

Codes LDPC non-binaires pour le codage de source

Anne SAVARD
Encadrant : Claudio WEIDMANN

*ETIS / ENSEA - Université de Cergy-Pontoise - CNRS UMR 8051
F-95000 Cergy-Pontoise Cedex, France*





Plan

- 1 Contexte : Information adjacente compressée
- 2 Définitions
- 3 Algorithme RBP
- 4 Algorithme ABP
- 5 Scheduling
- 6 Conclusion et perspectives



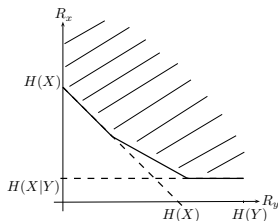
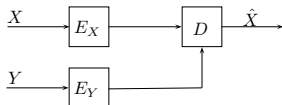
Plan

- 1 Contexte : Information adjacente compressée
- 2 Définitions
- 3 Algorithme RBP
- 4 Algorithme ABP
- 5 Scheduling
- 6 Conclusion et perspectives

Information adjacente compressée : région de rendements atteignables

Ahlswede et Körner : $\begin{cases} R_X \geq H(X|U) \\ R_Y \geq I(Y; U) \end{cases}$ avec $X - Y - U$ une chaîne de Markov ¹

1. R. Ahlswede and J. Körner, "Source coding with side information and a converse for degraded broadcast channels", *IEEE Transactions on Information Theory*, vol. : 21, pp. 629–637, 1975



Compression de sources binaires

- Critère : distance de Hamming ¹
- Pour les codes linéaires : problème NP-complet
- Difficulté des méthodes itératives : converger pour toutes les séquences source
- Compression avec des codes NB-LDPC ²

1. G. WeiHsin, R. Koetter, M. Effros and T. Ho, "On source coding with coded side information for a binary source with binary side informations", *ISIT 2007, Nice, France, June 24 - June 29*, pp. 1456-1460, 2007

2. A. Braunstein, F. Kayhan and R. Zecchina, "Efficient LDPC codes over $GF(q)$ for lossy data compression", *ISIT 2009, Seoul, Korea, June 28 - July 3*, pp. 1978-1982, 2009



Objectifs

- Étude de différentes perturbations de l'algorithme de propagation de croyance (BP)
- Amélioration de ces algorithmes



Plan

- 1 Contexte : Information adjacente compressée
- 2 **Définitions**
 - ▷ Codes LDPC
 - ▷ US-LDPC codes and b -reduction
 - ▷ Belief propagation decoder
 - ▷ Compression
- 3 Algorithme RBP
- 4 Algorithme ABP
- 5 Scheduling
- 6 Conclusion et perspectives

Encodeur

$C : u \in \mathbb{F}_q^k \mapsto x \in \mathbb{F}_q^n$ avec $\begin{cases} u \text{ vecteur de symbole d'information de taille } k \\ x \text{ mot de code de taille } n \end{cases}$

Mot de code

c mot de code ssi $H \cdot c^t = \underline{0}$

Codes binaires et non-binaires

- Codes binaires : composantes de H dans \mathbb{F}_2
- Codes non-binaires : composantes de H dans $\mathbb{F}_q, q = 2^r$

Graphe de Tanner

- Noeuds de variable : symboles d'un mot de code
- Noeuds de parité : équations de parité du code
- Connexions : symboles intervenant dans les équations de parité

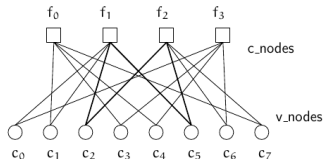
US-LDPC codes

Degré des noeuds de variable : 2

b -réduction

Suppression aléatoire de b noeuds de parité

$$H = \begin{bmatrix} 0 & 1 & 0 & 1 & 1 & 0 & 0 & 1 \\ 1 & 1 & 1 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 1 & 1 & 1 \\ 1 & 0 & 0 & 1 & 1 & 0 & 1 & 0 \end{bmatrix}$$



Girth

Longueur du plus petit cycle



Décodeur Belief Propagation (BP)

- Passage de messages le long du graphe de Tanner
- Messages échangés : probabilités
- Ordre de la mise à jour des messages : ordonnancement (scheduling)
- Calcul des *a posteriori* probabilities (APP) pour la prise de décision



Principe

- Pour une séquence d'entrée, converger vers un mot de code
- Compression : transmettre uniquement les k symboles d'information
- Pouvoir reconstruire le mot de code

Identification des k symboles d'information

- Avec b -réduction : Leaf-removal algorithm
- Sans b -réduction : encodage classique



Plan

- 1 Contexte : Information adjacente compressée
- 2 Définitions
- 3 **Algorithme RBP**
 - ▷ Principe
 - ▷ Mise à jour des messages
 - ▷ Simulations
 - ▷ Résultats
 - ▷ Conclusion
- 4 Algorithme ABP
- 5 Scheduling
- 6 Conclusion et perspectives

Reinforced Belief Propagation, proposé par Braunstein et al. ¹

1. A. Braunstein, F. Kayhan and R. Zecchina, "Efficient LDPC codes over $GF(q)$ for lossy data compression", *ISIT 2009, Seoul, Korea, June 28 - July 3, pp. 1978-1982, 2009*

Notations :

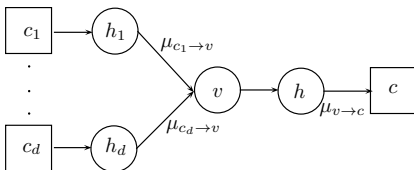
- y mot source à compresser
- Initialisation : $\forall a \in [0, q - 1]$, $\mu_{V_j}^{(1)}(a) \propto \exp(-Ld_H(y_j, a))$ avec
 - { L paramètre à fixer
 - { d_H distance de Hamming

Reinforcement

Modifie l'information reçue pour converger vers des séquence plus probables *a posteriori*

Messages

$$\mu_{v \rightarrow c} = \begin{bmatrix} P(v = 0 | \{\mu_{c_i \rightarrow v}\}_{i=1, \dots, d}) \\ P(v = 1 | \{\mu_{c_i \rightarrow v}\}_{i=1, \dots, d}) \\ \vdots \\ P(v = q - 1 | \{\mu_{c_i \rightarrow v}\}_{i=1, \dots, d}) \end{bmatrix}$$



Équations

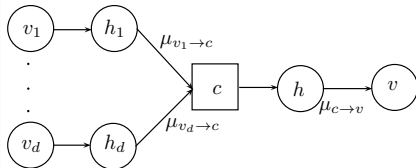
- $$\mu_{v \rightarrow c}^{(l+1)}(a) = (g_v^{(l)}(a))^{1-\gamma_0 \gamma_1^l} \mu_v^1(a) \prod_{c_i \neq c}^d \mu_{c_i \rightarrow v}^{(l)}(a)$$

- $$\text{APP} : g_v^{(l)}(a) = \mu_v^{(1)}(a) \prod_{i=1}^{d+1} \mu_{c_i \rightarrow v}^{(l-1)}(a)$$

- $$\gamma_0, \gamma_1 \in [0, 1]$$

Messages

$$\mu_{c \rightarrow v} = \begin{bmatrix} P(c = 0 \text{ if } v = 0 | \{\mu_{v_i \rightarrow c}\}_{i=1, \dots, d}) \\ P(c = 0 \text{ if } v = 1 | \{\mu_{v_i \rightarrow c}\}_{i=1, \dots, d}) \\ \vdots \\ P(c = 0 \text{ if } v = q - 1 | \{\mu_{v_i \rightarrow c}\}_{i=1, \dots, d}) \end{bmatrix}$$



Équations

- $\mu_{c \rightarrow v} = \mathcal{F}^{-1} \left(\prod_{v_i \neq v}^d \mathcal{F}(\mu_{v_i \rightarrow c}) \right)$
- $\mathcal{F}, \mathcal{F}^{-1}$: transformée de Fourier et inverse sur \mathbb{F}_q

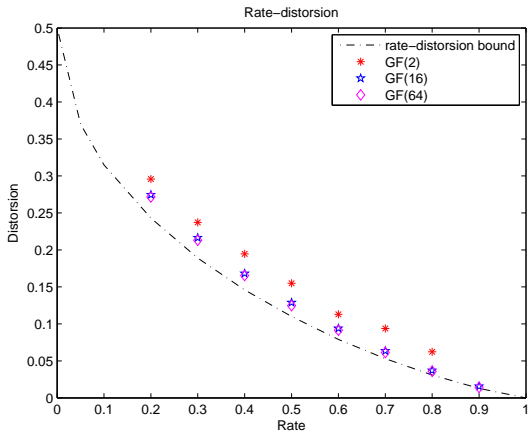


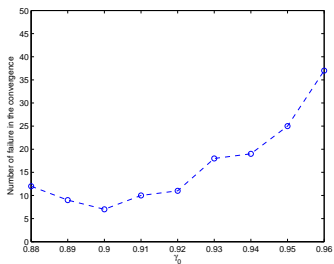
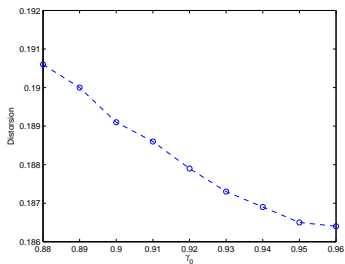
FIGURE: $n=1000$, 50 samples, $L=2$, $\gamma_0 = 0.92$, $\gamma_1 = 1$, $b=2$

Cadre choisi

- Code linéaire : mot de code nul transmis
- Augmenter la probabilité d'erreur
- "Bonne" réponse du décodeur : mot de code nul

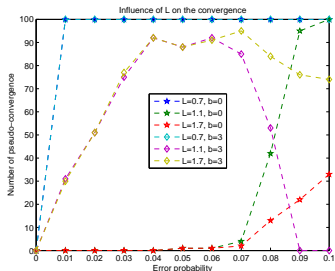
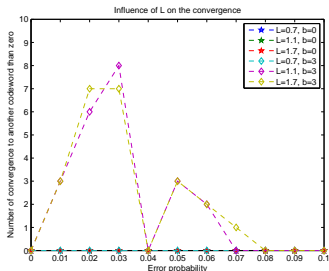
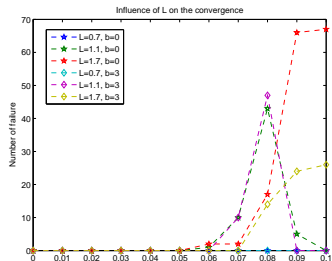
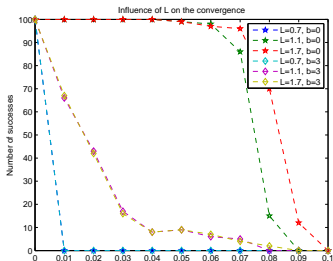
Convergence de l'algorithme RBP

- Vers un mot de code :
 - Vers le mot de code nul ou vers un mot de code plus proche de y que le mot de code nul : Succès
 - Vers un autre mot : Autre
- Stabilisation des messages $\mu_{c \rightarrow v}^{(t+1)} = \mu_{c \rightarrow v}^{(t)}$ sur une séquence qui n'est pas un mot de code : Pseudo-convergence

Influence de γ_0 FIGURE: $n=1600$, $\text{rate}=0.33$, $b=5$, $q=16$, 50 samples

$$\gamma_0^{opt} = 0.9$$

Influence de L et b



Conclusions

- Rôle des différents paramètres de l'algorithme RBP
 - Influence de L :
 - $L < 1$: Pseudo-convergence
 - $L > 1$: Convergence possible vers des mots de code
 - Influence de b :
 - $b > 0$: Convergence possible vers des autres mots de code (augmenter b revient à augmenter le nombre de mots de code)
 - Influence de γ_0 :
 - Existence d'un γ_0 optimal (minimum d'échec dans la convergence)
- Nécessité du scheduling



Plan

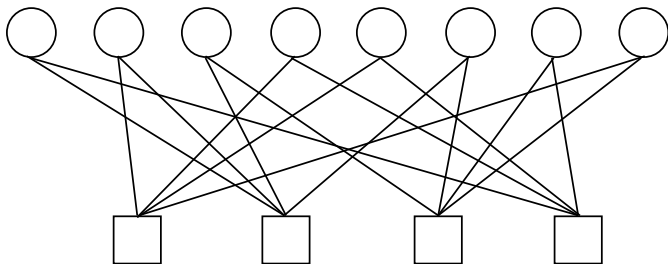
- 1 Contexte : Information adjacente compressée
- 2 Définitions
- 3 Algorithme RBP
- 4 **Algorithme ABP**
 - ▷ Algorithme ABP binaire
 - ▷ Algorithme ABP non-binaire
 - ▷ Résultats
 - ▷ Conclusion
- 5 Scheduling
- 6 Conclusion et perspectives

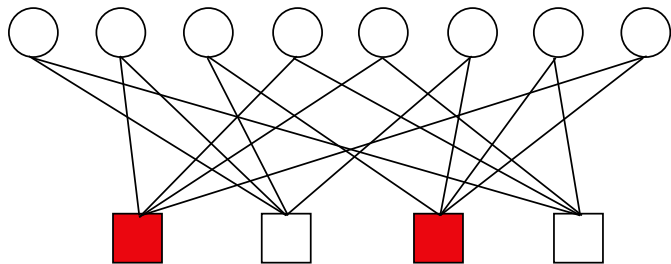


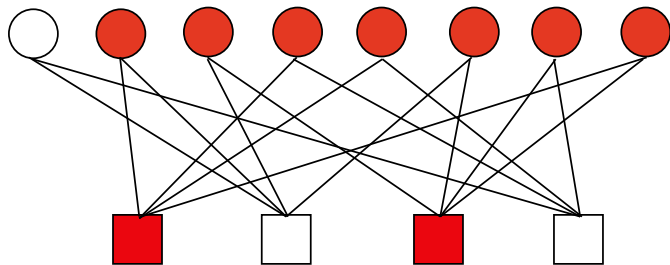
ABP

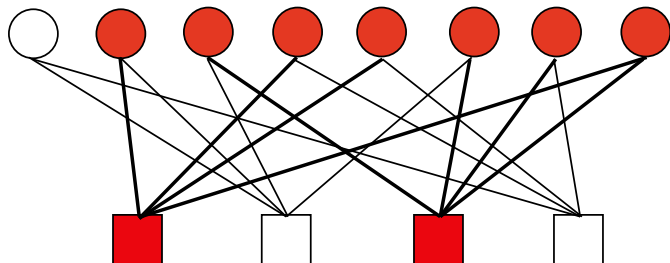
- Augmented Belief Propagation, proposé par Varnica et al. ¹
- Proposé dans le cas binaire
- Étendu au cas non binaire

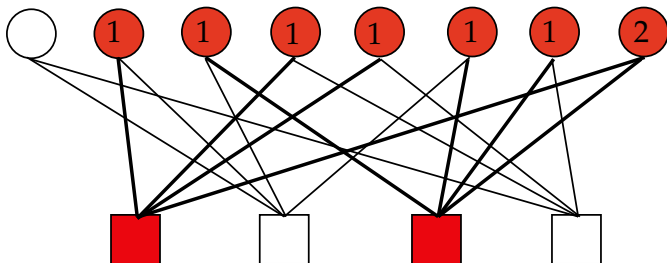
1. N. Varnica, M. P.C. Fossorier and A. Kavčić, "Augmented belief propagation decoding of low-density parity check codes", *IEEE Transactions on Communications*, vol. 55, pp. 1308-1317, 2007

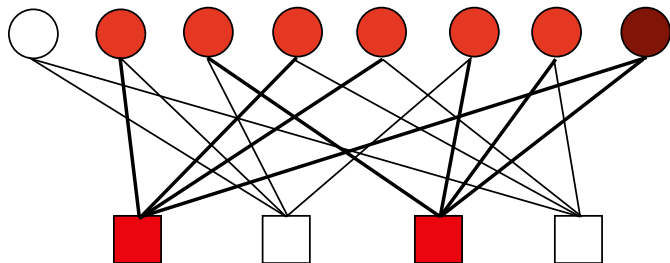


 $C_S^{(L)}$

 $V_S^{(L)}$

 $G_S^{(L)}$

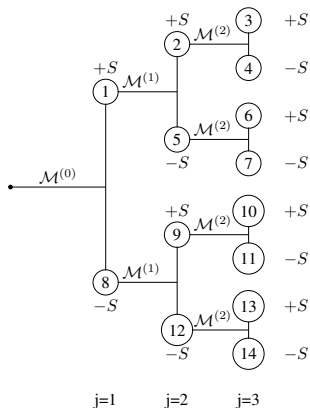


 S_v^{\max}

Principe général

- Perturbation de l'algorithme BP en cas d'échec
- Sélection d'un noeud
- Saturation de son LLR :

$$LLR(x) = \log \left(\frac{p(x_j=1|y_j)}{p(x_j=0|y_j)} \right) \text{ à } \pm\infty$$
- Nouvelle tentative de l'algorithme BP



Sélection d'un noeud

- Observations :
 - $d_{G_S}(v)$ élevé : forte probabilité que v soit en erreur
 - $|LLR(v)|$ faible : forte probabilité que v soit en erreur
- Sélectionner v_p tel que $v_p = \arg \min_{v \in S_v^{max}} |LLR(v)|$

Interprétation

- $v_p = \arg \min_{v \in S_v^{max}} |LLR(v)|$: revient à sélectionner le noeud de variable tel que $P(v_p = 0) \simeq P(v_p = 1)$
- Saturation du LLR du noeud : forcer ce noeud à être 0 ou 1

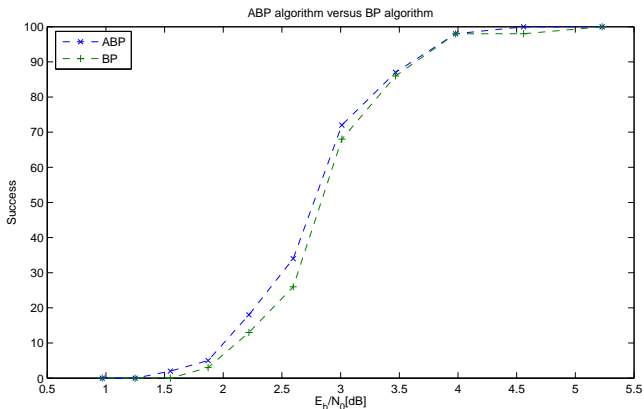


FIGURE: Nombre de succès, 1024×2048 (3,6)-regular LDPC code, 100 samples



Extension du cas binaire au cas non-binaire

- Mêmes définitions pour $C_S^{(L)}$, $V_S^{(L)}$, $G_S^{(L)}$, $d_{G_S}(v)$, S_v^{max}
- Sélectionner le noeud de variable dans $V_S^{(L)}$ ayant ses probabilités symbole les plus uniformes
- Saturer le message initial arrivant à ce noeud

Problèmes

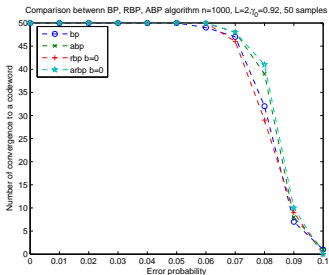
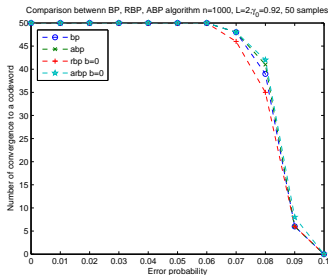
- Domaine des probabilités vs domaine des LLR ?
- Saturation d'un bit, de plusieurs bits ?

Saturation d'un seul bit

- Calculer les probabilités marginales bits à partir des APP
- En chercher le maximum
- Saturer l'une des probabilités bit marginales

Saturation de plusieurs bits

- Calculer les probabilités marginales bits à partir des APP
- Identifier les bits tels que $|P_{bin} - 0.5| < seuil$
- Saturer les probabilités bit marginales correspondantes





Conclusions

- Méthode efficace
- Peut s'utiliser sur le RBP et améliore les performances
- Inconvénient : Ralentissement du décodeur à cause de la procédure de sélection et de saturation

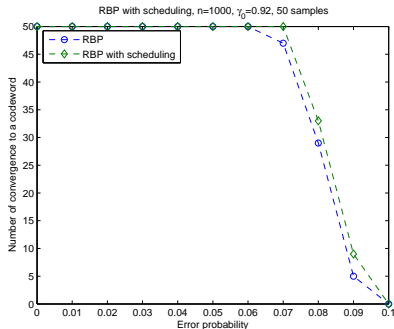


Plan

- 1 Contexte : Information adjacente compressée
- 2 Définitions
- 3 Algorithme RBP
- 4 Algorithme ABP
- 5 **Scheduling**
 - ▷ Scheduling proposé
 - ▷ Scheduling proposé par Beermann et al.
 - ▷ Conclusion
- 6 Conclusion et perspectives

Scheduling

- Tirage aléatoire d'un noeud de parité
- Calcul de $V = v \in \mathcal{N}(c)$
- Mise à jour des messages
 $\mu_{c \rightarrow v}, \forall v \in V$
- Mise à jour des messages
 $\mu_{v \rightarrow c}, \forall v \in V$
- Calcul de $C = c \in \mathcal{N}(v), \forall v \in V$
- Tirage aléatoire d'un noeud de parité dans C



Scheduling

Proposé par Beermann et al. (basé sur les travaux de Mao, Zhang et Sharon)¹

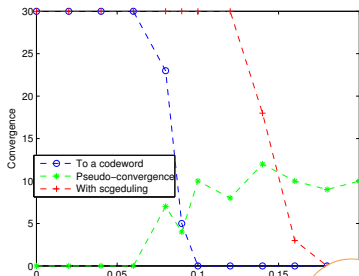
- Calcul du girth local g_n et du plus long cycle dans le graphe g_{max}
- Calcul de $p_n^{[g]} = \frac{g_n}{g_{max}}$

1. M. Beermann, L. Schmalen and P. Vary, "Improved decoding of binary and non-binary LDPC codes by probabilistic shuffled belief propagation", *Communications ICC, IEEE International Conference on*, pp. 1-5, 2011

Algorithm 1 Scheduling

```

1: Compute  $p_n^{[g]}$ 
2: for  $i \rightarrow \max_{iter}$  do
3:   for  $n \rightarrow N$  do
4:      $\forall c \in \mathcal{N}(v_n), \mu_{v_n \rightarrow c}^{(i)} = \mu_{v_n \rightarrow c}^{(i-1)}$  with probability  $1 - p_n^{[g]}$ 
5:      $\forall c \in \mathcal{N}(v_n), \mu_{c \rightarrow v_n}^{(i)} = \mu_{c \rightarrow v_n}^{(i-1)}$  with probability  $1 - p_n^{[g]}$ 
6:   end for
7:   for  $n \rightarrow N$  do
8:     Compute all missing incoming messages to  $v_n$ 
9:     Compute all missing outgoing messages from  $v_n$ 
10:  end for
11: end for
    
```





Conclusions

- Méthode performante
- Méthode pouvant être utilisée pour tout décodeur itératif
- Deuxième scheduling très efficace mais calcul des girths et du cycle max très lent



Plan

- 1 Contexte : Information adjacente compressée
- 2 Définitions
- 3 Algorithme RBP
- 4 Algorithme ABP
- 5 Scheduling
- 6 **Conclusion et perspectives**
 - ▷ Comparaison des méthodes proposées
 - ▷ Conclusion et perspectives

Paramètres de simulation

$N = 1000$, $rate = 1/2$, $q = 16$

RBP : $iter_{max} = 300$, $\gamma_0 = 0.92$, $\gamma_1 = 1$, $b = 0$, $nb_{trials} = 5$

ABP : $nb_{iter}/step = 30$

Error probability	0.08	0.09	0.1
Nb success BP	33	8	2
Nb success ABP	42	13	0
Nb iter ABP	73.88	72.2	max
Nb success RBP	31	7	1
Nb success ARBP	43	12	0
Nb iter ARBP	76.81	80.2	max

TABLE: Comparaison BP/ABP, RBP/ARBP

Paramètres de simulation

$N = 1000$, $rate = 1/2$, $q = 16$

RBP : $iter_{max} = 300$, $\gamma_0 = 0.92$, $\gamma_1 = 1$, $b = 0$, $nb_{trials} = 5$

ABP : $nb_{iter}/step = 30$

Error probability	0.08	0.09	0.1	0.12	0.14	0.16	0.18
Nb of success	23	5	0	0	0	0	0
Nb iter success (RBP)	24.52	33.4	max	max	max	max	max
Nb of pseudo-convergence	4	4	10	8	12	10	9
Nb iter pseudo (RBP)	162	178.5	151.6	143.25	172.5	167	164.88
Nb of success with scheduling1	26	7	0	0	0	0	0
Nb iter scheduling1 (RBP)	149.66	47	max	max	max	max	max
Nb trials	3.33	2.5	max	max	max	max	max
Nb of success with scheduling2	30	30	30	30	18	3	0
Nb iter scheduling2 (RBP)	1222	1286.4	1527.6	2029.9	3210.8	5061	max
Nb trials	2	2	2	2	2.11	2.33	max

TABLE: Comparaison entre les deux scheduling

Paramètres de simulation

$N = 1000$, $rate = 1/2$, $q = 16$

RBP : $iter_{max} = 300$, $\gamma_0 = 0.92$, $\gamma_1 = 1$, $b = 0$, $nb_{trials} = 5$

ABP : $nb_{iter}/step = 30$

Error probability	0.08	0.09	0.1	0.12	0.14	0.16	0.18
Nb of success	23	5	0	0	0	0	0
Nb iter success (RBP)	24.52	33.4	max	max	max	max	max
Nb of pseudo-convergence	4	4	10	8	12	10	9
Nb iter pseudo (RBP)	162	178.5	151.6	143.25	172.5	167	164.88
Nb of success with scheduling1	26	7	0	0	0	0	0
Nb iter scheduling1 (RBP)	149.66	47	max	max	max	max	max
Nb trials	3.33	2.5	max	max	max	max	max
Nb of success with scheduling2	30	30	30	30	18	3	0
Nb iter scheduling2 (RBP)	1222	1286.4	1527.6	2029.9	3210.8	5061	max
Nb trials	2	2	2	2	2.11	2.33	max

TABLE: Comparaison entre les deux scheduling

Paramètres de simulation

$N = 1000$, $rate = 1/2$, $q = 16$

RBP : $iter_{max} = 300$, $\gamma_0 = 0.92$, $\gamma_1 = 1$, $b = 0$, $nb_{trials} = 5$

ABP : $nb_{iter}/step = 30$

Error probability	0.08	0.09	0.1	0.12	0.14	0.16	0.18
Nb of success	23	5	0	0	0	0	0
Nb iter success (RBP)	24.52	33.4	max	max	max	max	max
Nb of pseudo-convergence	4	4	10	8	12	10	9
Nb iter pseudo (RBP)	162	178.5	151.6	143.25	172.5	167	164.88
Nb of success with scheduling1	26	7	0	0	0	0	0
Nb iter scheduling1 (RBP)	149.66	47	max	max	max	max	max
Nb trials	3.33	2.5	max	max	max	max	max
Nb of success with scheduling2	30	30	30	30	18	3	0
Nb iter scheduling2 (RBP)	1222	1286.4	1527.6	2029.9	3210.8	5061	max
Nb trials	2	2	2	2	2.11	2.33	max

TABLE: Comparaison entre les deux scheduling

Paramètres de simulation

$N = 1000$, $rate = 1/2$, $q = 16$

RBP : $iter_{max} = 300$, $\gamma_0 = 0.92$, $\gamma_1 = 1$, $b = 0$, $nb_{trials} = 5$

ABP : $nb_{iter}/step = 30$

Error probability	0.08	0.09	0.1	0.12	0.14	0.16	0.18
Nb of success	23	5	0	0	0	0	0
Nb iter success (RBP)	24.52	33.4	max	max	max	max	max
Nb of pseudo-convergence	4	4	10	8	12	10	9
Nb iter pseudo (RBP)	162	178.5	151.6	143.25	172.5	167	164.88
Nb of success with scheduling1	26	7	0	0	0	0	0
Nb iter scheduling1 (RBP)	149.66	47	max	max	max	max	max
Nb trials	3.33	2.5	max	max	max	max	max
Nb of success with scheduling2	30	30	30	30	18	3	0
Nb iter scheduling2 (RBP)	1222	1286.4	1527.6	2029.9	3210.8	5061	max
Nb trials	2	2	2	2	2.11	2.33	max

TABLE: Comparaison entre les deux scheduling



Conclusion

- Compression possible avec des codes LDPC et décodage itératif
- Importance de rajouter de l'aléatoire dans les processus itératif

Contributions

- Étude de l'influence des paramètres du RBP sur la convergence
- Adaptation du ABP au cas non binaire
- Efficacité du scheduling basé sur le graphe de Tanner

Perspectives

- Intégrer ces méthodes aux systèmes avec information adjacente compressée
- Identifier comment utiliser au mieux l'information adjacente



Merci de votre attention.
Questions ?