Hindawi
Security and Communication Networks
Volume 2021, Article ID 8872140, 13 pages
https://doi.org/10.1155/2021/8872140

Research Article

Hardware Sharing for Channel Interleavers in 5G NR Standard

Xiaokang Xiong, Yuhang Dai, Zhuhua Hu, Kejia Huo, Yong Bai, Hui Li, and Dake Liu

School of Information and Communication Engineering and State Key Laboratory of Marine Resource Utilization in South China Sea, Hainan University, Haikou 570228, China

Correspondence should be addressed to Zhuhua Hu; eagler_hu@hainu.edu.cn and Yong Bai; bai@hainanu.edu.cn

Received 9 April 2020; Revised 16 June 2020; Accepted 7 January 2021; Published 27 January 2021

Academic Editor: Huaming Wu

Copyright © 2021 Xiaokang Xiong et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Interleaver module is an important part of modern mobile communication system. It plays an important role in reducing bit error rate and improving transmission efficiency over fading channels. In 5G NR (5th Generation New Radio) standards, LDPC (low-density parity-check) and polar channel codes are employed for data channels and control channels, respectively. If multiple interleavers are implemented separately for them, the cost increases significantly. To address this issue, a hardware multiplexing scheme for channel interleavers based on LDPC and polar codes is proposed in this paper. Firstly, the formulas for the processes of the control channel interleaving and data channel interleaving are derived with respect to 5G NR standard. Then, the hardware implementation structures of the two interleavers are given. Subsequently, hardware reuse is proposed by sharing the similar or identical parts between the two hardware structures. Simulation results verify the correctness of our proposed scheme and demonstrate that it can realize the hardware sharing of the two kinds of channel interleavers to reduce the cost of silicon.

1. Introduction

In the modern mobile communication, some important technologies are used, such as interleaving [1], offloading [2], spectrum sensing [3, 4], partitioning [5], hardware reusing, and resource sharing and allocation [6, 7]. Specifically, channel interleaving technology has been widely used. Channel interleaving aims to distribute transmitted bits in time to achieve desirable bit error distribution to counter the effects of fading channels. The interleaver can change the permutation of the signal bit stream to the utmost without changing the information content. Therefore, interleaver can maximize the dispersion of continuous error bits generated by bursts in the process of transmission. In this way, the error correction and error detection capabilities of the receiver can be improved. In the traditional LUT (lookup table) based interleaving and deinterleaving scheme, a large amount of silicon is used with high cost. Therefore, it is important to reuse the hardware for different types of interleavers to reduce the cost of silicon.

At present, the hardware equipment based on multimode and fast-switching has been studied for channel interleavers in WLAN (wireless local area network, which includes IEEE 802.11a/b/g and IEEE 802.11n standards), WiMAX (Worldwide Interoperability for Microwave Access, which includes IEEE 802.16e standard), 3GPP-WCDMA (3rd Generation Partnership Project-Wideband Code-Division Multiple Access), 3GPP-LTE (3GPP-long-term evolution), and DVB-T/H (Digital Video Broadcasting-Terrestrial/Handheld) standards [1]; multistandard hardware interleaver structure was proposed for HSPA (High Speed Packet Access) evolution, 3GPP-LTE, WiMAX, WLAN, and DVB-T/H in [8]. A parallel architecture for decoding reconfigurable interleavers was proposed to support HSPA evolution, DVB-SH (DVB-Satellite Services to Handhelds), 3GPP-LTE, and WiMAX standards [9]. The issues of address conflicts for hardware sharing were analyzed and resolved in [10]. Among these multistandard interleaver implementations, it is common to simplify and improve the interleaver algorithm of various standards such that the hardware implementation structure becomes simple and easy to reuse [11]. Then, the identical hardware structure is reused by careful comparison to reduce the cost of silicon for multistandards [12]. Although these works cover 2G, 3G, and even 4G standards [13], the latest 5G standard has not

been studied in them. Therefore, with respect to the 5G NR standard 3GPP TS 38.212 [14], this paper proposes a scheme of hardware reuse and cost-saving for polar-encoded channel interleaver [15] and LDPC-encoded channel interleaver [16]. We first derive the formulas for the interleaving schemes of data channel and control channel in 5G NR standard. Then, we design the corresponding hardware structure for them. Next, by comparative analysis, we obtain a multiplexing structure with a reused module to achieve the hardware sharing of two-channel interleavers.

The contributions of this paper are as follows:

(1) The interleaving schemes of data channel and control channel in 5G NR standard are formulized, and the corresponding hardware structures are given.

(2) The hardware structure diagrams of two kinds of channel interleaver are compared, and the hardware sharing structure is given to realize low-cost implementation.

The structure of the remaining parts of this paper is as follows: in Section 2, we introduce the interleaver schemes for LDPC and polar codes channel. In Section 3, we derive the interleaving formulas of two kinds of channels to facilitate the subsequent interleaver reuse. In addition, we give the hardware structure designs of two interleavers. Then, according to the derived formulas, we also give the hardware structure after hardware sharing. Subsequently, the feasibility verification of the final design is given. Finally, Section 4 summarizes the work of this paper.

2. Brief Introduction of Channel Interleavers in 5G NR Standard

In this paper, our work is mainly carried out in accordance with the final standard of 3GPP R15, which is the first version of the 5G standard and meets the part of IMT-2020 (International Mobile Telecommunications-2020) requirements of ITU (International Telecommunication Union). The interleaving method used in the 5G standard is the optimal conclusion after repeated discussion and demonstration [14, 17].

Channel interleaving mainly includes two modes: control channel interleaving and data channel interleaving. This paper focuses on the hardware sharing of these two interleaving methods in 5G NR uplink and downlink. The position of our work in the 5G NR standard is highlighted in Figure 1.

2.1. Interleaver for Data Channel. LDPC code is a new type of error correction code. Its performance in mobile channel is improved compared with turbo code. Even without interleaver, the error correction ability of irregular LDPC code is better than turbo code. Therefore, the LDPC code is listed as one of the candidate schemes in 5G communications. In addition, the simulation results show that LDPC has good performance in all block lengths and code rates, and the complexity is relatively low [18]. In the latest 5G standard, the construction, coding, and interleaving scheme of parity matrix H of LDPC code is specified. In 5G standard, QC-LDPC (quasi-cyclic-LDPC) code is adopted. QC-LDPC code belongs to a structured irregular LDPC code [19], which is composed of basic matrix H_b and lifting factor Z. In 5G standard, two basic matrices (i.e., BG1 and BG2) are determined. Two basic matrices have eight basic matrices, respectively, and they have different dimensions. The corresponding basic matrix [20] is selected according to the size and code rate of transmission block [21]. After the basic matrix is determined, the lifting factor is selected, and then, the basic matrix is modified according to the lifting factor to get the modified parity matrix H. Finally, according to the check matrix H, the encoded code word is directly obtained.

In essence, interleaver is a device which can change the information distribution structure without changing the information content. It is employed to make the burst errors generated in the process of channel transmission decentralized. The LDPC code interleaving scheme adopted in 5G standard is bit interleaving with block interleaver [22]. As shown in Figure 2, the interleaving method is to read the input sequence into a matrix by rows and then read out by columns. The process of deinterleaving is the opposite operation, i.e., read the interleaved sequence into the matrix by columns and then read it out by rows. The matrix is determined by the length and interleaving depth of the input sequence. The number of rows in the matrix is the interleaving depth, and the number of columns is the length of the input sequence divided by the interleaving depth. The interleaving depth is related to the modulation order. There are five modulation schemes specified in 5G NR standard, i.e., BPSK (binary phase shift keying), QPSK (quadrature phase shift keying), 16QAM (quadrature amplitude modulation), 64QAM, and 256QAM. The corresponding modulation orders are 1, 2, 4, 6, and 8, respectively. For example, if 16QAM modulation is used and the input sequence length is 8000 symbols, then the matrix size is 4×2000. After adding the interleaving function, the coding performance has a corresponding improvement, as shown in Figure 3.

2.2. Interleaver for Control Channel. Due to its low complexity of encoding and decoding, the polar code has become a research hotspot of error correction code. The core of polar code construction is related to the channel polarization. In the process of coding, each subchannel is made to show a different reliability [23, 24]. When the length of information code to be transmitted continues to increase, some channels tend to the perfect channel with capacity close to 1 (error-free code), and the other channels tend to the pure noise channel with capacity close to 0. On this basis, we can select those channels whose capacity is close to 1 to transmit information directly to approximate the channel capacity. In addition, the polar code is the only coding scheme that can be strictly proved to achieve the Shannon limit.

The construction of polar code is composed of error detection, code matrix generation, sequence, rate matching [25], and interleaving. In the interleaving part, we can also divide it into two steps, interleaving before coding and interleaving after coding. Interleaving before coding is applicable to 5G-NR DCI (downlink control information), and there is no upstream interleaving; the interleaving after coding is applicable to 5G-NR UCI (uplink control information), and there is no downstream interleaving. This paper discussed interleaving of UCI.

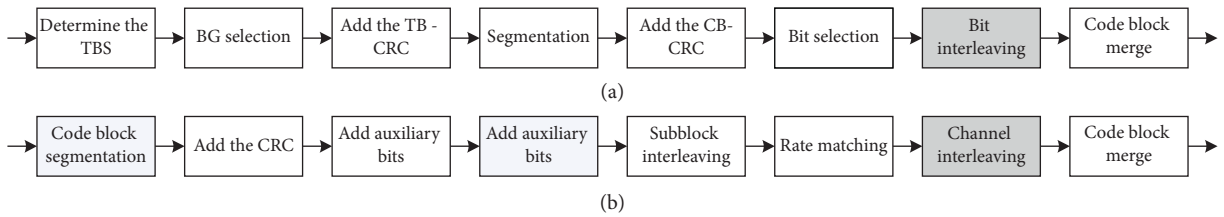

Figure 1: The position of our work in the 5G NR standard. (a) LDPC encoding process of PUSCH and PDSCH in 5G. (b) Polar encoding process for UCI in 5G.

In Introduction, we have briefly introduced that the interleaving is to disrupt the information structure without changing the information content and reduce the relevance between information bits to improve the resistance to burst interference. In the interleaving of UCI, the right triangle interleaving method is specified [26], as shown in Figure 4. In this method, we assume that the storage unit is an isosceles right triangle with a right angle side length of P, and the side length P is clearly defined in 3GPP, that is,

$$P \times \frac{P}{2} \leq 8192. \tag{1}$$

In 3GPP, the interleaver has a maximum of 8192 bits [27]. In this case, M is set as the number of bits after rate matching. At this time, it requires

$$P \times \frac{P+1}{2} \geq M. \tag{2}$$

When the equation takes the equal sign, we write the information into the interleaver line by line and then read it out in the order of columns. When the equation takes the greater than sign, there is still some unused space after all the information is loaded into the interleaver. At this time, we load dummy elements (nulls) into the interleaver and discard the dummy elements when reading out by columns. From the above process, we can see that this is similar to the interleaving process of block interleaver [28], but the rules of interleaving are not unitary because the number of rows in each column or the number of columns in each row is different. We can find that, after the right triangle interleaving, the spacing between each adjacent information data becomes P, P–1, and P–2, and they are not equidistant. With the right triangle interleaving theory, we use Matlab to simulate. We set up comparison groups; that is, one group contains isosceles right triangle interleaving method, while the other group does not. As shown in Figure 5, we can find that the performance for reducing the bit error rate is improved after using the interleaver. Among them, the red dotted line does not use the interleaving function, while the blue line uses the interleaving function.

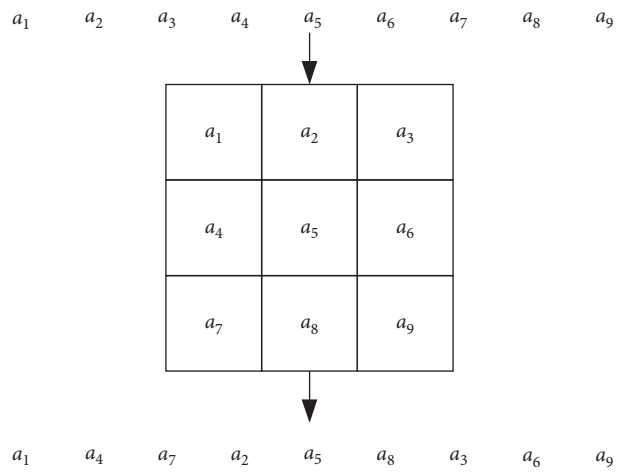

Figure 2: Interleaving process of row/column interleaver.

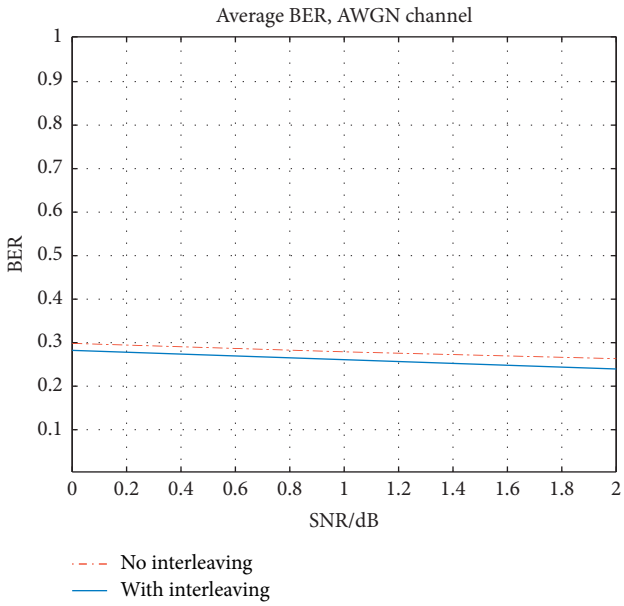

Figure 3: Comparison of BER before and after interleaving.

3. Multiplexing of Two Interleavers in 5G NR Standards

3.1. Formula Representation of Standardized Interleavers

3.1.1. Formula Representation of LDPC-Coded Data Channel Interleaver. The channel interleaving process based on LDPC encoding in 5G is given in Table 1.

In Table 1, E is the length of the input sequence, Q_m is the modulation order, e is the sequence before interleaving, and f is the sequence after interleaving.

From Table 1, the essence of the whole interleaving process is to write the input sequence $x(n)$ in rows and read the interleaving sequence as $f(n)$ in columns. Hence,

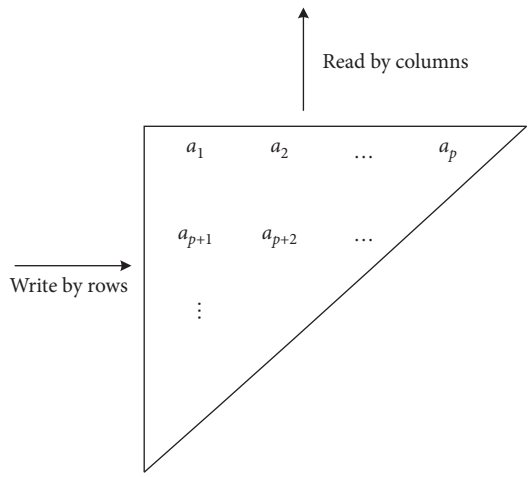

Figure 4: Right triangle interleaving process.

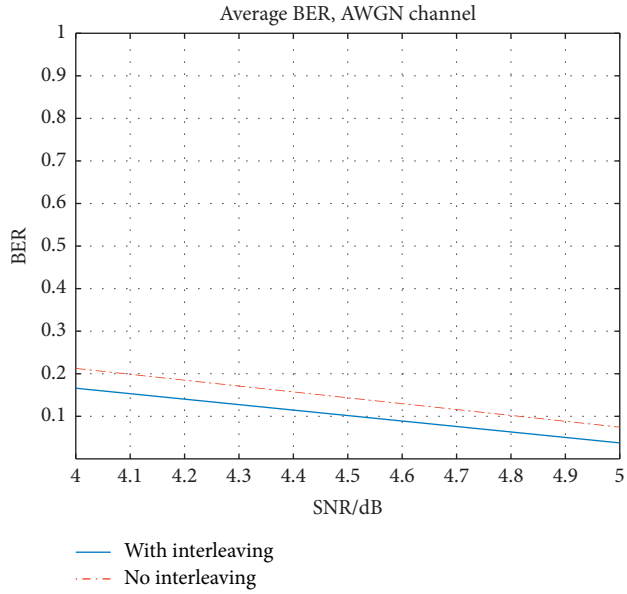

Figure 5: Comparison of BER before and after interleaving.

Table 1: The process of data channel interleaver.

```
for j = 0 to E/Q_m − 1
  for i = 0 to Q_m − 1
    f_{i+j·Q_m} = e_{i·(E/Q_m+j)}
  end for
end for
```

the realization of interleaver is to find the corresponding relationship between $f(n)$ and $x(n)$, that is, the interleaving address. Since the interleaving process can be equivalent to that in a matrix, the parameter i in the interleaving process can be equivalent to a row parameter, j can be equivalent to a column parameter, and the row and column correspond to the row and column in the matrix, respectively, where the range of i is $[0, Q_m - 1]$, and the range of j is $[0, E/Q_m - 1]$. Then, set the interleaving result as $J_{i,j}$, which is

$$f_{i+j\cdot Q_m} = e_{J_{i,j}}, \tag{3}$$

where j is the outer loop and its value increases from 0 to $E/Q_m - 1$ and i is the inner loop whose value increases from 0 to Q_m. Thus, we can get the value of $i + j \times Q_m$ is 0, 1, 2, 3, ..., $E - 1$. That is, with the increase in i and j, the value of $J_{i,j}$ is the position of the elements in the output sequence corresponding to the sequence before interleaving. For example, the calculated value of $J_{i,j}$ is written as [1–4] in order. If the input sequence is e, then the output sequence is [$e(4)$, $e(3)$, $e(2)$, $e(1)$]. In the original process,

$$J_{i,j} = \frac{i \cdot E}{Q_m + j}. \tag{4}$$

However, because the formula is not convenient for subsequent hardware reuse, this paper adopts a new method to achieve the result.

Let us first assume that the value of input sequence e is 0, 1, 2, 3, . . ., 19. In other words, the value of the element in the input sequence is equal to its position in the input sequence, i.e., $J_{i,j}$. It is convenient for the later observation. $Q_m = 4$ denotes 16QAM modulation, and the rectangle after the data in the modulation process is shown in Figure 6. When the value of j is 0, the data in the first column are readout. $J_{i,j}$ corresponds to the next data, and it is always 5 more than that of the previous data. For example, the first element in the first row corresponding $J_{i,j}$ is 0, the next element corresponding $J_{i,j}$ is 5, and the next element corresponds to 10. The law of the following columns is the same as that of the first column. Therefore, when the row parameter i is not equal to 0, the value of $J_{i,j}$ is the value of the last read-out data $J_{i,j}$ (can be set as $J_{i-1,j}$) and plus E/Q_m. Then, when i is equal to 0, it can be observed that $J_{i,j}$ is the value of the column parameter j, so the formula of $J_{i,j}$ can be derived as

$$J_{i,j} = \begin{cases} j, & i = 0, \\ J_{i-1,j} + \dfrac{E}{Q_m}, & i \neq 0, \end{cases} \tag{5}$$

where the value of E/Q_m (i.e., the number of columns of the rectangle) can be given in the precalculation stage.

3.1.2. Formula Representation of Polar-Coded Control Channel Interleaver. In the above part, we give a brief overview of the whole interleaving system. Here, we will refine the formula and implement the hardware diagram according to the interleaving process. First of all, we need to make it clear that the data entered the isosceles right triangle interleaver according to the order of rows but readout according to the order of columns. Therefore, we can think that the data are read in line order, and then, a transpose of rows and columns is carried out in the interleaver. Then, we read in line order, which is more convenient for us to derive the formula. Then, we introduce two variables, i and j, as row and column counter, respectively. Here, we make the following provisions for i and j

0	1	2	3	4
5	6	7	8	9
10	11	12	13	14
15	16	17	18	19

Figure 6: Input sequence after filling in rectangle.

($i \le P-1, j \le P-1$), where P is the size of the right-angle side of an isosceles right triangle. Meanwhile, when i increases from 0 to $P-1$, j adds 1; j increases from 0 to $P-1$, then i adds 1. When the information data enter the interleaver and are transposed, it is easy to find the order of elements $a_{j,i}$ of row j and column i in the sequence after reading them out by row. We can obtain the formula as

$$a_{j,i} = \begin{cases} i, & j = 0, \\ \left(C_1 + C_j\right) \times (j+1) \times \frac{1}{2} - \left[C_j - (i+1)\right], & j \neq 0. \end{cases} \tag{6}$$

It should be noted that the above formula does not consider the existence of dummy elements, so it only describes the case when the information sequence completely fills the interleaver. In equation (6), C_1 is the length of the first line of the right triangle interleaver, which is P. C_j is defined by

$$C_j = \begin{cases} P, & j = 0, \\ C_{j-1} - 1, & j \neq 0, \end{cases} \tag{7}$$

where C_j is the number of columns in the $(j+1)$th row. Through the above two formulas, we have known the relationship between $a_{j,i}$ and the sequence after interleaving after the internal transposition of interleaver. However, we still do not know the corresponding relationship between $a_{j,i}$ after transposition and the information a_k before entering the interleaver. Through the observation of the internal data of the interleaver after transposition, it is found that when we read the data in the order of columns, it is exactly the order in which the data are stored in the interleaver. Therefore, we can get a corresponding relationship as

$$a_k = \begin{cases} j, & i = 0, \\ \left(C_1 + C_i\right) \times (i+1) \times \frac{1}{2} - \left[C_i - (j+1)\right], & i \neq 0, \end{cases} \tag{8}$$

where C_i is the number of rows in each column. The definition is similar to the above C_j. In the deinterleaving, we should subtract the number of dummy elements, and equation (6) becomes (9) for such a purpose:

$$aj, i = \begin{cases} i, & j = 0, \\ \left(C_1 + Cj\right) \times (j+1) \times \frac{1}{2} - [Cj - (i+1)] - C, & j \neq 0. \end{cases} \tag{9}$$

After deducing the interleaved address without dummy elements, we now address a more realistic situation, namely,

$$P \times \frac{P+1}{2} > M. \tag{10}$$

In this case, the isosceles right triangle interleaver is filled with information elements and many dummy elements. When the dummy elements are taken into account, formula (6) no longer holds. However, we can still use the above numbers to calculate the interleaving address of a certain information unit including several dummy elements, and then, we can calculate the dummy element number C before this information unit and then make a subtraction, and we can obtain the interleaved address of this information unit.

Let us explain in detail how to calculate the number of dummy elements, which needs to be discussed in several cases. Before that, we first define several variables: i_{d} denotes the number of columns of the information unit to be calculated, j_{d} denotes the number of rows of the information unit to be calculated, i_{s} denotes the number of columns of the first dummy element, and $j_{\max}$ denotes the number of columns in the last cell of the first dummy element. We have the following three situations:

(1) $j_{\max} < j_d$. All dummies should be considered at this time. That is,

$$\left[1 + P - i_s\right] \times (j \max + 1) \times \frac{1}{2} - j_s. \tag{11}$$

(2) $j_{\max} = j_d$. First calculate the number of all dummy elements and then subtract 1 to get the total number of dummy elements to be subtracted. That is,

$$\left[1 + P - i_s\right] \times \left(j_{\max} + 1\right) \times \frac{1}{2} - j_s - 1. \tag{12}$$

(3) $j_{\max} > j_d$, under this condition, and it can be further divided into the following two situations. The first case: $i_d = i_s$; at this time, all the dummy elements

included in the (i_s+1)th column and the (j_d-1)th row to be requested are subtracted. That is,

$$\left[P-(i_s+1)+Cj_{d-1}-(i_s+1)\right]\times j_d\times\frac{1}{2}. \tag{13}$$

The second case: $i_d<i_s$, can be divided into the following three scenarios:

(a) $j_d>j_s$. We first calculate the total number of the i_s-th column and the j_d−1th row to be calculated and then subtract the number of information units in this range. That is,

$$\left[P-i_s+C_{j_d-1}-i_s\right]\times j_d\times\frac{1}{2}-j_s. \tag{14}$$

(b) $j_d<j_s$. At this time, the number of dummy elements in the i_s-st column and the j_d-th row are calculated. That is,

$$\left[P-(i_s+1)+C_{j_d-1}-(i_s+1)\right]\times j_d\times\frac{1}{2}. \tag{15}$$

(c) $j_d=j_s$. This case is the same as scenario (b).

After we analyzed all the required formulas, we start to design the hardware implement scheme. The first is the implementation of C_j, and here, we can use a loop with a judgment to achieve it. According to the formula, we can design a hardware structure with the subtraction gate as the main structure. On this basis, we add a judgment on the position of the output. When $j=0$, the output is 4. If this condition is not satisfied, we set a delay through the register and then make a subtraction with 1 in turn. On this basis, we implement the hardware structure step by step according to the formula. We add a judgment to the final output to meet the requirements of the formula. For the implementation of dummy computing hardware, the formulas can be divided into three categories. Among them, case 1 and case 2 belong to one category, and the hardware implementation of equation (11) can be reused; then, a logical judgment is added, and if the second case is satisfied, one is subtracted. The case one of (3) and (b) can be reused, while (a) cannot be reused because they do not have the same structure.

Note: when the number of dummy rows to be considered is only 1, if $j_s=0$ at this time, we only need to consider the number of $P-i_s$. When j_s is not equal to 0, we only need to consider the number of $P-i_s-1$. If the information unit to be calculated is on the first line, there is no need to subtract the number of dummy elements.

3.2. Verification of Formula for Interleaved Addresses

3.2.1. Verification of Interleaved Address Formula for Data Channel. We use Matlab to simulate the LDPC interleaving formula. First assume that the input sequence is [0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19] and the modulation order is 4. The corresponding figure of this sequence is shown in Figure 7.

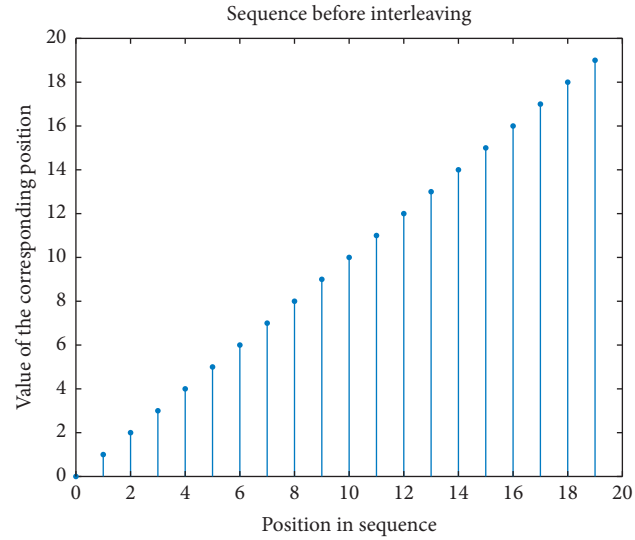

Figure 7: Sequence before interleaving.

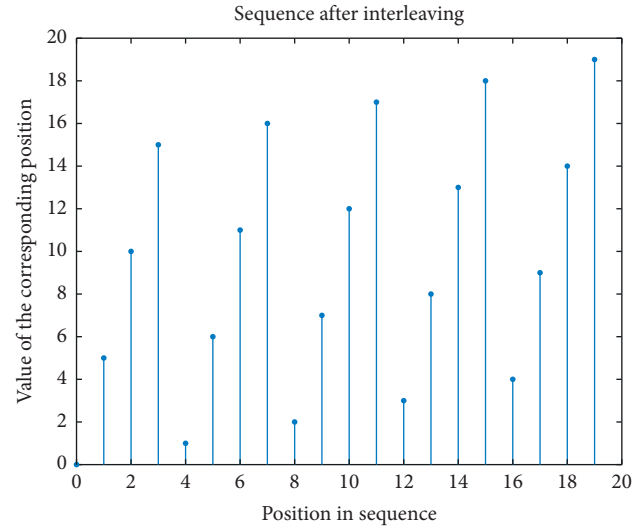

Figure 8: Interleaved sequence.

Using equation (5), $J_{i,j}$ is obtained. The corresponding position element is taken out from the input sequence according to the value of $J_{i,j}$, which is the value in the output sequence, as shown in Figure 8.

After verification, the interleaved sequence can be deinterleaved back to the original sequence. The formula is the same as the interleaving formula. We only need to exchange the ranges of i and j with each other and change the E/Q_m in the formula to Q_m. That is,

$$J_{i,j}=\begin{cases} j, & i=0,\\ J_{i-1,j}+Q_m, & i\neq 0. \end{cases} \tag{16}$$

Then, the interleaved sequence is deinterleaved in the same way as the interleaving process, and the deinterleaved sequence (i.e., the original input sequence) is obtained. The result of the deinterleaved sequence is shown in Figure 9.

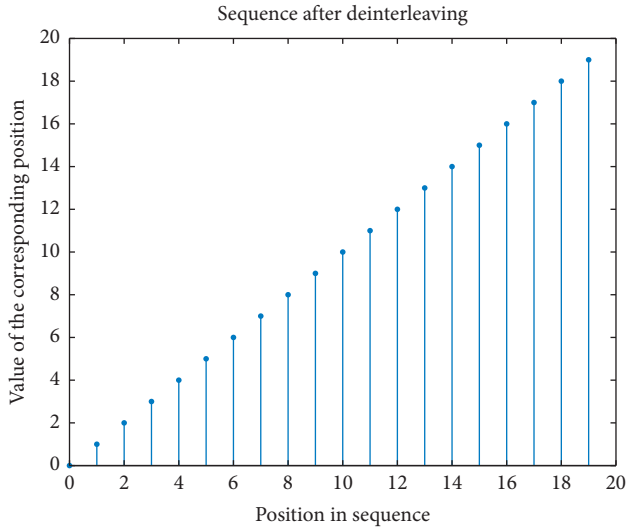

Figure 9: Deinterleaved sequence.

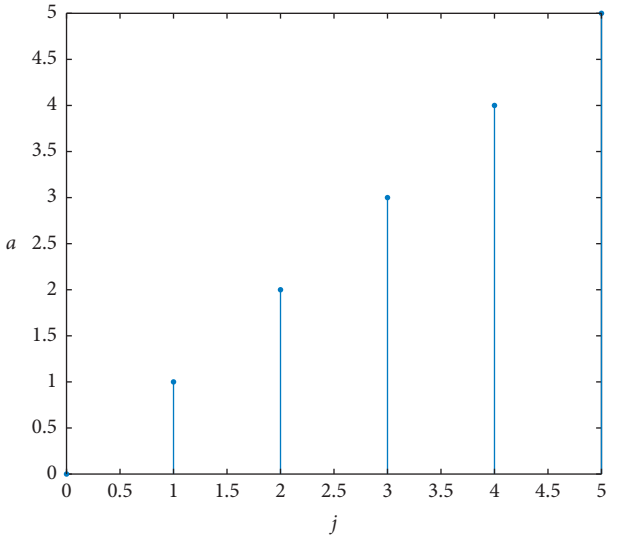

Figure 10: Verification 1 for formula (8).

From Figure 9, the deinterleaving method successfully restores the interleaved sequence to the original sequence. Hence, the interleaving formula and deinterleaving formula work correctly.

3.2.2. Verification of Interleaved Address Formula for Control Channel. In order to verify the formula, we define an isosceles right-angled triangle with $P = 6$ and load the data a_1 to a_{21} into the triangle interleaver in the order of the rows. The interleaving process is equivalent to the data in this triangle matrix. After transposing, we take them out in rows, so we make $a_{i,\ j}$ into $a_{j,\ i}$. First, we verify the correctness of formula (8). For the number of the first column in the interleaver, we can get it as shown in Figure 10.

For the second column number in the interleaver, the corresponding C_1 is 5 at this time, and then, the function formula we determined becomes

$$a_k = (6 + 5) \times 2 \times \frac{1}{2} - [5 - (j + 1)]. \tag{17}$$

Among them, the corresponding values of j are 0, 1, 2, 3, and 4, which can be obtained through Matlab. The result is shown in Figure 11.

For the third column number in the interleaver, the corresponding C_2 is 4 at this time, and then, the function formula becomes

$$a_k = (6 + 4) \times 3 \times \frac{1}{2} - [4 - (j + 1)], \tag{18}$$

where the corresponding values of j are 0, 1, 2, and 3. The result is shown in Figure 12.

After comparison, we find out that this corresponds to the actual serial number of the information after it is loaded into the interleaver and after transpose. Hence, the formula is theoretically feasible.

Next, we verify the formula of the interleaved address information given in equation (6). For $j = 0$, the first row of elements according to (6) is obtained and shown in Figure 13.

For $j = 1$, the second row of elements is obtained. According to equation (6), we get

$$a_{j,i} = (6 + 5) \times 2 \times \frac{1}{2} - [5 - (i + 1)]. \tag{19}$$

The simulation result is shown in Figure 14.

For $j = 2$, the third row of elements is obtained. According to equation (6), we get

$$a_{j,i} = (6 + 4) \times 3 \times \frac{1}{2} - [4 - (i + 1)]. \tag{20}$$

And the simulation result is shown in Figure 15.

After comparison, we find that this is consistent with the sequence corresponding to the actual information loaded into the interleaver and readout row by row after a transpose. From these, we conclude that the formula is theoretically feasible. In the following, we verify the corresponding C value for different situations, that is, the number of dummy elements to be subtracted.

For the first case, that is, $j_{\max} < j_d$ to be calculated, the number of dummy elements must be considered in calculating the address of the information at this time. This case corresponds to equation (11). Suppose we now find a_6, and we let the head of i_s in equation (11) be 1 and 2 to verify the correctness of the expression.

When $i_s = 1$, the corresponding $j_{\max} = 4$, and the value of j_s is in the range [0, 1, 2, 3, 4], that is, from a_6, a_7, a_8 to a_{11}, respectively, as the first dummy element. The parameters i_s and $j_{\max}$ are substituted into equation (11), and the result can be obtained and shown in Figure 16.

When $i_s = 2$, corresponding to $j_{\max} = 3$, the range of j_s is [0, 1, 2, and 3], that is, from a_{12}, a_{13}, a_{14} to a_{15}, respectively, as the first dummy element. The parameters i_s and $j_{\max}$ are substituted into equation (11), and the result can be obtained and shown in Figure 17.

After comparison, we verified the correctness of the formula. For the above second case, $j_{\max} = j_d$ is to be

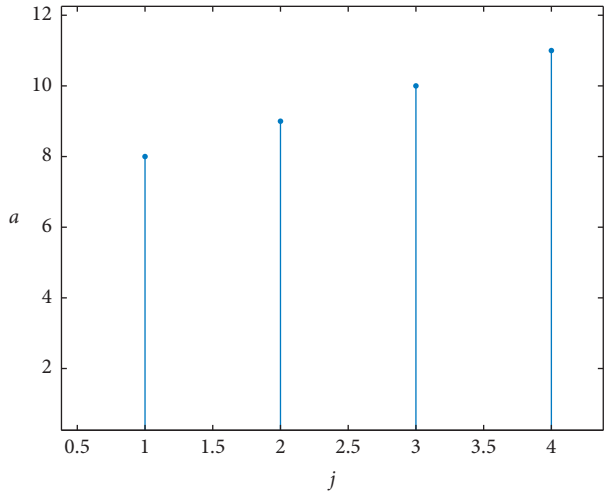

Figure 11: Verification 2 for formula (8).

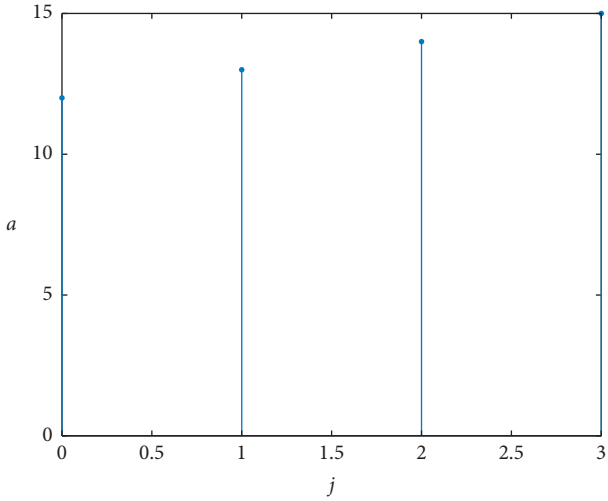

Figure 12: Verification 3 for formula (8).

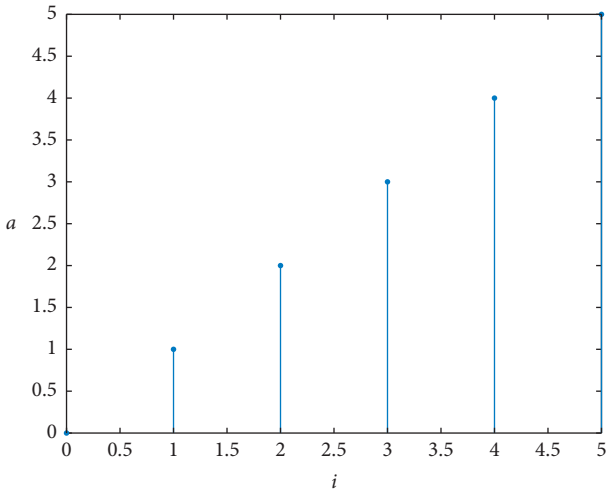

Figure 13: Verification 1 for formula (6).

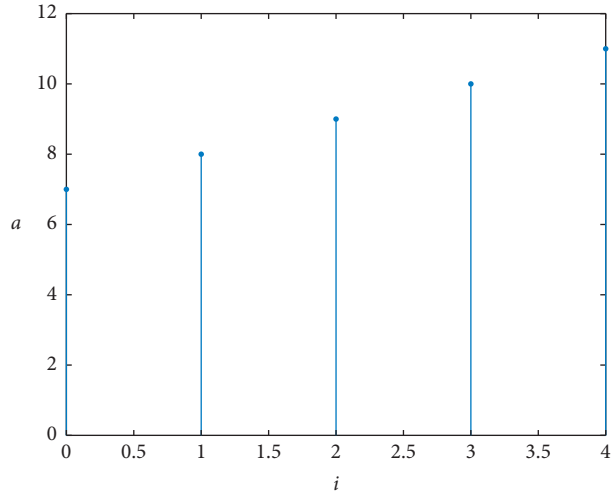

Figure 14: Verification 2 for formula (6).

calculated; that is, the information unit to be obtained is on the same line as the dummy with the largest number of rows. At this time, the number we need to consider is the total number of dummies minus 1 (only considering the number of dummy elements before this information unit). In view of this situation, we can subtract 1 if equation (11) is correct.

For the case 3, that is, $j_{\max} > j_d$ to be decided, first, we discuss $i_d = i_s$; that is, the information unit to be decided and the first dummy element are in the same column. At this time, we need to consider the number of all units contained in the $(i+1)th$ column and the $(j-1)th$ row based on the information unit. This case corresponds to equation (13). In equation (13), we choose $i_s = 1$ and $i_s = 2$ for verification. When the i_s is 1, j_d can be 2 and 3. When i_s is 2, the j_d can be 2 and 3 as well.

When $j_d = 2$, according to equation (13), for different i_s, the simulation result can be obtained in Figure 18.

When $j_d = 3$, according to equation (13), for different i_s, the simulation result is shown in Figure 19.

After verification, it is the same as the theoretical value. When $i_d < i_s$, first discuss $j_d > j_s$; that is, the number of rows of the information unit to be decided is greater than the first dummy element. At this time, the formula for calculating C differs from the above formula only in that we are considering the i_s column starts. Meanwhile, we can subtract the number of information units in the i_s row. When the first dummy element starts from 8 and 12, respectively, we use (14) to solve the address after a_3 interleaving. Because the corresponding j_s and t_s are different, the verification results are more general.

For the case where the first dummy starts at 8, the formula is

$$C = (6 - 1 + 5 - 1) \times 2 \times \frac{1}{2} - 1. \tag{21}$$

The calculated result is 8, which is the correct result.

For the case where the first dummy starts at 12, the formula is

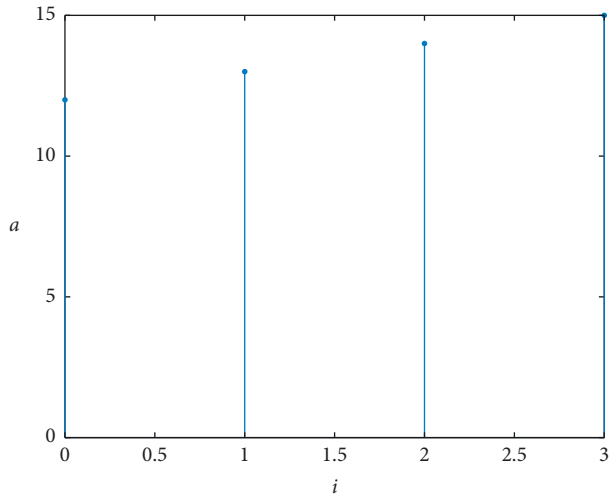

Figure 15: Verification 3 for formula (6).

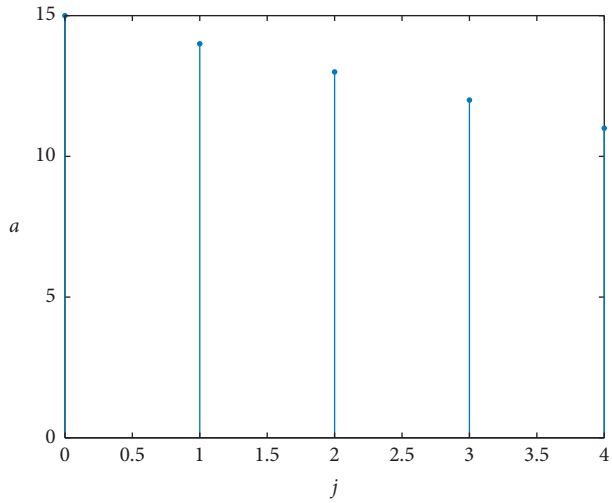

Figure 16: Verification 1 for formula (11).

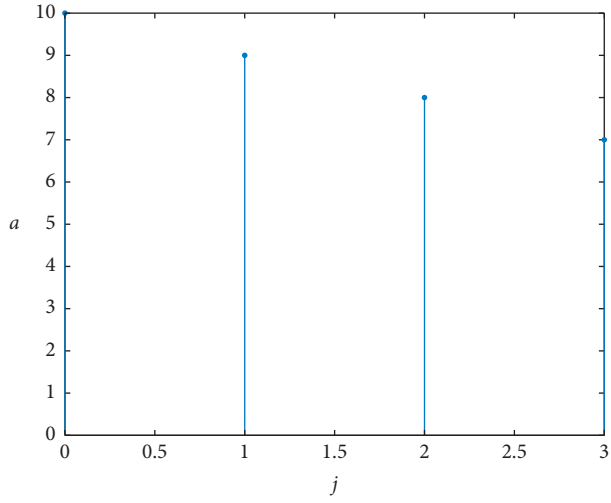

Figure 17: Verification 2 for formula (11).

$$C = (6 - 2 + 5 - 2) \times 2 \times \frac{1}{2} - 0. \tag{22}$$

The calculated result is 7, which is the correct result.

When $j_d \le j_s$, the number of dummy elements we need to calculate at this time is all the numbers in $i_s + 1$, $j_d - 1$. The calculation formula is equation (15).

3.3. Hardware Design and Reuse of Two Coding Interleavers

3.3.1. Interleaver Hardware Design for LDPC-Coded Data Channel. From the formula of $J_{i,j}$, it can be concluded that the hardware required for its implementation is an adder, a selector, and an address register, which can realize the interleaver of the data channel. It is shown in Figure 20.

3.3.2. Interleaver Hardware Design for Polar-Coded Control Channel. The interleaver hardware design for polar-coded control channel is shown in Figure 21. In the first part of the figure, we can get C_i and then pass through a few adders and subtractors. Before entering the second part, the output of the subtraction gate is

$$(C_i + C_1) \times (i + 1) \times \frac{1}{2} - [C_i - (i + 1)]. \tag{23}$$

When judged by a logic gate, if $j = 0$ (that is, the one in the first row after replacement), the output is i. If $j = 0$ is not satisfied, the output is equation (23).

3.3.3. Hardware Multiplexing of Two Interleavers. By observing and comparing the hardware implementation diagrams of two interleavers, we can find that the hardware structure of LDPC-coded data channel interleaver has also appeared in the polar-coded control channel interleaver. Thus, the hardware structure of the data channel interleaver can be set to a new module M, and its structure diagram is shown in Figure 22. It has a total of three input terminals (a, b, c) and one output terminal y. The input and output parameters can be determined according to the selection of the interleaving scheme. If it is selected for the data channel interleaving, the input parameters a, b, and c are E/Q_m, j, and i; the output is the interleaved address $J_{i,j}$. If it is selected for the control channel interleaving, the input parameters are 1, 4, and j, respectively. The output is C_j. Therefore, the final design of the multiplexing structure can be obtained as shown in Figure 23.

3.3.4. Flow Charts of Precalculation Stage and Execution Stage. The flow charts of precalculation stages and execution stages are shown in Figures 24 and 25, respectively.

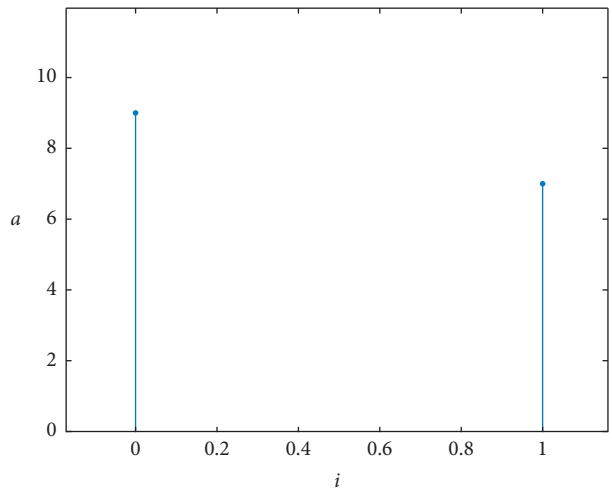

Figure 18: Verification 1 for formula (13).

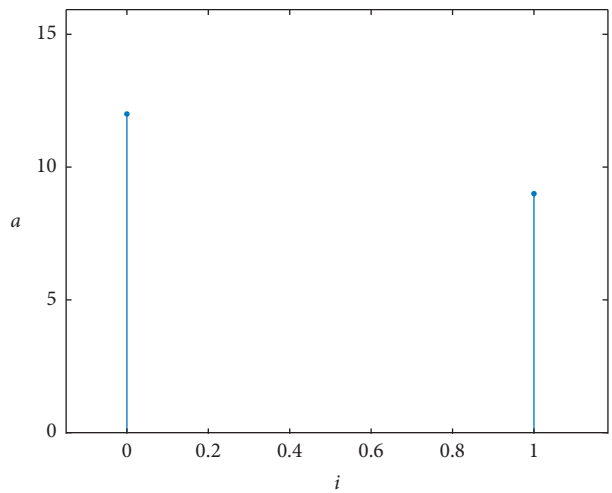

Figure 19: Verification 2 for formula (13).

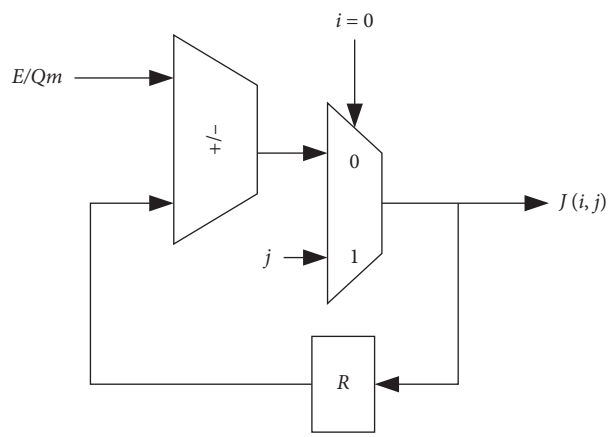

Figure 20: Hardware diagram of data channel interleaver.

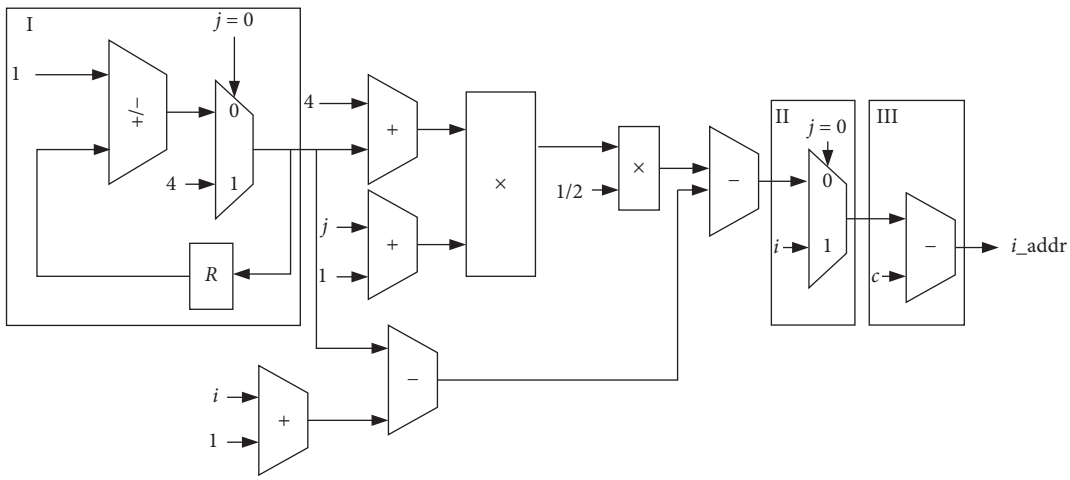

Figure 21: Hardware diagram of control channel interleaver.

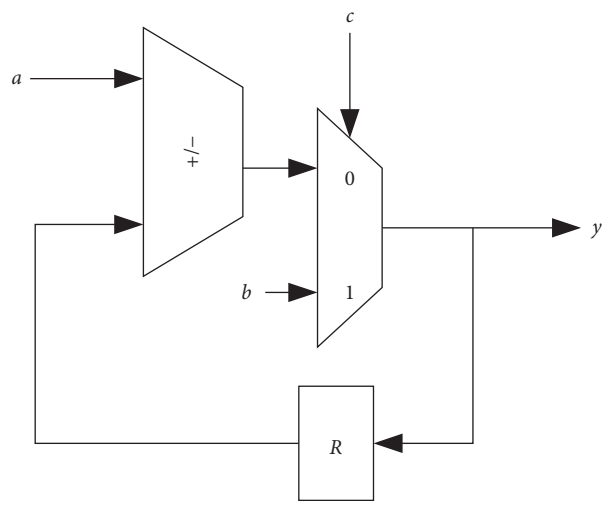

Figure 22: Multiplexing module *M*.

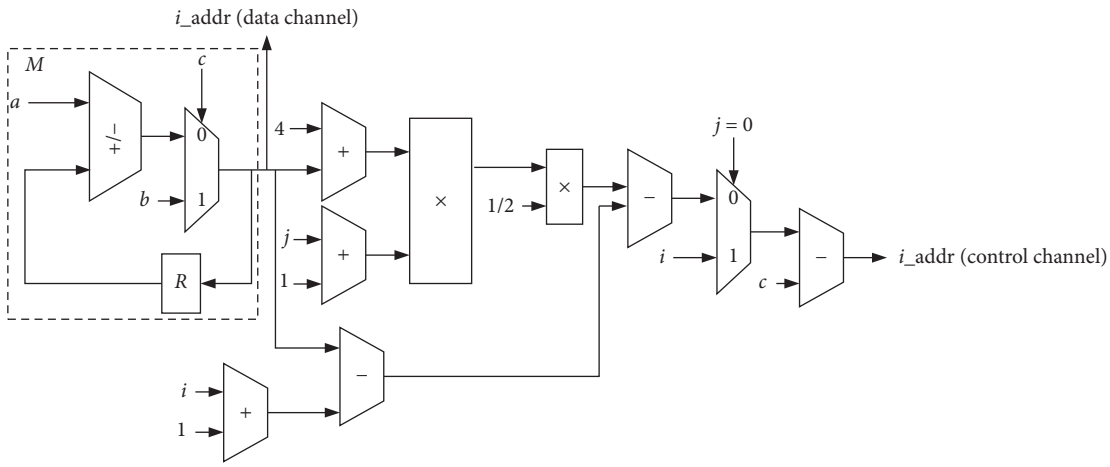

Figure 23: Hardware sharing structure after reuse.

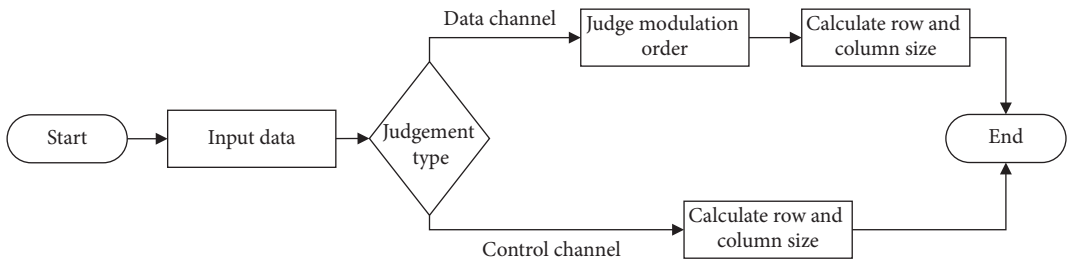

Figure 24: Flow chart of precalculation stage.

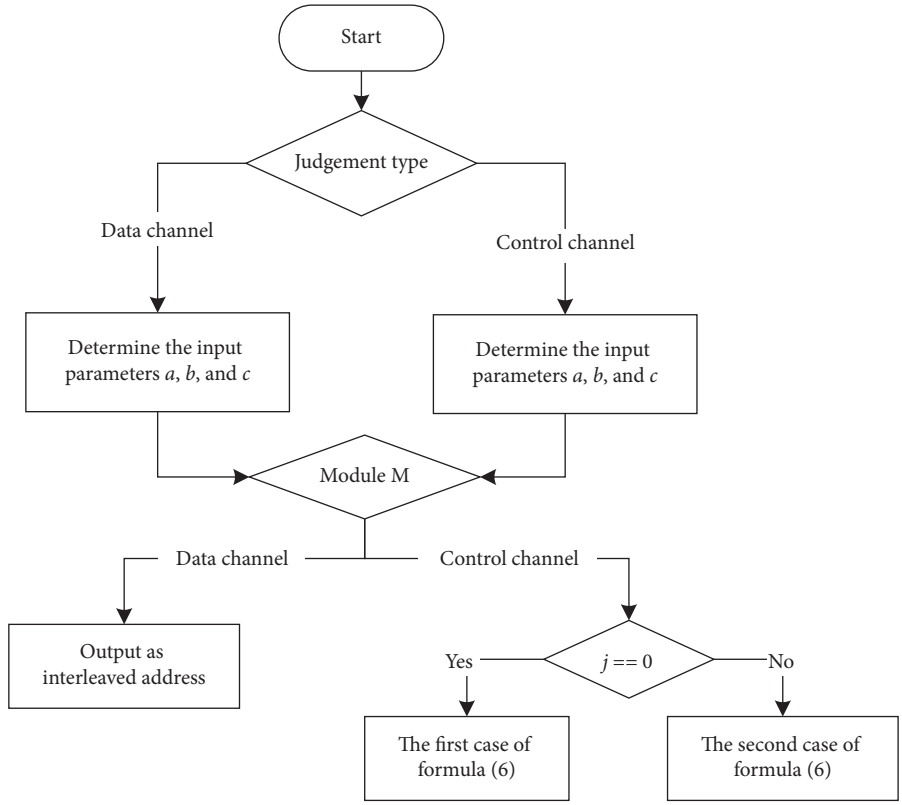

Figure 25: Flow chart of execution stage.

4. Conclusions and Future Work

This paper presents an interleaver multiplexing scheme for the LDPC and polar encoding channel which are specified in 5G NR standards. First, we analyze the two interleaving methods and then refine and improve the formulas according to the interleaving process to achieve the hardware reuse. Then, according to the formulas, the hardware realization of interleaving address is derived. Finally, the hardware implementation of the two-channel interleavers is reused as much as possible to achieve the purpose of reducing the hardware cost. However, there are still some issues to be improved in our research work. For example, the formula for generating interleaving address extracted is complicated; especially the formulas for refining interleaving process of control channel need to be further simplified. In the future work, we will also consider the parallelization processing under a variety of channel encoding standards in combination with rate matching.

Data Availability

The code and data of ".m" and ".mat" format files used to support the findings of this study have been deposited in the GitHub repository (https://github.com/huzhuhua/Data-and-Code-for-Security-and-Communication-Networks).

Conflicts of Interest

The authors declare that there are no conflicts of interest regarding the publication of this paper.

Acknowledgments

This work was supported by the National Natural Science Foundation of China (nos. 61963012, 61961014, and 61661018), Natural Science Foundation of Hainan Province, China (no. 619QN195), and Key R&D Project of Hainan Province, China (no. ZDYF2018015).

References

[1] R. Asghar, *Flexible Interleaving Sub–systems for FEC in Baseband Processors*, Linköping University Electronic Press, Linköping, Sweden, 2010.

[2] H. Wu and K. Wolter, "Stochastic analysis of delayed mobile offloading in heterogeneous networks," *IEEE Transactions on Mobile Computing*, vol. 17, no. 2, pp. 461–474, 2017.

[3] Z. Hu, Y. Bai, Y. Zhao, and M. Xie, "Adaptive and blind wideband spectrum sensing scheme using singular value decomposition," *Wireless Communications and Mobile Computing*, vol. 2017, Article ID 3279452, 14 pages, 2017.

[4] Z. Hu, Y. Bai, M. Huang, M. Xie, and Y. Zhao, "A self-adaptive progressive support selection scheme for collaborative wideband spectrum sensing," *Sensors*, vol. 18, no. 9, p. 3011, 2018.

[5] H. Wu, W. J. Knottenbelt, and K. Wolter, "An efficient application partitioning algorithm in mobile environments," *IEEE Transactions on Parallel and Distributed Systems*, vol. 30, no. 7, pp. 1464–1480, 2019.

[6] X. Liu, M. Jia, X. Zhang, and W. Lu, "A novel multichannel Internet of things based on dynamic spectrum sharing in 5G communication," *IEEE Internet of Things Journal*, vol. 6, no. 4, pp. 5962–5970, 2019.

[7] X. Liu and X. Zhang, "NOMA-based resource allocation for cluster-based cognitive industrial Internet of Things," *IEEE Transactions on Industrial Informatics*, vol. 16, no. 8, pp. 5379–5388, 2020.

[8] R. Asghar and D. Liu, "Multimode flex-interleaver core for baseband processor platform," *Journal of Computer Networks and Communications*, vol. 2010, p. 2010.

[9] Y. Sun, Y. Zhu, M. Goel, and J. R. Cavallaro, "Configurable and scalable high throughput turbo decoder architecture for multiple 4G wireless standards," in *Proceeding of the IEEE International Conference on Application-specific Systems, Architectures and Processors*, pp. 209–214, IEEE, Leuven, Belgium, July 2008.

[10] R. Asghar, D. Wu, J. Eilert, and D. Liu, "Memory conflict analysis and implementation of a re-configurable interleaver architecture supporting unified parallel turbo decoding," *Journal of Signal Processing Systems*, vol. 60, no. 1, pp. 15–29, 2010.

[11] Z. Zhang, B. Wu, Y. Zhou, and X. Zhang, "Low-complexity hardware interleaver/deinterleaver for IEEE 802.11a/g/n WLAN," *VLSI Design*, vol. 2012, Article ID 948957, 7 pages, 2012.

[12] P. Benoit, L. Torres, G. Sassatelli et al., "Dynamic hardware multiplexing: improving adaptability with a run time reconfiguration manager," in *Proceeding of the IEEE Computer Society Annual Symposium on Emerging VLSI Technologies and Architectures*, March 2006.

[13] R. Asghar and D. Liu, "Low complexity hardware interleaver for MIMO-OFDM based wireless LAN," in *Proceeding of the IEEE International Symposium on Circuits and Systems*, pp. 1747–1750, IEEE, Taipei, China, May 2009.

[14] 3GPP TS 38.212. NR, "Multiplexing and Channel Coding," 2017.

[15] E. Arikan, "Channel polarization: a method for constructing capacity-achieving codes for symmetric binary-input memoryless channels," *IEEE Transactions on Information Theory*, vol. 55, no. 7, pp. 3051–3073, 2009.

[16] W. E. Ryan, "An introduction to LDPC codes," *CRC Handbook for Coding and Signal Processing for Recording Systems*, pp. 1–23, CRC Press, Boca Raton, FL, USA, 2004.

[17] R1-1713474. "Design and Evaluation of Interleaver for Polar Codes," Qualcomm Inc., 3GPP TSG RANWG1 #90 Meeting, Prague, Czechia, 2017.

[18] H. Gamage, N. Rajatheva, and M. Latva-Aho, "Channel coding for enhanced mobile broadband communication in 5G systems," in *Proceeding of the European Conference on Networks and Communications (EuCNC)*, IEEE, Oulu, Finland, 2017.

[19] D. J. C. MacKay and R. M. Neal, "Near Shannon limit performance of low density parity check codes," *Electronics Letters*, vol. 33, no. 6, 1997.

[20] M. G. Luby, M. Mitzenmacher, M. A. Shokrollahi, and D. A. Spielman, "Efficient erasure correcting codes," *IEEE Transactions on Information Theory*, vol. 47, no. 2, pp. 569–584, 2001.

[21] J. Xu and J. Xu, "Structured LDPC applied in IMT-advanced system," in *Proceeding of the 4th IEEE International Conference on Wireless Communication*, IEEE, Dalian, China, 2008.

[22] R. Asghar and D. Liu, "Multimode flex-interleaver core for baseband processor platform," *Journal of Computer Networks and Communications*, vol. 2010, Article ID 793807, 16 pages, 2010.

[23] R. Mori and T. Tanaka, "Performance of polar codes with the construction using density evolution," *IEEE Communications Letters*, vol. 13, no. 7, pp. 519–521, 2009.

[24] 3GPP R1-167209, "Polar Code Design and Rate Matching," Huawei and HiSilicon, 3GPP TSG RAN WG1 #86 Meeting, Gothenburg, Sweden, 2016.

[25] 3GPP, R1-1713705, "Polar rate-matching design and performance," MediaTek, WG1#90, 2017.

[26] 3GPP, R1-1708649, "Interleaver design for polar codes," qualcomm, RAN#89, 2017.

[27] 3GPP, "Draft_Minutes_report_RAN#91_v020," 2017, http://www.3gpp.org/ftp/tsg_ran/WG1_RL1/a) TSGR1_91/Report/Draft_Minutes_report_RAN1%2391_v020.zip.

[28] E. Tell and D. Liu, "A hardware architecture for a multi-mode block interleaver," in *Proceeding of the International Conference on Circuits and Systems for Communications (ICCSC)*, Moscow, Russia, 2004.