

S-Distance Based Deterministic Block Parallel Interleaver for Turbo Codes

Yong Chen, Lina Wang

Department of Communication Engineering, School of Computer and Communication Engineering,

University of Science and Technology Beijing, Beijing, China

kaola830812@163.com, wln_ustb@126.com

Abstract

In this paper, an improved deterministic block parallel interleaver based on S -distance for Turbo codes is designed according to the theory of Bubble Sort. The aim is to overcome the long delay of S -random interleaver. The improved interleaver has the larger spreading factor ' S ', lower delay and better bit error rate performance, as well as it requires less memory hardware. Simulation results verify the effectiveness of the S -distance based interleaver for Turbo codes.

Keywords: Turbo codes; parallel interleaver; S-distance

Introduction

Turbo codes were proposed by Berrou in 1993. Due to the idea of random coding and iterative decoding, the error correction performance of Turbo codes come closer to approaching Shannon's limit [1, 2]. Turbo encoder includes two component encoders, interleaver, puncturing matrix and multiplexer [3]. In general, component code is chosen as Recursion System Convolutional code (RSC) using the same generator matrix [4]. Interleaver is an important part of encoding and decoding structure for Turbo codes. The idea of random coding is realized by constructing long codes from simple short codes with interleaver. In order to overcome the long delay of S -random interleaver, an improved deterministic block parallel interleaver based on S -distance is proposed according to the theory of Bubble Sort. The performance of the improved interleaver is demonstrated by simulations.

The remainder of the paper is organized as follows. Section II introduces the pseudorandom interleaver and S -random interleaver. Section III describes the improved interleaver based on S -distance. Section IV provides the simulation

results and the analysis of the proposed interleaver. Finally, the conclusions are given in Section V.

Preliminary

Interleaver has an important role in encoding and decoding of Turbo codes. Design simple and useful interleavers is critical for Turbo codes. Its main function is to increase the distance between the two RSC subcode for encoding, and to transmit burst errors in the previous iteration into random errors for decoding with the cause of reducing the correlation of iterative decoding outputs. There are two classical intertleave, namely pseudorandom and S -random intertleave. Because of the pseudorandom intertleave's simple arithmetic, it is widely applied, but the performance is not good. S -random intertleave has better performance, but the delay will increases exponentially with increasing of interleaving length.

Pseudorandom Intertleave. Pseudorandom intertleave is randomly generated mapping. Here's how it works: suppose the interleaving length is N , first mark the position of each bit of information, then a random number is generated between 0 and 1, which is corresponding to every position, and then rearrange the size of the random number. Last read the corresponding position index of each information data, and the position index sequence is mapping table.

S -random intertleave. A pseudorandom intertleave with distance limit S becomes the S -random intertleave which was firstly proposed by D.Divsalar [5]. Suppose the interleaving length is N . Its interleaving process is as follows.

Step 1. Define the matrix $A = [1, 2, 3, \dots, N]$, $B_{1 \times N} = [0, 0, 0, \dots, 0]$, and distance limit S .

Step 2. Choose a random number from matrix A to write to matrix B sequentially, and the numbers of whose are written base on the criterion that each number is not within $\pm S$ of any the previously written S numbers. If any of the previously written S numbers is within a distance S of the current position, then the random number is rejected and choose again. Once the random number is written successfully, delete it from matrix A .

Step 3. The process of Step 2 is repeated until all N numbers of matrix A are written to matrix B , the end matrix B is mapping table.

Step 4. If the rest numbers of matrix A don't meet the criterion of distance limit S , go back to Step 1 and repeat.

The size of S value is crucial importance to decide whether mapping table can be generated successfully. In general, The appropriate value range of S is $\sqrt{N/4} \leq S \leq \sqrt{N/2}$ [6], and if $S > \sqrt{N/2}$, it is difficult to generate mapping table. In general, $S > \lfloor \sqrt{N/2} \rfloor$ (round down). Intuitive sense of the distance limit S is that the distance value between two information bits before interleaving is less than S value, but after interleaving it must be greater than or equal to S value. S -random interleaver has the advantage of performance characteristics that is very excellent, but it has the drawback of requiring larger time delay with increasing the S value, as well as it is impossible to generate mapping table for each S value. When interleaving length is larger, the above-mentioned process could lead to long delay.

S-Distance Based Deterministic Block Parallel Interleaver

In this section, an improved deterministic block parallel interleaver is proposed according to the theory of Bubble Sort. The improved interleaver is based on S -distance and an iteration construction. The algorithm process is as follows.

Step 1. Determine the number of block. According to interleaving length N , break it into m rows and n columns, usually m is far less than n , then define $TH=10N$ as the maximum times of Bubble Sort switching, and $CZ=5N$ as the maximum times of reset matrix $C_{m \times n}$, and distance limitation S . Besides, TH and CZ can be adjusted according to circumstance of the block.

Step 2. Initializes the variables, let the times of bubble sort switching $th=0$, and the times of reset matrix $cz=0$.

Step 3. Set $cz=cz+1$. If $cz=CZ$, then set $S=S-1$ and return to Step 2. If $cz < CZ$, then generate n random permutations from 1 to m , and sequentially write these numbers into the matrix $B_{1 \times N}$ defined as the row index number, while define the matrix $M_{1 \times m}$ as the column index number. For example, the value of $M_{1 \times m}(3)$ shows that current search position is the third row and the $M_{1 \times m}(3)$ column of the matrix $C_{m \times n}$.

Step 4. First write the N numbers $[1, N]$ to matrix $A_{m \times n}$ according to the line sequentially, then with performing S -random interleaver arithmetic on n

numbers of each row, the matrix $C_{m \times n}$ can be obtained, this ensures that each row numbers meet the $\lfloor \sqrt{n/2} \rfloor$ distance limit. Initialize the variable, let $M_{1 \times m} = \overrightarrow{1_{1 \times m}}$.

Step 5. Let $i=1$, loop until $i=N$, then find the row index number $B_{1 \times N}(1)$, set $t = B_{1 \times N}(1)$, and write number $C_{m \times n}(t, 1)$ to the mapping matrix $D_{1 \times N}(1, 1)$, set $M_{1 \times m}(t) = M_{1 \times m}(t) + 1$; while $i > 1$, find the row index number $B_{1 \times N}(i)$, set $t = B_{1 \times N}(i)$, then find out number $C_{m \times n}(t, M_{1 \times m}(t))$ in matrix $C_{m \times n}$, and determine whether number $C_{m \times n}(t, M_{1 \times m}(t))$ meet the condition that each number is not within $\pm S$ of any the previously S numbers written into matrix $D_{1 \times N}$. If meeting the condition, set $D_{1 \times N}(i) = C_{m \times n}(t, M_{1 \times m}(t))$, $i=i+1$ and $M_{1 \times m}(t) = M_{1 \times m}(t) + 1$. If not meeting condition, the number sequence in t row from $M_{1 \times m}(t)$ to the end performs cyclic shift to the left, then determine whether the new number $C_{m \times n}(t, M_{1 \times m}(t))$ meet the condition. If the rest of the all numbers does not meet condition, go to Step 6.

Step 6. Let $th=th+1$. If $th=TH$, set $th=0$, and return to Step 3; if $th < TH$, remove the rest of the all numbers not meeting condition to the front of the t row, set $i=0$ and $M_{1 \times m} = \overrightarrow{1_{1 \times m}}$, then return to Step 5.

Step 7. End loop.

This improved interleaver algorithm can quickly generate mapping table with S distance limit. It actually takes advantage of the S -random interleaver algorithm for shorter information frame length to generate mapping table within a short time. Suppose the interleaving length is N , first divide information sequence position indexes into some smaller blocks, and each block performs S -random interleaver arithmetic. It guarantees that each row numbers meet the smaller S distance limit. Then the number meeting the condition of bigger distance limit S is searched in the whole sequence and the bubble sort is used during the row search. The design can reduce the unnecessary circulation and judgment, thereby reducing the delay.

Simulation Experiments and Results

In this section, the S -distance based deterministic block parallel interleaver for Turbo codes was evaluated. We compared the improved interleaver with pseudorandom and S -random interleavers through the simulation experiments.

Firstly, the distance characteristic was studied. Fig. 1 depicted the position of input and output with interleaving length 256. The output position of pseudorandom interleaver was irregular, and a few points overlapped. It showed that the distance of neighboring information bit is also very close after interleaving. The discrete effect could not be ensured by pseudorandom interleaver. The output position of S -random interleaver was decentralized and irregular. And no points overlapped because of distance limit S . S -random interleaver had the better interleaving effect. the improved interleaver has the similar performance with the S -random interleaver. However, the improved interleaver has the better decentralized and scrambling effects.

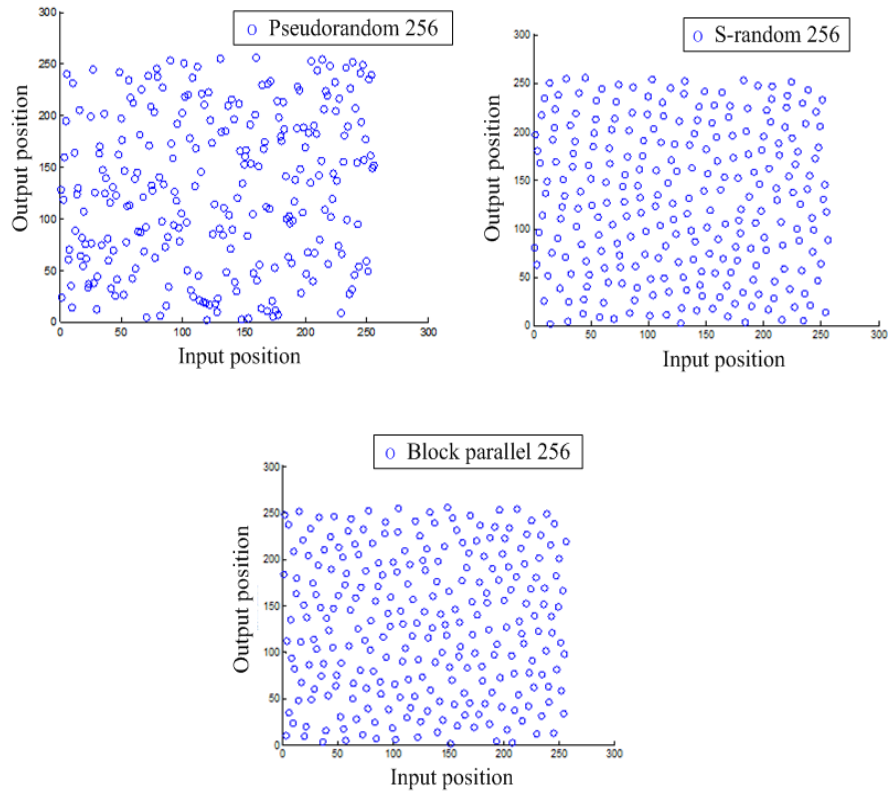


Fig. 1. Input/output position of three different types of interleavers with interleaving length 256

Secondly, the time to generate mapping table was calculated under different interleaving length. The simulation experiments run 300 times. The average delay of three different interleavers was shown in Table 1.

Table 1 The average delay of three different interleavers

Type Length	pseudorandom interleaver's delay (s)	S-random interleaver's delay (s)	improved interleaver's delay (s)
256	0.00005	3.229	2.091
512	0.00006	52.378	5.341
1024	0.0001	496.162	16.567

From Table 1, it can be found that the time to generating mapping table of the improved interleaver is greatly reduced compared with that of S-random interleaver. Especially, this advantage is more and more obvious with the increasing of interleaving length.

The bit error rate (BER) is the most important metric for interleavers. Therefore, the BER performance of three interleavers was evaluated. The simulation parameters are shown in Table 2.

Table 2 Simulation parameters of different interleavers for Turbo codes

Generator matrix	(11, 13)
Decoding algorithm	Log-MAP
Interleaving length	512
Iterations	5
Bit rate	1/3

The simulation result was shown in Fig. 2. It can be seen that the performance of the improved interleaver is superior to that of pseudorandom interleaver with the increasing of signal-to-noise ratio (SNR). And its performance approaches to that of S-random interleaver. However, the improved interleaver need less storage space.

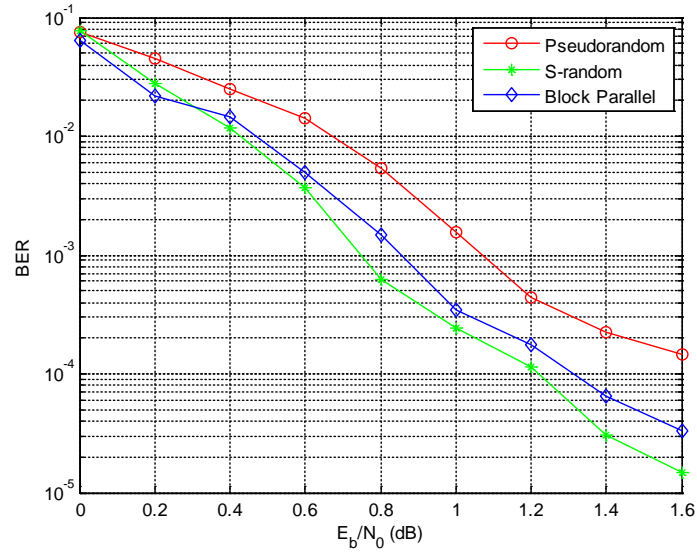


Fig. 2. The effect on performance with different interleaver algorithm

Conclusions

The improved deterministic block parallel interleaver based on S -distance is proposed in this paper. It has the advantages of a simple algorithm, excellent performance and low delay. The performance of the improved interleaver outperform the pseudorandom interleaver and needs less memory hardware. Consequently, the improved interleaver has good application prospect.

Acknowledgements

This work was supported by the Fundamental Research Funds for the Central Universities (FRF-TP-14-050A2).

References

- [1] Berrou C, Glavieux A, Thitimajshima P. Near Shannon limit error-correcting coding and decoding: Turbo-codes. Proceedings of ICC, Geneva, Switzerland, 1993, 1064-1070.

- [2] Berrou C, Glavieux A. Near optimum error correcting coding and decoding: Turbo-codes. IEEE Transactions on Communications, vol.44, No. 10, 1996, 1261-1271.
- [3] Donghua Liu. The theory and application of Turbo code. Beijing Publishing House of Electronics Industry, 2004
- [4] Shu Lin, Daniel J. Costello. Error control coding. Beijing: Mechanical industry press, 2007.
- [5] D. Divsalar, F. Pollara. Turbo codes for PCS applications. IEEE International Conference on Communication (ICCC 95), Seattle, Gateway to Globalization, 1995, 5: 54-59.
- [6] Kashif N K, Jinat R, Rameswar D. An improved interleaver design for Turbo codes. International Conference on Information and Communication Technology, 2007, 131-134.