IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Date of publication xxxx 00, 0000, date of current version Sept 21, 2018.

Digital Object Identifier 10.1109/ACCESS.2018.DOI

Conceiving Extrinsic Information Transfer Charts for Stochastic Low-Density Parity-Check Decoders

ASUN PÉREZ-PASCUAL[1], ALEX HAMILTON[2], ROBERT G. MAUNDER[2], and LAJOS HANZO[2]

[1]Institute of Telecommunications and Multimedia Applications, Valencia Polytechnic University, Valencia 46021, Spain(e-mail: asperez@eln.upv.es)
[2]Department of Electronics and Computer Science, University of Southampton, Southampton SO17 1BJ., U.K. (e-mail: [alex.hamilton,rm,lh]@ecs.soton.ac.uk)

Corresponding author: Asun Pérez-Pascual (e-mail: asperez@eln.upv.es).

ABSTRACT Stochastic Low-Density Parity-Check Decoders (SLDPC) have found favour recently both for correcting transmission errors as well as for improving the hardware efficiency. The main drawback of these decoders is that they require hundreds of time periods to decode each frame, but their chip-area is smaller than that of their fixed point counterparts, so they can achieve higher hardware efficiency and may consume less energy. In this paper we propose a novel EXIT chart technique for characterizing the iterative decoding convergence of all the sequences involved in the Stochastic Low-Density Parity-Check Decoder. We have conceived a new model, which takes into consideration not only the sequences exchanged between the decoders, but also the sequences generated inside the variable node decoder (those which are stored in the edge memories). In this way, the model is able to predict the number of decoding iterations required for achieving iterative decoding convergence, as confirmed by own decoder simulations. The proposed technique offers new insights into the operation of SLDPCs, which will facilitate improved designs for the research community.

INDEX TERMS EXIT chart, Low-Density Partity-Check Decoder, Stochastic arithmetic.

GLOSSARY

AWGN	Additive White Gaussian Noise
BER	Bit Error Rate
BPSK	Binary Phase-Shift Keying
CN	Check Node
CND	Check Node Decoder
DCmax	Maximum number of decoding cycles
E_b/N_0	Signal to Noise Ratio per bit
EXIT	Extrinsic Information Transfer
MI	Mutual Information
LLR	Logarithmic Likelihood Ratio
RN	Random Number
LDPC	Low-Density Parity-Check
SLDPC	Stochastic Low-Density Parity-Check
SP	Sum of Products
VN	Variable Node
VND	Variable Node Decoder
5G-NR	5G new radio

I. INTRODUCTION

THE interest in Low-density parity-check block codes [1] has substantially grown in recent years due to their ability to achieve performances close to Shannon's limit. Recently, LDPC codes have been adopted by the 3GPP New Radio technical specifications [2], because their parallel implementation is cable of meeting the high-throughput and the low-latency requirement of the Fith-Generation (5G) systems. They have also been adopted in long-haul fiber optic communications systems to mitigate the physical impairments and to fully exploit the system capacity [3]. Recently, they have been used in the Physical-layer Key Reconciliation method [4] as well for enhancing communication security.

LDPC decoding operates on the basis of a serial concatenation of two decoders, namely of the variable node decoder (VND) and of the check node decoder (CND). The iterative decoding process consists of the two decoders feeding and processing soft-valued bits back and forth between each other through an interleaver and a de-interleaver, which reorder the soft-valued bits in a non-contiguous way for the sake of providing uncorrelated extrinsic information.

Typically, these decoders have been implemented using fixed point two's complement number representation. The

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2018.2872113, IEEE Access

number of quantization bits used in fixed point iterative decoders determines both their dynamic range and their hardware complexity [5]. Naturally, the higher the number of quantization bits, the higher the error correction performance of the decoder. However, having more quantification bits also increases the hardware complexity, owing to the employment of larger arithmetic circuits, larger number of interconnections and larger memory requirements. These increase the chip-area, limit the maximum throughput and increase the power disipation [6].

In order to alleviate the decoder complexity several sophisticated designs have appeared in the recent literature. One of these is based on the use of stochastic arithmetic which emerged in the 1960s [7] as a method of designing low-precision digital circuits, and has recently been used for implementing neural networks [8] and LDPC decoders [9]. The main drawback of SLDPC decoders is, however, that the number of decoding iterations required to converge is higher than that of other approaches.

Recently, researchers have started to investigate the convergence of diverse iterative decoding schemes. In [10], a novel Extrinsic Information chart (EXIT chart) is proposed for predicting the iterative convergence of the Fully Parallel Turbo Decoder. In this work, a new 3D EXIT chart is formulated for considering the mutual information of the Logarithmic Likelihood Ratios (LLRs) and of the state metrics. Inspired by this work, we have obtained a prediction of the SLDPC decoding convergence for the first time.

The new contributions of this paper are as follows:

1) *We demostrate that, in addition to the extrinsic probabilities exchanged between the VND and CND, the content of the memories included in the VND must be also considered when plotting the EXIT chart. This fact constitutes a novelty respect to the EXIT chart model used in conventional LDPC decoders.*
2) *We approach the EXIT chart analysis of the SLDPC from a new perspective which takes into consideration the inter-dependences between the extrinsic information components. This technique is capable of accurately predicting the convergence of the SLDPC to a vanisihingly low BER, as confirmed by our full-decode simulations.*

The paper is structured as follows. Section II sumarizes the LDPC algorithm. Section III introduces the Stochastic LDPC decoder, centering our attention on the most pertinent details of this contribution. The new EXIT chart conceived for the SLDPC is described in Section IV. In this section we compare the results predicted by the EXIT chart to the number of iterations obtained after simulating the stochastic full-decoder in order to quantify the accuracy attained. In Section V a comparison between the 2D projection of a Sum of Products LDPC decoder implementing using fixed point two's complement number representation and the stochastic approach is presented. Finally, our concluding remarks are offered in Section VI. For convenience the paper table of contents is included in Fig. 1.

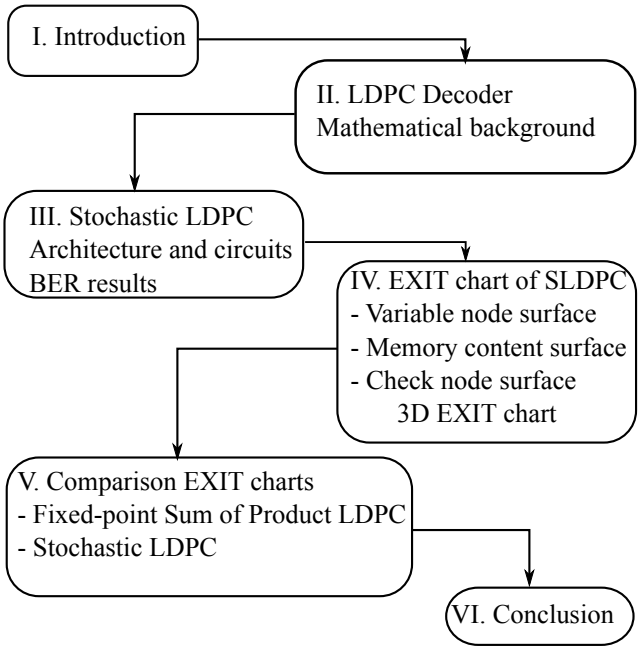


FIGURE 1: Paper outline.

II. LOW-DENSITY PARITY-CHECK DECODER

A binary LDPC code is specified by a sparse $n \times m$ parity-check matrix H, where n represents the number of parity checks and m represents the number of bits in the code block. It can be represented graphically by a bipartite Tanner graph using H as parity-check matrix, with n check nodes (CN) in one class and m variable nodes (VN) in the other. An edge exists between the variable node m and the check node n if and only if $H(n, m) = 1$.

An LDPC code is called regular when it has identical column and row weights, defining weight as the number of 1's in a column or in a row (d_v and d_c respectively). The specific choice of LDPC code construction is very important, with computer generated random LDPC codes providing the best performance, but also resulting in high complexity hardware implementations.

The LDPC codes are decoded using message-passing algorithms: at the i-th iteration, first, the variable-node decoder operation is performed to compute the variable-to-check messages q_{ij}^e, which denote an estimate of the *a posteriori* probability, and sent to the corresponding neighbour check nodes; second, in the check-node decoder operation the check-to-variable messages, r_{ij}^e, are computed and sent back to the neighbour variable-nodes. The message-passing decoding algorithm that achieves the best performance is the sum-product (SP) algorithm of [11]. However, the SP algorithm has a high implementational complexity in terms of the interconnection between the VND and the CND. This problem is particularly acute for fully parallel decoders, resulting in routing cogestion and interconnection problems.

Fig. 2 shows a simplified communication scheme, where

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2018.2872113, IEEE Access

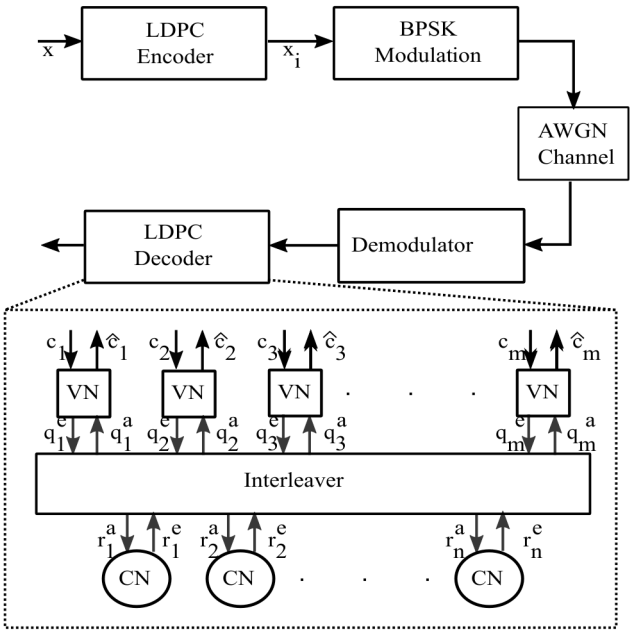


FIGURE 2: Schematic of an LDPC code, which comprises the CND and VND decoder.

the transmitter employs an LDPC encoder and the receiver employs an LDPC decoder, formed by a set of variable nodes (VN) and a set of check nodes (CN). The Tanner graph is represented by the interleaver box.

A sequence of n information bits x can be LDPC encoded into the m-bit encoded sequence x_i with the aid of a $n \times m$ generator matrix G. Here, G is designed to produce codewords that satisfy the corresponding parity-check matrix H. After this, the sequence may be BPSK modulated and it is transmitted over an Additive White Gaussian Noise (AWGN) channel. The received sequence is converted into Log-likelihood ratio (LLR) using (1), where σ^2 is the variance of the Gaussian noise channel. These LLRs values express how likely these bit values are.

$$c_i = 2 \cdot x_i / \sigma^2 \tag{1}$$

The decoder operates iteratively on the basis of the SP algorithm, which is detailed through (2) to (5). During the first iteration of the SP algorithm, the VNs initialize their extrinsic probabilities q^e_{ij} according to the demodulator inputs c_i. After this, the extrinsic probabilities q^e_{ij} are forwarded to the CNs as the *a priori* probabilities r^a_{ji}.Then the CNs employ (2) to obtain the extrinsic probabilities r^e_{ji} which pass through the de-interleaver in order to form the *a priori* VND probabilities q^a_{ij}.

$$r^e_{ji} = \frac{1}{2} - \frac{1}{2} \prod_{i' \in R_{j \setminus i}} \left(1 - 2\left(1 - r^a_{i'j}\right)\right) \tag{2}$$

These probabilities appropiately combined with the demodulator inputs c_i to obtain the extrinsic probabilities q^e_{ij}, as is detailed in (3).

$$q^e_{ij} = \frac{c_i \prod_{j' \in C_{i \setminus j}} q^a_{ij'}}{c_i \prod_{j' \in C_{i \setminus j}} q^a_{ij'} + [1 - c_i] \prod_{j' \in C_{i \setminus j}} \left(1 - q^a_{ij'}\right)} \tag{3}$$

Each iteration of the SP algorithm is finalised with the hard decision estimation of the decoded sequence, which is performed by the VNs using (4) and (5).

$$Q_i = \frac{c_i \prod_{j \in C_i} q^a_{ij}}{c_i \prod_{j \in C_i} q^a_{ij} + [1 - c_i] \prod_{j \in C_i} \left(1 - q^a_{ij}\right)} \tag{4}$$

$$\hat{c}_i = \begin{cases} 0 & Q_i < 0.5 \\ 1 & otherwise \end{cases} \tag{5}$$

This process is repeated until all parity-check equations are satisfied according to $H \cdot \hat{c}_i^{\,T} = 0$, or until the maximum affordable number of iterations has been carried out.

III. STOCHASTIC LDPC DECODER

Stochastic LDPC decoding was first proposed in [18] as a new approach for low-complexity iterative decoding of error-correcting codes on graphs. In this technique, probability messages are encoded into streams of bits using *Bernouilli* sequences in a way that the likelihood of observing a '1' in a stream is equal to the encoded probability. For instance, (6) shows some possible streams for a probability of $3/10$.

$$\begin{aligned} P &= [1010010000] \\ P &= [0011000010] \\ P &= [0100110000] \end{aligned} \tag{6}$$

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2018.2872113, IEEE Access

The main advantage of the stochastic representation is that it results in simple hardware structure for VNs and CNs, and it significantly reduces the interleaver complexity because of its bit-serial nature. These advantages are key in the case of fully-parallel LDPC decoders [6], where the number of processing nodes is high, and routing congestion is a major problem.

During the 2000s novel approaches have been proposed for improving the performance of SLDPC decoders. In [19] a SLDPC decoder that achieves comparable performance to floating point Sum of Products decoder is presented for a half rate (200, 100) LDPC code. This solution also suffers from the drawbacks that stochastic decoders present, such as the latching problem [12].

Explicitly, the latching problem is due to the presence of undesired cycles in the code's Tanner graph, which result in correlated stochastic messages because a group of nodes might retain in fixed states for several decoding iterations. This fact dramatically degrades the convergence of the decoder. In order to overcome this problem, beneficial architectures have been proposed in [9] and [20], which are based on the employment of edge memories. These memories are introduced in the variable nodes for rerandomizing and hence decorrelate the associated stochastic streams. As a benefit, edge memories significantly reduce the chance of latching.

Recently, edge memories based technique has been replaced by tracking forecast memories in [21] and [22]. This new approach has been involved for a (1056, 528) LDPC decoder used in the IEEE 802.16 (WIMAX) standard, and shows that tracking forecast memories can provide similar decoding performance to edge memories as a lower complexity.

In [23] a hardware efficient SLDPC decoder has been developed. This decoder occupies a silicon area of 3.93 mm^2 using 90-nm CMOS technology and provides maximum core throughput of 172.4 Gb/s at Eb/No of 5.5 dB, thereby its hardware efficiency is 43.86 $Gb/s/mm^2$. This is twice that the one reached by the best conventional LDPC decoder implemented in [24], whose hardware efficiency is 20.85 $Gb/s/mm^2$.

As we have mentioned, the main drawback of SLDPC decoders is the excessive number of time periods required for decoding each frame, albeit some strategies have been proposed in [25] for reducing the average number of decoding cycles.

The SLDPC decoder that we present in this paper is based on the further developing one implemented in [14]. The following subsections describe the components of our SLDPC decoder and present the implementation of these components for a (847, 363) LDPC code, which is the one used in the NATO Allied Communications Protocol 4724 [26], having check node degree of $d_c = 7$ and variable node degree of $d_v = 3$.

A. CONVERTING PROBABILITIES INTO STOCHASTIC BITS

In this approach, probabilities are represented by *Bernouilli* sequences which are generated by statistically independent processes and only one bit is processed in each clock cycle, that is referred to the decoding cycle of stochastic decoding. To convert the probabilities into the *Bernouilli* sequences we use the circuit shown in Fig. 3.

This structure consists of a comparator, which compares the demodulator's input probability (P) to a pseudo random number (R), at each decoding cycle. The probability P is fixed during the decoding of a block, while R is changing every decoding cycle. The decoder needs a dedicated comparator for each VN.

The comparator subtracts one input from another and uses the sign bit of the result as the output. This output bit is equal to 1 if $P > R$, while it is equal to 0 otherwise.

The pseudo random number is generated using a 7-bit linear feedback shift-regiter (LFSR). It is a classical digital circuit constructed of seven cascaded D-type flip-flops and a feedback loop closed through an XOR gate [15].

B. VARIABLE NODES AND EDGE MEMORIES

The VNs implemented in this decoder have degree-3 ($d_v = 3$), which means that they have three extrinsic inputs q_{ij}^a and three extrinsic outputs q_{ij}^e which obey (3). The stochastic arithmetic calculating (3) can be implemented using the circuit of Fig. 4, which was proposed in [9]. Notice that each VN contains three circuits like this, one for each output.

The behaviour of each part of the VN is described as follows: if all input bits q_{ij}^a have the same value, then this value is passed to the output port and it is written in the edge memory. We refer to this situation as the Agreed state. Otherwise, if the inputs do not agree, a random bit is selected from the edge memory, while is passed to the output port, this is called the Hold state. In this way, the latching problem is avoided by rerandomize the output of the VN. Fig. 5 presents the schematic of the edge memory, which consists of an addresable shift-register controlled by the updated input. This is activated when an Agreed state is produced.

The VN also includes a circuit that calculates the decoded bit $\hat{c}_i$. This is presented in Fig. 6. This circuit employs a 16-bit edge memory and two 2-bit shift-registers to avoid the latching problem. Notice that this bit is in stochastic format, so we have to convert it to a probability, which is carried out by passing $\hat{c}_i$ to an up/down counter. The counter is decremented in case of a 0 and incremented in case of a 1. At any given decoding cycle, the sign bit of the counter indicates the corresponding hard decision, with a 0 sign bit indicating a decoded +1 BPSK symbol, and a 1 sign bit indicating a decoded -1.

We use the decoded bits to compute the parity check equations. If the parity check equation of all the VNs is asserted in the same decoding cycle, the decoder infers that all the parity-check equations of the LDPC code have been

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2018.2872113, IEEE Access

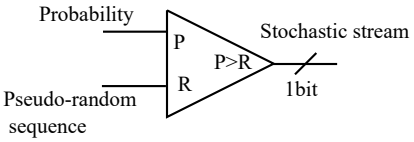


FIGURE 3: Schematic of Stochastic Converter..

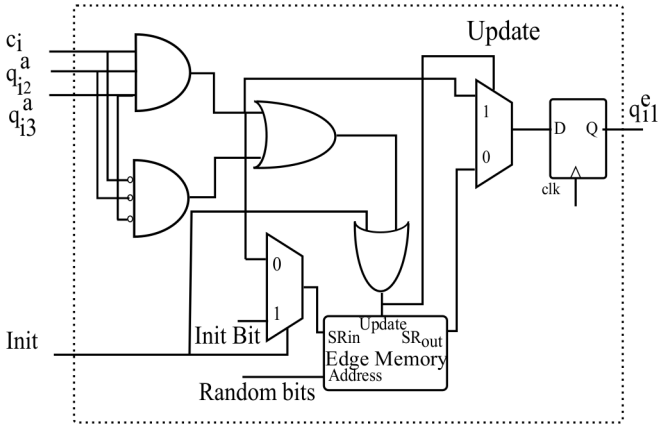


FIGURE 4: Schematic of Stochastic Variable Node having a degree of $d_v = 3$ [9]©.

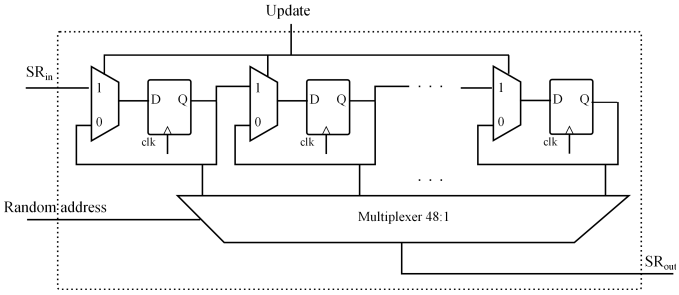


FIGURE 5: Schematic of the Edge Memory [13]©.

satisfied, whereupon the decoding process is terminated and the VNs output the decoded bits.

C. STOCHASTIC CHECK NODE

Every CN has to evaluate equation (2). Fortunately, stochastic arithmetic multiplications can be performed using XOR gates. In this way, the stochastic implementation of a CN having a degree of $d_c = 7$ employs a 7-input XOR gate for the computation of the parirty check signal. The rest of the outputs r^e_{ij} are calculated by the XOR of each input and the parirty check signal, as is shown in Fig. 7.

D. SIMULATION RESULTS

In order to characterize the operation of the SLDPC decoder we have compared it to that of the floating point SP algorithm

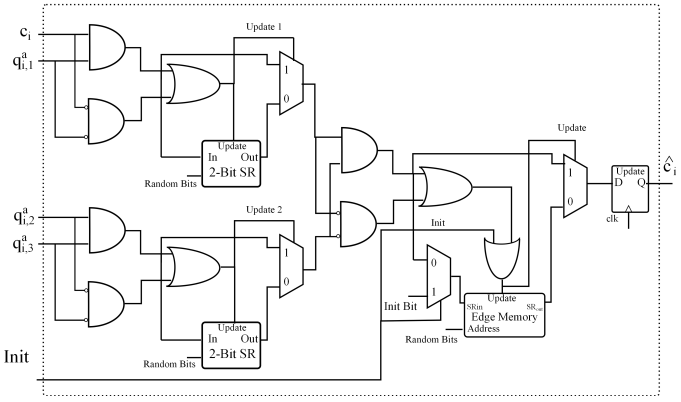


FIGURE 6: Schematic of the circuit that calculates the decoded bit $\hat{c}_i$.

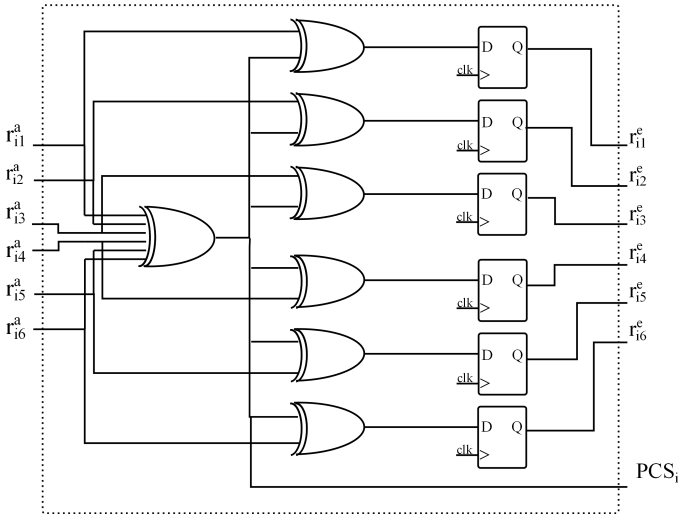


FIGURE 7: Schematic of Stochastic Check Node having a degree of $d_c = 7$.

for the same code. Fig. 8 shows the BER performance of our SLDPC decoder for a (847, 363) LDPC code with respect to floating point SP decoding (using 30 iterations).

The SLDPC decoder is programmed with a maximum decoding cycles of 500.

LDPC codes can perform close to Shanon limit in the waterflow region, however, as the signal-to-noise ratio continues to increase, LDPC codes may suffer from the error-floor phenomenon. This fact limits the use of LDPC codes in applications requiring very low error-rate. The selected code, NATO Allied Communication Protocol 4724 [26], has this error floor as a characteristic, as is observed in Fig. 8.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2018.2872113, IEEE Access

Fig. 8 demonstrates that the SLDPC decoder implemented provides comparable performance to floating point SP decoder.

IV. EXIT CHART OF THE STOCHASTIC DECODER

Extrinsic Information Transfer (EXIT) charts were proposed in [28] as an alternative method for analysing the convergence behaviour of iterative decoders. This technique has also been widely applied for LDPC decoders, but never has been used to analyse SLDPC decoders.

In [29] an in-depth study on the uses of EXIT charts has been presented. It concludes that EXIT charts constitutes a powerful tool for characterizing iteratively decoded systems. Moreover, in [30] a complete evaluation of EXIT charts for conventional LDPC decoders is presented.

According to this technique, the mutual information (MI) $I(\tilde{x};x)$ of the extrinsic probabilities can be used for characterizing the exchange of extrinsic information between the CND and the VND. This parameter quantifies the accuracy of the information in the probability sequence $\tilde{x}$ regarding the bit sequence x. The MIs values are constrained to the range of [0,1], where 0 indicates that the probability has no information concerning the corresponding bits, while 1 indicates a perfect match. Concretely, an EXIT chart function plots the MI of the corresponding extrinsic probabilities of the decoder's output versus the MI of the corresponding *a priori* input probabilities.

The mutual information is quantified by estimating the distributions of the *Bernouilli* sequences bits $p(q/x_i)$, where we have $q, x_i \epsilon \{0,1\}$. The MI may then be obtained according to

$$MI = 0.5 \cdot \sum_{x_i=0,1} \int_{-\infty}^{\infty} p_A(q|x_i) \cdot log_2 \frac{2 \cdot p_A(q|x_i)}{p_A(q|x_i=1) + p_A(q|x_i=0)} \cdot dq \quad (7)$$

where p_A is the conditional probability density function of the sequence (q).

In the case of a SLDPC, we have to consider the MI of the *a priori* and extrinsic *Bernouilli* sequences of each decoder, in addition to the mutual information of the content of the edge memories because the behaviour of the VND depends on these bits.

Fig. 9 shows the interdependence among the different MI that must be considered for drawing the EXIT chart functions of the SLDPC decoder. In this case, the output of the CND depends only on the input *a priori* probabilities (r^a), hence the EXIT chart function of this decoder (f_{CND}) can be calculated using the conventional method of [16]. However, in the VND, the output depends both on the demodulator's input probability, on the output of the edge memories and on the input *a priori* probabilities (q^a), so the function of the VND (f_{VND}) has three variables, and it can be represented by a surface for a given value of signal to noise ratio (E_b/N_0).

Additionally, the function of the edge memory (f_{EM}) depends on it's previous state and on the input *a priori* probabilities (q^a). This function should also be represented by a surface. In this way, the 3D EXIT chart contains three surfaces: the CND surface (CND_{surf}) , the VND surface (VND_{surf}) and the memory content surface $(Memo_{surf})$. (8), (9) and (10) show the behaviour of this model, which are visualized with the aid of Fig. 9

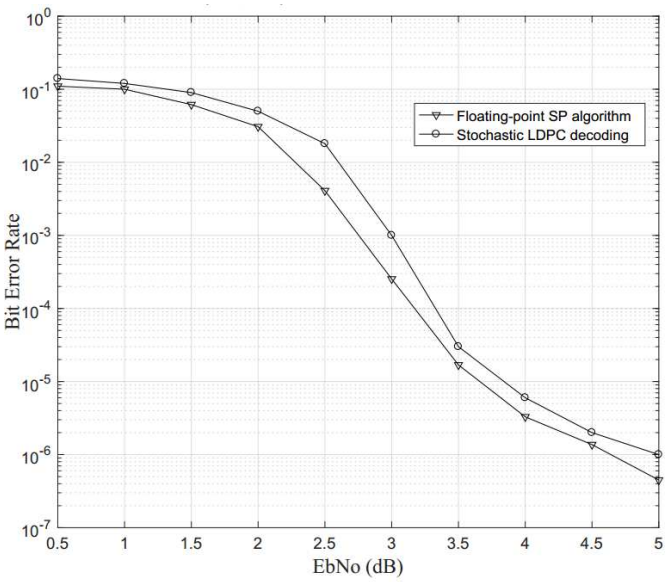


FIGURE 8: Simulation results for a (847,363) code using BPSK modulation for communication over a AWGN channel. The *edge memory* length is 49 and DCmax=500.

$$I(q^a;x_i) = f_{CND}\left[I\left(q^e;x_i\right)\right] \quad (8)$$

$$I(q^e;x_i) = f_{VND}\left[I\left(q^a;x_i\right), I\left(mem_{out};x_i\right), E_b/N_o\right] \quad (9)$$

$$I(mem_{out};x_i) = f_{EM}\left[I\left(q^a;x_i\right), I\left(mem_{in};x_i\right)\right] \quad (10)$$

The next subsections present the models used for drawing each surface of the 3D EXIT chart.

A. STOCHASTIC VARIABLE NODE AND MEMORY CONTENT SURFACES

Fig. 10 shows the schematic of the system used for plotting the VND EXIT chart function surface and the memory content surface, when using BPSK modulation for comunication over a AWGN channel. In this system, each bit of the encoded sequence x_i is repeated d_v times to form the longer sequence, which can be used for computing the *a priori* probabilities (q^a) for a given mutual information (IA_v) using the Gaussian process presented in [27]. All of these bits are converted into their stochastic representations, and are passed to the stochastic VN in a serial way, one bit each decoding cycle.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2018.2872113, IEEE Access

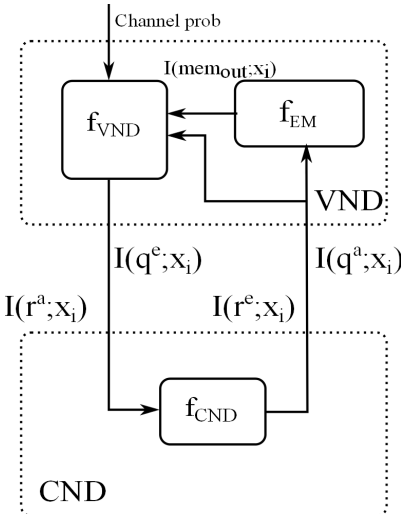


FIGURE 9: Interdependence among MI measurements.

Additionally, this longer sequence is also used for computing the memory content bits for a given mutual information (Imem) using the Gaussian non-binary process detailed in [10]. These bits are converted into their stochastic representation and are used for filling up the edge memories.

While the decoding process continues, the content of the edge memories before the update process is compared to the encoded sequence for computing the mutual information $Imem_{in}$. In the same way, the content of the edge memories after the update process is used for obtaining the mutual information $Imem_{out}$. The last one is the function that plots the memory content (Fig. 11). As we expected, this function depends linearly on the Imem, and has a weak dependence on the IA_v. This is because the *a priori* input *Bernouilli* sequences bits cannot affect more than one of the 48 bits stored in the edge memory in each decoding cycle.

The outputs of the VN are used for performing the mutual information calculation of the extrinsic *Bernouilli* sequences (IE_v). This parameter is represented by $I(q^e; x_i)$ in Fig. 9. As we have discussed previously, this parameter depends on the channel E_b/N_0 value, on the *a priori Bernouilli* sequences and on the memory content. This function represents the behaviour of the stochastic VN and it will be very useful for predicting the convergence of the decoder. Fig. 12 shows this function at $E_b/N_0 = 3.5dB$. As we can observe, this function depends strongly on the content of the memory and this dependence is stronger when the variable node degree is higher. This is because the higher the variable node degree, the lower the probability of producing an Agreed state, so the lower the dependence on the IA_v becomes.

If we simulate this system for different values of E_b/N_0 we realise that upon increasing E_b/N_0 the surface is raised, hence achieving higher values of IE_v.

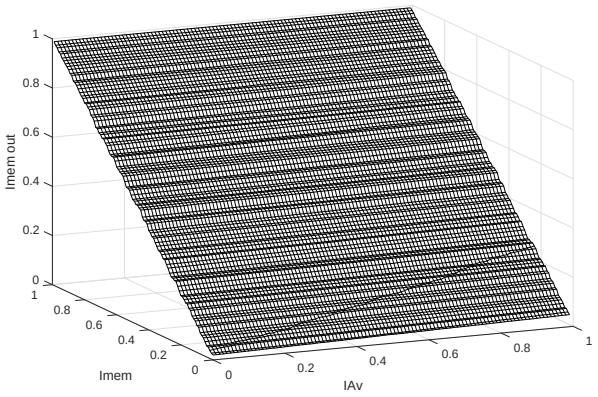


FIGURE 11: Memory content surface having $d_v = 3$ and $E_b/N_0 = 3.5dB$.

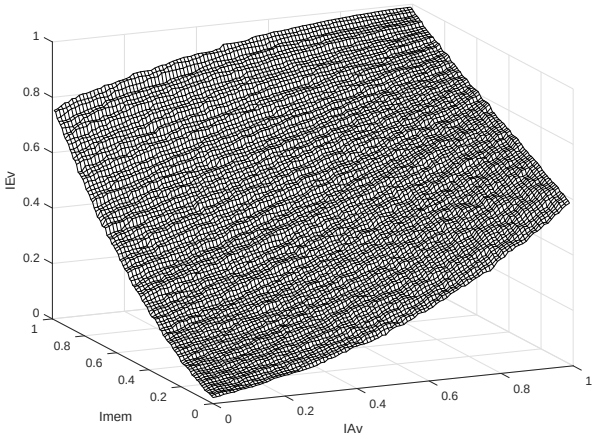


FIGURE 12: VND surface having $d_v = 3$ and $E_b/N_0 = 3.5dB$.

B. STOCHASTIC CHECK NODE SURFACE

The EXIT chart function of the CND is based on the one evaluated in [16] and it has been adapted to the case of a stochastic decoder in this treatise for the first time. It is presented in Fig. 13, where the interleaver box (π) permutes the bit sequence in order to obtain the *a priori* inputs of the CND (r^a). The simulation of this system generates Gaussian-distributed *a priori* LLRs using the method detailed in [27]. These LLRs are converted to *Bernouilli* sequences using a stochastic converter and are passed to the check node decoder. The evaluation of the mututal information relies on using the histogram based method, [27], because the MI cannot be accurately characterized for this relatively short code by the Gaussian distribution.

The CND surface represents the mututal information between the sequence and the output of the CND $IE_c = I(r; r^e)$ versus IA_c. Observe that this function is independent of the channel's E_b/N_0 value. We represent this function in

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2018.2872113, IEEE Access

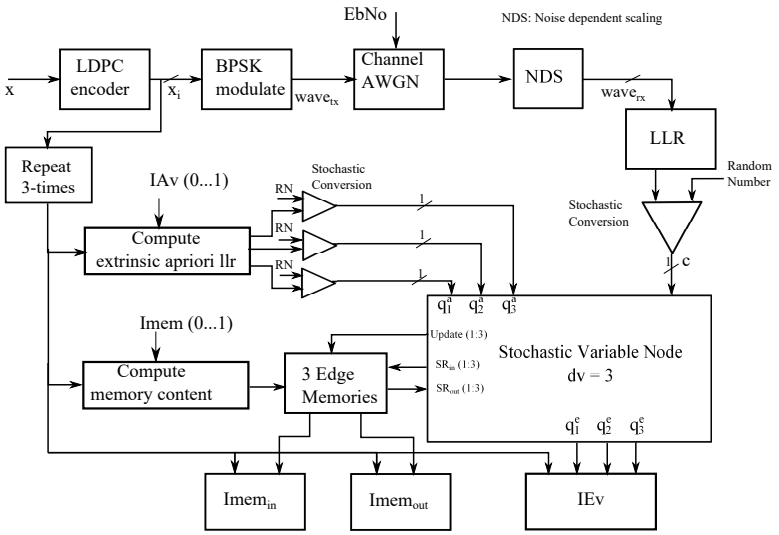


FIGURE 10: Schematic of the system used for creating the VND surface.

the EXIT chart by plotting IE_c along the abcissa axis, and IA_c along the ordinate axis.

C. 3D EXIT CHART OF THE STOCHASTIC LDPC DECODER

The VND and CND EXIT chart functions of the SLDPC decoder may be drawn together in a 3D figure (see Fig. 14) to present the 3D EXIT chart. In order to plot the CND EXIT function, we have repeated the values of the function for each Imem value, since IE_c is independent of I_{mem}.

In parallel, the memory content surface must be plotted in order to adequately model the behaviour of the stochastic decoder, as presented in Fig. 11.

1) Predicted Trajectory

Once we have combined all functions in a pair of 3D plots (the memory content surface and the 3D EXIT chart), we may predict the future evolution of the extrinsic information exchanged between the decoders, as the iterative decoding process continues. In this way, the number of decoding cycles required for attaining convergence can be calculated using the stair-case-shaped decoding trajectory, which is drawn over the 3D EXIT chart. Consequently, next we will elaborate on the procedure of plotting the predicted trajectory of the SLDPC decoder. More particularly, the algorithm of producing the predicted trajectory is as follows:

- **Step 1**: In the first decoding cycle the decoder is initialized by filling up the edge memories with the demodulator's input probabilities and initializing IA_v also with this probability. We use the memory surface of Fig. 15b to determine the MI of the output of the memory $Imem_{out}$. This is marked in Fig. 15b as point 1.
- **Step 2**: After that, we use the VND surface of Fig. 15a to determine the MI of the extrinsic VND probability $IE_v = I(q^e; x_i)$. To arrange for this, we

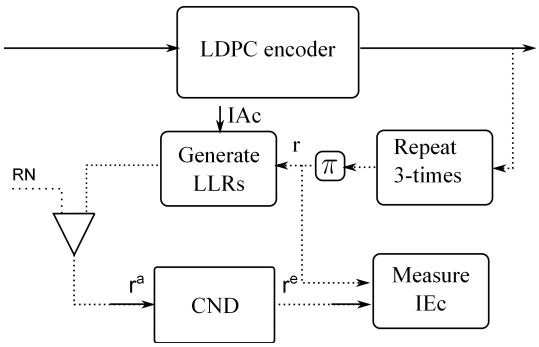


FIGURE 13: Schematic of the system used for evaluation the CND surface.

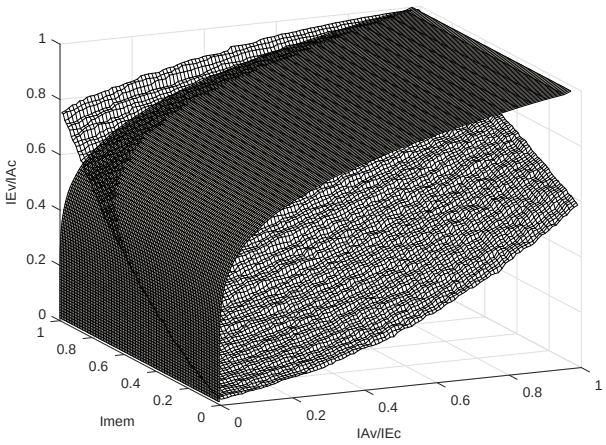


FIGURE 14: 3D EXIT chart of the SLDPC Decoder at $E_b/N_0 = 3.5dB$.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2018.2872113, IEEE Access

have considered IA_v equal to the demodulator's input probability and I_{mem} equal to $Imem_{out}$ obtained in the previous step. This may be represented by a vertical step 2 in the trajectory of Fig. 15a.

- **Step 3**: Taking into account that $IA_c = IE_v$, we start from the point (IA_c, I_{mem}) and proceed along the CND surface in order to find the value of the extrinsic MI IE_c. This has been drawn in Fig. 15a as a horizontal step 3.
- **Step 4**: The next step represents the begining of the next iteration. This step consists of obtaining the value of $Imem_{out}$ given $IA_v = IE_c$ and I_{mem} equal to $Imem_{out}$ of the last iteration. This is carried out by using the memory surface of Fig. 15b. This position is translated to Fig. 15a as represented by the vector 4.

  The iterative process continues repeating Steps 2 to 4 until the end of the intersection between the surfaces CND_{surf} and VND_{surf} is reached. The shortest the intersection curve between the CND_{surf} and VND_{surf}, the fewer the number of periods required for achieving iterative decoding convergence.

2) Iterative Decoding Convergence

In order to evaluate the reliability of the proposed method we have simulated the iterative SLDPC decoding presented in Section III for different values of E_b/N_0. Specifically, we have conducted simulations at $E_b/N_0 = 1dB$, $E_b/N_0 = 2.5dB$, $E_b/N_0 = 3.5dB$ and $E_b/N_0 = 4.5dB$. We have measured the values of IA_v, IE_v, $Imem_{in}$ and $Imem_{out}$ after each decoding iteration.

On the other hand, the 3D EXIT charts have been drawn for these values of E_b/N_0 and the predicted trajectories have been calculated. For these processes the model of the decoder has been initialized in the way that the channel probabilities are passed to the *a priori* extrinsic information q^a, and the contents of the edge memories are also updated with this probability. In this way, the convergence is expedited. Fig. 16 to 19 show the predicted trajectory and the full-decoder trajectory obtained by own simulations. The simulated trajectory is always matching closely with the predicted one. In all of the plots, the meassured trajectory fits on the CND surface and on the memory surface, and it is always very close to the VND surface. This fact allows us to confirm that the proposed model is quite accurate.

Observing these figures, we can point out that the lower the channel E_b/N_0 the longest the intersection curve between the CND_{surf} and VND_{surf}, so the most the number of periods required for achieving iterative decoding convergence, such as we expected. In the case of $E_b/N_0 = 1dB$ neither the simulated nor the predicted trajectories converge.

On the other hand, observing the memory content (Fig. 16b to 19b) we realise that the mismatch between the predicted and the measured trajectories is higher than the one observed in Fig. 16a to 19a. This mismatch is produced because we are using a simplified model of the memories. However, this mismatch does not produces a significant drawback in terms of the predicted number of iteration cycles that the full-decoder needs to converge. In table 1 we figure out the number of iteration cycles that the full-decoder needs to converge and the predicted ones. As you can see, both numbers are very close, hence it may be concluded that the proposed 3D EXIT chart result are in agreement with the measured simulation results.

EXIT charts are also useful to find the starting E_b/N_0 point for decoding convergence. In this work, further E_b/N_0 values have been simulated and, from the results obtained, we can conclude that the SLDPC decoder is able to reach convergence when E_b/N_0 is higher than 1.5 dB for the proposed code, as it can be observed in Fig. 20.

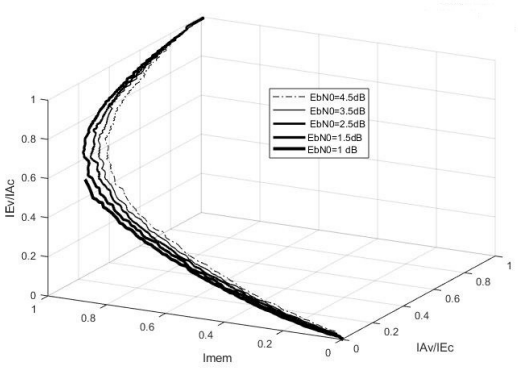


FIGURE 20: Comparison of the predicted trajectories for different values of E_b/N_0 (4.5dB, 3.5 dB, 2.5 dB, 1.5 dB and 1 dB).

3) Projection into Two Dimensions

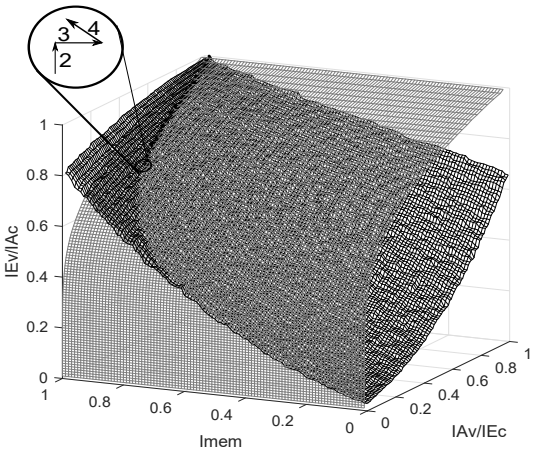
The performance of the SLDPC decoder can be readily interpreted by using the previous EXIT charts. However, this plot is not the best one for visualizing the iterative decoding convergence and for quantifying the number of iteration's number. Therefore, it is conducive to project the 3D EXIT chart into two dimensions. Thanks to this, a clearer observation of the predicted trajectory emerges. Specifically, this new plot allows us to clearly observe the MI improvement after each decoding period. This fact facilitates comparison with the SLDPC full-decoder.

This 2D EXIT chart has been obtained by observing the $IA_v - I_{mem}$ plane and the $IE_c - I_{mem}$ plane of figures 16a, 18a and 19a. The 2D projection of Fig. 16 to 19 are provided in Fig. 21 to 24. Here the circular markers represent the projected 2D version of the predicted trajectory, while the lines denote the mean of the measured MI observed after

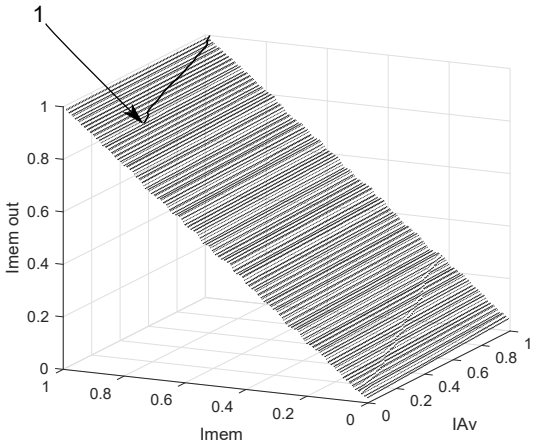
TABLE 1: Number of iterations needed to converge for the Simulated and the Predicted decoder.

E_b/N_0	Simulated	Predicted
4.5 dB	170	150
3.5 dB	370	360
2.5 dB	1500	1350

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2018.2872113, IEEE Access



(a) Predicted trajectory over the VND and CND surfaces



(b) Predicted trajectory over the memory surface

FIGURE 15: Predicted trajectories over the VND and CND surfaces and over the memory surface for $E_b/N_0 = 4.5dB$.

each decoding cycle of a decoding iteration having $N = 847$ bit-coded frames.

This 2D EXIT chart is also useful to count the number of predicted decoding cycles. Every two marks of the graph represent a new decoding cycle, so the total number of decoding cycles is half of the mark number.

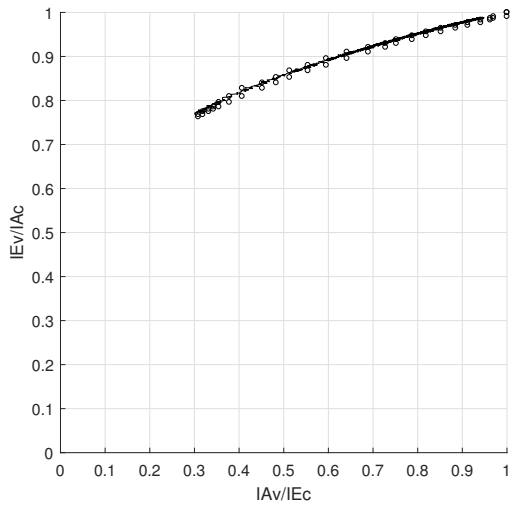


FIGURE 21: 2D projection of the predicted trajectory and full-decoder trajectory ($E_b/N_0 = 4.5dB$).

The achievable accuracy may be confirmed by comparing the predicted trajectory to the full-decoder trajectory using the 2D EXIT chart of Fig. 21 to 24. Although there is some mismatch, both trajectories follow the same direction and reach convergence at a similar number of iterations. The mismatch is particularly pronounced at $E_b/N_0 = 1dB$. This mismatch may be explained by the random behaviour of the memories, which affect the behaviour of the VND surface. This behaviour significantly increases the complexity of the model. Our future work will be focus on analysing and modeling this behaviour.

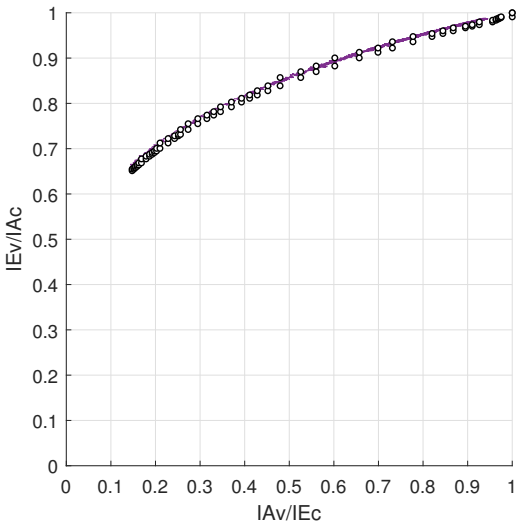


FIGURE 22: 2D projection of the Predicted trajectory and full-decoder trajectory ($E_b/N_0 = 3.5dB$).

With the aim of highlighting the benefit that this proposal provides, we include a computational complexity analysis in terms of number of frames that must be simulated in order to get smooth EXIT charts versus the number of frames for the BER plot. EXIT charts have been implemented by simulating 5000 frames and the surfaces obtained has been interpolated in order to get a fine grid. On the contrary, to perform a useful BER plot we need to simulate almost 100,000 frames for each E_b/N_0 value, which is a long and time consuming process.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2018.2872113, IEEE Access

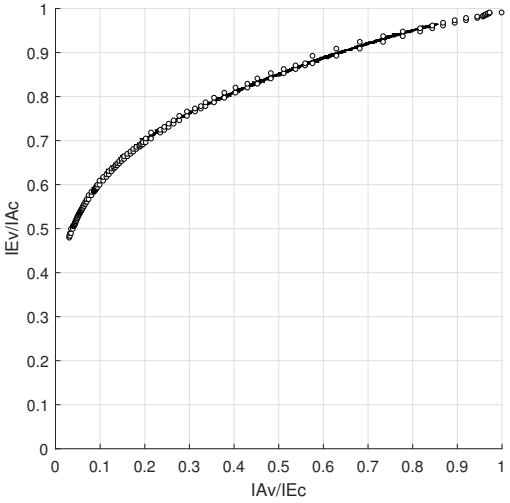


FIGURE 23: 2D projection of the predicted trajectory and full-decoder trajectory ($E_b/N_0 = 2.5dB$).

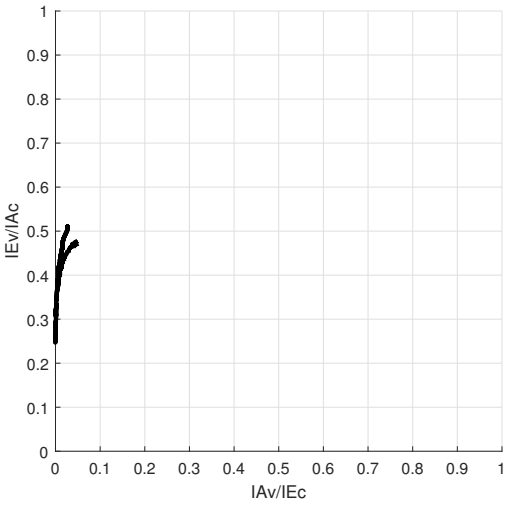


FIGURE 24: 2D projection of the predicted trajectory and full-decoder trajectory ($E_b/N_0 = 1dB$).

V. COMPARISON OF SUM-PRODUCT LDPC AND STOCHASTIC LDPC USING EXIT CHARTS

The novel approach presented in this paper allow us to characterize the performance of a decoder without doing long simulations.

The novelty of this approach lies in the application of decoders based on Stochastic arithmetic. EXIT charts have indeed been used for characterizing other LDPC decoders, but they never have been applied to decoders relying on Stochastic arithmetics.

In this section we are going to compare the EXIT charts of the Sum-Product LDPC decoder (SP-LDPC) using fixed point and the SLDPC decoder presented in this paper, using the NATO Allied Communication Protocol 4724 code [26].

Fig. 25 and 26 present the 2D EXIT chart of the SP-LDPC (dotted line) and the 2D projection of the SLDPC at $E_b/N_0 = 4.5dB$ and $E_b/N_0 = 1dB$. In these figures, each pair of cross markers corresponds to one of the decoding iteration of the SP-LDPC decoder, and each pair of circle markers corresponds to one of the decoding iteration of the SLDPC decoder. It may be observe from Fig. 25 that the SP-LDPC requires 15 iterations to achieve decoding convergence at $E_b/N_0 = 4.5dB$, meanwhile the SLDPC needs 150 iterations. On the other way, when the EXIT charts tunnels of both decoders are closed, such as happens at $E_b/N_0 = 1dB$, the convergence does not occur. This is confirmed by the high BER for both decoders presented in Fig. 8.

VI. CONCLUSIONS

We presented an innovative EXIT chart, which is useful for characterizing the iterative convergence of the SLDPC decoder. Our new approach takes into consideration all the MIs of both the LLRs and the content of the edge memories for the first time. The 2D projection of the EXIT chart allows us to clearly visualize the decoder trajectory. This plot is also useful for comparing the predicted and simulated results.

It was shown that the diference between the number of iterations predicted by the EXIT chart and that obtained after simulation are between the 3% for $E_b/N_0 = 3.5dB$ and 10% for the $Eb/N_0 = 2.5dB$.

The method proposed in this paper is applicable to other code lengths. In [30] the impact of the code length on the EXIT chart functions of a LDPC decoder is studied concluding that the EXIT functions of the CND and VND of different codes are nearly identical when the channel E_b/N_0 is 3dB. In the case of a Stochastic LDPC, the code length is not a drawback provided that the channel E_b/N_0 is higher than 1dB.

The proposed techniques offer new insights into the operation of the Stochastic LDPC decoder, which will enable improved designs in the future, in the same way that conventional EXIT charts have enhanced the design and understanding of conventional LDPC decoders. On the other hand, the idea introduced in this work could be used as inspiration for modelizing other systems formed by concatenated structures within memories.

ACKNOWLEDGMENT

This work was supported by Generalitat Valenciana under Grant number 065, of the program BEST/2016 and funded by the Spanish *Ministerio de Economia y Competitividad* and FEDER under the grant TEC2015-70858-C2-2-R.

Professor L. Hanzo would like to give thanks the ERC for the finantial support of this Advanced Fellow Grant. The financial support of the EPSRC projects EP/Noo4558/1, EP/PO34284/1 as well as of the European Research Council's Advanced Fellow Grant QuantCom is gratefully acknowledged.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2018.2872113, IEEE Access

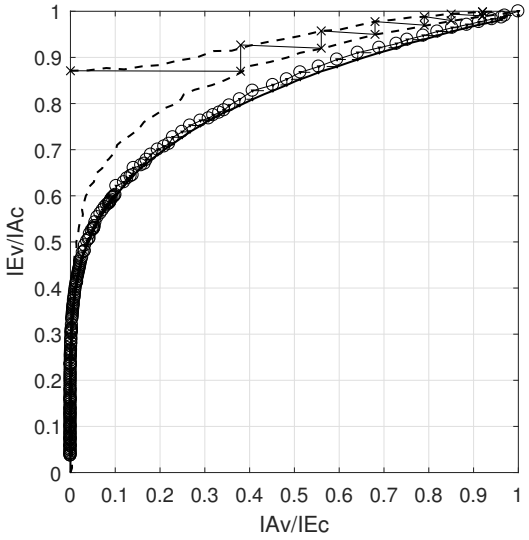


FIGURE 25: EXIT Chart comparison between the SP and the Stochastic LDPC, having a coding rate of $3/4$ and a frame length of 847 bits is employed along with BPSK modulation for communication over AWGN channel ($E_b/N_0 = 4.5dB$).

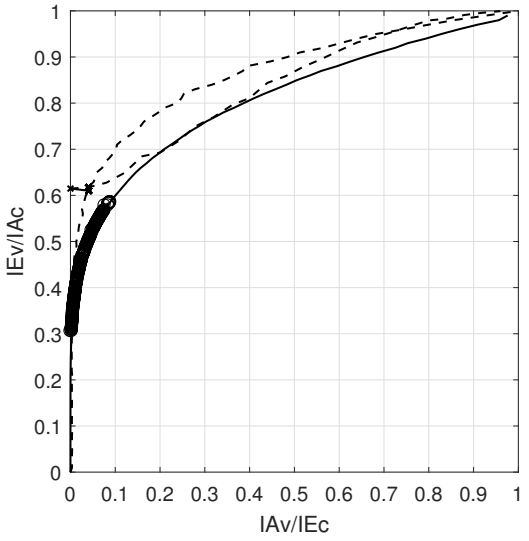


FIGURE 26: EXIT Chart comparison between the SP and the Stochastic LDPC, having a coding rate of $3/4$ and a frame length of 847 bits is employed along with BPSK modulation for communication over AWGN channel ($E_b/N_0 = 1dB$).

REFERENCES

[1] R.G. Gallager, "Low-Density Parity-Check codes", in *MIT Press*, 1963.

[2] 3GPP TS 38.212, "NR; Multiplexing and channel coding", V0.0.0, May 2017.

[3] H. Song, M. Brandt-Pearce, T. Xie and S. G. Wilson, "Combined Constrained Code and LDPC Code for Long-Haul Fiber-Optic Communication Systems", in *2012 IEEE Global Communications Conference (GLOBECOM)*, pp. 2984 - 2989, 2012.

[4] Y. Lu, Z. Zheng and X. Hou, "Modified decoding algorithm of LDPC codes for physical-layer key reconciliation", in *2017 4th International Conference on Systems and Informatics (ICSAI)*, pp. 996 - 1000, 2017.

[5] W. Yufei and B. Woerner, "The influence of quantization and fixed point arithmetic upon the BER performance of turbo codes", in *49th IEEE Vehicular Technology Conf. (VTC-1999)*, vol. 2, Houston, TX, pp. 1683âĂŞ1687 vol.2, Jul. 1999.

[6] A. Blanksby and C. Howland, "A 690-mW 1-Gb/s 1024-b rate-1/2 low-density parity-check code decoder", in *IEEE J. Solid-State Circuits*, vol. 37, no. 3, pp.404-412, Mar. 2002.

[7] B.R. Gaines, "Stochastic computing systems", in *Advances in information systems science*, vol. 2, no. 2, pp. 37-172, 1969.

[8] J. Misra, I. Saha, "Artificial neural networks in hardware: A survey of two decades of progress", in *Neurocomputing*, vol. 74, no. 1, pp. 239-255, 2010.

[9] S. Tehrani, S. Mannor and W. J. Gross, "Fully Parallel Stochastic LDPC Decoders", in *in IEEE Transactions on Signal Processing*, vol. 56, no. 11, pp. 5692-5703, Nov. 2008.

[10] H. A. Ngo, R. G. Maunder and L. Hanzo, "Extrinsic Information Transfer Charts for Characterizing the Iterative Decoding Convergence of Fully Parallel Turbo Decoders", in *IEEE Access*, vol. 3, pp. 2100-2110, 2015.

[11] D.J.C.. MacKay, "Good error-correcting codes based on very sparse matrices", in *IEEE Trans. Inf. Theory*, vol. 45, pp.399-431, 1999

[12] C. Winstead, A. Rapley, V.C. Gaudet, and C. Schlegel, "Stochastic Iterative Decoders", in *Proceedings on International Symposium on Information Theory*, pp. 1116-1120, 2005.

[13] I. Perez-Andrade, X. Zuo, R. G. Maunder, B. M. Al-Hashimi, and L. Hanzo, "Analysis of voltage- and clock-scaling-induced timing errors in stochastic LDPC decoders", in *IEEE Wireless Communications and Networking Conf. (WCNC-2013)*, pp. 4293-4298, April, 2013.

[14] I. Perez-Andrade, S. Zhong, R. G. Maunder, B. M. Al-Hashimi, and L. Hanzo, "Stochastic computing improves the timing-error tolerance and latency of turbo de- coders: Design guidelines and trade-offs", in *IEEE Access*, vol. 4, pp. 1008-1038, February, 2016.

[15] A. Dinu, M. N. Cirstea, and M. McCormick, "Stochastic Implementation of Motor Controllers", *IEEE* International Symposium on Industrial Electronics,p.p. 639-644, 2002.

[16] X. Zuo, R. G. Maunder, and L. Hanzo, "Design of Fixed-Point Processing Based LDPC Codes Using EXIT Charts", in *IEEE Vehicular Technology Conference (VTC Fall)*, pp. 1-5, 5-8 Sept. 2011.

[17] J. Hagenauer, E. Offer, and L. Papke, "Iterative decoding of binary block and convolutional codes", in *IEEE Trans. Inf. Theory*, vol. 42, no. 2, pp. 429-445, March 2006.

[18] V.C. Gaudet, and A.C. Rapley, "Iterative decoding using stochastic computation", in *Electronics Letters*, vol. 39, no. 3, pp. 299-301, 2003.

[19] S.S. Tehrani, J.W. Gross, and S. Mannor, "Stochastic decoding of LDPC codes", in *IEEE Communications Letters*, vol.10, no. 10, pp. 716-718, 2006.

[20] S.S. Tehrani, S. Manno, and W.J. Gross, "An area-efficient FPGA-based architecture for fully-parallel stochastic LDPC decoding", in *IEEE Workshop in Signal Processing Systems*, pp. 255-260, 2007.

[21] S. S. Tehrani, A. Naderi, G. A. Kamendje, S. Mannor and W. J. Gross, "Tracking Forecast Memories in stochastic decoders", in *2009 IEEE International Conference on Acoustics, Speech and Signal Processing*, pp. 561-564, Taipei, 2009.

[22] S. S. Tehrani, A. Naderi, G. A. Kamendje, S. Hemati, S. Mannor and W. J. Gross, "Majority-Based Tracking Forecast Memories for Stochastic LDPC Decoding", in *IEEE Transactions on Signal Processing*, vol. 58, no. 9, pp. 4883-4896, 2010.

[23] A. Naderi, S. Mannor, M. Sawan and W. J. Gross, "Delayed Stochastic Decoding of LDPC Codes," in *IEEE Transactions on Signal Processing*, vol. 59, no. 11, pp. 5617-5626, Nov. 2011.

[24] T. Mohsenin, D. N. Truong and B. M. Baas, "A Low-Complexity Message-Passing Algorithm for Reduced Routing Congestion in LDPC Decoders," in *IEEE Transactions on Circuits and Systems I: Regular Papers*, vol. 57, no. 5, pp. 1048-1061, May 2010.

[25] D. Wu, Y. Chen, Q. Zhang, Y. Ueng and X. Zeng, "Strategies for Reducing Decoding Cycles in Stochastic LDPC Decoders", in *IEEE Transactions on Circuits and Systems II: Express Briefs*, vol. 63, no 9, pp. 873 - 877, 2016.

[26] NATO Allied Communications Protocol 4724, VLF / LF MSK MULTI CHANNEL BROADCAST, NATO Standardisation Office

[27] S. ten Brink, "Convergence behavior of iteratively decoded parallel concatenated codes", *IEEE Transactions on Communications*, vol. 49, no. 10, pp. 1727-1737, 2001.

[28] S. ten Brink, "Designing iterative decoding schemes with the extrinsic information transfer chart", *AEU Internations Journal on Electronics and Communications*, vol. 54, pp. 389-398, 2000.
[29] M. El-Hajjar and L. Hanzo, "EXIT Charts for System Design and Analysis", *IEEE Communications Surveys and Tutorials*, vol. 16, no 1, pp. 127-153, 2014.
[30] X. Zuo, "Fully Parallel Implementation of Timing-Error-Tolerant LDPC Decoders," M.S. thesis, Dept. Electronics and Computer Science, Univ. Southampton, City of Southampton, U.K., 2016.

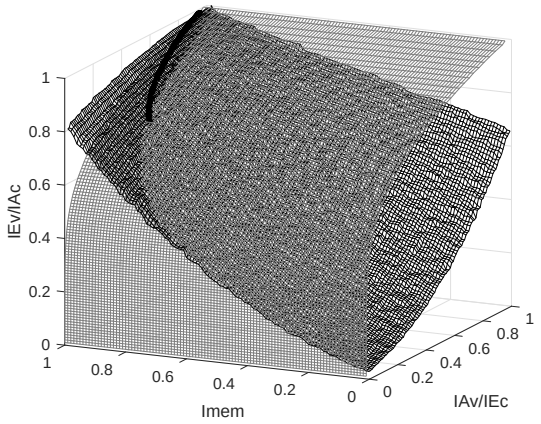
ASUN PÉREZ-PASCUAL received her telecommunication engineering and Ph.D. (telecommunication engineering) degrees from the Valencia Polytechnic University, Spain, in 1997 and 2002, respectively. She has been an Associate Professor in the Department of Electronics at Valencia Polytechnic University since 2002. Her current research interests include the design of FPGA-based systems, computer arithmetic, VLSI signal processing, and digital communications. For further information on her research, please refer to www.upv.es/ficha-personal/asperez .

ALEX HAMILTON Alex Hamilton is a Principal Communications Scientist with the UK Defence Science and Technology Laboratory. He received the BEng degree in Electronic Engineering from the University of Southampton in 2011 and the MSc degree in Communications Systems Engineering from the University of Portsmouth in 2015. He is currently undertaking Ph.D studies at the University of Southampton. He is the UK representative on several NATO panels and research working groups in the area of wireless communications. At the time of writing Alex is the secretary of the UK IEEE Information Theory chapter and is on the technical panel of several IEEE military communications conferences. He is also a Chartered Engineer of the Institute of Engineering and Technology. His research interests include joint source/channel coding, iterative decoding, underwater communications, non-linear channels and modulation techniques.

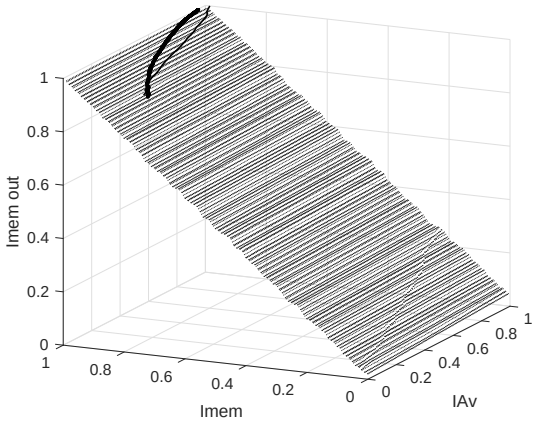
ROBERT G. MAUNDER has been with the Department of Electronics and Computer Science, University of Southampton, U.K., since 2000. He was awarded a first class honors B.Eng. in elec- tronic engineering in 2003, and the Ph.D. in wire- less communications in 2007. He became a Lecturer in 2007 and Full Professor in 2017. His research interests include joint source/channel coding, iterative decoding, irregular coding, and modulation techniques. For further information on this research, please refer to users.ecs.soton.ac.uk/rm.

LAJOS HANZO (http://www-mobile.ecs.soton.ac.uk) FREng, FIEEE, FIET, Fellow of EURASIP, DSc received his degree in electronics in 1976 and his doctorate in 1983. In 2009 he was awarded an honorary doctorate by the Technical University of Budapest and in 2015 by the University of Edinburgh. In 2016 he was admitted to the Hungarian Academy of Science. During his 40-year career in telecommunications he has held various research and academic posts in Hungary, Germany and the UK. Since 1986 he has been with the School of Electronics and Computer Science, University of Southampton, UK, where he holds the chair in telecommunications. He has successfully supervised 112 PhD students, co-authored 18 John Wiley/IEEE Press books on mobile radio communications totalling in excess of 10 000 pages, published 1782 research contributions at IEEE Xplore, acted both as TPC and General Chair of IEEE conferences, presented keynote lectures and has been awarded a number of distinctions. Currently he is directing a 60-strong academic research team, working on a range of research projects in the field of wireless multimedia communications sponsored by industry, the Engineering and Physical Sciences Research Council (EPSRC) UK, the European Research Council's Advanced Fellow Grant and the Royal Society's Wolfson Research Merit Award. He is an enthusiastic supporter of industrial and academic liaison and he offers a range of industrial courses. He is also a Governor of the IEEE ComSoc and VTS. During 2008 - 2012 he was the Editor-in-Chief of the IEEE Press and a Chaired Professor also at Tsinghua University, Beijing. For further information on research in progress and associated publications please refer to http://www-mobile.ecs.soton.ac.uk

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2018.2872113, IEEE Access

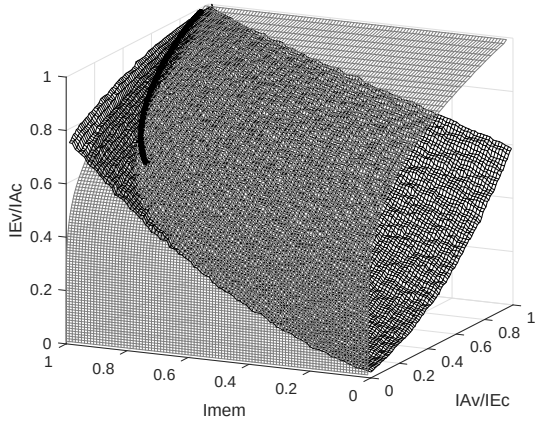


(a) Predicted (thin line) and simulated trajectories over the VND and CND surfaces

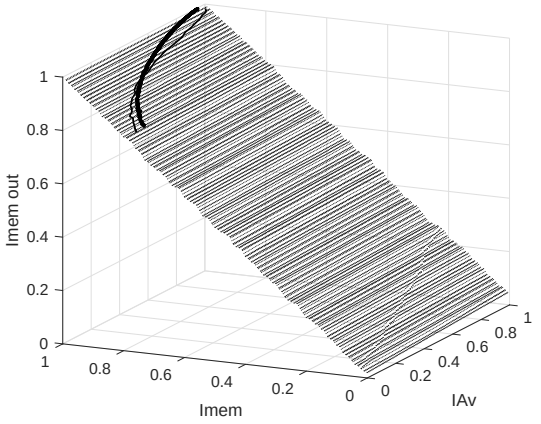


(b) Predicted (thin line) and simulated trajectories over the memory surface

FIGURE 16: Predicted (thin line) and Simulated trajectories over the VND and CND surfaces and over the memory surface $E_b/N_0 = 4.5dB$.



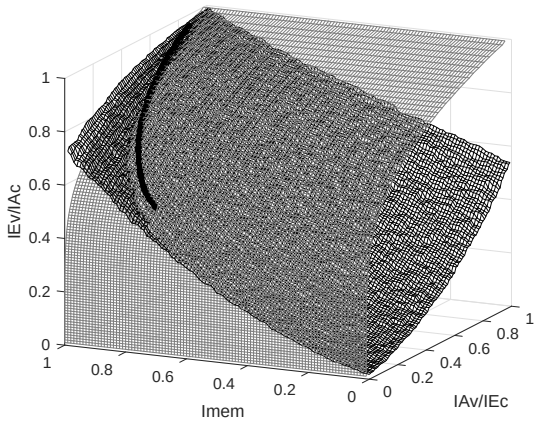
(a) Predicted and simulated trajectories over the VND and CND surfaces.



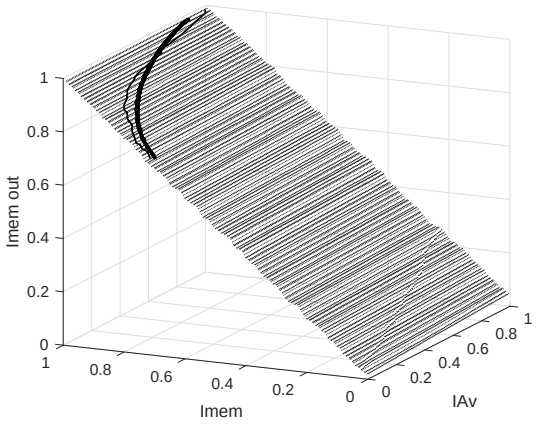
(b) Predicted and simulated trajectories over the memory surface.

FIGURE 17: Predicted and Simulated trajectories over the VND and CND surfaces and over the memory surface $E_b/N_0 = 3.5dB$.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2018.2872113, IEEE Access

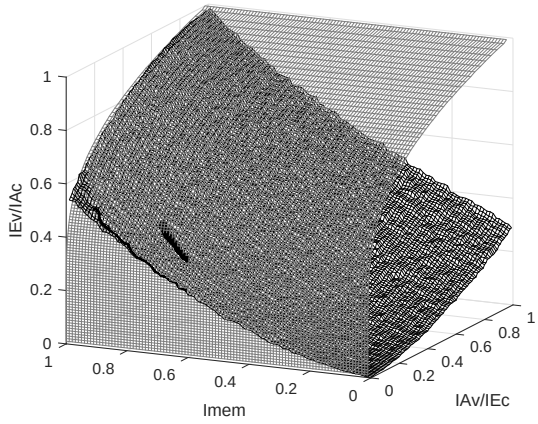


(a) Predicted (thin line) and simulated trajectories over the VND and CND surfaces.

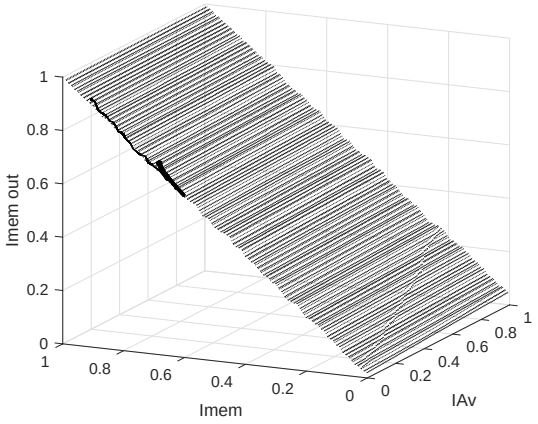


(b) Predicted (thin line) and simulated trajectories over the memory surface.

FIGURE 18: Predicted (thin line) and Simulated trajectories over the VND and CND surfaces and over the memory surface $E_b/N_0 = 2.5dB$.



(a) Predicted (thin line) and simulated trajectories over the VND and CND surfaces.



(b) Predicted (thin line) and simulated trajectories over the memory surface.

FIGURE 19: Predicted (thin line) and Simulated trajectories over the VND and CND surfaces and over the memory surface $E_b/N_0 = 1dB$.