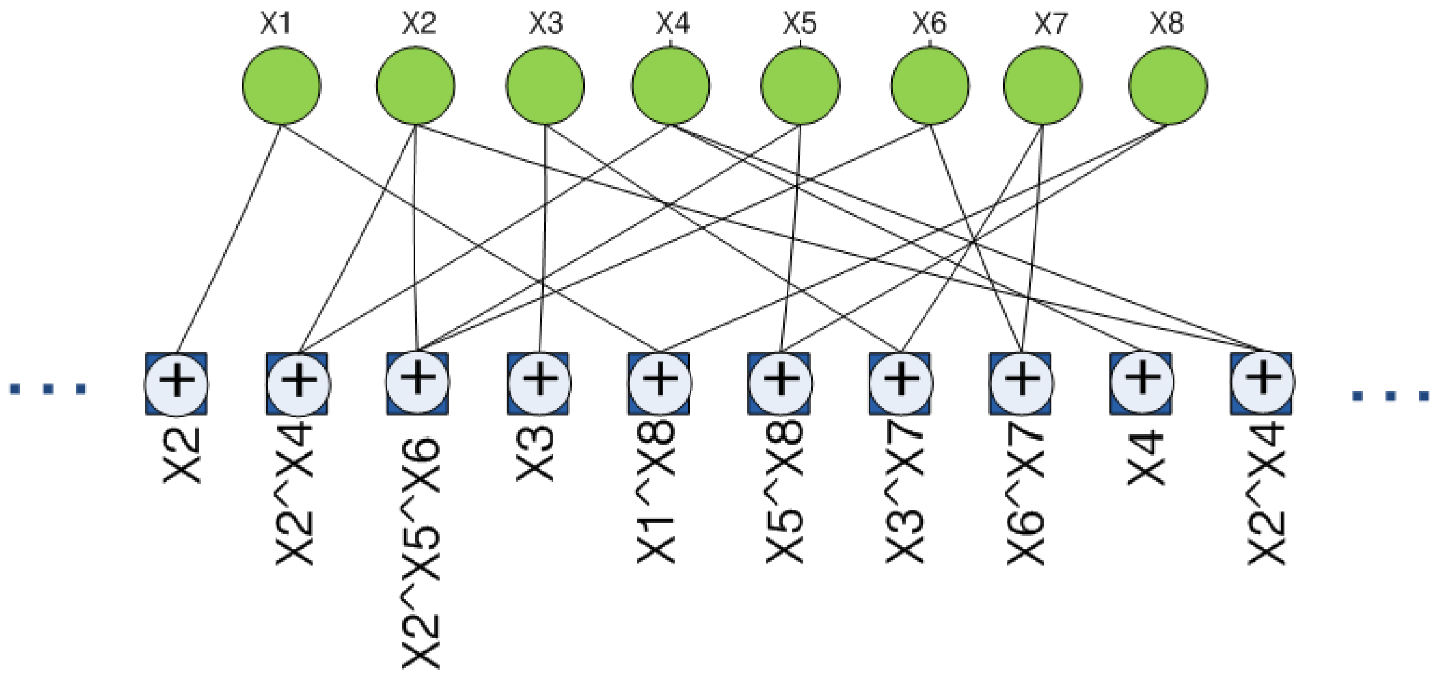
JCCI 2009

Column-filling Scheme for Luby-Transform Codes

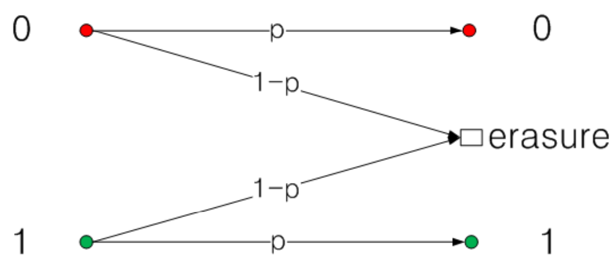
Coding & Crypto Lab.
YONSEI UNIVERSITY
남미영, 송홍엽
{my.nam, hysong}@yonsei.ac.kr

FOUNTAIN CODES



SYSTEM MODEL

- **NOTATION**
  - k : number of information symbols
  - n : number of output symbols
  - $\varepsilon = n/k - 1$: overhead
  - $H, |H|$: binary encoding matrix and the number of $1's$ in H
  - Complexity : number of the edges of the Tanner graph of LT code
- **Binary Erasure Channel**



Binary Erasure Channel

GOAL :

Universality - minimize the overhead

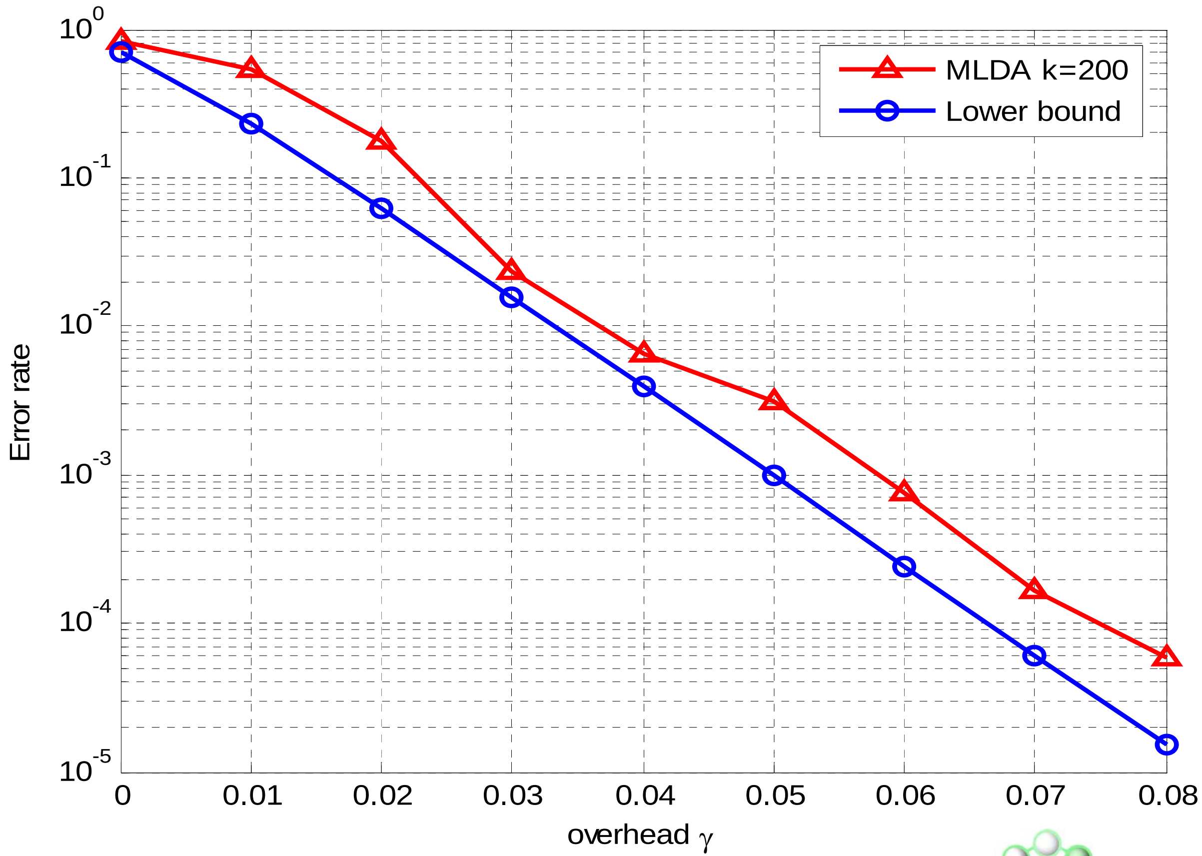
Efficiency - minimize the complexity

LT CODES

- Invented by M. Luby in 1998
- Output symbols should be generated by simple distribution – Robust Soliton Distribution
- First class of *universal* and almost *efficient* Fountain Codes
- Encoding and decoding are very simple

LT CODES

- WER performance of an LT code with k=200 using Maximum Likelihood Decoding Algorithm

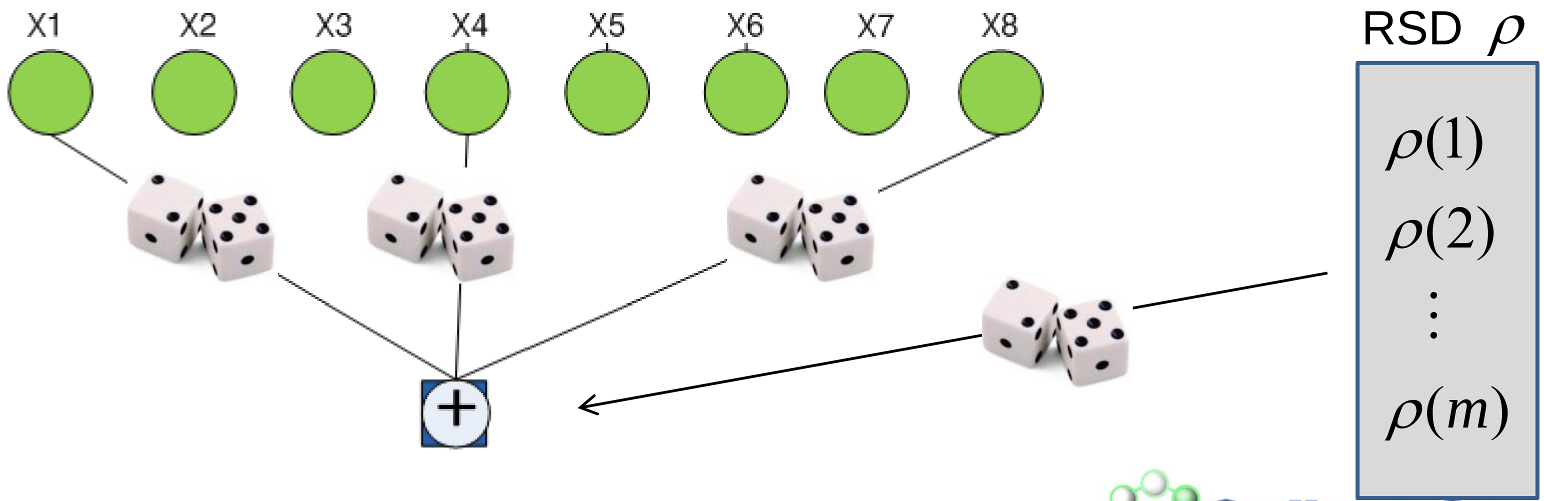


LT CODES

Encoding of LT Codes

Algorithm 1 A general LT encoding algorithm

repeat
  choose a degree d from degree distribution $\rho(d)$.
  choose uniformly at random d input symbol blocks $m_{i_1}, \ldots, m_{i_d}$.
  send $m_{i_1} \oplus m_{i_2} \oplus \cdots \oplus m_{i_d}$.
until enough output symbols are received.

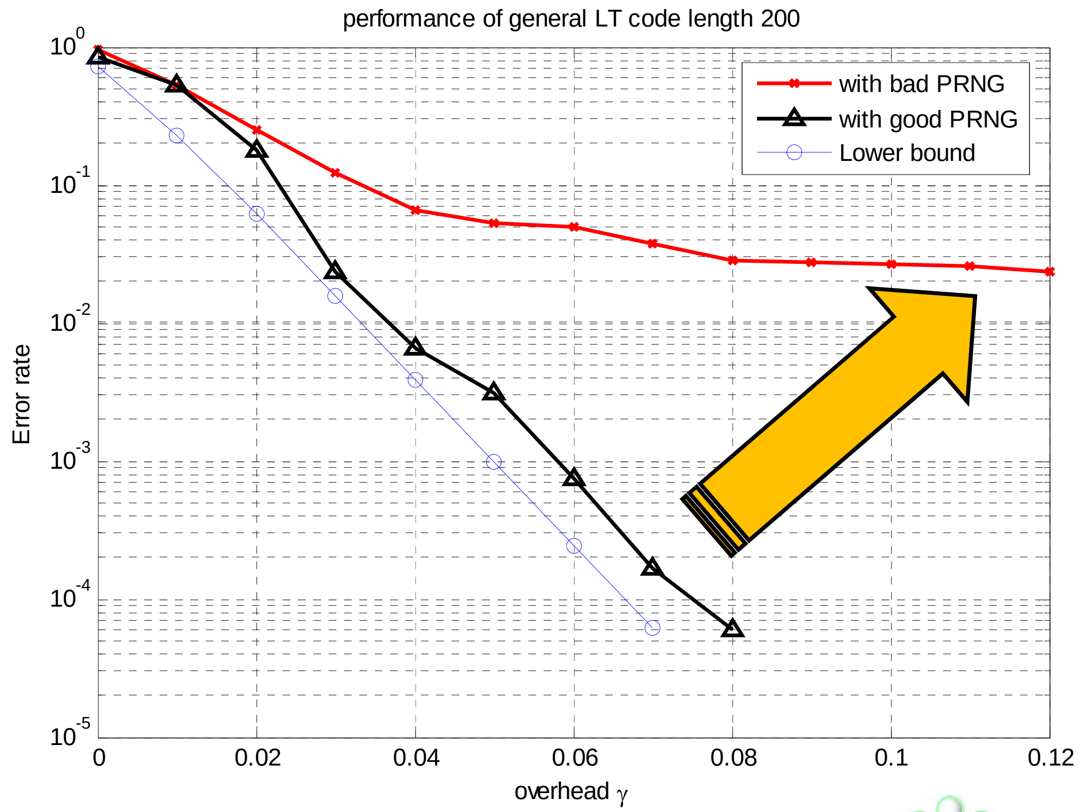


LT CODES

- The output symbol distribution follows RSD
- The input symbol should be selected uniformly at the same time
  - The number of selection of each input symbols should be equal.
- GOOD PRNGs are needed
  - Guarantee perfect uniformity
  - Allow flexible selection range

LT CODES

- The negative influence of non-uniform column weight distribution (k=200, using MLDA)



LT CODES

- WHY?
  - Null column effect – sometimes some input symbols never be chosen
    - ✓ These are never recovered

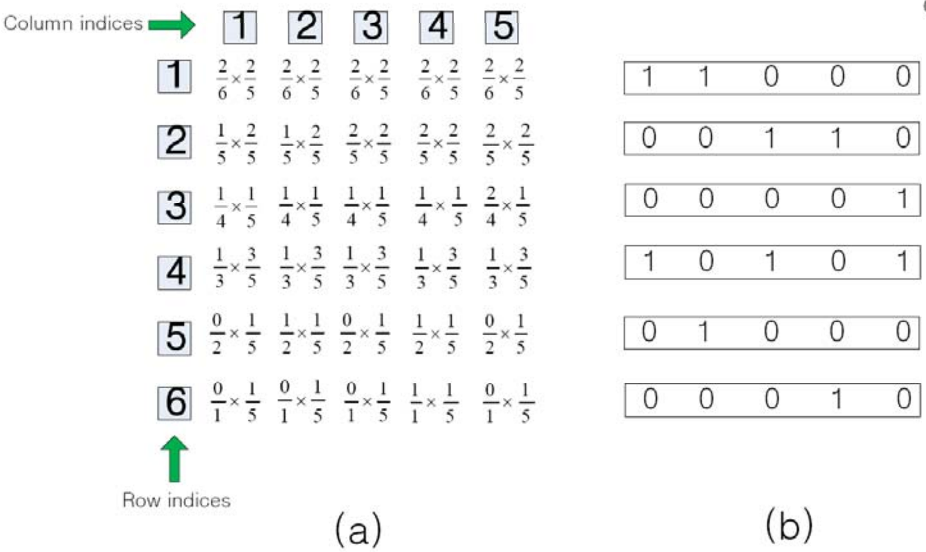
overheads	Null columns / Frame	overheads	Null columns / Frame
0.00	0.0796460177	0.04	0.0466507177
0.01	0.0821256039	0.05	0.0439461883
0.02	0.0714285714	0.06	0.0351380423
0.03	0.0501138952	0.07	0.0368344274

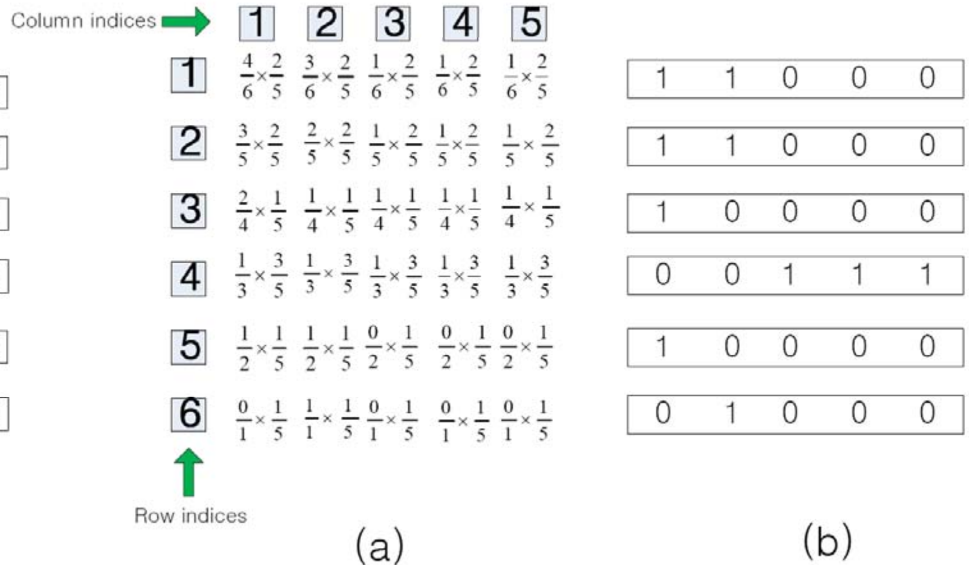
LT CODES

- decrease the full-rank probability
  - ✓ Example for $|H| = 10,\ k = 5,\ n = 6$

    row-weights = [2,2,1,3,1,1]

    column-weights = [2,2,2,2,2] vs. [4,3,1,1,1]





LT CODES

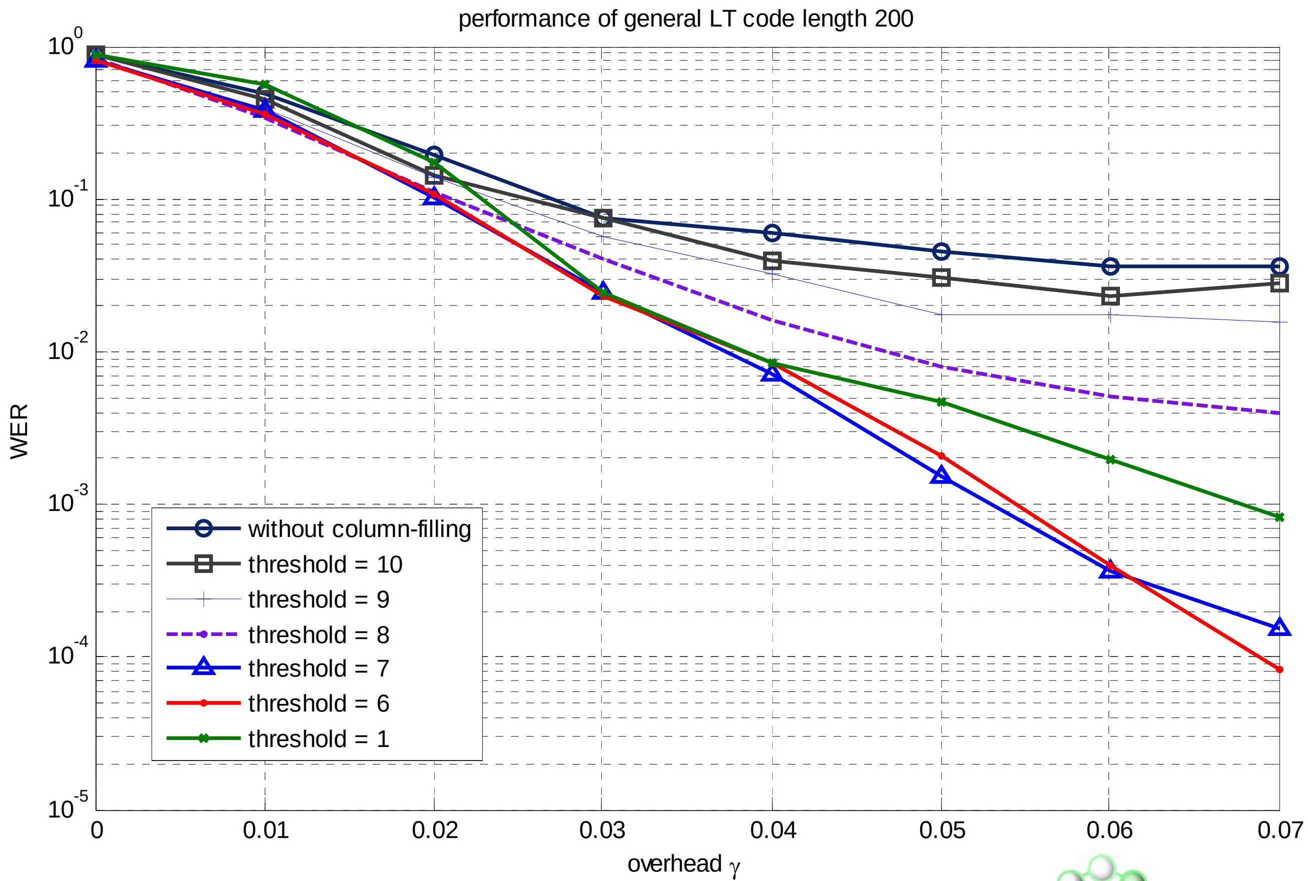
- Proposed algorithm
  - Just counting and control the number of selection of input symbols
  - No need for the PRNG guarantees very very uniform selection

Algorithm 2 An LT encoding algorithm together with the column-filling

repeat
  choose a degree d from degree distribution $\mu(d)$.
  for $j = 1$ to $j = d$
    repeat
      $count = count + 1$
      choose an input symbol block m_{i_j} at random.
       if $cw[i_j] \leq C_t$
        $cw[i_j] = cw[i_j] + 1$
        break
       end if
    until $count < K$
  end for
  send $m_{i_1} \oplus m_{i_2} \oplus \cdots \oplus m_{i_d}$.
until enough output symbols are received.

LT CODES

- **Performance comparison**



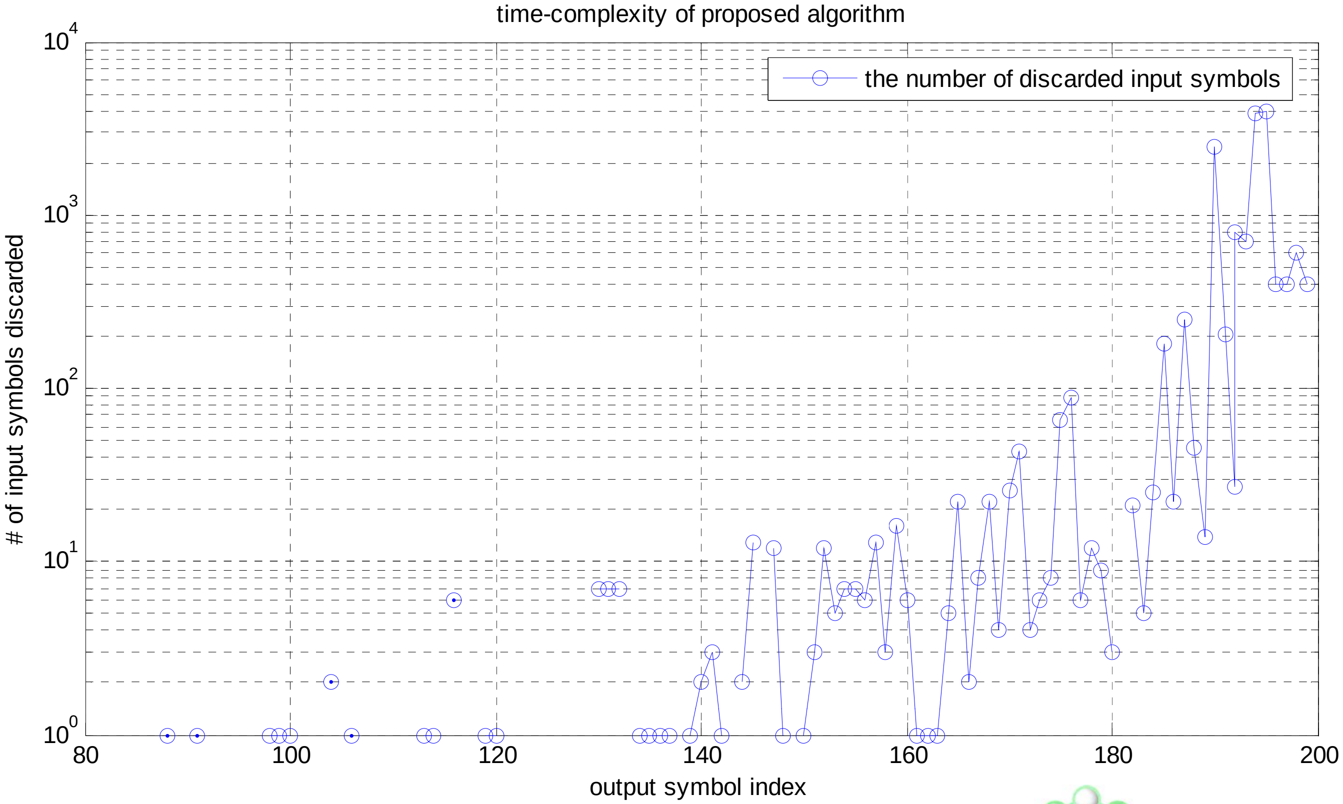
LT CODES

- Fraction of null columns

C_t \ ε	0.00	0.01	0.02	0.03	0.04	0.05	0.06	0.07
no	0.0796	0.0821	0.0714	0.0501	0.0466	0.0439	0.0351	0.0368
8	0.0248	0.0171	0.0168	0.0134	0.0080	0.0063	0.0047	0.0037
7	0.0082	0.0037	0.0010	0.0002	0	0	0	0
6	0	0	0	0	0	0	0	0
1	0	0	0	0	0	0	0	0

LT CODES

- Complexity comparison
  - The case of $C_t = 6,\ k = 200,\ \varepsilon = 0$



CONCLUSION

- The uniform column weight distribution of the encoding matrix for LT Codes is the best for the performance
- The proposed column-filling scheme is simple way to guarantee the uniform selection of input symbols
- However there exists some latency for applying the column-filling scheme
- More efficient way making the uniform distribution should be investigated

THANK YOU!

{my.nam, hysong} @ yonsei.ac.kr