Deterministic Interleaver Design for Turbo Codes

Bohulu Kwame Ackah, 1631133

Information Transmission Labroratory

February 1, 2018

1. Turbo Codes : Brief Introduction

- AWGN channel capacity approacing code
- Parallel concatenation of 2 convolutional codes via an interleaver
- Good performance depends on interleaver

2. Interleavers

- Divided into 2 groups
  - Random Interleavers
    - Advantage: Good performance for large frame sizes
    - Disadvantage: storage of interleaver tables required.
  - Deterministic Interleavers
    - Advantage : Interleaving done via algorithm
    - Disadvantage: For large frame sizes, interleaver better than random not yet found.
    - Design is very challenging

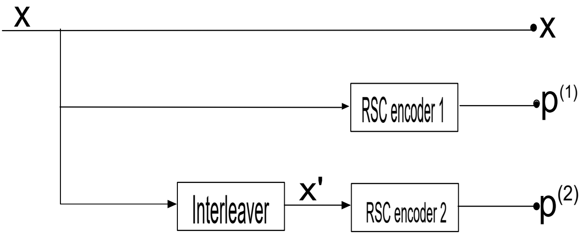
---

[5] Jing Sun, Oscar Y. Takeshita "Interleavers for Turbo Codes Using Permutation Polynomials over Integer Rings", IEEE Trans. Inform. Theory, vol. 51, pp. 101 - 119 Jan. 2005.

3. Purpose of Research

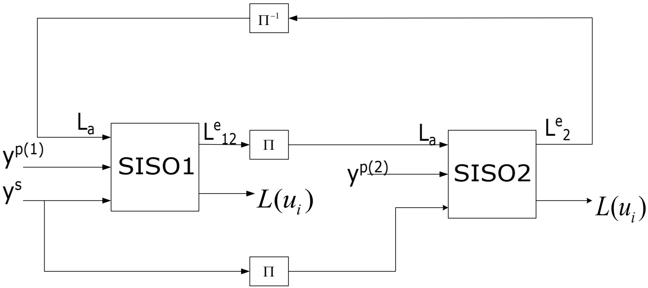
- Main Goal of Research
  - Deterministic interleaver that outperforms random interleaver for large frame sizes
- Current Step
  - Deterministic interleaver that outperforms linear interleaver for large frame sizes
  - multi-shift interleaver is proposed
- Why Linear Interleaver?
  - Better than random interleaver for short frame sizes.
  - Easy to design

4. Turbo Encoder



- N is interleaver size, M is No. of memory elements
- **x** is information bits with length $N - M$
- $\mathbf{p}^{(1)}$ is upper parity checkbits, $\mathbf{p}^{(2)}$ is lower parity checkbits
  - both have length N

5. Turbo Decoder

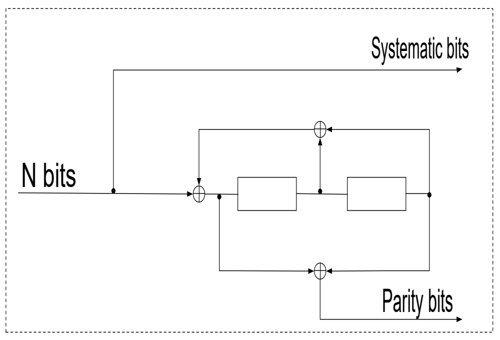


- y^s is systematic bits
- $y^{p(1)}$ is upper parity check bits, $y^{p(2)}$ is lower parity check bits
- $L(u_i)$ is Log-Likelihood Ratio, L^e_{12}, L^e_{21} is extrinsic information

6. RSC Encoders

- cycle length (τ) of RSC encoders
  - length of the cycle with input $[1, 0, 0, 0, 0, ...]$

Example



- $\left[\frac{1+D^2}{1+D+D^2}\right]$ (5/7) RSC Encoder
- output : $[1, |1, 1, 0|, |1, 1, 0|, |1, 1, 0|...]$
- cycle : $[1, 1, 0]$, $\tau = 3$

7. RSC Encoders and $a\tau$-seperated weight 2 errors

- weight 2 information sequences
  - "1" bit pair seperated by $a\tau - 1$ "0" bits

- effective free distance d_{eff}
  - minimum codeword weight due to weight 2 input

Example

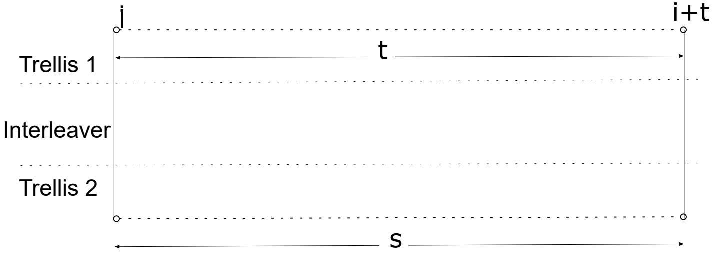
$$N = 16, \text{ input}=[1, 0, 0, 1, \ldots\ldots, 0]$$

$$1\;0\;0\;0\;0\;\ldots \qquad\qquad 1\;1\;1\;0\;1\;1\;0\;1\;1\;0\;1\;1\;0\;1\;1\;0$$

$$\underbrace{.\qquad}_{\tau}1\;0\;0\;0\;\ldots \qquad\qquad \underbrace{0\;0\;0}_{\tau}\;1\;1\;1\;0\;1\;1\;0\;1\;1\;0\;1\;1\;0$$

$$1\;1\;1\;1\;0\;0\;0\;0\;0\;0\;0\;0\;0\;0\;0\;0$$

8. t-seperated weight 2 error in Turbo Codes



- $t = a\tau \mapsto s = b\tau$, low-weight turbo codewords.

9. τ-seperated Weight 2 Error - based Parameter Optimization

- Linear Interleaver with depth D

$$\Pi_{\mathfrak{L}_N}(i) = Di \mod N, \quad gcd(N, D) = 1; \tag{1}$$

- input - output distance relationship

$$s = Dt \mod N \tag{2}$$

- prevent $t = a\tau \mapsto s = b\tau$
  - Solution: $\min(a+b)$ $,D$ with largest $\min(a+b)$

10. $a\tau$ weight 2 error : Interleaver Search

Linear Interleaver Search Results

D	13	121	17	23	21
a	19	17	15	11	12
b	9	9	1	3	4
d_{eff}	30	30	15	26	15
N_{free}	1	1	2	1	2

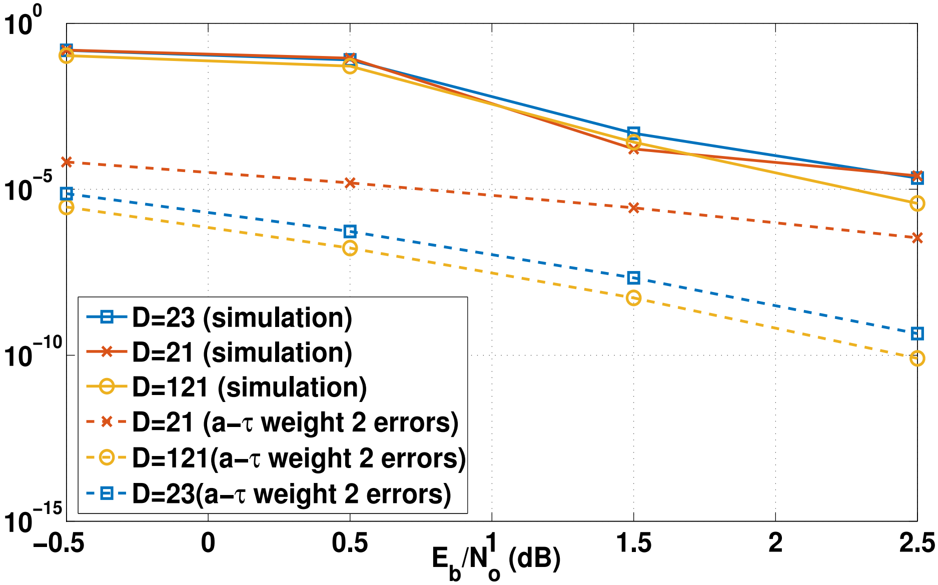
BER Approximation

$$P_b \approx \frac{1}{2} \sum_{w_c} D_{w_c} erfc\left(\sqrt{w_c \frac{R_c E_b}{N_o}} \right) \tag{3}$$

where

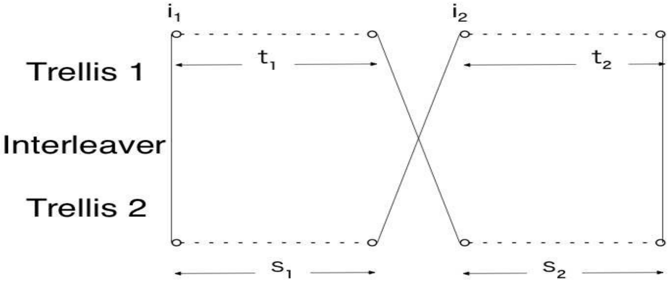
$$D_{w_c} \triangleq \sum_{w_x + w_p = w_c} \frac{w_x}{N} A_{w_x, w_p}$$

11. BER Approximation vs Simulation



12. τ-seperated weight 4 errors

- Dominate BER performance [2]



- Weight 4 input :$(1 + D^{v})(1 + D^{\tau})$
  - $\tau = Dv \mod N$

[2] Oscar Y. Takeshita, Member, IEEE, and Daniel J. Costello , "New Deterministic Interleaver Designs for Turbo Codes",IEEE Trans. Inform. Theory, vol. 46,pp. 1988-2006,Nov. 2000

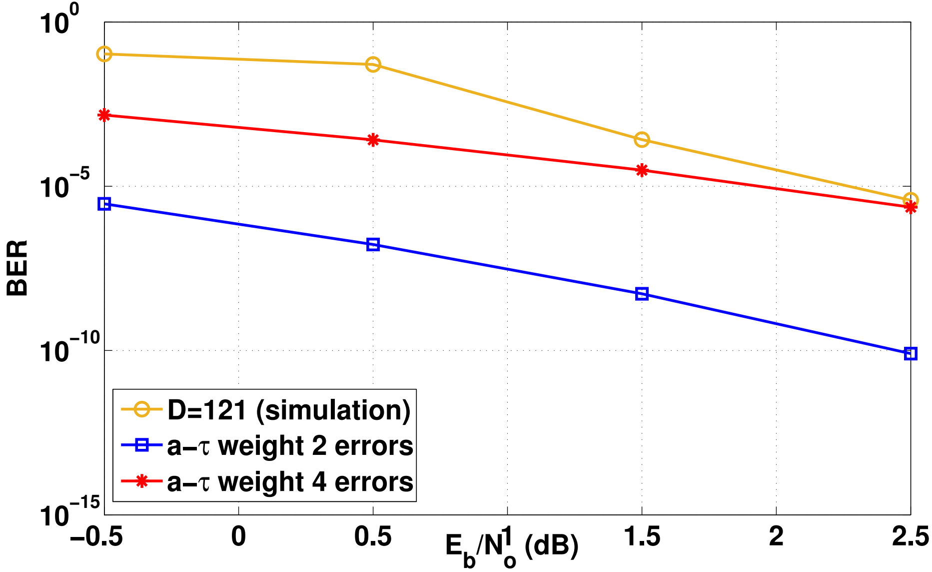
12. τ weight 4 errors

Example

- $N = 32$, $\tau = 3$, $D = 5$, $v = 7$
- input : $(1 + D^3)(1 + D^7)$, output :$(1 + D^3)(1 + D^{15})$

- codeword weight : 20, multiplicity $\approx N$
- same results for different D and N

13. τ weight 4 error : BER Approximation vs Simulation



14. Sequential representation of Linear Interleaver

- Weight 4 error dominates performance
- Algorithm for linear interleaving
  - 1. $p_0 = 0$
  - 2. $p_i = (p_{i-1} + D) \mod N$
- element positions shifted by constant D

p_i is element position

15. Multi-Shift Interleaver

- For $N = 2^r, \;\; r \in \{1, 2, ...\}$ set $\Delta s = 2^q, \;\; q \in \{2, 3, .., r-1\}$
- cycle set $\mathbb{D} = \{d_0, d_1, ..., d_{V-1}\}, \;\; V = N/\Delta s$
  - $d_i = d_{i-1} + \Delta s$
- Algorithm for multi-shift interleaver
  - 1. $p_0 = 0$
  - 2. $p_i = p_{i-1} + d_{((i-1) \mod V)} \mod N$, $\;\; d_0$ *is an odd integer*
  - Shift value of D for each position shift

Example

- original vector

$$\mathbf{x} = [0, 1, 2, 3, ..., 31]$$

- $N = 32, \ \ \Delta s = 4, \ \ d_0 = 5$
  - $\mathbb{D} = \{5, 9, 13, 17, 21, 25, 29, 1\}$
  - $$\mathbf{p} = [0, 5, 14, 27, 12, 1, 26, 23, 24, 29, 6, 19, 4, 25, 18, 15, 16, 21 \\ , 30, 11, 28, 17, 10, 7, 8, 13, 22, 3, 20, 9, 2, 31]$$
  - interleaved vector:

$$\mathbf{x}' = [0, 5, 30, 27, 12, 1, 10, 23, 24, 29, 22, 19, 4, 25, 2, 15, 16, 21, 14, 11, 28 \\ , 17, 26, 7, 8, 13, 6, 3, 20, 9, 18, 31]$$

16. Optimal Parameter Search for Multi-Shift Interleaver

- procedure for choosing good interleavers
  - choose d_0 from $(\sqrt{N}, N/2)$
  - calculate hamming weight for $\Delta s \in 2^q$
  - best Δs = largest d_{eff}
  - repeat for d_0 within range
  - best parameter, $(d_0, \Delta s)$ with largest d_{eff}, least value of Δs and multiplicity

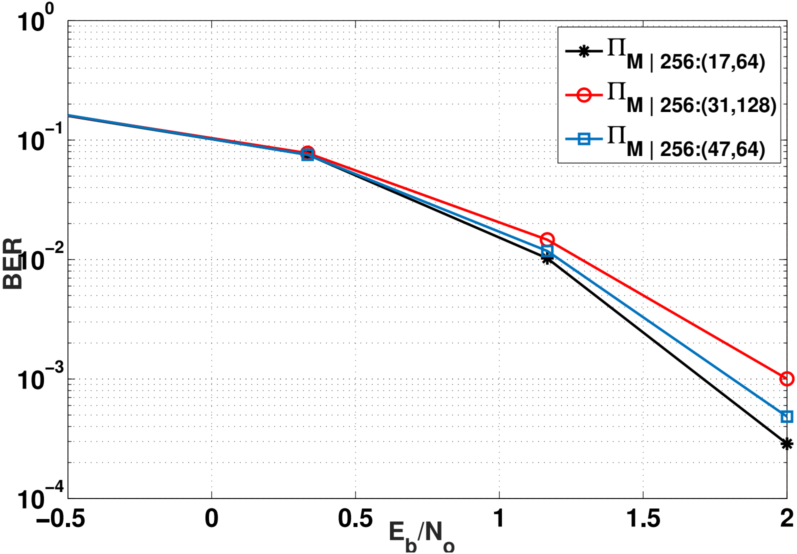
17.MSI Search Results : 5/7 component encoder. $N = 256$

d_0	17	31	47
d_{eff}	38	38	38
Δs	64	128	64
$N_{free,eff}$	207	208	209

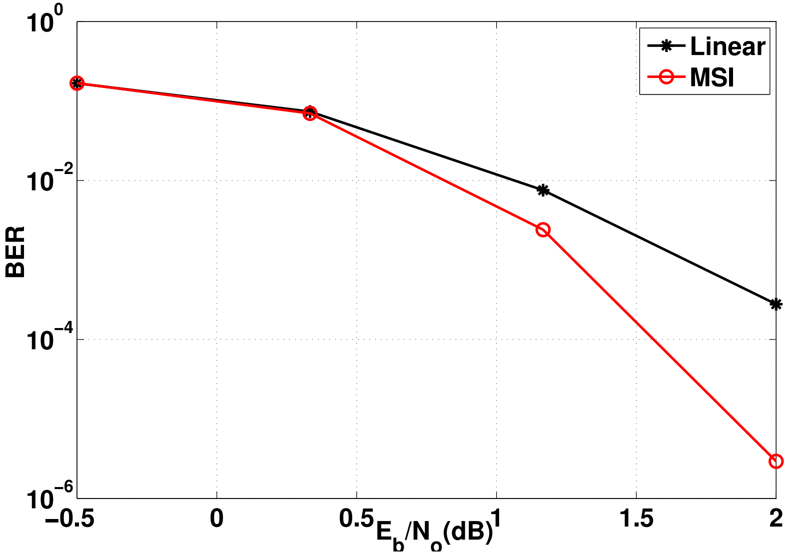
- best parameter $(d_0 = 17, \Delta s = 64)$

$N_{free,eff}$: multiplicity of d_{eff} codewords

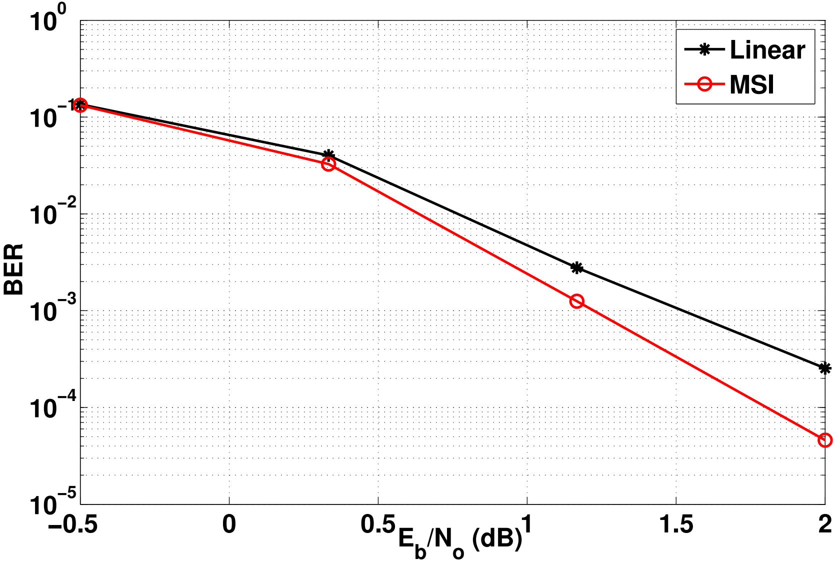
17. Simulation Results for Table



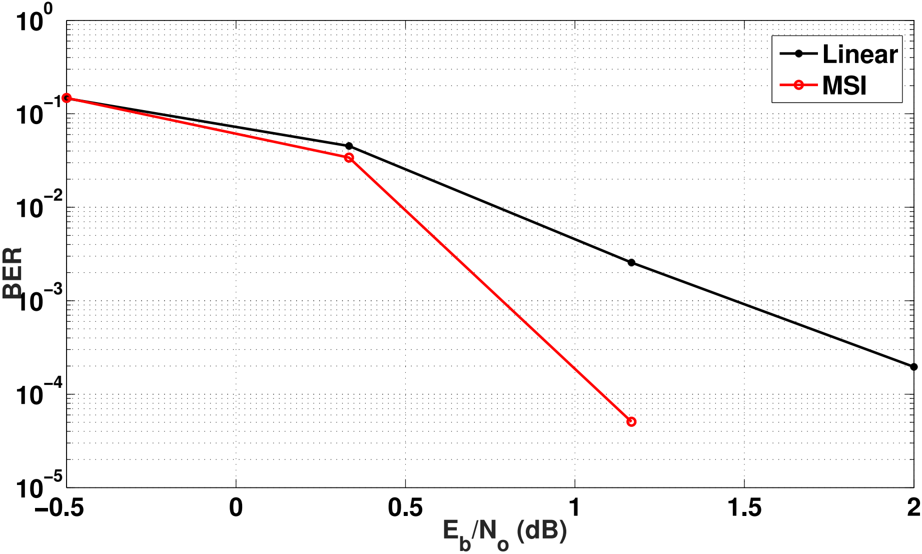
17. Results for 5/7 Component Code. $N = 1024$



17. Results for 7/5 Component Code. $N = 1024$



17. Results for 5/7 Component Code. $N = 16384$



18. Conclusion and Future Works

Conclusion

- The multi-shift interleaver proposed.
- Easier design than linear interleaver
- Better performance than linear interleaver for medium and long frame sizes.

Future Research

- Deterministic interleaver that outperform random interleaver
- Good interleaver design using BER bounds approach
- LDPC code application