Fixed Point LDPC Decoder Saturation & Precision

Mehrzad Malmirchegini, Shadi Abu-Surra, Thomas Henige, and Eran Pisek

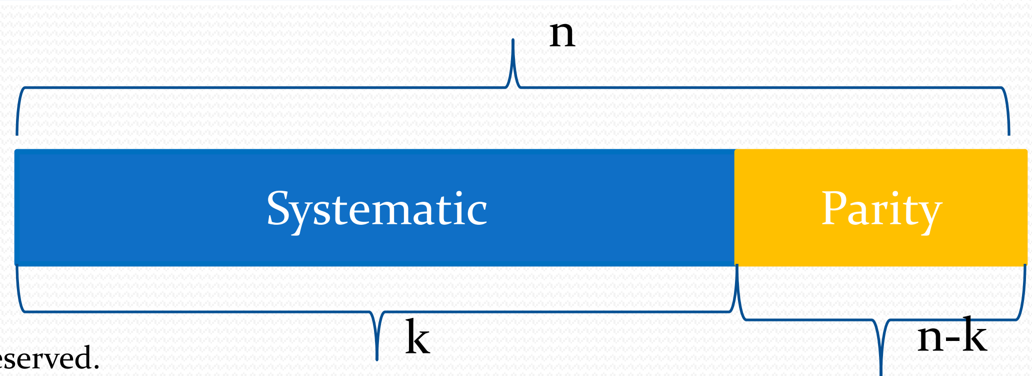
Samsung Research Institute America – Dallas
Samsung Electronics
Richardson, Texas, USA

Agenda

- LDPC codes Introduction
- Traditional LDPC Decoding Methods
- Fixed Point LDPC decoder issues
- Propose Saturation mechanism
- Results
- Summary

LDPC Codes Intro. (performance)

- Initially developed by Gallager in his Thesis on 1963.
- Systematic code
- Theoretically (long block sizes) achieving near-Shannon capacity performance (within 0.0045dB).
- Based on an sparsed H-Matrix that indicates the location of the systematic data and the related parity bits.
- Coding Rate R=k/n



LDPC Codes Intro. (Con't)

Computational Complexity can be high for both the Encoder and the decoder

Up until the mid 90's, LDPC codes considered theoretical only, and were not adopted in any commercial communication standards

Together with the Turbo Coding development by Berrou et al in 1993, came new interest in LDPC Codes.

During the 2000's LDPC codes were already part of several standards:

- DVB-S2
- WiMax (IEEE802.16e)
- IEEE802.11n/ac
- WiGig (IEEE802.11ad)
- Etc.

LDPC Decoding Methods (Power)

The three major LDPC decoding methods are:

- **Maximum Likelihood** (Alternative MAP (Maximum A Posteriori probability)
- **Log domain Sum of Products**
- **Log domain Min-Sum**

Maximum Likelihood

- Selects the codeword that maximizes the likelihood of the received signal.
- Impractical for large block sizes.
- Represented by (MAP):

$$\operatorname*{argmax}_{i} P(\overline{r}|\overline{x_i})$$

- where r is the input signal to the decoder and x_i , $i=\{0:2^N-1\}$ is the set of all possible codes in that satisfy:

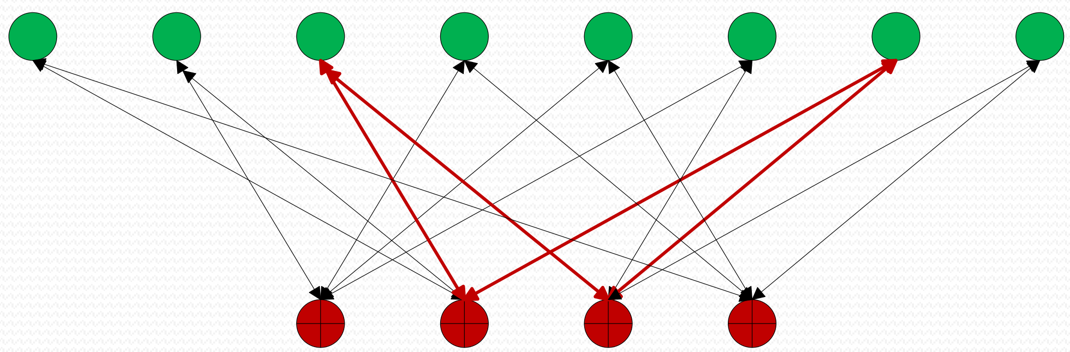
$$\overline{x_i}H^T = \overline{0}$$

Tanner Representation (Regular)

Parity Check Matrix

$$H = \begin{bmatrix} 0 & 1 & 0 & 1 & 1 & 1 & 0 & 0 \\ 1 & 1 & 1 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 & 0 & 1 & 1 & 1 \\ 1 & 0 & 0 & 1 & 1 & 0 & 0 & 1 \end{bmatrix} \qquad w_r = 4,\ w_c = 2,\ r = \frac{w_c}{w_r} = \frac{1}{2}$$

Tanner Graph Representation



Variable Nodes

Check Nodes

Low Density Parity Check codes

- Linear block codes
- Sparse -Row and column weights << dimension of the parity check matrix
- “Capacity-approaching”
- Iterative decoding with linear decoding complexity (e.g., belief propagation)

Log-Domain Min-Sum

- A lower complexity decoder
- Uses the min operation to find the magnitude of the soft bits per check node.

$$L(r_{ji}) = A \prod_{i' \in V_{j\backslash i}} \alpha_{i'j} \min_{i' \in V_{j\backslash i}} \beta_{i'j}$$

- Frame Error Rate (FER) is inferior compared to SPA and ML.
- Iterative (Message Passing)

Log-Domain Min-Sum methods

Flooding

- Multiple minimum (min) processors supporting simultaneous row processing in the H-Matrix.
- All CN entries per VN are summed to produce the extrinsic information for the next iteration.
- Low processing time per iteration (down to 1 cyc / iter)

Layered

- Serially processes each row to avoid contention.
- The Extrinsic values are updated each row.
- Fast convergence (33% less iterations than Flooding)

40	-1	38	-1	13	-1	5	-1	18	-1	-1	-1	-1	-1	-1	-1
34	-1	35	-1	27	-1	-1	30	2	1	-1	-1	-1	-1	-1	-1
-1	36	-1	31	-1	7	-1	34	-1	10	41	-1	-1	-1	-1	-1
-1	27	-1	18	-1	12	20	-1	-1	-1	15	6	-1	-1	-1	-1
35	-1	41	-1	40	-1	39	-1	28	-1	-1	3	28	-1	-1	-1
29	-1	0	-1	-1	22	-1	4	-1	28	-1	27	-1	23	-1	-1
-1	31	-1	23	-1	21	-1	20	-1	-1	12	-1	-1	0	13	-1
-1	22	-1	34	31	-1	14	-1	4	-1	-1	-1	13	-1	22	24

Example: WiGig

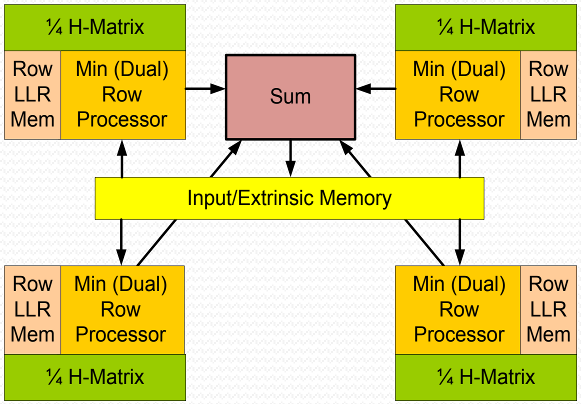
- Single Block size: 672 bits with R=1/2, 5/8, ¾, 13/16.
- Lifting Factor: Z=42, No. of columns: 16
- Example: R=1/2 H-Matrix

40	-1	38	-1	13	-1	5	-1	18	-1	-1	-1	-1	-1	-1	-1
34	-1	35	-1	27	-1	-1	30	2	1	-1	-1	-1	-1	-1	-1
-1	36	-1	31	-1	7	-1	34	-1	10	41	-1	-1	-1	-1	-1
-1	27	-1	18	-1	12	20	-1	-1	-1	15	6	-1	-1	-1	-1
35	-1	41	-1	40	-1	39	-1	28	-1	-1	3	28	-1	-1	-1
29	-1	0	-1	-1	22	-1	4	-1	28	-1	27	-1	23	-1	-1
-1	31	-1	23	-1	21	-1	20	-1	-1	12	-1	-1	0	13	-1
-1	22	-1	34	31	-1	14	-1	4	-1	-1	-1	13	-1	22	24

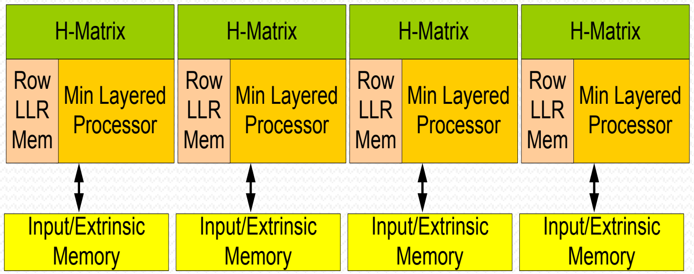
- 4-Layer decodable (Row independency)

Layered vs. Flooding

Flooding



Layered



- Assume same active power consumption
- Layered requires 33% less iterations – higher processing rate
- Flooding processes a single block at a time. Layered processes 4 blocks in parallel. Flooding has lower latency and requires less memory.

Fixed-Point issues

- Quantization of LLR values

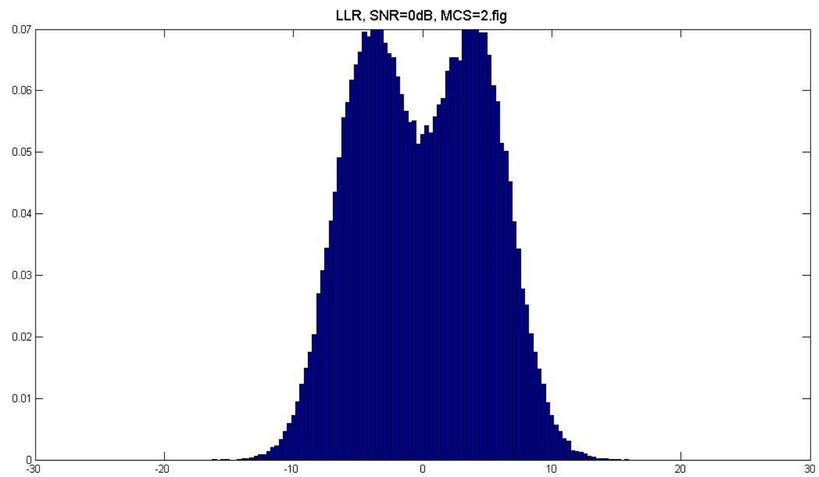
- Fixed point constraints

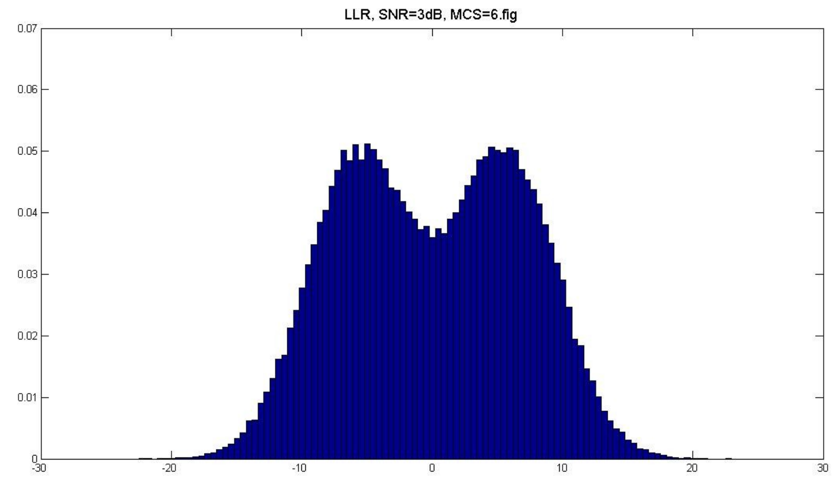
- As the number of decoding iteration increases, LLR values grow, resulting in saturation.

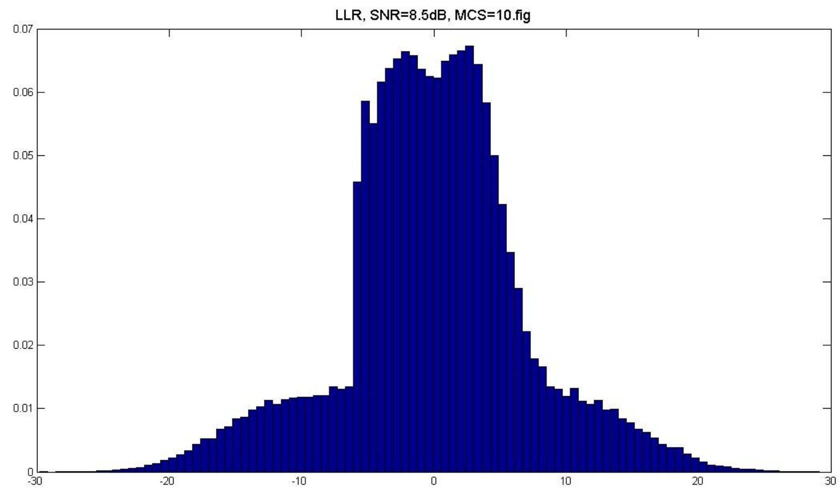
- Degradation in decoding performance

Behavior of Different Variables in LDPC Decoding

Dynamic Range of LLR values

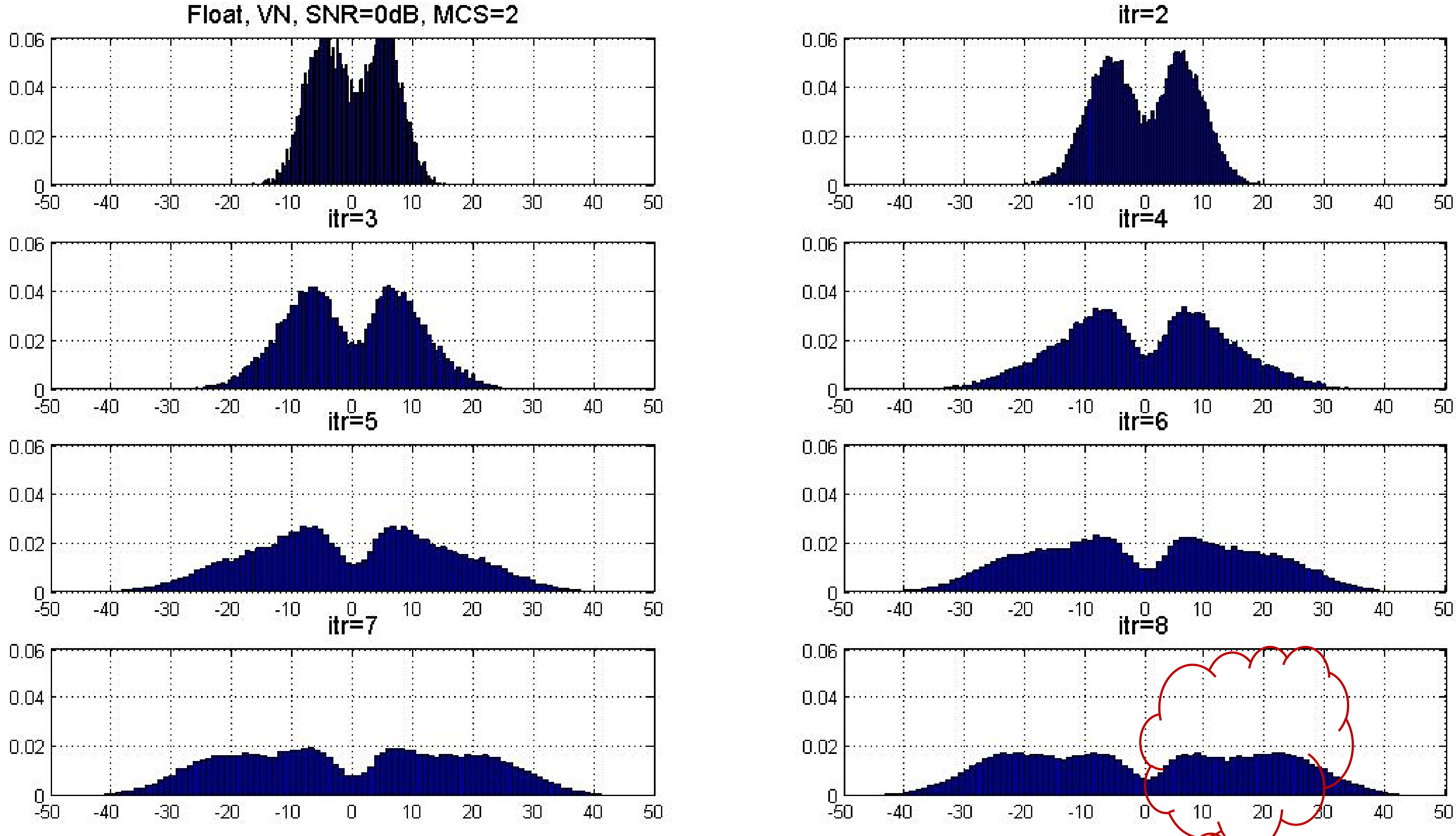






Floating-Point VN PDF

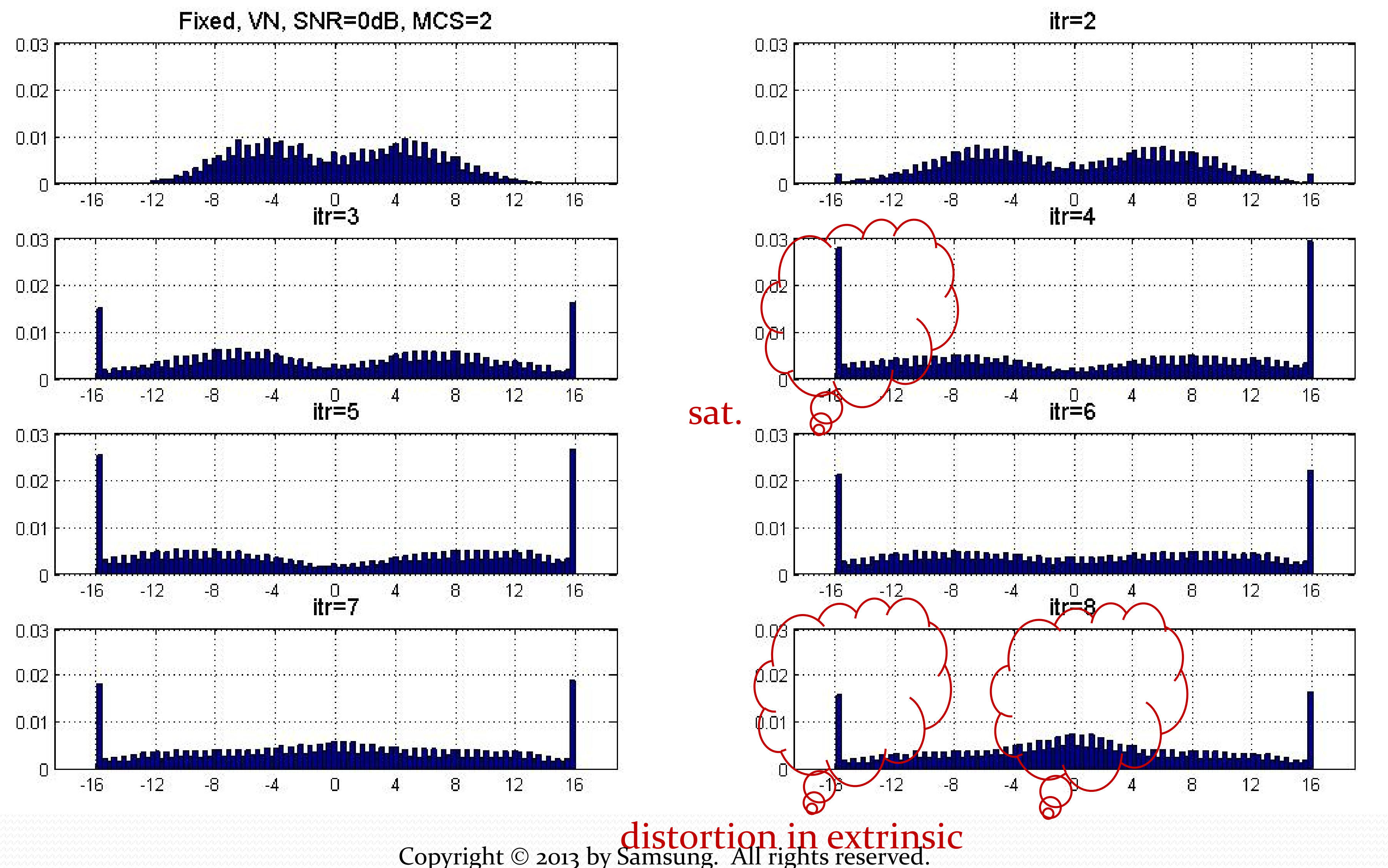
- PDF of the VNs with the same degree of 3.



growth in LLR values

Fixed-Point VN PDF (4.3)

- PDF of the VNs with the same degree of 3.



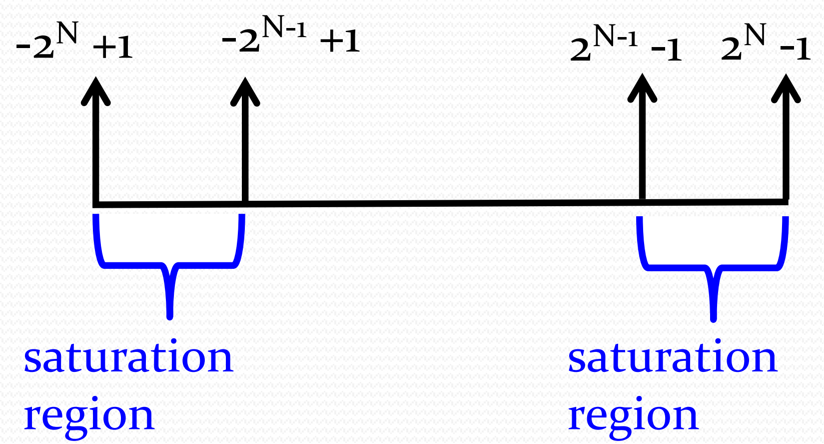
Proposed Fixed Point Algorithm

Saturation Regions

$$f_N(x) = \begin{cases} 1 & |x| \geq 2^{N-1} - 1 \\ 0 & |x| \leq 2^{N-1} - 2 \end{cases}$$ and $$Q_N(x) = \begin{cases} 2^{N-1} - 1, & x \geq 2^{N-1} - 1 \\ x, & |x| \leq 2^{N-1} - 2 \\ -2^{N-1} + 1, & x \leq -2^{N-1} + 1 \end{cases}$$

saturation indicator

quantization function



Proposed Algorithm

- Freezing the value of LQ (N+1 bits), when this value goes to the saturation regions.

$$LQ^v(k+1) = \begin{cases} LQ^v(k) & F^v(k) = 1 \\ LQ^v_{temp}(k) & F^v(k) = 0 \end{cases}$$

$$F^v(k) = f_N\left(LQ^v(k)\right)$$

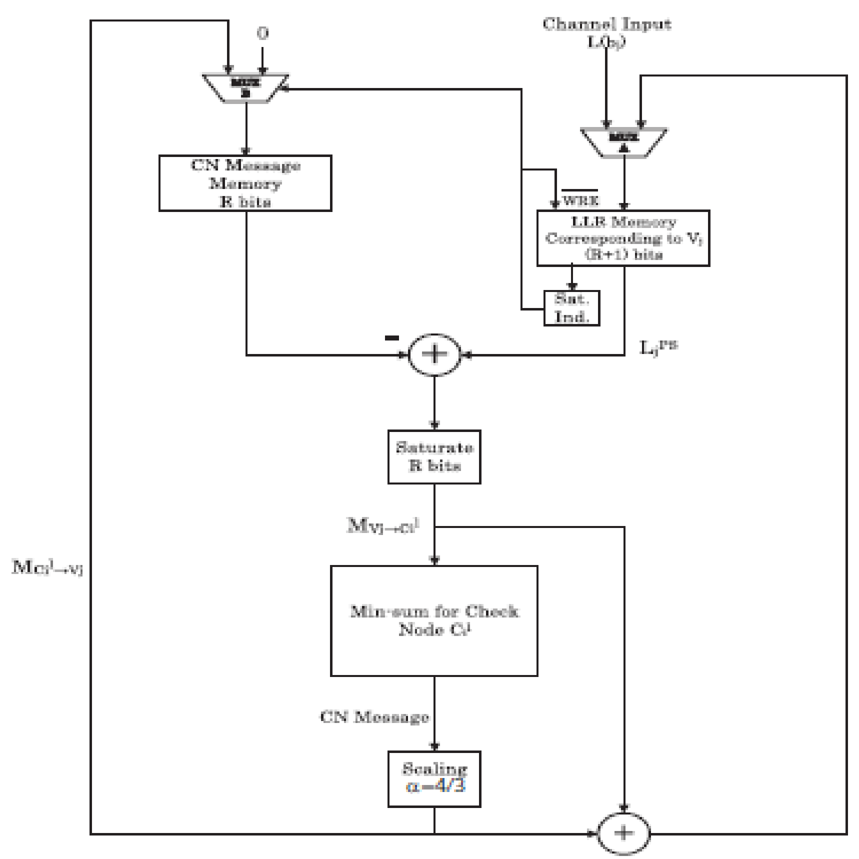
$$LQ^v_{temp}(0) = LLR^v$$

$$F^v(0) = 0$$

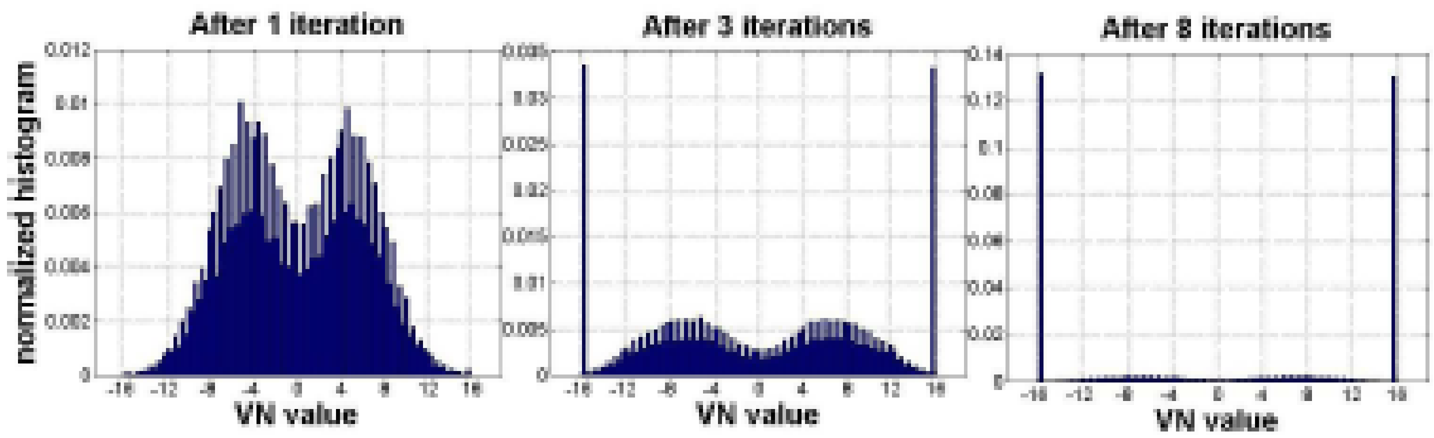
- No subtraction if LQ is in the saturation region.

$$M_{v\to c}(k) = \begin{cases} Q_N\left(LQ^v(k)\right) & F^v(k) = 1 \\ Q_N\left(LQ^v(k) - M_{c\to v}(k-1)\right) & F^v(k) = 0 \end{cases}$$

Proposed Freezing Mechanism



Fixed LDPC Saturation (QPSK 0dB)

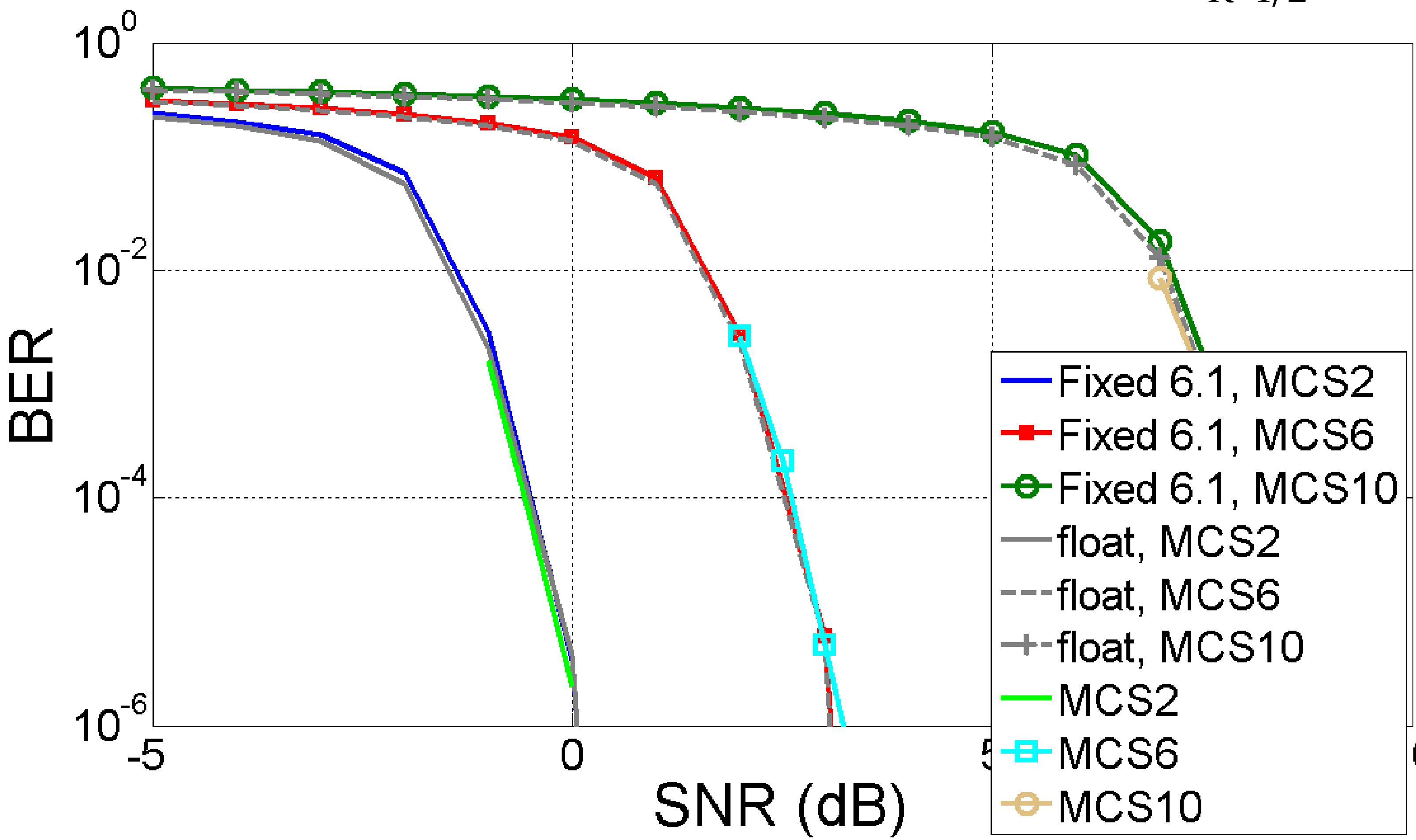


WGA MCS Table

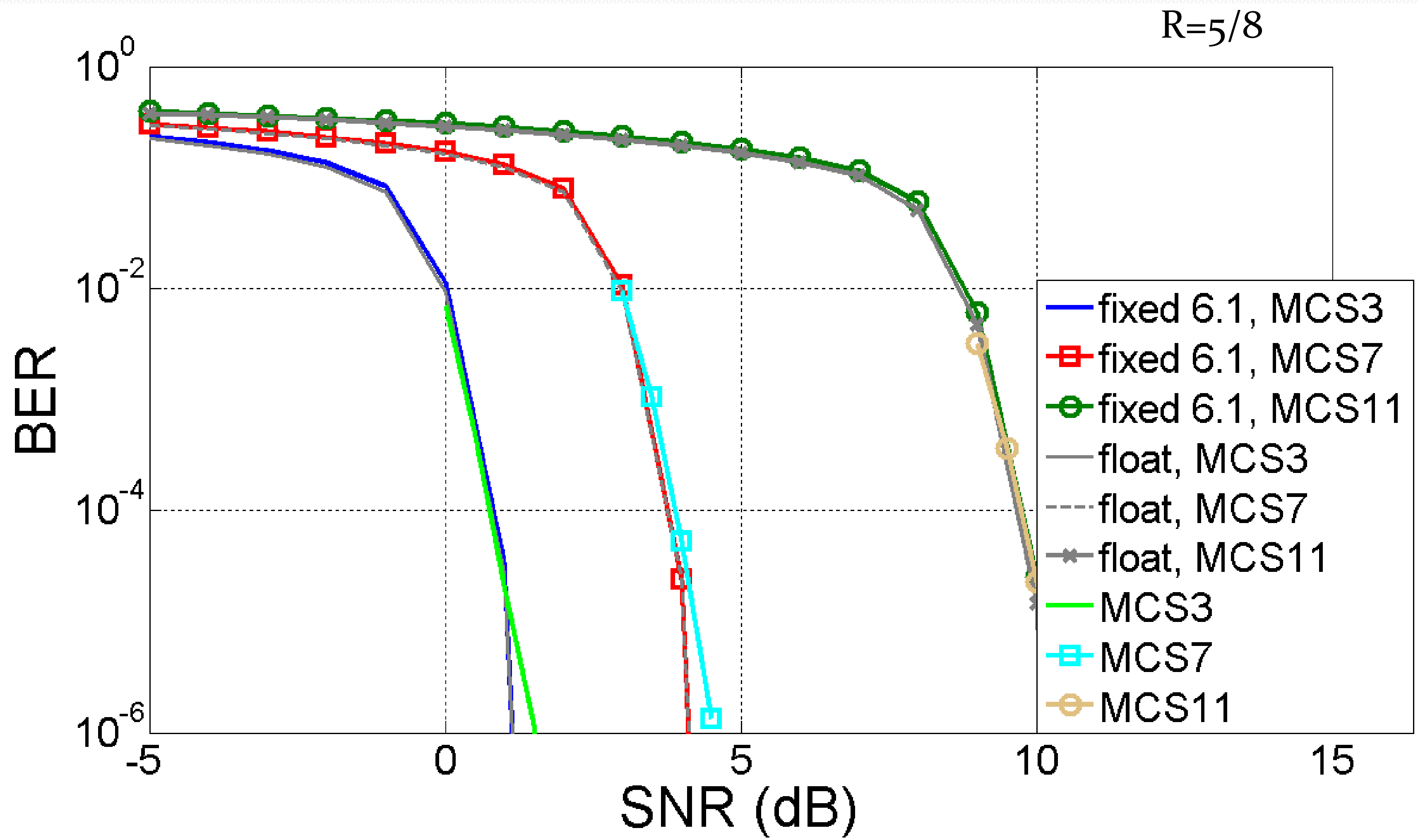
MCS	Modulation	Code Rate
1	BPSK	1/2
2	BPSK	1/2
3	BPSK	5/8
4	BPSK	3/4
5	BPSK	13/16
6	QPSK	1/2
7	QPSK	5/8
8	QPSK	3/4
9	QPSK	13/16
10	16QAM	1/2
11	16QAM	5/8
12	16QAM	3/4

Comparison between floating and modified fixed point MCS 2, 6 and 10

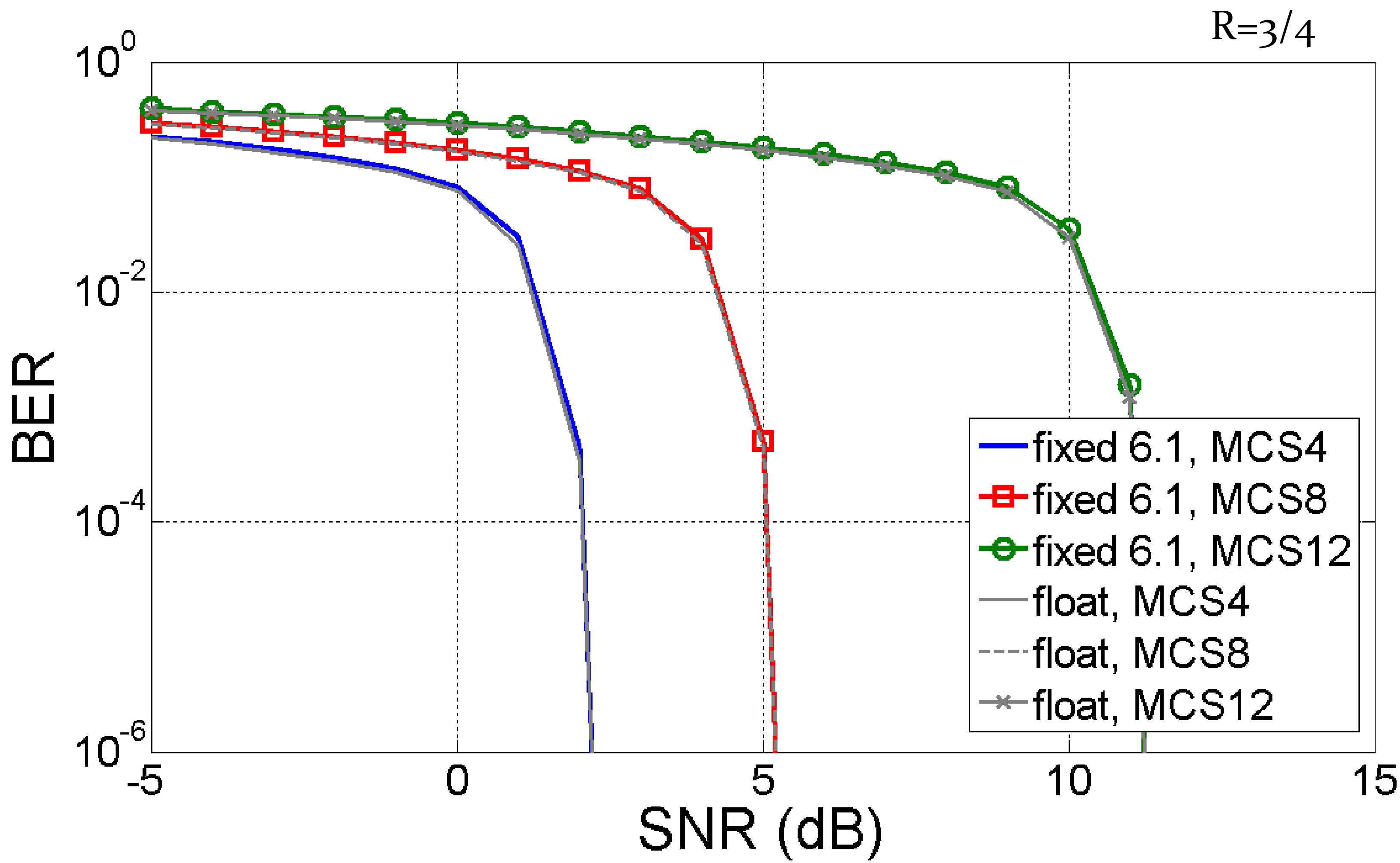
R=1/2



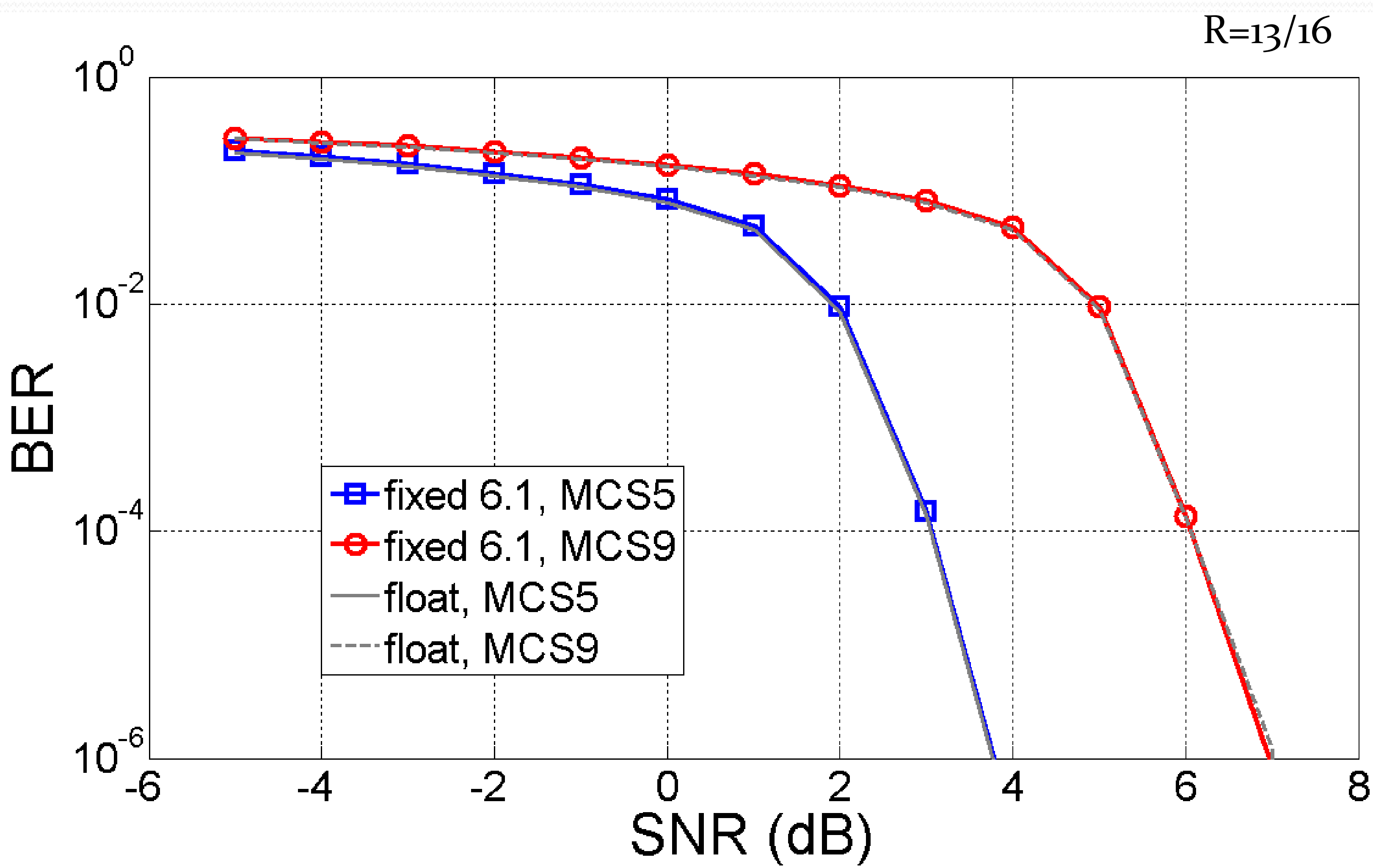
Comparison between floating and modified fixed point MCS 3, 7 and 11



Comparison between floating and modified fixed point MCS 4, 8 and 12

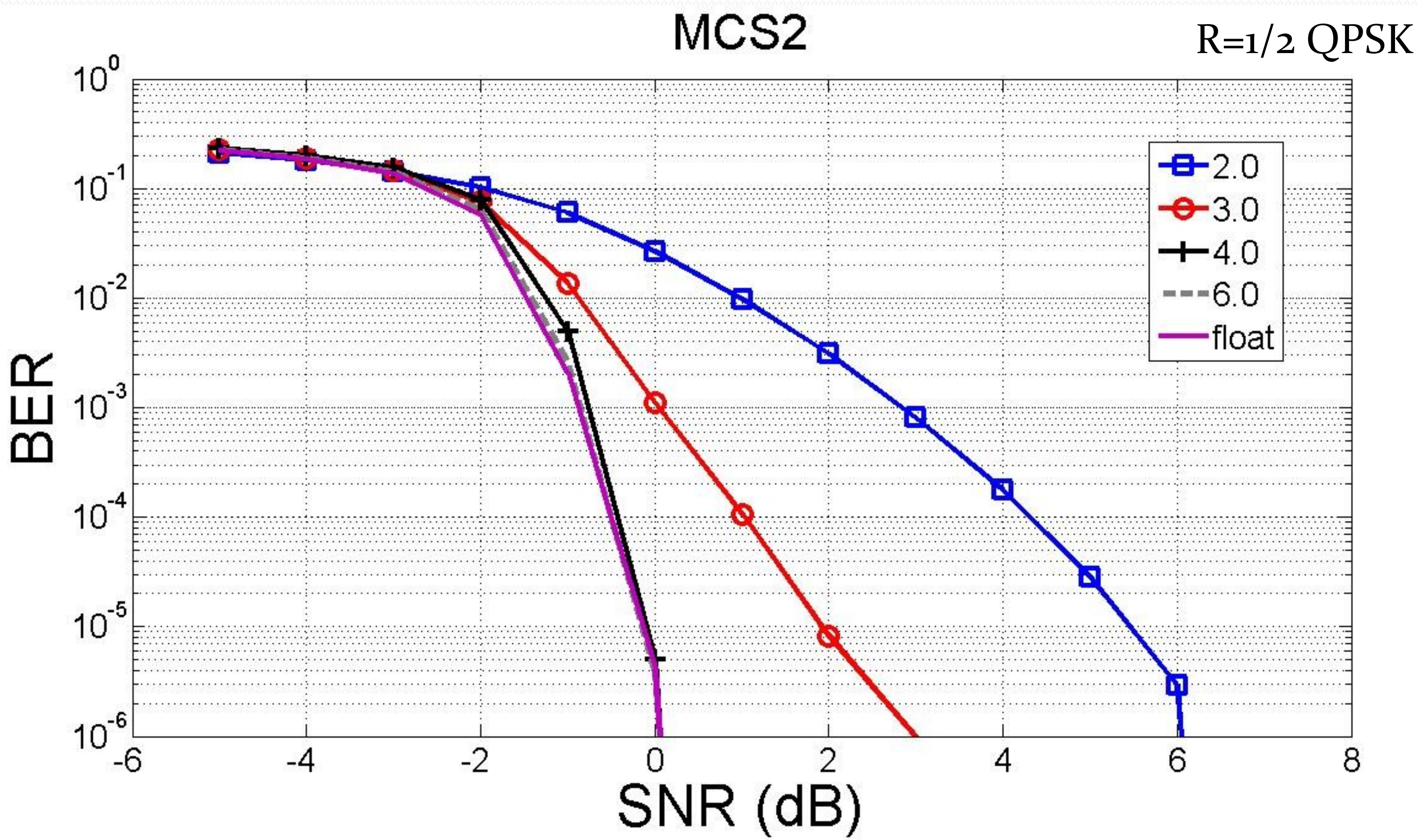


Comparison between floating and modified fixed point MCS 5 and 9



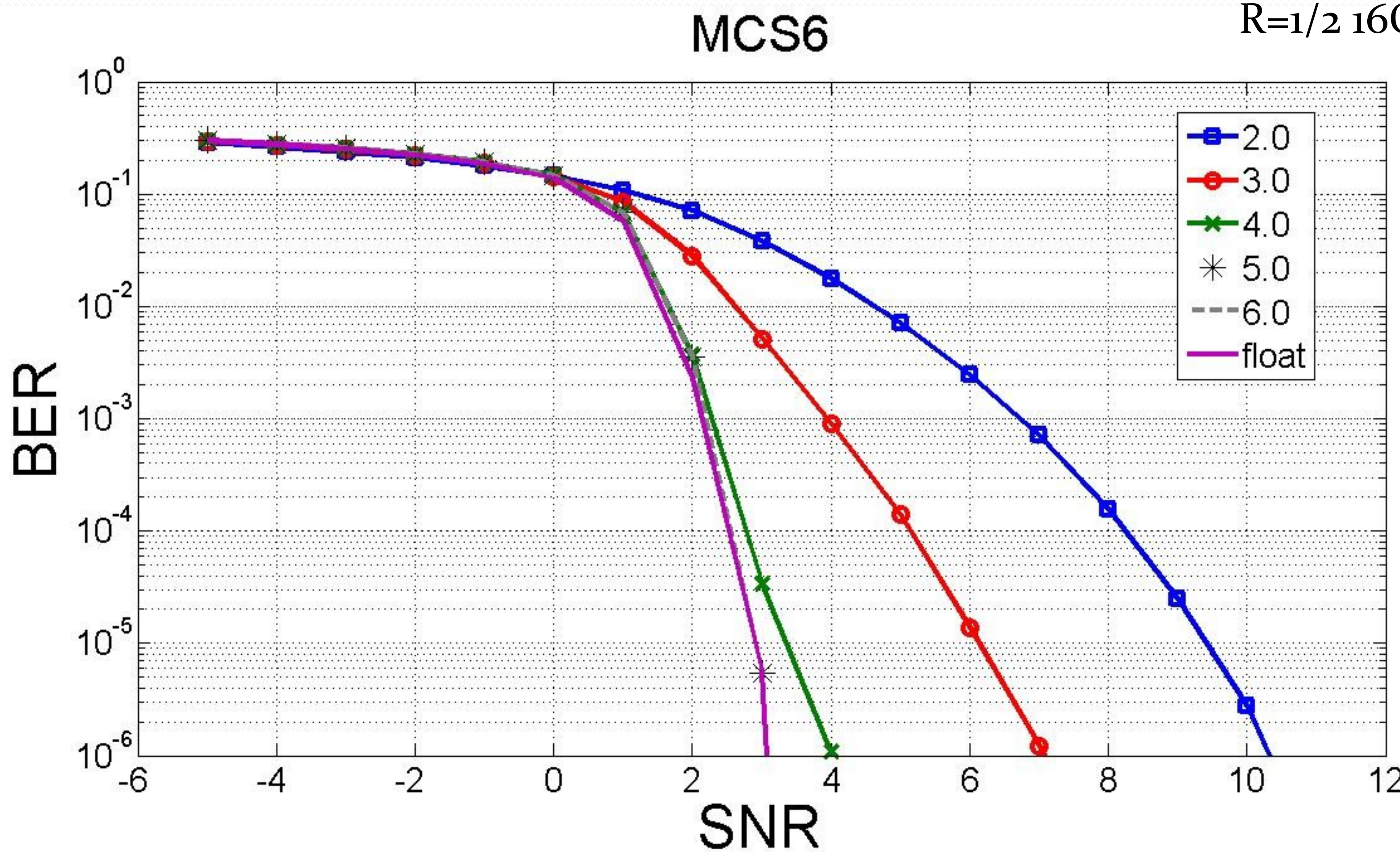
Precision Simulations

Precision simulations of modified fixed point for MCS2



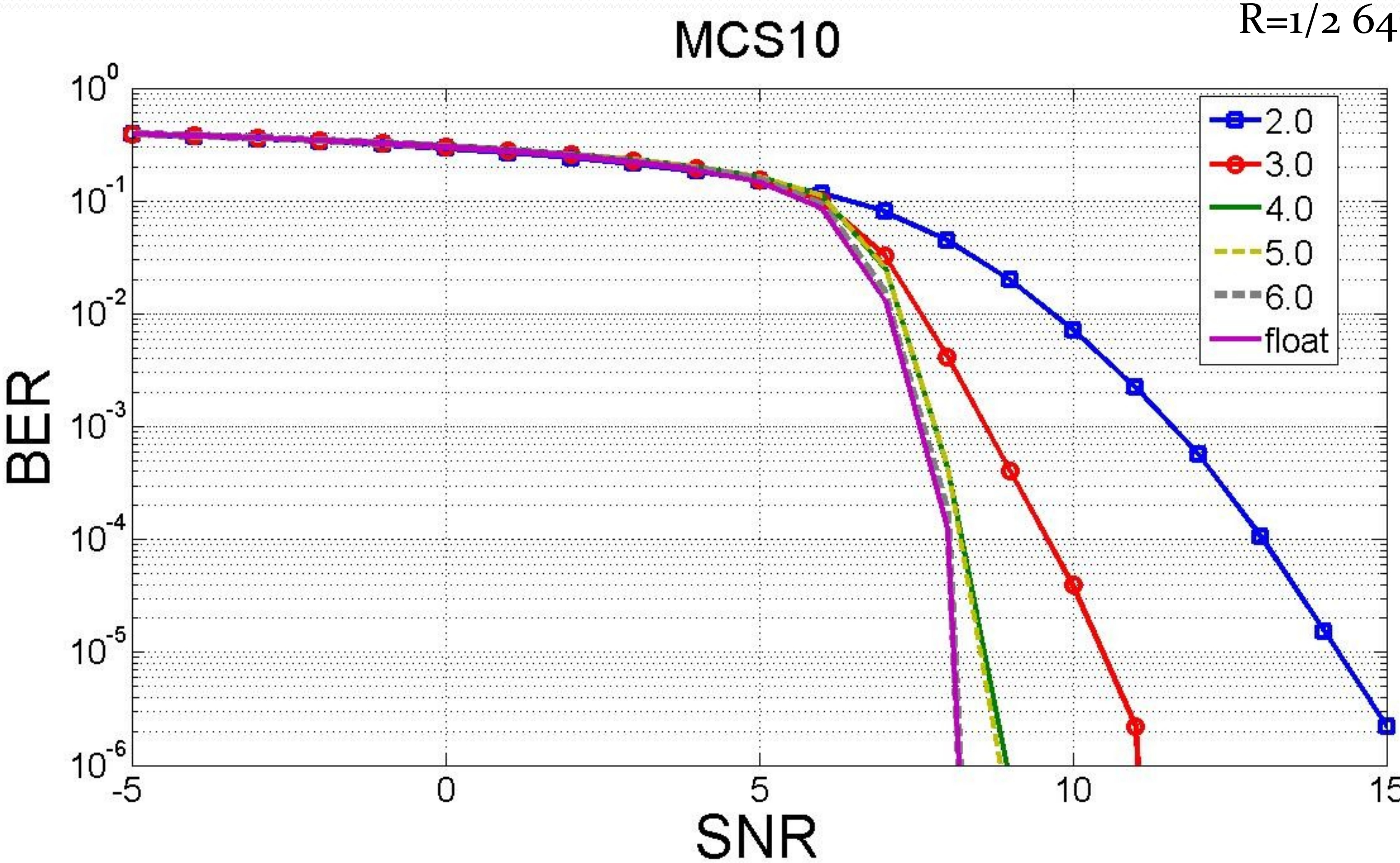
Precision simulations of modified fixed point for MCS6

R=1/2 16QAM



Precision simulations of modified fixed point for MCS10

R=1/2 64QAM



Summary

- LDPC codes introduced
- LDPC Decoding methods covered
- Saturation & Precision issues explained
- Proposed new Freezing-based mechanism

Thank You

Questions?