



Generalized Belief Propagation to break trapping sets in LDPC codes

Jean-Christophe Sibel, Sylvain Reynal, David Declercq

► To cite this version:

Jean-Christophe Sibel, Sylvain Reynal, David Declercq. Generalized Belief Propagation to break trapping sets in LDPC codes. 2014 Australian Communications Theory Workshop (AusCTW), Feb 2014, Sydney, Australia. pp.132, 2014. <hal-00968251>

HAL Id: hal-00968251

<https://hal.archives-ouvertes.fr/hal-00968251>

Submitted on 1 Apr 2014

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Generalized Belief Propagation to break trapping sets in LDPC codes

J.-C. Sibel
UEB, IETR
INSA de Rennes, CNRS UMR 6164
Rennes, France
jean-christophe.sibel@insa-rennes.fr

S. Reynal, D. Declercq
ETIS, ENSEA
Univ. of Cergy-Pontoise, CNRS UMR 8051
Cergy-Pontoise, France
{reynal,declercq}@ensea.fr

Abstract—In this paper, we focus on the Generalized Belief Propagation (GBP) algorithm to solve trapping sets in Low-Density Parity-Check (LDPC) codes. Trapping sets are topological structures in Tanner graphs of LDPC codes that are not correctly decoded by Belief Propagation (BP), leading to exhibit an error-floor in the Bit-Error Rate (BER). Stemming from statistical physics of spin glasses, GBP consists in passing messages between clusters of Tanner graph nodes in another graph called the region-graph. Here, we introduce a specific clustering of nodes, based on a novel local loopfree principle, that breaks trapping sets such that the resulting region-graph is locally loopfree. We then construct a hybrid decoder made of BP and GBP that proves to be a powerful decoder as it clearly improves the BER and defeats the error-floor.

Index Terms—LDPC codes, Generalized Belief Propagation, trapping sets, error-floor, local clustering

I. INTRODUCTION

Low-Density Parity-Check (LDPC) codes, discovered by Gallager [1] and rediscovered by MacKay and Neal [2], are known as powerful codes to make information robust against transmission channel noise. Great advantage is that they make decoders fast and accurate. Among them, the Belief Propagation (BP) algorithm, introduced by Pearl [3], still arouses much interest as it does not require for complex hardware implementation and it may help LDPC codes to approach Shannon's limit [4]. BP has been widely studied according to messages behavior [5], [6], convergence and stability [7], [8], [9] and dynamics [10], [11]. In [12] Richardson introduced the problem of error-floor: an abrupt degradation of the Bit-Error-Rate of BP when channel noise power is very low. In [13], [14] were connected failures responsible for this phenomenon to trapping sets. These topological structures of LDPC codes Tanner graph are combinations of loops that prevent BP from correctly decoding. In [15], [16] authors examined in detail Finite Alphabet Iterative Decoders (FAID) to solve trapping sets and decrease error-floor, overcoming BP performance. In spite of clear domination of FAID over BP in error-floor region, these decoders are not always defined for LDPC codes of arbitrary nodes degrees. In addition, they are aimed at treating error events from discrete channels, *e.g.* the Binary Symmetric Channel (BSC) [17], therefore error events induced by continuous channels, as the Gaussian channel, are not easily handled. In parallel work in [18], [19], Yedidia *et al.*

focused on another decoding approach based on inference in spin glasses [20], [21]: Region-Based Approximation (RBA). This method consists in clustering the Tanner graph to create a new graph, the region-graph, which nodes are called regions. Messages are iteratively exchanged between regions according to the Generalized Belief Propagation (GBP) algorithm. The way clusters are chosen totally determines decoding performance of GBP, given that many clusterings may emerged from a single Tanner graph, [22], [23]. In this paper we propose an application of RBA and GBP on LDPC codes full of small trapping sets to outperform BP especially in error-floor region. We experimentally show that splitting trapping sets clearly makes GBP decoding more powerful and accurate than BP.

In section II, we shortly describe LDPC codes and BP update rules. We present in section III statistical physics connected to BP, that we extend in section IV to present the GBP algorithm. Follows in section V an introduction of our novel principle of the region-graph construction that locally breaks trapping sets. We end with experimental results in VI that demonstrate the relevancy of our novel method on the Tanner code [24]. This is an LDPC code of column-weight three, length 155 and rate $2/5$, entirely covered with 155 small trapping sets, providing BP performance far enough from Maximum Likelihood Decoding (MLD) such that improvements on iterative decoding are visible [25].

II. PRELIMINARIES

A. LDPC codes

An LDPC code is defined as a set of N -codewords $\mathcal{C} \subset \{0, 1\}^N$ such that any codeword $\mathbf{x} = [x_1 \dots x_N]$ is in the kernel of M parity-check equations $\mathbf{c} = \{c_1, \dots, c_M\}$. Any bit x_i is represented by a random variable X_i valued in $\{0, 1\}$, any parity-check equation c_j is represented by a random variable f_j valued in $\{0 \text{ (unsatisfied)}, 1 \text{ (satisfied)}\}$. As an example, the Hamming code is defined by 16 codewords that satisfy altogether the following parity-check equations:

$$\begin{aligned} (f_a) \quad X_1 \oplus X_2 \oplus X_3 \oplus X_5 &= 0 \\ (f_b) \quad X_1 \oplus X_2 \oplus X_4 \oplus X_6 &= 0 \\ (f_c) \quad X_1 \oplus X_3 \oplus X_4 \oplus X_7 &= 0 \end{aligned}$$

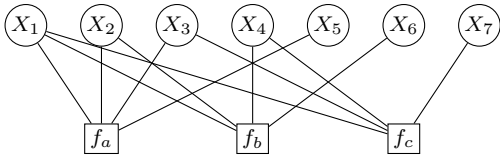


Fig. 1. Tanner graph of Hamming code

We usually represent an LDPC code by its graphical representation called the Tanner graph $\mathcal{G} = \mathbf{X} \cup \mathbf{F} \cup E$, in which $\mathbf{X} = [X_1 \dots X_N]$ and $\mathbf{F} = \{f_1, \dots, f_M\}$. We draw an edge $e_{ia} \in E$ between variable node X_i and check node f_a if and only if X_i is an argument of f_a , e.g. the Tanner graph of the Hamming code depicted in Fig.1.

Given N channel observations y_1, \dots, y_N , decoding an LDPC codes is searching for the most likely word $\hat{\mathbf{x}} \in \{0, 1\}^N$ of following distribution:

$$p(\mathbf{x}, \mathbf{y}) = p(\mathbf{y}|\mathbf{x})p(\mathbf{x}) \propto \prod_{i=1}^N p_i(y_i|x_i) \prod_{a=1}^M f_a(\mathbf{x}_a). \quad (1)$$

where \mathbf{x}_a is the state of all variable nodes connected by edges to the check node f_a , denoted by $\mathbf{X}_a = \{X_i \in \mathbf{X} | e_{ia} \in E\}$, and $p_i(y_i|x_i)$ is called the likelihood of X_i .

B. Belief Propagation

Computing $\hat{\mathbf{x}} = \arg \max_{\mathbf{x}} p(\mathbf{x}, \mathbf{y})$ is intractable as it requires to scan 2^N words. N usually reaches several hundreds in tests, as the Tanner code of length $N = 155$, and several thousands in practice, as DVB-S2 codes of length $N = 64800$ [26]. BP algorithm is a tractable and practical solution to (1) by approximating all marginal distributions, or marginals, on variable nodes with beliefs $\{b_i\}_{1 \leq i \leq N}$ s.t.:

$$\hat{\mathbf{x}} = \bigcup_{i=1}^N \arg \max_{x_i} b_i(x_i). \quad (2)$$

BP is an iterative decoder that passes messages along edges of \mathcal{G} , which equations, given in [4], are for any edge $e_{ia} \in E$, for any value $x_i \in \{0, 1\}$, at any iteration $k \geq 1$:

$$n_{ia}^{(k)}(x_i) \propto p_i(y_i|x_i) \prod_{\mathbf{X}_b \ni X_i, f_b \neq f_a} m_{bi}^{(k-1)}(x_i) \quad (3)$$

$$m_{ai}^{(k)}(x_i) \propto \sum_{\mathbf{x}_a \cup x_i} f_a(\mathbf{x}_a) \prod_{X_j \in \mathbf{X}_a \setminus X_i} n_{ja}^{(k)}(x_j). \quad (4)$$

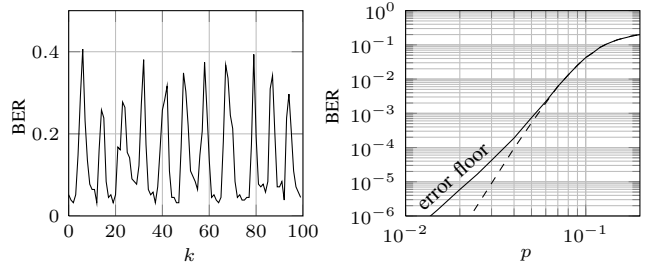
Quantity $n_{ia}^{(k)}$ (resp. $m_{ai}^{(k)}$) is the message from X_i to f_a (resp. from f_a to X_i). These messages are usually initialized with likelihoods of adjacent variable nodes. Belief on X_i is computed at any iteration $k \geq 1$ for any state $x_i \in \{0, 1\}$ as:

$$b_i^{(k)}(x_i) \propto p_i(y_i|x_i) \prod_{\mathbf{X}_a \ni X_i} m_{ai}^{(k)}(x_i). \quad (5)$$

BP runs while messages still vary from k to $k+1$, i.e. messages have not converged yet, or while output word $\hat{\mathbf{x}}^{(k)}$ does not satisfy all parity-check equations.

C. Failures

When Tanner graph is tree-like, BP is ensured to converge to optimal MLD. In case the graph presents loopy structures, BP may be trapped into infinite process as neither it converges nor it results in a codeword. Decoding performance are then hard to expect and might present non trivial behavior, see oscillations in Fig.2(a), studied in [11].



(a) Oscillations for $p = 10^{-5}$ (b) Error floor

Fig. 2. BP decoding of Tanner code on BSC(p)

Here we focus on specific low-weight error events connected to Tanner graph topology: trapping sets. They are particularly harmful for low channel noise power, reflected by the error-floor of the BER exhibited in Fig.2(b). A trapping set $TS(a, b)$ is a structure of a variable nodes such that induced subgraph has b odd-degree, or unsatisfied, check nodes, see $TS(5, 3)$ depicted on Fig.3.

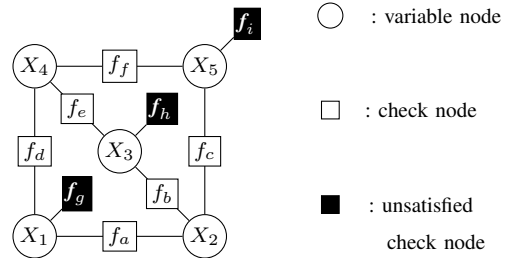


Fig. 3. Trapping-set of $a = 5$ variable nodes and $b = 3$ unsatisfied check nodes

Overcoming error-floor may be done according to two strategies: either we construct codes free of small trapping sets as proposed in [27], or we design dedicated decoders able to handle trapping sets as proposed in this paper.

III. STATISTICAL PHYSICS OF LDPC CODES

A. Spin glasses

A spin glass [20] is a vector of N spins $\mathbf{S} = [S_1 \dots S_N]$, each one randomly valued in $\{-1, +1\}$, that are correlated according to coupling constants $\{J_{ij}\}_{1 \leq i, j \leq N}$, as bits of an LDPC code are correlated by parity-check equations. Coupling constants are summarized by an energy function defined for any state $\mathbf{s} \in \{-1, +1\}^N$:

$$E_J(\mathbf{s}) = - \sum_{\langle i, j \rangle} J_{ij} s_i s_j. \quad (6)$$

where $\langle i, j \rangle$ stands for spins s.t. $J_{ij} \neq 0$. Noise on a spin glass is typically modeled by an external magnetic field $\mathbf{h} = [h_1 \dots h_N]$ that independently influences all spins such that associated energy function is the scalar product between \mathbf{h} and \mathbf{s} :

$$E_H(\mathbf{s}, \mathbf{h}) = - \sum_{i=1}^N h_i s_i \quad (7)$$

According to Boltzmann's law, a noisy spin glass is described by the following distribution for any state $\mathbf{s} \in \{-1, +1\}^N$:

$$p(\mathbf{s}, \mathbf{h}) \propto e^{-E_H(\mathbf{s}, \mathbf{h}) - E_J(\mathbf{s})} \quad (8)$$

which is equivalent to:

$$p(\mathbf{s}, \mathbf{h}) \propto \prod_{i=1}^N e^{h_i s_i} \prod_{\langle i, j \rangle} e^{J_{ij} s_i s_j} \quad (9)$$

This distribution equals distribution Tanner graphs distribution (1) when check nodes degree is two, then solving a spin glass is decoding the equivalent LDPC code which the parity-check equation related to J_{ij} is:

$$J_{ij} s_i s_j \begin{cases} > 0 & \iff x_i \oplus x_j = 0 \quad (\text{parity satisfied}), \\ < 0 & \iff x_i \oplus x_j = 1 \quad (\text{parity unsatisfied}), \\ = 0 & \iff \text{no parity-check equation.} \end{cases}$$

B. Variational approach

Again, computing (9) for all spin states \mathbf{s} is not tractable. According to [18], we may use a variational approach by means of a tractable distribution b , called belief, that we make vary to approximate p . The usual fair indicator to estimate relevancy of b is the Kullback-Liebler divergence [28] defined as:

$$\text{KL}(b, p) = \sum_{\mathbf{s}} b(\mathbf{s}, \mathbf{h}) \log \frac{b(\mathbf{s}, \mathbf{h})}{p(\mathbf{s}, \mathbf{h})}. \quad (10)$$

Anyone can show that minimizing $\text{KL}(b, p)$ is minimizing variational free energy:

$$F_b = U_b - S_b \quad (11)$$

with U_b and S_b the variational averaged energy and entropy, which depend on the arbitrary expression of b .

Bethe Approximation (BA) [29] is a variational approach coming from mean field theory [30] in which b is factorized over variable and check nodes:

$$\forall \mathbf{x} \in \{0, 1\}^N, \quad b(\mathbf{x}) \propto \frac{\prod_{a=1}^M b_a(\mathbf{x}_a)}{\prod_{i=1}^N b_i^{d_i-1}(x_i)}. \quad (12)$$

Denoting by d_i the degree of variable node X_i , variational quantities U_b and S_b are then defined as:

$$U_b = - \sum_{a=1}^M \sum_{\mathbf{x}_a} b_a(\mathbf{x}_a) \log f_a(\mathbf{x}_a), \quad (13)$$

$$S_b = - \sum_{a=1}^M \sum_{\mathbf{x}_a} b_a(\mathbf{x}_a) \log b_a(\mathbf{x}_a) - \sum_{i=1}^N (1 - d_i) \sum_{x_i} b_i(x_i) \log b_i(x_i). \quad (14)$$

As proved in [7], [19], stationary points of F_b computed by BA strictly equal BP fixed-points, *i.e.* minimizing F_b is determining beliefs $\{b_i\}_{1 \leq i \leq N}$ of variable nodes (5). Messages (3) and (4) stand for Lagrange multipliers [31] used to constrain the optimization [23].

IV. GENERALIZED BELIEF PROPAGATION

A. Region-based approximation

In [19], [32] authors introduced a generalization of (12) called the Region-Based Approximation (RBA). Instead of factorizing b according to variable and check nodes, RBA consists in factorizing b over subgraphs of \mathcal{G} , called regions, any of them being denoted by $r = \{\mathbf{X}_r \subseteq \mathbf{X}\} \cup \{\mathbf{F}_r \subseteq \mathbf{F}\} \cup \{E_r \subseteq E\}$. A set of regions \mathcal{R} is constrained to cover the whole Tanner graph:

- any region that includes a check node f_a has also to include \mathbf{X}_a ,
- all variable and check nodes must be included in at least one region.

A set \mathcal{R} that fulfills these constraints may be used to approximate joint distribution with:

$$b(\mathbf{x}) \propto \prod_{r \in \mathcal{R}} b_r^{c_r}(\mathbf{x}_r) \quad (15)$$

where b_r is the belief of region r . Bayesian rule constrains, by means of counting numbers $\{c_r\}_{r \in \mathcal{R}}$, each variable node to only contribute once to RBA, *i.e.* any X_i must participate once on each side of the equation, as in a chemical equation. According to [19], counting numbers are computed as:

$$\forall s \in \mathcal{R}, \quad c_s = 1 - \sum_{r \supset s} c_r \quad (16)$$

where $s \subset r$ is equivalent to $\mathbf{X}_s \subset \mathbf{X}_r$, $\mathbf{F}_s \subset \mathbf{F}_r$, $E_s \subset E_r$. We define a restrictive inclusion law: $s \prec r$ if and only if $s \subset r$ and no region t could be found in \mathcal{R} s.t. $s \subset t \subset r$. Relationships between regions are shared such that:

- $s \prec r$ means that s belongs to \mathcal{E}_r the set of *children* of region r and r belongs to \mathcal{P}_s the set of *parents* of s ,
- $s \subset r$ means that s belongs to \mathcal{D}_r the set of *descendants* of r and r belongs to \mathcal{A}_s the set of *ancestors* of s ,
- $r \cup \mathcal{D}_r$ is the family of r denoted by \mathcal{F}_r .

By associating to each region a node and to each restrictive inclusion an edge, we associate to any set of regions \mathcal{R} a graphical representation called a region-graph which construction rules are given in [33]. First regions selected to cover the whole Tanner graph are called clusters. Once they have been selected, we construct a second generation of regions made of intersections between nodes of the clusters. We continue this rule to build any generation l from the former region $l - 1$. A single Tanner graph may be mapped to numerous region-graphs, each one offering a RBA of specific accuracy. The main issue when dealing with RBA is thus the way the Tanner graph is clustered.

B. Message-passing

Equation (15) makes variational averaged energy and entropy be:

$$U_b = - \sum_{r \in \mathcal{R}} c_r \sum_{\mathbf{x}_r} b_r(\mathbf{x}_r) \log \prod_{f_a \in \mathcal{F}_r} f_a(\mathbf{x}_a) \quad (17)$$

$$S_b = - \sum_{r \in \mathcal{R}} c_r \sum_{\mathbf{x}_r} b_r(\mathbf{x}_r) \log b_r(\mathbf{x}_r). \quad (18)$$

Minimizing variational free energy F_b provides regions belief equation. For any $r \in \mathcal{R}$, for any state $\mathbf{x}_r \in \{0, 1\}^{|\mathcal{X}_r|}$:

$$b_r(\mathbf{x}_r) \propto \prod_{X_i \in \mathcal{X}_r} p_i(y_i|x_i) \prod_{p \in \mathcal{P}_r} m_{pr}(\mathbf{x}_r) \prod_{q \in \mathcal{D}_r} \prod_{s \in \mathcal{P}_q \setminus \mathcal{F}_r} m_{sq}(\mathbf{x}_q) \quad (19)$$

where m_{rs} is a message between connected regions $r, s \in \mathcal{E}_r$ with:

- $f_r(\mathbf{x}_r) = \prod_{f_a \in \mathcal{F}_r \setminus \mathcal{F}_s} f_a(\mathbf{x}_a)$,
- $p_r(\mathbf{y}_r|\mathbf{x}_r) = \prod_{X_i \in \mathcal{X}_r \setminus \mathcal{X}_s} p_i(y_i|x_i)$.

These messages are iteratively passed according to GBP equations where for any couple of regions $r, s \in \mathcal{E}_r$, for any state $\mathbf{x}_r \in \{0, 1\}^{|\mathcal{X}_r|}$, at any iteration $k \geq 1$:

$$m_{rs}^{(k)}(\mathbf{x}_s) = \frac{\sum_{\mathbf{x}_r \cup \mathbf{x}_s} f_r(\mathbf{x}_r) p_r(\mathbf{y}_r|\mathbf{x}_r) \prod_{\substack{u \in \mathcal{R} \setminus \mathcal{F}_r \\ v \in \mathcal{F}_r \setminus \mathcal{F}_s}} m_{uv}^{(k-1)}(\mathbf{x}_v)}{\prod_{\substack{u \in \mathcal{D}_r \setminus \mathcal{F}_s \\ v \in \mathcal{D}_s}} m_{uv}^{(k)}(\mathbf{x}_v)}. \quad (20)$$

As BP, GBP runs while messages do not converge or while output word $\hat{\mathbf{x}}$, computed according to (2), is not a codeword. To compute $\hat{\mathbf{x}}$ are needed beliefs on variable nodes. As region-graph does not systematically contain regions reduced to single variable nodes, these beliefs are determined by marginalization, *e.g.* for a variable node $X_i \in \mathbf{X}$, for any value $x_i \in \{0, 1\}$:

$$b_i^{(k)}(x_i) = \sum_{\mathbf{x}_r \cup x_i} b_r^{(k)}(\mathbf{x}_r) \quad (21)$$

where region r is the smallest region that contains X_i in \mathcal{R} .

V. A NOVEL TANNER GRAPH CLUSTERING

A. Systematic clustering

In [19] is introduced a region-graph construction in which each cluster is made of only one check node f_a and its neighborhood \mathbf{X}_a , such that the region-graph is firstly made of M clusters. The low density of LDPC codes implies that parity-check equations does not intersect a lot. Therefore, by this construction, the region-graph of any LDPC code is made of only two generations according to the construction rules mentioned before: the clusters generation and the next generation in which any region is made of a single variable node. The major advantage of this construction is that the implementation is easy as only the knowledge of parity-check equations is enough. Yet, RBA is aimed at offering better performance than BA lowering the influence of harmful topological structures of \mathcal{G} . The systematic construction does

not exhibit any connection to the Tanner graph topology, then GBP might not be more accurate than BP by this way.

The optimal but unrealistic clustering consists in gathering all variable and check nodes in a single cluster, GBP is then equivalent to compute Boltzmann's distribution (9) which is intractable. We specify that a relevant clustering is balanced between two crucial properties:

- any cluster is aimed at absorbing a harmful topological structure to reduce its effect on decoding,
- any cluster must be sensibly sized to make GBP of practical interest as a decoder.

The systematic construction fulfills the first property as check nodes of LDPC codes have several connections, but the first property is not taken into account.

B. Novel principle

Here, we extend a study carried out in [22] that helps improving GBP performance: for any region-graph, upper clusters are added to split, merge and remove regions under specific rules, detailed in this paper. The authors introduce the following assumption:

Welling's assumption: *Region-based approximation will improve if we add a new region to the region-graph.*

This assumption appears wise in the sense that, at the expense of the complexity as upper clusters are larger than former clusters, it is aimed at removing loops in region-graph. Tanner graph is a particular region-graph, therefore Welling's addition helps modify Tanner graph introducing clusters that we connect to check nodes.

In our work, we take into account that adding upper clusters might increase complexity in such a way that GBP is not practical. Thus, instead of gathering nodes of harmful structures inside clusters, we break them according to our novel construction principle:

Local loopfree principle: *Given a harmful subgraph \mathcal{T} of a Tanner graph \mathcal{G} , when breaking \mathcal{T} in n_c clusters, resulting region-graph \mathcal{R}_s made of these n_c clusters and their descendants must be loopfree such that GBP algorithm is locally optimal on \mathcal{R}_s .*

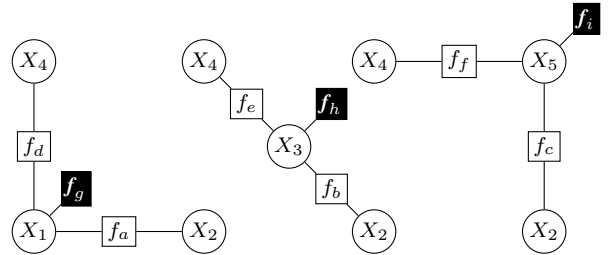


Fig. 4. Breaking a TS(5, 3) into three clusters

This principle is an extension of the Welling's addition that helps GBP keep a practical interest as it improves its decoding

performance. As our goal is to annihilate the error-floor, we apply our principle on trapping sets. We represent in Fig.4 the way we break a TS(5, 3) and in Fig.5 the resulting local region-graph on which GBP optimally performs as it is loopfree.

When applied on an LDPC code, *e.g.* the Tanner code, this construction cannot provide a whole loopfree region-graph, as trapping sets may have complex connections between them, that entails suboptimal GBP performance. Fortunately, we emphasize that region-graph loops are larger than Tanner graph loops by our construction, GBP is then less influenced by trapping sets than BP.

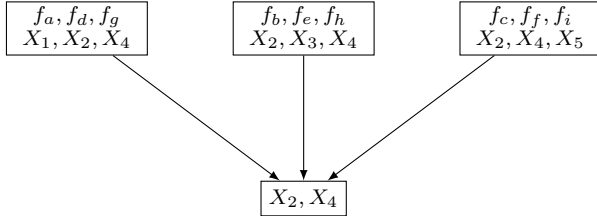


Fig. 5. Region-graph resulting from the split of a TS(5, 3)

VI. HYBRID DECODER

We introduced GBP to solve error events that are not decoded by BP, then it appears irrelevant to systematically use GBP even when BP well performs. In addition, even though our local loopfree principle helps not unreasonably increase complexity, BP is a faster decoder.

We then introduce a hybrid decoder that first runs BP, then runs GBP if BP fails. This decoder results in improved BER performance without seriously enlarging computation complexity. We performed it on the Tanner code entirely covered of trapping sets TS(5, 3) breaking them into three clusters as in Fig.5. We considered BSC and Additive White Gaussian Channel (AWGNC). BP and GBP decoders are run for at most 100 iterations, and quantities depicted in this section are averaged over $L = 10^{12}$ channel realizations to observe behavior in BP error-floor region.

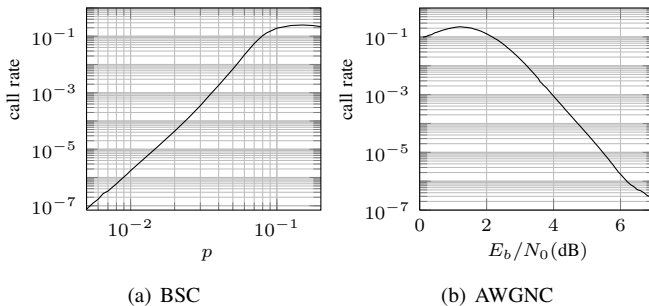


Fig. 6. GBP call rate in hybrid decoder on Tanner code

First quantity to observe in Fig.6 is the GBP call rate defined as the ratio between the number of BP failures and L . As channel noise power is lowered, GBP is less called given that BP is more and more efficient. A slope degradation takes place

on BSC around $p = 3.10^{-2}$ and around 5.95dB on AWGNC that matches with the error-floor emergence. Consequently, even though GBP calls still diminishes, GBP is necessary to decode problematic low-weight error events.

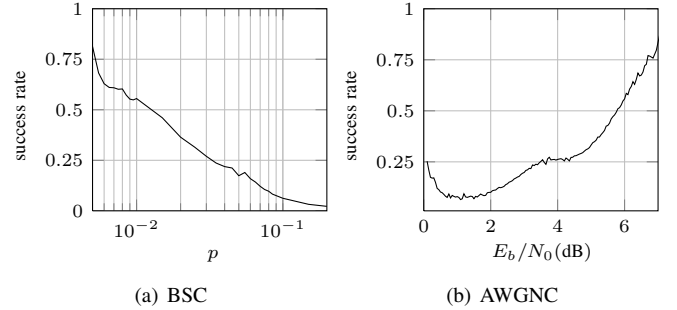


Fig. 7. GBP success rate in hybrid decoder on Tanner code

This change in GBP calls is shown in Fig.7 that depicts the success rate of GBP defined as the ratio between the number of GBP decodings that converge or that reach codewords and the number of GBP calls. We observe that as p is decreased and E_b/N_0 is increased, GBP is more successful. This confirms that our novel region-graph construction entails a decline in trapping sets influence. In addition, when approaching error-floor region, slope of success rate gradually increases, *i.e.* robustness against low-weight error events is stronger.

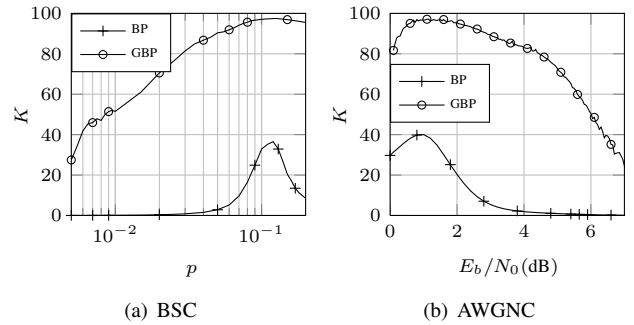


Fig. 8. Number of iterations of BP and GBP on Tanner code

Trapping sets are deeply connected one with each other, preventing region-graph from being loopfree. It entails that GBP decoding may not be as fast as BP when dealing with non problematic error events. We display in Fig.8 the number of iterations K needed to converge or to reach a codeword for BP and GBP. We see that for small values of p and large values of E_b/N_0 , BP does not require for many iterations as it is considered only if it performs well in our hybrid decoder. In contrast, GBP needs for a non-negligible number of iterations as it is used to treat non trivial error events. As an example, for $p = 10^{-2}$ and $E_b/N_0 = 6$ dB, $K_{BP} = 0.9$ and $K_{GBP} = 49.9$. Fortunately, as noise power is lowered, we see that K_{GBP} is gradually reduced, meaning that for very low values of p and very high values of E_b/N_0 , when BP fails, *i.e.* $k_{BP} = 100$, time spent to decode by GBP is not significant. Peculiar behavior around $p = 0.1$ and 1dB where decoders suffers from

an unexpected resonance are due to problem of dynamics that is extensively studied in [11].

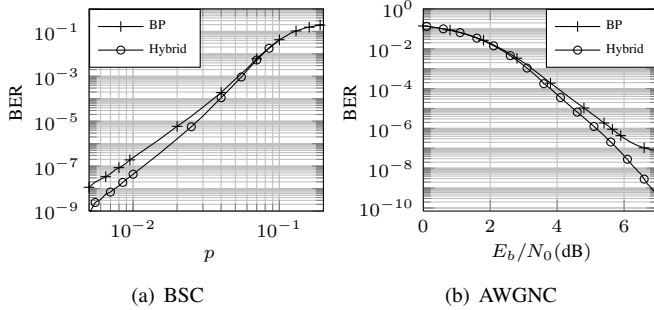


Fig. 9. Hybrid decoder BER on Tanner code

To evaluate decoding performance of hybrid decoder compared with BP, we represent in Fig.9 corresponding BER. Breaking trapping sets proves to be relevant as hybrid decoder outperforms BP, particularly for low values of channels noise power. As examples, we see for $p = 5 \cdot 10^{-2}$ that BER of hybrid decoder is ten times less than BER of BP, and that BP error-floor on AWGNC occurs around 5.8dB whereas GBP does not exhibit such a phenomenon even at 7.0dB. Especially on BSC, slope decreases less than BP slope, that indicates that:

- GBP suffers from an error-floor,
- all TS(5, 3) though wield less influence on GBP.

Error-floor is not completely defeated, as Tanner graph also contains other trapping sets of various topologies and sizes, see [14] for detail. Nevertheless, we succeed in making a decoder more robust against noise.

VII. CONCLUSION

In this paper, we introduced GBP algorithm to decode pathological error events for BP. We brought out a novel principle for constructing region-graph, that ensures local optimality of GBP. This helped break trapping sets to reduce their harmful influence on performance. Simulation results demonstrated that GBP running on the new region-graph clearly offers reliability and robustness compared with BP, at the expense of a slight increase in computation time that is though reduced as we increase channel noise power. Error-floor is then decreased, proving that breaking trapping sets ensures a better error correction capability, *i.e.* a better tolerance to low-weight error events.

REFERENCES

- [1] R.G. Gallager, "Low-Density Parity-Check Codes," Ph.D. dissertation, MIT, 1963.
- [2] D. J. C. MacKay and R. M. Neal, "Good codes based on very sparse matrices," in *Proc. 5th IMA Conference*, 1995.
- [3] J. Pearl, *Probabilistic Reasoning in Intelligent Systems: Networks of Plausible Inference*. Morgan Kaufmann, 1988.
- [4] F.R. Kschischang, B. J. Frey and H.A. Loeliger, "Factor Graphs and the Sum-Product Algorithm," *IEEE Transactions on Information Theory*, vol. 47, 2001, NO. 2.
- [5] S.Y Chung, T.J. Richardson and R.L. Urbanke, "Analysis of Sum-Product Decoding of Low-Density Parity-Check Codes Using a Gaussian Approximation," *IEEE Transactions on Information Theory*, vol. 47, 2001, NO. 2.

- [6] T.J. Richardson and R.L. Urbanke, "The Capacity of Low-Density Parity-Check Codes Under Message-Passing Decoding," *IEEE Transactions on Information Theory*, vol. 47, 2001, NO. 2.
- [7] T. Heskes, "Stable fixed points of loopy belief propagation are minima of the Bethe free energy," in *Advances in Neural Information Processing Systems 15*, Becker, S. and Thrun, S. and Obermayer, K., Ed. MIT Press, 2003, pp. 359–366.
- [8] —, "On the uniqueness of loopy belief propagation fixed points," *Neural Computation*, vol. 16, pp. 2379–2413, 2004.
- [9] K.P. Murphy, Y. Weiss and M.I. Jordan, "Loopy belief propagation for approximate inference: an empirical study," in *Proc. Uncertainty in Artificial Intelligence*, 1999.
- [10] X. Zheng, F.C.M. Lau, C.K. Tse and S.C. Wong, "Study of bifurcation behavior of LDPC decoders," *Int. Journal of Bifurcation and Chaos*, vol. 16, pp. 3435–3449, 2005.
- [11] J.-C. Sibel, S. Reynal and D. Declercq, "Evidence of chaos in the Belief Propagation for LDPC codes," *Chaotic Modeling and Simulations Journal*, vol. 6, 2013.
- [12] T.J. Richardson, "Error floors of LDPC codes," in *Proc. 41st Annual Allerton Conference on Communication, Control and Computing*, 2003.
- [13] S. Sankaranarayanan, S. K. Chilappagari, R. Radhakrishnan and B. Vasic, "Failures of the Gallager B Decoder: Analysis and Applications," in *Proc. Information Theory and Applications Workshop UCSD*, 2006.
- [14] B. Vasic, S.K. Chilappagari, D.V. Nguyen and S.K. Planjery, "Trapping set ontology," in *Proc. 47th Allerton Conference on Communication, Control and Computing*, 2009.
- [15] S.K. Planjery, D. Declercq, L. Danjean and B. Vasic, "Finite alphabet iterative decoders for LDPC codes surpassing floating-point iterative decoders," *Electronics Letters*, vol. 47, 2011.
- [16] D. Declercq, E. Li, S.K. Planjery and B. Vasic, "Approaching maximum likelihood decoding of finite length LDPC codes via FAID diversity," in *Proc. Information Theory Workshop*, 2012.
- [17] S.K. Planjery, D. Declercq, S.K. Chilappagari and B. Vasic, "Multilevel decoders surpassing belief propagation on the binary symmetric channel," in *Proc. International Symposium on Information Theory*, 2010.
- [18] J.S. Yedidia, "Generalized Belief Propagation and Free Energy Minimization," in *Proc. Neural Information Processing System*, 2002.
- [19] J.S. Yedidia, W.T. Freeman and Y. Weiss, "Constructing free energy approximations and Generalized Belief Propagation algorithms," *IEEE Transactions on Information Theory*, vol. 51, pp. 2282–2312, 2005.
- [20] M. Mezard and A. Montanari, *Information, Physics, and Computation*. Oxford University Press, 2006.
- [21] H. Nishimori, *Statistical physics of spin glasses and information processing an introduction*. Clarendon Press - Oxford, 2001.
- [22] M. Welling, "On the Choice of Regions for Generalized Belief Propagation," in *Proc. 20th Conference on Uncertainty in Artificial Intelligence*, 2004.
- [23] P. Pakzad et V. Anantharam, "Estimation and marginalization using Kikuchi approximation methods," *Neural Computation*, vol. 17, pp. 1836–1873, 2003.
- [24] R.M. Tanner, R. Michael, D. Sridhara and T. Fuja, "A Class of Group-Structured LDPC Codes," 2001.
- [25] R. Koetter and P. Vontobel, "Graph-covers and Iterative Decoding of Finite Length Codes," in *Proc. Int. Symp. on Turbo codes and Iterative Inform. Process.*, 2003.
- [26] A. Morello and V. Mignone, "DVB-S2: The Second Generation Standard for Satellite Broad-band Services," *Proc. IEEE*, vol. 94, 2006.
- [27] D.V. Nguyen, S.H. Chilappagari, M.W. Marcellin and B. Vasic, "On the Construction of Structured LDPC Codes Free of Small Trapping Sets," *IEEE Transactions on Information Theory*, vol. 58, 2012.
- [28] S. Kullback, *Information Theory and Statistics*. Dover, 1968.
- [29] H.A. Bethe, "Statistical Theory of Superlattices," in *Proc. Royal Society of London*, 1935.
- [30] "Advanced Mean Field Methods. Theory and Practice," M. Opper and D. Saad, Ed. MIT Press, 2001.
- [31] D.P. Bertsekas, *Nonlinear Programming*. Athena Scientific, 2001.
- [32] R. Kikuchi, "A Theory of Cooperative Phenomena," *Phys. Rev.*, vol. 81, pp. 988–1003, 1951.
- [33] J.-C. Sibel, S. Reynal and D. Declercq, "A novel region graph construction based on trapping sets for the Generalized Belief Propagation," in *Proc. Int. Conf. on Communication Systems*, 2012.