Non-Binary LDPC codes

Cédric Marchand
Emmanuel Boutillon
name.surname@unv-ubs.fr

CNRS, UMR 6285, Lab-STICC
Centre de Recherche - BP 92116
F-56321 Lorient Cedex - FRANCE

Séminaire CentraleSupélec 3 mars 2016

Introduction

- Lab-STICC IAS team (interaction Algorithm architecture)
- Lab-STICC works on NB-LDPC since 2007 in the framework of FP7 DaVinci project.
- Oussama Abassi defended is PhD in 2014 on NB-LDPC architecture optimization.
- Since 2015 Lab-STICC a research engineer work on NB-LDPC implementation.
- Hassan Harb just started a PhD on the NB-LDPC and the associated architecture.
- Ahmed Abdmouleh PhD ending in 2016 studies the NB constellation optimization, matrix construction, spectral efficiency.
- Web page: http://www-labsticc.univ-ubs.fr/nb_ldpc/

Lab-STICC cnrs

OUTLINE

1) Introduction
   - LDPC
   - NB-LDPC
   - Galois Field

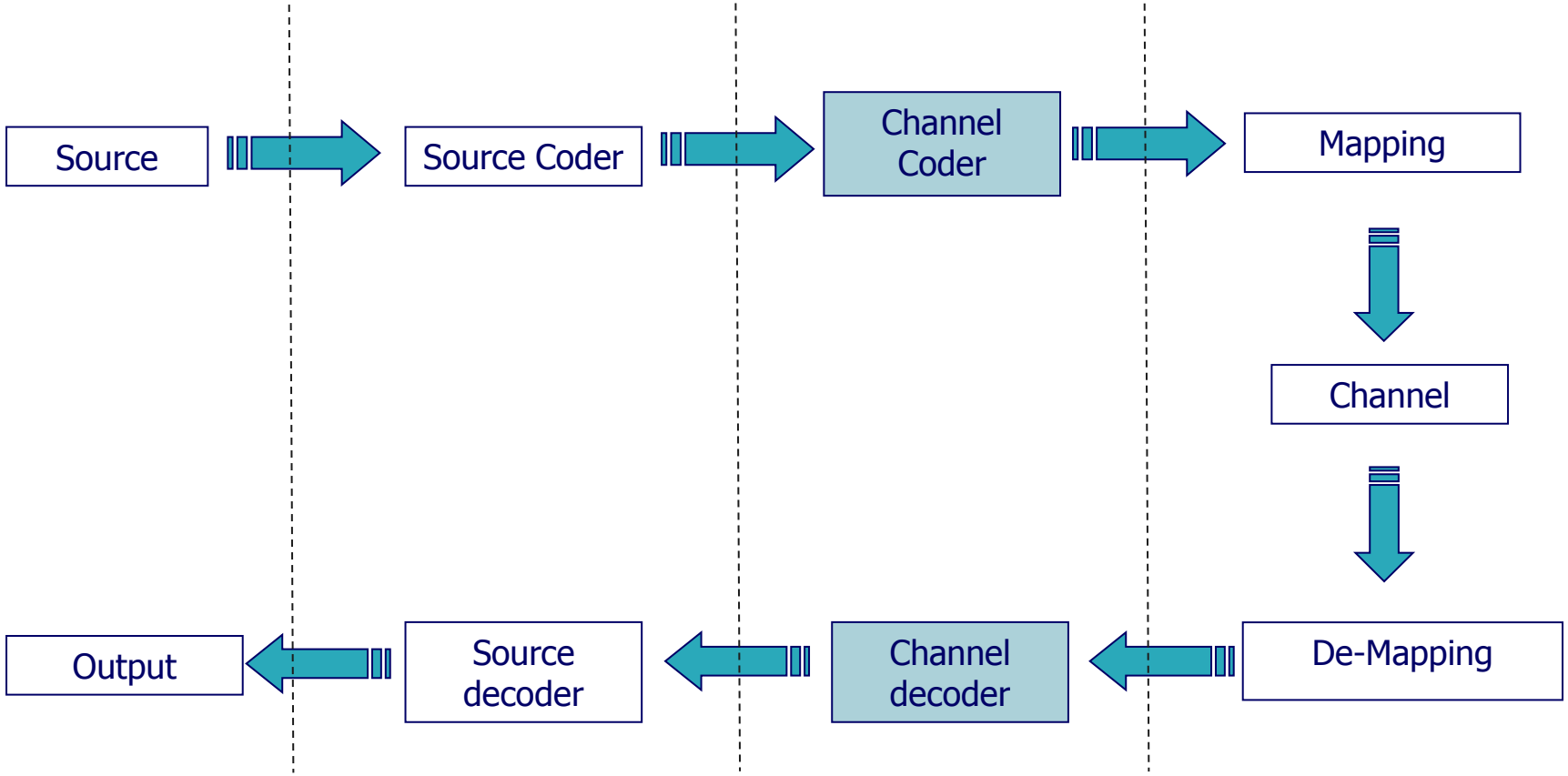
2) Decoding NB-LDPC

3) What are the pro and cons of NB-LDPC ?

Lab-STICC

cnrs

Digital Communication model



A brief history Of Low Density Parity Check

- Discovery of LDPC Codes — R.Gallager,1962.
- Turbo-Code — C.Berrou, A.Glavieux,P.Thitimasjshima,1993.
- Rediscovery of the LDPC Codes — D.MacKay,1996.

- LDPC codes are included many Standards
  - DBV-S2 (2003), DVB-T2(2009), DVB-C2, DVB-S2X
  - WiFi(2009), WiMax(2005),WPAN
  - 10GBase-T
  - ...
- Davey and MacKay prove that Non Binary LDPC have better performance than binary LDPC in 1998
- NB-LDPC is not included in any standard

Parity check equation

Given a word x[x_1 x_2 x_3 x_4]

$$x_1 \oplus x_2 \oplus x_3 \oplus x_4 = 0$$

$$P = \mathrm{mod}_2\left(\sum_{i=1}^{N} x_i\right)$$

Examples:

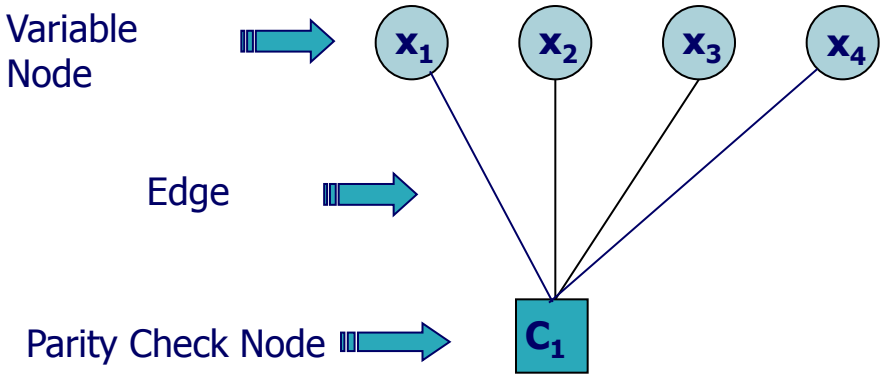
$0 \oplus 0 \oplus 0 \oplus 0 = 0$ ✓

$1 \oplus 0 \oplus 0 \oplus 0 = 1$ ✗

$1 \oplus 1 \oplus 0 \oplus 0 = 0$ ✓

$1 \oplus 0 \oplus 1 \oplus 1 = 1$ ✗

Tanner Graph representation:



Parity Check matrix

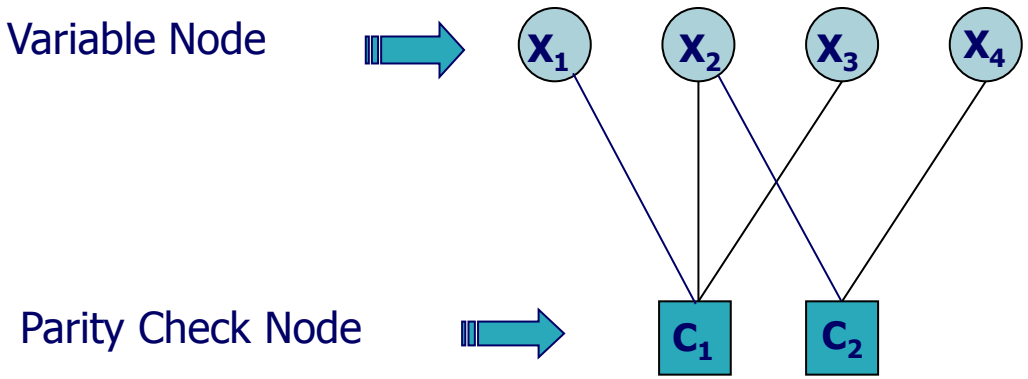
The parity check matrix is the set of parity equations:

$$H = \begin{bmatrix} 1 & 1 & 1 & 0 \\ 0 & 1 & 0 & 1 \end{bmatrix} \quad \begin{matrix} \text{(} X_1, X_2, X_3 \text{)} \\ \text{(} X_2, X_4 \text{)} \end{matrix}$$

$$c_1 = x_1 \oplus x_2 \oplus x_3$$

$$c_2 = x_2 \oplus x_4$$

Tanner Graph representation:



Decoding

- Belief propagation algorithm:
  - Based on graphical representation of the codes (Tanner graph)
  - Iterative decoding

Log Likelihood Ratio (LLR)

$$LLR_{x_0} = \ln\left(\frac{P_r(x_0 = 0 / y)}{P_r(x_0 = 1 / y)}\right)$$

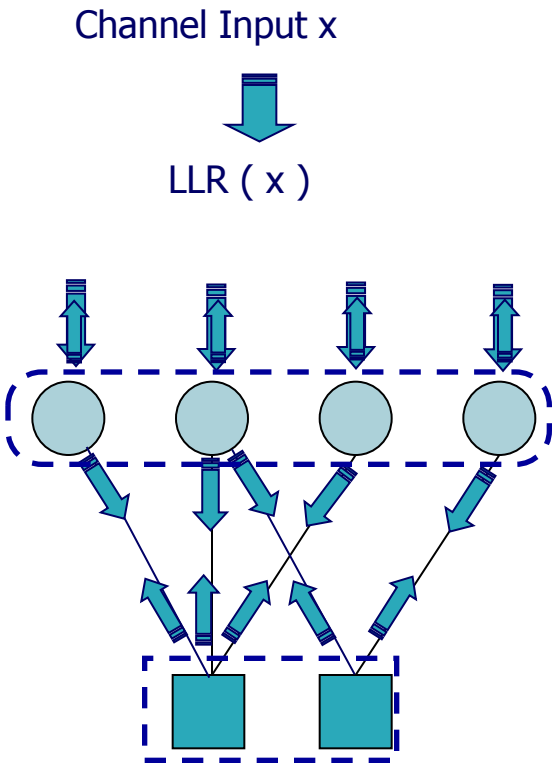
- Sign(LLR) = Hard decision

$$if\ (LLR_{x_0} \geq 0) \Rightarrow X_0 = 0$$
$$if\ (LLR_{x_0} < 0) \Rightarrow X_0 = 1$$

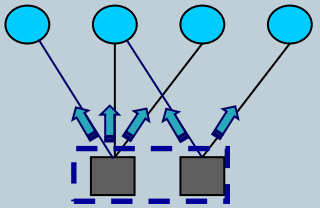
- |LLR| = Confidence factor

Belief Propagation algorithm by message passing

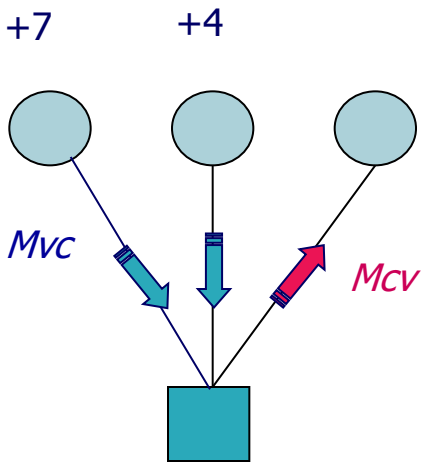
- initialization
- Iterative process
  - Check Node Update
  - Variable Node Update
- Hard Decision making



Check node update



Sub-optimal algorithm



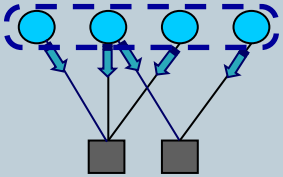
Min-Sum algorithm

\+ 4

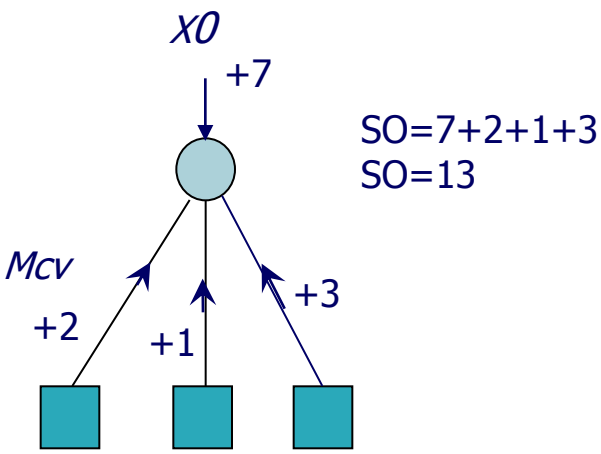
Normalized
Min-Sum algorithm

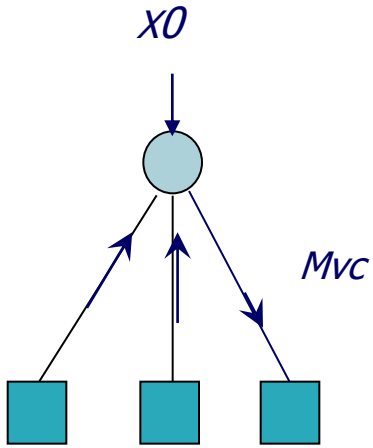
+4 x (0.75)=3

Variable node update



SO: Soft Output





$$M_{vc} = SO_0 - M_{cv}^{i-1}$$

M_{cv}=13-3=10

What is a NB-LDPC?

It is an LDPC… except that parity check equations are done on a Galois Field GF(q=2^m) of cardinality q>2.

	Binary LDPC	Non Binary LDPC
Codeword	$C = \left[c_i \in \{0,1\}; 0 \le i \le 5 \right]$	$C = \left[c_i \in GF\left(2^m\right); 0 \le i \le 5 \right]$
H	$\begin{bmatrix} 1 & 1 & 0 & 1 & 0 & 0 \\ 1 & 1 & 1 & 0 & 0 & 1 \\ 0 & 1 & 1 & 0 & 1 & 0 \end{bmatrix}$	$\begin{bmatrix} h_{00} & h_{01} & 0 & h_{03} & 0 & 0 \\ h_{10} & h_{11} & h_{12} & 0 & 0 & h_{15} \\ 0 & h_{21} & h_{22} & 0 & h_{24} & 0 \end{bmatrix}_{\substack{h_{ij} \in GF(2^m) \\ 0 \le i \le 2 \\ 0 \le j \le 5}}$
Parity-check equation	$c_0 + c_1 + c_3 = 0$	$h_{00} \cdot c_0 + h_{01} \cdot c_1 + h_{03} \cdot c_3 = 0$

Lab-STICC

cnrs

What is a Galois field?

A Galois Field has a Galois Field structure, i.e.:

addition: (GF($q=2^m$),+)

multiplication: (GF($q=2^m$), x)

…and all associated nice properties

By convention GF($q=2_m$) is represented by $\{0, \alpha^0, \alpha^1, \ldots \alpha^{q-2}\}$

GF($q=2^m$) have a binary representation

Operation in GF(8)

Binary representation:

GF(8)	bin
0	000
α^0	100
α^1	010
α^2	001
α^3	110
α^4	011
α^5	111
α^6	101

Addition:

$x = (x_1x_2x_3)$ and $y = (y_1y_2y_3) \in GF(8)$

$x + y = (x_1\ x_2\ x_3)$ XOR $(y_1\ y_2\ y_3)$

Example:

$\alpha^2 + \alpha^5 = (001)$ XOR $(111) = (110) = \alpha^3$

Multiplication:

$0 \times \alpha^i = 0$

$\alpha^i \times \alpha^j = \alpha^{(i+j) \bmod (q-1)}$

Example:

$\alpha^3 \times \alpha^5 = \alpha^{(3+5) \bmod 7} = \alpha^1$

Lab-STICC cnrs

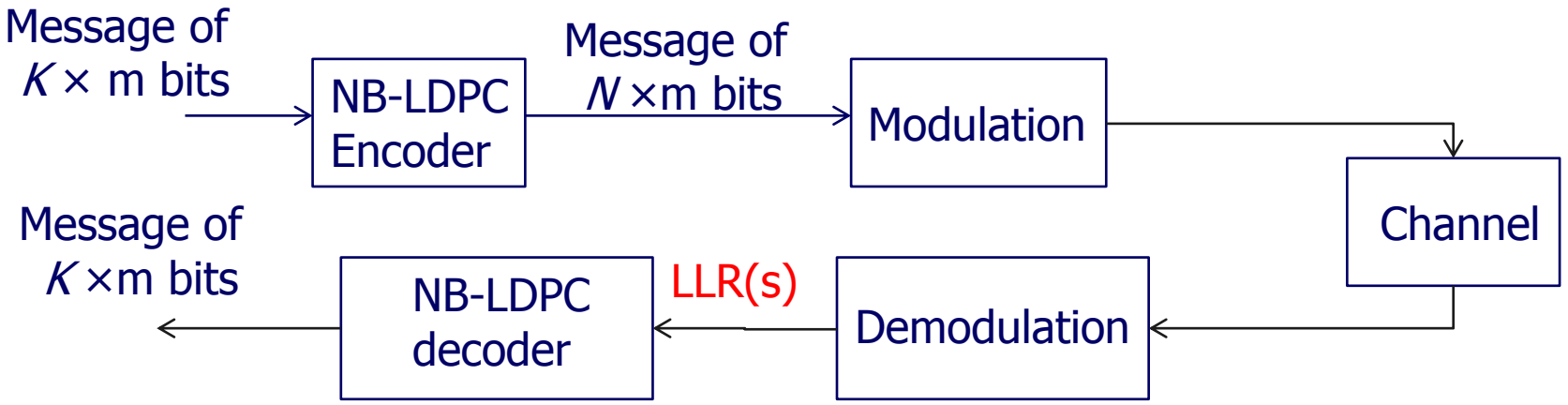
OUTLINE

1) Introduction
   - LDPC
   - NB-LDPC
   - Galois Field

2) Decoding NB-LDPC

3) What is the pro and cons of NB-LDPC?

Representation of intrinsic information



Binary:

(P(b=0), P(b=1))
→ LLR = ln(P(b=1)/P(b=0)) = ln(P(b=1))-ln(P(b=0))

NB-Binary:

P_s = (P(s=0), P(s=α^0), P(s=α^1), …, P(s=α^{q-2}))

In log domain:

LLR_s = -ln(P_s) + Cst with Cst = ln(arg max(Ps)).

Lab-STICC

cnrs

Representation of intrinsic information

NB-Binary:

$P_s = (P(s=0), P(s=\alpha^0), P(s=\alpha^1), \ldots, P(s=\alpha^{q-2}))$

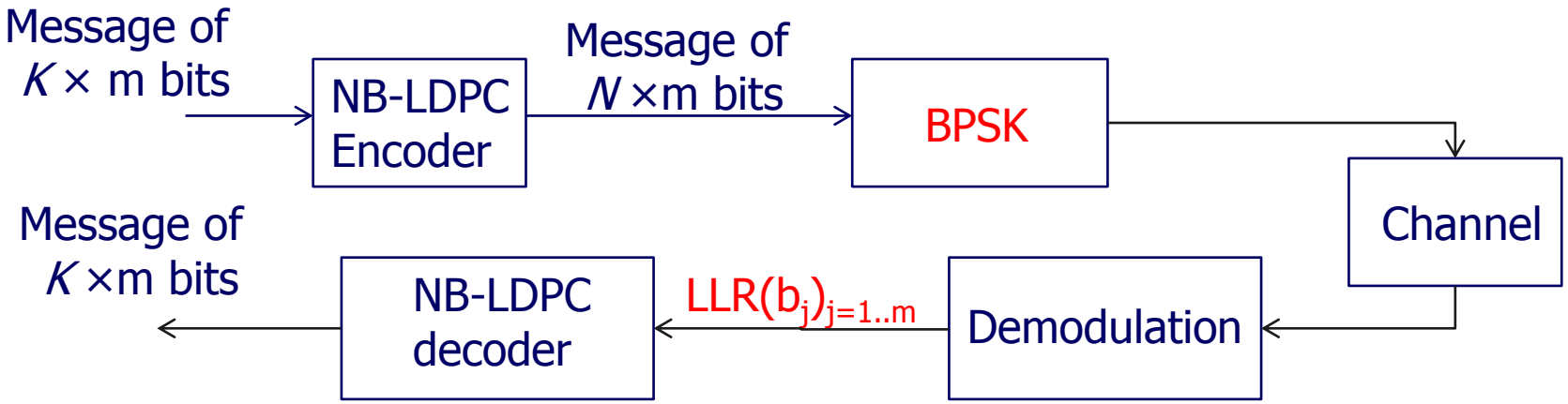
In log domain:

$LLR_s = -\ln(P_s) + Cst$ with $Cst = \ln(\arg\max(Ps))$.

Example on GF(8):

GF	0	α^0	α^1	α^2	α^3	α^4	α^5	α^6
P_s	0.1	0.85	10^{-3}	10^{-7}	10^{-10}	0.05	10^{-10}	10^{-10}
$-\ln(P_s)$	2.3	0.2	6.9	16.1	23.0	3.0	23.0	23.0
LLR_s	2.1	0	6.7	15.9	22.8	2.8	22.8	22.8

LLR computation for BPSK



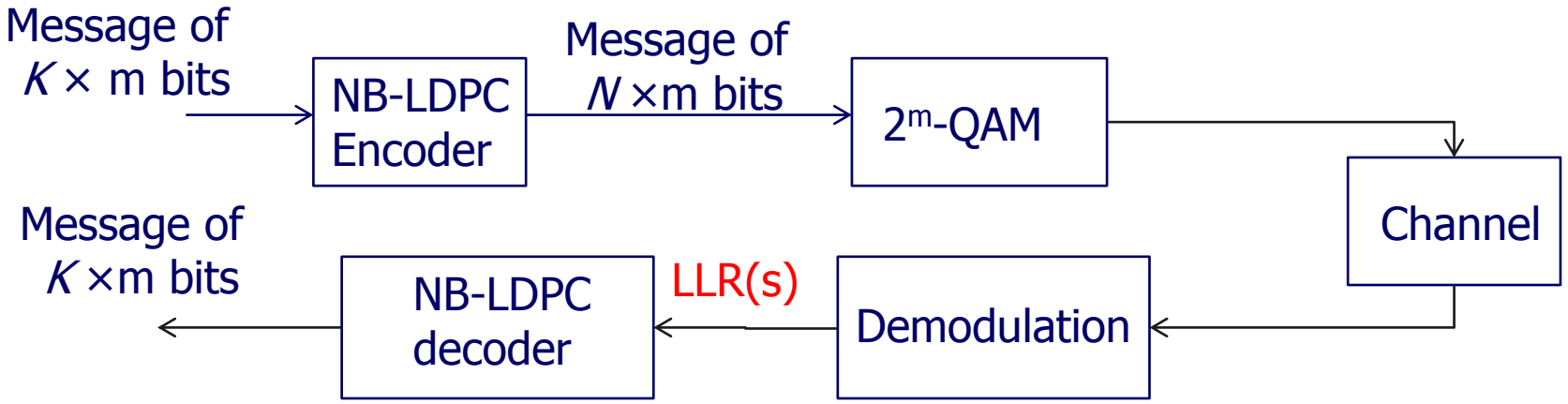
$$LLR(s=\alpha^i)=\sum_{j=0}^{q-1}((\alpha^i(j))\oplus HD(b_j))\times\left|LLR(b_j)\right|$$

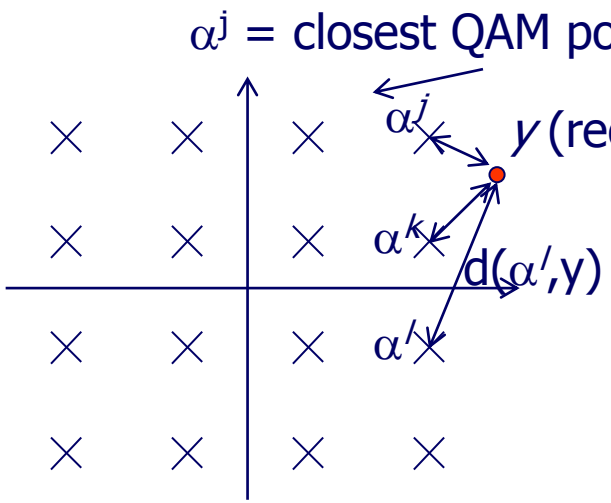
Example:

$LLR(b_1)=2$; $LLR(b_0)=-4$ => Hard Decision : (1,0) => α^1

LLR(s=0) = 2	0 = (0,0)
LLR(s=α^0) = 2 + 4 = 6	α^0 = (0,1)
LLR(s=α^1) = 0	α^1 = (1,0)
LLR(s=α^2) = 4	α^2 = (1,1)

LLR computation for 2^m-QAM using Coded Modulation





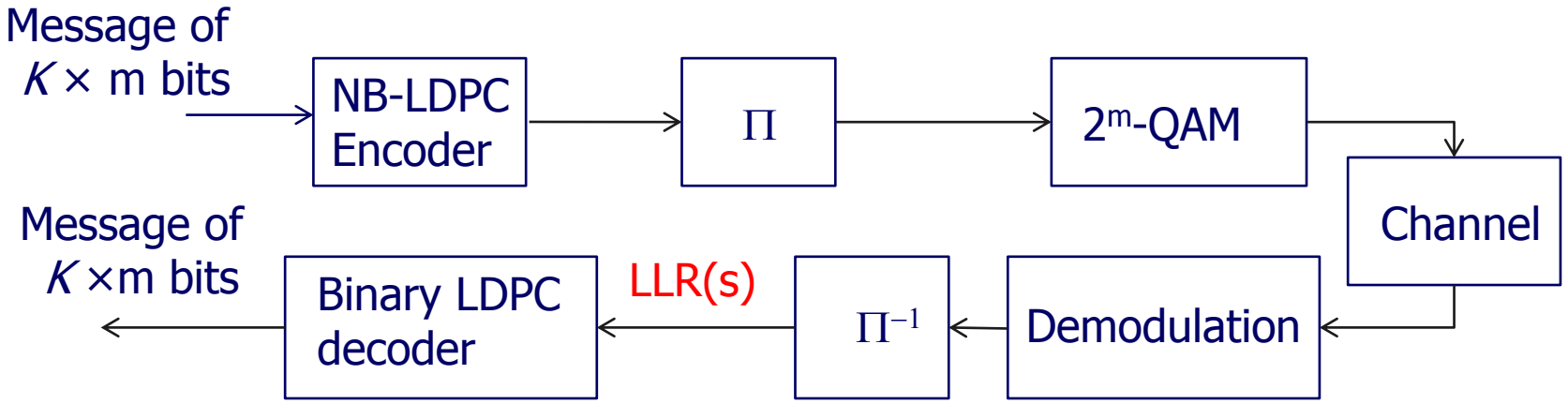
$$LLR(s=\alpha^j) = 0$$

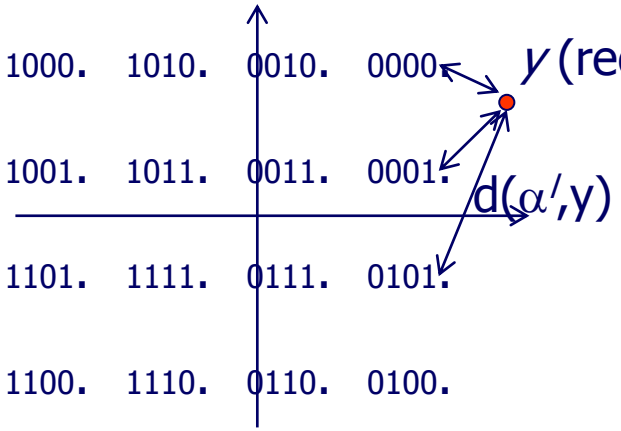
$$LLR(s=\alpha^k) = (d(y,\alpha^k)^2 - d(y,\alpha^j)^2)\times 2/\sigma^2$$

Lab-STICC

cnrs

LLR computation for 2^m-QAM using Bit-Interleaved Coded Modulation





$$LLR(b_j) = \log\left(\frac{\sum_{x_0 \in GF_j^0} p(x_0)}{\sum_{x_1 \in GF_j^1} p(x_1)}\right)$$

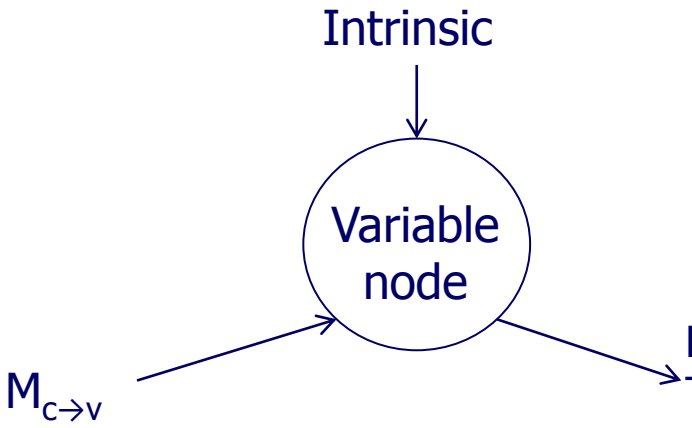
$$LLR(s = \alpha^i) = \sum_{j=0}^{q-1} ((\alpha^i(j)) \oplus HD(b_j)) \times \left|LLR(b_j)\right|$$

Bit marginalization lead to loss of information

Lab-STICC

cnrs

Variable node processing



$M_{v\to c}$ = Adition term by term addition of I and $M_{c\to v}$
Then normalization.

Example in GF(4):

Intrinsic

GF	LLR
0	3
α^0	17
α^1	0
α^2	9

+

$M_{c\to v}$

GF	LLR
0	8
α^0	15
α^1	7
α^2	8

=

$M_{v\to c}$

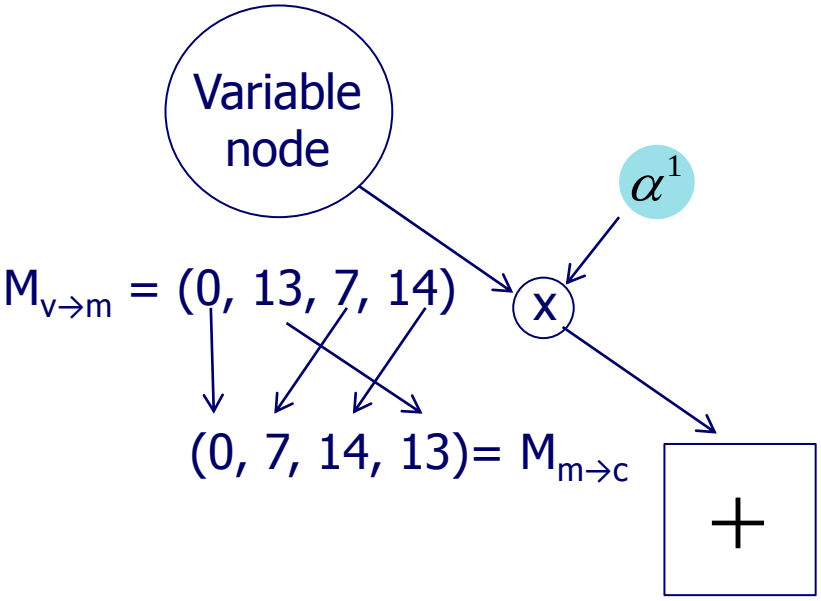
GF	LLR
0	11
α^0	32
α^1	7
α^2	17

-

$M_{v\to c}$

GF	LLR
0	4
α^0	25
α^1	0
α^2	10

Edge processing

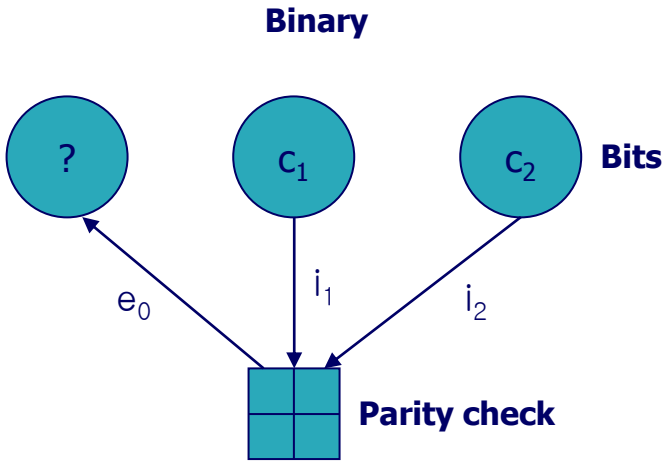


$\alpha^1(0, \alpha^0, \alpha^1, \alpha^2)$

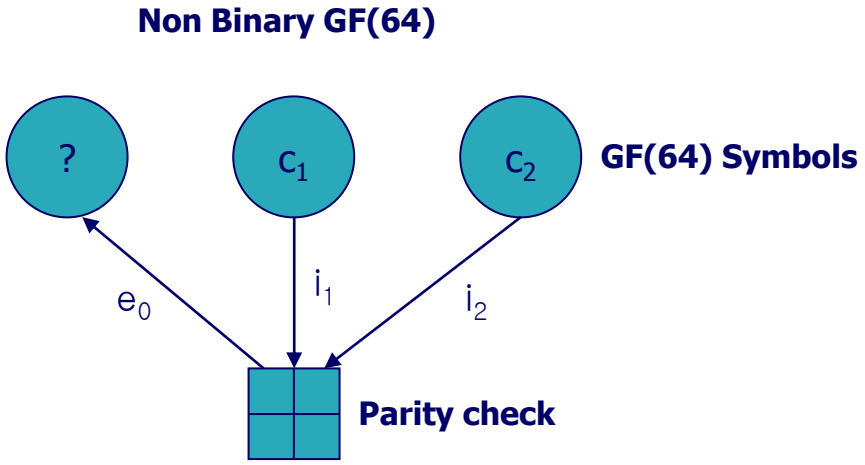
$= (0, \alpha^1, \alpha^2, \alpha^0)$

The effect of edge multiplication is just a permutation of the LLR

Check node processing



4 input configurations to evaluate

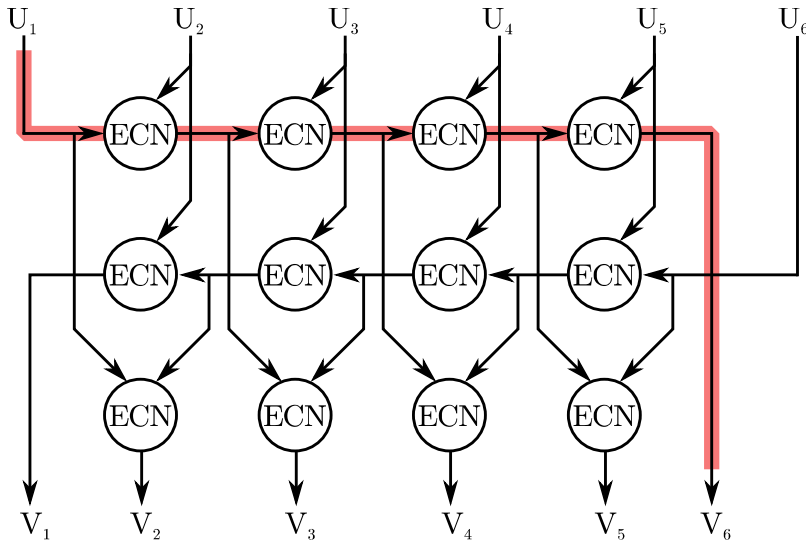


64x64=4048 input configurations to evaluate

dc=4 → 64^3 input configurations to evaluate

dc=12→64^{12} input configuration to evaluate

Check node processing



Forward Backward (FWBW) processing is the state-of-the-art check node algorithm.

With a divide and conquer approach using Elementary Check Nodes (ECN) the most reliable messages for each outgoing edge are computed. Each ECN considers two GF(q) vectors. The intermediate results are combined in a smart way to generate the output vectors.

The FWBW scheme allows for small hardware implementations but suffers from low throughput and high latency.

Extended Min-Sum algorithm

ECN Processing: $$LLR_E(\alpha^k) \approx \underset{\alpha^i,\alpha^j \in GF(q)^2 / \alpha^i+\alpha^j=\alpha^k}{MIN} LLR_U(\alpha^i) + LLR_V(\alpha^j)$$

The higher LLR values of U and V are rarely, if never, used in the output.

Idea: keep only the most n_m smallest LLR sorted in ascending order to simplify the ECN computation.

Examples: LLR_U = (3; 0; 12; 6) => ((0, α^0), (3, 0))
LLR_V = (18; 7; 9; 0) => ((0, α^2), (7, α^0))

U\V	18;0	7; α^0	9; α^1	0; α^2
3;0	21;0	10;α^0	12;α^1	3;α^2
0; α^0	18;α^0	7;0	9;α^2	0;α^1
12;α^1	30;α^1	19;α^2	21;0	12;α^0
6; α^2	24,α^2	13;α^1	15;α^0	6;0

U\V	0, α^2	7, α^0
0, α^0	0; α^1	7;0
3, 0	3;α^2	10;α^0

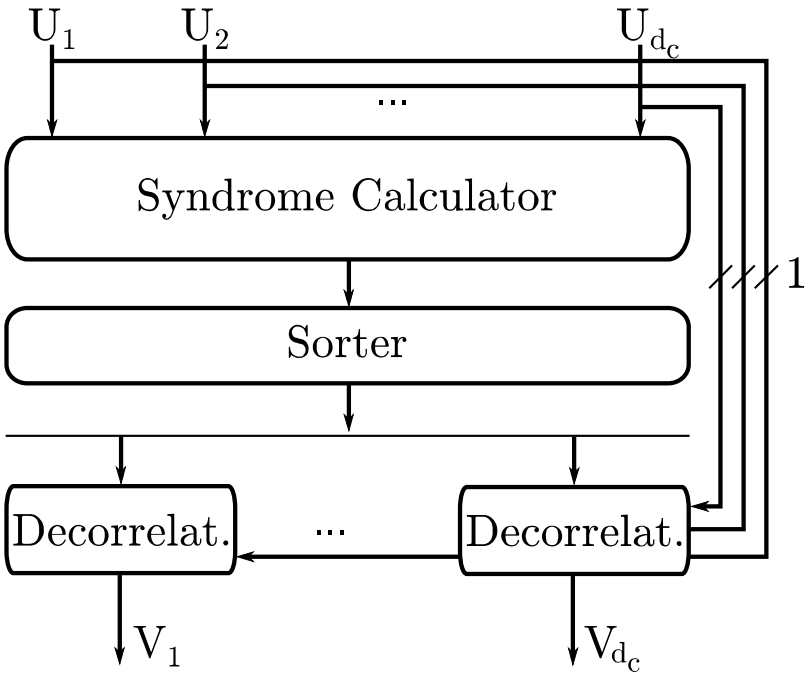
Extract the n_m smallest values among the n_m^2 values

Complexity: $2q^2$ => 4 $\times n_m$ additions (L-Bubble algorithm)

Lab-STICC

cnrs

Syndrome based decoder



Syndrome based CN processing:

1. Calculate most probable syndromes
(Syndrome = sum of one element (GF and LDR) per edge)
2. Decorrelate syndromes for each edge
3. Generate outputs

+ No separate handling of the edges
+ Possible parallel computation of all messages
+ Allows for low latency processing

[1] P. Schläfer, N. Wehn, M. Alles, T. Lehnigk-Emden and E. Boutillon, "Syndrome based check node processing of high order NB-LDPC decoders," *Telecommunications (ICT), 2015 22nd International Conference on*, Sydney, NSW, 2015, pp. 156-162.

OUTLINE

1) Introduction
   - LDPC
   - NB-LDPC
   - Galois Field

2) Decoding NB-LDPC

3) What are the pro and cons of NB-LDPC ?

Lab-STICC

cnrs

Cons

- Changing from LDPC to NB-LDPC is a revolution (no more compatibility).

-Complexity

Pros…

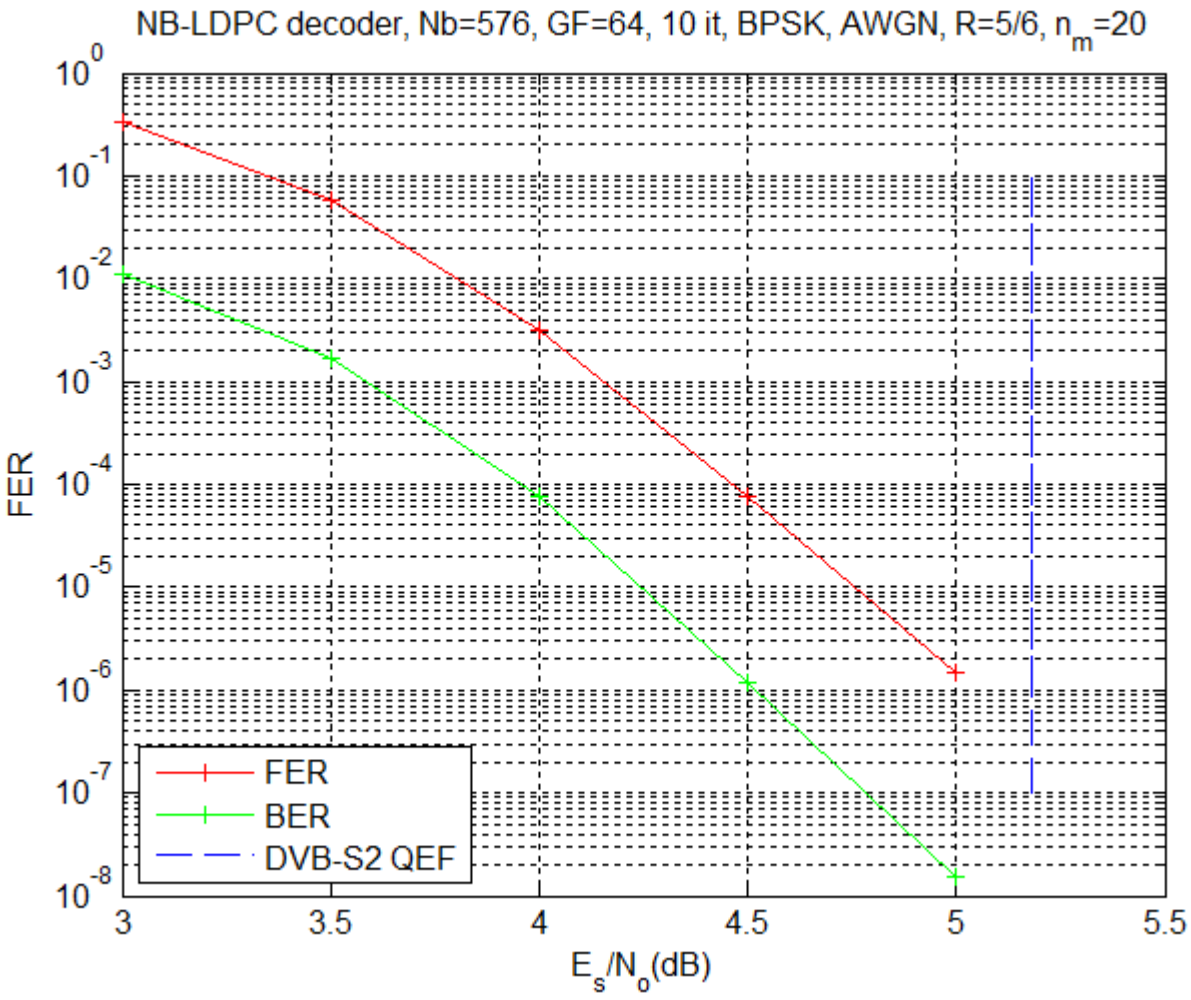
- Under BP decoding, NB-LDPC has significant better performance than LDPC for low code size and low code rate.

-Low error floor

-No need of bit marginalization during the demodulation (high spectral efficiency).

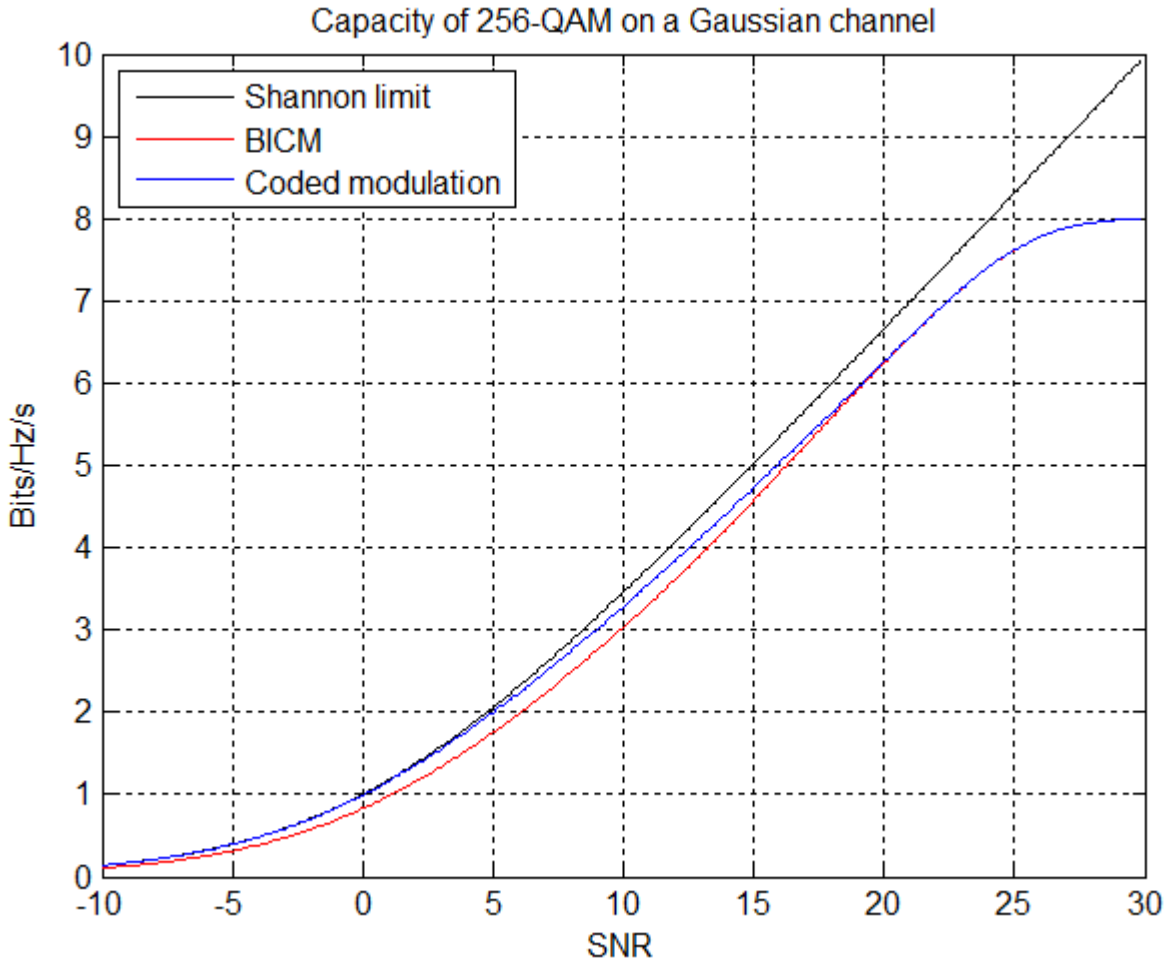
- Higher mutual information of Coded Modulation VS BICM (in SISO, SIMO, MIMO, … channel).

Good performance for short length

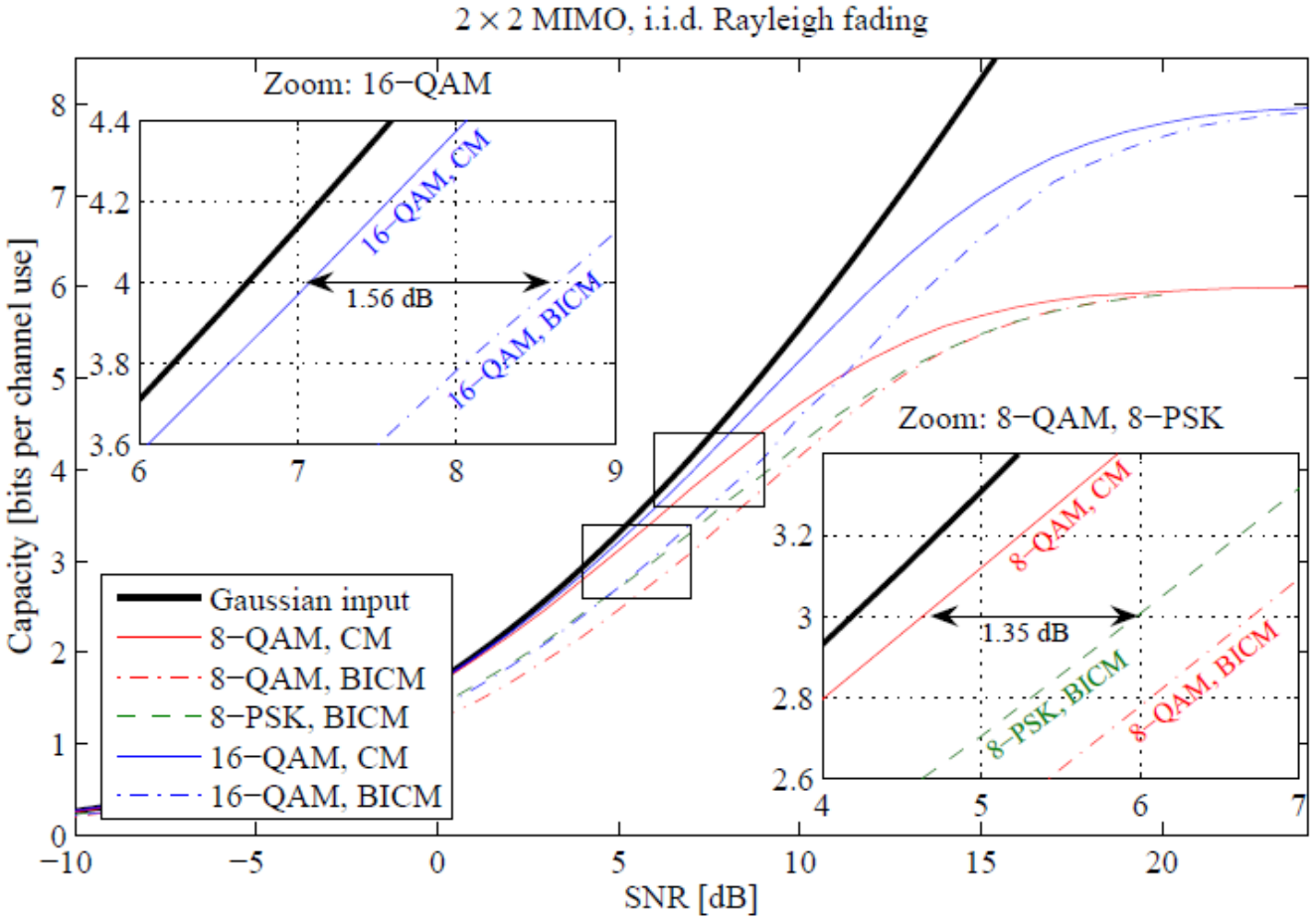


More simulation results at http://www-labsticc.univ-ubs.fr/nb_ldpc/

Higher capacity than BICM



Higher capacity than BICM for MIMO channel



Taken from: D. Declercq, IEEE SSC SCV Tutorial, Santa Clara, October 21st, 2010

Lab-STICC

cnrs

Conclusion

NB-LDPC is not yet a mature technology : great potential improvement.

- Under BP decoding, NB-LDPC has significant better performance than LDPC for low code size and low code rate.

-No need of bit marginalization during the demodulation (high spectral efficiency).

- Higher mutual information of Coded Modulation VS BICM (in SISO, SIMO, MIMO, … channel).

Conclusion

- Thank you for your attention

- Questions are welcome