Etude de la dynamique pour le décodage itératif par Propagation de Croyances Généralisée

Jean-Christophe SIBEL, Sylvain REYNAL, David DECLERCQ

ETIS
ENSEA - Université de Cergy-Pontoise - CNRS, UMR8051
6, avenue du Ponceau 95000 Cergy-Pontoise, France
{jean-christophe.sibel,reynal,declercq}@ensea.fr

Résumé – Le codage correcteur d'erreurs assure une robustesse de l'information face au bruit du canal dans les communications numériques. L'algorithme de décodage de propagation de croyances est altéré par les cycles des codes, en termes de taux d'erreurs binaires et de dynamique. Une généralisation sophistiquée de cet algorithme permet d'obtenir des propriétés de décodages différentes. Nous proposons une première étude du comportement dynamique du décodage par l'algorithme de propagation de croyances généralisée afin de comparer les caractéristiques dynamiques avec la propagation simple afin de savoir quel algorithme est le plus stable.

Abstract – Channel coding is a particular processing which ensures a reliable transmission against the channel noise for digital communications. The Belief Propagation is a decoding algorithm whose binary error rate and dynamics are damaged by the loops of the codes. A sophisticated generalization of this algorithm can lead to different decoding properties. In this article, we propose a first comparative study of the dynamical behaviour of the Generalized Belief Propagation and the Belief Propagation to bring out their stability properties.

1 Introduction

L'utilisation des codes correcteurs d'erreurs est primordiale pour assurer une transmission fiable de l'information. Pour exploiter au mieux les capacités de correction de ces codes, on utilise des algorithmes de décodage itératifs de type *message-passing*, par exemple lla *Propagation de Croyances* (Belief Propagation-BP) [1]. Les codes Low Density Parity Check (LDPC), présentent certaines structures topologiques sur leur graphe associé, dit *graphe de Tanner*, qui dégradent les performances du BP. Etablie d'abord en physique statistique, l'approximation de Kikuchi détaillée dans [2, 3] est une approche qui permet d'absorber de telles structures au sein d'un graphe dit *graphe des régions*, et d'utiliser ce graphe comme support d'un autre algorithme, la *Propagation de Croyances Généralisée* (Generalized Belief Propagation-GBP) détaillée dans [2] que nous résumons dans la section 2. Sous-réserve d'une construction appropriée de ce graphe, le décodage devient plus performant que le décodage par BP et est optimal si ce graphe est acyclique, sachant que la construction optimale, si elle existe, n'est pas triviale. Nous présentons dans la section 3 la méthode de construction standard du graphe des régions, puis nous proposons une nouvelle méthode de construction sur un code particulier permettant d'établir des résultats. La section 4 comporte des résultats en termes de taux d'erreurs binaires (Bit Error Rate-BER)pour le BP et le GBP sur des codes simples. La section 5 est une étude comparative sur le comportement dynamique de ces deux algorithmes, en utilisant certains quantifieurs tels que l'exposant de Lyapunov et l'entropie.

2 Algorithmes

Sur le graphe de Tanner d'un code LDPC, N noeuds de variable $(x_1, ..., x_N)$ sont liés à M noeuds de parité $(c_1, ..., c_M)$ grâce aux connexions provenant des valeurs non-nulles de la matrice de parité associée au code. Le BP est un algorithme itératif basé sur la propagation de messages sur les branches du graphe, un message étant la probabilité a posteriori du noeud récepteur conditionnellement au noeud émetteur. On initialise chaque message $m_{i\to j}$ d'un noeud de variable x_i vers un noeud de parité c_j par la distribution de probabilité de l'observation en sortie du canal conditionnellement à la probabilité du bit x_i en entrée, appelée *vraisemblance* et notées v_i. L'algorithme est dit *itératif* puisque les opérations de mise à jour des messages [1] se répètent. On dit que le *graphe a convergé* lorsque les messages sont identiques d'une itération sur l'autre. On évalue à chaque itération pour chaque noeud x_i la distribution de probabilité a posteriori b_i, appelée *croyance*. L'argument du maximum de cette distribution est la valeur estimée du bit x_i en sortie. Les cycles du graphe peuvent cependant empêcher la convergence ou la diriger vers de mauvaises valeurs ne correspondant pas au mot en entrée du canal.

D'après l'approximation de Kikuchi, en absorbant certains de ces cycles dans de nouveaux types de noeuds, appelés *régions*, on peut parvenir à éviter cette altération. Le principe de construction consiste à réorganiser le graphe de Tanner tel que chaque noeud de variable et de parité soient inclus au minimum dans un nouveau type de noeud appelé *région*. On recherche ensuite les intersections entre ces régions pour for-

mer de nouvelles régions. On réitère ce procédé jusqu'à ce qu'il n'y ait plus d'intersections à trouver. Le graphe résultant de cette méthode est le graphe des régions.

Une méthode de construction automatique expliquée dans [2] consiste à considérer comme premières régions chaque noeud de parité accompagné de son voisinage direct, i.e. l'ensemble des noeuds de variable qui lui sont connexes dans le graphe de Tanner. Cette méthode est simple à implémenter mais risque de générer des cycles dans le graphe. Une construction plus particulière est détaillée dans la section 3.

L'algorithme utilisé avec ce nouveau graphe est le GBP, dans lequel des messages sont échangés entre les régions. Une fois que le graphe a convergé, on extrait les probabilités a posteriori sur les régions, appelées *croyances*. La règle de mise à jour $\mathcal{F}(A \rightarrow B)$ pour chaque message $m_{A\rightarrow B}$ est pondérée pour que l'algorithme puisse converger. Le message d'une région A vers une région fille B est

$$m_{A\rightarrow B}^{(k)} = w\mathcal{F}(A \rightarrow B) + (1-w)m_{A\rightarrow B}^{(k-1)} \qquad (1)$$

avec $w \in [0,1]$ le poids de la mise à jour. Dans le cas où le graphe des régions est acyclique, on fixe le poids à 1, pour ne considérer que la règle de mise à jour. Dans le cas contraire, on donne à w une valeur initiale suffisamment grande inférieure à 1, puis on la fait décroître pour forcer la convergence. Les observations indiquent de meilleures résultats avec cette pondération (voir section 5).

3 Nouvelle méthode de construction

On considère ici le code de Tanner ($N = 155$ bits) qui a de bonnes performances de décodage. Pour ce code il est possible de construire un graphe des régions plus sophistiqué qu'avec la méthode expliquée précédemment. Le graphe de Tanner se décompose en 155 structures particulières, correspondant à des *Trapping-Sets* (TS), qui sont responsables en partie des échecs de décodage. Introduits dans [4], ces TS sont composés de 5 bits (◯) et 9 équations de parité (□) dont 3 non vérifiées (■) comme le montre la figure 1.a). Ces structures, notées TS$(5,3)$, sont considérées isolés du reste du graphe de Tanner afin d'établir une construction du graphe des régions. En réalité, chaque noeud de parité c_j est connecté à cinq noeuds de variable qui ne sont pas tous inclus dans les TS correspondants, ce qui revient à ne considérer qu'une partie du voisinage directe de c_j.

Pour chaque TS on encapsule dans une région chaque noeud de parité non vérifié, les deux noeuds de parité connexes et les voisinages directs de chacun de ces noeuds (cf. figure 1.b)). On construit ainsi 465 régions, mais certaines sont identiques, étant donné que les noeuds de parité appartiennent à plusieurs TS. Après avoir supprimé les régions redondantes, on comptabilise au total 155 régions différentes. La construction se poursuit comme expliquée précédemment par recherche d'intersections. La motivation d'une telle construction est de réduire l'impact néfaste des TS sur le décodage en tentant d'en absorber une partie. L'inconvénient majeur d'une telle construction est

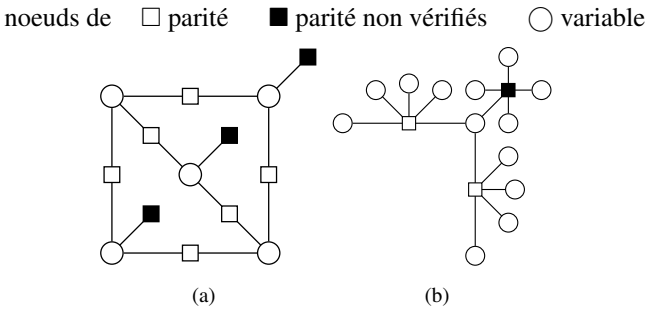

FIGURE 1 – Trapping-set (5,3) - Région

l'augmentation considérable de la complexité. On comptabilise par itération environ 10^6 calculs pour la mise à jour des messages du graphe des régions par le GBP, alors que le BP n'en comptabilise que 10^3 environs pour le graphe de Tanner. La complexité est une fonction croissante du nombre de noeuds de variable par régions, on ne peut donc pas choisir comme premières régions du graphe les TS eux-mêmes qui comportent bien plus de noeuds variables que les régions définies précédemment. La conséquence pratique n'est pas négligeable, et c'est pourquoi nous n'étudierons ce code qu'en termes de dynamique et non en terme de BER, cela n'empêchant pas d'observer les performances de décodage.

4 Performances de décodage

Nous présentons sur la figure 2 les BER du BP et du GBP pour deux codes de petites tailles sur un canal gaussien. Le premier code à 32 bits permet une construction optimale du graphe des régions, en utilisant un critère de réduction du graphe introduit dans [3], nous appelons ce code le *code de Pakzad*. On observe en accord avec la théorie que le GBP a de biens meilleures performances que le BP. Sur un tel code, la mise à 1.0 du poids w, après réduction du graphe par le critère de Pakzad, rend le décodage par GBP optimal, alors qu'une valeur plus faible donne un BER similaire à celui du BP. Le second code est le code de Hamming à 7 bits, qui présente des cycles qui ne sont pas absorbés dans la construction automatique du graphe des régions. On observe des performances quasi-équivalentes entre les deux algorithmes.

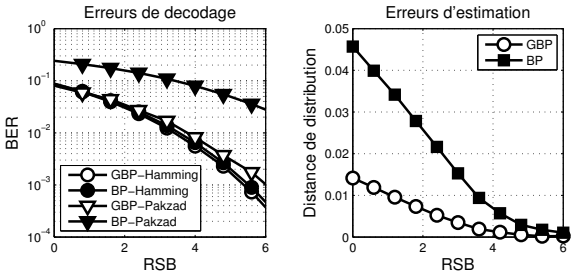

FIGURE 2 – BER et distance de probabilités pour le BP et le GBP

Notons cependant que pour des BER similaires, le GBP est souvent plus précis que le BP sur l'estimation des probabilités, ce qui en fait un algorithme d'inférence statistique plus per-

formant. Le deuxième graphe de la figure 2 montre la distance entre les distributions de probabilité exactes des bits et les distributions estimées par BP et GBP. L'estimation du GBP est en moyenne plus précise que celle du BP mais reste relativement voisine. On observe une différence très faible de l'ordre de 10^{-2}. Ainsi, lors du calcul du BER — revenant à seuiller les probabilités par rapport à 0.5 — les performances de correction d'erreur sont équivalentes.

5 Quantifieurs de la dynamique

On considère le mot de code identiquement nul à l'entrée d'un canal gaussien. Nous notons $b_i^{(k)} = b_i^{(k)}(x_i = 0)$ la croyance sur le noeud de variable i en la valeur 0 du bit i à l'itération k. b_i est alors l'estimation d'avoir une valeur correcte sur le noeud de variable i. La première grandeur de la dynamique utilisée est l'erreur quadratique moyenne des croyances, $E(k) = \frac{1}{N}\sum_{i=1}^{N}\left(b_i^{(k)}\right)^2$ introduit dans [5].

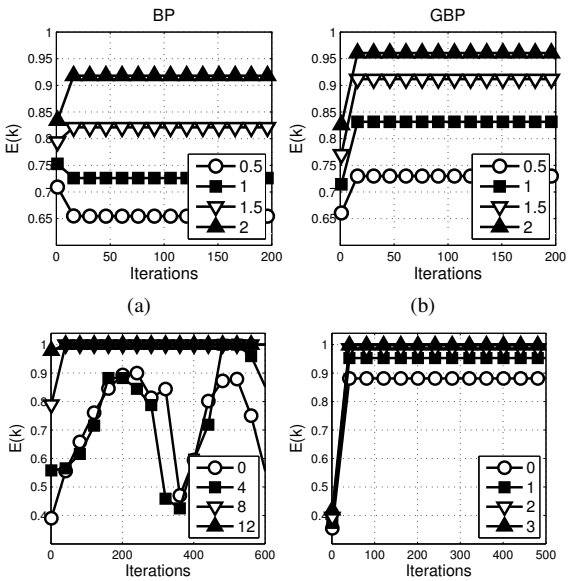

FIGURE 3 – $E(k)$ par BP et GBP pour le code de Hamming (a),(b) et le code de Pakzad (c),(d)

En étudiant l'évolution de $E(k)$ sur les codes de Hamming et de Pakzad, sur des tirages de canal particuliers, dans le cas du BP, on observe que l'erreur quadratique suit une évolution sur trois zones principales de RSB (voir figure 3, les légendes sont les valeurs de RSB en dB) :

- une zone d'indécision où $E(k)$ converge vers une valeur entre 0 et 1 exclus rendant le décodage incertain,
- une zone où $E(k)$ semble bifurquer ou entrer en régime non stationnaire,
- une zone où $E(k)$ converge vers la valeur 1 correspondant au mot de code nul indiquant un décodage réussi.

Ce comportement est en revanche moins marqué sur le GBP, la deuxième zone n'apparaissant que très peu.

Afin de cibler plus précisément les RSB correspondant à des comportements dynamiques particuliers, on calcul de l'exposant de Lyapunov λ. Nous sommes dans une situation non analytique, c'est pourquoi la méthode de calcul de λ est empirique. Pour deux décodages par le même algorithme, à conditions initiales (tirage de canal) très proches, on évalue à chaque itération k la distance d_k entre les croyances. Si l'algorithme est stable, alors $\lim_{k\to\infty} d_k = 0$. En revanche, s'il est instable, d_k diverge. Dans le pire cas, le chaos, l'amplification de la distance grandit exponentiellement, $\frac{d_{k+1}}{d_k} = e^{\lambda}$ pour tout k, avec $\lambda > 0$. En utilisant cette donnée, on obtient une valeur empirique de λ en calculant la moyenne sur les différentes itérations : $\lambda = \frac{1}{L-1}\sum_{k=1}^{L-1} \ln \frac{d_{k+1}}{d_k}$, où L est le nombre d'itérations de décodage.

Des simulations de λ ont confirmé l'importance du poids w dans la mise à jour (1). Dans le cas où $w = 1$, l'exposant de Lyapunov diverge indiquant que le système est dans un régime instable, voire chaotique. On choisira, par conséquent, pour tout code dont le graphe des régions n'est pas acyclique, $w < 1$.

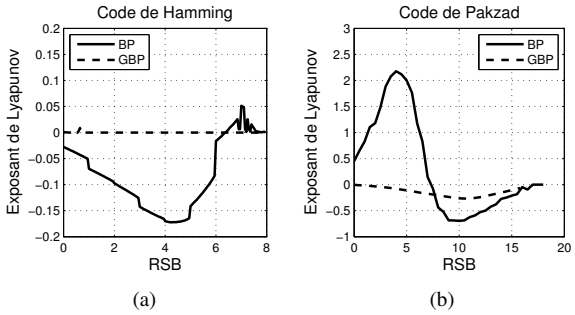

FIGURE 4 – Exposants de Lyapunov pour le BP et le GBP sur le code de Hamming et le code de Pakzad

On observe sur les figure 4.a) et b) que l'exposant de Lyapunov du GBP est très faible pour toutes les valeurs de RSB alors que celui du BP évolue différemment selon le RSB. Conformément à la définition de λ, on discerne trois zones particulières pour le BP. De 0dB à 5dB, λ est positif et croissant, indiquant une différence majeure de comportement entre deux erreurs quadratiques initialement proches. De 5dB à 10dB, λ diminue jusqu'à devenir négatif, indiquant que les erreurs quadratiques csont en train de se rejoindre. Au-delà de 10dB le BP a un comortement stable, proche de celui du GBP. On met en lien ces trois comportements avec les trois zones de RSB pour lesquelles l'erreur quadratique change d'évolution.

Dans le cas du code de Tanner pour lequel nous utilisons la construction détaillée à la section 3, les courbes de $E(k)$ sur la figure 5 révèlent une stabilité plus marquée du GBP, bien qu'il ne converge pas parfaitement. Le BP est en régime quasi-oscillant entre 2dB et 3dB tandis que le GBP est en régime permanent stationnaire. Les courbes du nombre d'erreurs par itération sur la figure 6 confirment cette observation.

L'oscillation de $E(k)$ correspond à un rapprochement quasi-périodique des croyances autour de la valeur d'indécision 0.5. $E(k)$ est une moyenne, il est donc très probable que certaines croyances passent périodiquement en-dessous de cette

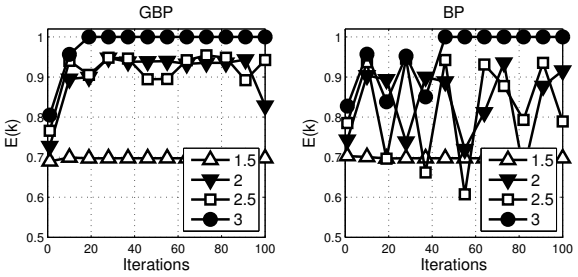

FIGURE 5 – Erreur quadratique sur le code de Tanner

valeur critique, ce qui représente une source de multiples erreurs de décodage comme l'indique la figure 6 aux alentours des itérations $20, 40, 60$ et 80.

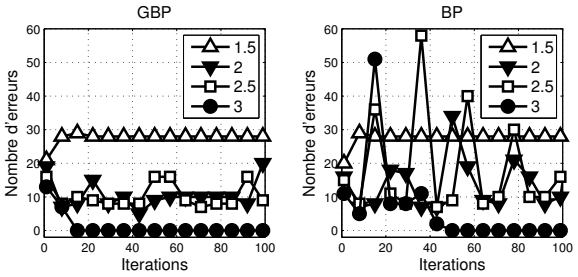

FIGURE 6 – Nombre d'erreurs sur le code de Tanner

L'exposant de Lyapunov du GBP sur la figure 7 progresse pour de faibles RSB comme celui du BP ce qui indique que les deux algorithmes ont la même dynamique. Autour de 2.25dB, l'exposant du GBP diminue beaucoup laissant le BP dans un régime moins stable, ce que nous avons vu avec les courbes de $E(k)$. Le dernier quantifieur utilisé ici pour traduire la dynamique de décodage est l'entropie. Cette grandeur traduit la quantité d'information connue sur un ensemble de variables aléatoires. Une entropie faible correspond à des probabilités très proches de 0 ou de 1, i.e. à un état des variables quasiment déterminé. Au contraire, une entropie importante correspond à des probabilités proches de la valeur d'indécision 0.5.

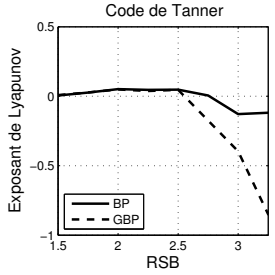

FIGURE 7 – Exposant de Lyapunov - code de Tanner

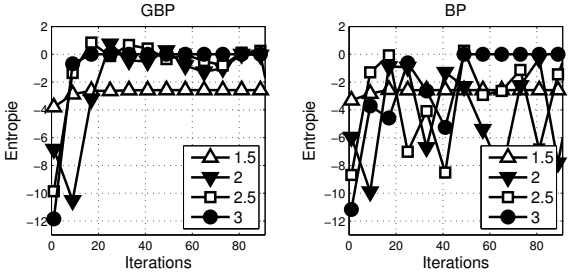

FIGURE 8 – Entropie sur le code de Tanner

Les approximations de l'entropie pour le BP et le GBP sont données dans [2]. On observe sur la figure 8 des valeurs négatives, car au cours des itérations, les croyances sur des régions ayant des variables en commun n'indiquent pas toutes les mêmes distributions pour ces variables : les croyances ne sont pas *cohérentes* entre elles. Il est donc plus pertinent de s'intéresser à la variation de l'estimation de l'entropie et sa distance à 0 plutôt qu'à son signe. On observe pour le GBP de faibles valeurs autour de 0 avec une variance minime comparée à celle du BP. Pour cette réalisation de canal, le GBP est plus stable, il ne subit pas de comportement particulier. A partir de 20 itérations, l'entropie ne change pas quelque soit le RSB, tandis que l'entropie du BP évolue encore beaucoup.

D'autres expériences avec des réalisations de canal différentes ont montré que les comportements d'instabilité apparaissent et évoluent de différentes façons mais dans des zones de RSB similaires. La nature des bifurcations correspondantes n'est cependant pas simple à retrouver. Une classification des bifurcations par zone de RSB pour le BP est proposée dans [5], où l'auteur effectue ses calculs à partir des expressions analytiques des messages de l'algorithme de BP. Les équations du GBP étant plus complexes, il n'a pas été proposé jusqu'à présent de telle classification.

6 Conclusion

Les résultats obtenus sont favorables pour l'hypothèse consistant à préférer le décodeur GBP au décodeur BP par rapport à leur comportement dynamique. L'étude du GBP nécessite cependant une approche algorithmique importante en raison de sa complexité supérieure à celle du BP. Les travaux en cours consistent en la définition et la recherche des attracteurs accompagnés de leur bassin d'attraction, ceci dans le but de classer et prévoir des comportements connaissant l'initialisation des algorithmes, conjointement au calcul des bifurcations pour le GBP à partir des expressions analytiques des messages.

Références

[1] F. R. Kschischang, B. J. Frey et H. A. Loeliger. Factor Graphs and the Sum-Product Algorithm. *IEEE Trans. on Inf. Theory*, 2001.

[2] J. S. Yedida, W. T. Freeman et Y. Weiss. Constructing free energy approximations and Generalized Belief Propagation algorithms. *IEEE Trans. on Inf. Theory*, 2004.

[3] P. Pakzad et V. Anantharam. Estimation and marginalization using Kikuchi approximation methods. *Neural Computation*, 2003.

[4] T. Richardson. Error floors of LDPC codes. In *Proc. 41st Annual Allerton Conf. on Comm. Cont. and Comp.*, 2003.

[5] X. Zheng, F. C. M. Lau, C. K. Tse et S. C. Wong. Study of bifurcation behavior of LDPC decoders. *Int. Journal of Bifurcation and Chaos*, 2005.