bruit gaussien indépendant du signal. On peut à la réception prendre une décision définitive sur chacun des symboles, et le décodeur reçoit une suite de symboles appartenant au même alphabet que les symboles émis. On peut aussi alimenter le décodeur avec l'ensemble des valeurs, par exemple les sorties échantillonnées des filtres adaptés, qui constitue la réponse du démodulateur en amont de la prise de décision. Le procédé de décodage opère alors sur des grandeurs continues ; il est dit pondéré.

Une décision ferme implique une perte d'information et il est paradoxal de l'employer dans des systèmes destinés à protéger contre les perturbations (ainsi, dans le cas binaire antipodal avec bruit gaussien, il résulte de la décision ferme une perte de l'ordre de 2 dB sur le rapport signal à bruit). Bien entendu, la pondération accroît la complexité du décodage.

Nous considérerons deux familles d'algorithmes de décodage pondéré des codes récurrents :

- les algorithmes qui reposent sur une description graphique du code (arbre ou treillis) c'est-à-dire l'algorithme de Viterbi [9, 10] et le décodage séquentiel [11, 12] ;
- le décodage à seuil [1].

La première famille d'algorithmes s'applique à un code récurrent quelconque. Elle considère le codage comme le choix d'un chemin dans un graphe qui représente toutes les séquences codées possibles ; le décodage consiste à déterminer le chemin le plus vraisemblable (par rapport à la séquence reçue). L'algorithme de Viterbi est optimal au sens du maximum de vraisemblance. Sa probabilité d'erreur résiduelle tend théoriquement vers zéro quand la longueur de contrainte (ou mémoire) du codeur tend vers l'infini, mais la complexité du décodeur limite pratiquement l'emploi de l'algorithme à des codes courts.

Le décodage à seuil est très différent : il ne s'applique qu'à une classe restreinte de codes (les codes "orthogonalisables") ; la probabilité d'erreur résiduelle ne tend pas vers zéro quand la longueur du code croît indéfiniment et il n'est efficace que pour des codes de complexité faible ou moyenne. Son principal avantage est sa simplicité de mise en oeuvre ; il permet aussi, lorsqu'il est employé avec certains types de codes, tels que les codes diffus, d'offrir une protection contre des erreurs indépendantes aussi bien que des erreurs groupées, ce qui est avantageux dans beaucoup d'applications [13].

A titre d'exemple, nous allons considérer deux codes binaires qui nous serviront à illustrer : le premier, l'algorithme de Viterbi ; le second, le décodage à seuil.

La figure 1 représente le codeur correspondant au premier exemple, pour lequel $k_o = 1$, $n_o = 2$ et $m_o = 2$.

Les symboles binaires émis satisfont à l'équation

(1) $\quad x_o + x_2 + y_o + y_1 + y_2 = 0$

où $x_o \ldots$ et $y_o \ldots$ sont définies sur la figure, et où les additions sont effectuées modulo 2.

La figure 2 représente le treillis qui lui est associé, schéma où l'on a porté en ordonnée l'état du codeur à l'instant représenté par l'abscisse (l'état étant défini comme le contenu des $m_o = 2$ derniers étages du registre). Chaque "branche", joignant un point (dit "noeud") associé à un état à un instant donné, à un noeud à l'instant suivant, représente une transition possible entre deux états du codeur.

L'algorithme de Viterbi consiste à calculer la vraisemblance de chacun des chemins aboutissant à chacun des noeuds et à ne conserver pour chaque noeud que le plusvraisemblable, dit "survivant". Suffisamment en arrière des noeuds pour lesquels cette opération vient l'avoir lieu, il est très probable qu'il ne subsiste qu'un chemin ininterrompu, qui définit l'estimation à vraisemblance maximale de la séquence reçue.

Le volume des calculs nécessaires à la mise en oeuvre de l'algorithme dépend donc exponentiellement de la mémoire m_o du codeur. Or supposons que, pour obtenir un effet d'entrelacement (destiné à réduire la vulnérabilité du code aux erreurs groupées), on multiplie les indices dans (1) par une constante entière I. Le nombre des états se trouve alors porté à $2^{m_o I} = 4^I$: la croissance exponentielle de la complexité rend ra-

REINTERPRETATION ET GENERALISATIONS DU DECODAGE A SEUIL PONDERE DES CODES RECURRENTS

pidement l'algorithme inutilisable(x) ; pourtant, on s'attend à ce que cette simple transformation modifie peu les possibilités du code en présence d'erreurs indépendantes : elle laisse en particulier invariante la "distance limite minimale" ("minimum free distance"), caractéristique du code la plus significative à cet égard.

Les algorithmes de Viterbi et du décodage séquentiel exploitent la succession temporelle des branches du graphe, puisque l'objet considéré y est le chemin, suite ininterrompue de branches ; toute transformation qui modifie cette succession a une incidence sur la complexité des algorithmes.

On peut donc se demander si le chemin dans un graphe est le seul objet lié à la structure du code dont on puisse calculer la vraisemblance et permettant de ce fait de définir un algorithme de décodage pondéré.

Nous allons montrer qu'une réinterprétation du décodage à seuil [2] permet de définir un objet par rapport auquel des décisions à vraisemblance maximale peuvent être prises et possédant la propriété, souhaitée, d'invariance par rapport à un entrelacement par multiplication des indices.

De plus, il apparaît dans la perspective ainsi dégagée que les limitations du décodage à seuil proviennent d'une exploitation incomplète des données reçues. Des généralisations de l'algorithme sont donc possibles, améliorant sa probabilité d'erreur pour un code donné, son efficacité pour les codes longs et même levant la restriction à des codes orthogonalisables. Bien entendu, ces généralisations se font aux dépens de la simplicité initiale de l'algorithme. Nous nous restreindrons dans toute la suite à des codes récurrents binaires et à progression $k_o = 1$. Seuls seront donc considérés les codes, à forte redondance, ayant un taux d'émission de la forme $1/n_o$.

II. REINTERPRETATION DU DECODAGE A SEUIL

La figure 3 représente le dispositif de codage qui va servir d'exemple pour l'étude du décodage à seuil. Il s'agit d'un code systématique (les symboles d'information sont émis). Le codeur leur adjoint des symboles de contrôle également émis définis par :

$$(2) \quad p_7 = i_0 + i_1 + i_3 + i_7 ,$$

où i_0, i_1 ... sont les bits d'information mis en mémoire dans le registre ; les additions sont faites modulo 2.

(x) bien entendu, il reste la possibilité de combiner l'algorithme de Viterbi et un entrelacement distinct du codage.

Nous ne reprendrons pas la théorie classique du décodage à seuil [1], et allons montrer directement que le décodage du code choisi comme exemple (comme de n'importe quel code binaire orthogonalisable) équivaut à prendre une décision à partir d'un certain nombre de répliques indépendantes du bit i_0, en appelant réplique un symbole, reçu ou déduit de grandeurs reçues, qui serait identique à un symbole émis en l'absence d'erreur de transmission.

La relation (2) définissant le bit de contrôle reste évidemment vérifiée pour n'importe quel décalage de ses indices. Considérons en particulier tous les décalages faisant apparaître l'indice 0, et écrivons les relations obtenues en mettant i_0 au premier membre :

$$(3) \quad \begin{cases} i_0 = i_1 + i_3 + i_7 + p_7, \\ i_0 = i_{-1} + i_2 + i_6 + p_6, \\ i_0 = i_{-3} + i_{-2} + i_4 + p_4, \\ i_0 = i_{-7} + i_{-6} + i_{-4} + p_0. \end{cases}$$

A partir des bits reçus correspondants, qu'on affectera d'une prime pour les distinguer des bits émis, il est donc possible de former 5 répliques de i_0, à savoir
- i'_0 (réplique simple) ;
- les 4 répliques déduites de (3) en affectant d'une prime les bits des seconds membres.

Le choix du code, qui détermine les indices dans (2), est tel que tous les bits aux seconds membres de (3) sont différents (propriété d'"orthogonalité"). Si les erreurs sur les bits reçus sont indépendantes, l'indépendance des répliques en découle.

S'il est possible d'assigner une probabilité d'erreur à chacun des symboles reçus, le problème du décodage se ramène à formuler la règle de décision à vraisemblance maximale sur l'ensemble des répliques disponibles et à définir les moyens de la mettre en oeuvre, en tenant compte de l'expression (3) des répliques obtenues par combinaison de symboles reçus (dites "composées").

III. DECISION A VRAISEMBLANCE MAXIMALE SUR UN ENSEMBLE DE REPLIQUES INDEPENDANTES

La règle de décision à vraisemblance maximale à partir d'un ensemble de répliques va être établie directement, ce qui permettra d'introduire les conventions de notations et de vocabulaire employées dans la suite.

Le problème est, connaissant n répliques indépendantes d'un même symbole binaire b inconnu, de prendre une décision à vraisemblance maximale quant à b à partir de cet ensemble de répliques.

Supposons d'abord qu'à chacune des répliques b_1, b_2, ... b_n, on sache associer une probabilité d'erreur

REINTERPRETATION ET GENERALISATIONS DU DECODAGE A SEUIL PONDERE
DES CODES RECURRENTS

$p_1, p_2 \ldots p_n$. On définit la valeur relative a_i associée à la réplique b_i de la façon suivante :
- a_i est positive si $b_i = 0$, négative si $b_i = 1$;
- le module de a_i vaut $\text{Log}\frac{1-p_i}{p_i}$, et sera désigné par "coefficient de vraisemblance".

On appréciera l'importance et la signification "physique" de la notion de valeur relative (v. r.) en remarquant qu'en présence de bruit additif gaussien avec un système de modulation binaire antipodal, la v. r. d'un symbole reçu est proportionnelle à la grandeur qui, dans le récepteur optimal, détermine la décision. La disponibilité des v. r. des symboles reçus est donc plausible.

Soient deux répliques b_1 et b_2 d'un symbole b inconnu, de probabilités a priori de valoir 1 ou 0 égales, p_1 et p_2 leurs probabilités d'erreur ($p_i = Pr(b_i|b)$, $i = 1, 2$). Les probabilités conditionnelles que b vaille 0 ou 1 s'expriment par la règle de Bayes :

$$Pr(b|b_1,b_2) = \frac{Pr(b_1,b_2|b)\ Pr(b)}{Pr(b_1,b_2)}, \text{ pour } b = 0 \text{ ou } 1.$$

Le choix à vraisemblance maximale consiste à prendre pour valeur estimée b^* de b celle qui maximise

$$\frac{Pr(b|b_1, b_2)}{Pr(b)} = \frac{Pr(b_1, b_2|b)}{Pr(b_1,b_2)}.$$

Le dénominateur du second membre est indépendant du choix de b. Il suffit donc que b^* en maximise le numérateur, égal à $Pr(b_1|b)\ Pr(b_2|b)$ à cause de l'indépendance.

Considérons l'expression :

$$\text{Log}\frac{Pr(b_1,b_2|b=0)}{Pr(b_1,b_2|b=1)} = \text{Log}\frac{Pr(b_1|b=0)}{Pr(b_1|b=1)} + \text{Log}\frac{Pr(b_2|b=0)}{Pr(b_2|b=1)}$$

Le premier membre est, d'après la règle de Bayes, le logarithme du rapport des probabilités d'avoir $b = 0$ et $b = 1$, connaissant b_1 et b_2 : c'est la v. r. a de b^*.

Chacun des termes du second membre est le logarithme du rapport des probabilités que b vaille 0 ou 1, connaissant l'une des répliques reçues, rapport que la règle de Bayes permet d'exprimer comme celui des probabilités de réalisation de cette réplique conditionnellement à $b = 0$ et $b = 1$, ou v. r. a_1 et a_2 des répliques, donc :

$$a = a_1 + a_2.$$

L'additivité des valeurs relatives entraîne la généralisation à n répliques par associativité, d'où

$$(4) \quad a = a_1 + a_2 + \ldots + a_n .$$

Le signe de l'expression (4) représente donc la décision binaire (+ signifiant 0, - signifiant 1). La probabilité d'erreur après décision, soit p_r vaut :

$$(5) \quad p_r = 1/(1 + e^{|a|}).$$

On montre que la décision selon (4) améliore toujours la probabilité d'erreur résiduelle moyenne, quelle que soit la distribution de probabilité des v. r. des répliques.

IV. FORMULATION DE L'ALGORITHME DE DECODAGE PONDERE, QUELQUES EXEMPLES D'APPLICATION

L'application de la règle de décision (4) exige que l'on calcule la v. r. de répliques telles que (3), formées par addition modulo 2 de symboles reçus, en fonction de la v. r. de chacun d'eux.

D'après la définition de la v. r., si q est la probabilité qu'une variable binaire aléatoire vaille 1, on a :

$$(6) \quad \text{th}\,(a/2) = 1 - 2q.$$

La somme modulo 2 de deux telles variables indépendantes, valant 1 avec les probabilités respectives q_1 et q_2 vaut 1 avec la probabilité :

$$q = q_1\,(1 - q_2) + (1 - q_1)\,q_2.$$

Donc :

$$\text{th}\,(a/2) = 1-2|q_1(1-q_2) + (1-q_1)q_2$$
$$= (1-2q_1)\,(1-2q_2) = \text{th}\,(a_1/2)\ \text{th}\,(a_2/2).$$

La généralisation à m bits combinés est immédiate par associativité, d'où les expressions de la v. r. a de la somme modulo 2 de m bits indépendants de v. r. a_i, $i = 1, \ldots m$:

$$(7) \quad \text{th}\,(a/2) = \prod_1^m \text{th}\,(a_i/2),$$

ou :

$$(8) \quad a = \text{Log}\frac{1 + \prod_1^m \text{th}\,(a_i/2)}{1 - \prod_1^m \text{th}\,(a_i/2)}.$$

L'opération qui combine des bits reçus pour former une réplique composée dégrade la probabilité d'erreur par rapport à celle des bits constituants puisque $|a| < \min_i \{|a_i|\}$.

On peut alors formuler l'algorithme de décodage de la façon suivante : on associe à chacune des répliques "composées" (3) sa v. r. calculée selon (8), et on déduit de la somme (4) la décision (pondérée) sur i_0.

Il est clair que l'entrelacement par multiplication des indices modifie seulement l'adresse, dans la séquence reçue, des symboles dont la valeur relative est utilisée dans les calculs, sans incidence sur leur complexité.

La présence de bits d'information d'indice négatif dans les expressions (3) des répliques appelle un commentaire. Ces bits ont déjà fait l'objet d'une décision, dont il résulte une amélioration de la probabili-

té d'erreur ; il est logique d'utiliser leur v. r. a posteriori au lieu de leur v. r. initiale, dans le calcul par (7) ou (8) de la v. r. des répliques composées (malgré la dépendance que la décision même peut induire entre son résultat et les autres bits combinés).

La simulation montre qu'en procédant ainsi on améliore effectivement le résultat.

On montre sans difficulté que le procédé de décodage qui vient d'être décrit est presque équivalent au décodage à seuil pondéré [1] ; la forme que nous en avons donnée est plus facile à exposer aussi bien qu'à réaliser. La seule différence, qui est un avantage, réside dans l'utilisation de la v. r. a posteriori des bits d'information d'indice négatif.

Dans le dispositif de décodage décrit par Massey [1, p 62] le complément binaire d'un bit d'information d'indice négatif lui est substitué en cas de correction, mais la pondération n'est pas modifiée (feedback decoding").

Or, la plupart des erreurs résiduelles sont prises avec une faible marge ; les répliques contenant un bit d'indice négatif erroné ont en général un faible coefficient de vraisemblance et jouent de ce fait un faible rôle dans les décisions ultérieures. Nous désignerons par "décodage réfléchi pondéré" ou "rétropondération" la substitution de la v. r. résultant des décisions antérieures, qui est un résultat original de cette étude.

Avant d'aborder les généralisations du décodage a seuil, nous mentionnerons sans développement d'autres applications possibles de la décision à vraisemblance maximale à partir d'un ensemble de répliques. En effet, le schéma considéré s'applique aussi bien aux cas de "diversité" ou de répétition systématique. Les deux applications suivantes combinent répétition systématique et emploi d'un code orthogonalisable.

a) Répétition systématique

Dans de mauvaises conditions, la simple répétition a l'avantage de toujours améliorer le résultat, alors que des formes plus élaborées de codage perdent toute efficacité si la probabilité d'erreur dans la voie devient trop mauvaise.

On peut envisager de combiner la répétition et le codage en codant par un procédé récurrent orthogonalisable puis en répétant un nombre déterminé de fois. A la réception, la marge de décision sur chacun des bits répétés sert de coefficient de pondération pour le décodage à seuil. Ainsi la décision sur les données répétées n'est pas simplement binaire : la pondération réalisée conserve et exploite l'information recueillie.

b) Répétition sur demande

Il s'agit d'une variante des systèmes utilisant une voie de retour, dans laquelle on combine de même un code orthogonalisable et la répétition, mais celle-ci n'a plus lieu que sur demande. L'intérêt du procédé est le même que dans le cas précédent dans de très mauvaises conditions de probabilité d'erreur. La répétition à la demande permet d'adapter la vitesse de transmission, qui augmente jusqu'à n'être limitée que par la redondance du code quand la qualité de la voie s'améliore. Ce procédé est donc applicable à des conditions de transmission variables dans le temps et pouvant devenir très mauvaises telles qu'en radio en ondes longues, où la qualité dépend fortement des conditions orageuses [3].

V. GENERALISATION ITEREE DU DECODAGE A SEUIL PONDERE

L'effet favorable de la substitution des v. r. après décision suggère de recommencer le décodage après que toutes les décisions ont été prises. En effet, une décision substitue une estimation mieux informée (a posteriori) à l'estimation a priori de la v. r. initiale. L'estimation a posteriori s'est enrichie d'une information puisée dans le contexte, mais qui dépendait elle-même de la validité de l'estimation a priori. On va ainsi considérer la première décision comme provisoire, destinée à améliorer l'estimation des v. r. en vue d'une décision suivante ; ce processus est évidemment itératif.

On obtient alors, pour le code (2) le dispositif de la figure 4, dont l'itération constitue le décodeur [4].

Si l'itération apparaît comme logique (la simulation en confirme l'efficacité), son analyse est difficile. En effet, pourvu qu'elle soit initialement assez grande, la moyenne du module de la v. r. des symboles croît indéfiniment avec le rang de l'itération. L'application de la relation (5) conduit au résultat absurde d'une probabilité d'erreur résiduelle tendant vers zéro, alors que le code est de longueur finie, parce que l'algorithme néglige la dépendance entre les répliques induite par les décisions mêmes ; les grandeurs calculées s'écartent de plus en plus de la v. r. correspondant à la probabilité d'erreur résiduelle vraie, et l'itération cesse d'apporter une amélioration à partir de la troisième décision.

Nous n'avons pas trouvé de méthode d'analyse satisfaisante de l'itération et nous sommes donc contentés de la simuler.

La modulation a été supposée binaire antipodale avec addition de bruit stationnaire, gaussien et blanc, caractérisée par le rapport E_b/N_0, E_b étant l'énergie disponible à la réception par symbole binaire d'information, N_0 la densité spectrale unilatérale du bruit.

REINTERPRETATION ET GENERALISATIONS DU DECODAGE A SEUIL PONDERE DES CODES RECURRENTS

La simulation a consisté à générer des échantillons de bruit indépendants g, de densité de probabilité gaussienne et de variance prise comme unité, à les ajouter à des échantillons valant $s = (E_b/n_oN_0)^{1/2}$, représentant ceux du signal codé démodulé.

La v. r. a d'un symbole reçu est alors égale à :

(9) $a = 2(s + g)s$.

Le choix de s positif revient à supposer le message constitué d'une suite de zéros, comme la séquence codée, ce qui ne restreint pas la généralité (les codes sont linéaires). Seul le décodage simulé se réfère explicitement au code employé.

Pour que les résultats aient une signification statistique, on s'est restreint à des valeurs basses du rapport E_b/N_0 (de 2,5 à 5 dB) afin que les probabilités d'erreur résiduelles restent voisines de 10^{-2}. Le nombre de bits d'information généralement employé pour chaque valeur du rapport signal à bruit était de 35 000.

Les résultats de simulation sur le code (2) sont résumés sur la figure 5 : probabilité d'erreur résiduelle à la 1ère et 3ème décision, ainsi qu'en l'absence de codage (calculée par $P_{nc} = \frac{1}{2}\mathrm{erfc}\,(\sqrt{E_b/N_0})$, $\mathrm{erfc}(x) \triangleq \frac{2}{\sqrt{\pi}}\int_x^\infty e^{-t^2}dt$) et sa prévision selon Viterbi [14] pour le décodeur optimal dans le cas du code (1), comparable au code (2) en ce sens que leur distance limite commune D est 5, soit :

$P_{rV} \simeq \mathrm{erfc}(\sqrt{5E_b/2N_0})/2(1 - 2e^{-E_b/N_0})$.

Des essais ont été fait aussi avec une pondération grossièrement quantifiée, où les v. r. des symboles reçus étaient représentées par leur signe et leur module quantifié à 4 niveaux : 1, 3, 5 et 7 avec une unité arbitraire, les points de discontinuité ayant pour abscisses 0, $c = \sqrt{E_b/2N_0}$, 2c et 3c.

En outre, (8) a été remplacée par l'approximation très grossière (qui permet l'emploi d'une unité arbitraire) :

$$(10)\qquad \begin{aligned} |a| &= \min_{i\in I}(|a_i|), \\ \mathrm{sgn}(a) &= \mathrm{sgn}\left[\prod_{i\in I} a_i\right], \end{aligned}$$

où a_i représente la valeur relative d'un symbole et I l'ensemble des indices des symboles combinés pour former une réplique.

La dégradation mesurée, traduite en augmentation de E_b/N_0 à probabilité d'erreur résiduelle constante, est d'environ 0,25 dB, quel que soit le rang de la décision considérée dans l'itération.

VI. EXTENSION A DES REPLIQUES NON INDEPENDANTES

Nous avons pu donner du décodage à seuil pondéré de codes orthogonalisables une formulation simple, grâce à la notion de répliques indépendantes. Des généralisations ont été introduites, malgré la contradiction d'employer une règle de décision fondée sur l'indépendance des répliques dans des conditions où elle cesse d'être strictement réalisée. Par ailleurs, les codes orthogonalisables ne constituent qu'une classe particulière qui est loin de contenir tous les bons codes. Il a donc paru intéressant de généraliser la règle de décision aux répliques non indépendantes.

Les principaux problèmes qui se posent en vue de cette généralisation sont les suivants :

- caractériser la dépendance mutuelle de n répliques ;
- formuler la règle de décision quand n répliques dépendantes sont disponibles ;
- quand ces répliques résultent d'un calcul à partir de symboles reçus (par exemple, en fonction de résultats précédents, dans l'itération), exprimer numériquement leur dépendance.

VI.1. Caractériser la dépendance

Nous exprimerons n répliques d'une grandeur binaire b par :

$$(11)\qquad b_i = b + e_i + \sum_{j\neq i}^{C} e_{ij} + \sum_{j,k\neq i}^{C} e_{ijk} + \ldots + e_{12\ldots i\ldots n},$$

$i = 1, 2, \ldots n$,

où b_i est la i-ème réplique, où tous les "bits-erreurs" e_i, e_{ij}, etc sont des variables aléatoires binaires indépendantes, où les additions sont faites modulo 2, $\sum^{C}_{i,j\ldots}$ désignant la somme modulo 2 pour toutes les combinaisons sans répétition des indices i,j... (chacun variant de 1 à n). (Nous emploierons pour désigner le produit de termes ayant les mêmes indices le symbole $\prod^{C}_{i,j\ldots}$.)

Nous supposerons connues les probabilités que chacun des bits-erreurs vaille 0 ou 1.

En plus des v. r. a_i liées à (11), nous introduirons :

- des v. r. partielles u_i, associées à $b + e_i$ (a priori, donc pour b donné) ou $b_i + e_i$ (a posteriori) ;
- des coefficients de vraisemblance mutuels :

$$v_{ij\ldots} = \mathrm{Log}\,\frac{\Pr(e_{ij\ldots} = 0)}{\Pr(e_{ij\ldots} = 1)},$$

On déduit immédiatement de (11) :

$$(12)\qquad \mathrm{th}(a_i/2) = \mathrm{th}(u_i/2)\prod_j \mathrm{th}(v_{ij}/2)\prod_{j,k}^{C}\mathrm{th}(v_{ijk}/2)\ldots,$$

$i = 1, 2 \ldots n$.

On définit le polynôme énumérateur :

$$(13)\qquad E(e_1,e_2\ldots e_n) = \prod_1^n (q_i + p_i s_i)\prod_{i,j}^{C}(q_{ij} + p_{ij}s_i s_j)\prod_{i,j,k}^{C}(q_{ijk} + p_{ijk}s_i s_j s_k)\ldots$$

où $p_i = 1 - q_i = 1/(1 + e^{u_i})$,

$p_{ij\ldots} = 1 - q_{ij\ldots} = 1/(1 + e^{v_{ij\ldots}})$.

L'exposant de chaque indéterminée s_i est calculé modulo 2.

En interprétant les u_i a priori, la probabilité a priori d'obtenir pour le vecteur $(b_1, b_2, \ldots b_n)$ une configuration où les répliques $b_{\alpha_1}, b_{\alpha_2} \ldots$ valent 1, les autres valant 0, est le coefficient dans E du terme $s_{\alpha_1} s_{\alpha_2} \ldots$

En outre, la probabilité que certaines répliques, par exemple les k premières, vaillent conjointement 1, est le coefficient de $s_1 s_2 \ldots s_k$ dans E où les autres indéterminées ont été égalées à 1 (en interprétant les u_i a priori), et celle que les estimations de b déduites séparément de chacune des répliques b_1, $b_2 \ldots b_k$ vaillent conjointement 1 est égale au même coefficient en interprétant les u_i a posteriori.

VI.2. Extension de la règle de décision à des répliques non indépendantes

Considérons l'ensemble des n répliques reçues. La règle de décision à vraisemblance maximale à partir de ces répliques conduit à calculer

$$(14)\quad a_{(n)} = \mathrm{Log}\,\frac{\Pr(b=0 \mid b_1, b_2 \ldots b_n)}{\Pr(b=1 \mid b_1, b_2 \ldots b_n)}\ .$$

Récrivons (11) en mettant b dans les premiers membres et b_1, $b_2 \ldots b_n$ dans les seconds. Pour b_1, $b_2 \ldots b_n$ données, les probabilités en numérateur et dénominateur de (14) sont le terme constant et le coefficient de $s_1 s_2 \ldots s_n$ de E, en interprétant les u_i a posteriori.

On peut donner de la v. r. $a_{(n)}$ une expression approchée plus simple en divisant E par $\exp(\sum_c v_{ij\ldots})$, le symbole $\sum_c$ (ou $\prod_c$) désignant la somme (ou le produit) de tous les termes définis par les combinaisons 2 à 2, 3 à 3, ... n à n de leurs indices, et en traitant les exponentielles $\exp(-v_{ij\ldots})$ comme des infiniment petits dont on ne conserve que les termes du premier ordre.

Le polynôme ainsi transformé s'écrit :

$$(15)\quad E' \simeq \prod_i (e^{u_i} + s_i)\ (1 + \sum_c e^{-v_{ij\ldots}} s_i s_j \ldots)$$

par conséquent :

$$(16)\quad a_{(n)} \simeq \sum_i u_i + \mathrm{Log}\,\frac{1 + \sum_c e^{-v_{ij\ldots} - u_i - u_j - \ldots}}{1 + \sum_c e^{-v_{ij\ldots} + u_i + u_j + \ldots}}$$

où les u_i, u_j ... présentes dans chacun des exposants sont toutes les v. r. ayant pour indice l'un de ceux du coefficient de vraisemblance mutuel $v_{ij\ldots}$.

On retrouve donc une somme de v. r. (mais partielles) plus un terme correctif qui tient compte des erreurs communes.

La formulation de la règle de décision exacte à partir de coefficients du polynôme énumérateur résout le problème théorique. Sa version approchée (16) est d'un emploi plus facile. Mais il reste que l'application de cette règle dans un algorithme pratique requiert la détermination des grandeurs u_i et $v_{ij\ldots}$, problème que nous allons maintenant examiner.

VI.3. Le problème de la détermination des paramètres caractérisant la dépendance

Considérons la suite des symboles obtenus à une étape de l'itération. A l'étape suivante, les décisions devront être prises à partir de répliques formées pour la plupart par combinaison des résultats de décisions précédentes. Par conséquent, la dépendance entre les nouvelles répliques se manifeste de façon indirecte, du fait que les symboles combinés pour les former dépendent des répliques de l'étape précédente, elles-mêmes obtenues par combinaison de symboles dont certains sont communs (voir la figure 6).

On remarquera le caractère "diffus" de cette dépendance d'autant que, dans une combinaison de symboles constituant une réplique, les termes dominants ont les plus petits coefficients de vraisemblance et son résultat en a un plus petit que les termes combinés ; au contraire, dans la règle de décision, les termes de grande valeur absolue contribuent le plus au résultat.

Les calculs des grandeurs u_i et $v_{ij\ldots}$ ne peuvent faute de place être reproduits ici. L'étape la plus difficile en est l'expression de la dépendance entre le résultat d'une décision et les répliques qui ont été utilisées pour l'obtenir. La complexité de (16) suggère la difficulté de cette tâche ; même avec deux répliques initiales indépendantes, on aboutit déjà à une expression compliquée : les calculs qui détermineraient les paramètres caractérisant la dépendance mutuelle seraient inutilisables dans un algorithme pratique.

Nous avons renoncé à déterminer chaque valeur particulière de $v_{ij\ldots}$ en fonction des données, en la remplaçant par une estimation de sa valeur moyenne. Il est clair qu'en procédant ainsi, on n'exploite pas entièrement l'information a priori mais qu'on introduit dans (16) un terme dont l'effet est favorable même si une meilleure évaluation de la dépendance en permettrait un calcul plus précis.

La transformation de l'algorithme de décodage peut alors être schématisée ainsi.

On dispose des v. r. a_i des répliques à une certaine étape de l'itération et des coefficients de vraisemblance mutuels moyens $v_{ij\ldots}$ caractérisant leur dépendance à cette étape. On calcule alors u_i correspon-

dant à a_i par la relation (12). (Il est essentiel à ce stade que les estimations des v_{ij} soient plausibles, en ce sens que le rapport exprimant $th(u_i/2)$ soit inférieur à 1).

L'application de la règle de décision (16) est faite alors et donne une évaluation de la valeur relative a du symbole dont on possédait les répliques.

Si l'itération doit être poursuivie, les v. r. des répliques doivent à nouveau être calculées. On admettra par exemple que les v. r. résultant de la règle de décision (16) sont utilisées directement dans (8) (négligeant l'effet de la dépendance, c'est ici pessimiste).

Une fois la v.r. a_i des répliques déterminée, il faudra appliquer à nouveau (12) puis la règle de décision (16), en employant dans ces deux opérations une nouvelle évaluation des coefficients $v_{ij...}$, pertinente à cette étape de l'itération.

Pour un code déterminé, l'évaluation des $v_{ij...}$ moyens peut être faite une fois pour toutes en fonction de données expérimentales ou de simulation. On disposerait ainsi, par exemple, de courbes donnant les $v_{ij...}$ en fonction de rapport signal à bruit normalisé ou de tout autre paramètre caractérisant la qualité de la transmission, après chaque étape de l'itération.

VII. DONNEES BIBLIOGRAPHIQUES ET REFERENCES

La principale référence sur le décodage à seuil est le livre de Massey [1]. L'emploi de la notion de réplique a été proposée par Rudolph [2]. La formalisation de la règle de décision que nous avons employée avait été introduite par Pierce [5] dans un contexte différent. L'idée d'une itération du décodage à seuil est à notre connaissance due à Alexis [6], pour le décodage majoritaire.

L'étude de Blizard et Korgel [7], pour autant que nous en puissions juger, décrit un algorithme équivalent à notre algorithme d'itération. Cet article souligne de plus la parenté entre le décodage à seuil itéré et l'algorithme de Gallager de décodage des codes de matrice de contrôle à faible densité de uns [8]. En fait, nos relations (4) et (8) sont équivalentes à la formule (4.6), page 46, de son livre.

[1] J.L. MASSEY, 'Threshold decoding", The MIT Press, Cambridge Mass., 1963.

[2] L.D. RUDOLPH, "A class of majority-logic decodable codes", IEEE Trans.on Information Theory, IT-13 n°2, Avril 1967, pp.305-307.

[3] Demande de brevet français, "Systèmes de transmission de données binaires fortement protégé contre les erreurs", 19 Février 1974, n°7405592.

[4] Demande de brevet français, "Perfectionnements aux dispositifs de décodage à seuil des codes récurrents", 7 Janvier 1972, n°7200497 ; 1ère addition déposée le 8 Décembre 1972, n°7243744.

[5] W.H. PIERCE, "Adaptive vote-takers improve the use of redundancy", in "Redondancy techniques for computing systems", R.H. Wilcox et W.C. Mann, Eds., Spartan books, Washington D.C., 1962, symposium du 6-7 Février 1962, pp.229-250.

[6] R.P.J. ALEXIS, "Codeur et décodeur à seuils multiples et autocorrecteur d'erreurs pour système de transmission d'information", Demande de brevet français du 10 Octobre 1969, n° de publication : 2 062 884.

[7] R.B. BLIZARD et C.C. KORGEL, "An iterative probabilistic threshold decoding technique", IEEE Nat. Telecommunications Conf. 4-6 Décembre 1972, Houston, Tex., NTC '72 Record, pp.13D-1 - 13D-5.

[8] R.G. GALLAGER, "Low-density parity-check codes", The MIT Press, Cambridge, Mass., 1963.

[9] A.J. VITERBI, "Error bounds for convolutional codes and an asymptotically optimum decoding algorithm", IEEE Trans. on Inf. Th., IT-13 n°2, Avril 1967, pp.260-269.

[10] G.D. FORNEY Jr., "The Viterbi algorithm", Proc. IEEE, 61 n°3, Mars 1973, pp.268-278.

[11] J.M. WOZENCRAFT et B. REIFFEN, "Sequential decoding", The technology Press et John Wiley & Sons, Inc., New York, 1961.

[12] R.M. FANO, "A heuristic discussion of probabilistic decoding", IEEE Trans. on Inf. Th., IT-9 n°2, Avril 1963, pp.64-74.

[13] A. KOHLENBERG et G.D. FORNEY Jr, "Convolutional coding for channels with memory", IEEE Trans. IT-14 n°5, Septembre 1968; pp.618-626.

[14] A.J. VITERBI, "Convolutional codes and their performance in communication systems", IEEE Trans. COM-19 n°5, Octobre 1971, pp.751-772.

Remerciements

L'étude relatée ici a pour origine des travaux de l'un des auteurs, alors ingénieur à la division "Télécommunications" de Thomson-CSF, pour lesquels il a bénéficié en 1970 et 1971 de crédits internes d'étude. Elle s'est poursuivie au laboratoire "Théorie des communications" de l'ENST, où elle a bénéficié d'un contrat de recherche interne du CNET en 1974.

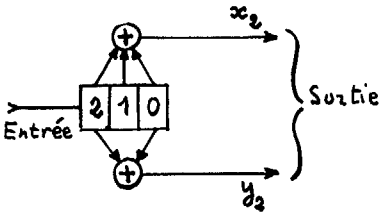


Fig. 1
Schéma de codeur

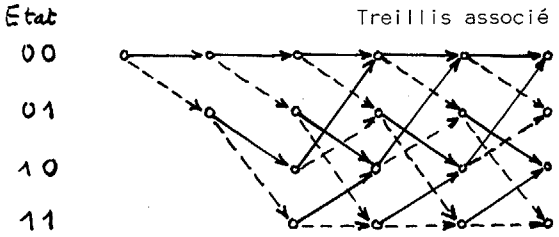
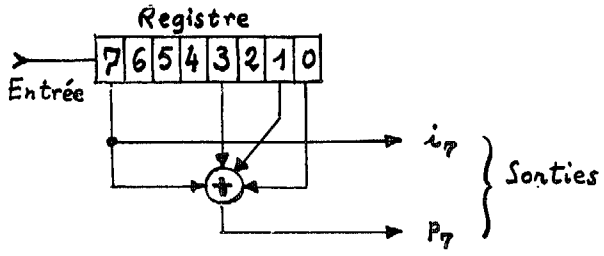


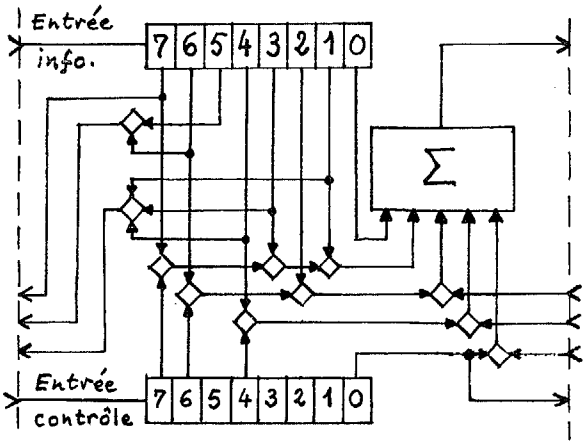
Fig. 2
Treillis associé

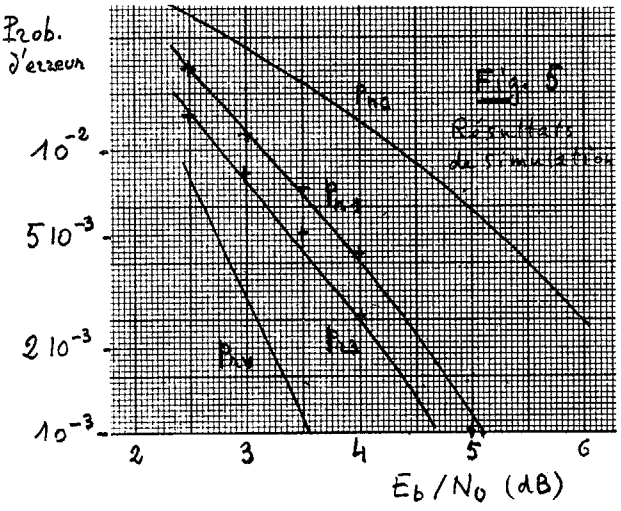
REINTERPRETATION ET GENERALISATIONS DU DECODAGE A SEUIL PONDERE
DES CODES RECURRENTS

Fig. 3 (ci-contre)
Schéma de codeur

Fig. 4 (ci-dessous)
Elément répétitif du décodeur itéré correspondant. Les registres contiennent les v.r., ◇ représente l'opération (8) du texte.







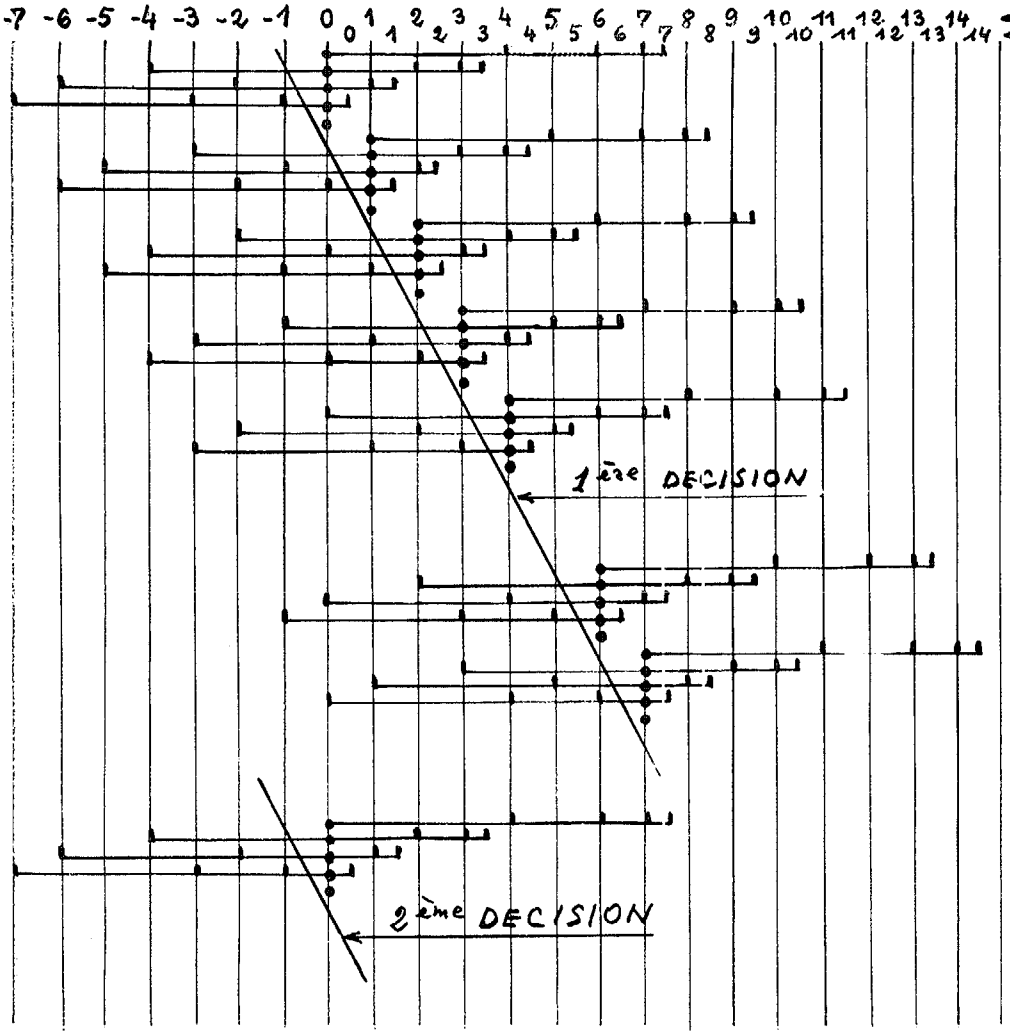


Fig. 6

Progression du décodage itéré (code (2)). Chaque trait horizontal représente une réplique composée ; les traits verticaux indiquent les symboles combinés, le point identifie le symbole dont elle est la réplique. Un point isolé représente une réplique simple. Le temps s'écoule de haut en bas. Les lignes obliques représentent les instants de décision en fonction du rang des symboles pour 2 décisions successives.