



An efficient pseudo-codeword search algorithm for Belief Propagation decoding of LDPC codes

Shahkar Kakakhail, Sylvain Reynal, David Declercq, Vincent Heinrich

► To cite this version:

Shahkar Kakakhail, Sylvain Reynal, David Declercq, Vincent Heinrich. An efficient pseudo-codeword search algorithm for Belief Propagation decoding of LDPC codes. International Conference on Ultra Modern Telecommunications & Workshops, 2009., Oct 2009, St. Petersburg, Russia. pp.1-5, 10.1109/ICUMT.2009.5345453 . hal-00540492

HAL Id: hal-00540492

<https://hal.science/hal-00540492>

Submitted on 26 Nov 2010

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.

An Efficient Pseudo-Codeword Search Algorithm for Belief Propagation Decoding of LDPC Codes

S. Kakakhail^{†*}, S. Reynal[†], D. Declercq[†], V. Y. Heinrich^{*}

[†]ETIS/ENSEA/UCP/CNRS UMR-8051

95014 Cergy-Pontoise, France

kakakhail,reynal,declercq@ensea.fr

^{*}STMicroelectronics

Crolles, France

vincent.heinrich@st.com

Abstract

We introduce the use of Fast Flat Histogram (FFH) method employing Wang Landau Algorithm in an adaptive noise sampling framework using Random Walk to find out the pseudo-codewords and consequently the pseudo-weights for the Belief Propagation (BP) decoding of LDPC codes over an Additive White Gaussian Noise (AWGN) channel. The FFH method enables us to tease out pseudo-codewords at very high Signal-to-Noise Ratios (SNRs) exploring the error floor region of a wide range of codes varying in length and structure. We present the pseudo-weight (effective distance) spectra for these codes and analyze their respective behavior.

I. INTRODUCTION

Low Density Parity Check (LDPC) codes [1] make a class of forward error correcting codes which employ a computationally efficient iterative decoding scheme based on a message passing algorithm. The decoding process is, however, known to be subject to decoding failures due to the so-called pseudo-codewords. The failures can cause the high Signal-to-Noise Ratio (SNR) performance of message passing decoding to be worse than that predicted by the maximum likelihood decoding union bound in the error floor regime [2] which is characterized by very low error rates. Standard Monte Carlo (SMC) simulation which comprises of simulating a system by generating random inputs according to a probability distribution and then evaluating the system response, becomes extremely cumbersome at such high SNRs. With the advancement in the code design and better decoders, it has become very important to gauge the performance of the system in the error floor regime.

To explore the error floor phenomenon, a physics inspired approach coined as instanton amoeba was

proposed and developed in [3], [4], [5]. The scheme is generic in that there are no restrictions related to decoding or channel. Chertkov *et al.* [6] presented the pseudo-codeword landscape using an efficient pseudo-codeword search algorithm detailed in [7]. The algorithm is mainly valid for Linear Programming (LP) decoding [8] and the authors reported that a direct attempt to extend the LP-based pseudo-codewords search algorithm to Belief Propagation (BP) decoding [1],[9] did not yield desirable results.

In the pseudo-codeword literature, LP decoding has been predominantly used as it proposes to relax the polytope, expressing σ in terms of a linear combination of local codewords. If the LP decoding does not decode to a correct codeword then it usually yields a non-codeword pseudo-codeword which is a special configuration of beliefs containing some rational values [10]. Pseudo-codewords are not codewords in general but codewords are pseudo-codewords [11]. The nature of pseudo-codewords with different origins is further investigated in [12],[13].

To characterize the pseudo-codewords, the notion of fundamental polytope was introduced in [14] which is the most important concept relevant to pseudocodewords found through LP decoding. It was also argued that the large minimum distance of the code does not determine the performance of the code if the code has low pseudo-weight spectrum. For the sum-product decoding, if the messages are converged, then the vector formed by the marginal probabilities of having a bit position in the state 1 is a fundamental polytope vector. More about this vector and the Bethe variational free energy can be found in [15].

In this paper, we investigate the use of a physics inspired algorithm known as Fast Flat Histogram (FFH) method [16] which has already been implemented for the efficient performance evaluation of forward error correcting codes [17], [18]. The method consists of a random walk scheme employing Wang-Landau Algorithm

in adaptive noise sampler framework. We employ our scheme to find the pseudo-codewords in the high SNR region using the BP decoding algorithm on an AWGN channel.

The rest of the paper is organized as follows: Section 2 sets the background for the use of FFH method employing BP decoding for the search of pseudo-codewords. Section 3 gives our numerical scheme and its validation. Section 4 reports the results for our test-bench containing regular, irregular and cyclic LDPC codes. Section 5 concludes our work.

II. NOTATIONS AND BACKGROUND

The background and notations for this work remains the same as in [5],[6]. Sending a codeword $\sigma = \{\sigma_i = \pm 1; i = 1, \dots, N\}$ into a noisy channel results with the probability $P(\mathbf{x}|\sigma)$ in corruption of the original signal, $\mathbf{x} \neq \sigma$. The decoding goal is to infer the original message from the received output $\mathbf{x}$. Assuming that coding and decoding are fixed, one studies Frame Error Rate (FER) to characterize performance of the scheme $FER = \int d\mathbf{x} \chi_{error}(\mathbf{x}) P(\mathbf{x}|\mathbf{1})$, where $\chi_{error} = 1$ if an error is detected and $\chi_{error} = 0$ otherwise. In symmetric channel, FER is invariant with respect to the original codeword, thus all- $(+1)$ codeword can be assumed for the input. AWGN channel is defined by,

$$P(\mathbf{x}|\sigma') = \prod_i p(x_i|\sigma'_i, p(x|\sigma)) \propto \exp\left(-\frac{(x-\sigma)^2 s^2}{2}\right) \quad (1)$$

If the detected signal at a bit is x , the respective likelihood at the bit is $h = (p(x|1)/p(x|-1))$. These likelihoods are translated into beliefs $b_i(\sigma_i)$ which are defined as trial probabilities for bit i to be in the state σ_i . Belief Propagation constitutes in iteratively propagating messages through different nodes of the code graph following message update rules and computing beliefs using certain non-linear equations called BP equations. The BP equations are equations for extrema of the Bethe free energy [15].

At high SNR, the difference between the performance of Maximum Likelihood (ML) decoding and approximate decoding (BP, LP, etc.) is due to the pseudo-codewords. This performance is gauged in terms of Frame Error Rate (FER) which calculates the probability of a decoding failure. For AWGN channel, the actual asymptotics of the performance curves (FER vs SNR) of ML and BP decoding at very high SNRs, in the so-called error-floor region, are $FER_{ML} \sim \exp(-d_{ML} \cdot s^2/2)$ and $FER_{BP} \sim \exp(-d_{BP} \cdot s^2/2)$ where d_{ML} is the Hamming Distance of the code and d_{BP} is the effective distance of the code, specific to BP decoding.

BP decoding turns into LP decoding at $SNR \rightarrow \infty$. In the high SNR (error floor) region, the values of FER are inaccessible by Monte-Carlo simulations. It is in this context, that we use FFH method which comprises of a Markov Chain Monte Carlo (MCMC) sampler capable of sampling the noise vectors from the tails of the AWGN probability density function. Suppose a pseudo-codeword $\tilde{\sigma} = \{\tilde{\sigma}_i = b_i(1); i = 1, \dots, N\}$ corresponding to the most damaging configuration of the noise (instanton) is found. Then the effective distance is given by the same formula $d_{eff} = (\sum_i \tilde{\sigma}_i)^2 / \sum_i \tilde{\sigma}_i^2$ as in [6],[14]. This definition of the effective distance was first described in [19] where the formulas derived by Wiberg *et al.* [20], [21] for AWGN channels were extended to non-binary codes, Binary Erasure Channel (BEC) and Binary Symmetric Channel (BSC).

III. FAST FLAT HISTOGRAM METHOD

A. Description

The basic skeleton of our technique is the same as that in [17],[18]. Let Γ be the n -dimensional probability space of the noise in the n bits of a codeword. The noise vector $\mathbf{z} = (z_1, z_2, \dots, z_n)$ is a multivariate Gaussian with joint pdf $\rho(\mathbf{z}) = \prod_{l=1}^n \rho_l(z_l)$. The transmitted bit vector is represented by $\mathbf{t} = (t_1, t_2, \dots, t_n)$ and $\mathbf{x} = (x_1, x_2, \dots, x_n)$ represents the received codeword. The algorithm is controlled by a scalar control quantity V given as $V(\mathbf{z}) = \left[\frac{1}{n} \sum_{l=1}^n [t_l - z_l]^2\right]^{1/2}$ where t_l and z_l are the transmitted bit and the noise value in the l th position respectively. This definition of $V(\mathbf{z})$ is different from the one that we used in [17],[18].

Given a range $[V_{min}, V_{max}]$ for V , Γ is partitioned into L subsets $\Gamma_k = \{\mathbf{z} \in \Gamma | V_{k-1} \leq V(\mathbf{z}) < V_k\}$, where $V_k = V_{min} + k\Delta V$, $1 \leq k \leq L$ and $\Delta V = V_k - V_{k-1} = (V_{max} - V_{min})/L$ is the width of each bin in the partition of $[V_{min}, V_{max}]$.

Let P_k be the probability of selecting a realization $\mathbf{z}$ from ρ such that $\mathbf{z} \in \Gamma_k$ [22]. Then,

$$P_k = \int_{\Gamma} \chi_k(\mathbf{z}) \frac{\rho(\mathbf{z})}{\rho^*(\mathbf{z})} \rho^*(\mathbf{z}) d\mathbf{z} \approx \frac{1}{N} \sum_{i=1}^N \chi_k(\mathbf{z}^{*,i}) \frac{\rho(\mathbf{z}^{*,i})}{\rho^*(\mathbf{z}^{*,i})} \quad (2)$$

where $\rho^*(\mathbf{z})$ is a positive biasing pdf, $\chi_k = 1$ if $\mathbf{z} \in \Gamma_k$ and $\chi_k(\mathbf{z}) = 0$ otherwise. $\mathbf{z}^{*,i}$ are N random sample points in Γ selected according to the pdf $\rho^*(\mathbf{z})$. The variance of the estimate of (2) is zero if the optimal biasing pdf $\rho_{opt}^*(\mathbf{z}) = \chi_k(\mathbf{z})\rho(\mathbf{z})/P_k$ is used. However, $\rho_{opt}^*(\mathbf{z})$ depends on P_k which is initially unknown. In standard IS, one uses physical intuition to guess a biasing pdf that is close to ρ_{opt}^* . The FFH method instead iterates over

a sequence of biasing pdfs $\rho^{*,j}$ that approach ρ_{opt}^* . We define $\rho^{*,j}$ for j th iteration by $\rho^{*,j}(\mathbf{z}) = \rho(\mathbf{z})/(c^j P_k^j)$ where k is such that $\mathbf{z} \in \Gamma_k$ is satisfied. The quantities P_k^j satisfy $P_k^j > 0$ and $\sum_{k=1}^L P_k^j = 1$ and c^j is an unknown constant that ensures $\int_{\Gamma} \rho^{*,j}(\mathbf{z}) d\mathbf{z} = 1$. The vector P_k completely determines the bias and is initialized with $1/L, \forall k = 1, \dots, L$.

Our aim is to explore the whole of probability space Γ using random walk [23]. By employing Metropolis algorithm [24], we produce a random walk of *samples* $\mathbf{z}^{*,i}$ whose pdf equals $\rho^{*,j}(\mathbf{z})$. We consider a Markov chain of transitions consisting of small steps in the noise space. Each transition goes from $\mathbf{z}^{*,i} = \mathbf{z}_a^* \in \Gamma_{k_a}$ to $\mathbf{z}_b^* = (\mathbf{z}_a + \epsilon \Delta \mathbf{z}) \in \Gamma_{k_b}$ where $\Delta \mathbf{z}$ is random and symmetric, i.e., it does not favor any direction in Γ and the transition is accepted with probability π_{ab} . Here, ϵ is the perturbation constant. If a transition from $\mathbf{z}^{*,i}$ to $\mathbf{z}_b^*$ is accepted, we set $\mathbf{z}^{*,i+1} = \mathbf{z}_b^*$, else we set $\mathbf{z}^{*,i+1} = \mathbf{z}^{*,i} = \mathbf{z}_a^*$. The ratio π_{ab}/π_{ba} equals $\rho^{*,j}(\mathbf{z}_b^*)/\rho^{*,j}(\mathbf{z}_a^*)$ which is the *detailed balance equation* that ensures that the limiting (stationary) pdf for infinitely many steps of this random walk is $\rho^{*,j}$ [24].

We consider the perturbation of the noise component in each bit $z_{a,l}^*$ of $\mathbf{z}_a^*$ separately and accept it or reject it independently with the probability $\min[\rho(z_{b,l}^*)/\rho(z_{a,l}^*), 1]$. We pick each perturbation Δz_l from a zero mean symmetric pdf. We obtain a trial state $\mathbf{z}_b^*$ in which only some of the components are different from their previous values in $\mathbf{z}_a^*$. Then we compute k_b , the bin corresponding to $\mathbf{z}_b^*$ and finally accept the step from $\mathbf{z}_a^*$ to $\mathbf{z}_b^*$ with the probability $\min[P_{k_a}^j/P_{k_b}^j, 1]$. The compound transition probability thus becomes

$$\pi_{ab} = \left\{ \prod_{l=1}^n \min \left[\frac{\rho(z_{b,l}^*)}{\rho(z_{a,l}^*)}, 1 \right] \right\} \min \left[\frac{P_{k_a}^j}{P_{k_b}^j}, 1 \right] \quad (3)$$

The Asymptotically Optimal Acceptance Rate AOAR $\alpha \triangleq (\text{number of accepted steps})/(\text{total number of steps})$ for a Metropolis algorithm for target distributions with IID components is 0.234 [25]. The perturbation constant ϵ is adjusted so as to keep α close to this value. The noise realizations are recorded in the histogram $H^{*,j}$ where $H_k^{*,j} = \sum_{i=1}^N \chi_k(\mathbf{z}^{*,i})$ is the number of $\mathbf{z}^{*,i}$ in iteration j that fall into Γ_k . Each noise vector is used in the channel to deteriorate the transmitted codeword which is then fed into the decoders to verify if the errors are corrected within the specifies number of decoder iterations.

P_k is updated *on the fly* such that when k bin is visited, P_k is modified by the refinement parameter $f > 1$, i.e. $P_k \rightarrow P_k \cdot f$ [26], [27]. In practice, we have to use the log domain $\ln P_k \rightarrow \ln P_k + \ln f$ in order to fit all

possible P_k into double precision numbers. If the random walk rejects a possible move and stays in the same bin k , we modify the same P_k with the modification factor to keep the detailed balance equation in equilibrium.

In case of rejection of a possible move, a very significant additional step is to permute the components of the noise vector and to add this permuted sequence to the transmitted codeword on which decoding is carried out for error correction. It is important to note that the random walk is performed with the new permuted sequence so as not to disturb the detailed balance equilibrium. We keep on permuting the noise components until a possible move is accepted. The preceding step stems from the fact that different sequences of the same components of a noise vector lead to different decoding outputs since we are employing a message passing decoding algorithm. The orientation of the permuted noise components remain the same leading to the same V value and consequently staying in the same bin. Without effecting the basic modification of P_k values, we are thus able to check the system response of all entries in histogram $H^{*,j}$ thus adding to the robustness of the method.

It is to be noted that if the proposed noise vectors which move the system outside the permitted $[V_{min}, V_{max}]$ interval are systematically rejected, the P_k value would increase at edges by an unwanted excessive amount. This problem is counteracted by adopting the n -fold way [28], i.e., leaving P_k value unchanged whenever a move update attempts to take the system outside the allowed interval.

The histogram $H_k^{*,j}$ is checked after about each $10L$ Monte Carlo (MC) sweeps. When the histogram is flat (flatness criterion is the same as in [26], [27]), the modification factor is reduced to a finer one using the function $f_{j+1} = \sqrt{f_j}$ ($f_{init} = e = 2.7182818$), the histogram is reset and the next iteration of random walk is started where P_k are now modified with the finer modification factor. We continue doing so until the histogram is flat again and then we begin the next Wang-Landau (WL) iteration with a finer f and so on. The above detailed random walk can also be carried out in a parallel fashion by dividing the range $[V_{min}, V_{max}]$ into W partitions and then exploring each partition separately, combining the results in the end.

It is extremely important to determine the optimum $[V_{min}, V_{max}]$ interval with the optimum number of bins since the accuracy and speed of the simulation depend heavily on it. Following is a self adaptive procedure to determine this interval which intrinsically takes into account the code length and the code error correcting capacity. Lines of similarity can be drawn between our procedure of determining the optimum $[V_{min}, V_{max}]$ interval and Domain Sampling Run of [29].

$[V_{min}, V_{max}]$ is initialized to $[0, 5]$ and this interval is divided into 1000 bins. Let the Global Acceptance Ratio (GAR) correspond to the ratio of the number of accepted noise vectors to the number of noise vectors produced in total. We initialize GAR with a value (0.3 in our case). The bins are initialized with $P_k = 1/1000, \forall k = 1, \dots, 1000$. Now the random walk is performed to produce noise vectors for which the corresponding bins are visited with the consequent update of the P_k value. With the bin filling, we start getting rejections for the proposed move. At each step, we calculate the GAR and as soon as we obtain its pre-defined value, the walk is ceased and the farthest bins on either side are detected which were approached by the random walk. These two bins on either side determine the $[V_{min}, V_{max}]$ interval. While determining the interval, the noise vectors produced are not added to the codewords and no decoder runs are performed. Their sole purpose is to locate the bins naturally accessible for the code.

IV. RESULTS AND DISCUSSION

Our test-bench comprises of six codes namely Tanner [155, 64, 20] code [30], Margulis $p = 7$ [672, 336, 16] code [31]; [648, 324, 15], [1296, 648, 23] and [1944, 972, 27] codes from the 802.11 draft [32] and the [504, 252, 13] irregular Progressive Edge Growth (PEG) code [33]. The MHDs of the last four codes are measured through the improved impulse method [34]. BPSK modulation is employed using symmetric signal levels of +1 and -1 for logical 0s and 1s respectively. An all zeros codeword is transmitted since the code is linear and the noise is symmetric. We employ 1000 decoding iterations in our BP decoding so the pseudo-codewords correspond to the instantons which could survive such a high number of decoding iterations.

Fig. 1 depicts the frequency spectra for the codes under study. For [155, 64, 20] and [672, 336, 16] codes, our conclusions are the same as in [6]. We observe that the two codes demonstrate qualitatively different features for the pseudo-codeword frequency spectra. Pseudo-codeword spectrum for [155, 64, 20] code starts with the lowest effective distance ≈ 10.004 and grows up going through the fundamental polytope pseudo-codewords at effective distance ≈ 19.98 (convergence to valid codewords). In case of [672, 336, 16] code, the spectrum starts at effective distance ≈ 12.056 but grows abruptly to the fundamental polytope pseudocodewords at effective distance ≈ 15.66 (valid codewords).

For the 802.11 cyclic LDPC codes, we observe that there is a significant increase in effective distance with the increase in code-length. As compared to the preceding two codes, these three cyclic codes exhibit a

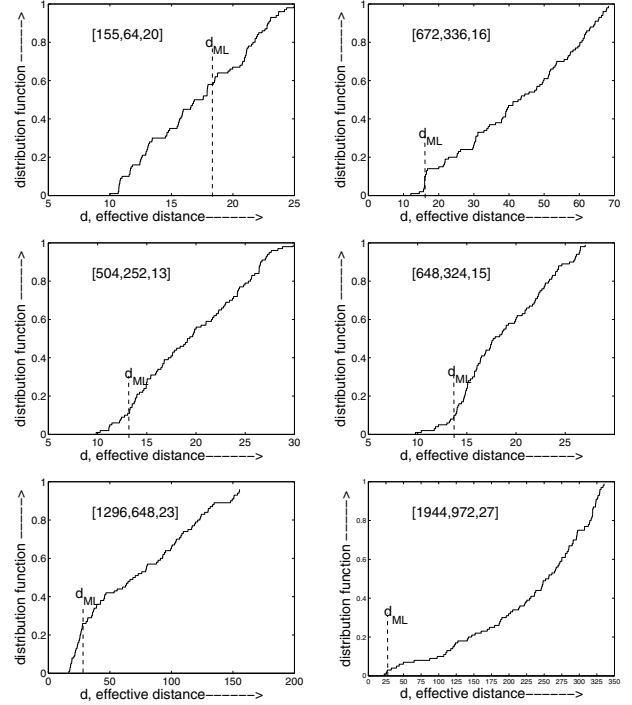


Figure 1. The frequency spectrum of the effective distance constructed using FFH method as the pseudo-codeword search algorithm for six codes: Tanner [155, 64, 20]; Margulis $p = 7$ [672, 336, 16]; 802.11 standard [648, 324, 15], [1296, 648, 23] and [1944, 972, 27]; irregular progressive edge growth [504, 252, 13]

relatively high number of fundamental polytope pseudo-codewords. Convergence to invalid codewords increases with the increase in code-length. The least effective distances of the fundamental polytope pseudo-codewords for [648, 324, 15], [1296, 648, 23] and [1944, 972, 27] codes that we found are 14.64 (valid codeword), 26.48 (invalid codeword) and 64.24 (invalid codeword) respectively. In the case of irregular PEG code, the pseudo-codeword spectrum is similar to the cyclic codes. The least effective distance ≈ 9.855 and the least effective distance of fundamental polytope pseudo-codeword is 12.74 (valid codeword).

We also applied our method to Margulis $p = 11$ [2640, 1320, 40] code and found the least effective distance ≈ 54.055 . We observe that the fundamental polytope pseudo-codewords are in dominance however the messages converge to invalid codewords. The observations made by Koetter *et al.* [14] that the pseudo-weights are far more important than the Hamming Distance are further bolstered when we analyze the pseudo-codeword spectra in our case. For example, the Hamming Distance of [1944, 972, 27] code is only slightly higher than [1296, 648, 23] code, however there is a big difference in their pseudo-

codeword spectra. Similarly, the Hamming Distance of [155, 64, 20] code is higher than [672, 336, 16] code, the latter performs better in terms of pseudo-weight. How this pseudo-weight spectrum depends on the code-length or other code properties is a subject of ongoing research.

V. CONCLUSIONS

In this paper, we have proposed the use of Fast Flat Histogram method to find the pseudocodewords for different codes using BP decoding. The FFH method is a powerful tool to explore the code performance at very high SNRs (in the error floor region) which is otherwise computationally intractable using standard Monte Carlo simulation. Since the decoder failures in the error floor region are mostly due to pseudo-codewords, the FFH method is an excellent means to study the code behavior at high SNRs. Our future work consists of integrating the FFH method in multiple error impulse framework to increase its effectiveness.

VI. ACKNOWLEDGEMENTS

We wish to thank Pascal O. Vontobel for the fruitful discussions related to this work. His website www.pseudocodewords.info also proved very useful. We are grateful to Mikhail G. Stepanov for providing the matrices of Margulis [672, 336, 16] and Tanner [155, 64, 20].

REFERENCES

- [1] R. G. Gallager. "Low Density Parity Check Codes". PhD thesis, MIT Press, Cambridge, 1963.
- [2] T. Richardson. "Error floors of LDPC codes". *Allerton Conference*, 2001.
- [3] V. Chernyak, M. Chertkov, M.G. Stepanov, and B. Vasic. "Error correction on a tree: an instanton approach". *Phys. Rev. Lett.*, 93(198702), 2004.
- [4] M.G. Stepanov, V. Chernyak, M. Chertkov, and B. Vasic. "Diagnosis of weaknesses in modern error correction codes: a physics approach". *Phys. Rev. Lett.*, 95(228701), 2005.
- [5] M.G. Stepanov and M. Chertkov. "Instanton analysis of Low-Density-Parity-Check codes in the error floor regime". In *IEEE Intern. Symp. on Inform. Theory*, Seattle, USA, July 2006.
- [6] M. Chertkov and M. Stepanov. "Pseudo-codeword landscape". In *IEEE Intern. Symp. on Inform. Theory*, Nice, France, June 2007.
- [7] M. Chertkov and M.G. Stepanov. "An efficient pseudo-codeword search algorithm for linear programming decoding of LDPC codes". *IEEE Trans. Inf. Theory*, IT-54(4):1514–1520, Apr. 2008.
- [8] J. Feldman, M. Wainright, and D. R. Karger. "Using linear programming to decode linear codes". In *Conference on Information Sciences and Systems*, The John Hopkins University, Mar. 2003.
- [9] D.J.C. MacKay. "Good error correcting codes based on very sparse matrices". *IEEE Trans. Inf. Theory*, 45(2):399–431, 1999.
- [10] P.O. Vontobel and R. Koetter. "On the relationship between LP decoding and Min-Sum Algorithms decoding". In *ISITA*, Parma, Italy, 2004.
- [11] R. Smarandache, A. Pusane, P. O. Vontobel, and Jr. D. J. Costello. "Pseudo-codewords in LDPC convolutional codes". In *IEEE Intern. Symp. on Inform. Theory*, Seattle, USA, July 2006.
- [12] N. Axvig, D. Dreher, K. Morrison, E. Psota, L.C. Perez, and J. Walker. "A universal theory of decoding and pseudocodewords". In *Allerton Conference on Communication, Control, and Computing*, Monticello, IL, USA, Sep. 2007.
- [13] C. Kelley and D. Sridhara. "Pseudocodewords of Tanner graphs". *IEEE Trans. Inform. Theory*, 53(11):4013–4038, Nov. 2007.
- [14] R. Koetter and P.O. Vontobel. "Graph covers and iterative decoding of finite-length codes". In *3rd Intl. Symp. on Turbo Codes and Related Topics*, pages 75–82, Brest, France, Sept. 2003.
- [15] J.S. Yedidia, W.T. Freeman, and Y. Weiss. "Constructing free energy approximations and generalized belief propagation algorithms". *IEEE Trans. Inf. Theory*, IT-51(2282), 2005.
- [16] S. Reynal and H. T. Diep. "Fast flat histogram algorithm for generalized spin models". *Phys. Rev.*, E 72(56710), 2005.
- [17] S. Kakakhail, S. Reynal, D. Declercq, and V. Y. Heinrich. "Fast simulation for the performance evaluation of LDPC codes using Fast Flat Histogram Method". In *IEEE Sarnoff Symposium*, Princeton NJ, USA, Apr. 2008.
- [18] S. Kakakhail, S. Reynal, D. Declercq, and V. Y. Heinrich. "Efficient Performance Evaluation of Forward Error Correcting Codes". In *IEEE ICCS'08*, Guangzhou, China, Nov. 2008.
- [19] Jr. G.D. Forney, R. Koetter, F.R. Kschischang, and A. Reznik. "On the effective weights of pseudocodewords for codes defined on graphs with cycles". *Codes, Systems and graphical models (Minneapolis, MN, 1999) IMA Vol; Math. Appl.*, 123:101–112, 2001.
- [20] N. Wiberg. "Codes and decoding on general graphs". PhD thesis, Linköping University, 1996.
- [21] N. Wiberg, H.-A. Loeliger, and R. Koetter. "Codes and iterative decoding on general graphs". *Europ. Trans. Telecommunications*, 6(513), 1995.
- [22] R. Holzlohner, A. Mahadevan, C. Menyuk, J. Morris, and J. Zweck. "Evaluation of the very low BER of FEC codes using dual adaptive importance sampling". *IEEE Comm. Letters*, vol 2, Feb. 2005.
- [23] W. Krauth. "Introduction to Monte Carlo Algorithms". *Advances in computer simulation, lectures held at Eotvos summer school in Budapest, Hungary*, 1996.
- [24] N. Metropolis, A. W. Rosenbluth, M. N. Rosenbluth, A. M. Teller, and E. Teller. "Equation of state calculations by fast computing machines". *J. Chem. Phys.*, vol 21:1087–1092, 1953.
- [25] G. O. Roberts, A. Gelman, and W. R. Gilks. "Weak Convergence and Optimal Scaling of Random Walk Metropolis Algorithms". *Ann. Appl. Probab.*, vol 7:110–20, 1997.
- [26] F. Wang and D. P. Landau. "Determining the density of states for classical statistical models: a random walk algorithm to produce a flat histogram". *Phys. Rev.*, E 64(056101), 2001.
- [27] F. Wang and D. P. Landau. "Efficient, multiple range random walk algorithm to calculate the density of states". *Phys. Rev. Lett.*, 86(2050), 2001.
- [28] A. B. Bortz, M. H. Calos, and J. L. Lebowitz. *J. Comput. Phys.*, 17(10), 1975.
- [29] A. Troster and C. Dellago. *Phys. Rev.*, E 71(066705), 2005.
- [30] R.M. Tanner, D. Sridhara, and T. Fuja. "A class of group-structured LDPC codes". In *ISCTA*, Ambleside, England, 2001.
- [31] G.A. Margulis. "Explicit construction of graphs without short cycles and low density codes". *Combinatorica*, 2(71), 1982.
- [32] IEEE P802.11n/D1.05. *Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) specifications - Enhancements for Higher Throughput (Draft)*, Oct. 2006.
- [33] D.J.C. MacKay. Encyclopedia of sparse graph codes. <http://www.inference.phy.cam.ac.uk/mackay/codes/EN/C>.
- [34] D. Declercq and M. Fossorier. "Improved impulse method to evaluate the low weight profile of sparse binary linear codes". In *IEEE Intern. Symp. on Inform. Theory*, Toronto, Canada, July 2008.