Noise-Against-Noise Decoders
Low Precision Iterative Decoders

Franklin Cochachin*†

Supervised by:
Emmanuel Boutillon*, David Declercq†, and Lounis Kessal†

*Lab-STICC, UMR 6285, Université de Bretagne Sud
†ETIS, UMR 8051, ENSEA/University of Cergy-Pontoise

NAND Project (ANR-15-CE25-0006-01)

May 2, 2019

Problems

Error correcting codes

Error correcting codes are everywhere: in mobile phones, in satellite links, etc. They are used to detect and correct transmission errors.

Problems: future applications will require

- High throughput (from Gbit/s to Tbit/s).
- Very good decoding performance.
- Low area print.
- Low energy consumption.

Example

For the IEEE 802.3 LDPC code (10 GbitEthernet) with 28 nm technology: the new low precision decoder achieves 181.44 Gbit/s per mm^2.

Outline

Objective

A priori of the study

- Low-density Parity-check (LDPC) codes.
- Low precision decoders with 3 or 4 inputs bits.
- The injection of some randomness in the decoding process.

Objective

- Understand/Optimize the effect of noise on low precision decoders.
- Measure/Compare the performance and area of the new proposed decoders.

Outline

Low-density Parity-check (LDPC) codes

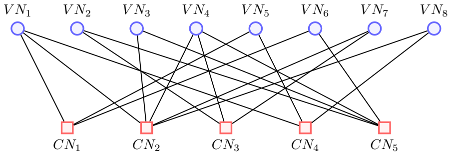
LDPC codes are used for communications standards (DVB-S2, ETHERNET code, WIMAX code, etc.) and storage applications.

- Regular LDPC codes: all VNs (resp. CNs) have the same degree d_v (resp. d_c).
- Irregular LDPC codes: the nodes have different connection degrees: $\lambda(x) = \sum_{i=2}^{d_{v,max}} \lambda_i x^{i-1}$, and $\rho(x) = \sum_{j=2}^{d_{c,max}} \rho_j x^{j-1}$.

- Matrix Representation

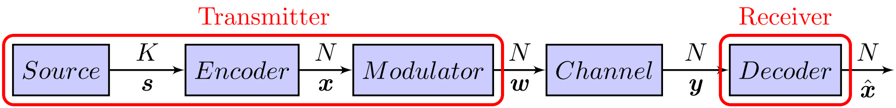
$$H_{5\times 8} = \begin{bmatrix} 1 & 0 & 0 & 0 & 1 & 1 & 0 & 0 \\ 1 & 0 & 1 & 1 & 0 & 0 & 1 & 1 \\ 0 & 1 & 0 & 1 & 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 & 1 & 0 & 0 & 1 \\ 0 & 1 & 1 & 1 & 0 & 1 & 0 & 0 \end{bmatrix}$$

- Graphical Representation: Tanner graph



Binary LDPC Decoders

A simple communication system shows where the decoder is located.



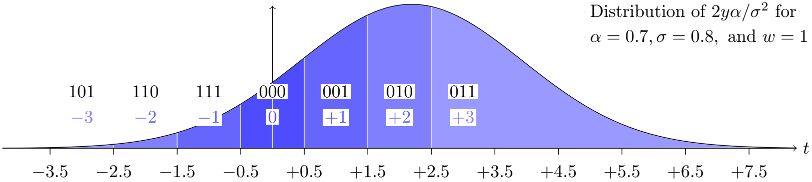
For binary LDPC decoders we have $\boldsymbol{y} \in \mathbb{R}^N$, $\boldsymbol{s} \in \{0,1\}^K$, $\boldsymbol{x} \in \{0,1\}^N$, $\boldsymbol{w} \in \{-1,1\}^N$ (BPSK modulation), and $\hat{\boldsymbol{x}} \in \{0,1\}^N$.

Message Passing decoders

- Soft-decision MP decoders : Belief Propagation (BP) decoder, Min-Sum (MS) decoder, Offset-Min-Sum (OMS) decoder, etc.
- Hard-decision MP decoders : Gallager-B decoder, Gallager-A decoder, etc.

Classical OMS decoder: quantization of channel LLR

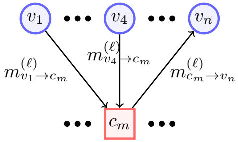
The channel output in a LLR form: $LLR(y) = 2y/\sigma^2$ for the BI-AWGN. The quantized version of the intrinsic LLR: $I = \mathcal{Q}(LLR(y))$, with a quantification rule $\mathcal{Q}$ from $\mathbb{R}$ to $\mathcal{A}_C = \{-3, -2, -1, 0, +1, +2, +3\}$. defined as $\mathcal{Q}(a) = \mathcal{S}(\lfloor \alpha \times a + 0.5 \rfloor, N_q)$, where α is *the channel gain factor*, $N_q = 2^{q-1} - 1 = 3$ for $q = 3$ bits, and $\mathcal{S}$ is the saturation function toward [-3, 3].



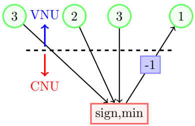
- Among the 8 levels of a 3 bits representation, only 7 are used.

Let us denote $\ell \in \mathbb{N}$ the iteration number.

- **Update rule at a CNU**



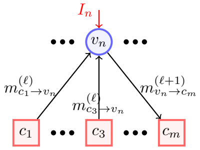
$$m^{(\ell)}_{c_m \to v_n} = \left(\prod_{v \in \mathcal{V}(c_m) \setminus \{v_n\}} \text{sign} \left(m^{(\ell)}_{v \to c_m} \right) \right) \times \max \left\{ \min_{v \in \mathcal{V}(c_m) \setminus \{v_n\}} \left| m^{(\ell)}_{v \to c_m} \right| - 1, 0 \right\}.$$



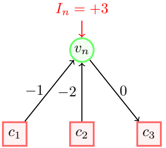
- Note that for low precision $q = 3$, the offset applied in CNUs only gives us the possibility to use 5 values ($m^{(\ell)}_{c_m \to v_n} \in \{-2, -1, 0, +1, +2\}$) instead of the 7 values of $\mathcal{A}_C$.

- **Update rule at a VNU**

$$m_{v_n \to c_m}^{(\ell+1),U} = I_n + \sum_{c \in \mathcal{V}(v_n) \backslash \{c_m\}} m_{c \to v_n}^{(\ell)}.$$



exponent U indicates unsaturated message.



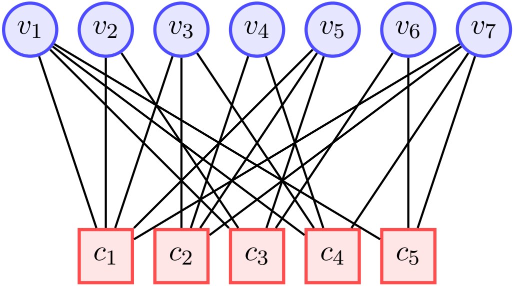
$$m_{v_n \to c_m}^{(\ell+1)} = \mathcal{S}\left(m_{v_n \to c_m}^{(\ell+1),U}, N_q\right).$$

- For $q = 3$ bits of precision: $m_{v_n \to c_m}^{(\ell+1)} \in \{-3, -2, -1, 0, +1, +2, +3\}$.
- **The *a posteriori* probability update rule at a VNU**

$$\gamma_n^{(\ell)} = I_n + \sum_{c \in \mathcal{V}(v_n)} m_{c \to v_n}^{(\ell)}.$$

MS Based decoder

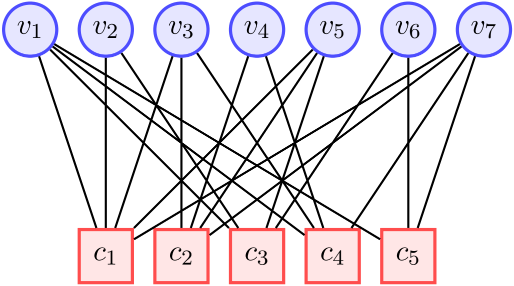
$\boldsymbol{x} =$ x_1 x_2 x_3 x_4 x_5 x_6 x_7



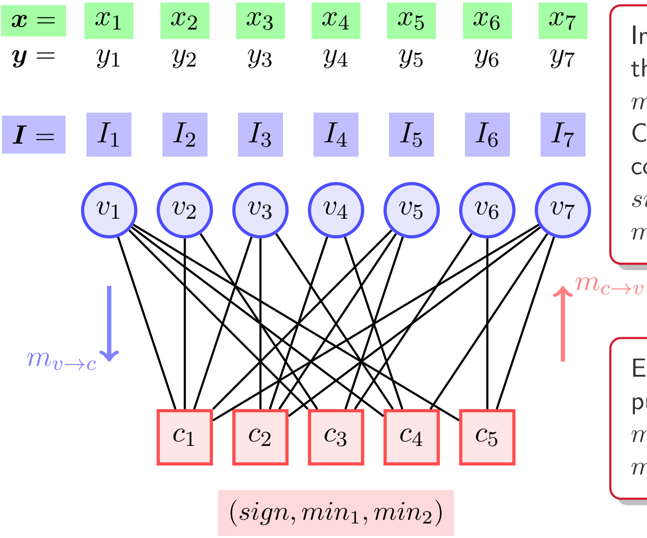
A codeword $\boldsymbol{x}$, which is mapped by *e.g.* the BPSK modulation, is sent through a noisy channel, *e.g.* the BI-AWGN.

MS Based decoder

$\boldsymbol{x} =$	x_1	x_2	x_3	x_4	x_5	x_6	x_7
$\boldsymbol{y} =$	y_1	y_2	y_3	y_4	y_5	y_6	y_7
$\boldsymbol{I} =$	I_1	I_2	I_3	I_4	I_5	I_6	I_7



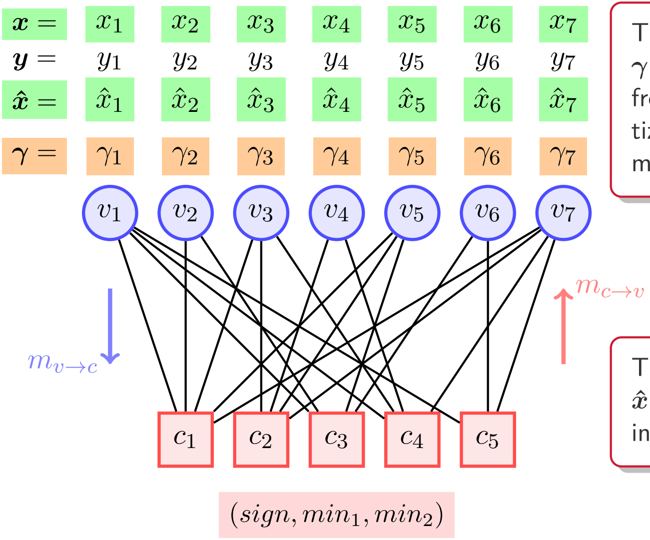
From the output channel $\boldsymbol{y}$, the quantized LLR values $\boldsymbol{I}$ are computed for a fixed precision q.



In each iteration the messages $m_{v \to c}$ are sent to CNUs, each CNU computes 3 values: $sign$, min_1, and min_2.

Each CNU computes the messages $m_{c \to v}$ using $sign$, min_1, and min_2.

MS Based decoder



The APP values γ are obtained from the quantized LLRs $\boldsymbol{I}$ and messages $m_{c \to v}$.

The estimated bits $\hat{\boldsymbol{x}}$ are obtained using the APPs γ.

Low precision Offset Min-Sum Based Decoders

From the analysis of OMS-based decoders with $q = 3$

(i) The quantized LLRs $I_n \in \mathcal{A}_C = \{-3, -2, -1, 0, +1, +2, +3\}$,
(ii) v-to-c messages $m^{(\ell+1)}_{v_n \to c_m} \in \mathcal{A}_C = \{-3, -2, -1, 0, +1, +2, +3\}$, and
(iii) c-to-v messages $m^{(\ell)}_{c_m \to v_n} \in \mathcal{A}_C \backslash \{-3, +3\} = \{-2, -1, 0, +1, +2\}$.

The OMS-based decoders are suboptimal

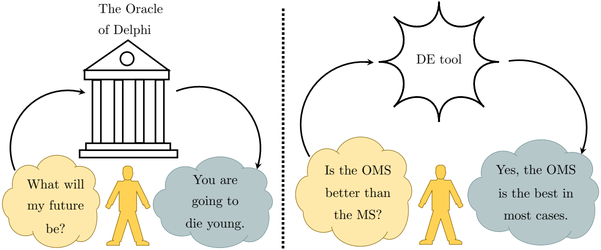
It can be clearly noted that all combinations that can be obtained from $q = 3$ bits are not used.

How to use the 8 quantization levels for precision $q = 3$

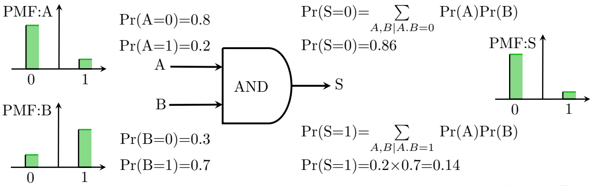
- The quantization of LLRs has to be changed.
- CNU has to be changed.
- VNU has to be changed.

Density Evolution for quantized decoders

- Idea of the Density evolution tool.



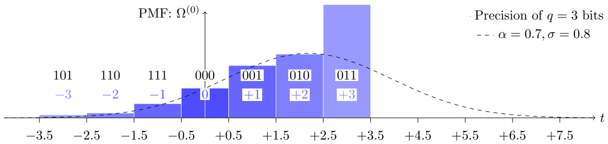
- Probability mass function (PMF) of a nonlinear system.



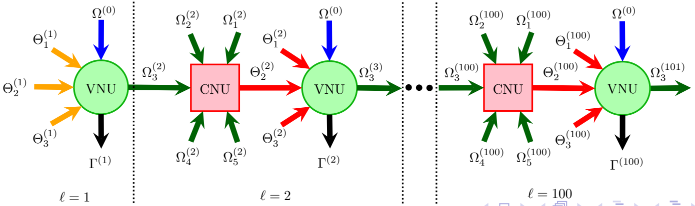
Density Evolution for quantized decoders

The goal of DE is to recursively compute the PMF of the exchanged messages in the Tanner graph along the iterations.

- DE is initialized with the PMF of the BI-AWGN $\Omega^{(0)}$.



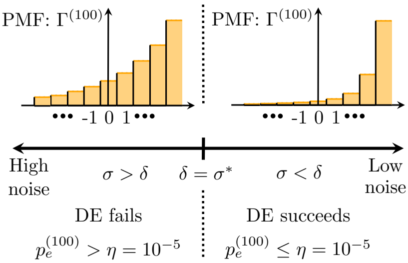
- $\Theta^{(\ell)}$ (resp. $\Omega^{(\ell)}$): PMF of c-to-v (resp. v-to-c) messages in the ℓ^{th} iteration. $\Gamma^{(\ell)}$: PMF of the APP.



Density Evolution for quantized decoders

The bit error probability $p_e^{(\ell)}$ is obtained from $\Gamma^{(\ell)}$.

- DE threshold δ.



- In the literature DE is used to compare different LDPC codes using the BP decoder [1]. It is used for example to optimize the degree distribution $(\lambda(x), \rho(x))$ [2].

In this work we fix the LDPC code and compare the DE thresholds of different decoders.

- Example: $(3, 6)$-regular LDPC code, BI-AWGN, and $q = 3$, $\delta^{OMS} = 2.204$ dB, $\delta^{MS} = 1.789$ dB $\Rightarrow$ DE gain of 0.415 dB.

Contributions

Contribution 1

The definition of a improved decoder named Noise-Against-Noise Min-Sum (NAN-MS) decoder.

Contribution 2

The definition of a improved decoder named Sign-Preserving Min-Sum (SP-MS) decoder..

Contribution 3

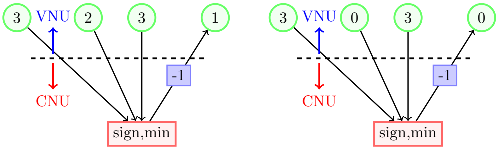
The definition of a post-processing algorithm for low precision iterative decoders.

Outline

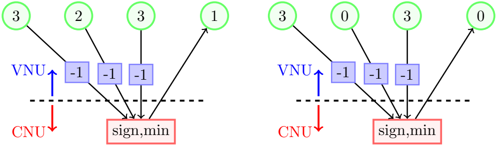
CNU of NAN Decoders

The update rule at a CNU can be written in two equivalent ways.

Classical method: $\left|m^{(\ell)}_{c_m \to v_n}\right| = \max\left\{ \min_{v \in \mathcal{V}(c_m)\setminus\{v_n\}} \left|m^{(\ell)}_{v \to c_m}\right| - 1, 0 \right\}.$



Equivalent method: $\left|m^{(\ell)}_{c_m \to v_n}\right| = \min_{v \in \mathcal{V}(c_m)\setminus\{v_n\}} \left(\max\left(\left|m^{(\ell)}_{v \to c_m}\right| - 1, 0 \right) \right).$



Therefore, we can move the offset from CNs to VNs.

VNU of NAN-MS Decoders

Moving the offset from CNs to VNs, the update rule at a VNU of OMS-based decoders can be rewritten as

$$m_{v_n \to c_m}^{(\ell+1),U} = I_n + \sum_{c \in \mathcal{V}(v_n) \setminus \{c_m\}} m_{c \to v_n}^{(\ell)}.$$

The offset is subtracted before saturation to allow -3 and 3 values in the variable to check message.

$$m_{v_n \to c_m}^{(\ell+1)} = \text{sign}\left(m_{v_n \to c_m}^{(\ell+1),U}\right) \times \mathcal{S}\left(\max\left(\left|m_{v_n \to c_m}^{(\ell+1),U}\right| - 1, 0\right), N_q\right)$$

Questions to answer

Where do we add the noise?

- Several possible locations.

Are NAN-MS decoders better?

- Performance to be compared with noiseless MS/OMS decoder.

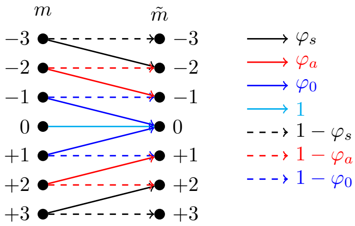
Are the NAN-MS decoders very complex?

- Complexity to be compared with noiseless MS/OMS decoder.

Noise Models Υ

The noise models need to be memoryless and have to satisfy the symmetry condition $\Pr(\tilde{m} = \beta_2 | m = \beta_1) = \Pr(\tilde{m} = -\beta_2 | m = -\beta_1), \forall(\beta_1, \beta_2)$.

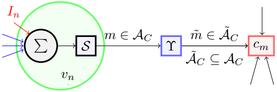
- $\Upsilon : \mathcal{A}_N \to \tilde{\mathcal{A}}_N$ denotes the function which transforms a noiseless message $m \in \mathcal{A}_N$ into a noisy message $\tilde{m} \in \tilde{\mathcal{A}}_N$. Unless otherwise stated $\mathcal{A}_N = \tilde{\mathcal{A}}_N = \mathcal{A}_C$
- Υ is defined by the conditional PDF (CPDF) $\Pr(\tilde{m}|m)$.



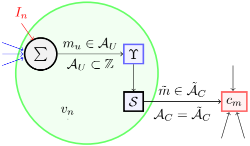
$\boldsymbol{\varphi} = (\varphi_s, \varphi_a, \varphi_0)$ will be optimized with the noisy DE tool.

Localization of the Noise Injection

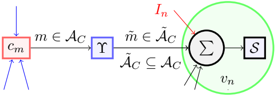
- Noise Outside the Variable Node Update (NOV)



- Noise Inside the Variable Node Update (NIV)



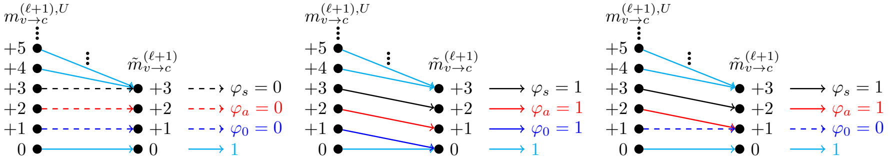
- Noise Outside the Check Node Update (NOC)



Noisy DE

DE takes into account the noise injection model.

Example of NAN-MS decoders for regular LDPC codes

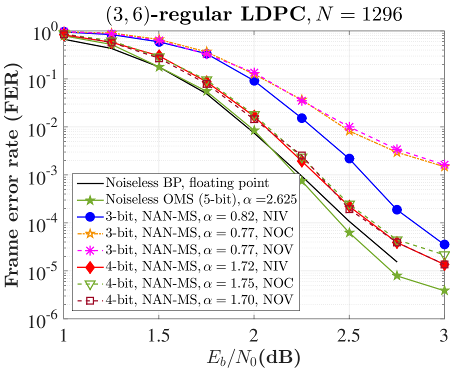


(a) MS. (b) OMS. (c) Modified OMS.

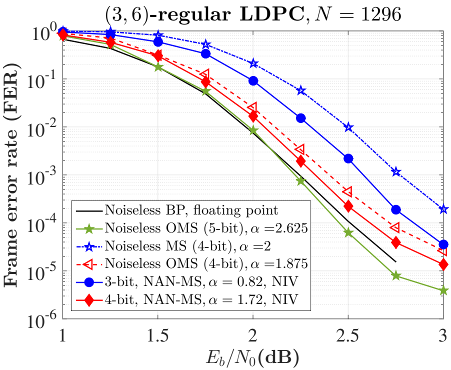
		MS	OMS	NIV	NOV	NOC		
(d_v, d_c)	q	δ_{db}	δ_{db}	$\tilde{\delta}_{db}$	$\tilde{\delta}_{db}$	$\tilde{\delta}_{db}$	DE gain	SNR gain
$(3, 6)$	3	**1.7888**	2.2039	**1.5711**	1.5995	1.6306	**0.2177**	**0.2**
	4	1.6437	**1.3481**	**1.2877**	1.2917	1.2931	**0.0604**	**0.05**
$(4, 8)$	3	2.7360	**2.3219**	**2.1056**	2.1105	2.1119	**0.2163**	**0.2**
	4	2.5389	**1.7509**	**1.7411**	**1.7411**	1.7488	**0.0098**	**0.000**

SNR gains in the waterfall correspond to what was predicted with DE.

FER performance for the (3,6)-regular LDPC code



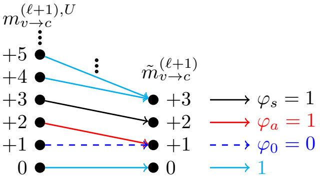
(a) FER performance of NAN-MS decoders implemented with the NIV, NOV, and NOC model.



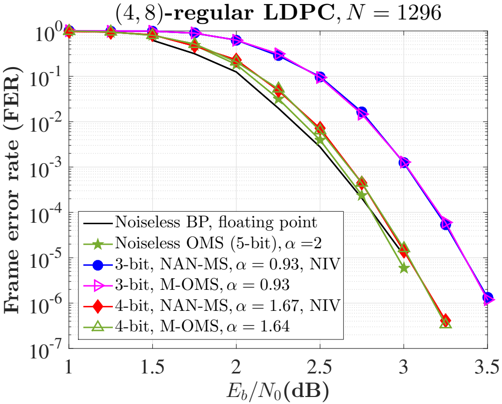
(b) For $q = 3$, DE gain $=$ 0.2177 and SNR gain $=$ 0.2. For $q = 4$, DE gain $=$ 0.0604 and SNR gain $=$ 0.05

Implementation of NAN-MS: Modified OMS decoders

DE thresholds results and the FER performance for the $(4, 8)$-regular LDPC code.

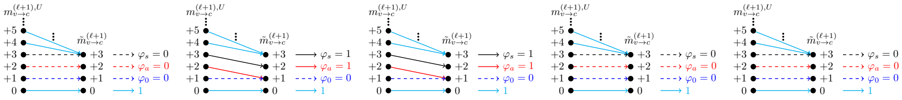


Regular LDPC code, BI-AWGN channel					
		NAN-MS		M-OMS	
R	q	α^*	$\tilde{\delta}_{db}$	α^*	δ_{db}
1/2	3	0.93	**2.1056**	0.93	**2.1061**
	4	1.67	**1.7411**	1.64	**1.7514**



Implementation of NAN-MS Using an Offset Vector

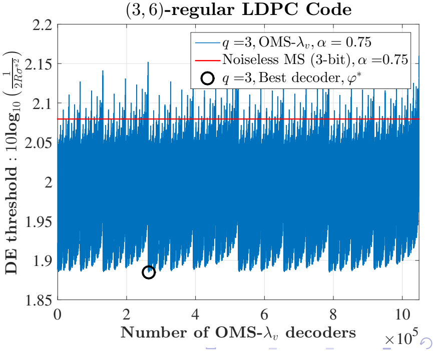
$\boldsymbol{\lambda_v} = ({\lambda_v}^{(5)}, {\lambda_v}^{(4)}, {\lambda_v}^{(3)}, {\lambda_v}^{(2)}{\lambda_v}^{(1)}) = 00110_2 = 6_{10}$ for 5 iterations.



(a) ${\lambda_v}^{(1)} = 0.$ (b) ${\lambda_v}^{(2)} = 1.$ (c) ${\lambda_v}^{(2)} = 1.$ (d) ${\lambda_v}^{(4)} = 0.$ (e) ${\lambda_v}^{(5)} = 0.$

DE thresholds results and the FER performance for the $(3, 6)$-regular LDPC code.

Regular LDPC code, BI-AWGN channel					
		M-OMS		OMS-$\boldsymbol{\lambda_v}$	
R	q	α^*	$\tilde{\delta}_{db}$	α^*	δ_{db}
1/2	3	0.78	**2.1352**	0.75	**1.8848**
	4	1.78	**1.6612**	1.47	**1.6094**



Implementation of NAN-MS Using an Offset Vector

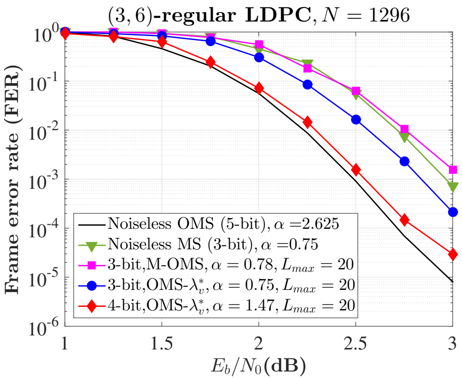


Figure: FER performance of the M-OMS and the OMS-$\boldsymbol{\lambda_v}$ decoders.

The CNU is the same for the OMS/M-OMS/OMS-$\boldsymbol{\lambda_v}$. The VNU of the M-OMS/OMS-$\boldsymbol{\lambda_v}$ is almost the same as the VNU of the OMS.

Answer

Where do we add the noise?

- Inside the VNU (before the saturation function).

Are NAN-MS decoders better?

- The NAN-MS decoders are better decoders than the noiseless MS decoders and noiseless OMS decoders. The NAN-MS decoder presents a SNR gain up to 0.40 dB.

Are the NAN-MS decoders very complex?

- The NAN-MS decoder is more complex than a noiseless MS decoder and a noiseless OMS decoder.
- The M-OMS decoder and the OMS-$\boldsymbol{\lambda}_v$ have the same complexity than a noiseless OMS decoder.

Outline

Questions to answer

How to use the 8 quantization levels for precision $q = 3$

The NAN-MS decoder uses 7 levels. We define a new decoder called Sign-Preserving decoder where:

- The quantization of LLRs has to be changed.
- VNU has to be changed.

Channel quantization and exchanged messages

- If we exclude the 0 LLR value, can we conceive good decoders?

Are SP-MS decoders better?

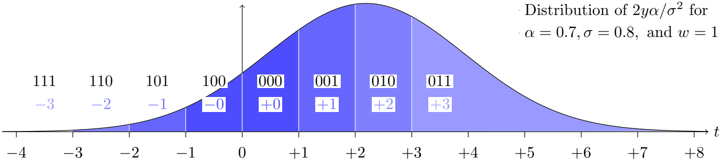
- Are SP-MS decoders better than MS and OMS decoders?
- Are SP-MS decoders better than NAN-MS decoders?

Are the SP-MS decoders very complex?

- Is a SP-MS decoder more complex than a noiseless OMS decoder?

Quantization used for Sign-Preserving Min-Sum Decoders

- In order to use the 2^q levels of q bits, we define the message alphabet as $\mathcal{A}_S = \{-N_q, ..., -1, -0, +0, +1, .., +N_q\}$. Example: we have $\mathcal{A}_S = \{-1, -0, +0, +1, \}$ for $q = 2$, using the sign-and-magnitude representation we get $10_2 = -0$, $00_2 = +0$, etc.
- A new quantization process $\mathcal{Q}^*$ is used: $I = \mathcal{Q}^*(LLR(y)) \in \mathcal{A}_L$ where $\mathcal{Q}^*(a) = (\text{sign}(a), \mathcal{S}(\lceil \alpha \times |a| \rceil - 1, N_{ch}))$, with $N_{ch} = 2^{q_{ch}-1} - 1$ and $2 \leq q \leq q_{ch}$.



The *channel gain factor* α represents a degree of freedom in the decoder definition.

VNU of Sign-Preserving Min-Sum Decoders

Studying the NAN-MS decoders, we obtained the unsaturated v-to-c message

$$m_{v_n \to c_m}^{(\ell+1),U} = I_n + \sum_{c \in \mathcal{V}(v_n) \setminus \{c_m\}} m_{c \to v_n}^{(\ell)}.$$

Problem of the update rule at VNUs

The v-to-c message $m_{v_n \to c_m}^{(\ell+1)}$ can be zero, zero does not have any information about the bit value, this means that the VNU can erase the bit value.

Idea to avoid the zero value.

- **Modified VNU:** we propose to use the signs of the c-to-v messages and the sign of the quantized LLR.

Example of SP-MS decoders for regular LDPC codes

		MS	OMS	NAN-MS	SP-NA-MS	SP-MS		
(d_v, d_c)	(q_{ch}, q)	δ_{db}	δ_{db}	$\tilde{\delta}_{db}$-NIV	$\tilde{\delta}_{db}$	δ_{db}	DE gain	SNR gain
$(3, 6)$	$(3, 2)$	–	–	–	**1.9033**	1.9315	–	–
	$(3, 3)$	**1.7888**	2.2039	1.5711	**1.4994**	1.5096	**0.2894**	**0.27**
	$(4, 3)$	–	–	–	**1.3726**	1.3910	–	–
	$(4, 4)$	1.6437	**1.3481**	1.2877	**1.2688**	**1.2688**	**0.0793**	**0.06**
$(6, 32)$	$(3, 2)$	–	–	–	**3.3979**	**3.3979**	–	–
	$(3, 3)$	4.0812	**3.5928**	3.5766	**3.3963**	**3.3963**	**0.1965**	**0.19**
	$(4, 3)$	–	–	–	**3.1740**	**3.1740**	–	–
	$(4, 4)$	3.8154	**3.1685**	3.1685	**3.1787**	**3.1787**	**-0.0102**	**0.0**

SNR gains in the waterfall correspond to what was predicted with DE.

For $(q_{ch}, q = q_{ch})$, the SP-MS is better decoder compared to NAN-MS, MS, OMS decoders. SP-MS gives a SNR gain up to 0.43 dB.

For $d_v > 4$, the SP-MS can change the precision of messages from $q = q_{ch}$ to $q = q_{ch} - 1$ maintaining the same performance.

FER performance for the ETHERNET code.

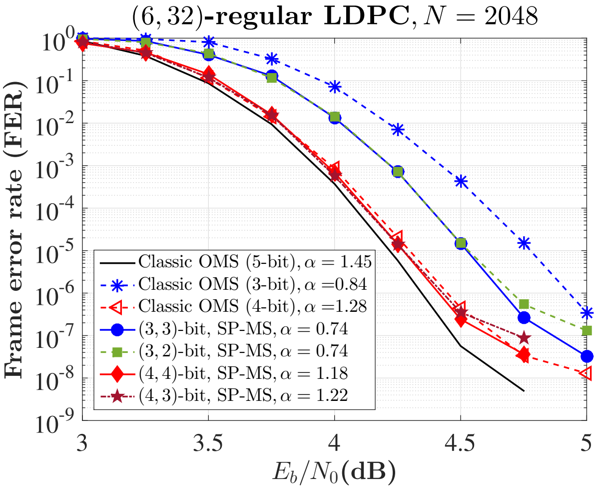
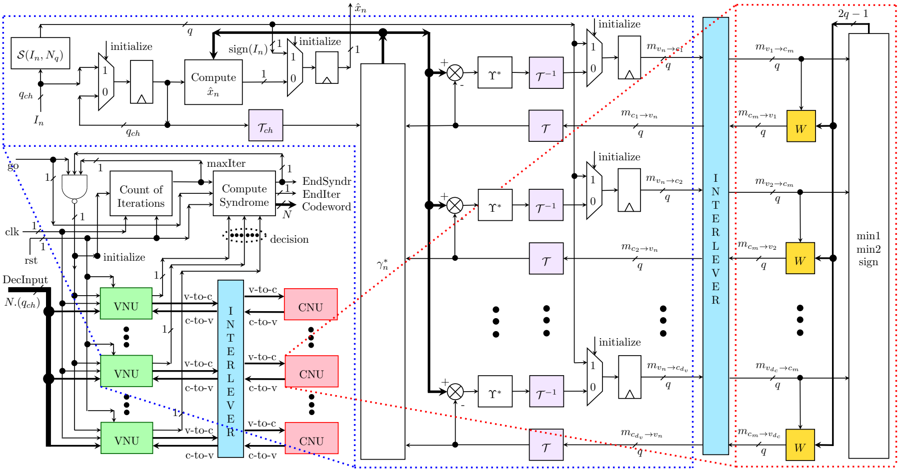


Figure: FER performance of SP-MS decoders for precision $q \in \{2, 3, 4\}$.

The fully parallel architecture of SP-MS decoders implemented on a FPGA and on an ASIC of 28 nm techology.

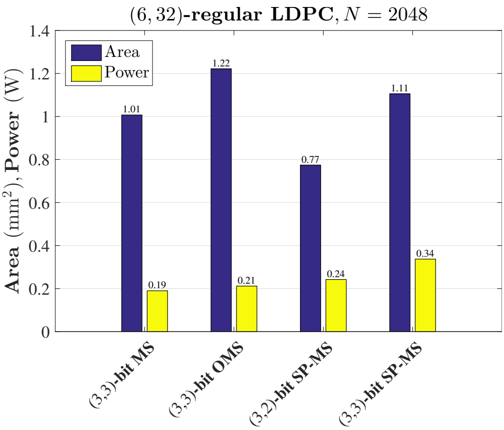


Implementation of SP-MS decoders using the 28 nm technology

Here we present the ASIC synthesis results for the $(6, 32)$-regular LDPC code.

Regular LDPC code, $q_{ch} = q = 3$			
Decoder	Freq. (MHz)	Area (mm^2)	Power (mW)
MS	600	1.007180	189.64
OMS		1.221662	211.45
SP-MS		1.105530	336.91

IEEE 802.3 ETHERNET code, SP-MS decoders				
Precision (q_{ch}, q)	Area (mm^2)		Power (mW)	
$(3, 3)$	1.105530	(0.0%)	336.91	(0.0%)
$(3, 2)$	0.774331	(-29.96%)	242.19	(-28.11%)



Answer

Channel quantization and exchanged messages

- We can conceive good decoders excluding the 0 LLR value for low precision decoders.

Are SP-MS decoders better?

- The SP-MS decoders are better decoders than the OMS and NAN-MS decoders. The SP-MS decoders present a SNR gain up to 0.43 dB.
- The precision of messages can be reduced by one bit maintaining the same error-correcting performance.

Are the SP-MS decoders very complex?

- The reduction of one bit in the precision of the messages helps to reduce the area consumed by the SP-MS decoder by at least 25%.

Outline

Context and Question to answer

Context

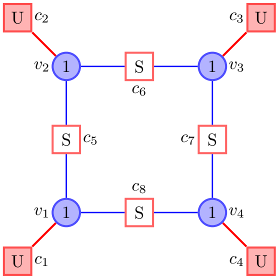
- Considering the IEEE 802.3 ETHERNET code, the $(q_{ch} = 3, q = 2)$-bit SP-MS decoder exhibit an error floor at a FER level of 10^{-7} making the SP-MS unacceptable.

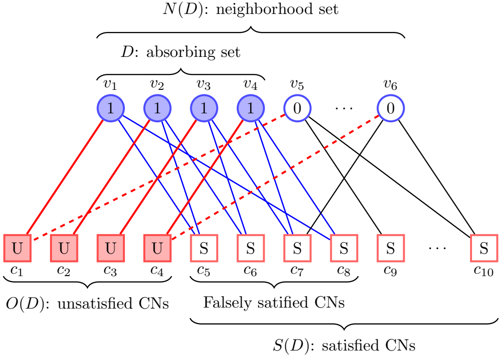
Error floor

- Can we lower the error floor of this low precision decoder?

Trapping set in SP-MS decoders

In the literature, a general (a, b) trapping set is defined as a set of a VNs which induces a subgraph with exactly b odd-degree CNs and an arbitrary number of even-degree CNs.



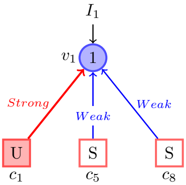


The $(8, 8)$ trapping sets of the IEEE 802.3 ETHERNET gives $|O(D)| = 8$, $|N(D)| = 256$, and $|N(D) \backslash D| = 248$.

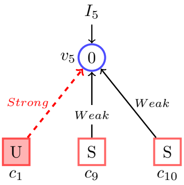
Post-Processing by Z. Zhang [3]: Message biasing

During one iteration (at $\ell = \ell_{pp} = 12$), the update rule of the CNU is changed: $|m_{c \to v}| = 1$ $(Strong)$ if c is an unsatisfied CN, and $|m_{c \to v}| = 0$ $(Weak)$ if c is a satisfied CN.

VNU in the set D



VNU in the set $N(D)\backslash D$

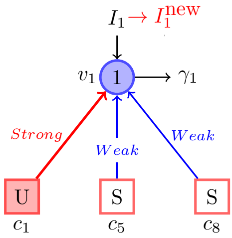


Message biasing can only correct around 30% of all the $(8, 8)$ trapping sets.

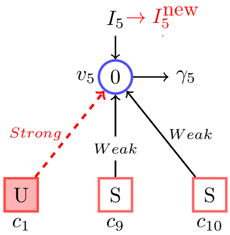
Precision (q_{ch}, q)	SNR (dB)	Before post-proc. run the SP-MS Nb. of $(8, 8)$ sets	After post-proc. message biasing Nb. of $(8, 8)$ sets
$(3, 2)$	4.9	281	189
	5.0	88	61

Proposed Post-Processing Algorithm

VNU in the set D



VNU in the set $N(D) \backslash D$

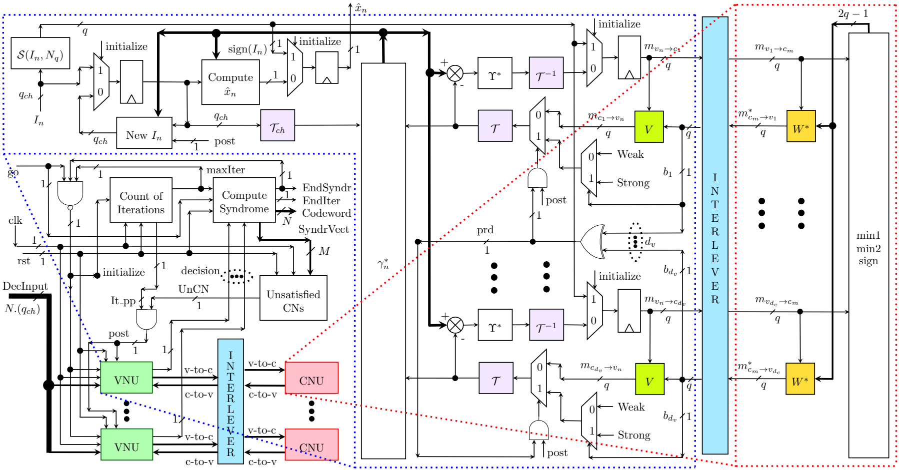


Enhanced Post-Processing Algorithm.

- **Message biasing** at $\ell = \ell_{pp}$ with the same method.
- **Compute the APP:** for all VNs, determine $\gamma_n^{(\ell_{pp})}$, for $n = 1, 2, ..., N$.
- **Change I values**: Set LLR I_n to I_n^{new} for the remaining iteration with $I_n^{\text{new}} = I_n$ or $(-\text{sign}(I_n), 0)$ according to some joint condition on $|\gamma_n^{(\ell_{pp})}|$ and $|I_n|$.

Post-processing architecture of the (3,2)-bit SP-MS

The fully parallel architecture for the IEEE 802.3 ETHERNET code implemented on a FPGA and on an ASIC of 28 nm techology.

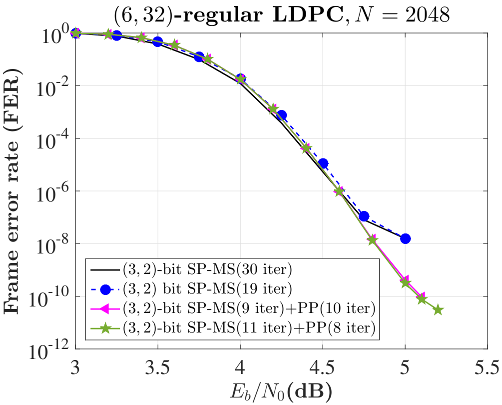


Simulation and emulation results

Here we present the simulations and emulation results for the IEEE 802.3 ETHERNET code using a fully parallel architecture.

Number of uncorrected $(8,8)$ absorbing set errors		
	Before post-proc.	After post-proc.
SNR	run the SP-MS	proposed post-proc.
(dB)	Nb. of $(8,8)$ sets	Nb. of $(8,8)$ sets
4.9	281	**0**
5.0	88	**0**

$(3,2)$-bit SP-MS decoder		
SNR (dB)	$L_{pp} = 12,\ L_{max} = 19$	
	Average Iter.	Throughput (Gbps)
4.0	6.74	127.92
4.5	4.27	201.92
5.0	3.26	264.48
5.5	2.70	319.34

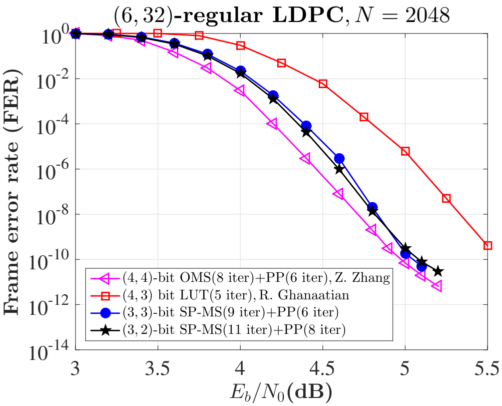


Place and route results

Place and route results for the IEEE 802.3 ETHERNET code using a fully parallel architecture and 28 nm technology.

Place and Route		
SP-MS decoder	$(3,2)$-bit	$(3,3)$-bit
Frequency (MHz)	421	**500**
Core area (mm^2)	**1.76**	2.56
Density	**60%**	40%
Throughput (Gbps)†	319.34	**384.96**
Hardware Eff. (Gbps/mm^2)†	**181.44**	150.38

† At $E_b/N_0 = 5.5$ dB (preliminary results).



The precision $(3,2)$ saves 31.25% of the area and it helps to reduce the routing congestion.

The proposed post-processing algorithm is very efficient lowering the error floor below a FER of 10^{-10}.

Comparison with other works

IEEE 802.3 LDPC code: Rate 0.8143 $(2048, 1723)$ with regular $(6, 32)$ degree distribution, and precision (3,2).

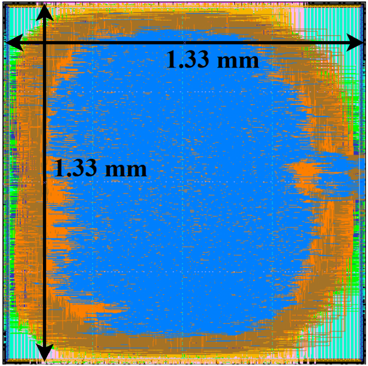
	This work	Ghanaatian,2018,[4]	Zhang,2010,[5]
Technology	28nm FDSOI	28nm FDSOI	65nm CMOS
Decoder	SP-MS	finite-alphabet	OMS
Precision (q_{ch}, q)	(3,2) bits	(4,3) bits	(4,4) bits
Iterations	11 + 8 post-proc.	5	8 + 6 post-proc.
Architecture	full-parallel	unrolled full-parallel	partial-parallel
E_b/N_0 @ FER$= 10^{-10}$	5.07 dB	5.51 dB	4.98 dB
Frequency	421 MHz†	862 MHz	700 $\xrightarrow{28nm}$ 1000 MHz
Core area	1.76 mm^{2†}	16.2 mm^2	5.05 $\xrightarrow{28nm}$ 0.9371 mm^{2*}
Throughput (5.5 dB)	319.34 Gbit/s/mm^{2†}	588 Gbit/s/mm^2	47.7 $\xrightarrow{28nm}$ 68.14 Gbit/s/mm^2
Hardware eff. (5.5 dB)	181.44 Gbit/s/mm^{2†}	36.3 Gbit/s/mm^2	9.45 $\xrightarrow{28nm}$ 72.71 Gbit/s/mm^2

† Preliminary results.

* The scale factor for for the area is k^2, where k is the relative dimension to 28 nm.

The (3,2)-bit SP-MS decoder layout

The $(3, 2)$-bit SP-MS decoder was synthesized from a VHDL description using Synopsys Design Compiler and the placed and routed using Cadence Encounter Digital Implementation.



Thanks to Prof. Andreas Burg and Reza Ghanaatian (Ecole polytechnique fédérale de Lausanne (EPFL)) for their support in the ASIC implementation.

Outline

Conclusions

Nosiy decoder
Low precision →

Full message
precision →

Waterfall Region

Noisy		Noiseless	
NAN-MS	→	M-OMS OMS-$\boldsymbol{\lambda_v}$	[PIMRC 2017] unrolled decoder [-]
SP-NA-MS	→	SP-MS $(q_{ch}, q = q_{ch} - 1)$ good for $d_v > 4$	[Patent, ISTC 2018] [submission to IEEE TCOM]

LDPC $(d_v, d_c) = (6, 32)$
(3,2)-bit precision
Fully parallel
FPGA/ASIC
implementation →

Error Floor Region

New post-processing algorithm [-]

FER$< 10^{-10}$, Hardware eff. > 100 Gbit/s/mm^2

Future work

Improved 10Gbit ETHERNET LDPC ASIC Implementation: still room for improvement (preliminary results: $+$ 50 $\%$ of efficiency before place&route).

Study high speed implementation of others LDPC codes (post-doc position).

Extend the method for others type of low precision decoders (Turbo-Code, Polar Code,...).

Thank you for listening!

Questions?

Publications

[P1] **F. Cochachin**, E. Boutillon "ITERATIVE DECODER FOR DECODING A CODE COMPOSED OF AT LEAST TWO CONSTRAINT NODES", 03 december 2018, application number: EP18306599.4

[J1] **F. Cochachin**, E. Boutillon, D. Declercq, "Sign-Preserving Min-Sum Decoders", Submitted to IEEE Transactions on Communications, February 2019.

[C1] **F. Cochachin**, E. Boutillon and D. Declercq, "Optimization of Sign-Preserving Noise-Aided Min-Sum Decoders with Density Evolution," 2018 IEEE 10th International Symposium on Turbo Codes & Iterative Information Processing (ISTC), Hong Kong, Hong Kong, 2018, pp. 1-5.

[C2] **F. Cochachin**, D. Declercq, E. Boutillon and L. Kessal, "Density evolution thresholds for noise-against-noise min-sum decoders," 2017 IEEE 28th Annual International Symposium on Personal, Indoor, and Mobile Radio Communications (PIMRC), Montreal, QC, 2017, pp. 1-7.

SP-MS decoder and the Binary Message Passing decoder

DE thresholds results for the $(7, 112)$-regular LDPC code.

OMS			
R	q	α^*	δ_{db}^{OMS}
0.9375	5	1.32	**4.7186**

SP-MS				
R	(q_{ch}, q)	α^*	δ_{db}^{SP-MS}	$\delta_{db}^{OMS} - \delta_{db}^{SP-MS}$
0.9375	$(3, 2)$	0.76	**4.9985**	0.2799
	$(3, 3)$	0.76	**4.9985**	0.2799
	$(4, 3)$	1.20	**4.7537**	0.0351
	$(4, 4)$	1.20	**4.7540**	0.0354

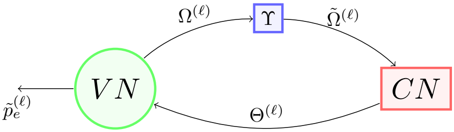
- For the decoder proposed by G. Lechner [6]: SNR (soft) $-$ SNR ([6]) $= 0.24$ dB.
- For the (3,2)-bit SP-MS decoder: $\delta_{db}^{OMS} - \delta_{db}^{SP-MS} = 0.2799$ dB.
- The (3,2)-bit SP-MS decoder has the same performance as the decoder of [6].

Noisy Density Evolution for the NAN-MS decoder

- In the literature a noisy version of DE is used to study a faulty hardware, *e.g.* the work done by Lav R. Varshney [7].
- A noisy MS decoder is studied with a noisy DE by C. K. Ngassa [8].

In this work we have defined a noisy DE to compute noisy DE thresholds of NAN-MS decoders.

- Concept of noisy DE calculation for the NOV model.



$\tilde{\Omega}^{(\ell)}$: PMF of noisy v-to-c messages in the ℓ^{th} iteration.

VNU of Sign-Preserving Min-Sum Decoders

Studying the NAN-MS decoders, the update rule at a VNU of OMS-based decoders was rewritten as

$$m_{v_n \to c_m}^{(\ell+1)} = \text{sign}\left(m_{v_n \to c_m}^{(\ell+1),U}\right) \times \mathcal{S}\left(\max\left(\left|m_{v_n \to c_m}^{(\ell+1),U}\right| - 1, 0\right), N_q\right)$$

where the unsaturated v-to-c message is given by

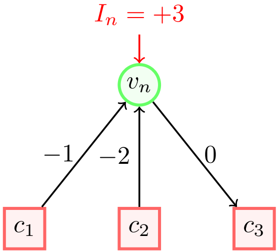
$$m_{v_n \to c_m}^{(\ell+1),U} = I_n + \sum_{c \in \mathcal{V}(v_n) \setminus \{c_m\}} m_{c \to v_n}^{(\ell)}.$$

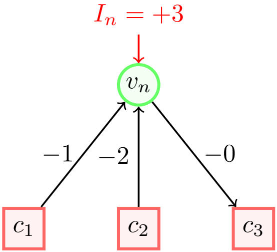
Problem of the update rule at VNUs

The v-to-c message $m_{v_n \to c_m}^{(\ell+1)}$ can be zero, zero does not have any information about the bit value, this means that the VNU can erase the bit value.

VNU of Sign-Preserving Min-Sum Decoders

Classical VNU: **Modified VNU:**





Idea to avoid the zero value.

- **Modified VNU:** we propose to use half the sum of the signs of the c-to-v messages and the sign of the quantized LLR
  Example: $a = (+\mathbf{3}-\mathbf{1}-\mathbf{2}) + (+\mathbf{1}-\mathbf{1}-\mathbf{1})/2 = -\mathbf{0.5}$, hence $m^{(\ell),U}_{v_n \to c_3} = (\text{sign}(a), \lfloor |a| \rfloor) = -\mathbf{0}$ and $m^{(\ell)}_{v_n \to c_3} = -\mathbf{0}$.

We define the Sign-Preserving Min-Sum decoders as decoders that always preserve the sign of the messages [9].

Update rules for SP-MS Decoders

- Let us denote by $\mu^{(\ell)}_{v_n \to c_m}$ the *sign-preserving factor* of the message $m^{(\ell+1)}_{v_n \to c_m}$, defined as

$$\mu^{(\ell)}_{v_n \to c_m} = \xi \times \text{sign}(I_n) + \sum_{c \in \mathcal{V}(v_n) \setminus \{c_m\}} \text{sign}\left(m^{(\ell)}_{c \to v_n}\right),$$

where $\xi = \begin{cases} 0, & d_v = 2, \\ 1, & d_v \in \{3, 5, 7, ...\}, \\ 2, & d_v \in \{4, 6, 8, ...\}. \end{cases}$

- Let us redefine $m^{(\ell+1),U}_{v_n \to c_m}$ as

$$m^{(\ell+1),U}_{v_n \to c_m} = \frac{\mu^{(\ell)}_{v_n \to c_m}}{2} + I_n + \sum_{c \in \mathcal{V}(v_n) \setminus \{c_m\}} m^{(\ell)}_{c \to v_n}$$

$m^{(\ell+1),U}_{v_n \to c_m}$ takes its value in the set $\mathcal{A}_U = \{\dots, -1.5, -0.5, +0.5, +1.5, \dots\}$.

Update rules for SP-MS Decoders

- We obtain a message $m_{v_n \to c_m}^{(\ell+1)}$ that has always a defined sign as

$$m_{v_n \to c_m}^{(\ell+1)} = \left(\mathsf{sign}\left(m_{v_n \to c_m}^{(\ell+1),U}\right), \mathcal{S}\left(\mathsf{max}\left(\left\lfloor \left| m_{v_n \to c_m}^{(\ell+1),U} \right| \right\rfloor - 1, 0\right), N_q\right)\right).$$

- The APP update at a VN v_n of the SP-MS decoder is defined as follows

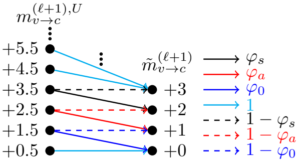
$$\gamma_n^{(\ell)} = I_n + \frac{1}{2} \times \xi \times \mathsf{sign}(I_n) + \sum_{c \in \mathcal{V}(v_n)} \left(m_{c \to v_n}^{(\ell)} + \frac{1}{2} \times \mathsf{sign}\left(m_{c \to v_n}^{(\ell)}\right)\right).$$

Optimization of Sign-Preserving Min-Sum Decoders

In order to optimize the SP-MS decoders, we use the optimization method proposed for NAN-MS decoders [10]. The optimization process is given by

$$\tilde{m}^{(\ell+1)}_{v_n \to c_m} = \Upsilon\left(m^{(\ell+1),U}_{v_n \to c_m}\right),$$

Where Υ is a noise model that also performs the saturation function.



$\boldsymbol{\varphi} = (\alpha, \varphi_s, \varphi_a, \varphi_0)$ will be optimized with the noisy DE tool.

After optimization we can obtain two kinds of decoders:

- A decoder that keeps some randomness named Sign-Preserving Noise-Aided Min-Sum (SP-NA-MS) decoder.
- Or a deterministic decoder named Sign-Preserving Min-Sum (SP-MS) decoder.

References I

D. Declercq, M. Fossorier, and E. Biglieri, *Channel Coding: Theory, Algorithms, and Applications*. Academic Press Library in Mobile and Wireless Communications, ISBN:978-0-12-396499-1, 2014.

T. J. Richardson, M. A. Shokrollahi, and R. L. Urbanke, "Design of capacity-approaching irregular low-density parity-check codes", *IEEE Transactions on Information Theory*, vol. 47, no. 2, pp. 619–637, Feb 2001, ISSN: 0018-9448. DOI: 10.1109/18.910578.

Z. Zhang, L. Dolecek, B. Nikolic, V. Anantharam, and M. J. Wainwright, "Lowering ldpc error floors by postprocessing", in *IEEE GLOBECOM 2008 - 2008 IEEE Global Telecommunications Conference*, Nov. 2008, pp. 1–6. DOI: 10.1109/GLOCOM.2008.ECP.590.

References II

R. Ghanaatian, A. Balatsoukas-Stimming, T. C. Müller, M. Meidlinger, G. Matz, A. Teman, and A. Burg, "A 588-gb/s ldpc decoder based on finite-alphabet message passing", *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, vol. 26, no. 2, pp. 329–340, Feb 2018, ISSN: 1063-8210. DOI: 10.1109/TVLSI.2017.2766925.

Z. Zhang, V. Anantharam, M. J. Wainwright, and B. Nikolic, "An efficient 10gbase-t ethernet ldpc decoder design with low error floors", *IEEE Journal of Solid-State Circuits*, vol. 45, no. 4, pp. 843–855, April 2010, ISSN: 0018-9200. DOI: 10.1109/JSSC.2010.2042255.

G. Lechner, T. Pedersen, and G. Kramer, "Analysis and design of binary message passing decoders", *IEEE Transactions on Communications*, vol. 60, no. 3, pp. 601–607, Mar. 2012, ISSN: 0090-6778. DOI: 10.1109/TCOMM.2011.122111.100212.

References III

L. R. Varshney, "Performance of ldpc codes under faulty iterative decoding", *IEEE Transactions on Information Theory*, vol. 57, no. 7, pp. 4427–4444, July 2011, ISSN: 0018-9448. DOI: 10.1109/TIT.2011.2145870.

C. K. Ngassa, V. Savin, E. Dupraz, and D. Declercq, "Density evolution and functional threshold for the noisy min-sum decoder", *IEEE Transactions on Communications*, vol. 63, no. 5, pp. 1497–1509, May 2015, ISSN: 0090-6778. DOI: 10.1109/TCOMM.2015.2388472.

F. Cochachin, E. Boutillon, and D. Declercq, "Optimization of sign-preserving noise-aided min-sum decoders with density evolution", in *2018 IEEE 10th International Symposium on Turbo Codes Iterative Information Processing (ISTC)*, Dec. 2018, pp. 1–5. DOI: 10.1109/ISTC.2018.8625309.

References IV

F. Cochachin, D. Declercq, E. Boutillon, and L. Kessal, "Density Evolution Thresholds for Noise-Against-Noise Min-Sum Decoders", *2017 IEEE 28th Annual International Symposium on Personal, Indoor, and Mobile Radio Communications (PIMRC)*, pp. 1–7, Oct 2017. DOI: 10.1109/PIMRC.2017.8292326.