CORA Cork Open Research Archive
Cartlann Taighde Oscailte Chorcaí

Title	Optimized code design for constrained DNA data storage with asymmetric errors
Authors	Deng, Li;Wang, Yixin;Noor-A-Rahim, Md.;Guan, Yong Liang;Shi, Zhiping;Gunawan, Erry;Poh, Chueh Loo
Publication date	2019-06-26
Original Citation	Deng, L., Wang, Y., Noor-A-Rahim, M., Guan, Y. L., Shi, Z., Gunawan, E. and Poh, C. L. (2019) 'Optimized Code Design for Constrained DNA Data Storage With Asymmetric Errors', IEEE Access, 7, pp. 84107-84121. (14pp.) DOI: 10.1109/ACCESS.2019.2924827
Type of publication	Article (peer-reviewed)
Link to publisher's version	https://ieeexplore.ieee.org/document/8746106/figures#figures - 10.1109/ACCESS.2019.2924827
Rights	©This work is licensed under a Creative Commons Attribution 3.0 License. For more information, see http://creativecommons.org/licenses/by/3.0/ - https://creativecommons.org/licenses/by/3.0/
Download date	2026-11-03 02:27:08
Item downloaded from	https://hdl.handle.net/10468/8531

UCC
University College Cork, Ireland
Coláiste na hOllscoile Corcaigh

IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received June 10, 2019, accepted June 21, 2019, date of publication June 26, 2019, date of current version July 12, 2019.

Digital Object Identifier 10.1109/ACCESS.2019.2924827

Optimized Code Design for Constrained DNA Data Storage With Asymmetric Errors

LI DENG[1,2,5], YIXIN WANG[2], MD. NOOR-A-RAHIM[3], YONG LIANG GUAN[2], ZHIPING SHI[1], ERRY GUNAWAN[2], AND CHUEH LOO POH[4]

[1]National Key Laboratory of Science and Technology on Communications, University of Electronic Science and Technology of China, Chengdu 611731, China
[2]School of Electrical and Electronic Engineering, Nanyang Technological University, Singapore 639798
[3]School of Computer Science and IT, University College Cork, T12 K8AF Cork, Ireland
[4]Department of Biomedical Engineering, National University of Singapore, Singapore 119077
[5]School of Electronic Information and Automation, Guilin University of Aerospace Technology, Guilin 541004, China

Corresponding author: Chueh Loo Poh (poh.chuehloo@nus.edu.sg)

This work was supported in part by the Natural Science Foundation of China under Grant 61671128, and the Sichuan Key Research and Development Project under Grant 2019YFG0105, in part by the EDGE COFUND Marie Sklodowska Curie Grant under Agreement No. 713567, and in part by the Guangxi Natural Science Foundation under Grant 2018GXNSFAA281161, and the Guangxi Education Department Youth Science Foundation under Grant 2019KY0796.

ABSTRACT With ultra-high density and preservation longevity, deoxyribonucleic acid (DNA)-based data storage is becoming an emerging storage technology. Limited by the current biochemical techniques, data might be corrupted during the processes of DNA data storage. A hybrid coding architecture consisting of modified variable-length run-length limited (VL-RLL) codes and optimized protograph low-density parity-check (LDPC) codes is proposed in order to suppress error occurrence and correct asymmetric substitution errors. Based on the analyses of the different asymmetric DNA sequencer channel models, a series of the protograph LDPC codes are optimized using a modified extrinsic information transfer algorithm (EXIT). The simulation results show the better error performance of the proposed protograph LDPC codes over the conventional good codes and the codes used in the existing DNA data storage system. In addition, the theoretical analysis shows that the proposed hybrid coding scheme stores $\sim$1.98 bits per nucleotide (bits/nt) with only 1% gap from the upper boundary (2 bits/nt).

INDEX TERMS DNA data storage, protograph LDPC codes, asymmetric substitutions, constrained codes, DNA sequencing.

I. INTRODUCTION

With the advent of the big data era, the increasing capacity of the traditional storage media is gradually lagging behind the data growth, which increases the demand for exploring new storage medium. Considering the ultra-high density (bits per gram) and durability, DNA has been recognized as a new medium for long term data storage [1]. Generally, the four basic units (*nucleotide* (nt)) of DNA, denoted by 'A', 'T', 'C' and 'G', offer a twice larger capacity and a much higher density ($\sim$ 200Pbytes/g [2]) than the traditional binary systems ($\sim$ 200Gbytes/in^2 [3]). In addition, data stored in DNA can be readable after many generations as long as the DNA is preserved under favorable conditions [4]. Attracted by these significant features, several proof-of-concept DNA storage schemes have been implemented [2], [5]–[12].

The associate editor coordinating the review of this manuscript and approving it for publication was Zilong Liu.

Similar to the traditional data storage system, the DNA based data storage mainly includes three main processes of DNA synthesis (writing), storage and DNA sequencing (reading). During the processes, different types of errors occur, such as deletions, insertions, and substitutions. Many research works have been devoted to handle these errors from the following two aspects. First, since the DNA strands with a biased proportion of 'G' and 'C' (GC content) or long consecutively repetition of alphabets (homopolymer runs) are prone to sequencing errors [13], several mapping strategies such as differential mapping, constrained mapping, etc., have been designed to meet these biochemical constraints [6]–[12]. However, most of these mapping strategies are limited in mapping potential that indicates stored bits per nucleotide. Second, error correction coding schemes, such as single parity checking code, repetition code, Reed-Solomon code (RS) and fountain code [2], [6], [7], [9], [12], have been incorporated to address the data corruptions

This work is licensed under a Creative Commons Attribution 3.0 License. For more information, see http://creativecommons.org/licenses/by/3.0/

that occur in the processes, such as DNA synthesis, sample preparation and DNA sequencing. Most of these error correction schemes are implemented at DNA strand-level (similar to packet-level coding in traditional communication scenarios), which correct the erroneous DNA strands based on other error-free strands. These schemes have performed well for DNA storage prototypes with relative short DNA strands (i.e., ∼200nt in [2], [6], [7], [9], [10], [12]). However, these strand-level correction schemes might suffer from high decoding complexity and decoding failure for DNA data storage with long DNA strands (e.g., 1000 base pair in [8], [11]) due to the limited accuracy of the current DNA sequencing techniques, e.g., Nanopore sequencing [14].

Recently, the binary Low-density parity-check (LDPC) codes with a turbo-like decoder within the DNA strand have been designed to correct the asymmetric substitution errors caused by Nanopore sequencing in [15]. However, the degree distribution of the LDPC codes used in [15] is obtained using the conventional density evolution [16] regardless of the characteristics of asymmetric DNA sequencer channels. Additionally, although the substitution probability of each DNA symbol is asymmetric, the uniform distribution of the stored DNA symbols gives the mutual information (MI) which is very close to the capacity with the best distribution under practical channel parameters, thus the linear block codes designed for uniform codeword symbols are suggested in [15]. In other words, the design methods of LDPC codes over the traditional binary asymmetric channel (i.e., Z channel), such as methods in [17], [18], are not suitable for the asymmetric DNA sequencer channels. Because in Z channel, only one type of errors occurs (e.g., symbol '1' could be corrupted to symbol '0'), which requires an asymmetric (biased) distribution of the transmitted symbols (e.g., suppressing the occurrence of symbol '1' in the codeword) to approach the channel capacity.

In this work, the protograph-based LDPC codes, a subclass of the LDPC codes with low complexity and good error performance [19], [20], is used for error correction over the DNA data storage channel. Due to the superior behaviors over the additive white Gaussian noise (AWGN) channels, the designs of the protograph LDPC codes for different communication channels have been well studied [21]–[23]. However, the analysis algorithms of the protograph LDPC codes over the traditional communication systems are not suitable for the DNA data storage channels where the channel parameters are different. Thus, the traditional code optimization methods could not be applied in a straight forward manner for DNA data storage systems.

Motivated by the above concerns, addition to the observations of distinct behaviors of asymmetric substitution errors in two widely utilized sequencers (i.e., Nanopore sequencers [24] and Illumina's NextSeq sequencers [12]), the optimized code design for constrained DNA data storage with asymmetric errors is proposed. The contributions of this work can be concluded as follows. First, we propose a hybrid coding architecture consists of modified variable-length run-length limited (VL-RLL) constrained codes and protograph LDPC codes at the nucleotide level (within the DNA strand).[1] Second, we further optimize the protograph LDPC codes in this hybrid coding system based on the capacity analysis of the asymmetric DNA data storage channel. A modified extrinsic information transfer (EXIT) algorithm is proposed to estimate the error performance of protograph LDPC codes; and the code optimizations are processed accordingly to achieve better error performance for both Nanopore sequencer channel and Illumina sequencer channel.

The remainder is organized as follows. Section II contains a preliminary introduction of the constrained DNA data storage, the VL-RLL codes and the traditional protograph LDPC codes. Section III gives the asymmetric channel models based on two different DNA sequencing techniques. Section IV describes the proposed hybrid coding architecture with modified VL-RLL codes. Section V presents the optimization of protograph LDPC codes for DNA data storage. Section VI presents several simulation results and discussion, followed by the conclusion in Section VIII.

II. PRELIMINARIES

A. CONSTRAINED DNA DATA STORAGE SYSTEM

DNA strands with limited maximum homopolymer runs are desired to avoid sequencing errors [13], [14], so that the DNA data storage can be regarded as an M-ary run-length limited (RLL) constrained system due to the limit on the maximum homopolymer runs. The M-ary RLL constrained system is commonly denoted by (M, d, k) as the $(M, d+1, k+1)$ RLL codes can be easily constructed from the (M, d, k) constrained codes via a differential operation. The $(M, d+1, k+1)$ RLL codes have the minimum $d+1$ run-length and the maximum $k+1$ run-length, and the (M, d, k) constrained codes consist of codewords with at least d and at most k zeros between consecutive non-zeros. The differential operation is based on $y_i = y_{i-1} + x_i \pmod M$, where y_i is the current processing symbol, y_{i-1} is the last processed symbol, and x_i is the current symbol from the (M, d, k) constrained codes. In other words, the symbol in a (M, d, k) constrained codeword works as a transition symbol that performs between two neighboring symbols in the associative $(M, d+1, k+1)$ RLL codeword. For convenience, the (M, d, k) constrained codeword is denoted as a *transition word*.

With the maximum homopolymer run to 3nt, DNA data storage becomes a (4, 0, 2) constrained system, representing by a finite state transition diagram (FSTD) as Fig. 1 [25], in which the state s_i, where $i \in \{0, 1, \ldots, k\}$, records i consecutively repetitive zeros in the output (4, 0, 2) transition words. The output words are generated by the labels of the

[1] Part of this work has been submitted to 2019 IEEE Global Communications Conference (GLOBECOM).

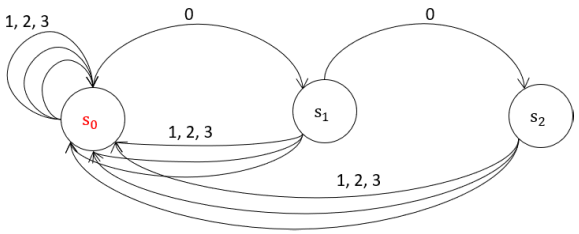


FIGURE 1. Finite state transition diagram of (4, 0, 2) constrained DNA storage.

edges of a path in the FSTD before transited into the (4, 1, 3) RLL codewords via the differential operation.

B. VL-RLL CODES FOR DNA DATA STORAGE

In this work, we employed the variable-length RLL codes rather than the fixed-length RLL codes since the former exhibits much lower coding complexity for achieving an equivalent coding rate with the latter. However, we found that the error-propagation in VL-RLL codes is more severe than the fixed-length codes as the variable-length characteristics of the source words and codewords might bring in additional synchronization problems. To address this, we have devised a new coding architecture with modified VL-RLL codes in Section IV.

To construct the VL-RLL code for the $(4, 0, 2)$ DNA data storage, we first generate a basic set consisting of variable-length $(4, 0, 2)$ transition words. By starting from and ending with the state s_0 in Fig. 1, we build a finite set ($\{1, 2, 3, 01, 02, 03, 001, 002, 003\}$) of which each element can be arbitrarily concatenated into long words that ultimately comply with the run-length limit [25]. The concatenations of elements in this basic set can be used as the $(4, 0, 2)$ transition word set to establish the bijection with the source word set. For simplicity, we directly use the basic set as the transition word set mapping to the source data. With the assumption of i.i.d binary source data, the variable-length source words are assigned to the variable-length codewords via the Huffman approach [26], optimizing the mapping potential (bits per symbol) of the code.

$$R = \frac{\sum_i 2^{-l_i} l_i}{\sum_i 2^{-l_i} o_i} \tag{1}$$

where l_i and o_i indicate the lengths of ith source word and transition word in Table 1, respectively. An example of encoding binary sequence to VL-RLL DNA sequence is shown as follows. Supposing we have a binary sequence '1110-10-01-1101-00-111100'. According to the mapping rule in Table 1, we obtain the transition sequence '03-3-2-02-1-001'. After the differential operation, the transition sequence is transited to the RLL sequence '03-2-0-02-3-330'. By mapping '0'→'A', '1'→'T', '2'→'G', and '3'→'C', we construct the final RLL DNA sequences 'ACGAAGCCCA', avoiding homopolymer longer than 3nt.

TABLE 1. VL-RLL mapping rule.

Source word	00	01	10	1100	1101	1110	111100	111101	11111
Transition word	1	2	3	01	02	03	001	002	003

C. PROTOGRAPH LDPC CODES

LDPC codes are well-known error correction codes due to the capacity-approaching and parallel decoding properties. In this paper, we consider a sub-type of LDPC codes known as the protograph-based LDPC codes, which have excellent error performance and low complexity. According to [20], [27], [28], the following sub-structures should be included in a good protograph design over the AWGN channel:

1) The sub-structures leading to good decoding threshold performance: at least one high-degree variable node (VN) and a certain proportion of degree-2 VNs are needed.

2) The sub-structures leading to good error floor performance: the number of degree-2 VNs should be no more than the total number of CNs except the one for the precoder if exists (see the definition in 3)). Too many degree-2 VNs would destroy the linear minimum distance growth property. Instead, VNs with degree-3 or higher can be added to maintain the linear minimum distance growth property for family codes construction.

3) A precoder structure for the punctured protograph codes: the precoder is a rate 1 accumulator including a degree-1 VN, a high-degree punctured VN, and a check node (CN) connecting the two VNs, which can both lead to good performances on the decoding threshold and the error floor.

The accumulate-repeat-by-4-jagged-accumulate (AR4JA) codes are considered as one kind of good protograph codes which have all the above-mentioned sub-structures, and perform well in the AWGN channel. Fig. 2 shows the protographs of the AR4JA family with rates 1/2 and higher [20], where the black circles and the white circles with cross represent VNs and CNs, respectively; and the white circles indicate the punctured VNs. The protographs can be described by the base matrix $\mathbf{B} = (b_{i,j})$, where the value of entry $b_{i,j}$ indicates the number of edges connecting the ith CN and the jth VN. Eq. (2) shows the base matrix of the AR4JA family codes, corresponding to the protographs in Fig. 2. The parity-check matrix is generated by lifting the base matrix, which can be summarized by copy-and-permute [19]. The progressive edge-growth (PEG) algorithm [29] is generally used to construct the parity-check matrix of LDPC codes with large girth, which is also applicable to lift the basematrix of the protograph LDPC codes.

$$\mathbf{B}_{AR4JA} = \begin{pmatrix} 1\,2\,0\,0\,0 & \overbrace{0\,0\,\cdots\,0\,0}^{2n} \\ 0\,3\,1\,1\,1 & 3\,1\,\cdots\,3\,1 \\ 0\,1\,2\,2\,1 & 1\,3\,\cdots\,1\,3 \end{pmatrix} \tag{2}$$

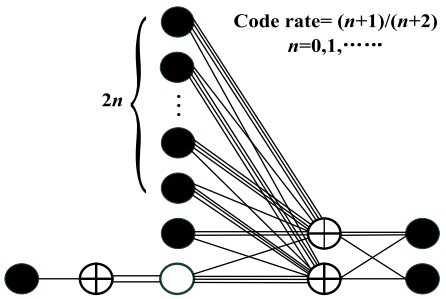


FIGURE 2. The protographs of AR4JA family with rates 1/2 and higher.

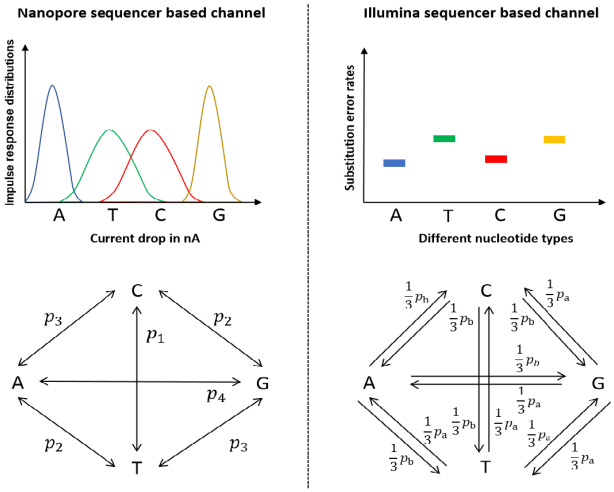


FIGURE 3. Two asymmetric channel models: Figures in the left refer to the Nanopore sequencer channel [24], and figures in the right refer to the Illumina sequencer channel [12].

III. CHANNEL MODELS

In the DNA based data storage system, errors might occur at any stage, including DNA synthesis, sample preparation, storage and DNA sequencing. In this work, we focus on the substitution errors that occur in the DNA sequencing process. In this section, two asymmetric error models based on the widely utilized sequencing techniques (i.e., Nanopore sequencing and Illumina sequencing) are presented.

A. NANOPORE SEQUENCER BASED ASYMMETRIC ERROR MODEL

In an ideal Nanopore sequencing process, the DNA strands migrate through the Nanopore at a constant rate, and one nucleotide of the DNA strands is read at a given time [14]. Due to the different atomic structures, the nucleotide can be detected by observing the changes of the ionic current drop. The current drop responses of nucleotides in the Nanopore sequencer were reported in [24] (the top left of Fig. 3). As can be seen, the overlapping drop distribution between the nucleotides 'T' and 'C' is the most significant; and the nucleotides 'A' and 'G' seem much less likely to overlap with each other, which indicates a higher mutation probability between 'T' and 'C'. Based on these observations, the Nanopore sequencer based asymmetric error model is built as shown at the bottom left of Fig. 3. Note that the error model is similar to the proposed model in [15], while consists of four error probabilities among nucleotides represented by p_1, p_2, p_3, and p_4, defined as

$$p_1 = 4\alpha, \quad p_2 = \alpha, \; p_3 = 0.01, \; p_4 \approx 0 \tag{3}$$

where α is the parameter related to the asymmetric error of the Nanopore sequencer channel, which satisfies $\alpha \in (0, 0.198)$, and $p_4 \ll p_3 < p_2 < p_1$.

The capacity of the Nanopore sequencer channel is analyzed in Appendix A.

B. ILLUMINA SEQUENCER BASED ASYMMETRIC ERROR MODEL

The Illumina sequencing is based on a sequence-by-synthesis approach, where four reversible dye-terminators are used to identify each added nucleotide in the strand. Based on the laser-induced excitation of the fluorophores and imaging, the nucleotides in the strand are determined [30].

From the experimental error analysis of [12], we could find that the asymmetric errors also exist in the Illumina's NextSeq process. Specifically, the substitution error probabilities of 'T' and 'G' (p_a) are higher than the substitution error probabilities of 'A' and 'C' (p_b). Moreover, p_a is approximately 1.5 times higher than p_b as shown in the top right of Fig. 3. Accordingly, the Illumina sequencer based asymmetric substitution error model is built with the assumption of equal mutation probabilities from one nucleotide to the other three nucleotides as shown in the bottom right of Fig. 3. In this model, the substitution probabilities p_a and p_b follow

$$p_a = 1.5\beta, \quad p_b = \beta \tag{4}$$

where β is the parameter related to the asymmetric error of the Illumina sequencer channel, which satisfies $\beta \in (0, 2/3)$.

The capacity of the Illumina sequencer channel can be found in Appendix B.

IV. HYBRID CODING ARCHITECTURE WITH MODIFIED VL-RLL CONSTRAINED CODES

The proposed coding architecture is shown in Fig. 4. This hybrid coding scheme writes the source data into the homopolymer-constrained DNA record (dotted red block), while protecting the encoded data from the asymmetric substitution errors occurring in the process of DNA sequencing upon a retrieval request (dotted blue block). In the encoding process (dotted red block), the binary message bits and the corresponding redundancy bits generated by the LDPC encoder are encoded by different constrained encoders to satisfy the homopolymer constraint before sending to the DNA data storage channel. After DNA sequencing, the DNA strands are sent to the decoding process (dotted blue block) for recovering the original user data. The sum-product algorithm (SPA) (the belief propagation (BP) algorithm in the *log*-domain) is implemented in the LDPC decoder using the Log-Likelihood Ratios (LLRs) passed from the received data over the channel.

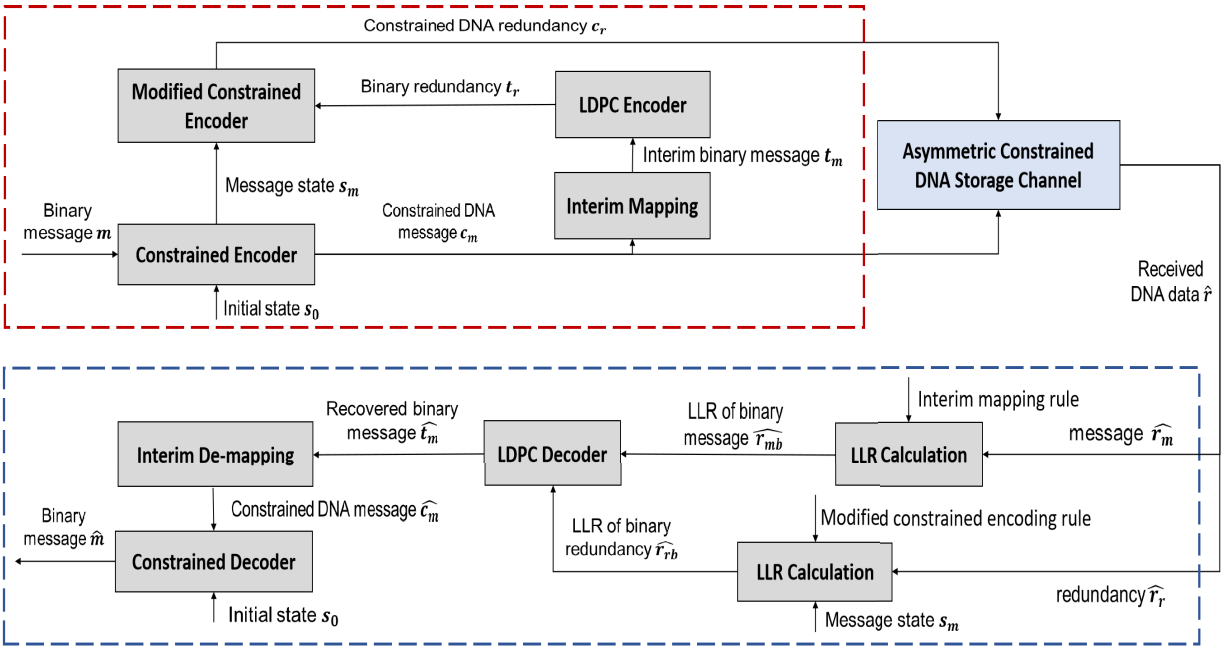


FIGURE 4. Proposed hybrid encoding/decoding for DNA data storage.

A. ENCODING WITH MODIFIED VL-RLL CONSTRAINED CODES

Firstly, the source binary message $\boldsymbol{m}$ is sent into the constrained encoder, i.e., VL-RLL encoder based on Table. 1. An output DNA message sequence $\boldsymbol{c_m}$ satisfying homopolymer constraint is obtained, which is a part of the entity that would be sent to the storage channel. Meanwhile, the sequence $\boldsymbol{c_m}$ is mapped into a binary message sequence $\boldsymbol{t_m}$ via an interim mapping that directly maps 'A' to '00', 'T' to '01', 'G' to '10', and 'C' to '11'. The mapping is chosen to keep consistent with the VL-RLL encoding in the previous step. After the interim mapping, a systematic binary LDPC encoding is performed based on the mapped binary message $\boldsymbol{t_m}$, generating parity check sequence $\boldsymbol{t_r}$.

Next, we convert the parity check sequence into a form that complies with the homopolymer (run-length) constraint for further storage. Through a modified constrained encoder, i.e., modified VL-RLL encoder based on Table. 2, the parity check sequence $\boldsymbol{t_r}$ is converted into constrained DNA redundancy sequence $\boldsymbol{c_r}$, and then attaching at the right of the reserved constrained DNA message $\boldsymbol{c_m}$. Note that a message state $\boldsymbol{s_m}$, that indicates the last alphabet of the information block $\boldsymbol{c_m}$, is fed to the encoding process of the parity check sequence $\boldsymbol{t_r}$, to ensure that the resultant DNA sequence consisting of $\boldsymbol{c_r}$ and $\boldsymbol{c_m}$ satisfy the homopolymer constraint. The constructed DNA sequences are then synthesized and stored in the DNA data storage.

Instead of performing LDPC encoding before constrained mapping (which is commonly used in the existing works), we move the LDPC encoding after the constrained mapping of the source bits. This step enables us to correct the channel errors that occur on the constrained codewords (which are the exact stored entity) before the errors diffuse to the recovered source data in the reverse process of the differential operation and mapping, which is the basic of most of the constrained codes. This design pipeline exhibits much importance as the error-propagation is much more severe in variable-length constrained codes (as we have used) than the conventional block codes. For the sake of simplicity, we consider binary LDPC code instead of quaternary LDPC code and hence use an interim mapping (to convert quaternary data to binary data) on the constrained DNA message $\boldsymbol{c_m}$. In other words, with quaternary LDPC code, we need to transform the unrestricted quaternary redundancy to the constraint-satisfying quaternary redundancy, which might incur more computation cost than the binary case.

TABLE 2. Modified VL-RLL Mapping Rule.

Source word	00	01	10	1100	1101	1110	111100	111101	11111X
Transition word	1	2	3	01	02	03	001	002	003

As seen in Fig. 4, a modified constrained encoder is designed for the redundancy (parity check) bits. In this modified VL-RLL mapping, we only change the last mapping of 11111→003 in the original VL-RLL mapping (Table 1) to 11111X→003 (Table 2), where 'X' represents either '0' or '1'. By using this fuzzy bit in the source word while mapping to a fixed transition word, we avoid the synchronization problem that may arise in the original VL-RLL mapping (i.e., all bijections have a rate 2 bits/symbol except the last one with rate 5/3 bits/symbol) at the cost of sacrificing the accuracy of de-mapping this fuzzy bit. The lost accuracy of de-mapping is the reason why we only use this mapping for the

redundancy bits ($\boldsymbol{t_r}$) generated by the LDPC codes rather than for the original message block ($\boldsymbol{m}$). Another merit of using the fuzzy bit in the modified constrained encoder is that it offers a mapping potential close to the theoretical limit. Mapping potential (also known as coding potential in [2], [12]) representing the number of bits encoded in one nucleotide, is considered as an important measure of DNA data storage. It can be seen that the upper boundary of mapping potential is 2 bits/nt. However, the practical mapping potential might be reduced due to the biochemical constraints, such as balanced GC content and maximum homopolymer runs. It has been discussed in [2], the mapping potential is reduced to 1.98 bits/nt by limiting the maximum homopolymer run to 3nt. However, by using the modified VL-RLL mapping with the fuzzy bit, the mapping potential approaches 2 bits/nt while the encoded redundancy blocks still satisfy the 3nt maximum homopolymer run constraint.

Now we analyze the overall mapping potential by using VL-RLL and modified VL-RLL for constrained encoding the information bits and the parity-check bits of the LDPC codewords, respectively. As discussed, VL-RLL offers an average mapping potential of $\mathbf{R_{info}} = 1.976$ bits/nt based on Eq. (1). Meanwhile, based on Table 2, the mapping potential of the modified VL-RLL becomes $\mathbf{R_{red}} \simeq 2$ bits/nt leveraging by the fuzzy bit. Thus, the average overall mapping potential becomes

$$\mathbf{R_{all}} = \frac{k + \frac{2k(1-\mathcal{R})}{\mathbf{R_{info}}\cdot\mathcal{R}}}{\frac{k}{\mathbf{R_{info}}} + \frac{2k\cdot(1-\mathcal{R})}{\mathbf{R_{info}}\cdot\mathbf{R_{red}}\cdot\mathcal{R}}}$$

where k is the length of the original information bits ($\boldsymbol{m}$), $\mathcal{R}$ is the code rate of the LDPC codes. Thus, we obtain the mapping potential,

$$\mathbf{R_{all}} \simeq (2 - 0.024\mathcal{R}) \tag{5}$$

B. DECODING WITH DIFFERENT LLR CALCULATIONS FOR THE INFORMATION AND THE REDUNDANCY

In this subsection, the decoding process is described with specific calculation methods of LLRs for the information blocks and the redundancy blocks. To recover the source data from the storage, we perform decoding on the received DNA data $\hat{\boldsymbol{r}}$ from the DNA sequencing process. With the assumption of the knowledge of the boundary between information blocks and redundancy blocks in the decoder, the received DNA data are first separated into information block $\widehat{\boldsymbol{r_m}}$, denoted by $r_{m_1} r_{m_2} \ldots r_{m_{i-1}} r_{m_i}$ and redundancy block $\widehat{\boldsymbol{r_r}}$, denoted by $r_{r_1} r_{r_2} \ldots r_{r_{j-1}} r_{r_j}$. This is because the nucleotide alphabets in these two blocks, i.e., r_{m_i} in the information block and r_{r_j} in the redundancy block, pass the channel information to bits of the LDPC codeword in different ways.

Specifically, for ith received nucleotide r_{m_i} in the information block $\widehat{\boldsymbol{r_m}}$, its associative received binary bits (b_i^1, b_i^2) in the LDPC codeword is determined by the interim mapping. The received alphabet r_{m_i} with the channel information, supplies the initial LLRs $(\mathcal{L}_{i^1}^0, \mathcal{L}_{i^2}^0)$ to the correlative two bits, facilitating the LDPC decoding using SPA algorithm. However, for jth received nucleotide r_{r_j} in the redundancy block $\widehat{\boldsymbol{r_r}}$, finding the associative binary bits are not as straightforward as in the information block. This is because in the redundancy block, the nucleotide alphabets are encoded from binary bits via a constrained encoding, in which the differential operation implicates that two neighboring nucleotide alphabets in the constrained codeword jointly determine a relevant transition symbol before de-mapping to the associative binary bits.

In the following, we introduce how the asymmetric channel information is passed to the initial LLR of each encoded binary bit in the LDPC codewords. We first explain the LLRs of binary bits that relate to the received information block $\widehat{\boldsymbol{r_m}}$. For a pair of ith received nucleotide alphabet r_{m_i} in $\widehat{\boldsymbol{r_m}}$ and ith stored nucleotide alphabet c_{m_i} in $\boldsymbol{c_m}$, we have $\Pr(x_i = c_{m_i} | y_i = r_{m_i})$ indicating the event probability, which is represented by the substitution probability in the channel models in Fig. 3. The initial LLRs $(\mathcal{L}_{i^1}^0, \mathcal{L}_{i^2}^0)$ of the associative (b_i^1, b_i^2) thus can be estimated on the basis of the interim mapping, i.e., A→00, T→01, G→10, C→11. The initial LLR of the relevant bit is derived by,

$$\mathcal{L}_{i^k}^0 = \log \frac{\Pr(b_i^k = 0 | y_i = r_{m_i})}{\Pr(b_i^k = 1 | y_i = r_{m_i})}$$

where $k \in \{1, 2\}$. An example for the Nanopore sequencer channel is shown as follows. If $r_{m_i} = A$, we have,

$$\begin{aligned}
\mathcal{L}_{i^1}^0 &= \log \frac{\Pr(x_i = A | y_i = A) + \Pr(x_i = T | y_i = A)}{\Pr(x_i = G | y_i = A) + \Pr(x_i = C | y_i = A)} \\
&= \log \frac{(1 - p_2 - p_3 - p_4) + p_2}{p_4 + p_1} \\
\mathcal{L}_{i^2}^0 &= \log \frac{\Pr(x_i = A | y_i = A) + \Pr(x_i = G | y_i = A)}{\Pr(x_i = T | y_i = A) + \Pr(x_i = C | y_i = A)} \\
&= \log \frac{(1 - p_2 - p_3 - p_4) + p_4}{p_2 + p_3}
\end{aligned}$$

The LLRs of received 'T', 'C' and 'G' in the information blocks can be estimated in similar way.

Next, we explain the LLRs associated with the redundancy block $\boldsymbol{c_r}$. As $\boldsymbol{c_r}$ is processed from the transition sequence that consists of transition words via a differential operation, one corrupted nucleotide in the received redundancy block $\widehat{\boldsymbol{r_r}}$ induces two erroneous transition symbols in the transition sequence, potentially resulting in burst errors in the corresponding binary redundancy $\boldsymbol{t_r}$ after performing the reverse of the mapping rule in Table. 2. In other words, each transition symbol that relates to two bits in the parity-check of the LDPC codeword is relevant to two neighboring nucleotide alphabets in the received constrained redundancy block $\widehat{\boldsymbol{r_r}}$. Thus, we consider two received nucleotides at the same time for determining the transition symbol before passing the initial LLRs to two parity-check bits in the LDPC codeword based on Table. 2. The event probability thus relies on the

substitution probabilities of each pair of the neighboring nucleotides ((*j-1*)th and *j*th) in $\boldsymbol{t_r}$, representing by $\Pr(x_{j-1}x_j = c_{r_{j-1}}c_{r_j}|y_{j-1}y_j = r_{r_{j-1}}r_{r_j})$. The initial LLRs $(\mathcal{L}^0_{j^1}, \mathcal{L}^0_{j^2})$ thus can be estimated on the basis of the modified VL-RLL mapping as shown in Table. 2.

Based on Table 2, it can be found that except symbol '3', each transition symbol in the transition words is uniquely mapped to two binary bits in the source word (i.e., 1→00, 2→01, 0→11). The transition symbol '3' is involved in three transition words, i.e., '3', '03', '003'. In transition words '3' and '03', the symbol '3' is mapped to the binary bits '10' in the corresponding source words. However, in the case of '003', '3' can be mapped to either '10' or '11' in the source word '11111X'. With the i.i.d assumption of binary bits in the source word, we can derive the probabilities p_w for symbol '3' mapping to binary bits '10',

$$p_w = \frac{p_{10} + p_{1110} + \frac{1}{2}p_{11111X}}{p_{10} + p_{1110} + p_{11111X}} = \frac{2^{-2} + 2^{-4} + 2^{-7}}{2^{-2} + 2^{-4} + 2^{-6}} = \frac{41}{42} \tag{6}$$

then the probability of symbol '3' mapping to binary bits '11' is $1 - p_w = \frac{1}{42}$.

The initial LLR of the relevant bit is computed by,

$$\mathcal{L}^0_{j^k} = \log \frac{\Pr(b^k_j = 0|y_{j-1}y_j = r_{r_{j-1}}r_{r_j})}{\Pr(b^k_j = 1|y_{j-1}y_j = r_{r_{j-1}}r_{r_j})}$$

where $k \in \{1, 2\}$. In the following, we show an example. If we receive $r_{r_{j-1}}r_{r_j} = TC$, then

$$P1 = \Pr(b^1_j = 0|y_{j-1}y_j = TC) = \Pr(x_{j-1}x_j|y_{j-1}y_j = TC)$$

where $x_{j-1}x_j = CT, CA, AG, AT, TC, TG, GA, GC$, all neighboring pairs produce transition symbols either '1' or '2'. Meanwhile,

$$P2 = \Pr(b^1_j = 1|y_{j-1}y_j = TC) = \Pr(x_{j-1}x_j|y_{j-1}y_j = TC)$$

where $x_{j-1}x_j = CC, CG, AA, AC, A, GG, GT$, all neighboring pairs produce transition symbols either '0' or '3'. Similarly, we have,

$$P3 = \Pr(b^2_j = 0|y_{j-1}y_j = TC) = \Pr(x_{j-1}x_j|y_{j-1}y_j = TC)$$

where $x_{j-1}x_j = CG, CA, AC, AT, TA, TG, GT, GC$, all neighboring pairs produce transition symbols either '1' or '3'. And

$$P4 = \Pr(b^2_j = 1|y_{j-1}y_j = TC) = \Pr(x_{j-1}x_j|y_{j-1}y_j = TC)$$

where $x_{j-1}x_j = CC, CT, AA, AG, TT, TC, GG, GA, CG, AC, TA, GT$, all neighboring pairs produce transition symbols from {'0', '2', '3'}. Therefore, we have,

$$\begin{aligned}
P1 =\ & \Pr(C|T) \cdot (\Pr(T|C) + \Pr(A|C)) + \Pr(A|T) \\
& \cdot (\Pr(G|C) + \Pr(T|C)) + \Pr(T|T) \cdot (\Pr(C|C) \\
& + \Pr(G|C)) + \Pr(G|T) \cdot (\Pr(A|C) + \Pr(C|C)) \\
P2 =\ & \Pr(C|T) \cdot (\Pr(C|C) + \Pr(G|C)) + \Pr(A|T) \\
& \cdot (\Pr(A|C) + \Pr(C|C)) + \Pr(T|T) \cdot (\Pr(T|C) \\
& + \Pr(A|C)) + \Pr(G|T) \cdot (\Pr(G|C) + \Pr(T|C)) \\
P3 =\ & \Pr(C|T) \cdot (p_w \cdot \Pr(G|C) + \Pr(A|C)) + \Pr(A|T) \\
& \cdot (p_w \cdot \Pr(C|C) + \Pr(T|C)) + \Pr(T|T) \cdot (p_w \\
& \cdot \Pr(A|C) + \Pr(G|C)) + \Pr(G|T) \cdot (p_w \cdot \Pr(T|C) \\
& + \Pr(C|C)) \\
P4 =\ & \Pr(C|T) \cdot (\Pr(C|C) + \Pr(T|C) + (1 - p_w) \\
& \cdot \Pr(G|C)) + \Pr(A|T) \cdot (\Pr(A|C) + \Pr(G|C) \\
& + (1 - p_w) \cdot \Pr(C|C)) + \Pr(T|T) \cdot (\Pr(T|C) \\
& + \Pr(C|C) + (1 - p_w) \cdot \Pr(A|C)) + \Pr(G|T) \\
& \cdot (\Pr(G|C) + \Pr(A|C) + (1 - p_w) \cdot \Pr(T|C))
\end{aligned}$$

Note that p_w is the probability of transition symbol '3' mapped to binary bits '10', calculating by Eq. (6). As a result, we obtain,

$$\mathcal{L}^0_{j^1} = \log \frac{P1}{P2};\ \mathcal{L}^0_{j^2} = \log \frac{P3}{P4}$$

The LLRs of other combinations of two neighbouring nucleotides can be estimated in similar way.

V. OPTIMIZATION OF PROTOGRAPH LDPC CODES FOR THE ASYMMETRIC DNA DATA STORAGE CHANNEL

A. MODIFIED PROTOGRAPH EXIT ALGORITHM

The EXIT algorithm is a theoretical tool to trace the convergence of the iterative decoders in the communication system. As the conventional EXIT algorithm [31] is not applicable to protograph codes, the modified EXIT algorithm for protogaph LDPC codes (PEXIT) over the AWGN channel has been proposed in [32]. However, as discussed in Section III, the nucleotides in the proposed DNA data storage channel models follow an asymmetric distribution and the LLRs of corresponding binary bits from the channel are a few of discrete values, which don't satisfy the symmetric Gaussian distribution assumption in [31], [32]. Thereby, we can not directly analyze the performance of protograph LDPC codes with PEXIT in [32]. In this section, a modified PEXIT algorithm is introduced, which can provide the error performance prediction of a given protograph base matrix over the asymmetric DNA data storage channel. The details are presented with some definitions given first.

$I_{EV}(i, j)$: the extrinsic MI between the message sent by VN_j to CN_i and the associative codeword bit;

$I_{EC}(i, j)$: the extrinsic MI between the message sent by CN_i to VN_j and the associative codeword bit;

$I_{AV}(i, j)$: the a priori MI between the message sent by VN_j to CN_i and the associative codeword bit;

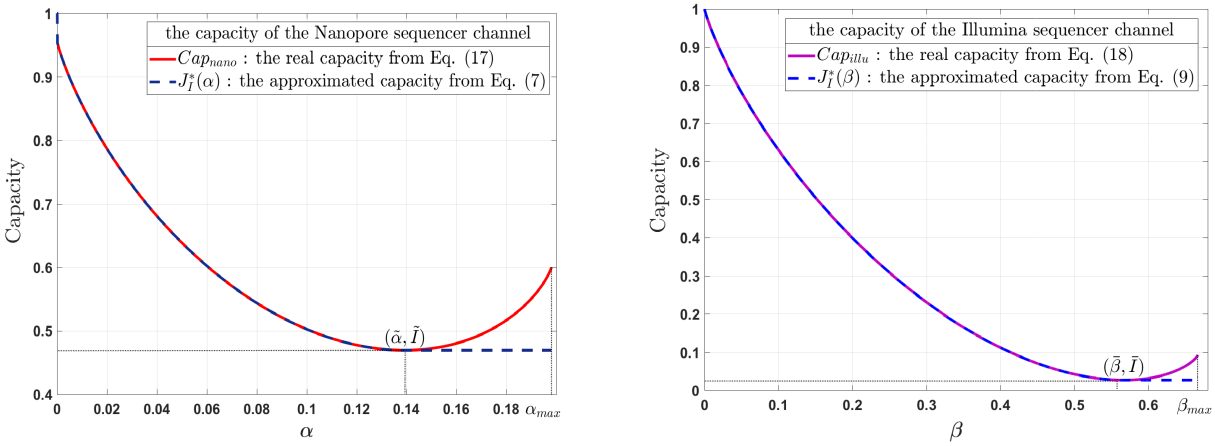


FIGURE 5. Real and approximated capacities: Left figure refers to the Nanopore sequencer channel, and right figure refers to the Illumina sequencer channel.

$I_{AC}(i, j)$: the a priori MI between the message sent by CN_i to VN_j and the associative codeword bit;

$I_{app}(j)$: the a posteriori MI between a posteriori LLR evaluated by VN_j and the associative codeword bit.

$N_V(j)$: the product of the non-zero entries in $B(:, j)$;

$N_C(i)$: the product of the non-zero entries in $B(i, :)$.

We define $J_N^*(\alpha)$ and $J_I^*(\beta)$ as the approximated channel capacities which are achieved by the least square curve fitting of the channel capacities Cap_{nano} and Cap_{illu} (see Eq. (17) and Eq. (18) in the Appendix) with the assumption of equal $q1$, $q2$, $q3$, and $q4$, in addition with some simplifications to keep the monotonicity (see Fig. 5). They are shown as follows, along with the corresponding inverse functions of $J_N^{*-1}(I)$ and $J_I^{*-1}(I)$.

$$J_N^*(\alpha) = \begin{cases} 1 & 0 \le \alpha < 0.0001 \\ 0.5327e^{(-17.3071\alpha)} + 0.4115 & 0.0001 \le \alpha < \tilde{\alpha} \\ \tilde{I} & \tilde{\alpha} \le \alpha \le \alpha_{max} \end{cases} \tag{7}$$

$$J_N^{*-1}(I) = \begin{cases} -0.1284I + 0.198 & 0 \le I < \tilde{I} \\ -\dfrac{log(1.8772I - 0.7725)}{17.3071} & \tilde{I} \le I < 0.9433 \\ 0 & 0.9433 \le I \le 1 \end{cases} \tag{8}$$

$$J_I^*(\beta) = \begin{cases} 1 & 0 \le \beta < 0.0001 \\ 1.1029e^{(-3.7788\beta)} - 0.1241 & 0.0001 \le \beta < \bar{\beta} \\ \bar{I} & \bar{\beta} \le \beta \le \beta_{max} \end{cases} \tag{9}$$

$$J_I^{*-1}(I) = \begin{cases} -18.5I + 0.6667 & 0 \le I < \bar{I} \\ -\dfrac{log(0.9067I + 0.1125)}{3.7788} & \bar{I} \le I < 0.9784 \\ 0 & 0.9784 \le I \le 1 \end{cases} \tag{10}$$

where $\tilde{I} = 0.4596$, which is the minimum value of Cap_{nano} with corresponding $\tilde{\alpha} = 0.1390$, and $\alpha_{max} = 0.198$ is the maximum allowable value of α. As to the Illumina sequencer channel, $\bar{I} = 0.0054$ is the minimum value of Cap_{illu} with corresponding $\bar{\beta} = 0.5668$, and $\beta_{max} = 2/3$ is the maximum allowable value of β.

The non-monotonicity of the channel capacity is resulted by the asymmetry of mutation probabilities. When $\alpha > \tilde{\alpha}$ ($\beta > \bar{\beta}$), the values of Cap_{nano} (Cap_{illu}) do not decrease monotonously but with a certain increase, which would cause the non-reversibility of $J_N^{*-1}(I)$ ($J_I^{*-1}(I)$). Therefore, we set $J_N^*(\alpha) = \tilde{I}$ with $\alpha > \tilde{\alpha}$, while $J_N^{*-1}(I)$ decreases linearly with $0 \le I < \tilde{I}$ during the iterative MI update in the proposed PEXIT; and the settings of $J_I^*(\beta)$ and $J_I^{*-1}(I)$ follow the similar way. Such approximation would result in a certain loss of accuracy for the error performance prediction, especially for the Nanopore sequencer channel due to the significant asymmetry. Therefore, the proposed PEXIT actually provides a lower bound of the maximum α and β, for which the protograph LDPC codes can correct all the asymmetric substitution errors in the sequencing process with the asymptotic code length. We name them as the decoding thresholds of the protograph LDPC codes over the asymmetric Nanopore sequencer and Illumina sequencer channels, denoted as $\underline{\alpha_{th}}$ and $\underline{\beta_{th}}$, respectively.

Taking the Illumina sequencer channel as an example, the proposed PEXIT algorithm for asymmetric DNA data storage channel is described as follows. The analysis of the Nanopore sequencer channel follows the same procedure with relevant $J_N^*(\alpha)$ and $J_N^{*-1}(I)$.

1) INITIATION

Given the base matrix $\mathbf{B} = (b_{i,j})$ with size of $M \times N$, and the channel parameter β, initiate the a priori MI from channel $I_{ch}(j)$ for each $VN_j, j = 1, 2, \ldots, N$:

$$I_{ch}(j) = J_I^*(\beta) \tag{11}$$

noticed that $I_{ch}(j) = 0$ if VN_j is punctured.

In addition, an indicator function is defined as

$$\phi(b_{i,j}) = \begin{cases} 1 & \text{if } b(i,j) \neq 0 \\ 0 & \text{otherwise} \end{cases} \tag{12}$$

2) THE MI UPDATE FROM VNs TO CNs

For $j = 1, 2, \ldots, N$, $i = 1, 2, \ldots, M$,

$$\begin{aligned} I_{EV}(i,j) = \phi(b_{i,j}) \cdot J_I^*\Big(J_I^{*-1}(I_{ch}(j)) \\ \cdot \frac{1}{N_V(j)} \prod_{s \neq i} b(s,j) J_I^{*-1}(I_{AV}(s,j)) \\ \cdot \frac{1}{N_V(j)} (b(i,j) - 1) J_I^{*-1}(I_{AV}(i,j))\Big), \end{aligned} \tag{13}$$

then set $I_{AC}(i,j) = I_{EV}(i,j)$.

3) THE MI UPDATE FROM CNs TO VNs

For $j = 1, 2, \ldots, N$, $i = 1, 2, \ldots, M$,

$$\begin{aligned} I_{EC}(i,j) = \phi(b_{i,j}) \\ \cdot \Bigg(1 - J_I^*\Big(\frac{1}{N_C(i)} \prod_{s \neq j} b(i,s) J_I^{*-1}(1 - I_{AC}(i,s)) \\ \cdot \frac{1}{N_C(i)} (b(i,j) - 1) J_I^{*-1}(1 - I_{AC}(i,j))\Big)\Bigg), \end{aligned} \tag{14}$$

then set $I_{AV}(i,j) = I_{EC}(i,j)$.

4) THE APP-LLR MI EVALUATION

For $j = 1, 2, \ldots, N$,

$$\begin{aligned} I_{APP}(j) = \phi(b_{i,j}) \cdot J_I^*\Big(J_I^{*-1}(I_{ch}(j)) \\ \cdot \frac{1}{N_V(j)} \prod b(i,j) J_I^{*-1}(I_{AV}(i,j))\Big), \end{aligned} \tag{15}$$

5) ITERATE UNTIL $I_{APP}(j) = 1, \forall j$

Table 3 gives the decoding thresholds of AR4JA codes over two channel models; and the gaps to the channel capacities are also provided. Noted that the capacities of two channel models in Table 3 are calculated from Eq. (17) and Eq. (18), respectively. As can be seen in Table 3 and the simulation results in Section VI, the performance of AR4JA codes designed for the AWGN channel are not very satisfying over the asymmetric DNA data storage channel. The gaps between the decoding thresholds and the corresponding capacity limits are relatively larger in view of the DNA data storage channel with quite small error rate. Thus the code optimization is processed based on the proposed PEXIT algorithm. Considering the practical error probabilities of the Nanopore sequencing (α from 0.03 to 0.04 [15]) and the Illumina sequencing (β from 0.5×10^{-3} to 1.5×10^{-3} [12]), the 1/2 coding rate is proper for the Nanopore sequencer channel, while the 5/6 coding rate or higher would be suitable for the Illumina sequencer channel.

TABLE 3. The decoding thresholds of AR4JA codes over the Nanopore sequencer channel ($\underline{\alpha_{th}}$) and the Illumina sequencer channel ($\underline{\beta_{th}}$).

Rate	Nanopore sequencer channel			Illumina sequencer channel		
	$\underline{\alpha_{th}}$	Capacity	Gap	$\underline{\beta_{th}}$	Capacity	Gap
1/2	0.0111	0.1038	0.0927	0.0016	0.1507	0.1491
2/3	3.283e-4	0.0425	0.0421	5.008e-4	0.0869	0.0863
3/4	1.445e-4	0.0262	0.0260	3.733e-4	0.0596	0.0592
4/5	1.179e-4	0.0179	0.0172	3.522e-4	0.0449	0.0445
5/6	1.112e-4	0.0131	0.0129	1.869e-4	0.0357	0.0355
6/7	1.093e-4	0.0099	0.0097	1.341e-4	0.0295	0.0293
7/8	1.087e-4	0.0077	0.0075	1.000e-4	0.0250	0.0249

TABLE 4. The decoding thresholds of rate 1/2 modified AR4JA codes over the Nanopore sequencer channel ($\underline{\alpha_{th}}$), in addition with the gaps to the channel capacities and the gains over the rate 1/2 AR4JA codes.

Codes	$\mathbf{B}_{I1}$	$\mathbf{B}_{I2}$	$\mathbf{B}_{I3}$	$\mathbf{B}_{I4}$	$\mathbf{B}_{I5}$	$\mathbf{B}_{I6}$	$\mathbf{B}_{I7}$	$\mathbf{B}_{I8}$
$\underline{\alpha_{th}}$	0.0668	0.0976	0.1027	0.0484	0.0870	0.0196	0.0749	0.0949
Gap	0.0370	0.0062	0.0011	0.0554	0.0168	0.0842	0.0289	0.0089
Gain	0.0557	0.0865	0.0916	0.0373	0.0759	0.0085	0.0638	0.0838

B. CODE OPTIMIZATION FOR THE NANOPORE SEQUENCER CHANNEL

Taking the AR4JA codes as references, the rate 1/2 AR4JA codes are optimized for better error performance. From the observations of PEXIT analyses, the decoding threshold of AR4JA codes can be remarkably improved by increasing the proportion of the VNs with degree-2 or the precoded VNs, although it is contrary to the design principles for the AWGN channel. In order to analyze the influence of the degree-2 VNs and precoded VNs, the following modifications of AR4JA codes are proposed with the degree-1 VN and the punctured VN of the highest degree retained. For the simplicity of analysis, all the increased precoded VNs have the same degree of three without multiple edges.

$$\mathbf{B}_{I1} = \begin{pmatrix} 1 & 2 & 0 & 1 & 0 \\ 0 & 3 & 1 & 1 & 1 \\ 0 & 1 & 2 & 1 & 1 \end{pmatrix} \quad \mathbf{B}_{I2} = \begin{pmatrix} 1 & 2 & 1 & 1 & 0 \\ 0 & 3 & 1 & 1 & 1 \\ 0 & 1 & 1 & 1 & 1 \end{pmatrix}$$

$$\mathbf{B}_{I3} = \begin{pmatrix} 1 & 2 & 1 & 0 & 0 \\ 0 & 3 & 1 & 1 & 1 \\ 0 & 1 & 1 & 1 & 1 \end{pmatrix} \quad \mathbf{B}_{I4} = \begin{pmatrix} 1 & 2 & 0 & 0 & 0 \\ 0 & 3 & 1 & 1 & 1 \\ 0 & 1 & 2 & 1 & 1 \end{pmatrix}$$

$$\mathbf{B}_{I5} = \begin{pmatrix} 1 & 2 & 0 & 0 & 0 \\ 0 & 3 & 1 & 1 & 1 \\ 0 & 1 & 1 & 1 & 1 \end{pmatrix} \quad \mathbf{B}_{I6} = \begin{pmatrix} 1 & 2 & 0 & 0 & 1 \\ 0 & 3 & 1 & 1 & 1 \\ 0 & 1 & 2 & 2 & 1 \end{pmatrix}$$

$$\mathbf{B}_{I7} = \begin{pmatrix} 1 & 2 & 0 & 1 & 1 \\ 0 & 3 & 1 & 1 & 1 \\ 0 & 1 & 2 & 1 & 1 \end{pmatrix} \quad \mathbf{B}_{I8} = \begin{pmatrix} 1 & 2 & 1 & 1 & 1 \\ 1 & 3 & 1 & 1 & 1 \\ 0 & 1 & 1 & 1 & 1 \end{pmatrix}$$

As can be seen from Table 4, all the modifications with different proportions of degree-2 VNs or precoded VNs can achieve gains on the decoding threshold over AR4JA codes from 0.0085 to 0.0916, and the gaps to the capacity limits are shrunken accordingly. However, not all the modified codes

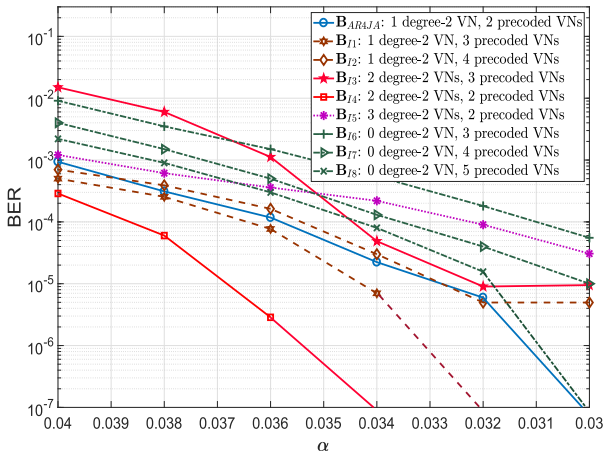


FIGURE 6. The BER performance of rate 1/2 modified AR4JA codes over the Nanopore sequencer channel.

perform well over the channel. Fig. 6 shows the bit error rate (BER) of the eight modified AR4JA codes over the Nanopore sequencer channel. Compared with AR4JA codes, for $\mathbf{B}_{I1}$ and $\mathbf{B}_{I2}$ with one degree-2 VN, both of them have better or similar waterfall region, while $\mathbf{B}_{I2}$ with one more precoded VNs suffers from higher error floor in low α region. As to $\mathbf{B}_{I3}$ and $\mathbf{B}_{I4}$ with two degree-2 VNs, $\mathbf{B}_{I4}$ with two precoded VNs has better error fall region performance than that of $\mathbf{B}_{I1}$, while $\mathbf{B}_{I3}$ with one more precoded VNs suffers high error floor although with highest decoding threshold. Similarly, $\mathbf{B}_{I5}$ with three degree-2 VNs performs even worse which almost has no obvious waterfall region as α decreases. $\mathbf{B}_{I6}$, $\mathbf{B}_{I7}$, and $\mathbf{B}_{I8}$ are modified versions with no degree-2 VNs but increasing precoded VNs. The waterfall region performance can also be improved by increasing the precoded VNs, however the improvements are not significant and higher error rates occur in the high α region.

As a conclusion, both increasing the degree-2 VNs and the precoded VNs can improve the error performance on the waterfall region. However, too many these VNs would result in a high error floor. An appropriate proportion of degree-2 VNs and precoded VNs is the key issue to achieve good performance in both aspects. Among the eight modified AR4JA codes, $\mathbf{B}_{I4}$ with two degree-2 VNs and two precoded VNs are suitable for the Nanopore sequencer channel, which achieves 0.0373 coding gains over AR4JA codes and with a smaller gap of 0.0554 to the capacity limit. Better protographs would be achieved as the increase of the base matrix size due to the large searching space at the cost of higher complexity. Considering the relatively higher error rate of Nanopore sequencer channel, the codes with rates higher than 1/2 are not considered in this paper.

C. CODE OPTIMIZATION FOR THE ILLUMINA SEQUENCER CHANNEL

Considering the larger size base matrices, the genetic algorithm (GA) [33] is processed to search the optimal higher rate codes over the Illumina sequencer channel. The PEXIT

TABLE 5. The decoding thresholds of the modified AR4JA codes with different numbers of precoded VNs and fixed two degree-2 VNs for the Illumina sequencer channel (β_{th}).

Precoded VNs	Rate 5/6	Rate 4/5	Rate 3/4	Rate 2/3	Rate 1/2
2	3.5190e-4	3.6350e-4	4.0816e-4	5.9487e-4	0.0017
3	3.5580e-4	3.7830e-4	4.6760e-4	8.8701e-4	**0.0071**
4	3.6350e-4	4.0810e-4	5.9480e-4	**0.0017**	—
5	3.7830e-4	4.6760e-4	**8.8700e-4**	0.0016	—
6	4.0810e-4	**5.8960e-4**	7.0970e-4	—	—
7	**4.6370e-4**	4.9280e-4	5.9990e-4	—	—
8	4.1400e-4	4.4000e-4	—	—	—
9	3.8790e-4	4.1390e-4	—	—	—
10	3.7340e-4	—	—	—	—
11	3.6620e-4	—	—	—	—

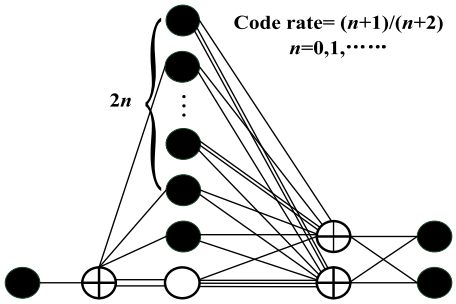


FIGURE 7. The protographs of proposed codes with rates 1/2 and higher for the Illumina sequencer channel.

analyses of the searched codes exhibit the regularity that a certain proportion of precoded VNs should be added to construct good protographs for higher rate codes. Table 5 shows the decoding thresholds of modified AR4JA codes with different proportions of precoding VNs and fixed two degree-2 VNs. To simplify the analysis, the other unprecoded VNs are set with the same degree of three. As seen in Table 5, the proper numbers of the precoded VNs for 5/6, 4/5, 3/4, 2/3 and 1/2 rate codes are 7, 6, 5, 4 and 3, respectively, which can achieve the highest decoding thresholds. Accordingly, the family protographs and the corresponding base matrix $\mathbf{B}_{Illu}$ for the Illumina sequencer channel can be given as in Fig. 7 and Eq. (16).

$$\mathbf{B}_{Illu}=\begin{pmatrix} 1\,2\,0\,0\,1 & \overbrace{0\,1\,\cdots\,0\,1}^{2n} \\ 0\,3\,1\,1\,1 & 1\,1\,\cdots\,2\,1 \\ 0\,1\,1\,1\,1 & 2\,1\,\cdots\,1\,1 \end{pmatrix} \tag{16}$$

VI. NUMERICAL RESULTS AND DISCUSSION

According to the practical DNA strand lengths which can be effectively processed by two sequencing techniques (i.e.,$\sim$ 1000 base pair in the Nanopore sequencing [11] and $\sim$200nt in the Illumina sequencing [12]), we fix the lengths of original message block $\boldsymbol{m}$ as 1000 bits and 300 bits for

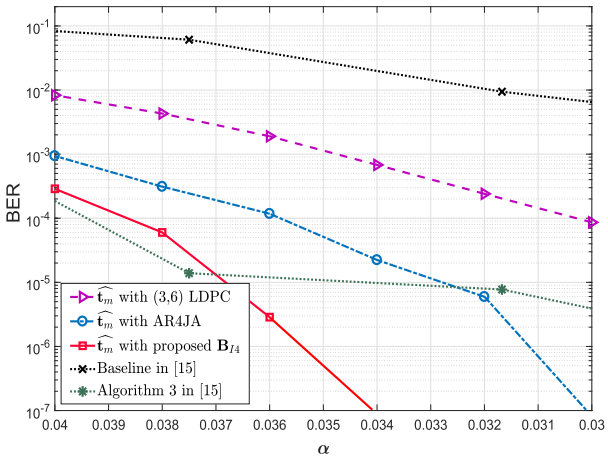


FIGURE 8. The BER performance of rate 1/2 LDPC codes over the Nanopore sequencer channel with the original message block length of 1000 bits, where $\widehat{\boldsymbol{t}_m}$ indicates the binary output from the LDPC decoder in Fig. 4.

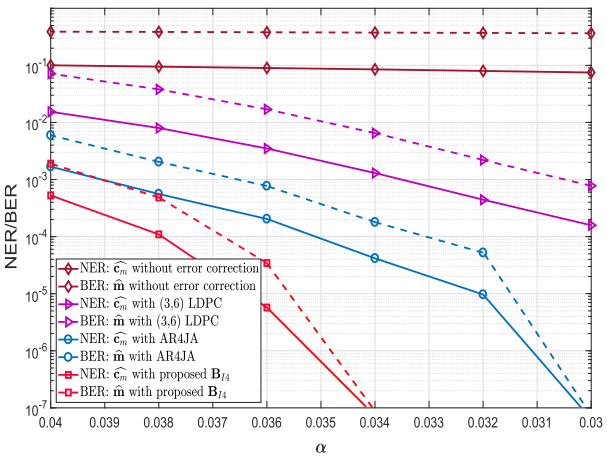


FIGURE 9. The NER performance of rate 1/2 LDPC codes and the BER performance of the hybrid system over the Nanopore sequencer channel, where $\widehat{\boldsymbol{c}_m}$ is the output of DNA message from interim de-mapping, and $\widehat{\boldsymbol{m}}$ indicates the binary output from the constrained decoder in Fig. 4.

the Nanopore sequencer channel and the Illumina sequencer channel, respectively. Considering the different error probabilities of the Nanopore sequencing ($\sim\ 10^{-2}$) and the Illumina sequencing ($\sim\ 10^{-3}$) [12], the coding rates of $\mathcal{R}\ =\ 1/2$ and $\mathcal{R}\ =\ 5/6$ are adopted correspondingly. However, in the Illumina case, after the VL-RLL encoding and interim mapping, the lengths of the data that need to be encoded ($\boldsymbol{t_m}$) change with continuous even values from 300 bits to 318 bits, which can not all be matched by the fixed size of rate 5/6 basematrix (i.e., $3\ \times\ 13$). Therefore, a few rows and columns of the matched parity-check matrices are punctured to generate some approximately 5/6 rate parity-check matrices for certain block lengths. For instance, the approximate 5/6 rate parity-check matrix for the encoding data with 308 bits block length can be obtained by puncturing the last three columns and last one row of the parity-check matrix with the size of 93×403 corresponding to the encoding data with 310 bits block length. In the simulation, we use practical channel parameters, i.e., α from 0.03 to 0.04 [15] and β from $0.5\ \times\ 10^{-3}$ to $1.5\ \times\ 10^{-3}$ [12]. The maximum iteration of SPA decoder is set as 100; and the frame numbers are set as 10^4 and 10^5 for the message block lengths with 1000 bits and 300 bits, respectively.

The BER performances of the rate 1/2 LDPC codes over the Nanopore sequencer channel and the rate 5/6 LDPC codes over the Illumina sequencer channel are shown in Fig. 8 and Fig. 10, respectively. The NER (nucleotide error rate) performance of the rate 1/2 LDPC codes and the BER performance of the whole hybrid coding system over the Nanopore sequencer channel are shown in Fig. 9; followed with Fig. 11 where the NER performance of the rate 5/6 LDPC codes and the BER performance of the whole hybrid coding system over the Illumina sequencer channel are presented.

A. COMPARISON WITH DIFFERENT LDPC CODES

The error performances of the proposed codes are compared with AR4JA codes and (3,6) LDPC codes with fixed degree-3

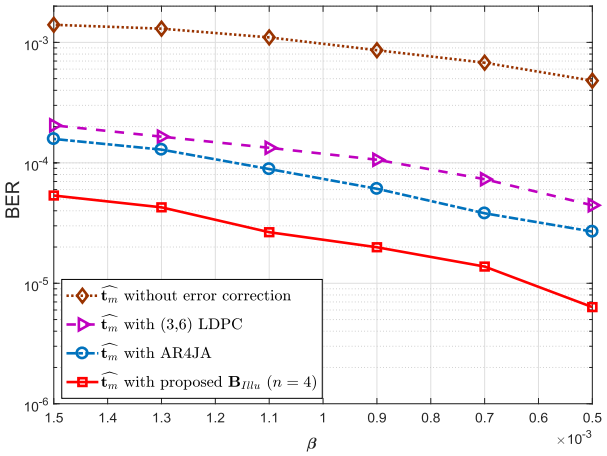


FIGURE 10. The BER performance of rate 5/6 LDPC codes over the Illumina sequencer channel with the original message block length of 300 bits, where $\widehat{\boldsymbol{t}_m}$ indicates the binary output from the LDPC decoder in Fig. 4.

VNs and degree-6 CNs. As shown in both Fig. 8 and Fig. 10, better error performance can be observed in the BER curves of the proposed codes. For the Nanopore sequencer channel, the proposed $\mathbf{B}_{I4}$ achieves higher decoding threshold than AR4JA codes about 0.0373 (see Table 4), and enables lower BER than 10^{-7} at larger α region (as shown in Fig. 8). For the Illumina sequencer channel in Fig. 10, the BER curves of the compared rate 5/6 codes decrease slowly in the practical error region, where the sharp waterfall regions could not be observed. Although the proposed codes (rate 5/6 $\mathbf{B}_{Illu}$ with $n\ =\ 4$) have significant coding gains over the (3,6) LDPC codes and AR4JA codes, the preponderance is not as prominent as that in the Nanopore sequencer channel. The possible reasons are as follows. First, the practical error rates of the Illumina sequencer channel are relatively smaller and the asymmetry of the mutation probabilities is less significant compared with the Nanopore sequencer case. Second,

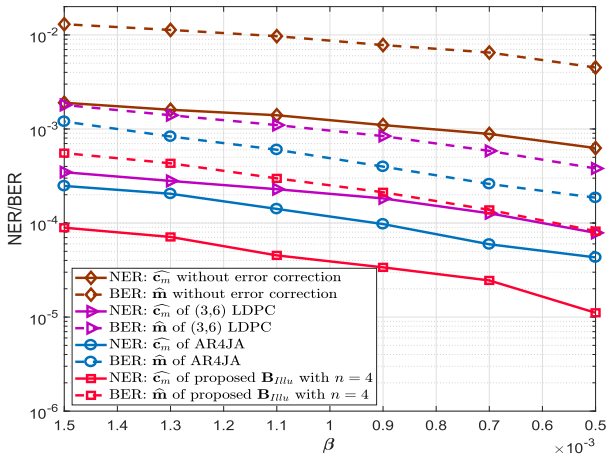


FIGURE 11. The NER performance of rate 5/6 LDPC codes and the BER performance of the hybrid system over the Illumina sequencer channel, where $\widehat{c_m}$ is the output of DNA message from the interim de-mapping, and $\hat{m}$ indicates the binary output from the constrained decoder in Fig. 4.

the length of the message block in the Illumina case is much shorter than the Nanopore case as the Illumina sequencing could only efficiently sequence DNA strands with lengths around 200nt, so that the finite length effect is relatively worse than the Nanopore case where much longer DNA strands (i.e., 1000nt) are accepted. Note that better error performance could be achieved for the proposed $\mathbf{B}_{Illu}$ with lower rates.

In Fig. 9 and Fig. 11, the solid brown curves with diamond markers of $\widehat{c_m}$ show the NER of information blocks from the channel without error correction, which actually refer to the received information block $\widehat{r_m}$ in Fig. 4. The other solid curves show the NER of the information blocks after LDPC decoding and interim de-mapping while before constrained decoding (c_m against $\widehat{c_m}$ in Fig. 4). Noted that, the trends of the NER curves of $\widehat{c_m}$ are consistent with the relevant BER curves of $\widehat{t_m}$ in Fig. 8 and Fig. 10. Since one nucleotide error may correspond to one or two bits errors, the NER of $\widehat{c_m}$ is a little higher than the BER of $\widehat{t_m}$. On the other side, compared with the BER of $\widehat{t_m}$, a certain increase of the error rates can be observed in the BER curves of $\widehat{m}$, which is caused by the constrained decoding process. As discussed, one nucleotide in error might lead to two transition symbols in error when the reverse of differential operation is performed. Furthermore, an erroneous transition symbol might lead to severe error propagation in the subsequent bits in the reverse of mapping based on Table 1. In addition, when the BER of $\widehat{t_m}$ is approaching 0, the NER of $\widehat{c_m}$ and the BER of $\widehat{m}$ are also approaching 0, which indicates that all errors are corrected by the LDPC decoder at the near-nucleotide-level, reducing the error propagation in the constrained decoding.

B. COMPARISON WITH [15]

The authors in [15] simply map 2-source bits to 1 nucleotide symbol and hence the resultant DNA sequences have 2 bits/nt mapping potential. However, the resultant sequence might have long homopolymer runs which increase the sequencing errors [13], [14]. In contrast, we propose a more practical coding scheme with efficient decoding for error resilience in DNA data storage, where the resultant DNA sequences satisfy the biochemical constraint, potentially suppressing error occurrence in DNA sequencing. In addition, although the constrained code is considered in this work with an expectation of reduction of mapping potential, the proposed modified VL-RLL constrained code offers a very high mapping potential. According to (5), for the Nanopore case where $\mathcal{R} = 1/2$, the mapping potential becomes $\sim$ 1.988 bits/nt; and for the Illumina case where $\mathcal{R} = 5/6$, the mapping potential becomes $\sim$ 1.980 bits/nt, presenting only 1% gap from the upper boundary 2 bits/nt.

We also compare the error performance of the proposed codes with that of the error correction codes adopted in [15], which are shown in Fig. 8. In [15], the authors focus on the design of the LDPC decoder, in which two binary LDPC codes are used to present one DNA strand, and the decoding is processed by two SPAs exchanging the side information with different modes. As the constrained codes are not considered in [15], only the performances of the LDPC encoders and decoders are compared rather than the whole system under the same channel condition of the overall substitution probability (i.e., 14λ in [15] and $12\alpha + 0.04$ in this work). After normalizing the parameter λ to the adopted α, the dotted black line with cross markers and the dotted green line with star markers in Fig. 8 represent the BER performances of the baseline decoder and the Algorithm 3 reported in [15], respectively. Notice that the LDPC codes with approximate (3,6) degree distribution are adopted in [15], and the baseline decoder runs two SPAs without side information. Therefore, the black dotted line with cross markers and the purple dashed line with triangle markers can be considered as the BER performances of the (3,6) LDPC codes with two types of decoders, i.e, the two parallel independent SPAs and one single SPA with double block length, respectively. The tradeoff exists between the decoding complexity and the error performance. The purple dashed line has better error performance but higher decoding complexity (at least twice running time of the baseline decoder). On the other side, compared with the approximate (3,6) LDPC codes with the improved parallel SPAs (Algorithm 3) in [15] (the green dotted line with star markers), the proposed $\mathbf{B}_{I4}$ codes with the traditional SPA (the red solid line with rectangle markers) have similar error performance in the large α region, while can exhibit sharper waterfall region and achieve lower BER than 10^{-7} in worse channel condition ($\alpha = 0.034$).

VII. CONCLUSION

We have introduced a hybrid coding architecture, which can correct the asymmetric substitution errors in the DNA sequencing processes while satisfying the biochemical constraint. The modified VL-RLL codes have been developed to limit the homopolymer runs, while achieving near limit mapping potential ($\sim$ 1.98 bits/nt). Furthermore, according

to the characteristics of the asymmetric DNA data storage channel, we have proposed a modified PEXIT algorithm, and optimized series of protograph LDPC codes accordingly for both Nanopore sequencer channel and Illumina sequencer channel. The simulation results indicate that the proposed hybrid coding scheme can tackle the asymmetric substitution errors that occur in the sequencing process, and the optimized codes can exhibit better error performance over the traditional protograph LDPC codes and the codes adopted in the existing DNA data storage system. In the future, a more generalized channel model including the effect of homopolymer runs would be developed. Besides, the PEXIT algorithm would be further improved for higher estimation accuracy with the consideration about the finite-length effect.

APPENDIX
CAPACITY OF THE ASYMMETRIC DNA DATA STORAGE CHANNEL

The capacities of the asymmetric Nanopore sequencer channel and the Illumina sequencer channel are analyzed with the definitions of q_1, q_2, q_3, and q_4 as the probabilities of the stored symbols 'A', 'G', 'C', and 'T', respectively, in addition with X and Y as the random variables of the input and output of the channel. Noticed that all the '*log*' operations in Eq. (17) and Eq. (18) mean '*log*' base 4 for the capacities of the corresponding binary channels.

A. CAPACITY OF THE NANOPORE SEQUENCER CHANNEL

$$
\begin{aligned}
&Cap_{nano}\\
&= \max_{q_1,q_2,q_3,q_4} I(X;Y)\\
&= \max_{q_1,q_2,q_3,q_4} (H(Y)-H(Y|X))\\
&= \max_{q_1,q_2,q_3,q_4} \Big(-(q_1(1-p_2-p_3-p_4)+q_2p_4+q_4p_2+q_3p_3)\\
&\quad \cdot \log(q_1(1-p_2-p_3-p_4)+q_2p_4+q_4p_2+q_3p_3)\\
&\quad -(q_2(1-p_2-p_3-p_4)+q_1p_4+q_4p_3+q_3p_2)\\
&\quad \cdot \log(q_2(1-p_2-p_3-p_4)+q_1p_4+q_4p_3+q_3p_2)\\
&\quad -(q_3(1-p_2-p_1-p_3)+q_4p_1+q_1p_3+q_2p_2)\\
&\quad \cdot \log(q_3(1-p_2-p_1-p_3)+q_4p_1+q_1p_3+q_2p_2)\\
&\quad -(q_4(1-p_2-p_1-p_3)+q_3p_1+q_1p_2+q_2p_3)\\
&\quad \cdot \log(q_4(1-p_2-p_1-p_3)+q_3p_1+q_1p_1+q_2p_3)\\
&\quad +q_1((1-p_2-p_3-p_4)\log(1-p_2-p_3-p_4)\\
&\quad +p_2\log(p_2)+p_3\log(p_3)+p_4\log(p_4))\\
&\quad +q_2((1-p_2-p_3-p_4)\log(1-p_2-p_3-p_4)\\
&\quad +p_2\log(p_2)+p_3\log(p_3)+p_4\log(p_4))\\
&\quad +q_3((1-p_2-p_1-p_3)\log(1-p_2-p_1-p_3)\\
&\quad +p_2\log(p_2)+p_3\log(p_3)+p_1\log(p_1))\\
&\quad +q_4((1-p_2-p_1-p_3)\log(1-p_2-p_1-p_3)\\
&\quad +p_2\log(p_2)+p_3\log(p_3)+p_1\log(p_1))\Big) \qquad (17)
\end{aligned}
$$

B. CAPACITY OF THE ILLUMINA SEQUENCER CHANNEL

$$
\begin{aligned}
&Cap_{illu}\\
&= \max_{q_1,q_2,q_3,q_4} I(X;Y)\\
&= \max_{q_1,q_2,q_3,q_4} (H(Y)-H(Y|X))\\
&= \max_{q_1,q_2,q_3,q_4} \Big(-(q_1(1-p_b)+q_2\frac{p_a}{3}+q_3\frac{p_b}{3}+q_4\frac{p_a}{3})\\
&\quad \cdot \log(q_1(1-p_b)+q_2\frac{p_a}{3}+q_3\frac{p_b}{3}+q_4\frac{p_a}{3})\\
&\quad -(q_2(1-p_a)+q_1\frac{p_b}{3}+q_3\frac{p_b}{3}+q_4\frac{p_a}{3})\\
&\quad \cdot \log(q_2(1-p_a)+q_1\frac{p_b}{3}+q_3\frac{p_b}{3}+q_4\frac{p_a}{3})\\
&\quad -(q_3(1-p_b)+q_1\frac{p_b}{3}+q_2\frac{p_a}{3}+q_4\frac{p_a}{3})\\
&\quad \cdot \log(q_3(1-p_b)+q_1\frac{p_b}{3}+q_2\frac{p_a}{3}+q_4\frac{p_a}{3})\\
&\quad -(q_4(1-p_a)+q_1\frac{p_b}{3}+q_2\frac{p_a}{3}+q_3\frac{p_b}{3})\\
&\quad \cdot \log(q_4(1-p_a)+q_1\frac{p_b}{3}+q_2\frac{p_a}{3}+q_3\frac{p_b}{3})\\
&\quad +q_1((1-p_b)\log(1-p_b)+\frac{p_b}{3}\log(\frac{p_b}{3})+\frac{2p_a}{3}\log(\frac{p_a}{3}))\\
&\quad +q_2((1-p_a)\log(1-p_a)+\frac{2p_b}{3}\log(\frac{p_b}{3})+\frac{p_a}{3}\log(\frac{p_a}{3}))\\
&\quad +q_3((1-p_b)\log(1-p_b)+\frac{p_b}{3}\log(\frac{p_b}{3})+\frac{2p_a}{3}\log(\frac{p_a}{3}))\\
&\quad +q_4((1-p_a)\log(1-p_a)+\frac{2p_b}{3}\log(\frac{p_b}{3})+\frac{p_a}{3}\log(\frac{p_a}{3}))\Big) \qquad (18)
\end{aligned}
$$

REFERENCES

[1] J. P. L. Cox, "Long-term data storage in DNA," *Trends Biotechnol.*, vol. 19, no. 7, pp. 247–250, 2001.

[2] Y. Erlich and D. Zielinski, "DNA Fountain enables a robust and efficient storage architecture," *Science*, vol. 355, no. 6328, pp. 950–954, 2017.

[3] M. Irving. (Aug. 3, 2017). *Sony and IBM Shatter Magnetic Tape Storage Density Record*. [Online]. Available: https://newatlas.com/sony-ibm-magnetic-tape-density-record/50743/

[4] C. Bancroft, T. Bowler, B. Bloom, and C. T. Clelland, "Long-term storage of information in DNA," *Science*, vol. 293, no. 5536, pp. 1763–1765, 2001.

[5] G. M. Church, Y. Gao, and S. Kosuri, "Next-generation digital information storage in DNA," *Science*, vol. 337, no. 6102, p. 1628, 2012.

[6] N. Goldman, P. Bertone, S. Chen, C. Dessimoz, E. M. LeProust, B. Sipos, and E. Birney, "Towards practical, high-capacity, low-maintenance information storage in synthesized DNA," *Nature*, vol. 494, pp. 77–80, Jan. 2013.

[7] R. N. Grass, R. Heckel, M. Puddu, D. Paunescu, and W. J. Stark, "Robust chemical preservation of digital information on DNA in silica with error-correcting codes," *Angew. Chem. Int. Ed.*, vol. 54, no. 8, pp. 2552–2555, 2015.

[8] S. M. H. T. Yazdi, Y. Yuan, J. Ma, H. Zhao, and O. Milenkovic, "A rewritable, random-access DNA-based storage system," *Sci. Rep.*, vol. 5, Sep. 2015, Art. no. 14138.

[9] M. Blawat, K. Gaedke, I. Huetter, X.-M. Chen, B. Turczyk, S. Inverso, B. W. Pruitt, and G. M. Church, "Forward error correction for DNA data storage," *Procedia Comput. Sci.*, vol. 80, no. 3, pp. 1011–1022, Feb. 2016.

[10] J. Bornholt, R. Lopez, D. M. Carmean, L. Ceze, G. Seelig, and K. Strauss, "A DNA-based archival storage system," *ACM SIGOPS Operat. Syst. Rev.*, vol. 50, no. 2, pp. 637–649, 2016.

[11] S. M. H. T. Yazdi, R. Gabrys, and O. Milenkovic, "Portable and error-free DNA-based data storage," *Sci. Rep.*, vol. 7, no. 1, 2017, Art. no. 5011.

[12] L. Organick, S. D. Ang, Y. J. Chen, R. Lopez, S. Yekhanin, K. Makarychev, M. Z. Racz, G. Kamath, P. Gopalan, B. Nguyen, and C. N. Takahashi, "Random access in large-scale DNA data storage," *Nature Biotechnol.*, vol. 36, no. 3, pp. 242–248, 2018.

[13] M. G. Ross, C. Russ, M. Costello, A. Hollinger, N. J. Lennon, R. Hegarty, C. Nusbaum and D. B. Jaffe, "Characterizing and measuring bias in sequence data," *Genome Biol.*, vol. 14, no. 5, p. R51, 2013.
[14] F. J. Rang, W. P. Kloosterman, and J. de Ridder, "From squiggle to basepair: Computational approaches for improving nanopore sequencing read accuracy," *Genome Biol.*, vol. 19, no. 1, p. 90, 2018.
[15] F. Peng and Z. Wang. (Mar. 2019). *LDPC Codes for Portable DNA Storage*. [Online]. Available: https: //faculty.sites.uci.edu/zhiying/files/2019/03/long-version.pdf
[16] W. Ryan and S. Lin, *Channel Codes: Classical and Modern*. Cambridge, U.K.: Cambridge Univ. Press, 2009.
[17] N. Marina, "LDPC codes for binary asymmetric channels," in *Proc. IEEE Conf. Telecommun. (ICT)*, Jun. 2008, pp. 1–7.
[18] R. Gabrys and L. Dolecek, "Coding for the binary asymmetric channel," in *Proc. IEEE Conf. Comput., Netw. Commun. (ICNC)*, Jan./Feb. 2012, pp. 461–465.
[19] J. Thorpe, "Low-density parity-check (LDPC) codes constructed from protograph," IPN Progr. Rep. 42-154, 2003.
[20] D. Divsalar, S. Dolinar, C. Jones, and K. Andrews, "Capacity approaching protograph codes," *IEEE J. Sel. Areas Commun.*, vol. 27, no. 6, pp. 876–888, Aug. 2009.
[21] Y. Fang, K.-K. Wong, L. Wang, and K.-F. Tong, "Performance analysis of protograph low-density parity-check codes for Nakagami-*m* fading relay channels," *IET Commun.*, vol. 7, no. 11, pp. 1133–1139, Jul. 2013.
[22] Y. Fang, G. Bi, and Y. L. Guan, "Design and analysis of root-protograph LDPC codes for non-ergodic block-fading channels," *IEEE Trans. Wireless Commun.*, vol. 14, no. 2, pp. 738–749, Feb. 2015.
[23] Y. Fang, Y. L. Guan, G. Bi, L. Wang, and F. C. M. Lau, "Rate-compatible root-protograph LDPC codes for quasi-static fading relay channels," *IEEE Trans. Veh. Technol.*, vol. 65, no. 4, pp. 2741–2747, Apr. 2016.
[24] R. Gabrys, H. M. Kiah, and O. Milenkovic, "Asymmetric Lee distance codes for DNA-based storage," *IEEE Trans. Inf. Theory*, vol. 63, no. 8, pp. 4982–4995, Aug. 2017.
[25] Y. Wang, M. Noor-A-Rahim, E. Gunawan, Y. L. Guan, and C. L. Poh, "Construction of bio-constrained code for DNA data storage," *IEEE Commun. Lett.*, vol. 23, no. 6, pp. 963–966, Jun. 2019.
[26] D. A. Huffman, "A method for the construction of minimum-redundancy codes," *Proc. Inst. Radio Eng.*, vol. 40, no. 9, pp. 1098–1101, Sep. 1952.
[27] D. Divsalar, C. Jones, S. Dolinar, and J. Thorpe, "Protograph based LDPC codes with minimum distance linearly growing with block size," in *Proc. IEEE Global Telecommun. Conf. (GLOBECOM)*, Nov./Dec. 2005, pp. 1152–1156.
[28] A. Abbasfar, D. Divsalar, and K. Yao, "Accumulate-repeat-accumulate codes," *IEEE Trans. Commun.*, vol. 55, no. 4, pp. 692–702, Apr. 2007.
[29] X.-Y. Hu, E. Eleftheriou, and D. M. Arnold, "Regular and irregular progressive edge-growth tanner graphs," *IEEE Trans. Inf. Theory*, vol. 51, no. 1, pp. 386–398, Jan. 2005.
[30] D. R. Bentley, S. Balasubramanian, H. P. Swerdlow, G. P. Smith, J. Milton, C. G. Brown, K. P. Hall, D. J. Evers, C. L. Barnes, H. R. Bignell, and J. M. Boutell, "Accurate whole human genome sequencing using reversible terminator chemistry," *Nature*, vol. 456, no. 7218, pp. 53–59, 2008.
[31] S. ten Brink, "Convergence behavior of iteratively decoded parallel concatenated codes," *IEEE Trans. Commun.*, vol. 49, no. 10, pp. 1727–1737, Oct. 2001.
[32] G. Liva and M. Chiani, "Protograph LDPC codes design based on EXIT analysis," in *Proc. IEEE Global Telecommun. Conf. (GLOBECOM)*, Nov. 2007, pp. 3250–3254.
[33] L. Deng, Z. Shi, O. Li, and J. Ji, "Joint coding and adaptive image transmission scheme based on DP-LDPC codes for IoT scenarios," *IEEE Access*, vol. 7, pp. 18437–18449, 2019.



LI DENG received the B.Sc. and M.Sc. degrees from Southwest University, Chongqing, China, in 2005 and 2008, respectively. She is currently pursuing the Ph.D. degree with the National Key Laboratory of Science and Technology on Communications, University of Electronic Science and Technology of China. In 2008, she joined the School of Electronic Information and Automation, Guilin University of Aerospace Technology. She is also a Visiting Scholar with the School of Electrical and Electronic Engineering, Nanyang Technological University (NTU). Her research interests include information and coding theory, low-density parity-check/protograph codes, deoxyribonucleic acid data storage, and image communication.



YIXIN WANG received the B.Sc. degree from the Harbin Institute of Technology, Weihai, China, in 2016, and the M.Sc. degree from Nanyang Technological University (NTU), Singapore, in 2017, where she is currently pursuing the Ph.D. degree with the School of Electrical Electronic Engineering. Her research interests include coding for deoxyribonucleic acid data storage, constrained codes, and error control codes.



MD. NOOR-A-RAHIM received the Ph.D. degree from the Institute for Telecommunications Research, University of South Australia, Australia, in 2015. He was a Postdoctoral Research Fellow with the Centre for Infocomm Technology (INFINITUS), Nanyang Technological University (NTU), Singapore. He is currently a Senior Postdoctoral Researcher and a Marie-Curie Research Fellow with the School of Computer Science and IT, University College Cork, Ireland. His research interests include information theory, wireless communications, and vehicular communications. He was a recipient of the Michael Miller Medal from the Institute for Telecommunications Research (ITR), University of South Australia, for the most outstanding Ph.D. thesis, in 2015.



YONG LIANG GUAN is currently a tenured Associate Professor with the School of Electrical and Electronic Engineering, Nanyang Technological University, Singapore. He has led 13 past and present externally funded research projects on advanced wireless communication techniques, coding for 10-Tb/in^2 magnetic recording, acoustic telemetry for drilling application, and so on with the total funding of over SGD 9 million. His research interests broadly include coding, signal design, and signal processing for communication systems, storage systems, and information security systems. He has published an invited monograph, three book chapters, and over 300 journal and conference papers. He was an Associate Editor of the IEEE Signal Processing Letters. He is also an Associate Editor of the IEEE Transactions on Vehicular Technology and the Chair of the IEEE ComSoc Singapore Chapter.



ZHIPING SHI received the master's and Ph.D. degrees from Southwest Jiaotong University, Chengdu, China, in 1998 and 2005, respectively. She has two years of Postdoctoral experience at the University of Electronic Science and Technology of China (UESTC), from 2005 to 2007. From 2009 to 2010, she was a Visiting Scholar with Lehigh University, PA, USA. In 2007, she joined the School of Communication and Information, UESTC. She is currently a Professor with the National Key Laboratory of Science and Technology on Communications, UESTC. Her research interests include coding theory, cognitive radio, and wireless communications.



ERRY GUNAWAN received the B.Sc. degree in electrical and electronic engineering from the University of Leeds, and the M.B.A. and Ph.D. degrees from Bradford University.

From 1984 to 1988, he was a Satellite Communication System Engineer with Communication Systems Research Ltd., Ilkley, U.K. In 1988, he moved to Space Communication (SAT-TEL) Ltd., Northampton, U.K. In 1989, he joined the School of Electrical and Electronic Engineering, Nanyang Technological University, where he is currently an Associate Professor. He has been a Consultant with Sytek Technical Associates, Singapore, on the development of a device to enhance the security of data transmitted through facsimile machines, Addvalue Communications Pte Ltd., on DECT and Bluetooth systems, and also RFNet Technologies Pte Ltd., Singapore, for IDA project on New Generation Wireless LAN (IEEE 802.11a). He conducted courses for MINDEF and NTUs MBA program. He is appointed as an External Examiner by Multimedia University for a M.Eng.Sc. candidate. He has published more than 80 papers in international journals and more than 70 international conference papers on error correction codings, modeling of cellular communications systems, power control for CDMA cellular systems, MAC protocols, multicarrier modulations, multiuser detections, space-time coding, radio-location systems, MIMO interference channel, and the applications of UWB radar for vital sign sensing and medical imaging.

Dr. Gunawan is a Technical Reviewer of various international journals, such as the IEEE TRANSACTIONS ON VEHICULAR TECHNOLOGY, IEEE JOURNAL ON SELECTED AREAS IN COMMUNICATIONS, IEEE TRANSACTIONS ON SIGNAL PROCESSING, and IEEE COMMUNICATIONS LETTERS.



CHUEH LOO POH received the B.Eng. degree in electrical and electronic engineering from Nanyang Technological University (NTU), Singapore, and the Ph.D. degree in bioengineering from Imperial College London, U.K. He is currently an Associate Professor with the Department of Biomedical Engineering, National University of Singapore (NUS), Singapore. He is also a Principal Investigator at NUS Synthetic Biology for Clinical and Technological Innovation (SynCTI) and leads the NUS Biofoundry. He is also the Assistant Dean (Outreach-External Relations and Outreach) of the Faculty of Engineering, NUS. He is also the Co-Founder of a Singapore start-up company, AdvanceSyn Pte Ltd., which specializes on providing model-assisted design tools and services for Synthetic Biology. His research group has been reprogramming microbes for medical and industrial applications. His current research interests include microbial biosensors, optogenetics, synthetic gene circuits design and automation, deoxyribonucleic acid data storage, modeling of biological systems for design, and computer-aided design (CAD) tools for SynBio. He has received a number of awards, including the Tan Chin Tuan Fellowship, in 2012, and the NTU Excellence in Teaching Award, in 2010. He is also the Co-Editor-in-Chief of *IET Engineering Biology* journal.