Effect of the threshold offset on the performance of UEP LDPC codes

Abdul Wakeel*, Werner Henkel*, and Feifei Gao**
*Jacobs University Bremen, 28759 Bremen, Germany
{a.wakeel, w.henkel}@jacobs-university.de
**Tsinghua National Laboratory for Information Science and Technology, Beijing, 100084, China

***Abstract*—This letter addresses the effect of threshold offset on the average performance and UEP capability of higher order constellation (HOC) unequal error protection low density parity check (UEP-LDPC) codes. We optimize the variable node degree distribution for an HOC UEP-LDPC code at a minimum threshold, with a fixed check node degree distribution and code rate. The average performance and the UEP capability of the ensemble code for a marginal offset in the threshold value were investigated. The simulation results show that the average performance and UEP capability can be enhanced by a slight offset in the threshold value. We looked into the connectivity between protection classes due to the threshold offset for a possible answer to the performance gain.**

***Index Terms*—Unequal error protection, LDPC, parity-check matrix, ACE algorithm, higher order constellation**

I. Introduction

Multimedia data usually have different sensitivities against transmission errors. There is often some data which is more important than other, for example the header of a frame is more important than the payload. Thus, the transmission system should provide unequal error protection (UEP) in order to account for the different properties of data. Suitable UEP may increase the perceived performance for applications where different bits have different sensitivities to errors. In multilevel coding, UEP is provided by assigning data to various levels depending on the sensitivity of the data, each level being protected by individual binary codes.

In here[1], we design an irregular Low Density Parity Check (LDPC) for UEP transmission using high order constellations (HOC) by exploiting the inherent UEP of the HOC where source bits have different error sensitivities. We optimize the variable node degree distribution after splitting it into sub-degree distributions for different protection and modulation classes at a minimum threshold with fixed check node degree distribution. We investigate the performance of the ensemble LDPC code for a threshold offset. It has been shown through simulation results that the average performance and the UEP capabilities are enhanced by a marginal threshold offset.

The rest of the paper is organized as follow: The system model is discussed in Section II, Section III presents the LDPC code construction for higher order constellation. Section IV provides the simulation results, and the paper is summarized in Section V.

[1]This paper is an extension to [7].

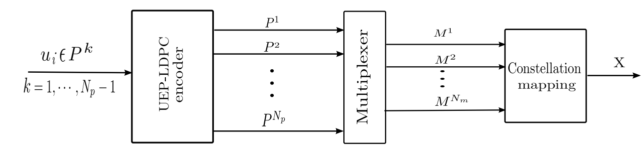

Fig. 1: Signal flow diagram of the system model, the information bits are first encoded by a UEP-LDPC code, re-multiplexed and assigned to different modulation classes before constellation mapping.

II. System Model

As shown in Fig.1, the information bits u_i are first assigned to different $N_p - 1$ protection classes, defined by the source coding unit with the parity bits being assigned to the least protection class N_p. The bits of different protection classes are then re-multiplexed and assigned to different modulation classes $M^1, \ldots, M^{N_m}$. Since UEP is desired, high order constellations with $M = 2^{l_m}$ constellation points are considered which already provide some UEP to the individual bits of a symbol. The bits inside a signal points are labeled by binary input vectors $\mathbf{d} = (d_{l_m-1}, \ldots, d_1, d_0)$. The input alphabet of size $M_m = 2^{l_m}$ can thus, be represented by l_m equivalent binary input channels. The noise vector $\sigma^2 = [\sigma_1^2 \ \ldots \ \sigma_{N_m}^2]$ is defined to be a vector that contains the equivalent noise variances for each individual bit-error rate, i.e., for each modulation class, ordered with the lowest variance first. For example, the expression for the individual bit error probabilities of an 8-PSK in a symbol are

$$P_{b,b_0} \approx \frac{1}{2} \cdot \text{erfc}\left(\sqrt{3 \cdot E_b/N_0} \sin\frac{\pi}{8}\right) , \quad (1)$$

$$P_{b,b_1} = P_{b,b_2} \approx \frac{1}{4} \cdot \text{erfc}\left(\sqrt{3 \cdot E_b/N_0} \sin\frac{\pi}{8}\right) , \quad (2)$$

where "erfc" stands for the complementary error function and E_b/N_0 is the signal-to-noise ratio per bit. For different modulation classes, the individual noise variance is computed from the corresponding error probabilities assuming an equivalent BPSK channel [7]. Thus, the equivalent noise variance is obtained as

$$\sigma_j^2 = \frac{1}{2\left(\text{erfc}^{-1}(2P_{b,b_j})\right)^2} . \quad (3)$$

III. LDPC code construction for HOC

LDPC codes are linear block codes with a sparse parity-check matrix $\mathbf{H}$, invented by Gallager in 1963 [1]. They exhibit a performance very close to the capacity. The parity-check matrix $\mathbf{H}$ has dimension $(n-k) \times n$, where k and n are the lengths of the information word and the codeword, respectively. Graphically, LDPC codes are represented by a bipartite graph (Tanner graph) consisting of two types of nodes, the check nodes and variable nodes which are connected through edges. An edge corresponds to a non-zero entry of $\mathbf{H}$. The total number of edges connected to a node is called the degree of that node. An $\mathbf{H}$ matrix with the same degree of variable nodes and check nodes is termed regular LDPC code. If the degree of the check nodes / variable nodes vary, then it is called an irregular LDPC code. The variable nodes and check nodes of an irregular LDPC code are represented by polynomials, defined from an edge and node perspective. From an edge perspective, the variable node and check node polynomials of an irregular LDPC codes are defined as [7]

$$\lambda(x) = \sum_{i=2}^{d_{v\,max}} \lambda_i x^{i-1} \quad \text{and} \quad \rho(x) = \sum_{i=2}^{d_{c\,max}} \rho_i x^{i-1} \ , \tag{4}$$

respectively, where d_{vmax} and d_{cmax} are the maximum variable node and the maximum check node degree of the code, respectively. λ_i and ρ_i represent the proportion of edges connected to variable and check nodes of degree i, respectively. From the node perspective, the degree distribution can be defined as

$$\tilde{\lambda}(x) = \sum_{i=2}^{d_{v\,max}} \tilde{\lambda}_i x^{i-1} \quad \text{and} \quad \tilde{\rho}(x) = \sum_{i=2}^{d_{c\,max}} \tilde{\rho}_i x^{i-1} \ , \tag{5}$$

where the coefficients represent the fractions of degree-i variable and check nodes [7]. The rate of an irregular LDPC code is defined as

$$R = 1 - \frac{\sum_{j=2}^{d_{c\,max}} \rho_j / j}{\sum_{i=2}^{d_{v\,max}} \lambda_i / i} \tag{6}$$

A. Notations

For a UEP LDPC code with a HOC, the information bits are assigned to different N_p protection classes. Let $\boldsymbol{\alpha} = [\alpha_1, \ \ldots, \ \alpha_{N_p-1}]$ represent the normalized length of each protection, i.e., α_i equals the number of bits belonging to protection class P^i divided by k. These bits are then re-multiplexed and assigned to different modulation classes N_m, defined by $\boldsymbol{\beta} = [\beta_1, \ \ldots, \beta_{N_m}]$, where β_i is the proportion of bits that belongs to modulation class i.
The variable node polynomial $\lambda(x)$, with the corresponding vector $\boldsymbol{\lambda}$ contains the overall variable node degree distribution, both for different protection classes and different modulation classes. Let $\lambda_{M^j,i}^{P^k}$ represent the fraction of edges connected to degree i variable nodes belonging to modulation class M^j and protection class P^k. We define $\boldsymbol{\lambda}_{M^j}^{P^k} = [\lambda_{M^j,2}^{P^k}, \ldots, \lambda_{M^j,d_{v_{max}}}^{P^k}]^T$ and $\boldsymbol{\lambda} = \left[\boldsymbol{\lambda}_{M^1}^{P^1\,T}, \ldots, \boldsymbol{\lambda}_{M^1}^{P^{N_p}\,T}, \ \ldots, \ \boldsymbol{\lambda}_{M^{N_m}}^{P^1\,T}, \ldots, \boldsymbol{\lambda}_{M^{N_m}}^{P^{N_p}\,T} \right]^T$, to be the vectors characterizing the sub-degree distribution, where $(\cdot)^T$ denotes transpose. The length of $\boldsymbol{\lambda}_{M^j}^{P^k}$ and $\boldsymbol{\lambda}$ column vectors are $d_{v_{max}} - 1$ and $((d_{v_{max}} - 1) \cdot N_p \cdot N_m)$ respectively and the vector $\boldsymbol{\rho} = [\rho_2, \ldots, \rho_{d_{c_{max}}}]^T$ defines the check node degree distribution.

B. Optimization of the degree distribution

LDPC codes are decoded by a message passing algorithm known as belief propagation (BP) algorithm. In BP, reliable messages are exchanged between variable and check nodes in an iterative fashion. In order to optimize the degree distribution of an irregular LDPC code, we investigate the decoding behavior based on a belief propagation algorithm. However, due to varying degrees, the densities of the mutual information at different nodes are not necessarily equal, thus, the mutual information at each node is averaged by the degree of that node. At the l^{th} iteration, the mutual information from a check node to variable node x_{cv} and from a variable node to a check node x_{vc} computed using density evolution with Gaussian approximation are given by

$$x_{cv}^{(l-1)} = 1 - \sum_{j=2}^{d_{c\,max}} \rho_j J\left(\sqrt{(j-1)J^{-1}(1 - x_{vc}^{(l-1)})}\right) \ , \tag{7}$$

$$x_{vc}^{(l)} = \sum_{j=1}^{N_m} \sum_{k=1}^{N_p} \sum_{i=2}^{d_{v_{max}}} \lambda_{M^j,i}^{P^k} J\left(\frac{2}{\sigma_j^2} + (i-1)J^{-1}(x_{cv}^{(l-1)})\right) \ , \tag{8}$$

where $J(.)$ computes the mutual information, i.e., $x = J(m)$ given as

$$J(m) = 1 - \frac{1}{\sqrt{4\pi m}} \int_{\mathbb{R}} \log_2(1 + e^{-z}) \cdot e^{-\frac{(z-m)^2}{4m}} dz \ , \tag{9}$$

where $z \sim N(m, 2m)$ is a consistent Gaussian random variable. With equations (7) and (8), the density evolution for the mutual information of the LDPC code is summarized as

$$x_{vc}^{(l)} = F(\boldsymbol{\lambda}, \boldsymbol{\rho}, \boldsymbol{\sigma}^2, x_{vc}^{(l-1)}) \ . \tag{10}$$

We optimize the variable node degree distribution given a check node degree distribution for different protection and modulation classes. Splitting the variable degree distribution into sub-degree distributions results in the sum constraint

$$\sum_{j=1}^{N_m} \sum_{k=1}^{N_p} \sum_{i=2}^{d_{v_{max}}} \lambda_{M^j,i}^{P^k} = 1 \ . \tag{11}$$

The DE algorithm is said to be converging for a code described by a variable node degree distribution λ and a check node distribution ρ, if $x_{vc}^{(l)} > x_{vc}^{(l-1)}$ for any $x_{vc}^{(l-1)}$, given as

$$x_{vc}^{(l)} \ > \ x_{vc}^{(l-1)} \quad \text{or} \quad F(\boldsymbol{\lambda}, \boldsymbol{\rho}, \boldsymbol{\sigma}^2, x_{vc}^{(l-1)}) \ > \ x_{vc}^{(l-1)} \ . \tag{12}$$

Another important constraint to be fulfilled by the ensemble LDPC code is the stability constraint which ensures convergence of the DE algorithm for mutual information close to one. The stability condition gives an upper limit for degree-2 variable nodes [7]

$$\frac{1}{\lambda'(0)\rho'(1)} > e^{-r} = \int_{\mathbb{R}} P_0(x) e^{-\frac{x}{2}} dx = e^{-\frac{1}{2\sigma^2}} \ , \tag{13}$$

with $P_0(x)$ being the message density for the received values and $\lambda'(x)$ and $\rho'(x)$ being the derivatives of the degree polynomials. For our scheme, $\lambda'(0) = \sum_{j=1}^{N_m} \sum_{k=1}^{N_p} \lambda_{M^j,2}^{P^k}$ and $\rho'(1) = \sum_{m=2}^{d_{c_{max}}} \rho_m \cdot (m-1)$. The bits in our schemes experience different channel noise with different equivalent noise variances σ_j^2, thus, we exploit the average density, given by utilizing the modulation class proportions β,

$$e^{-r} = \int_{\mathbb{R}} \sum_{j=1}^{N_m} \beta_j \cdot P_{0,j}(x) e^{-\frac{x}{2}} dx = \sum_{j=1}^{N_m} \beta_j \cdot e^{-\frac{1}{2\sigma_j^2}} \ . \tag{14}$$

C. Linear programming (LP) optimization algorithm

We optimize the variable node degree distribution by minimizing the threshold value for a fixed code rate, where the threshold is the lowest E_b/N_0 value for which the density evolution converges [7]. The optimization is performed at $E_b/N_0 = \delta + \epsilon$, where δ is the lowest possible threshold in dB for the given ρ and $d_{v_{max}}$, and ϵ is the offset from the lowest threshold that provides more flexibility in the choice of λ for the code design. The linear-programming routine for the optimization of the variable node degree distribution requires the check node degree distribution ρ, $E_b/N_0 = \delta + \epsilon$ in dB, $d_{v_{max}}$, and the code rate R. The algorithm for the variable node degree distribution is shown on the right.

D. Code Construction

The first step of the code design is the optimal degree distribution of the variable node. Once the variable node degree distribution is obtained, a parity check matrix is constructed using the approximate cycle extrinsic (ACE) message degree algorithm proposed by Tian *at al.* [6] with the aim of avoiding small cycle clusters isolated from the rest of the graph and lowering the error-floor of irregular LDPC codes. Tables I and II show the optimum variable node degree distribution for HOC-UEP 8-PSK and 64-QAM, respectively.

IV. Simulation Results

For the simulation results, we considered HOC-UEP 8-PSK and 64-QAM constellations. In 8-PSK, we have two modulation classes, i.e., $N_m = 2$, while for the 64-QAM constellation, $N_m = 3$. The variable node degree distributions are optimized based on the optimum check-node degree distribution of a rate $R = 1/2$ code with $N_p = 3$ protection classes and the proportion of bits per protection class $\alpha = [0.3,\ 0.7]$. All the parity bits are assigned to the least priority class, i.e., P^3. The maximum variable node degree was $d_{v_{max}} = 30$ and $\rho(x) = 0.00749x^7 + 0.99101x^8 + 0.00150x^9$. The codeword length is $N = 4096$, with $k = 2048$ are the information bits and the simulations were performed with 50 decoding iteration.

Optimization algorithm for HOC UEP-LDPC code.

Optimize

$$\max_{\lambda} \sum_{j=1}^{N_m} k_j \sum_{i=2}^{d_{v_{max}}} \lambda_{M^j,i}^{P^k} \tag{15}$$

subject to

[C$_1$] Rate constraint

$$\sum_{j=1}^{N_m} \sum_{k=1}^{N_p} \sum_{i=2}^{d_{v_{max}}} \frac{\lambda_{M^j,i}^{P^k}}{i} = (1-R)^{-1} \sum_{i=2}^{d_{c_{max}}} \frac{\rho_i}{i} \tag{16}$$

[C$_2$] Proportion distribution constraints

i)

$$\sum_{j=1}^{N_m} \sum_{k=1}^{N_p} \sum_{i=2}^{d_{v_{max}}} \lambda_{M^j,i}^{P^k} \mathbf{1} = 1 \tag{17}$$

ii) $\forall k \in \{1, \dots, N_p - 1\}$,

$$\sum_{j=1}^{N_m} \sum_{i=2}^{d_{v_{max}}} \frac{\lambda_{M^j,i}^{P^k}}{i} = \alpha_k \frac{R}{1-R} \sum_{i=2}^{d_{c_{max}}} \frac{\rho_i}{i} \tag{18}$$

iii) $\forall j \in \{1, \dots, N_m - 1\}$,

$$\sum_{k=1}^{N_p} \sum_{i=2}^{d_{v_{max}}} \frac{\lambda_{M^j,i}^{P^k}}{i} = \beta_j \frac{1}{1-R} \sum_{i=2}^{d_{c_{max}}} \frac{\rho_i}{i} \tag{19}$$

[C$_3$] Convergence constraint, see (12)

$$F(\boldsymbol{\lambda}, \boldsymbol{\rho}, \boldsymbol{\sigma}^2, x) > x \tag{20}$$

[C$_4$] Stability condition, see (13) and (14)

$$\sum_{j=1}^{N_m} \sum_{k=1}^{N_p} \lambda_{M^j,2}^{P^k} < \left[\sum_{j=1}^{N_m} \beta_j e^{-1/2\sigma_j^2} \cdot \sum_{m=2}^{d_{c_{max}}} \rho_m (m-1) \right]^{-1} \tag{21}$$

TABLE I: Degree distributions for the HOC-UEP LDPC codes optimized for 8-PSK with $N_p = 3$ and $N_m = 2$, $\alpha = [0.3,\ 0.7]$.

		C^1	C^2	C^3
$\epsilon = 0$ dB	M^1	$\lambda_9 = 0.1664$ $\lambda_{10} = 0.0591$ $\lambda_{30} = 0.1733$	$\lambda_3 = 0.1669$	$\lambda_2 = 0.1249$
	M^2	$\lambda_{30} = 0.0955$	$\lambda_4 = 0.0117$ $\lambda_5 = 0.0965$	$\lambda_2 = 0.0868$ $\lambda_3 = 0.0072$ $\lambda_4 = 0.0117$
$\epsilon = 0.1$ dB	M^1	$\lambda_{12} = 0.3292$ $\lambda_{30} = 0.1778$	$\lambda_3 = 0.1043$	$\lambda_2 = 0.1603$
	M^2		$\lambda_3 = 0.0589$ $\lambda_4 = 0.0575$ $\lambda_5 = 0.0453$	$\lambda_2 = 0.0529$ $\lambda_3 = 0.0138$
$\epsilon = 0.2$ dB	M^1	$\lambda_{15} = 0.4870$ $\lambda_{30} = 0.02678$	$\lambda_3 = 0.1049$	$\lambda_2 = 0.1599$
	M^2		$\lambda_3 = 0.0492$ $\lambda_4 = 0.1059$	$\lambda_2 = 0.0548$ $\lambda_3 = 0.0115$
$\epsilon = 0.3$ dB	M^1	$\lambda_{16} = 0.4973$	$\lambda_3 = 0.0272$	$\lambda_2 = 0.2163$
	M^2	$\lambda_{16} = 0.0365$	$\lambda_3 = 0.1845$ $\lambda_4 = 0.0291$	$\lambda_2 = 0.0092$

Figures 2 and 3 show performance curves of the protection classes P^1 and P^2 of HOC-UEP 8-PSK and 64-QAM, respectively, with threshold offsets ϵ = 0.1 and 0.3. The performance curves of the protection class P^3 has not been shown as it

TABLE II: Degree distributions for HOC UEP-LDPC codes optimized for 64-QAM modulation and $\alpha = [0.3,\ 0.7]$.

		C^1	C^2	C^3
$\epsilon = 0$ dB	M^1	$\lambda_3 = 0.3422$ $\lambda_{30} = 0.0213$	$\lambda_3 = 0.1057$	$\lambda_2 = 0.01941$
	M^2			$\lambda_2 = 0.1482$
	M^3	$\lambda_{30} = 0.124$	$\lambda_3 = 0.0521$ $\lambda_5 = 0.1165$ $\lambda_6 = 0.0116$	$\lambda_2 = 0.0465$ $\lambda_3 = 0.0124$
$\epsilon = 0.1$ dB	M^1	$\lambda_{13} = 0.3831$ $\lambda_{26} = 0.0413$	$\lambda_3 = 0.0997$	$\lambda_2 = 0.0198$
	M^2			$\lambda_2 = 0.1482$
	M^3	$\lambda_{27} = 0.0621$	$\lambda_3 = 0.0558$ $\lambda_5 = 0.1206$ $\lambda_6 = 0.0114$	$\lambda_2 = 0.0471$ $\lambda_3 = 0.0109$
$\epsilon = 0.2$ dB	M^1	$\lambda_{15} = 0.4486$	$\lambda_3 = 0.0729$	$\lambda_2 = 0.0399$
	M^2			$\lambda_2 = 0.1482$
	M^3	$\lambda_{15} = 0.0517$	$\lambda_3 = 0.0997$ $\lambda_5 = 0.1015$	$\lambda_2 = 0.0281$ $\lambda_3 = 0.0093$
$\epsilon = 0.3$ dB	M^1	$\lambda_{15} = 0.4887$	$\lambda_3 = 0.0094$	$\lambda_2 = 0.0769$
	M^2		$\lambda_3 = 0.0118$	$\lambda_2 = 0.1404$
	M^3	$\lambda_{15} = 0.0116$	$\lambda_3 = 0.1712$ $\lambda_6 = 0.0823$	$\lambda_2 = 0.0076$

contains parity bits only. As expected, the UEP capability, i.e., the difference in BER between P^1 and P^2, is increased with increasing the value of the threshold offset ϵ. Moreover, the figures also show that at low SNR, performances of P^1 and P^2 of HOC-UEP 8-PSK and 64-QAM are worse for an $\epsilon = 0.3$ than $\epsilon = 0.0$, but outperform the curves for $\epsilon = 0.0$ at higher SNR.

Figures 4 and 5 show the average performance curves for

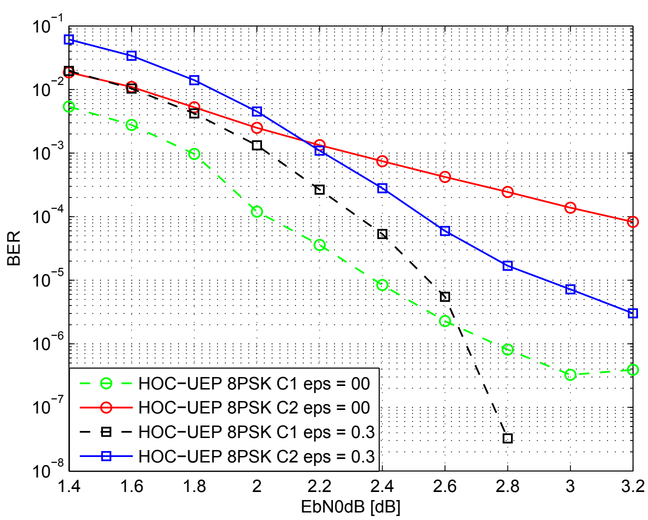

Fig. 2: BER curves of HOC-UEP 8-PSK with ϵ = 0.1 vs ϵ = 0.3

HOC-UEP 8-PSK and HOC-UEP 64-QAM with ϵ = 0.1, 0.2, and 0.3. The performances improve with increasing offset

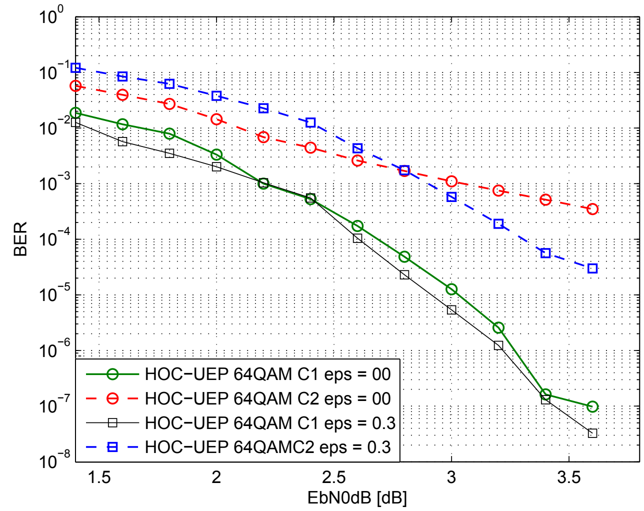

Fig. 3: BER curves of HOC-UEP 64-QAM with ϵ = 0.1 vs ϵ = 0.3

threshold values, i.e., ϵ. Moreover, as shown in figures 3 and 4, at high values of ϵ, codes perform worse at low SNR than at low ϵ values, but as the SNR increases, the code designed with high ϵ values outperforms the code for low ϵ. Figures 5 and 6 exactly depict the same property, as the average performance of the HOC-UEP 8-PSK and 64-QAM at ϵ = 0.3 is worse at low SNR and outperforms as E_b/N_0 increases.

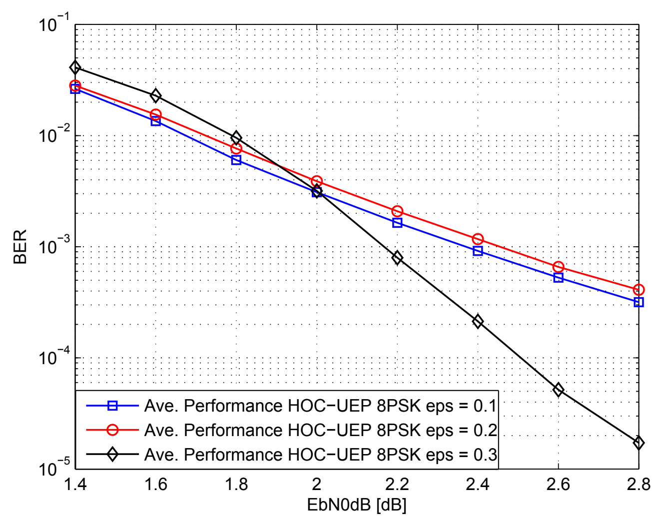

Fig. 4: Average Performance of HOC-UEP 8-PSK with ϵ = 0.1, 0.2, 0.3

A. Connectivity between protection classes

In order to have a possible explanation for the UEP capability and enhanced average BER performance of the codes when increasing ϵ, the connectivity between protection classes is investigated. It is interesting to investigate the distribution of the edges between different protection classes with an increase in the threshold value, i.e., to check if the edges are uniformly distributed between the protection classes or concentrated to

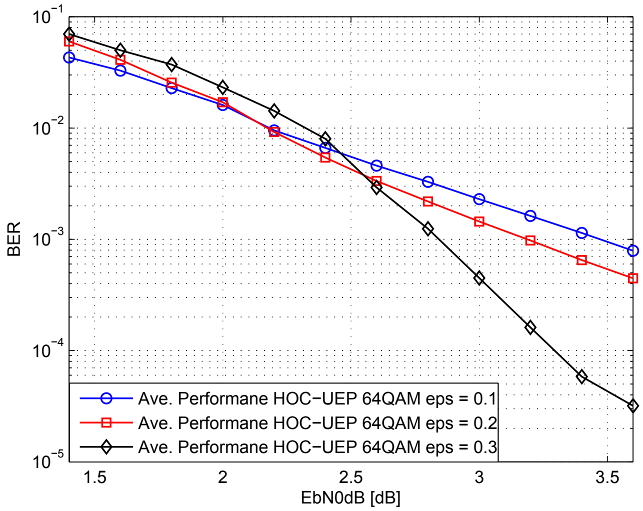

Fig. 5: Average Performance of HOC-UEP 64-QAM with ϵ = 0.1, 0.2, 0.3

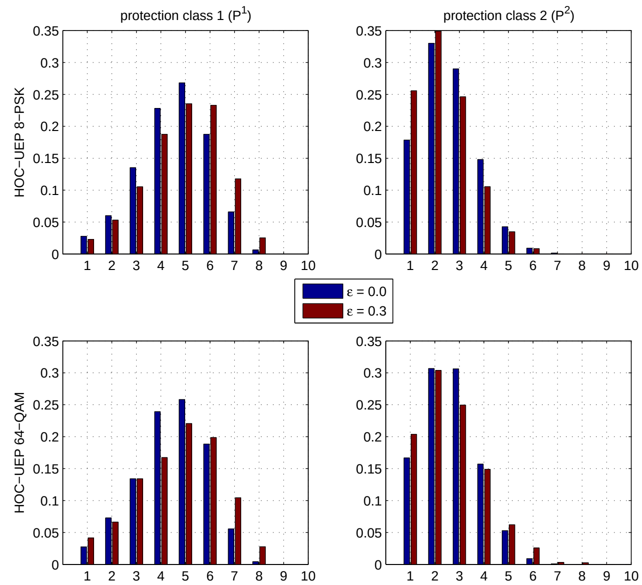

Fig. 6: Detailed check node degree distribution for HOC-UEP 8-PSK and 64-QAM constellations

a certain protection class with varying threshold offset.

In order to check the effect of the threshold offset on the connectivity of edges between different protection classes, we consider a detailed check node degree distributions for the protection classes P^j, defined as,

$$\tilde{\rho}^{(P^j)}(x) = \sum_{i=0}^{d_{c_{max}}} \tilde{\rho}_i^{(P^j)} x^{i-1} \ , \quad i = 1 \dots N_p \ . \tag{22}$$

The coefficients $\tilde{\rho}_i^{(P^j)}$ correspond to the fraction of check nodes with i edges connected to class-P^j variables nodes. By definition, we have $\sum_{i=0}^{d_{c_{max}}} \tilde{\rho}_i^{(P^j)} = 1, \ j = 1, \dots, N_p$.

If most of the check nodes have several edges connected to all protection classes, reliable and unreliable messages from different classes may proceed to other protection classes more easily and affect their performance [8]. Thus, in order to have better performance, check node connectivity in a particular class must have more edges. Figure 6 shows check node distributions per class, for HOC-UEP schemes. As the value of ϵ increases, the number of edges to check nodes with high degree in protection class P^1 increases. Thus, the enhancement of UEP capability for high values of ϵ is the high number of edges of the check nodes connected to protection class P^1.

V. Conclusion

Herein, we investigated the effect of threshold offsets on the UEP capability and average performance of UEP LDPC optimized for HOC. It has been observed through simulation results that the performance of the UEP LDPC codes can be enhanced with a slight offset in the threshold value. We explained the effect by higher degrees of the check-node distribution to the highest priority class with increasing offset.

References

[1] R. Gallager, "Low-density parity-check codes," *IEEE Trans. on Information Theory*, Vol. 8, pp. 21-28, Jan. 1962.

[2] Jilei Hou, Paul H. Siegel, Laurence B. Milstein, and Henry D. Pfister, "Capacity-approaching bandwidth-efficient coded modulation schemes based on low-density parity-check codes," *IEEE Trans. on Information Theory*, Vol. 49, pp. 2141-2155, Sept. 2003.

[3] Jorge C. Moreira and Patrick G. Farrell, *Essentials of error-control coding*, John Willey and Sons, Ltd. 2006.

[4] N. von Deetzen and S. Sandberg, "On the UEP capabilities of several LDPC construction algorithms," *IEEE Trans. on Communications*, Vol. 58, No. 11, pp. 3041-3046, Nov. 2010.

[5] X. Y. Hu, E. Eleftheriou and D. M. Arnold, "Regular and irregular progressive edge-growth Tanner graphs," *IEEE Trans. Inf. Theory,* Vol. 51, pp. 386-398, Jan. 2005.

[6] T. Tian, C. Jones D. Villasenor,and R. Wesel, "Selective avoidance of cycles in irregular LDPC code construction," *IEEE Trans. Commum,* Vol. 52, pp. 1242-1247, Aug. 2004.

[7] S. Sandberg and N. von Deetzen, "Design of bandwidth efficient unequal error protection LDPC codes" *IEEE Trans. on Communications*, Vol. 58, pp. 802-811, Mar. 2010.

[8] N. von Deetzen, *Modern Coding Schemes for Unequal Error Protection,* PhD Thesis, Jacobs University Bremen, Jan. 2009.