ANR-19-CE25-0013-01

Quasi Cyclic Short Pakect

LoRa-Alliance
22/2/2022

Emmanuel Boutillon

With the help of Kassem Saeid, Camille Monière and the QCSP'consortium

About QCSP project

- Project funded by ANR (French Research Funding Agency).
- 4 years project started in October 2019.
- 5 academics partners and CEA, Orange Labs and Sequans.

Objective : contribute to the evolution of IoT networks

Bit bet : Proposition of a new waveform.

Network problem

- Y. Polyanskiy (MIT), P. Popovski (ERC)

"In a network, asynchronism shouldn't affect capacity."

Classical model of frame is inefficient

Should be replaced by

From space to earth

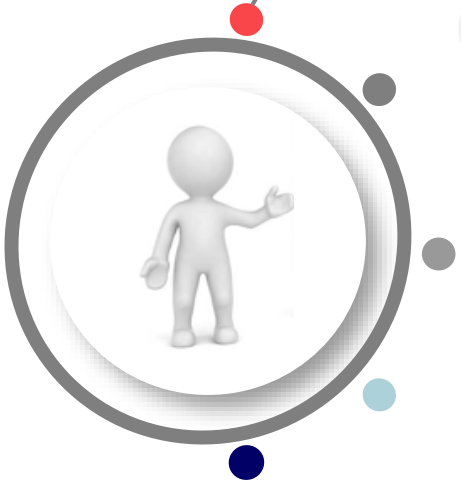
- **Cyclic-Code Shift Keying (CCSK) [1]** used in Quasi-Zenith Satellite system (Japanese GPS enhancement system).

- **Non-binary error correcting codes (NB-ECC)** used in BeiDou (Chinese GPS-like system) [2].

QCSP Approach: CCSK and NB-code association

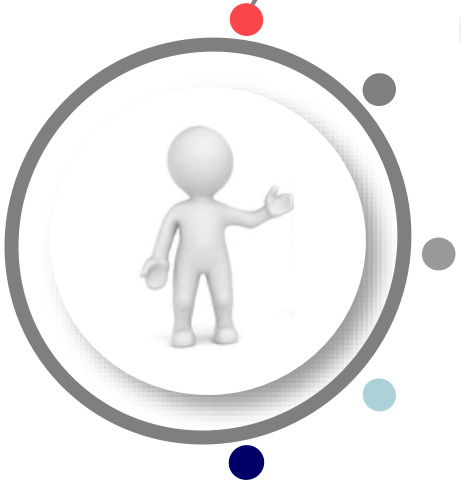
[1]: G. M. Dillard et al. “Cyclic code shift keying: a low probability of intercept communication technique”. In: *IEEE Transactions on Aerospace and Electronic Systems* 39.3 (2003), pp. 786–798.
[2]: China Satellite Navigation Office, *BeiDou Navigation Satellite System, Signal In Space,* *Interface Control Document, Open Service Signals*

Outline



- QCSP system model
- Receiver processing
- GNU Radio implementation
- Toward the multi-user context
- Conclusion and perspectives

Outline



- **QCSP system model**
- Receiver processing
- GNU Radio implementation
- Toward the multi-user context
- Conclusion and perspectives

Non-Binary Code

- Galois Filed of order q (GF(q)) is a finite field that contains q elements. All the operations (+, *, -, /) are performed "modulo q", where q is a power of 2, i.e, $q = 2^m$.

Example: $m = 3, q = 2^m = 8$, GF($q = 8$):

GF element	Binary represent.	Integer represent.
0	000	0
α^0	001	1
α^1	010	2
α^2	100	4
α^3	011	3
α^4	110	6
α^5	111	7
α^6	101	5

Addition example:

$X = (x_0x_1x_2)$, $Y = (y_0y_1y_2) \in$ GF(8) => $X \oplus Y = X$ XOR Y

Eg.: $\alpha^4 \oplus \alpha^1 = 110$ XOR $010 = 100 = \alpha^2$

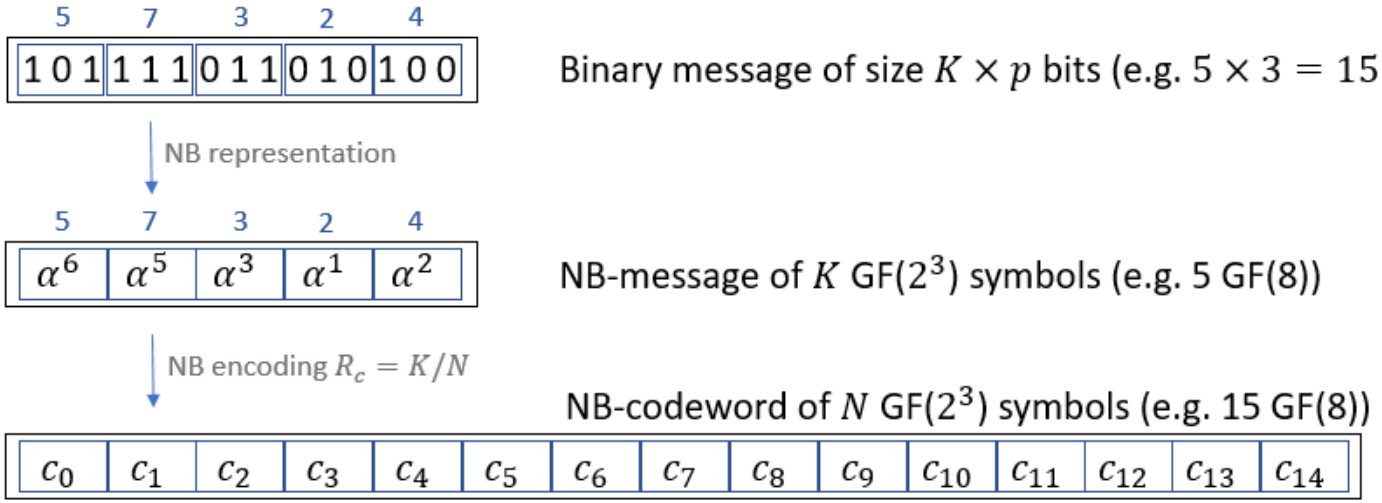
Multiplication example:

$0.\alpha^i = 0$ and $\alpha^i.\alpha^j = \alpha^{(i+j) \bmod (q-1)}$

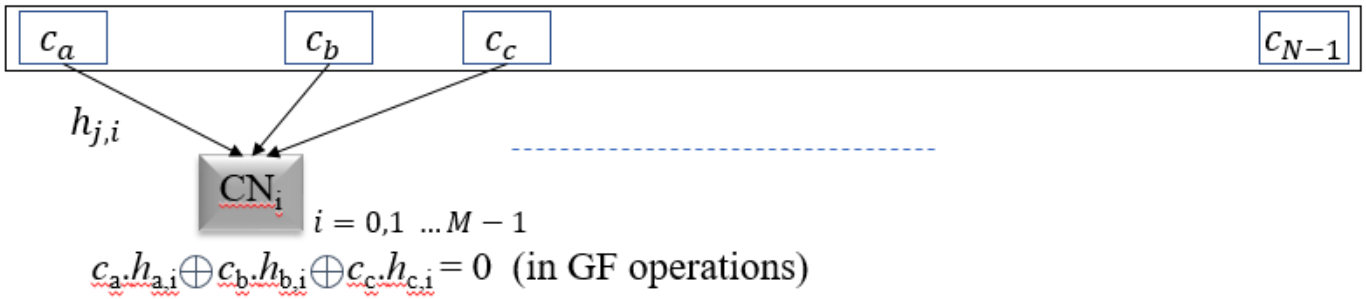
Eg.: $\alpha^4.\alpha^3 = \alpha^{7 \bmod (7)} = \alpha^0$

Example of NB-Code: NB-LDPC

NB Code, GF(2^3)



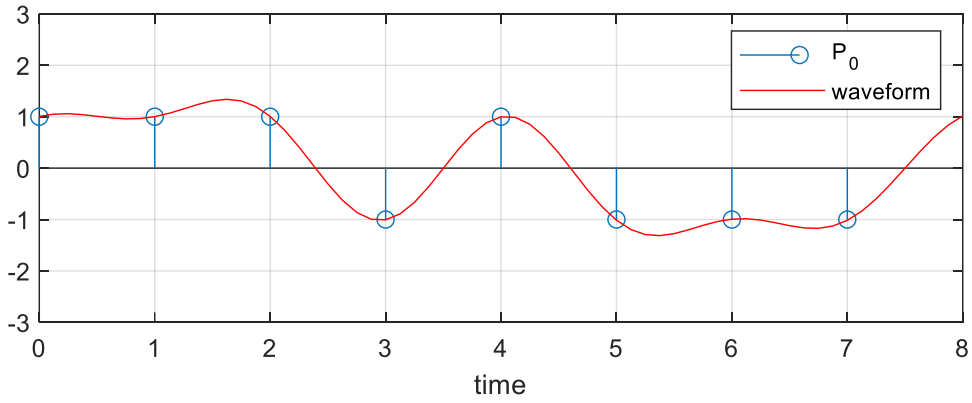
In a **NB-LDPC code**: a codeword exists if the set of GF parity checks are fulfilled:



Cyclic Code Shift Keying modulation

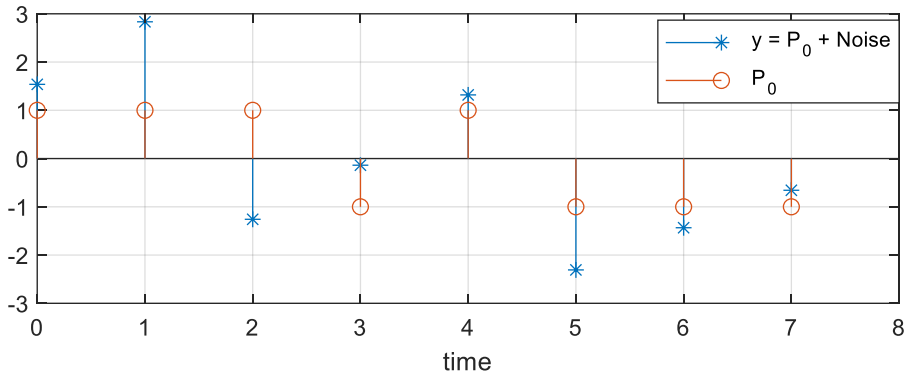
P_0 = 11101000 + BPSK modulation, roll-off factor 0.35, q=8

- CCSK modulation:
  - P_0 = 11101000
  - P_1 = 11010001
  - P_2 = 10100011
  - P_3 = 01000111
  - P_4 = 10001110
  - P_5 = 00011101
  - P_6 = 00111010
  - P_7 = 01110100



$P_a(k) = P_0(k-a)$, k = 0, 1, …, 7

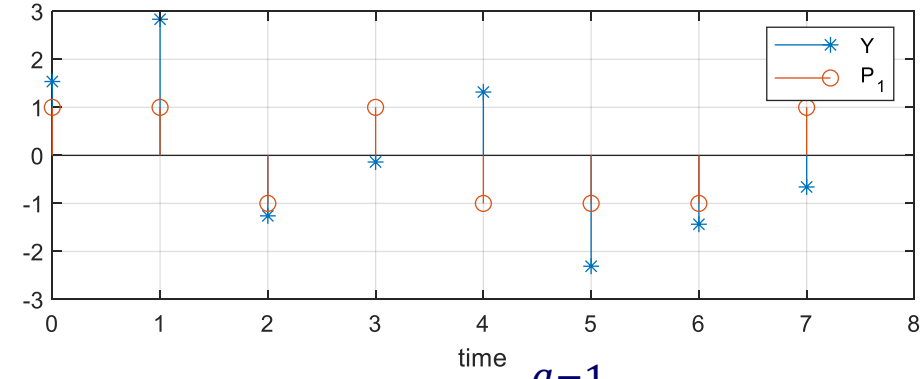
Demodulation of a CCSK frame



$$\log(\mathrm{P}(\mathrm{Y}|\mathrm{a})) = -\frac{d(P_a, y)^2}{2\sigma}$$

$$d(P_0, Y)^2 = Y^2 + {P_0}^2 + 2 < P_0, Y >$$

$$d(P_0, Y)^2 = 21.5 + 8 - 2 \times 9.0 = 11.5$$



$$d(P_1, Y)^2 = 21.5 + 8 - 2 \times 7.2 = 15.1$$

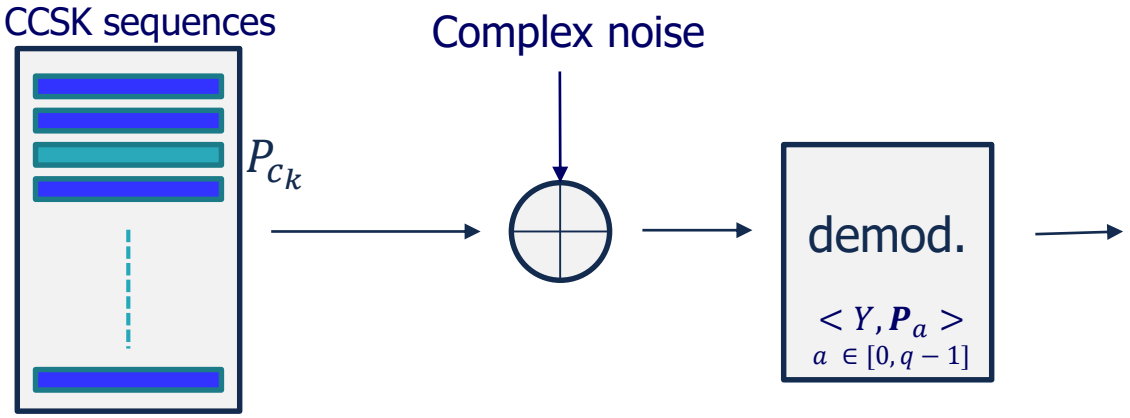
$$L(a) = < P_a, Y > = \sum_{k=0}^{q-1} P_0(k-a)Y(k)'$$

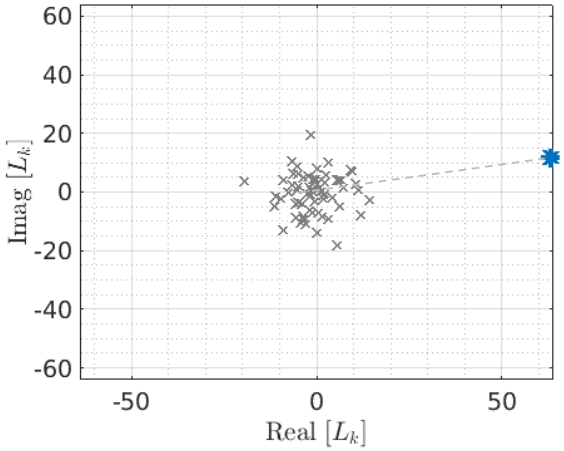
Circular convolution
=> product in frequency domain

$$L = \mathrm{IFFT}(\ \mathrm{FFT}(Y) \times \mathrm{FFT}(P_0))$$

Demodulation of CCSK frame in complex noise

Correlation between each of the received symbols Y and the q CCSK sequences.

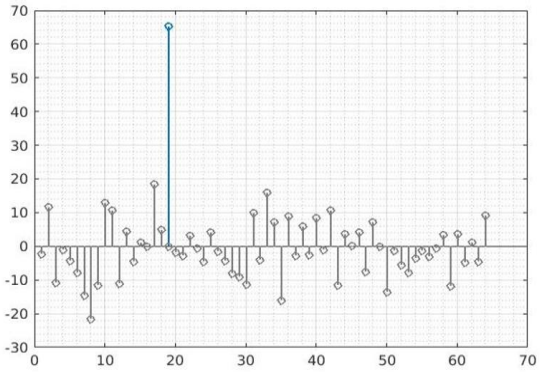




Y : noisy received sequence

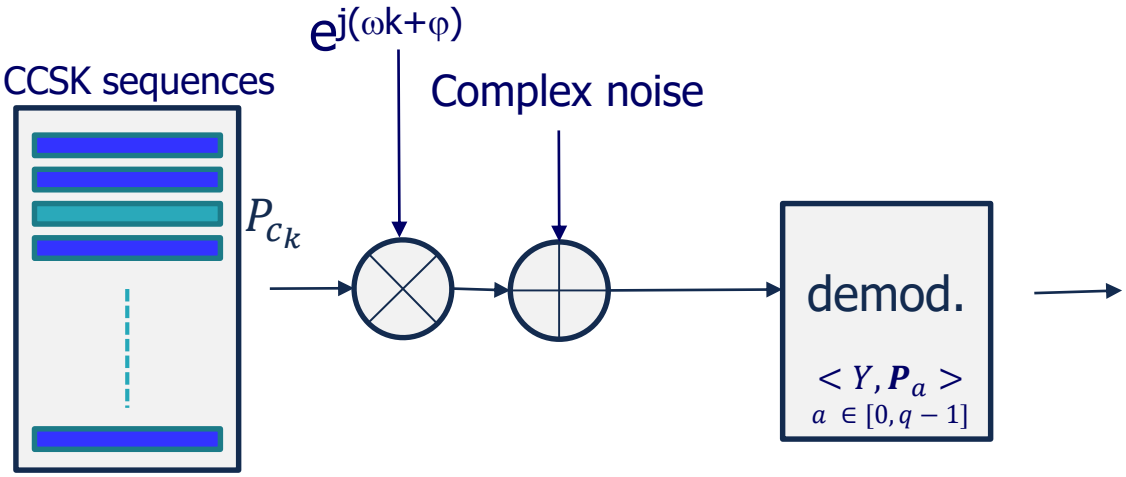
$\boldsymbol{L}$ = Log Likelihood Ratio

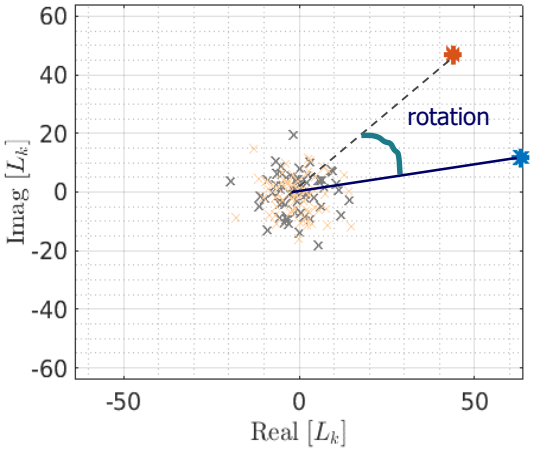
$$L(a) = Real(\log(\mathbb{P}(\boldsymbol{P}_a/Y))$$
$$\sim Real(< \boldsymbol{Y}, \boldsymbol{P}_a >)\ a = 0 \ldots q-1$$

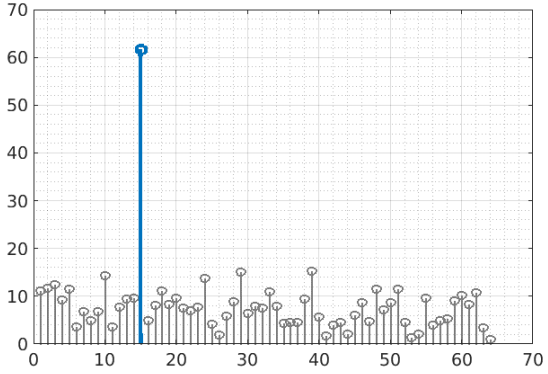


Demodulation with phase offset

Effect of Doppler of local oscillator mismatch







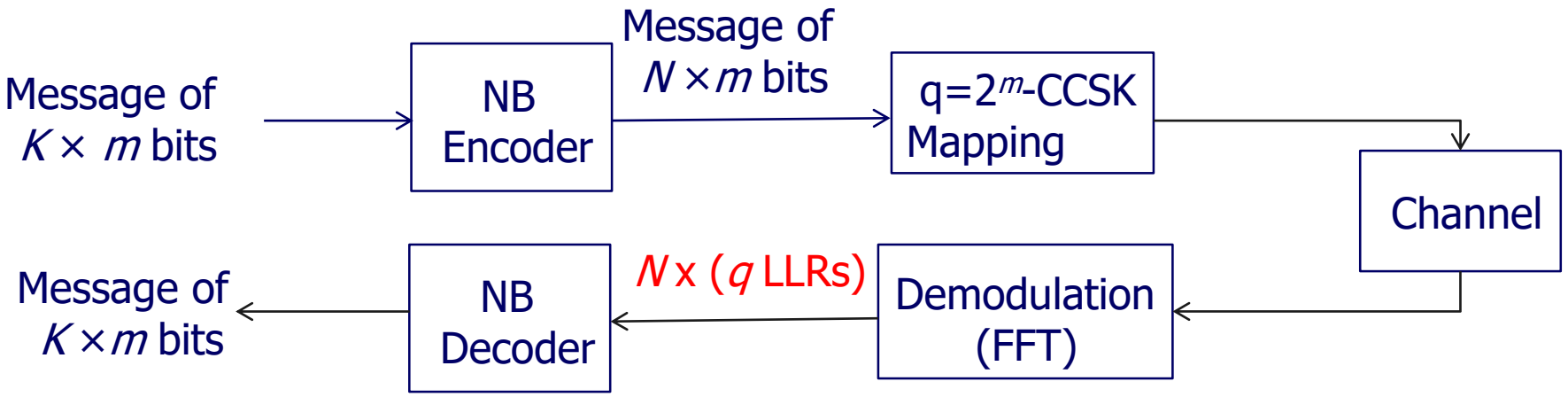
$\boldsymbol{Y}$: noisy received sequence

$\boldsymbol{L}$ = Log Likelihood Ratio

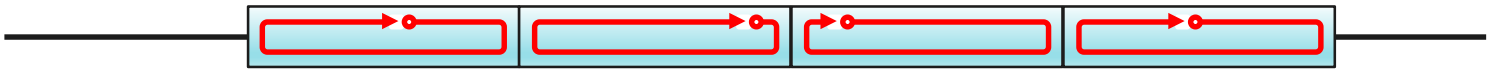
$L(a) = |\log(\mathbb{P}(\boldsymbol{P}_a/Y))| \sim |< Y, \boldsymbol{P}_a >|, a = 0 \ldots q-1$

With phase offset, non-coherent demodulation is required

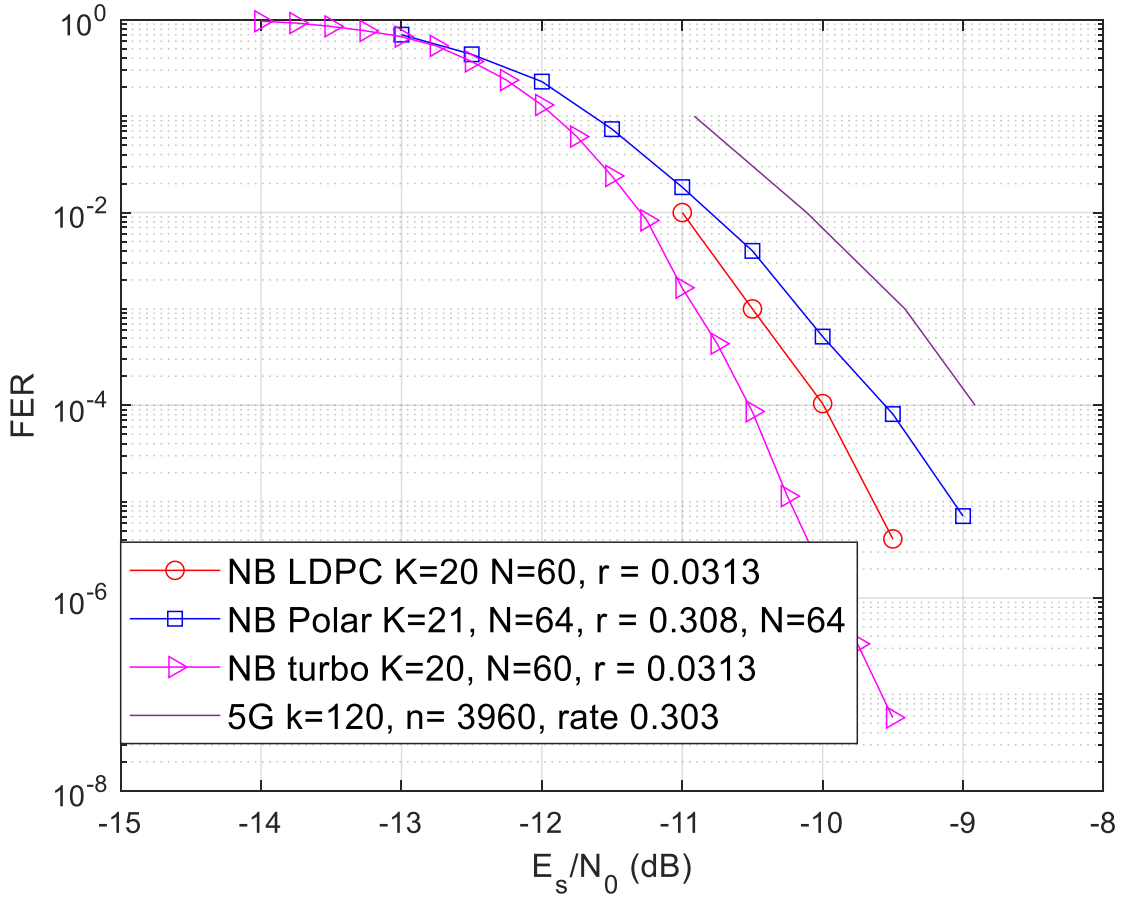
QCSP frame structure



- The frame is composed of N segments of CCSK sequence (or symbol)



Example of performance



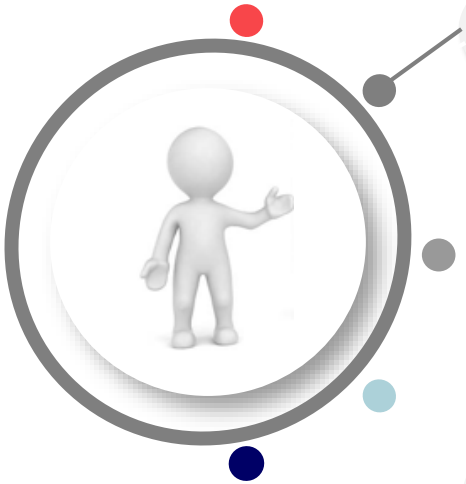
Perfect synchronisation.

QCSP:
NB code rate r ≈1/3
CCSK: m=6 bits, q = 64
=> rate 6/64.
Global rate ≈ 6/64x1/3=0.0313

5G:
rate 1/3 binary LDPC code + 11 repetitions.

Global rate: 1/33 = 0.0303

QCSP system model



Receiver algorithms

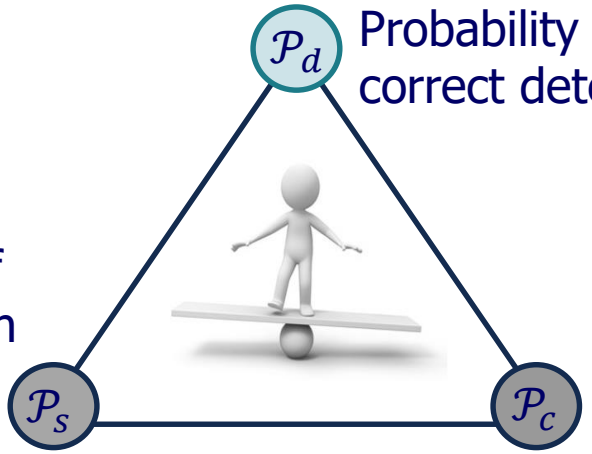
GNU Radio implementation

Toward the multi-user context

Conclusion and perspectives

Objective

Developing blind detection and self-synchronization algorithms for achieving correct preamble-less short packet reception at very low SNRs.



$\mathcal{P}_d$ Probability of correct detection

Joint overall probability:
$\mathcal{P} \approx \mathcal{P}_d \times \mathcal{P}_s \times \mathcal{P}_c$

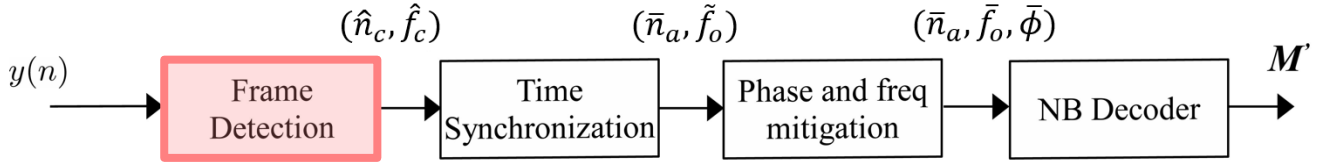
Probability of correct synchronization

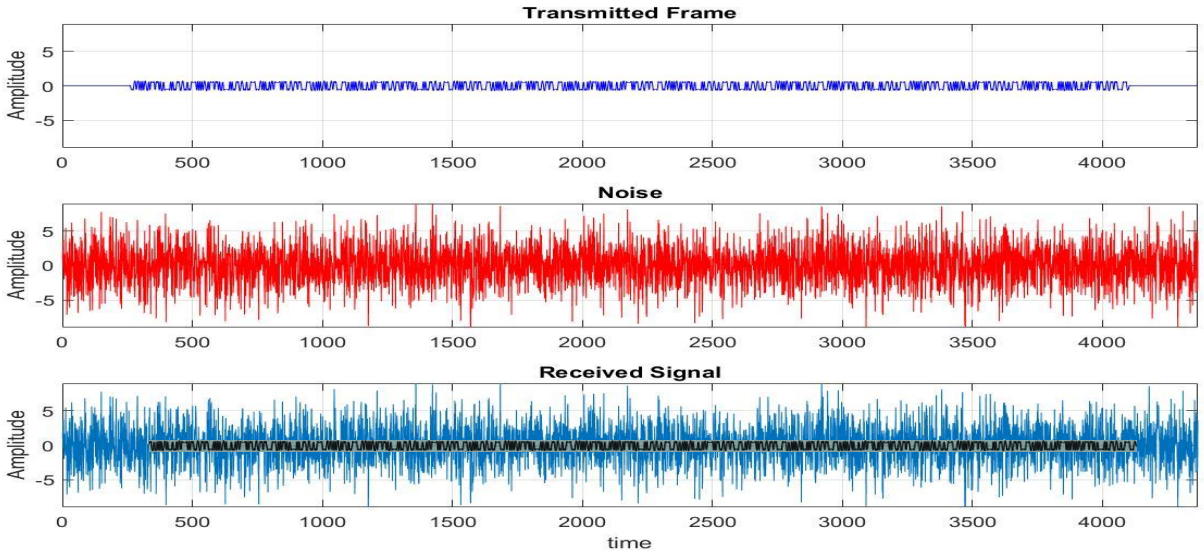
$\mathcal{P}_c$ Probability of correct decoding

➔ Aiming to maximize the overall probability is achieved by maximizing the weakest probability:

Max(min($\mathcal{P}_d, \mathcal{P}_s$, $\mathcal{P}_c$)).

Detection problem





- Find a reliable way to asses the presence of a CCSK frame or not.
- Coarse time and frequency offsets estimation $(\hat{n}_c, \hat{f}_c)$.

Toy example

P_0 = ABAABB

Message = {0,0,0} (N = 3)

{ABAABB, ABAABB, ABAABB}

0 0 0

Message = {3,0,5} (N = 3)

{ABBABA, ABAABB, BABAAB}

3 0 5

Toy example

Y_0

XXXXABBABAABAABBBABAABXXX

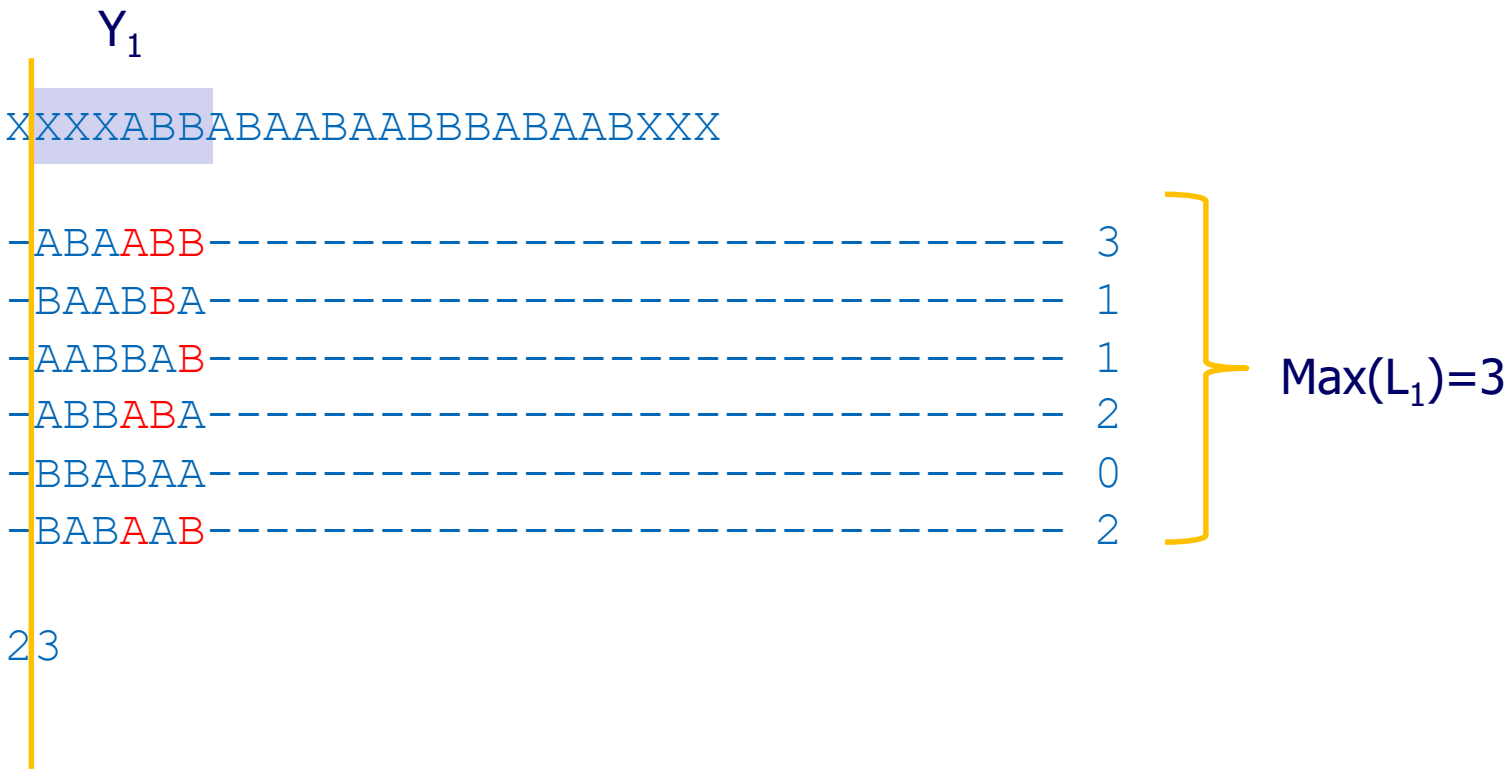
$L_0 = \{<Y_0, P_a>\}_{a = 0, 1,\ldots,5}$

ABAABB---------- 1
BAABBA---------- 0
AABBAB---------- 2
ABBABA---------- 0
BBABAA---------- 1
BABAAB---------- 2

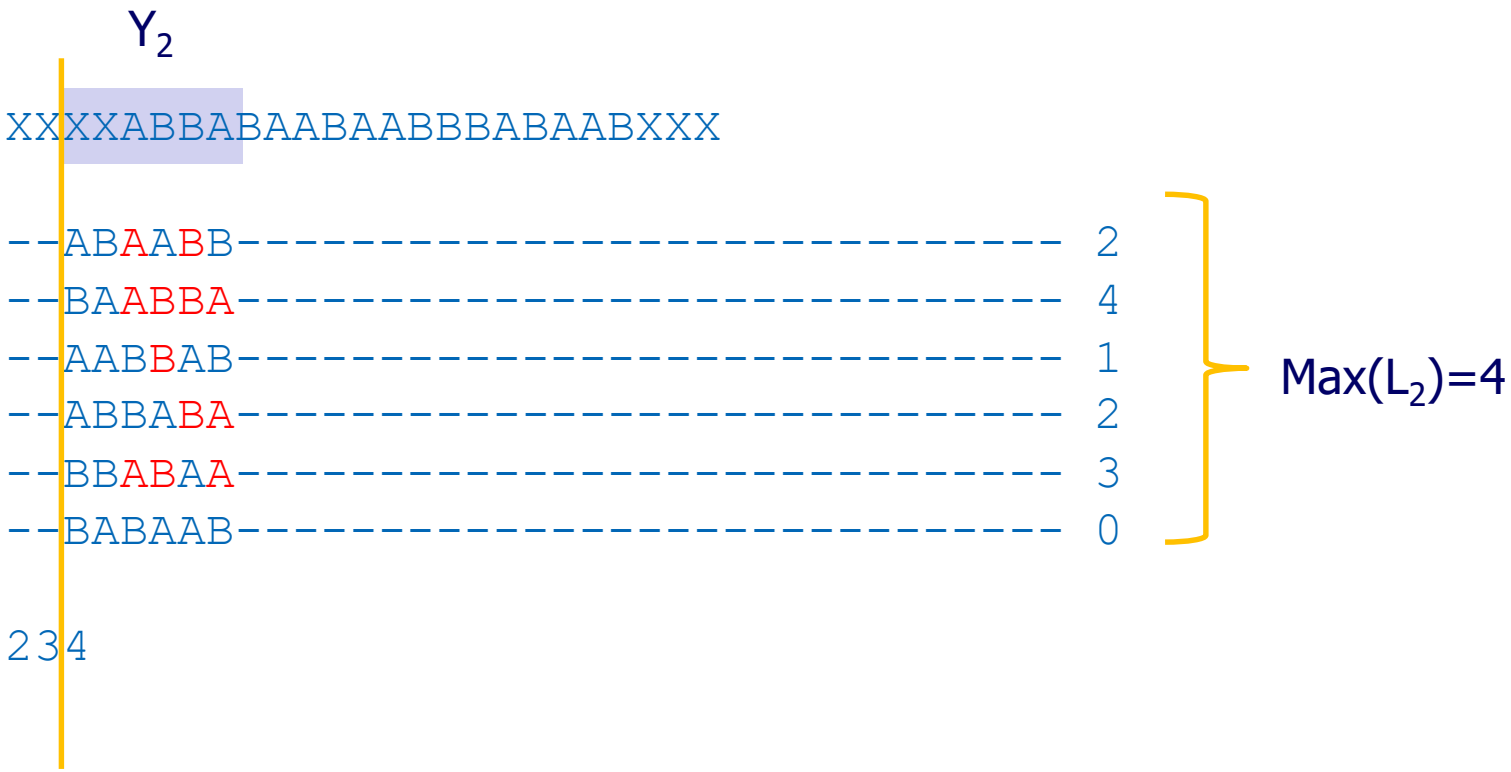
$Max(L_0)=2$

2

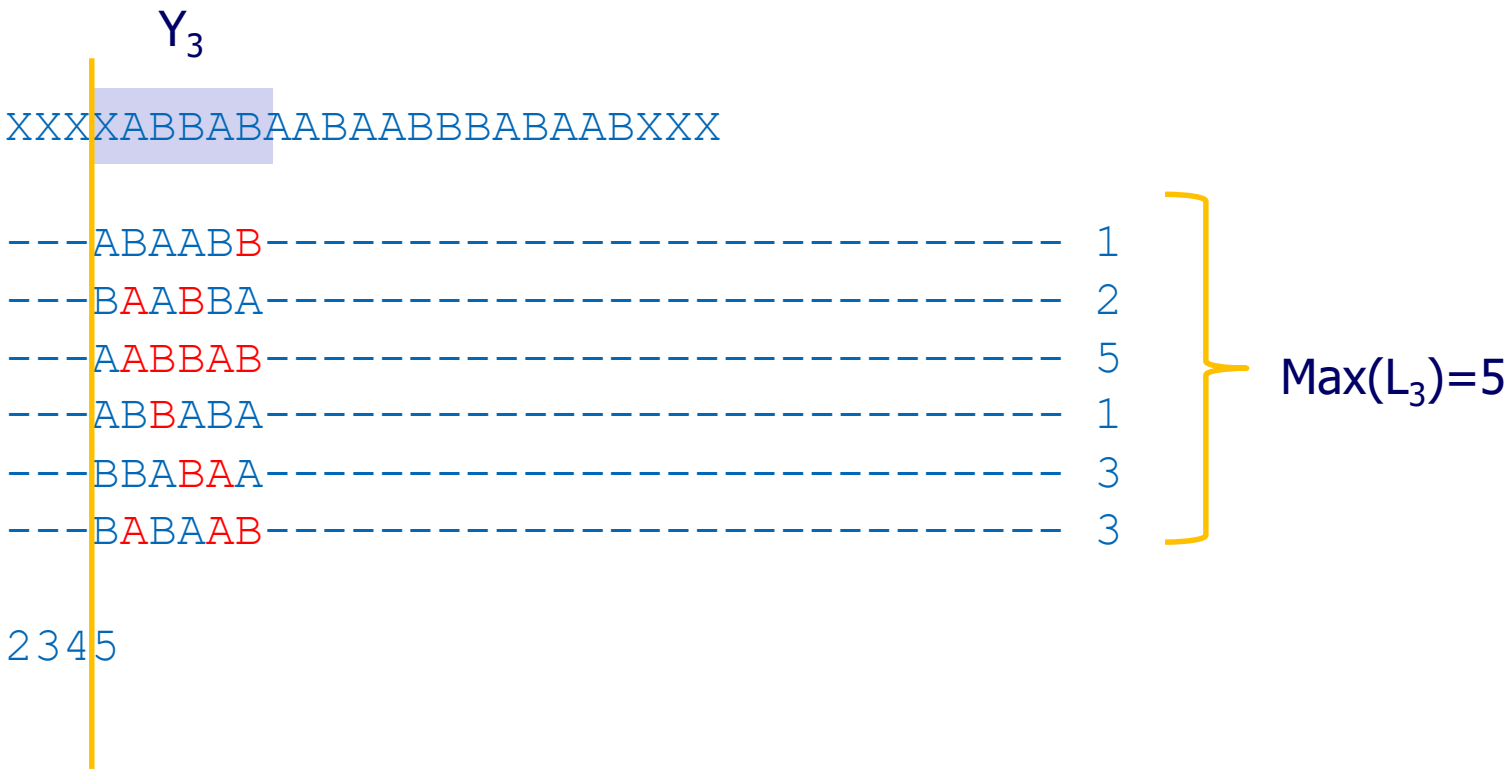
Toy example


Y_1
XXXXABBABAABAABBBABAABXXX
-ABAABB---------- 3
-BAABBA---------- 1
-AABBAB---------- 1
-ABBABA---------- 2
-BBABAA---------- 0
-BABAAB---------- 2
23
$Max(L_1)=3$

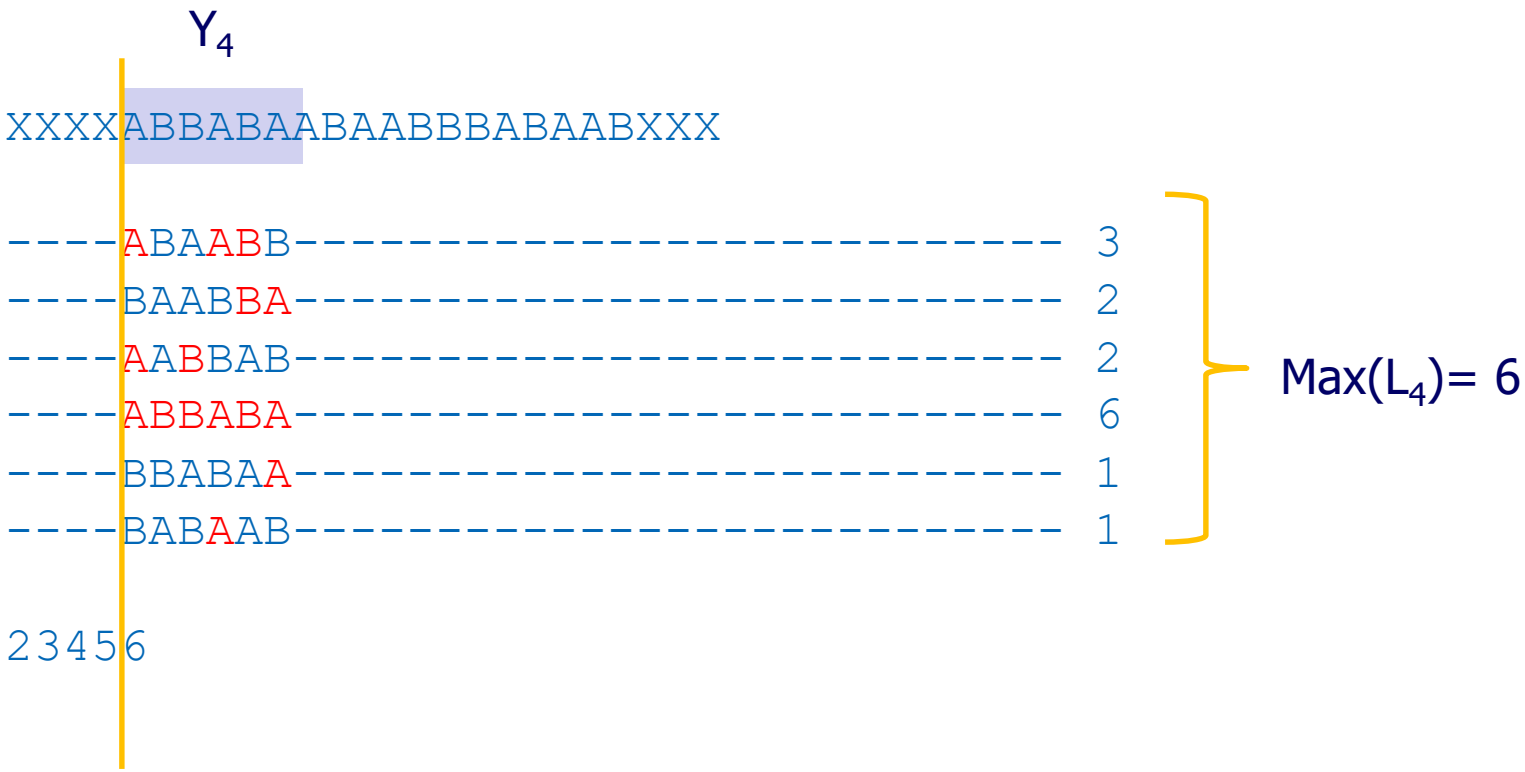
Toy example


Y_2
XXXXABBABAABAABBBABAABXXX
--ABAABB---------- 2
--BAABBA---------- 4
--AABBAB---------- 1
--ABBABA---------- 2
--BBABAA---------- 3
--BABAAB---------- 0
2 3 4
Max(L_2)=4

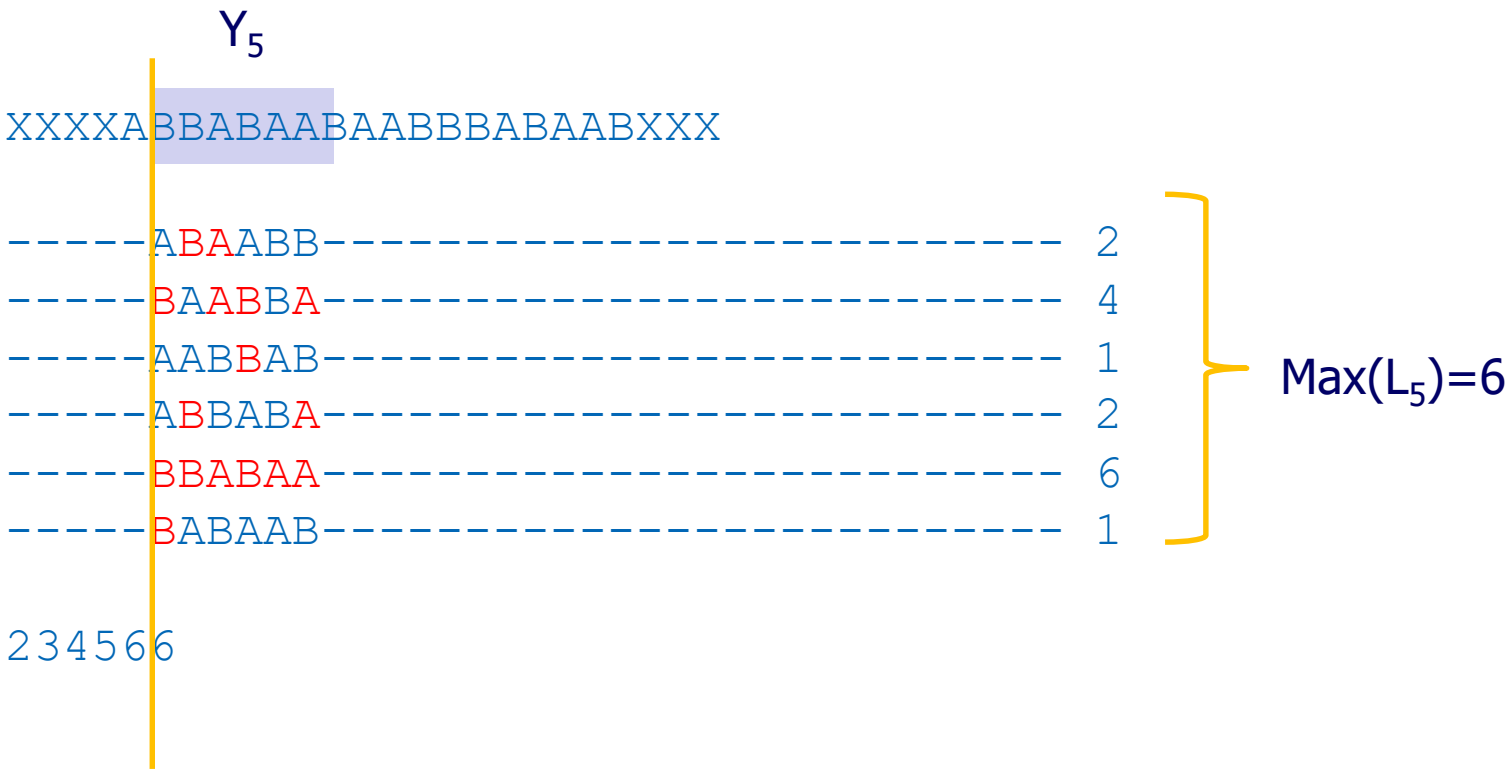
Toy example


Y_3
XXXXABBABAABAABBBABAABXXX
---ABAABB---------- 1
---BAABBA---------- 2
---AABBAB---------- 5
---ABBABA---------- 1
---BBABAA---------- 3
---BABAAB---------- 3
2345
Max(L_3)=5

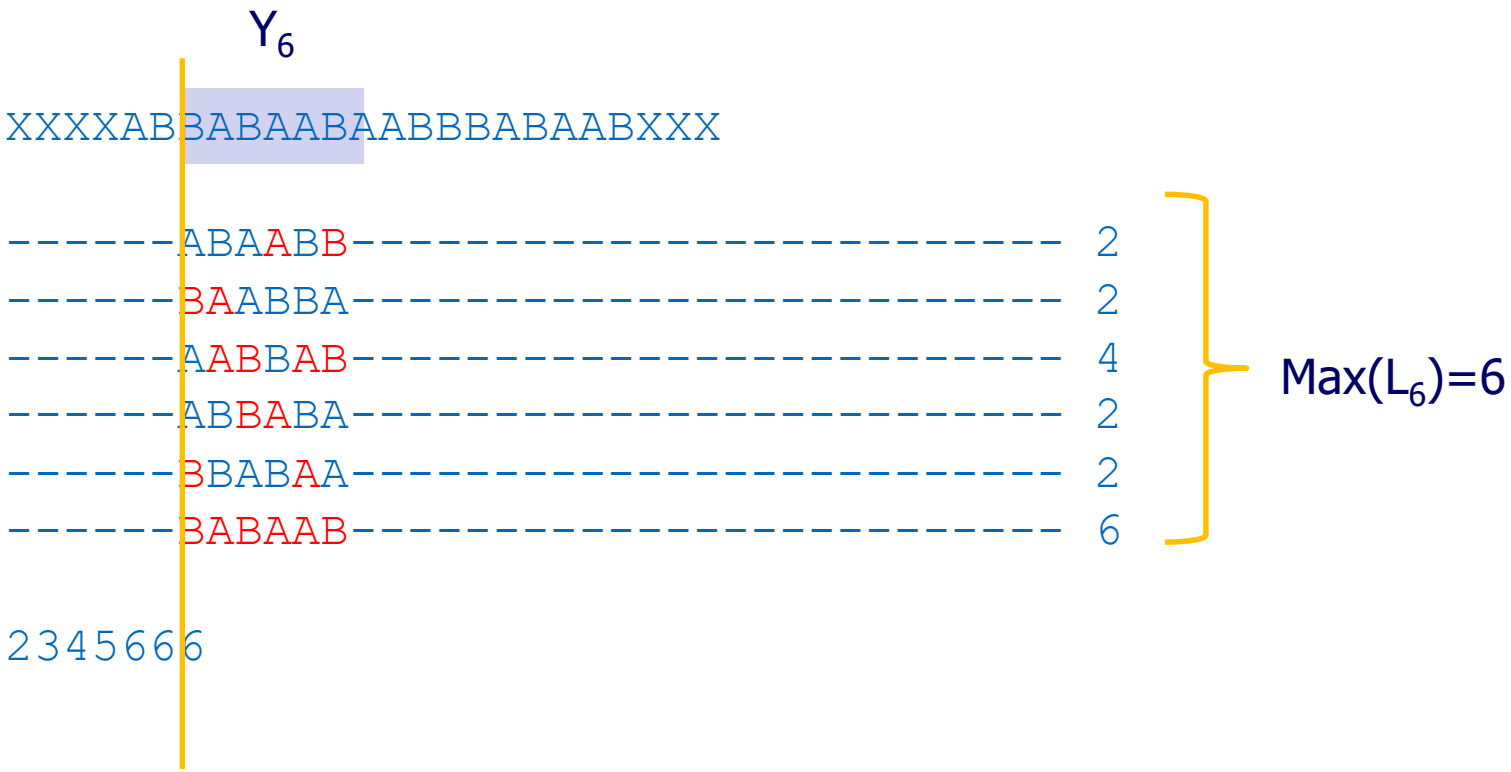
Toy example


Y_4
XXXXABBABAABAABBBABAABXXX
----ABAABB---------- 3
----BAABBA---------- 2
----AABBAB---------- 2
----ABBABA---------- 6
----BBABAA---------- 1
----BABAAB---------- 1
23456
Max(L_4)= 6

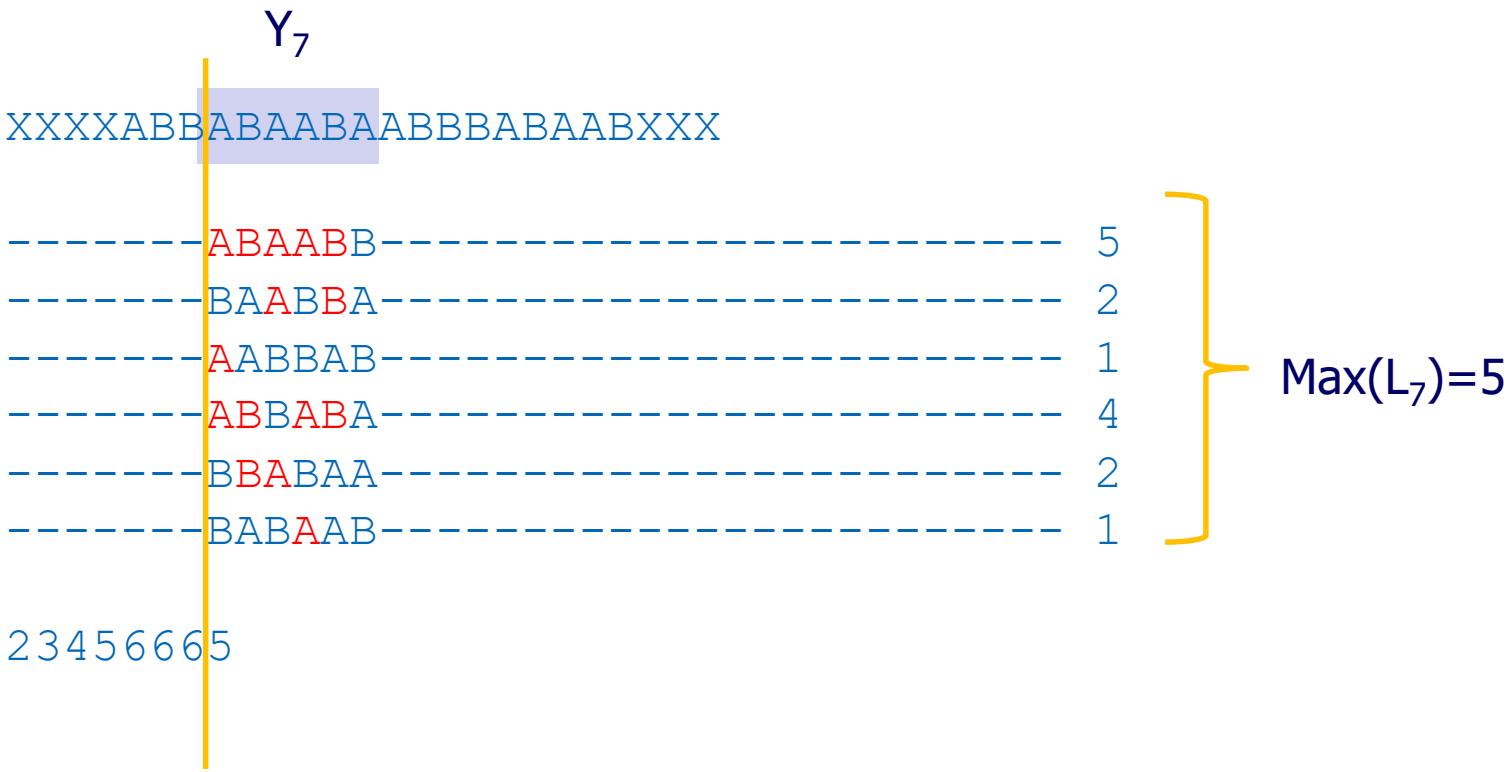
Toy example


Y_5
XXXXABBABAABAABBBABAABXXX
-----ABAABB---------- 2
-----BAABBA---------- 4
-----AABBAB---------- 1
-----ABBABA---------- 2
-----BBABAA---------- 6
-----BABAAB---------- 1
234566
$Max(L_5)=6$

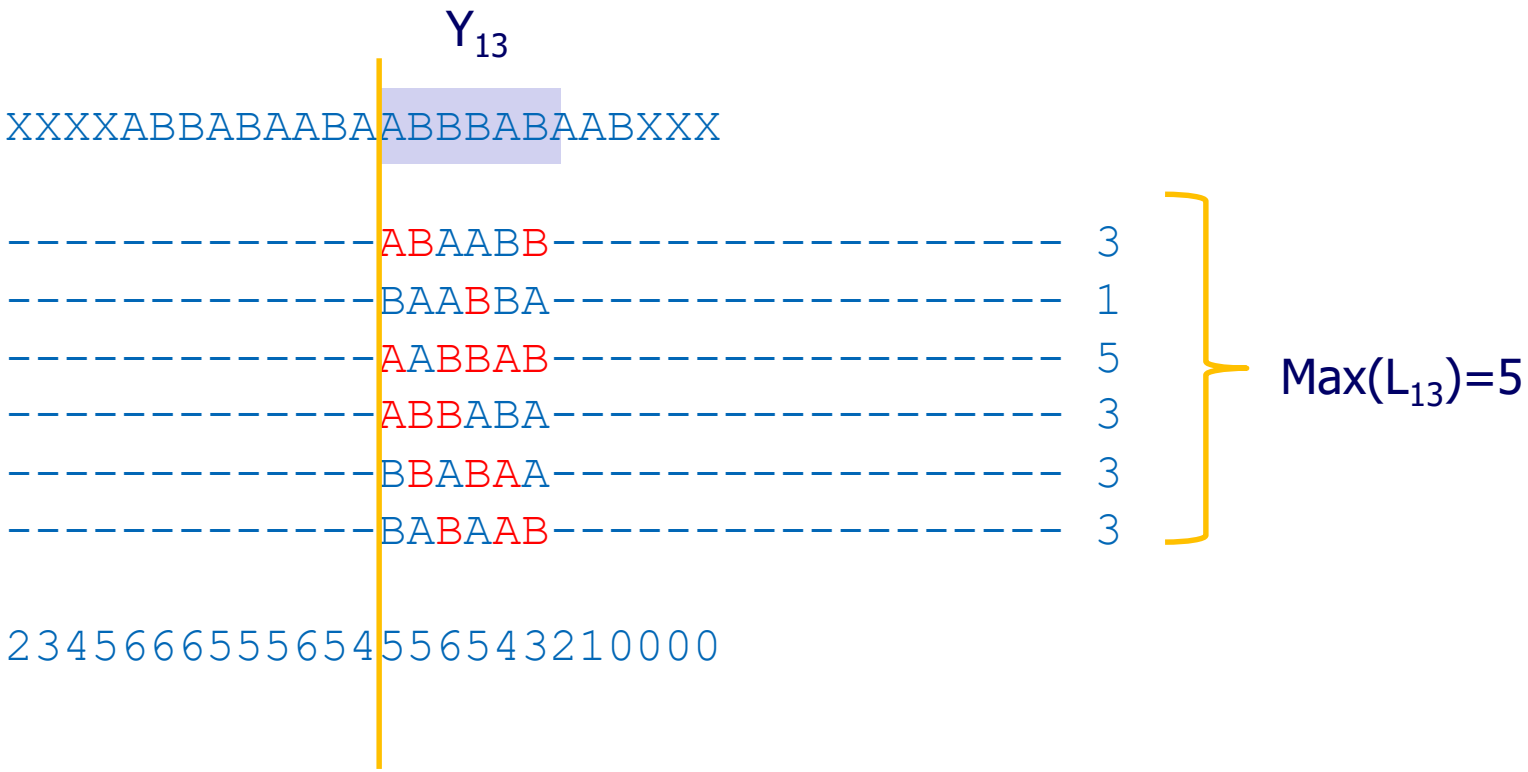
Toy example


Y_6
XXXXABBABAABAABBBABAABXXX
------ABAABB---------- 2
------BAABBA---------- 2
------AABBAB---------- 4
------ABBABA---------- 2
------BBABAA---------- 2
------BABAAB---------- 6
2345666
$Max(L_6)=6$

Toy example


Y_7
XXXXABBABAABAABBBABAABXXX
-------ABAABB------------------------- 5
-------BAABBA------------------------- 2
-------AABBAB------------------------- 1
-------ABBABA------------------------- 4
-------BBABAA------------------------- 2
-------BABAAB------------------------- 1
23456665
$Max(L_7)=5$

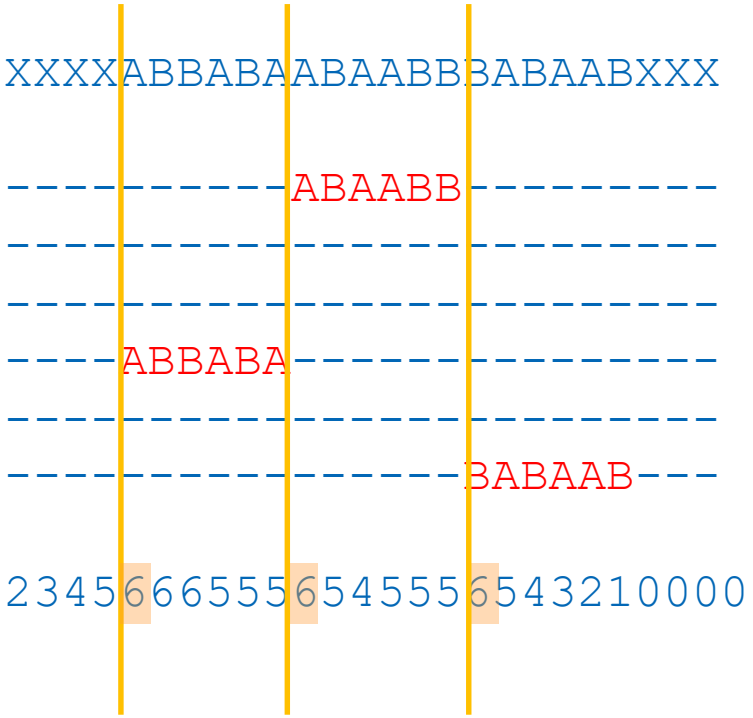
Toy example

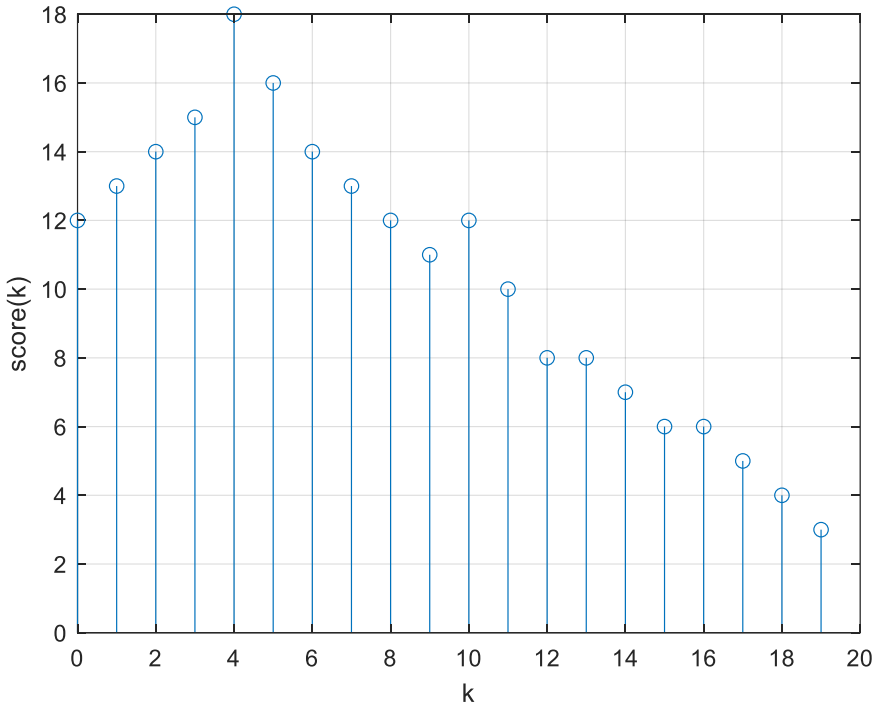

Y_{13}
XXXXABBABAABAABBBABAABXXX
----------ABAABB---------- 3
----------BAABBA---------- 1
----------AABBAB---------- 5
----------ABBABA---------- 3
----------BBABAA---------- 3
----------BABAAB---------- 3
$Max(L_{13})=5$
23456665556545565432100 00

Toy example

$$\text{Score}(k) = \max(L_k) + \max(L_{k+6}) + \max(L_{k+12})$$

$$\text{Score}(4) = \max(L_4) + \max(L_{10}) + \max(L_{16}) = 18$$



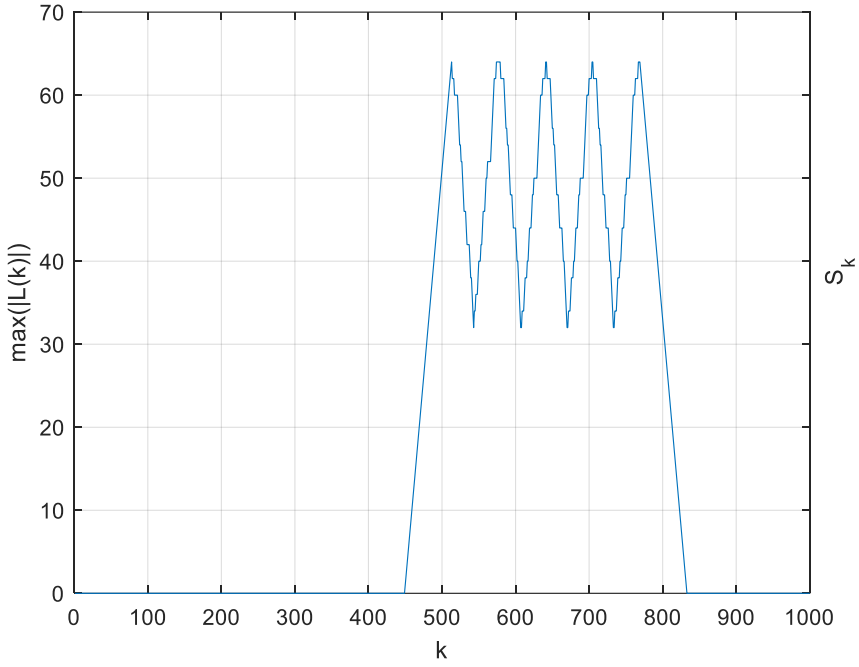


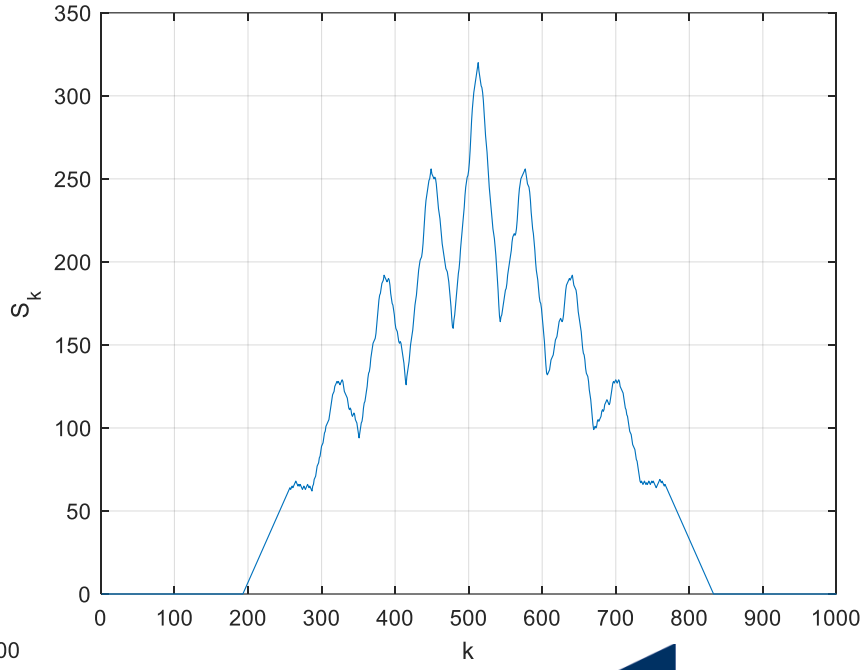
Detection problem: general case

N = 5, q = 64, no noise

$\max(|\boldsymbol{L_k}|)$

$$S_{\mathrm{k}}(\mathrm{Y}) = \sum_{n=0}^{4} \max(|\boldsymbol{L_{k+64n}}|)$$

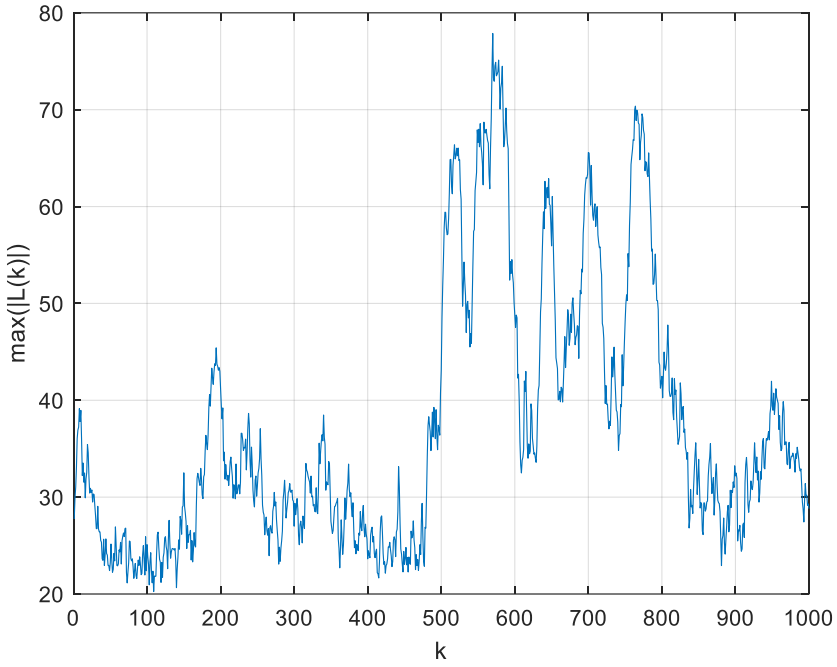




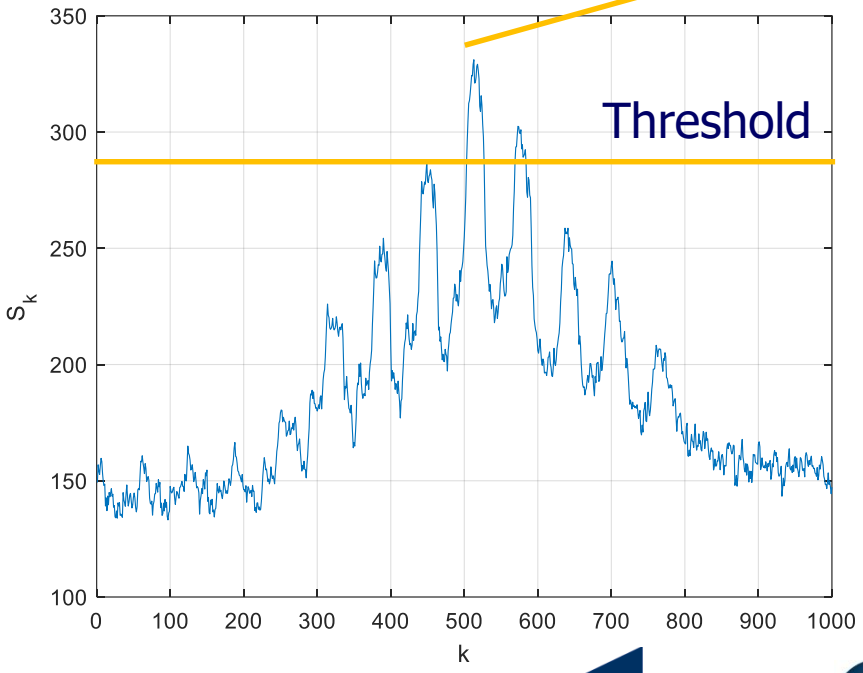
Detection problem: general case

N = 5, q = 64, with noise (-4 dB)

$\max(|\boldsymbol{L_k}|)$

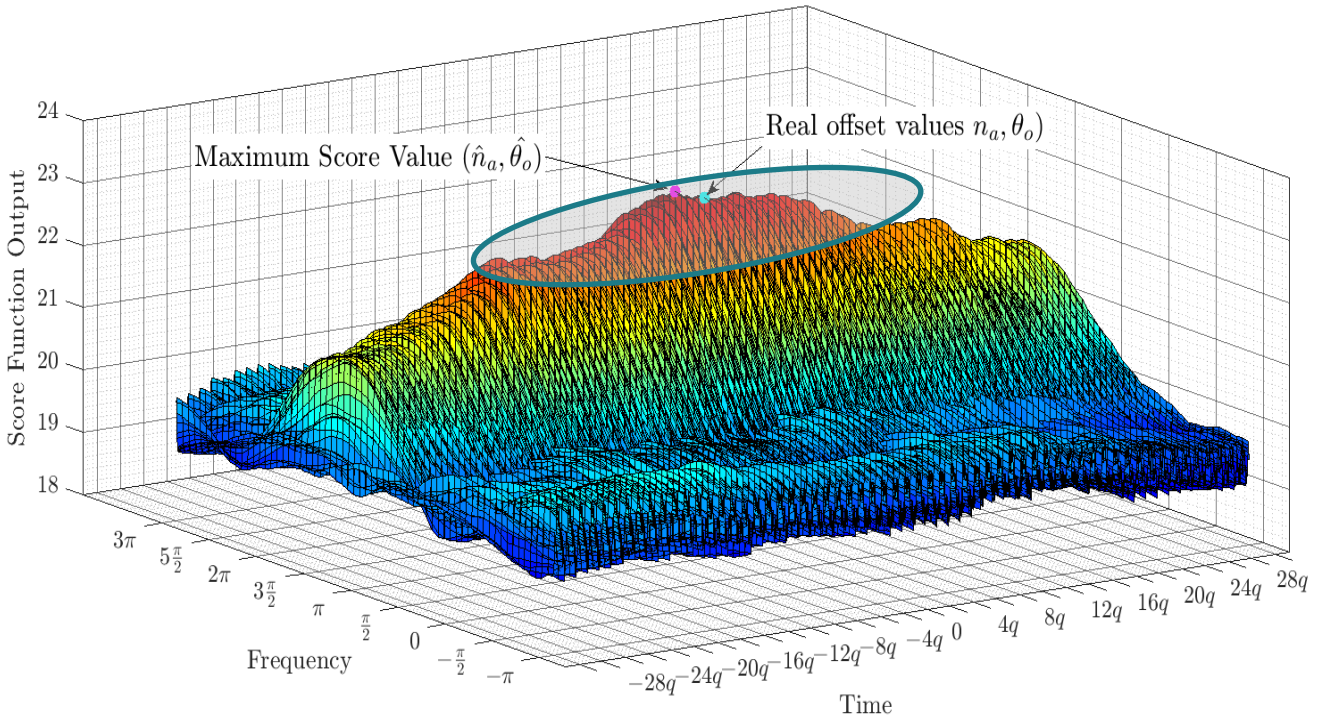


$$S_{\mathrm{k}}(\mathrm{Y}) = \sum_{n=0}^{4} \max(|\boldsymbol{L_{k+64n}}|)$$



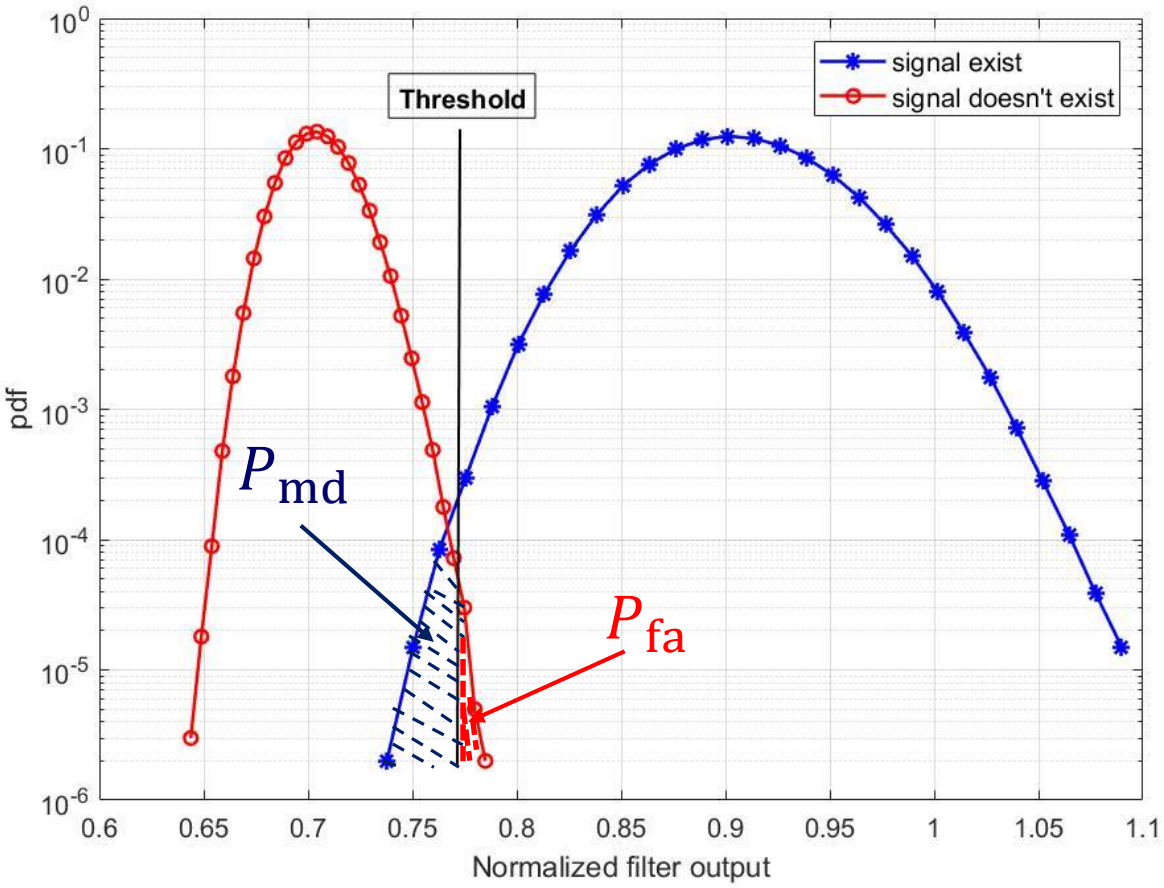
Impact of frequency offset

S_k^{θ} values in 3D grid where $\mathrm{N} = 60,\ q = 64$ and affected by a rotation between two consecutive chips of 0.0550 rd;
SNR = -10 dB.



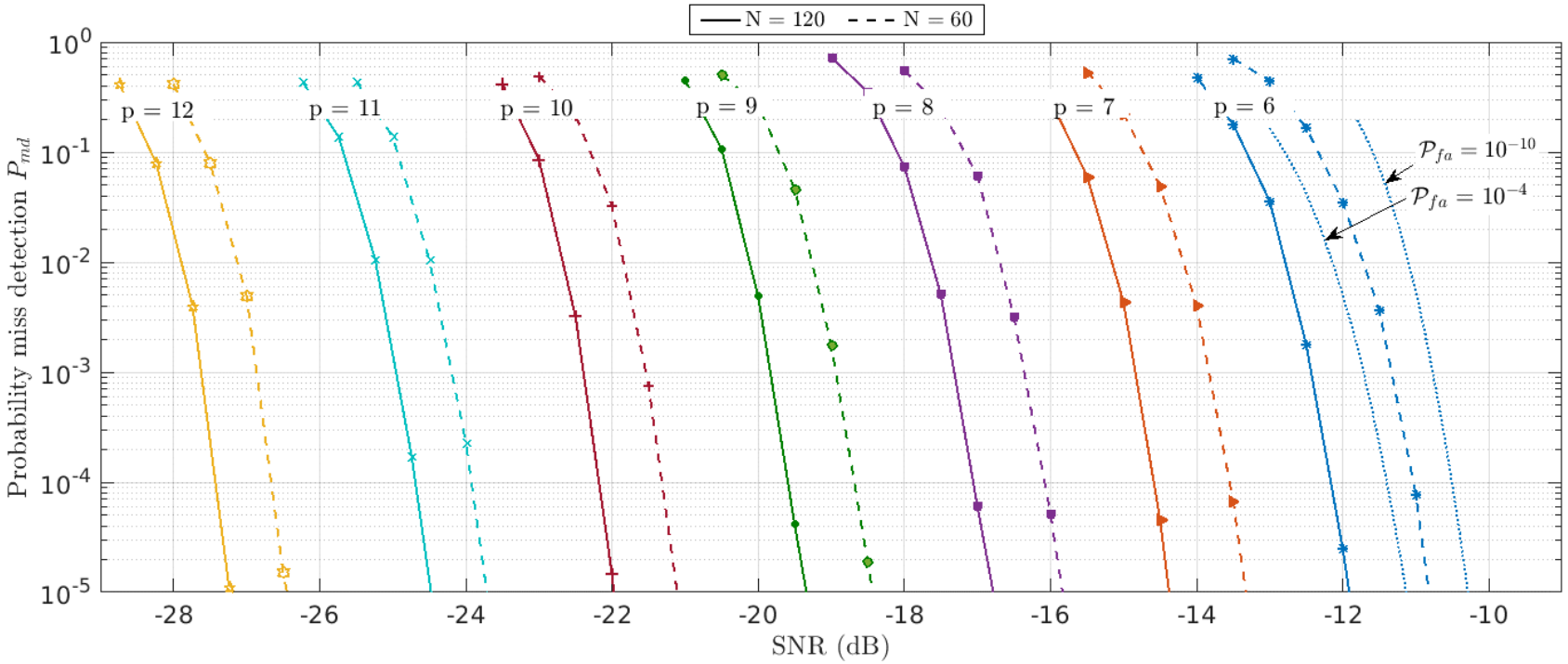
Trade-off False alarm/miss detection.

Output of filter distribution with and without a signal

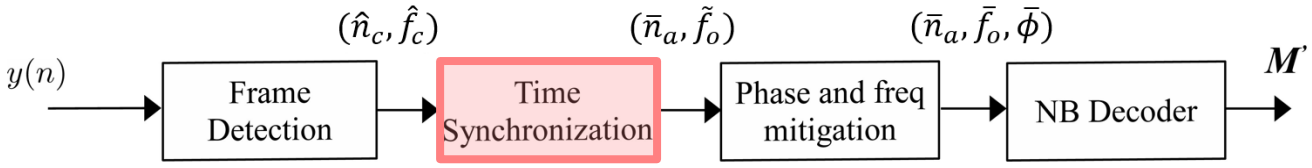


Example of detection performance

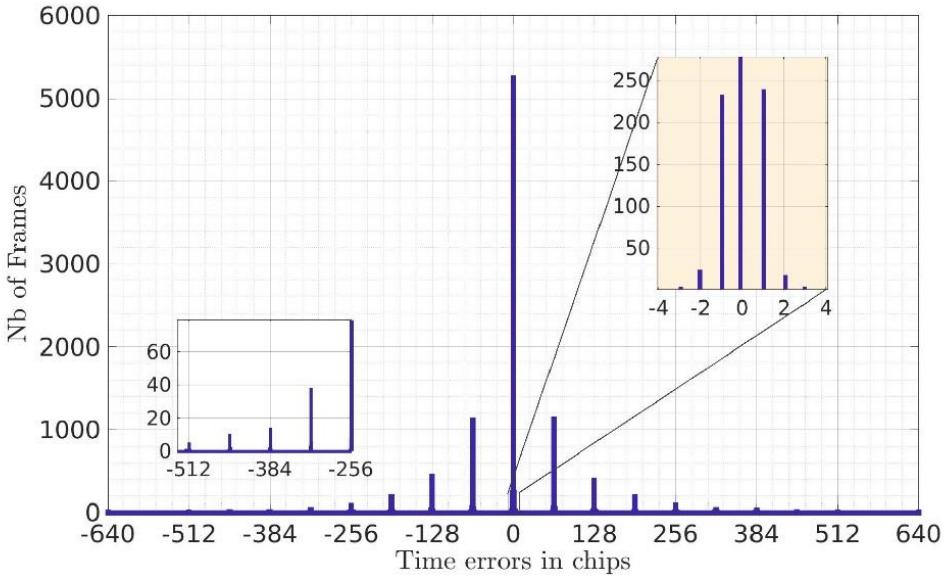
Theoretical performance (confirmed by Monte-Carlo simulation): P_{md} for P_{fa} = 10^{-6} with N=60, 120 over GF(64), GF(128), GF(256)… up to GF(2048).



Time Syncronization



- Chip errors for 10^4 detected QCSP frames of length N = 60 and $\boldsymbol{P}_0$ sequence of length q = 64 chips, at SNR = -10 dB.



- Errors at symbol level.
- Errors at chip level.

Symbol ambiguity: Over-modulation

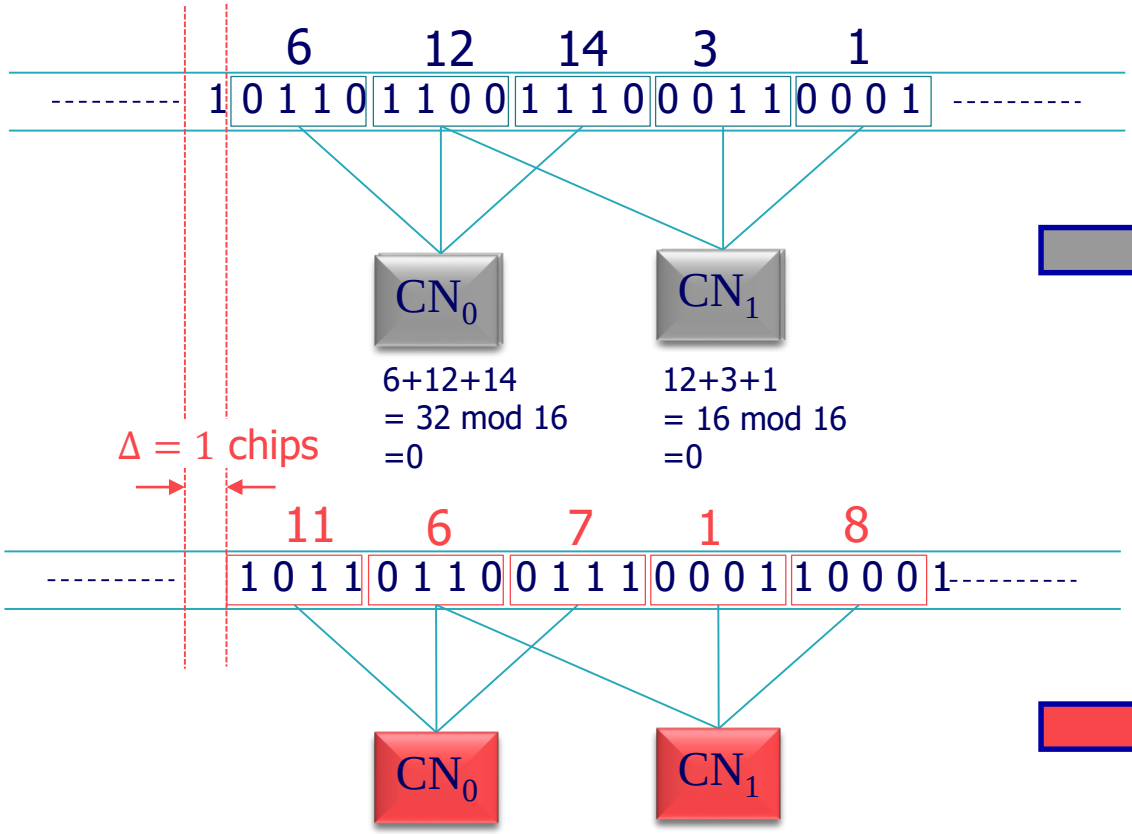
- OM generates a pre-defined phase pattern (a known sequence of ±1: +1 no phase change, and -1 (rotation)) within the sequence of the symbols being transmitted.

OM definition

- Sequence $\boldsymbol{B} = [b_0, b_1, \dots, b_{N-1}]$ with $b_k \in \{-1, 1\}$ and have good auto-correlation properties.
- QCSP frame defined as: $\boldsymbol{F} = [b_0\boldsymbol{P}_{c_0}, b_1\boldsymbol{P}_{c_1}, \dots, b_{N-1}\boldsymbol{P}_{c_{N-1}}]$

- Solve the time symbol ambiguity by testing several time hypothesis and by keeping the one that reproduce correctly the known phase pattern.
- Note: This method is also robust to a residual frequency offset.

Chip ambiguity: toy example



CN_1 Check Node
$\sum$ input mod 16

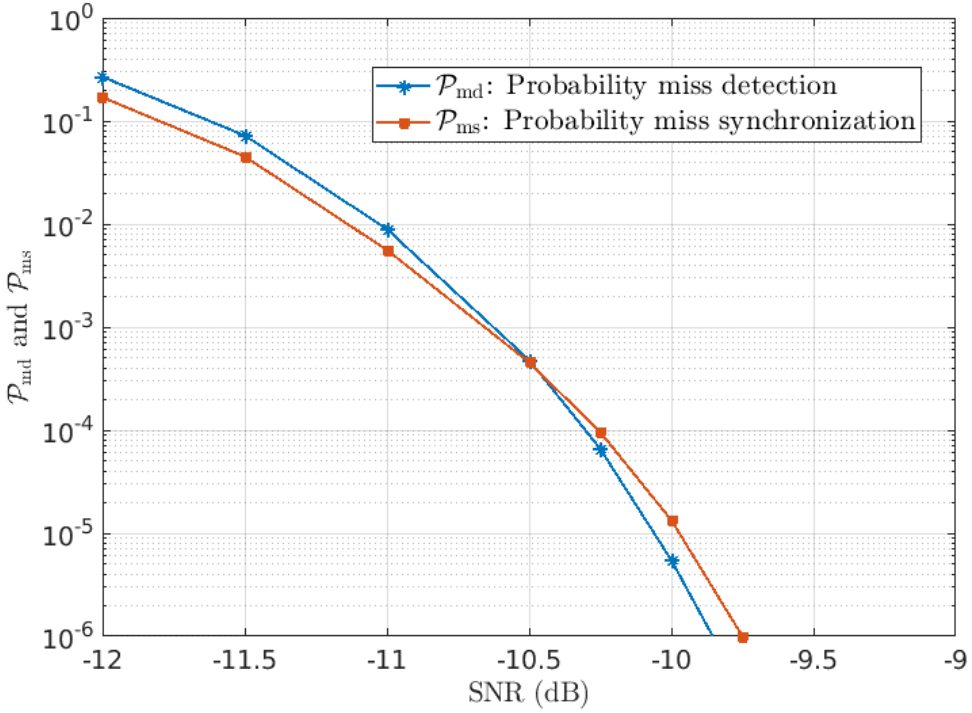
All PCs are fulfilled
$\sum$ input = 0 mod 16

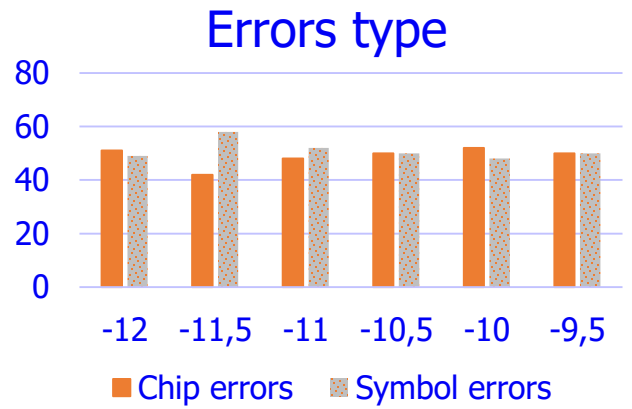
PCs are not fulfilled
$\sum$ input $\neq$ 0 mod 16

- Test several time hypothesis and keep the best one.
- Efficient/low complexity for NB-LDPC code.

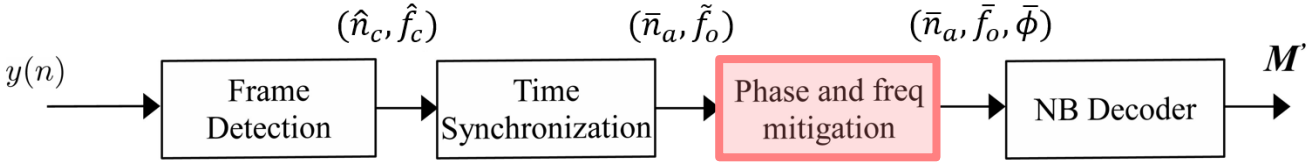
Time synchronization results

- $N = 60$ QCSP symbols
- NB-LDPC $R_c = 1/3$, $q = 64$
- Asynchronous AWGN channel
- Time and frequency shifts are uniformly randomly distributed.

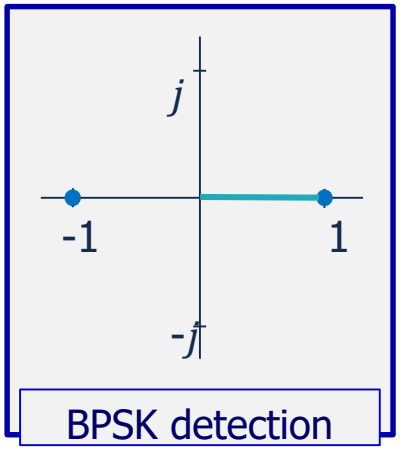




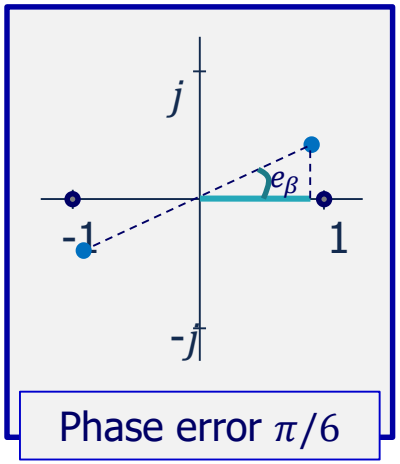
Impact off phase error



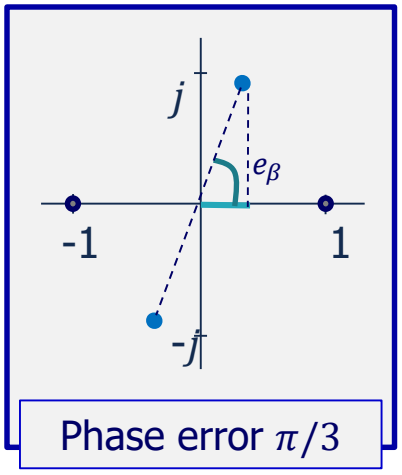
- Since LLR = Real(L), phase offset has a big impact on the generation of the LLRs.



BPSK detection



Phase error $\pi/6$

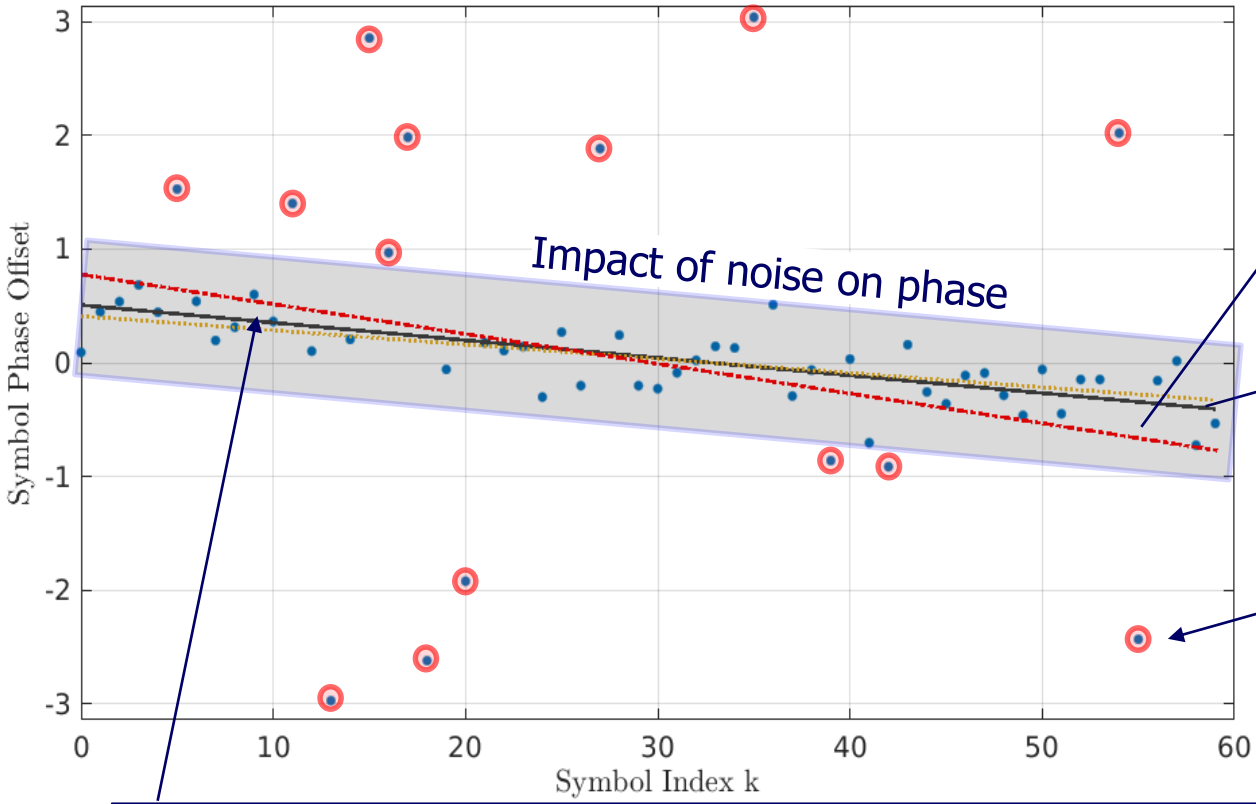


Phase error $\pi/3$

SNR degradation due to e_β

SNR degradation = $10\log_{10}(\cos(e_\beta)^2)$

Mitigation of phase error



Phase estimation using direct FFT method

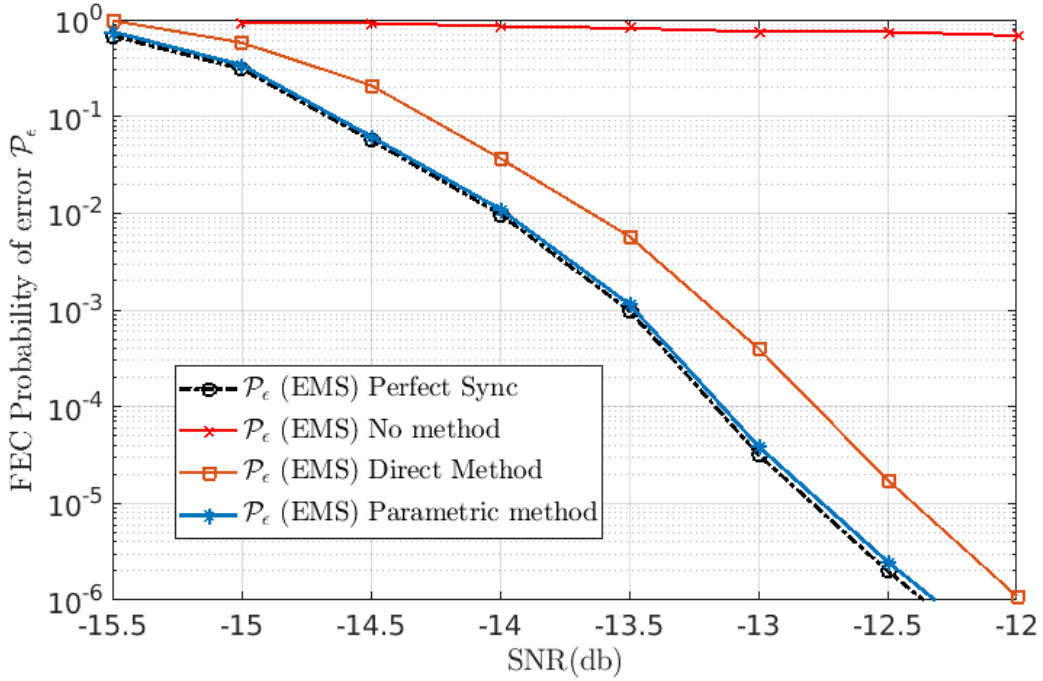
Real phase offsets

Detection of wrong maximum

Estimation using side information coming from:
1) Soft demodulation
2) Error control code

Simulation performance

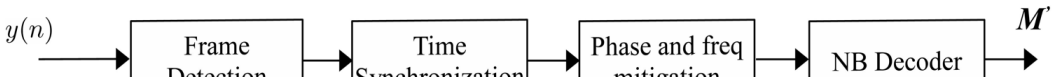
NB-LDPC performance in an AWGN channel with CCSK modulation. QCSP frame $K = 20$ symbols, $R_c = 1/3$, $q = 64$. The decoding algorithm used is the EMS with 30 decoding iterations [1].



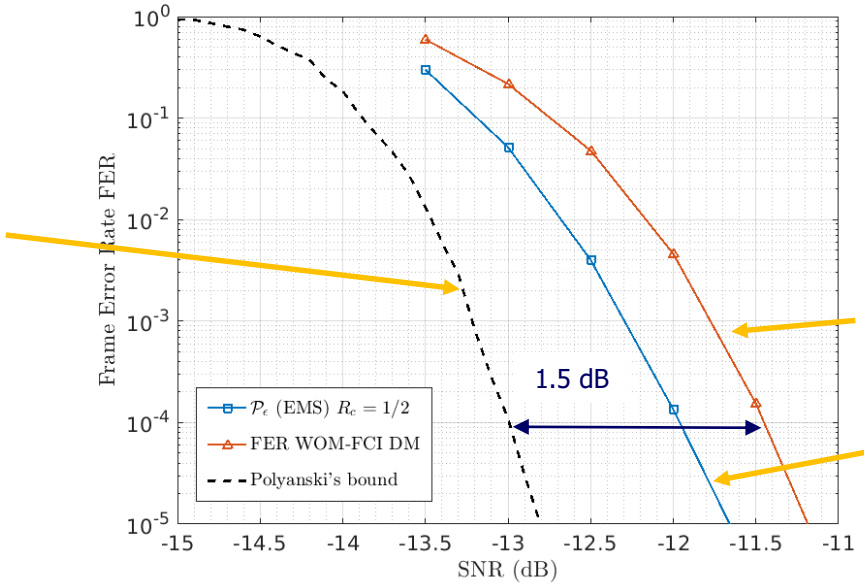
[1]: Saied, K., Ghouwayel, A. & Boutillon, E., *Phase Synchronization for NB-LDPC Coded CCSK Short Frames in Submitted to the 2022 IEEE Vehicular Technology Conference VTC2022*

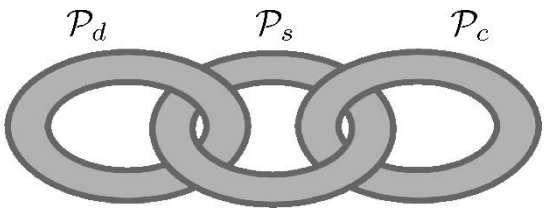
Overall performance

The QCSP parameters we choose to work on: $N = 120$ symbols, $q = 64$, $R_c = 1/2$
Asynchronous AWGN channel



Theoretical Lower bound [1,2]





FER async.

FER sync. channel

[1] Polyanskiy, Y., Poor, H. V. & Verdu, S., Channel Coding Rate in the Finite Blocklength Regime, *IEEE Transactions on Information Theory* **56,** 2307–2359, issn: 1557-9654 (May 2010).

[2] Savin, V., *Non-Binary Polar Codes for Spread-Spectrum Modulations in 2021 11th International Symposium on Topics in Coding (ISTC)* (2021), 1–5.

Outline

QCSP system model

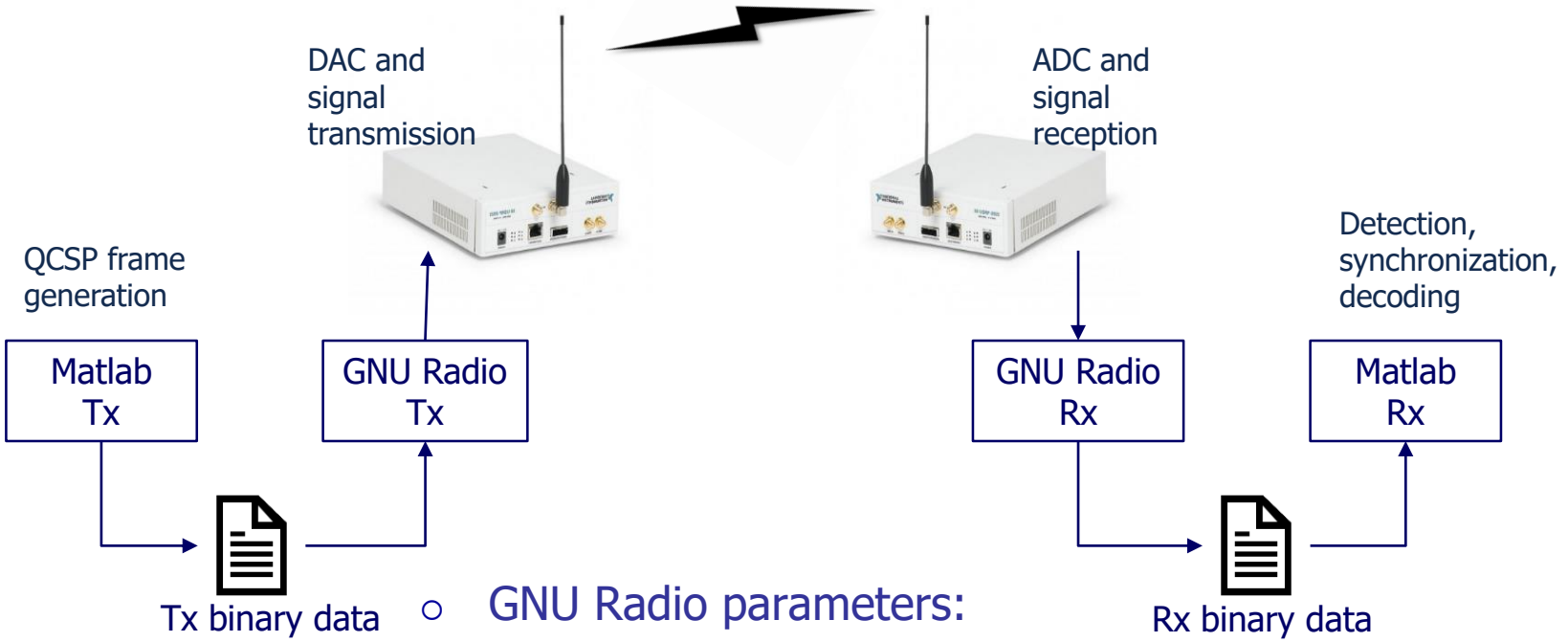
Receiver processing

GNU Radio implementation

Toward the multi-user context

Conclusion and perspectives

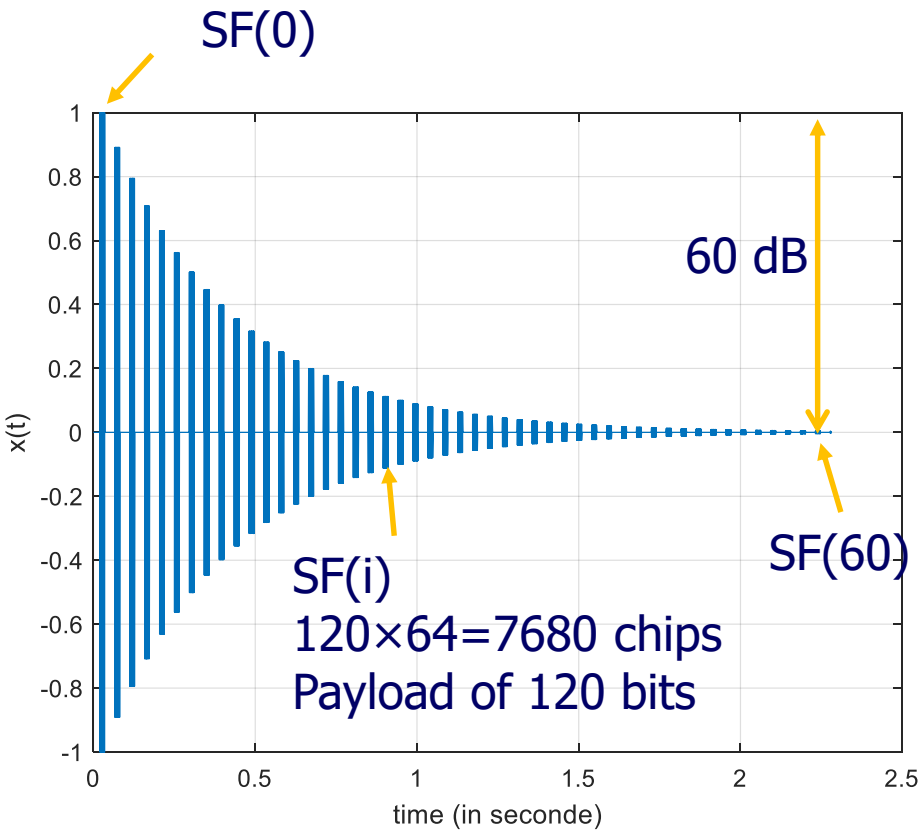
Offline experimental set-up



- GNU Radio parameters:
  - F = 433.950 MHz
  - Chip rate = 500 kHz
  - Bit information rate 0.5×6/64×1/3 = 15.6 Kbit/s
- Rise cosine filter (0.35)
- 8 samples/chip at receiver.

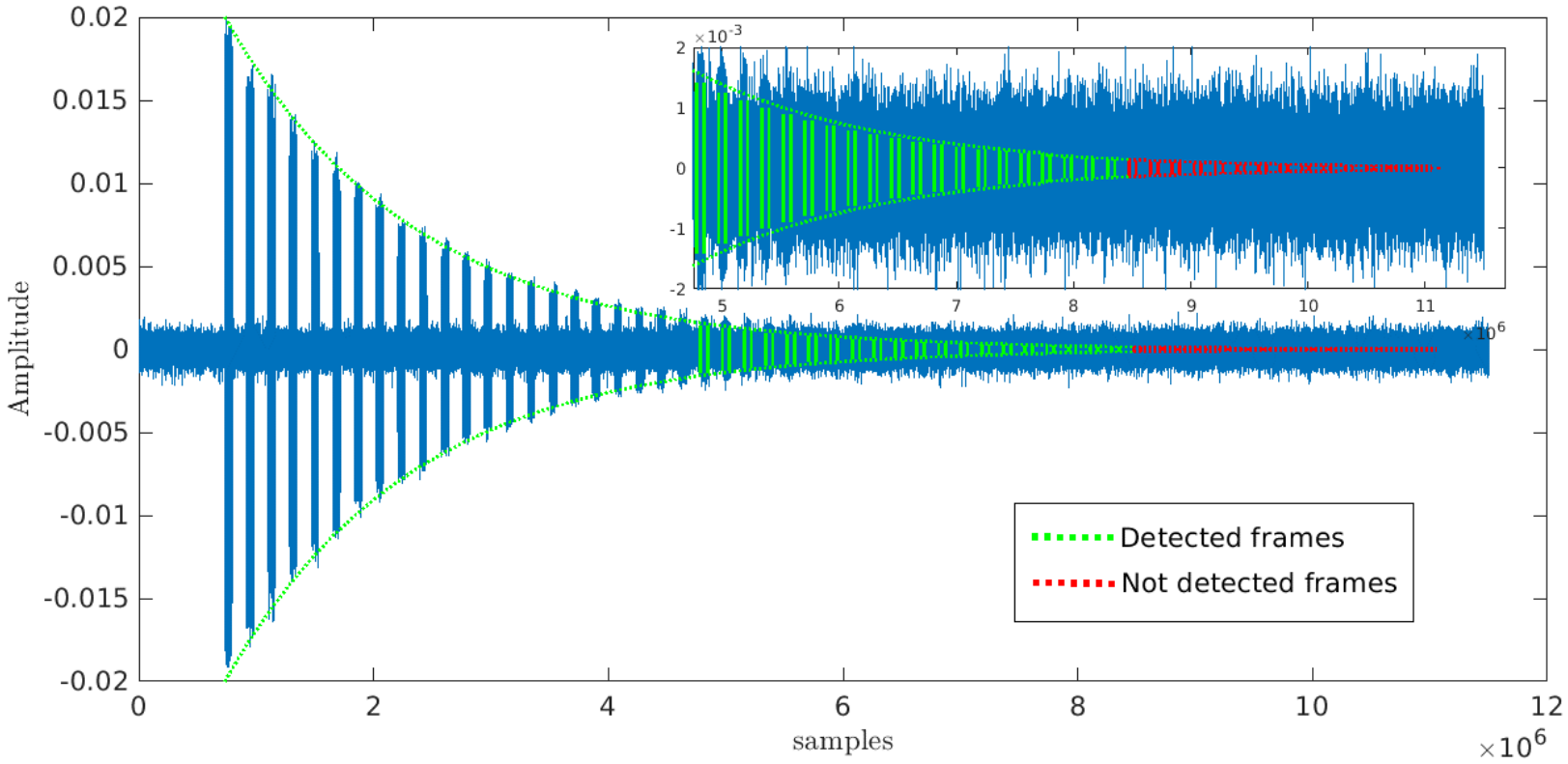
Note: Indoor test from 2 different rooms

Super-Frame structure

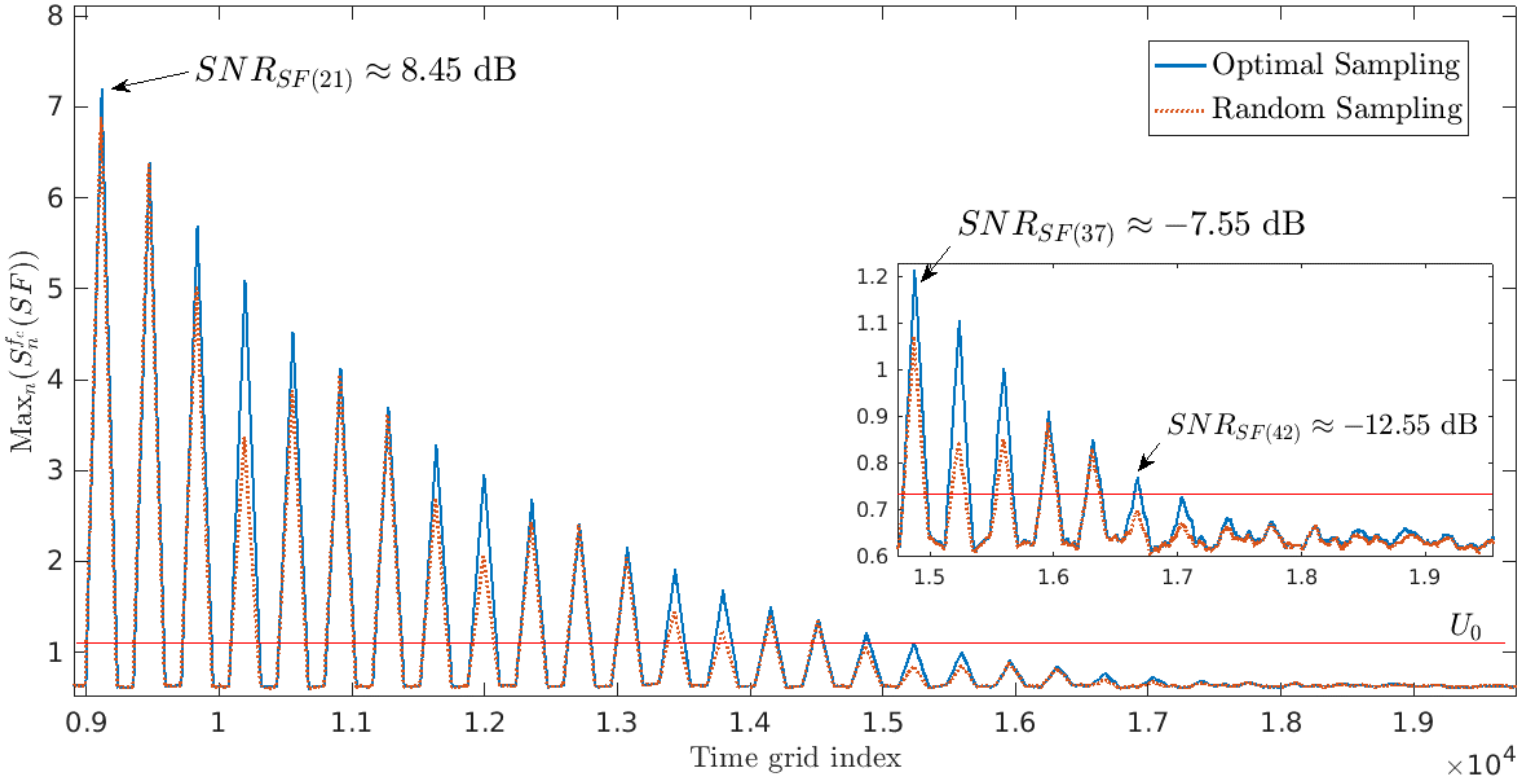


- Super Frame (SF) composed of 60 QCSP frames
- QCSP frame: K=40, N=120, r = 1/3 NB-LDPC code over GF(64).
- -1 dB of energy between two consecutive frame.

Received signal



Detection algorithm output



Analysis of the results

SF index SF(i)	SNR (dB)	Clock jitter effect (samples)	Detector No	Time Sync $\hat{r}$ (chips)	Time Sync $\hat{s}$ (symbols)	Freq and phase Sync. $\hat{f}_c$ (Hz)	Freq and phase Sync. $\hat{f}_o$ (Hz)	Freq and phase Sync. $\tilde{f}_o$ (Hz)	Freq and phase Sync. $\hat{\phi}$ (radian)	Decoding CCSK errors	Decoding NB-LDPC Is Codeword
0	29.41	0	Yes	0	0	0.007812	0.008057	0.008013	2.951911	0	Yes
1	28.40	2	Yes	0	0	0.007812	0.008057	0.008014	0.603615	0	Yes
2	27.26	3	Yes	0	0	0.007812	0.008057	0.008015	-1.588478	0	Yes
3	26.44	6	Yes	0	0	0.007812	0.008057	0.008016	2.668108	0	Yes
21	8.45	11	Yes	0	0	0.007812	0.008057	0.008018	-1.005000	0	Yes
36	-6.55	90	Yes	0	0	0.007812	0.008057	0.008023	1.620808	0	Yes
37	-7.55	92	Yes	0	0	0.007812	0.007812	0.008023	0.694619	1	Yes
38	-8.55	95	Yes	0	0	0.007812	0.007812	0.008023	-0.222060	1	Yes
39	-9.55	97	Yes	0	0	0.007812	0.008301	0.008022	-1.066529	9	Yes
40	-10.55	99	Yes	1	0	0.007812	0.008301	0.008022	-1.970423	12	Yes
41	-11.55	102	Yes	3	1	0.007812	0.008057	0.008023	-2.876781	29	Yes
42	-12.55	105	Yes	2	4	0.007812	0.007812	0.008024	2.448245	43	Yes
No detection No decoding											

$\Delta F = F_e - F_r$ 4.006 KHz

$\Delta F = F_e - F_r$ 4.012 KHz

Clock Jitter: 105 cycles over 6.8×10^6 cycles ≈15 ppm

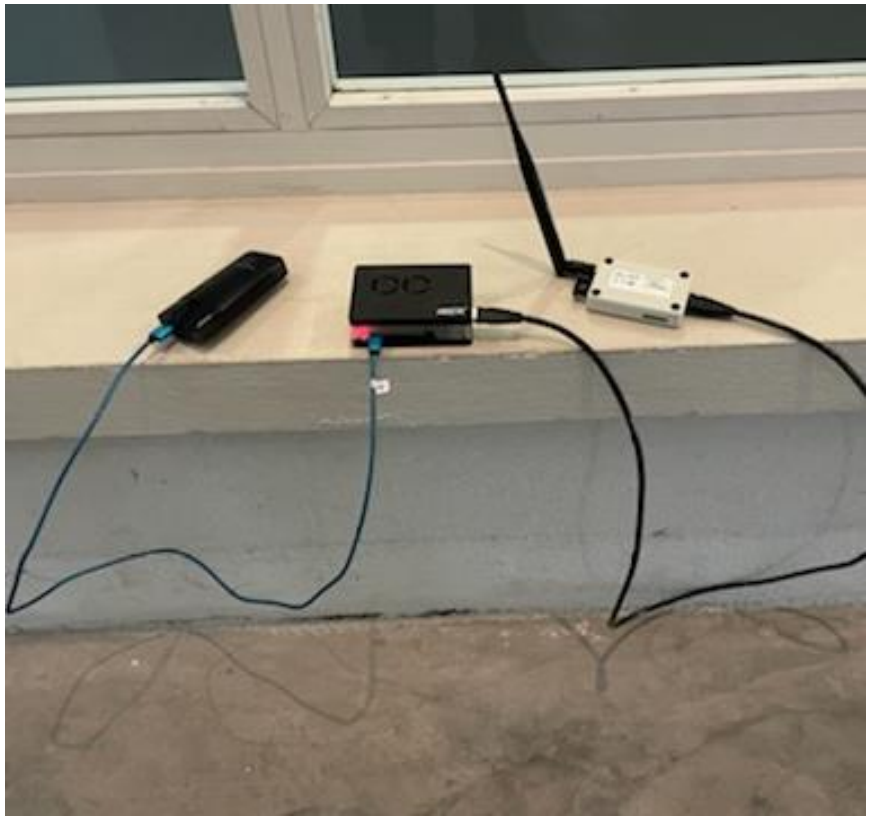
Real time implementation: emitter

Only XOR operation for coding + Half-rise-cosine filter: very simple.

Small, efficient transmitter, using software-defined radio.

The software alone can reach up to 300 Msps on a Raspberry Pi 4.

Runs up to 1.25 Mchip/s.

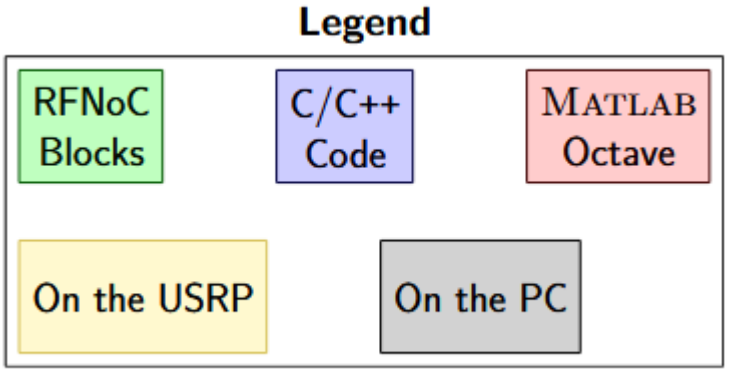


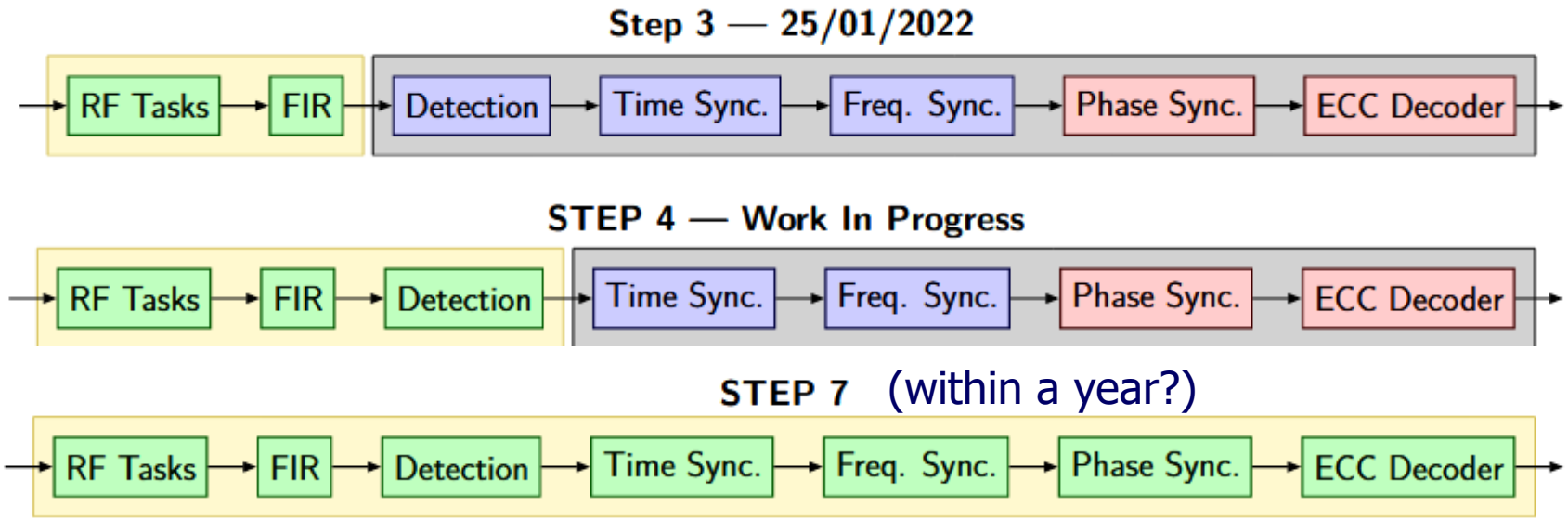
Real time implementation: receiver

Hybrid receiver with hardware&software.

Step 3: 500 kchip/s real time communication over GF(64) (raw bit-throughput: 46 Kbit/s)

Step 4: much higher rate will be obtained





Real time implementation: receiver

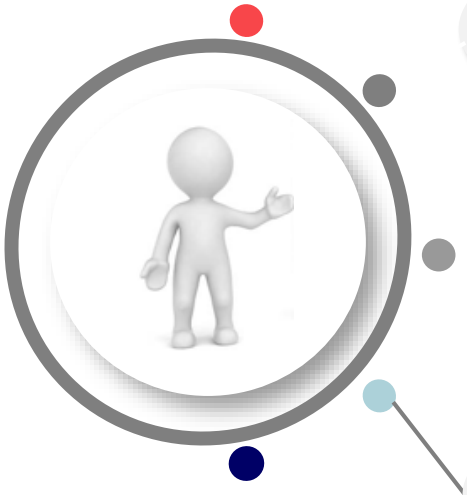
Results	Coarse Δ	Fine Δ	Symb Δ	NBCA Δ	Coarse f_0	Fine f_0	Accr f_0	LRCA f_0	Est. φ_0	LRCA φ_0	Message
FAILURE	1921	1921	2113	2127	0.000e+00	-3.906e-03	1.218e-03	1.167e-03	-1.476e+00	-2.828e+00	'XXXXXXXXXXXXXXX'
SUCCESS	1921	1921	1921	1921	0.000e+00	-3.906e-03	-6.532e-03	-6.498e-03	1.275e+00	1.680e+00	'00090|15:30:19A'
SUCCESS	1921	1921	1921	1921	0.000e+00	-3.906e-03	-6.543e-03	-6.509e-03	-5.043e-01	-9.912e-02	'00091|15:30:20A'
SUCCESS	1921	1921	1921	1921	0.000e+00	-3.906e-03	-6.533e-03	-6.497e-03	1.813e+00	1.874e+00	'00092|15:30:20A'
SUCCESS	1921	1921	1921	1921	0.000e+00	-3.906e-03	-6.533e-03	-6.497e-03	-1.127e+00	-1.065e+00	'00093|15:30:21A'
SUCCESS	1921	1921	1921	1921	0.000e+00	-3.906e-03	-6.531e-03	-6.495e-03	1.079e+00	1.140e+00	'00094|15:30:22A'
SUCCESS	1921	1921	1921	1921	0.000e+00	-3.906e-03	-6.538e-03	-6.504e-03	-2.423e+00	-2.018e+00	'00095|15:30:22A'
SUCCESS	1921	1921	1921	1921	0.000e+00	-3.906e-03	-6.530e-03	-6.496e-03	-1.754e+00	-1.349e+00	'00096|15:30:23A'
SUCCESS	1921	1921	1921	1921	0.000e+00	-3.906e-03	-6.526e-03	-6.491e-03	8.354e-01	1.241e+00	'00097|15:30:24A'
SUCCESS	1921	1921	1921	1921	0.000e+00	-3.906e-03	-6.536e-03	-6.499e-03	2.091e+00	2.153e+00	'00098|15:30:25A'
SUCCESS	1921	1921	1921	1921	0.000e+00	-3.906e-03	-6.527e-03	-6.491e-03	2.270e+00	2.332e+00	'00099|15:30:25A'
SUCCESS	1921	1921	1921	1921	0.000e+00	-3.906e-03	-6.540e-03	-6.506e-03	-1.453e+00	-1.048e+00	'00100|15:30:26A'
SUCCESS	1921	1921	1921	1921	0.000e+00	-3.906e-03	-6.540e-03	-6.506e-03	-1.462e+00	-1.057e+00	'00100|15:30:26A'
SUCCESS	1921	1921	1921	1921	0.000e+00	-3.906e-03	-6.531e-03	-6.495e-03	1.546e+00	1.687e+00	'00101|15:30:27A'
SUCCESS	1921	1921	1921	1921	0.000e+00	-3.906e-03	-6.524e-03	-6.490e-03	-2.347e+00	-1.942e+00	'00102|15:30:28A'
SUCCESS	1921	1921	1921	1921	0.000e+00	-3.906e-03	-6.531e-03	-6.495e-03	2.312e-01	2.927e-01	'00103|15:30:28A'
SUCCESS	1921	1921	1921	1921	0.000e+00	-3.906e-03	-6.528e-03	-6.491e-03	-2.455e+00	-2.394e+00	'00104|15:30:29A'
SUCCESS	1921	1921	1921	1921	0.000e+00	-3.906e-03	-6.529e-03	-6.493e-03	1.646e+00	1.707e+00	'00105|15:30:30A'
SUCCESS	1921	1921	1921	1921	0.000e+00	-3.906e-03	-6.528e-03	-6.494e-03	1.830e+00	2.235e+00	'00106|15:30:30A'
SUCCESS	1921	1921	1921	1921	0.000e+00	-3.906e-03	-6.526e-03	-6.491e-03	1.277e+00	1.683e+00	'00107|15:30:31A'
SUCCESS	1921	1921	1921	1921	0.000e+00	-3.906e-03	-6.527e-03	-6.491e-03	8.412e-01	9.028e-01	'00108|15:30:32A'
SUCCESS	1921	1921	1921	1921	0.000e+00	-3.906e-03	-6.526e-03	-6.492e-03	1.154e+00	1.559e+00	'00109|15:30:33A'
SUCCESS	1921	1921	1921	1921	0.000e+00	-3.906e-03	-6.528e-03	-6.494e-03	-1.746e+00	-1.341e+00	'00110|15:30:33A'

Real time information display about received messages
GF(64), N = 60, NB-LDPC code of rate 1/3, k = 20 (120 bits).

Outline

QCSP system model

Receiver algorithms



GNU Radio implementation

Toward the multi-user context

Conclusion and perspectives

Degrees of diversity

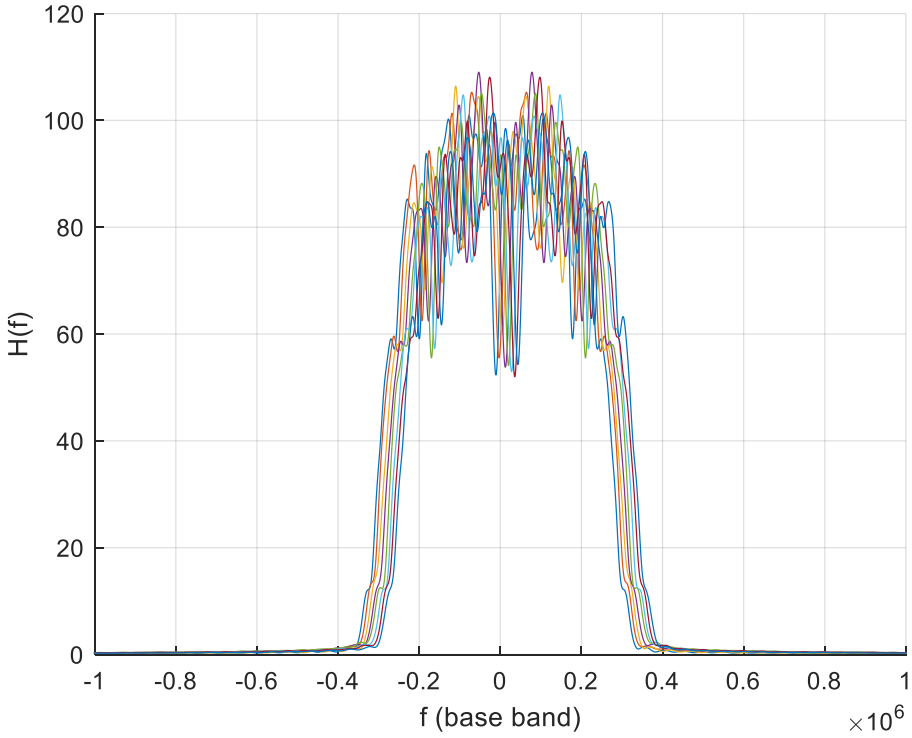
QCSP frame offers many degree of diversity

- Generic diversity
  - Time, frequency, space diversity (MIMO).
- Specific diversity
  - Phase (two frames in quadrature are orthogonal).
  - Spreading sequence of the CCSK modulation.
  - Overmodulation sequence of the frame
  - Error control code associated to the use
  - New: Almost overlapping frequency

Overlapping frequency

- Two frames with identical structure (CCSK sequence, overmodulation, code) can be received simultaneously if the are separated by a frequency offset of f = 1/q Hz per chip.
- Example: Chip rate of 500 KHz => Bandwidth of 1 MHz.
  - For q = 64, two frames separated by a multiple of 500/64 = 7.8 KHz can be received simultaneously.
  - Good spectral efficiency

Overlapping frequency transmission



- 8 frames received simultaneously with frequency offsets multiple of 7.8 KHz.

At 4 dB of SNR, up to 16 frames with same spreading sequence can be received simultaneously.

Future work

- Detection in the multi-path channel (ongoing study).
- How to take maximum profit of the available diversity?
- Multi-user detection algorithm to be proposed and tested (successive cancellation).

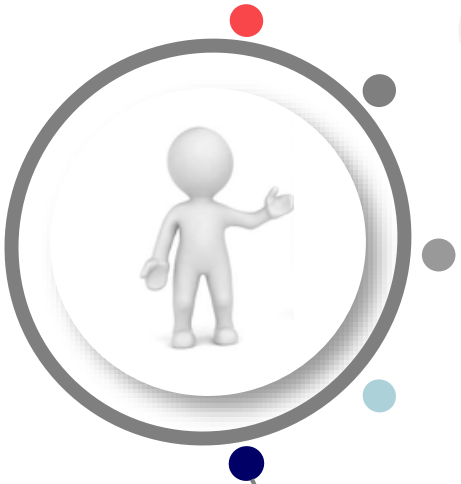
Outline

QCSP system model

Receiver algorithms

GNU Radio implementation

Toward the multi-user context



Conclusion and perspectives

Conclusion

Improving point to point communication (use of Zadoff-Chu sequence, use of NB-Turbo codes,…)

Improving the **design** of the **SDR demonstrator** using GNU radio with **real-time reception**.

Define theoretical topics to push the work towards **PHY/MAC layers** for IoT

Experimentation of QCSP in the context of **IoT multi-user access.**

Perspectives

- We are open for interactions
- QCSP: More information (deliverables, publications):

https://qcsp.univ-ubs.fr/

Questions?

From http://getdrawings.com/mc-escher-drawing

Thanks to
Quentin Lampin
Olivier Seller
Dominique Barthel
Sandoche Balakrichenan
May Dinh