

## Article

# An Efficient Method to Construct Parity-Check Matrices for Recursively Encoding Spatially Coupled LDPC Codes <sup>†</sup>

Zhongwei Si <sup>\*</sup>, Sijie Wang and Junyang Ma

Key Laboratory of Universal Wireless Communications, Ministry of Education, Beijing University of Posts and Telecommunications (BUPT), No. 10 Xitucheng Road, Haidian District, Beijing 100876, China; wangsijie@bupt.edu.cn (S.W.); majunyang@bupt.edu.cn (J.M.)

<sup>\*</sup> Correspondence: sizhongwei@bupt.edu.cn; Tel.: +86-10-6228-4023

<sup>†</sup> This paper is an extended version of our paper published in the 2015 IEEE International Symposium on Personal, Indoor and Mobile Radio Communications, Hong Kong, China, 30 August–2 September 2015.

Academic Editor: Raúl Alcaraz Martínez

Received: 15 June 2016; Accepted: 15 August 2016; Published: 17 August 2016

**Abstract:** Spatially coupled low-density parity-check (LDPC) codes have attracted considerable attention due to their promising performance. Recursive encoding of the codes with low delay and low complexity has been proposed in the literature but with constraints or restrictions. In this manuscript we propose an efficient method to construct parity-check matrices for recursively encoding spatially coupled LDPC codes with arbitrarily chosen node degrees. A general principle is proposed, which provides feasible and practical guidance for the construction of parity-check matrices. According to the specific structure of the matrix, each parity bit at a coupling position is jointly determined by the information bits at the current position and the encoded bits at former positions. Performance analysis in terms of design rate and density evolution has been presented. It can be observed that, in addition to the feature of recursive encoding, selected code structures constructed by the newly proposed method may lead to better belief-propagation thresholds than the conventional structures. Finite-length simulation results are provided as well, which verify the theoretical analysis.

**Keywords:** channel coding; spatially coupled LDPC codes; protograph codes; recursive encoding; density evolution

## 1. Introduction

Low-density parity-check (LDPC) codes [1] were the most researched channel coding in the last decade due to the good performance and the reasonable complexity. Their convolutional counterparts, the LDPC convolutional codes, were invented in [2] and further developed in, e.g., [3,4]. Different constructions and analytical results have been provided in [3–11]. For example, Lentmaier et al. [3] proposed a realization of LDPC convolutional codes and conjectured the capacity-achieving performance under belief-propagation (BP) decoding. The LDPC convolutional codes were considered in a realm of spatially coupled codes [4], and the capacity-achieving property of the code family has been proved theoretically [10–13]. The good features have triggered the application of spatially coupling in many scenarios, such as compressed sensing [14], rate-compatible coding [15,16], relay channels [17,18], wiretap channels [19], multiple access channels [20], multiuser detection [21], source coding [22], etc.

To realize the promising performance in practice, the implementation of spatially coupled LDPC codes is worth investigation. The best known implementation of spatially coupled LDPC codes is the one introduced in [6]. Based on the particular structure of the parity-check matrix, a recursive

encoder using a shift-register was proposed. The encoder requires only a small number of memory units and the computational cost per bit is proportional to the check node degree. This is a clear advantage compared with the encoding of LDPC block code regarding storage requirements and computational costs. Combined with the pipeline decoding or the sliding-window decoding [6,23], spatially coupled LDPC codes provide a low-delay and low-complexity realization of channel coding with capacity-approaching performance.

The structure in [6] is suitable for recursive encoding; however, several constraints need to be satisfied when constructing the parity-check matrix. Alternatively, Sridharan et al. [8] proposed a simpler way to generate the parity-check matrix using randomly permuted identity matrices on condition that the check node degree  $d_c$  and the variable node degree  $d_v$  are not relatively prime. If we further require that the ratio of  $d_c$  and  $d_v$  is an integer, then the structure in [8] enables recursive encoding. These constraints and requirements limit the practical application of the codes. To extend the construction to the codes with arbitrary rates, we propose in this paper an efficient method to construct the parity-check matrices with permuted identity matrices regardless of the choice of  $d_c$  and  $d_v$ . By rearranging the edge connections, each parity bit at a coupling position is jointly determined by the information bits at the current position and the encoded bits at former positions [24]. A generalized principle is proposed, which provides feasible and practical guidance for constructing the recursive structures. The requirement of memory units and the computational complexity retain similar to the recursive encoder in [6]. Performance analysis in terms of design rate and density evolution is provided. It can be observed that, with properly designed pattern, the modified code structure may converge faster in the belief-propagation (BP) decoding than the existing structure [8]. Consequently, the corresponding spatially coupled LDPC ensemble is able to provide better BP thresholds and smaller gaps to the Shannon limits. Bit erasure rate (BER) performance with finite code lengths is provided as well, which verifies the analysis of the BP thresholds.

In this manuscript we restrict ourselves only to the convolutional-like LDPC codes with nearly regular structures. More advanced spatially coupled protograph codes [25] have been proposed with better decoding thresholds, minimum distances and finite-length trade-offs [26], such as spatially coupled repeat-accumulate (RA) codes [27], spatially coupled accumulate-jagged-accumulate (ARJA) codes [28] and spatially coupled MacKay–Neal (MN) codes [29]. The easy encoding of these codes is definitely worth investigations, and we will look into it in the future work.

The remainder of the paper is organized as follows. An overview of spatially coupled LDPC codes and the recursive encoding proposed in [6,8] are given in Section 2. In Section 3, we propose a principle on constructing the parity-check matrices of spatially coupled LDPC codes and illustrate the recursive encoding using a shift-register. Performance analysis based on design rate and density evolution is provided in Section 4. Section 5 presents the finite-length performance evaluation in the binary erasure channel (BEC). Section 6 concludes the paper.

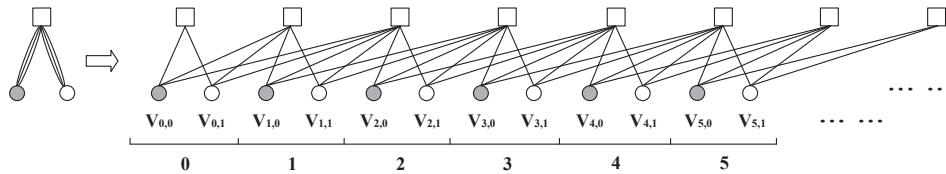
## 2. Overview of Spatially Coupled LDPC Codes

Spatially coupled LDPC codes are constructed by coupling together a series of disjoint LDPC block codes into a single coupled chain. As shown in Figure 1, the left unit is the protograph of a LDPC code, and the structure on the right represents the protograph of the corresponding spatially coupled LDPC code. The spatially coupled LDPC ensemble is obtained by duplication and edge rearrangement from the LDPC ensemble.

A  $(d_v, d_c)$  spatially coupled LDPC code with coupling chain  $L$  can also be defined using the parity-check matrix  $\mathbf{H}_{[0,L-1]}^T$  as follows:

$$\mathbf{H}_{[0,L-1]}^T = \begin{bmatrix} \mathbf{M}_0^T(0) & \dots & \mathbf{M}_0^T(m_s) & & \\ & \ddots & \mathbf{M}_l^T(l+m) & \ddots & \\ & & \mathbf{M}_{L-1}^T(L-1) & \dots & \mathbf{M}_{L-1}^T(L+m_s-1) \end{bmatrix},$$

where each submatrix  $\mathbf{M}_l^T(l+m)$  is a  $c \times (c-b)$  binary matrix. The parameter  $c$  corresponds to the number of variable nodes at each position, and  $(c-b)$  is the corresponding number of check nodes. In the notation,  $l$ ,  $0 \leq l < L$ , denotes the position in the chain, and  $m$ ,  $0 \leq m \leq m_s$ , indicates the position of edge connections. The parameter  $m_s$  is called the syndrome former memory, and  $(m_s + 1)c$  is regarded as the constraint length of the code.



**Figure 1.** Protograph of a  $(3, 6, L)$  spatially coupled low-density parity-check (LDPC) ensemble. The rectangles represent check nodes while the circles denote variable nodes. The gray circles correspond to information bits while the white circles associate with parity bits.

Once the node degrees  $(d_v, d_c)$  have been given, the parity-check matrix  $\mathbf{H}_{[0,L-1]}^T$  needs to satisfy the following constraints:

1. The number of ones in each row is  $d_v$ .
2. The number of ones in each column is  $d_c$ , except the beginning and the terminating ends of the chain.
3.  $\mathbf{M}_l^T(l+m) = 0$ , for  $m < 0$  and  $m > m_s, \forall l$ .
4. There exists  $l$  such that  $\mathbf{M}_l^T(l+m_s) \neq 0$ .
5.  $\mathbf{M}_l^T(l) \neq 0$  has full rank  $\forall l$ .

To implement the systematic and recursive encoding according to [6],  $\mathbf{M}_l^T(l+m)$  must additionally satisfy:

6. The last  $c-b$  rows of  $\mathbf{M}_l^T(l)$  is a  $(c-b) \times (c-b)$  identity matrix.

Note that, in this paper we assume that the encoder is systematic, i.e., the information bits is part of the codeword. We define the recursive encoding as that each parity bit to be encoded can be obtained through the module-2 addition of the known information bits and parity bits. Once a parity bit is determined, it will be utilized in the forthcoming calculation of other unknown parity bits.

It usually takes some efforts to simultaneously satisfy all the above constraints. Instead, if using the structure illustrated in [8], a syndrome matrix meeting all the requirements can be constructed simply.

We now describe the structure proposed in [8]. According to [8], the submatrix  $\mathbf{M}_l^T(l+m)$  is constructed as

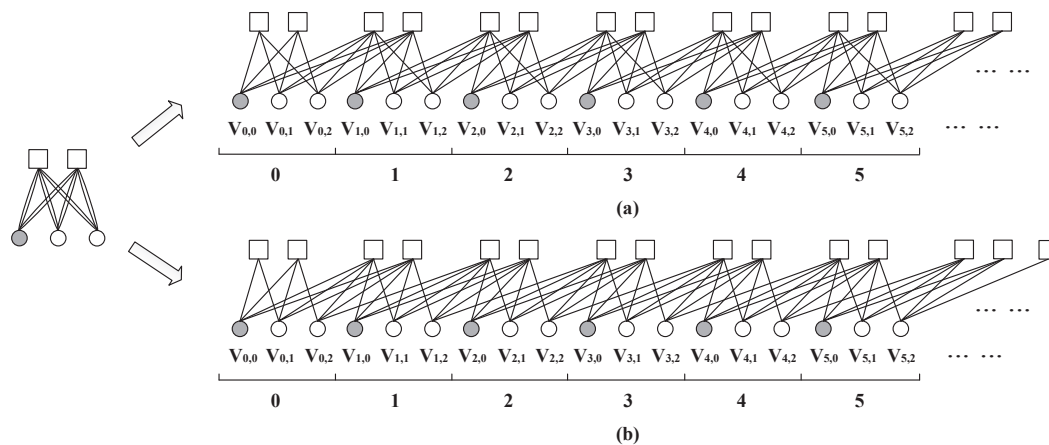
$$\mathbf{M}_l^T(l+m) = \begin{bmatrix} \mathbf{P}_{0,0}(l+m) & \cdots & \mathbf{P}_{0,d'_v-1}(l+m) \\ \vdots & \mathbf{P}_{i,j}(l+m) & \vdots \\ \mathbf{P}_{d'_c-1,0}(l+m) & \cdots & \mathbf{P}_{d'_c-1,d'_v-1}(l+m) \end{bmatrix},$$

where  $\mathbf{P}_{i,j}(l+m)$  is an  $M \times M$  random permutation matrix in the  $i$ -th row and the  $j$ -th column of  $\mathbf{M}_l^T(l+m)$ . Let  $m_b = \gcd(d_v, d_c)$  denote the greatest common divisor of  $d_v$  and  $d_c$ , then there exist positive integers  $d'_v = d_v/m_b$ ,  $d'_c = d_c/m_b$ , and  $\gcd(d'_v, d'_c) = 1$ . In this case, we have  $c = d'_c M$ ,  $b = (d'_c - d'_v)M$ , and the syndrome former memory  $m_s = m_b - 1$ . The satisfaction of constraints 1–5 is obvious. When  $d_c/d_v$  is an integer,  $\mathbf{M}_l^T(l+m)$  turns out to be a  $d'_c \times 1$  matrix. The constraint 6 is easy to access by setting  $\mathbf{P}_{d'_c-1,0}(l+m)$  as an identity matrix.

The recursive encoding of spatially coupled LDPC code for the case that  $d_c/d_v$  is an integer is discussed in the following. The protograph of the  $(3, 6, L)$  spatially coupled LDPC ensemble has been provided in Figure 1, where the gray circles correspond to information bits and the white circles

associate with parity bits. As shown in the protograph, the parity bits  $\mathbf{v}_{l,1}$  in each position  $l$  are determined according to the information bits  $\mathbf{v}_{l,0}$  and the related bits  $\mathbf{v}_{l',0}, \mathbf{v}_{l',1}, 0 \leq l' < l$ , from the previous positions. In this way, a recursive encoder is implemented.

However, it is disappointing that for the case  $d_c/d_v$  is not an integer, we cannot perform recursive encoding as above. Let us consider the case  $d_v = 4$  and  $d_c = 6$ . The protograph of the LDPC block code, which acts as the smallest coupling unit, is given on the left of Figure 2. The protograph of a  $(d_v, d_c, L) = (4, 6, L)$  spatially coupled LDPC ensemble constructed according to [8] is provided in Figure 2a. The notations are the same as in Figure 1. The protograph consists of  $d'_c = 3$  variable nodes and  $d'_v = 2$  check nodes at each position. The variable node degree equals  $d_v$ , and the check node degree is  $d_c$ . At each position  $l$  in Figure 2a,  $\mathbf{v}_{l,0}$  denotes the information bits while  $\mathbf{v}_{l,1}$  and  $\mathbf{v}_{l,2}$  represent the parity bits, for  $0 \leq l < L$ . It can be seen that the parity bits  $\mathbf{v}_{l,1}$  (and  $\mathbf{v}_{l,2}$ ) can not be calculated only depending on  $\mathbf{v}_{l,0}$  and the encoded bits at former positions, as the two variable nodes associated with  $\mathbf{v}_{l,1}$  and  $\mathbf{v}_{l,2}$  connect to the same check node. Therefore, recursive encoding cannot be applied for this case.



**Figure 2.** Protograph of a  $(4, 6, L)$  spatially coupled LDPC ensemble with (a) the structure in [8]; (b) a modified structure.

Fortunately, the problem may have a solution using a modified protograph structure as illustrated in Figure 2b. Similarly,  $\mathbf{v}_{l,0}$  represents the information bits, and  $\mathbf{v}_{l,1}, \mathbf{v}_{l,2}$  are the parity bits, at each position  $l$ . The edge connection on the variable node associated with  $\mathbf{v}_{l,2}$  is rearranged. It avoids connecting to both the check nodes at the current position but connects to a check node one position away. By doing this, each check node connects with only one unknown variable nodes and  $d_c - 1$  known variable nodes. The parity bit  $\mathbf{v}_{l,1}$  can be calculated depending on  $\mathbf{v}_{l,0}$  and the encoded bits at former positions. Afterwards,  $\mathbf{v}_{l,2}$  can be obtained in turn through  $\mathbf{v}_{l,0}, \mathbf{v}_{l,1}$  and the encoded bits at former positions. After the modification, all the parity bits at the current position can be calculated depending on the information bits and the previously encoded parity bits. The recursive encoding is realized through a structure modification.

The recursive encoding significantly reduces the delay and the complexity compared to the encoding of LDPC block codes, and it promotes the application of spatially coupled LDPC codes in practice. Therefore, it is important to find an efficient and effective method to implement the recursive encoding for arbitrarily chosen node degrees. In the next section, we propose a simple but generalized principle for constructing the parity-check matrix and implement the recursive encoding for the proposed structures.

### 3. Recursive Encoding of Spatially Coupled LDPC Codes with Arbitrary Node Degrees

In the following we first propose a general method which guides the construction of the parity-check matrix for recursive encoding. Afterwards, the details of the encoding procedure is provided.

#### 3.1. General Principle on Constructing Parity-Check Matrices

Instead of using the  $d'_c \times d'_v$  structure as the smallest coupling unit, we redefine a structure consisting of  $d_c \times d_v$  permuted identity matrices as:

$$\mathbf{B}_l^T = \begin{bmatrix} \mathbf{P}_{0,0}^{(0)}(l) & \cdots & \mathbf{P}_{0,d_v-1}^{(d_v-1)}(l) \\ \vdots & \mathbf{P}_{i,j}^{(n)}(l) & \vdots \\ \mathbf{P}_{d_c-1,0}^{(0)}(l) & \cdots & \mathbf{P}_{d_c-1,d_v-1}^{(d_v-1)}(l) \end{bmatrix}, \quad (1)$$

where  $\mathbf{P}_{i,j}^{(n)}(l)$  is an  $M \times M$  random permutation matrix in the  $i$ -th row and the  $j$ -th column of  $\mathbf{B}_l^T$ , and  $n$  ( $0 \leq n < d_v$ ) represents that  $\mathbf{P}_{i,j}^{(n)}(l)$  is the  $n$ -th non-zero permutation matrix in the  $i$ -th row. By varying the offsets between adjacent rows, different types of coupling units are obtained. We define a pattern  $\mathbf{a} = (a_0, a_1, \dots, a_i, \dots, a_{d_c-1})$ , where  $a_i \in \{0, 1\}$  for  $i = \{0, 1, \dots, d_c - 1\}$ , to represent the coupling unit  $\tilde{\mathbf{H}}_l^T$  after row offsetting as:

$$\tilde{\mathbf{H}}_l^T = \begin{bmatrix} a_0 = 1 & \mathbf{P}_{0,0}^{(0)}(l) & \mathbf{P}_{0,1}^{(1)}(l) & \cdots & \mathbf{P}_{0,j+1}^{(j+1)}(l) & \cdots & \mathbf{P}_{0,d_v-1}^{(d_v-1)}(l) \\ \vdots & \ddots & \ddots & & \ddots & & \ddots \\ a_{i-1} & & & \mathbf{P}_{i-1,j}^{(0)}(l) & \mathbf{P}_{i-1,j+1}^{(1)}(l) & \cdots & \mathbf{P}_{i-1,j+d_v-1}^{(d_v-1)}(l) \\ a_i = 1 & & & \mathbf{P}_{i,j+1}^{(0)}(l) & \mathbf{P}_{i,j+2}^{(1)}(l) & \cdots & \mathbf{P}_{i,j+d_v-1}^{(d_v-2)}(l) & \mathbf{P}_{i,j+d_v}^{(d_v-1)}(l) \\ \vdots & & & \ddots & \ddots & & \ddots & \ddots \\ a_{d_c-1} & & & & \mathbf{P}_{d_c-1,d_v-1}^{(0)}(l) & \mathbf{P}_{d_c-1,d_v}^{(1)}(l) & \cdots & \mathbf{P}_{d_c-1,2d_v-2}^{(d_v-1)}(l) \end{bmatrix} \quad (2)$$

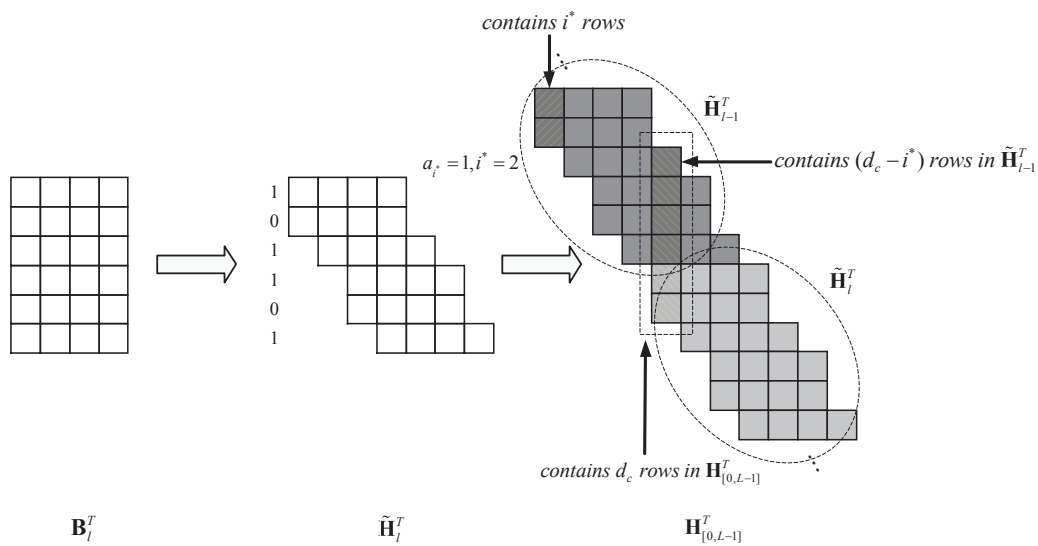
The element  $a_i = 1$  means that the beginning of the non-zero matrices in the  $i$ -th row shifts by one column compared to that in the  $(i-1)$ -th row, while  $a_i = 0$  indicates that the non-zero matrices in the  $i$ -th row are aligned with those in the  $(i-1)$ -th row, for  $0 < i < d_c$ . Note that  $a_0 = 1$  means that the beginning of the first row in  $\tilde{\mathbf{H}}_l^T$  shifts by one column compared to the last row of  $\tilde{\mathbf{H}}_{l-1}^T$ . Figure 3 illustrates how to obtain  $\tilde{\mathbf{H}}_l^T$  from  $\mathbf{B}_l^T$  according to pattern  $\mathbf{a}$ , and then how to assemble  $\mathbf{H}_{[0,L-1]}^T$  accordingly.

In the following we propose a theorem which selects all the parity-check matrix structures suitable for recursive encoding.

**Theorem 1.** For a coupling unit represented by  $\mathbf{a} = (a_0, a_1, \dots, a_i, \dots, a_{d_c-1})$ , if the pattern  $\mathbf{a}$  satisfies

$$\begin{cases} \sum_{i=0}^{d_c-1} a_i = d_v \\ a_0 = 1 \end{cases}, \quad (3)$$

the spatially coupled LDPC ensemble  $\{d_v, d_c, L\}$  constructed by coupling the unit of pattern  $\mathbf{a}$  allows recursive encoding.



**Figure 3.** The construction of the parity-check matrix for the  $(4, 6, L)$  spatially coupled LDPC ensemble.

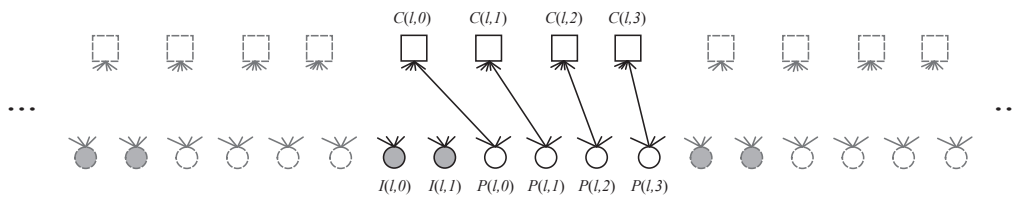
**Proof of Theorem 1.** We first prove that any matrix  $\mathbf{H}_{[0,L-1]}^T$  coupling from a structure  $\tilde{\mathbf{H}}_l^T$  satisfying condition (3) is the parity-check matrix of a spatially coupled LDPC ensemble. Each  $\mathbf{P}_{i,j}^{(n)}(l)$  is a permutation matrix, which implies that each column or each row of  $\mathbf{P}_{i,j}^{(n)}(l)$  only contains single one. Considering  $\sum_{i=0}^{d_c-1} a_i = d_v$  and  $a_0 = 1$ , the structure  $\tilde{\mathbf{H}}_l^T$  must have a form as in (2).

It is obvious that  $\tilde{\mathbf{H}}_l^T$  is a  $d_c \times (2d_v - 1)$  matrix, and each  $\mathbf{P}_{i,j}^{(n)}(l)$  is an element of it. Apparently each row in  $\tilde{\mathbf{H}}_l^T$  contains  $d_v$  non-zero columns, and so does  $\mathbf{H}_{[0,L-1]}^T$ . Then, as Figure 3 shows, the last  $d_v - 1$  columns of  $\tilde{\mathbf{H}}_{l-1}^T$  is aligned with the first  $d_v - 1$  columns of  $\tilde{\mathbf{H}}_l^T$  if  $a_0 = 1$ . Assuming that  $a_{i^*}$  represents the second 1 ( $a_0$  must be 1) in  $\mathbf{a}$ , all columns in row  $i'$ ,  $i' \geq i^*$ , has been shifted to the right by at least one column. Thus the first column in  $\tilde{\mathbf{H}}_{l-1}^T$  contains only  $i^*$  rows and the  $d_v$ -th column contains  $d_c - i^*$  rows. Note that the  $d_v$ -th column of  $\tilde{\mathbf{H}}_{l-1}^T$  corresponds to the first column of  $\tilde{\mathbf{H}}_l^T$ . After assembling  $\tilde{\mathbf{H}}_{l-1}^T$  and  $\tilde{\mathbf{H}}_l^T$ , the corresponding column in  $\mathbf{H}_{[0,L-1]}^T$  contains  $d_c$  rows. The same analysis can be applied to other columns except the beginning and terminating ends of  $\mathbf{H}_{[0,L-1]}^T$ . Therefore the matrix  $\mathbf{H}_{[0,L-1]}^T$  is the parity-check matrix of a spatially coupled LDPC ensemble with node degrees  $d_v$  and  $d_c$ .

We then prove why recursive encoding can be applied with this structure. We have assumed that the coupling unit has  $d_c$  variable nodes and  $d_v$  check nodes. We use  $C(l, k)$  to denote the  $k$ -th check node in the  $l$ -th coupling unit in the chain,  $0 \leq k < d_v$ . In each coupling unit, the number of variable nodes associated with the parity bits equals  $d_v$ , and the number of variable nodes corresponding to information bits is  $d_c - d_v$ . For convenience, we name the variable nodes associated with the parity bits as parity nodes, and we denote the  $k$ -th parity node in the  $l$ -th coupling unit as  $P(l, k)$ . The nodes related to the information bits are denoted as  $I(l, j)$ ,  $0 \leq j < d_c - d_v$ .

To guarantee recursive encoding, we set the following conditions on the edge connections from the check node  $C(l, k)$ : (i) there is one and only one edge connecting the check node  $C(l, k)$  and the parity node  $P(l, k)$ ; (ii) the remaining  $(d_c - 1)$  edges are only allowed to connect with the nodes in the set  $\{P(l', k') : d_v l' + k' < d_v l + k\} \cup \{I(l', j) : l' \leq l, 0 \leq j < d_c - d_v\}$ . A protograph which illustrates the above connections for  $d_v = 4$  and  $d_c = 6$  is given in Figure 4. Without the loss of generality, we assume the information bits are associated with the first two variable nodes in each coupling unit.





**Figure 4.** The connections for (4,6) spatially coupled LDPC codes assuming the information bits are associated with the first two variable nodes.

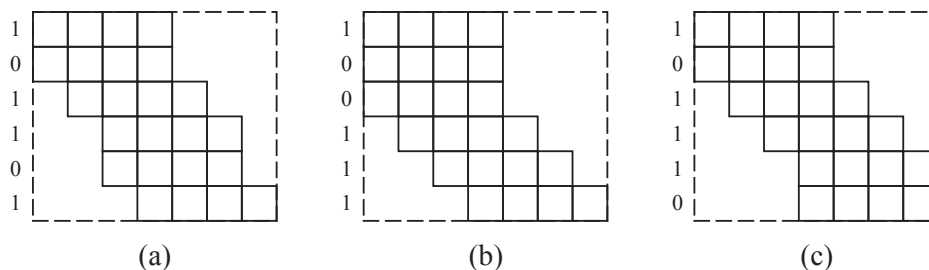
In the following we describe the above conditions in form of parity-check matrix. Each coupling unit is described by a  $d_c \times d_v$  structure of permutation matrices. Without disarranging the edge connections, we change the order of rows so that the first  $d_v$  rows relate to parity nodes  $P(l, k)$ ,  $0 \leq k < d_v$ . The indexes of the columns represent the check nodes  $C(l, k)$ . According to the condition (i), the permutation matrix at the  $k$ -th row and the  $k$ -th column for  $0 \leq k < d_v$  is non-zero. Meanwhile, based on the condition (ii), the permutation matrix at the  $k$ -th row and the  $k'$ -th column,  $k > k'$ ,  $0 \leq k, k' < d_v$ , must be a zero-matrix. We have defined that the element  $a_i = 1$  means that the beginning of the non-zero permutation matrices in the  $i$ -th row shifts by one column compared to that in the  $(i - 1)$ -th row. According to the conditions, there is obviously a number of  $d_v$  1s in the pattern **a**.  $\square$

As  $a_0 = 1$ , the above theorem selects a number of  $\binom{d_c-1}{d_v-1}$  structures which allow recursive encoding. However, some different structures may actually lead to the same edge connections in the chain. In order to screen out the structures which have truly different edge connections, we take into account the following restriction. The submatrices indexed by  $\mathbf{a}_a$  and  $\mathbf{a}_b$  are considered to result in different edge connections in the coupling chain only if they satisfy:

$$(\mathbf{a}_a)_N \neq \mathbf{a}_b \quad (4)$$

where  $(\mathbf{a}_a)_N$  indicates a sequence obtained by circular shifting the elements in  $\mathbf{a}_a$  for  $N$  times,  $N \in \{0, 1, \dots, d_c - 1\}$ .

Once the pattern **a** is obtained, the positions for the information bits are determined. If  $a_i = 0$ ,  $0 < i < d_v$ , the  $(i - 1)$ -th variable node is designated for one information bit. The structures selected by (3) and (4) for recursive encoding spatially coupled LDPC ensembles with  $d_v = 4$  and  $d_c = 6$  are  $\mathbf{a}_a = (1, 0, 0, 1, 1, 1)$ ,  $\mathbf{a}_b = (1, 0, 0, 1, 1, 1)$ , and  $\mathbf{a}_c = (1, 0, 1, 1, 1, 0)$ . The corresponding matrix structures are illustrated in Figure 5. The various structures lead to different performance which will be discussed later.



**Figure 5.** The coupling unit of (4,6,  $L$ ) spatially coupled LDPC ensembles with (a)  $\mathbf{a}_a = (1, 0, 1, 1, 0, 1)$ ; (b)  $\mathbf{a}_b = (1, 0, 0, 1, 1, 1)$  and (c)  $\mathbf{a}_c = (1, 0, 1, 1, 1, 0)$ .

The spatially coupled LDPC codes need to be initialized and terminated properly for finite  $L$ . In order to ensure that the spatially coupled LDPC codes with the modified structures exhibit good

performance, we try to maintain the degree of variable nodes for all the positions on the coupling chain but reduce the degree of check nodes at the boundary positions.

Take Figure 2b as an example. If we keep all the variable node degree to be  $d_v$ , there appears an additional check node at the end of the chain compared to Figure 2a. This check node will slightly reduce the design rate of the ensemble. On the other hand, if we remove this check node, the design rate will be the same as that of Figure 2a. For this case, the last variable node has a reduced degree  $d_v - 1$ . In this manuscript, we always choose to keep the variable node degree  $d_v$  on all the positions.

The same as in [6], the state of the register is non-zero after encoding. A solution to this problem has been proposed in [6], but it has to solve a system of linear equation which is complex. To solve the problem more efficiently, we adopt the efficient termination proposed in [9].

### 3.2. Implementation of Recursive Encoding

In this section, we introduce the implementation of recursive encoding based on the modified structure of spatially coupled LDPC codes as above.

To illustrate the recursive encoding, we first define the following notations. The subsequence of information bits at position  $l$  is defined as  $\mathbf{u}_l = [\mathbf{u}_{l,0}, \mathbf{u}_{l,1}, \dots, \mathbf{u}_{l,d_c-d_v-1}]$  for  $0 \leq l < L$ . The encoded subsequence at position  $l$  is denoted as  $\mathbf{v}_l = [\mathbf{v}_{l,0}, \mathbf{v}_{l,1}, \dots, \mathbf{v}_{l,d_c-1}]$ , and  $\mathbf{c}_l = [\mathbf{c}_{l,0}, \mathbf{c}_{l,1}, \dots, \mathbf{c}_{l,d_v-1}]$  is the parity-bit vector for  $0 \leq l < L$ . We have the entire code sequence as  $\mathbf{v}_{[0,L-1]} = [\mathbf{v}_0, \mathbf{v}_1, \dots, \mathbf{v}_{L-1}]$ , and it satisfies the equation

$$\mathbf{v}_{[0,L-1]} \tilde{\mathbf{H}}_{[0,L-1]}^T = 0. \quad (5)$$

We divide (5) into several sub-equations, and the  $l$ -th sub-equation is written as  $\mathbf{v}_l \tilde{\mathbf{H}}_l^T = \mathbf{p}_l = [\mathbf{s}_l, \mathbf{q}_l]$ ,  $0 \leq l < L$ . The vector  $\mathbf{p}_l = [\mathbf{s}_l, \mathbf{q}_l]$  is the partial syndrome from the  $l$ -th sub-equation, where  $\mathbf{s}_l = [\mathbf{s}_l^{(0)}, \mathbf{s}_l^{(1)}, \dots, \mathbf{s}_l^{(d_v-1)}]$  and  $\mathbf{q}_l = [\mathbf{q}_l^{(0)}, \mathbf{q}_l^{(1)}, \dots, \mathbf{q}_l^{(d_v-2)}]$ .

From the structure of  $\tilde{\mathbf{H}}_l^T$ , for  $1 \leq l < L$ , we have

$$\mathbf{p}_{l-1} = \mathbf{v}_{l-1} \tilde{\mathbf{H}}_{l-1}^T, \\ \mathbf{s}_l^{(k)} = \begin{cases} \mathbf{q}_{l-1}^{(k)} = \mathbf{p}_{l-1}^{(k+d_v)}, & 0 \leq k < d_v - 1 \\ \mathbf{0}, & k = d_v - 1 \end{cases}.$$

If  $\mathbf{P}_{i,j}^{(0)}(l)$  for  $a_{i+1} \neq 0$  ( $0 \leq i < d_c$ ,  $a_{d_c} = a_0 = 1$ ) is an identity matrix, then the parity bits can be calculated simply with the encoded bits obtained earlier. For  $0 \leq l < L$ , the  $i$ -th ( $0 \leq i < d_c$ ) encoded bit  $\mathbf{v}_{l,i}$  is determined depending on the information bits, the encoded parity bits and the partial syndrome as

$$\mathbf{v}_{l,i} = \begin{cases} \mathbf{u}_{l,g}, & a_{i+1} = 0 \\ \mathbf{c}_{l,k} = \mathbf{s}_l^{(k)} + \sum_{i'=0}^{i-1} \mathbf{v}_{l,i'} \mathbf{P}_{i',j}^{(\cdot)}(l), & a_{i+1} = 1 \end{cases}, \quad (6)$$

where  $\mathbf{u}_{l,g}$  and  $\mathbf{c}_{l,k}$  correspond to the coming information bits and the parity bits to be encoded, respectively. The indexes  $g$ ,  $k$  and  $j$  are determined by the realization of  $\tilde{\mathbf{H}}_l^T$ .

For  $l = 0$ ,  $\mathbf{s}_0 = \mathbf{0}$ , the encoder is initialized by

$$\mathbf{p}_0 = [\mathbf{0}, \mathbf{q}_0] = \mathbf{v}_0 \tilde{\mathbf{H}}_0^T,$$

and the encoded bits are

$$\mathbf{v}_{0,i} = \begin{cases} \mathbf{u}_{0,g}, & a_{i+1} = 0 \\ \mathbf{c}_{0,k} = \sum_{i'=0}^{i-1} \mathbf{v}_{0,i'} \mathbf{P}_{i',j}^{(\cdot)}[0], & a_{i+1} = 1 \end{cases} \quad (7)$$



According to (6) and (7), we can implement the recursive encoder of spatially coupled LDPC codes using shift-registers as in [6]. The encoder needs a number of  $(d_v - 1)M$  storage units, and the encoding complexity per bit is proportional to  $d_c$ .

#### 4. Theoretical Analysis

In this section, the design rate of the terminated spatially coupled LDPC ensemble with the modified structure is calculated. In addition, analysis based on density evolution is provided. For a fair comparison with [8], in the following we assume an idealized termination where the state of the shift-register comes back to zero. For the ease of presentation, we refer to the structure in [8] as the original structure in the rest of the paper.

##### 4.1. Design Rate

To get the design rate of a  $(d_v, d_c, L)$  spatially coupled LDPC ensemble, we calculate the total number of information bits and parity bits separately. Since there are  $M$  nodes for each position, the total number of variable nodes equals  $d_c LM$ , and the total number of check nodes is  $(d_v L + d_v - 1)M$ . Hence, the design rate of the spatially coupled LDPC code is

$$R = 1 - \frac{d_v L + d_v - 1}{d_c L} = 1 - \frac{d_v}{d_c} \times \frac{L + \frac{d_v - 1}{d_v}}{L}.$$

As we pointed out earlier, this rate is slightly lower than that of the original ensemble when  $\gcd(d_v, d_c) < d_v$ . When  $\gcd(d_v, d_c) = d_v$ , the modified ensemble is equivalent to the original structure, and the design rate is the same for both. For a large  $L$ , the rate reduction is negligible. When  $L$  tends to infinity, the design rate converges to  $1 - d_v/d_c$ .

##### 4.2. Density Evolution

In the following we derive the density evolution of the proposed code structures. For comparison, we plot the remaining erasure probabilities after iterations for both the original spatially coupled structure and the newly proposed structures.

There are  $d_c$  variable nodes and  $d_v$  check nodes at each position. Due to the variety of edge connections, we track the message passing on each individual node instead of each position. We use  $\mathcal{T}(k, n)$  to denote the set of indices  $(f, g)$  of the check nodes connected with variable node  $(k, n)$  and  $\mathcal{J}(f, g)$  to represent the set of indices  $(k, n)$  of the variable nodes related to check node  $(f, g)$ . The check node  $(f, g)$  is the  $g$ -th check node at position  $f$ , and variable node  $(k, n)$  is the  $n$ -th variable node at position  $k$ .

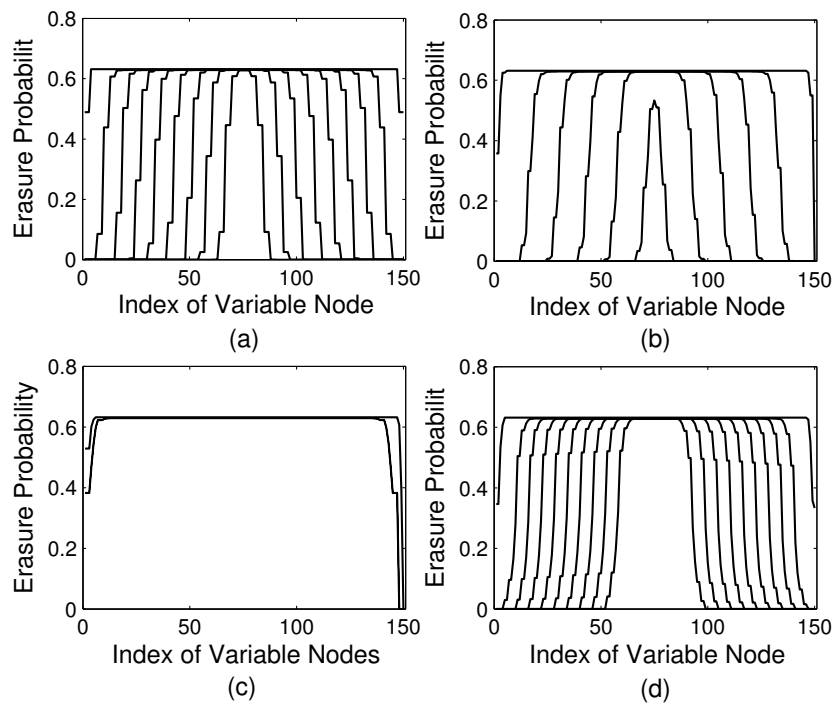
In the  $t$ -th iteration of the decoding, we use  $x_{k,f}^{n,g}(t)$  to denote the probability that the message from variable node  $(k, n)$  to check node  $(f, g)$  is an erasure, while the probability that the message from check node  $(f, g)$  to variable node  $(k, n)$  corresponds to an erasure is represented by  $y_{f,k}^{g,n}(t)$ . As the variable node  $(k, n)$  connects with the check nodes indexed by  $\mathcal{T}(k, n)$ ,  $x_{k,f}^{n,g}(t)$  can be updated as

$$x_{k,f}^{n,g}(t) = \varepsilon \prod_{\substack{(f',g') \in \mathcal{T}(k,n) \\ \text{s.t. } (f',g') \neq (f,g)}} y_{f',k}^{g',n}(t-1).$$

Similarly, for check node  $(f, g)$ , we get

$$y_{f,k}^{g,n}(t) = 1 - \prod_{\substack{(k',n') \in \mathcal{J}(f,g) \\ \text{s.t. } (k',n') \neq (k,n)}} [1 - x_{k',f}^{n',g}(t)].$$

In Figure 6, the density evolution for  $(d_v, d_c, L) = (4, 6, 50)$  spatially coupled LDPC ensemble with iteration numbers  $n = \{1, 101, 201, 301, \dots, 901\}$  is given for the binary erasure channel with erasure probability  $\varepsilon = 0.646$ . Figure 6a refers to the original spatially coupled LDPC ensemble in [8], while Figure 6b–d correspond to the spatially coupled LDPC ensembles modified for recursive encoding. The horizontal axis is the index of the variable nodes in the protograph, where every adjacent  $d_c$  variable nodes belong to the same position. The vertical axis presents the remaining erasure probability.



**Figure 6.** Density evolution for  $(4, 6, 50)$  spatially coupled LDPC ensembles with structure (a) in [8]; (b)  $\mathbf{a}_j = (1, 0, 1, 1, 0, 1)$ ; (c)  $\mathbf{a}_j = (1, 0, 0, 1, 1, 1)$  and (d)  $\mathbf{a}_j = (1, 0, 1, 1, 1, 0)$ .

The curves of density evolution reflect the irregularity of check node degree at the boundary positions. In the initialization, the degree of the left four check nodes is respectively  $\{3, 3, 6, 6\}$ ,  $\{2, 3, 5, 6\}$ ,  $\{3, 4, 5, 6\}$ , and  $\{2, 3, 4, 6\}$  for (a) to (d). In general, lower check node degree leads to better decoding performance, which is verified in Figure 6. We can see that (b) and (d) show an advantage over (a) and (c) on the left boundary of the protograph. Similar superiority is observed for (b) and (c) on the right boundary due to their smaller check node degree.

For the original spatially coupled LDPC ensemble, all the variable nodes at the same position has the same coupling behavior. Therefore, (a) exhibits a stair for every  $d_c$  variable nodes. In the protograph of the modified structures, even for the same position, the edge connection may be different from one variable node to another. Consequently, the curves of remaining erasure probability in (b), (c), and (d) are smoother than that in (a). Meanwhile, the manner of edge connection clearly affects the convergence behavior in iterative decoding. For the erasure probability we chose, the codes are in the water-fall region. We can see from the distance between neighboring curves, (b) converges the fastest among all the ensembles, and (c) performs the worst.

The BP decoding performance of spatially coupled LDPC ensembles is jointly determined by the boundary effect and the manner of edge connections. Combining the above observations, we expect that the ensemble associated with Figure 6b outperforms the others regarding the decoding threshold. The BP thresholds will be given in the next section.

On top of BP threshold, the performance of a spatially coupled LDPC code can be analyzed through the distance property of the ensemble. For example, the minimum distance growth rate of a spatially coupled LDPC ensemble can be studied via ensemble weight enumerator [30], and the smallest stopping/trapping set size growth can be studied via stopping/trapping set enumerator [31]. This manuscript mainly discuss the implementation aspects of the recursive encoding after modifying the structure, and a comprehensive performance analysis based on the distance property will be addressed in the future work.

## 5. Simulation Results and Discussions

In this section, we evaluate the performance of the proposed structures in the binary erasure channel numerically. We compare the BP thresholds of the spatially coupled LDPC codes after the modifications with the BP thresholds of the corresponding original structures in [8]. In addition, the bit erasure rates for different structures are provided.

We first provide the comparison of BP thresholds among different realizations of the modification. We choose the parameters  $d_v = 4$ ,  $d_c = 6$  and  $L = 50$  and carry out the density evolution for all the structures shown in Figure 5. The BP thresholds of the modified spatially coupled LDPC ensembles and the original implementation in [8] over the BEC are shown in Table 1, which are represented by  $\epsilon^{BP}$ . Besides,  $\epsilon^{Sh}$  denotes the Shannon limit corresponding to different design rate. As predicted in Section 4.2, the ensemble with the pattern (101101) performs the best among all the structures. The gap between the Shannon limit and the BP threshold is smaller than that of the original structure. The structure modification does not necessarily result in better BP threshold. However, a properly designed pattern can enable both easy encoding and better threshold.

**Table 1.** BP thresholds of (4, 6, 50) spatially coupled LDPC ensembles with different structures.

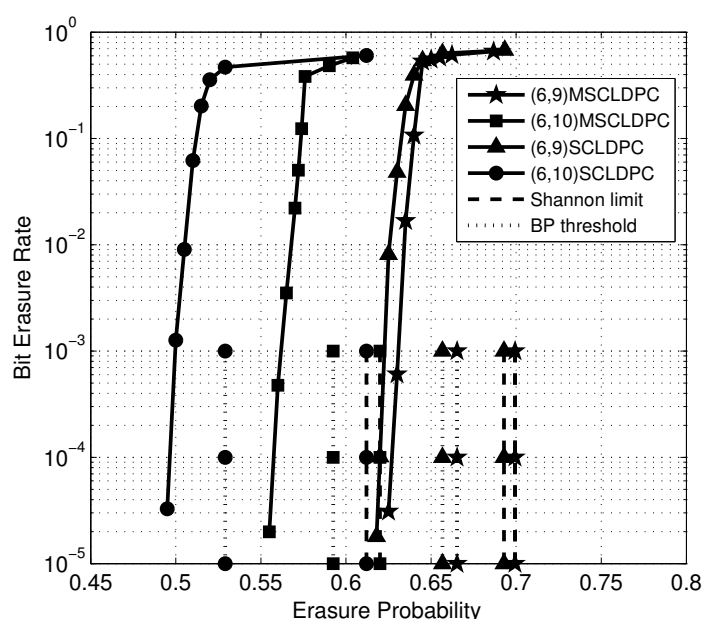
Pattern a	Design Rate	$\epsilon^{Sh}$	$\epsilon^{BP}$	Gap
Original	0.3200	0.6800	0.6567	0.0233
(101101)	0.3133	0.6867	0.6646	0.0221
(100111)	0.3133	0.6867	0.6364	0.0503
(101110)	0.3133	0.6867	0.6479	0.0388

Based on the observations in Table 1, we realize the modifications with properly selected patterns for different  $d_v$  and  $d_c$  and provide the BP thresholds in Table 2. The selected patterns are listed in the table, and the coupling length is  $L = 50$ . The gaps between the BP thresholds and the Shannon limits for the modified spatially coupled LDPC ensembles and for the original ensembles in [8] are represented by  $Gap$  and  $\widetilde{Gap}$ , respectively. Comparing  $Gap$  with  $\widetilde{Gap}$ , we can find that all the gaps are narrowed after proper modifications. For the modified structures, all the gaps are very small and can be further narrowed if we increase the parameters. Properly modified spatially coupled LDPC ensembles are able to provide capacity-approaching performance over the BEC. A similar conclusion can be extended to the general binary memoryless symmetric channel.

**Table 2.** BP thresholds of modified spatially coupled LDPC ensembles with different node degrees.

$(d_v, d_c)$	Pattern a	Design Rate	$\epsilon^{Sh}$	$\epsilon^{BP}$	Gap	$\widetilde{Gap}$
(4, 6)	(101101)	0.3133	0.6867	0.6646	0.0221	0.0233
(6, 9)	(101101101)	0.3007	0.6993	0.6653	0.0340	0.0361
(8, 12)	(101101101101)	0.2885	0.7115	0.6653	0.0462	0.0484
(6, 10)	(1001110011)	0.3800	0.6200	0.5925	0.0275	0.0830
(9, 15)	(100111001110011)	0.3686	0.6314	0.5925	0.0389	0.0945
(12, 20)	(10011100111001110011)	0.3577	0.6423	0.5925	0.0498	0.1055

Bit erasure rates for the modified spatially coupled LDPC codes and the original code over BECs are provided in Figure 7. Here, we set  $M = 500$  and  $L = 50$ , and we use the selected patterns as in Table 2. We also include the corresponding BP thresholds as the dashed lines and the Shannon limits as the dotted lines. Comparing the BER curves, we can find that the spatially coupled LDPC codes with properly modified structures clearly outperforms the original spatially coupled LDPC code in [8]. Note that, in addition to the good decoding performance, the proposed structure allows recursive encoding which facilitates the application of spatially coupled LDPC codes in practice.



**Figure 7.** Bit erasure rates of spatially coupled LDPC codes over binary erasure channels. In the legend, “MSCLDPC” represents the modified code while “SCLDPC” stands for the original code.

## 6. Conclusions

In this manuscript, structure modification for the spatially coupled LDPC code has been proposed, which enables the recursive encoding using a shift-register for arbitrary node degrees. The proposed principle provides feasible and practical guidance for the code construction and significantly reduces the computational complexity and memory requirement for the encoding. Performance analysis in terms of design rate and density evolution has been provided. The belief-prorogation decoding threshold is improved if proper edge rearrangement is applied in the modified spatially coupled LDPC ensemble. Finite-length simulation results are provided, which verify the theoretical analysis.

**Acknowledgments:** This work was supported in part by the National Natural Science Foundation of China (No. 61401037), the 863 project (No. SS2015AA011303), and the NSFC-RS joint funding grant (Nos. 61511130081 and IE141387).

**Author Contributions:** Zhongwei Si and Junyang Ma developed the first proposal of the study for the recursive encoding of spatially coupled LDPC codes with arbitrary rates. Zhongwei Si and Sijie Wang extended the study by proposing a generalized design principle and proving the theorem. The analysis and simulations were carried out by all the three authors. All authors have participated in writing the manuscript. All authors have read and approved the final manuscript.

**Conflicts of Interest:** The authors declare no conflict of interest. The founding sponsors had no role in the design of the study; in the collection, analyses, or interpretation of data; in the writing of the manuscript, and in the decision to publish the results.

## References

1. Gallager, R.G. *Low Density Parity Check Codes*; MIT Press: Cambridge, MA, USA, 1963.
2. Jimenez-Felström, A.; Zigangirov, K.S. Time-varying periodic convolutional codes with low-density parity-check matrix. *IEEE Trans. Inf. Theory* **1999**, *45*, 2181–2191.
3. Lentmaier, M.; Sridharan, A.; Costello, D.J., Jr.; Zigangirov, K.S. Iterative decoding threshold analysis for LDPC convolutional codes. *IEEE Trans. Inf. Theory* **2010**, *56*, 5274–5289.
4. Kudekar, S.; Richardson, T.; Urbanke, R. Threshold saturation via spatial coupling: Why convolutional LDPC ensembles perform so well over the BEC. *IEEE Trans. Inf. Theory* **2011**, *57*, 803–834.
5. Mitchell, D.G.M.; Pusane, A.E.; Zigangirov, K.S.; Costello, D.J., Jr. Asymptotically good LDPC convolutional codes based on protograph. In Proceedings of the 2008 IEEE International Symposium on Information Theory, Toronto, ON, Canada, 6–11 July 2008; pp. 1030–1034.
6. Pusane, A.E.; Jimenez-Felström, A.; Sridharan, A.A.; Lentmaier, M.M.; Zigangirov, K.S.; Costello, D.J., Jr. Implementation aspects of LDPC convolutional codes. *IEEE Trans. Commun.* **2008**, *56*, 1060–1069.
7. Tanner, R.M.; Sridhara, D.; Sridharan, A.; Fuja, T.E.; Costello, D.J., Jr. LDPC block and convolutional codes based on circulant matrices. *IEEE Trans. Inf. Theory* **2004**, *50*, 2966–2984.
8. Sridharan, A.; Truhachev, D.V.; Lentmaier, M.; Costello, D.J.; Zigangirov, K.S. Distance bounds for an ensemble of LDPC convolutional codes. *IEEE Trans. Inf. Theory* **2007**, *53*, 4537–4555.
9. Tazoe, K.; Kasai, K.; Sakaniwa, K. Efficient termination of spatially-coupled code. In Proceedings of the 2012 IEEE Information Theory Workshop, Lausanne, Switzerland, 3–7 September 2012; pp. 30–34.
10. Yedla, A.; Jian, Y.-Y.; Nguyen, P.S.; Pfister, H.D. A simple proof of threshold saturation for coupled scalar recursions. In Proceedings of the 2012 IEEE International Symposium on Turbo Codes and Iterative Information Processing, Gothenburg, Sweden, 27–31 August 2012; pp. 51–55.
11. Yedla, A.; Jian, Y.-Y.; Nguyen, P.S.; Pfister, H.D. A simple proof of threshold saturation for coupled vector recursions. In Proceedings of the 2012 IEEE Information Theory Workshop, Lausanne, Switzerland, 3–7 September 2012; pp. 25–29.
12. Kudekar, S.; Richardson, T.; Urbanke, R. Spatially coupled ensembles universally achieve capacity under belief propagation. In Proceedings of the 2012 IEEE International Symposium on Information Theory, Cambridge, UK, 1–6 July 2012; pp. 453–457.
13. Kumar, S.; Young, A.J.; Macris, N.; Pfister, H.D. Threshold saturation for spatially coupled LDPC and LDGM codes on BMS channels. *IEEE Trans. Inf. Theory* **2014**, *60*, 7389–7415.
14. Kudekar, S.; Pfister, H.D. The effect of spatial coupling on compressive sensing. In Proceedings of the Allerton Conference on Communications, Control, and Computing, Monticello, NY, USA, 29 September–1 October 2010; pp. 347–353.
15. Si, Z.; Thobaben, R.; Skoglund, M. Rate-compatible LDPC convolutional codes achieving the capacity of the BEC. *IEEE Trans. Inf. Theory* **2012**, *58*, 4021–4029.
16. Nitzold, W.; Lentmaier, M.; Fettweis, G.P. Spatially coupled protograph-based LDPC codes for incremental redundancy. In Proceedings of the International Symposium on Turbo Codes and Iterative Information Processing, Gothenburg, Sweden, 27–31 August 2012; pp. 155–159.
17. Si, Z.; Thobaben, R.; Skoglund, M. Bilayer LDPC convolutional codes for decode-and-forward relaying. *IEEE Trans. Commun.* **2013**, *61*, 3086–3099.
18. Schwandter, S.; Amat, A.G.; Matz, G. Spatially coupled LDPC codes for two-user decode-and-forward relaying. In Proceedings of the International Symposium on Turbo Codes and Iterative Information Processing, Gothenburg, Sweden, 27–31 August 2012; pp. 46–50.
19. Rathi, V.; Urbanke, R.; Andersson, M.; Skoglund, M. Rate-equivocation optimally spatially coupled LDPC codes for the BEC wiretap channel. In Proceedings of the 2011 IEEE International Symposium on Information Theory, Saint Petersburg, Russia, 31 July–5 August 2011; pp. 2393–2397.
20. Kudekar, S.; Kasai, K. Spatially coupled codes over the multiple access channel. In Proceedings of the 2011 IEEE International Symposium on Information Theory, Saint Petersburg, Russia, 31 July–5 August 2011; pp. 2816–2820.
21. Schlegel, C.; Truhachev, D. Multiple access demodulation in the lifted signal graph with spatial coupling. *IEEE Trans. Inf. Theory* **2013**, *59*, 2459–2470.

22. Yedla, A.; Pfister, H.D.; Narayanan, K.R. Universality for the noisy Slepian-Wolf problem via spatial coupling. In Proceedings of the 2011 IEEE International Symposium on Information Theory, Saint Petersburg, Russia, 31 July–5 August 2011; pp. 2567–2571.
23. Iyengar, A.R.; Papaleo, M.; Siegel, P.H.; Wolf, J.K.; Vanelli-Coralli, A.; Corazza, G.E. Windowed decoding of protograph-based LDPC convolutional codes over erasure channels. *IEEE Trans. Inf. Theory* **2012**, *58*, 2303–2320.
24. Ma, J.; Si, Z.; He, Z.; Niu, K. Recursive encoding of spatially coupled LDPC codes with arbitrary rates. In Proceedings of the 2015 IEEE International Symposium on Personal, Indoor and Mobile Radio Communications, Hong Kong, China, 30 August–2 September 2015; pp. 127–131.
25. Mitchell, D.; Lentmaier, M.; Costello, D.J. Spatially coupled LDPC codes constructed from protographs. *IEEE Trans. Inf. Theory* **2015**, *61*, 4866–4889.
26. Stinner, M.; Olmos, P.M. On the waterfall performance of finite-length SC-LDPC codes constructed from protographs. *IEEE J. Sel. Areas Commun.* **2016**, *34*, 345–361.
27. Johnson, S.; Lechner, G. Spatially coupled repeat-accumulate codes. *IEEE Commun. Lett.* **2013**, *17*, 373–376.
28. Divsalar, D.; Dolinar, S.; Jones, C.R.; Andrews, K. Capacity approaching protograph codes. *IEEE J. Sel. Areas Commun.* **2009**, *27*, 876–888.
29. Kasai, K.; Sakaniwa, K. Spatially-coupled MacKay–Neal codes and Hsu–Anastasopoulos codes. In Proceedings of the 2011 IEEE International Symposium on Information Theory, Saint Petersburg, Russia, 31 July–5 August 2011; pp. 747–751.
30. Divsalar, D. Ensemble weight enumerators for protograph LDPC codes. In Proceedings of the 2006 IEEE International Symposium on Information Theory, Seattle, MA, USA, 9–14 July 2006; pp. 1554–1558.
31. Abu-Surra, S.; Divsalar, D.; Ryan, W.E. Enumerators for Protograph-based ensembles of LDPC and generalized LDPC codes. *IEEE Trans. Inf. Theory* **2011**, *57*, 858–886.



© 2016 by the authors; licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC-BY) license (<http://creativecommons.org/licenses/by/4.0/>).