Properties of Maxentropic DNA Synthesis Codes †

Schouhamer Immink, Kees A.; Weber, Jos H.; Cai, Kui

DOI
10.3390/e26121028

Publication date
2024

Document Version
Final published version

Published in
Entropy: international and interdisciplinary journal of entropy and information studies

entropy

MDPI

Article

Properties of Maxentropic DNA Synthesis Codes [†]

Kees Schouhamer Immink [1,*], Jos H. Weber [2] and Kui Cai [3]

[1] Turing Machines Inc., Willemskade 15, 3016 DK Rotterdam, The Netherlands
[2] Department of Applied Mathematics, Delft University of Technology, 2628 CD Delft, The Netherlands; j.h.weber@tudelft.nl
[3] Science, Mathematics and Technology Cluster, Singapore University of Technology and Design (SUTD), 8 Somapah Rd, Singapore 487372, Singapore; cai_kui@sutd.edu.sg
[*] Correspondence: immink@turing-machines.com

[†] This work was supported by SUTD Kickstarter Initiative (SKI) Grant 2021_04_05.

Abstract: Low-weight codes have been proposed for efficiently synthesizing deoxyribonucleic acid (DNA) for massive data storage, where a multiple of DNA strands are synthesized in parallel. We report on the redundancy and information rate of maxentropic low-weight codes for asymptotically large codeword length. We compare the performance of low-complexity nibble replacement (NR) codes, which are designed to minimize the synthesis time, with the performance of maxentropic low-weight codes. Finally, the asymptotic redundancy and information rate of codes with a runlength limitation are investigated.

Keywords: code design; DNA synthesis; low-weight code; maximum runlength constraint; nibble replacement (NR) code

check for updates

Citation: Immink, K.S.; Weber, J.H.; Cai, K. Properties of Maxentropic DNA Synthesis Codes. *Entropy* **2024**, *26*, 1028. https://doi.org/10.3390/e26121028

Academic Editors: Olgiça Milenković and Jin Sima

Received: 28 October 2024
Revised: 19 November 2024
Accepted: 27 November 2024
Published: 27 November 2024

1. Introduction

The pioneering experiments conducted by Church et al. [1] demonstrated the feasibility to store data in synthetic deoxyribonucleic acid (DNA), promising a huge data capacity, nil dissipation during storage, and very long-term stability. Natural DNA consists of four types of nucleotides: adenine ('A'), cytosine ('C'), guanine ('G'), and thymine ('T'). Codes are used for translating user data into sequences of digits in the quaternary alphabet {A, C, G, T} that are suitable for the synthesis of DNA strands. Prior art studies have focused on error-correcting codes for restoring various kinds of defects in DNA [2–4] or constrained codes that avoid the generation of vexatious DNA sequences that are prone to error; see, for example, [5–8].

The synthesis of DNA strands is a relative expensive part of the storage chain. In array-based synthesis, multiple DNA strands are synthesized in parallel [9] by adding in each cycle a single nucleotide to a subset of the DNA strands. Lenz et al. [10], Makarychev et al. [11], Elishco and Huleihel [12], Immink et al. [13], and Nguyen et al. [14] presented and analyzed coding techniques for efficiently synthesizing multiple parallel strands so that overall synthesis time can be shortened. Of specific interest in minimizing the synthesis time are sets (codes) of words of low weight, which are dealt with in the next subsection.

1.1. Low-Weight Codes

Although our main interest is in the quaternary DNA case, we will consider q-ary sequences for generality. For clerical convenience, we assume that the alphabet used is $\{1, \ldots, q\}$, where $q > 1$ is a positive integer. For the DNA case, we represent the quaternary alphabet {A, C, G, T} by $\{1, \ldots, 4\}$. Let $\boldsymbol{a} = (a_1, \ldots, a_n)$, $a_i \in \{1, \ldots, q\}$, be a sequence of n symbols, called *word* of length n. The symbol sum

$$w(\boldsymbol{a}) = \sum_{i=1}^{n} a_i \qquad (1)$$

is termed the *weight* of the word $\boldsymbol{a}$. Clearly, $n \leq w(\boldsymbol{a}) \leq qn$. A *constant-weight code* of length n, denoted by $S_n(w)$, consists of all words of weight w, that is,

$$S_n(w) = \{\boldsymbol{a} \in \{1, \ldots, q\}^n : w(\boldsymbol{a}) = w\}. \tag{2}$$

The size of $S_n(w)$, denoted by $|S_n(w)|$, is found as the coefficient of z^w of the *generating function* [15]

$$\left(\sum_{i=1}^{q} z^i \right)^n. \tag{3}$$

For synthesizing multiple words into physical sequences in parallel, we assume the sequences are generated by adding symbols in cycles. In each cycle in the synthesis process, one particular symbol from $\{1, \ldots, q\}$ is added to the sequences of the words waiting for that symbol. Throughout this paper, we assume the symbol adding in the subsequent cycles is in the order $1, 2, \ldots, q, 1, 2, \ldots, q, 1, 2, \ldots$, which has been shown to be optimal; see [10,12]. In order to allow any word from $\{1, \ldots, q\}^n$ to be synthesized, qn cycles are needed. By restricting the set of words used for representing data, the number of required synthesis cycles can be reduced, as explained next.

Let the *low-weight code* $\cup_{w=n}^{t} S_n(w)$ be the union of the sets of words of weight $w \leq t$, where the integer t, $n \leq t \leq qn$ denotes the maximum weight of the codewords. As explained in [10,13], low-weight codewords $\boldsymbol{y}$ can be bijectively mapped to words $\boldsymbol{x} = (x_1, \ldots, x_n)$, $x_i \in \{1, \ldots, q\}$, by

$$x_i = x_{i-1} + y_i \bmod q, \tag{4}$$

with $x_0 = q$, such that the words $\boldsymbol{x}$ have a synthesis time of at most t cycles. Let the low-weight code be denoted as $Y_n(\gamma)$, where $\gamma = t/n$, and the associated set of words $\boldsymbol{x}$ as $X_n(\gamma)$. From the synthesis perspective, we are interested in properties of the codes $X_n(\gamma)$, but because of the bijective mapping we can also study the low-weight codes $Y_n(\gamma)$.

1.2. Redundancy and Information Rate

The *redundancy* (in bits per symbol) of a low-weight code $Y_n(\gamma)$ is defined by

$$\rho_n(\gamma) = \log_2(q) - R_n(\gamma), \tag{5}$$

where

$$R_n(\gamma) = \frac{1}{n} \log_2 |Y_n(\gamma)|. \tag{6}$$

Lenz et al. [10] also introduced the *information rate* (in bits per cycle) of a low-weight code $Y_n(\gamma)$ as

$$W_n(\gamma) = \frac{1}{n\gamma} \log_2 |Y_n(w)| = \frac{1}{\gamma} R_n(\gamma). \tag{7}$$

Of course, $\rho_n(\gamma)$ and $W_n(\gamma)$ are also the redundancy and information rate, respectively, of $X_n(\gamma)$. Note that $W_n(\gamma)$ is a measure for the synthesis efficiency of the codewords of $X_n(\gamma)$.

Using (3), we can straightforwardly compute $\rho_n(\gamma)$ and $W_n(\gamma)$ versus γ. Figures 1 and 2 show the results for $n = 16$, 32, 64, and $q = 4$. The curves suggest that $\rho_n(\gamma)$ and $W_n(\gamma)$ have a lower bound and upper bound, respectively, for asymptotic large n. A major goal of this paper is to determine these bounds.

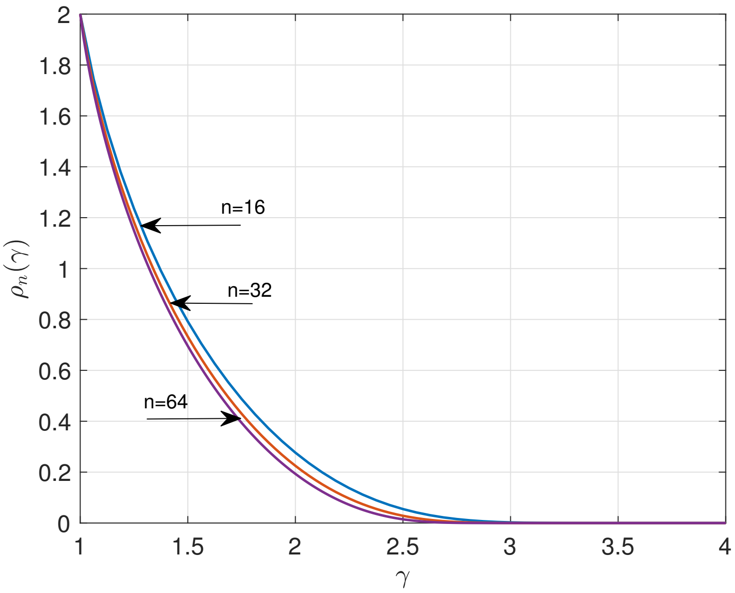

Figure 1. Redundancy $\rho_n(\gamma)$ versus γ for $n = 16, 32, 64$, and $q = 4$.

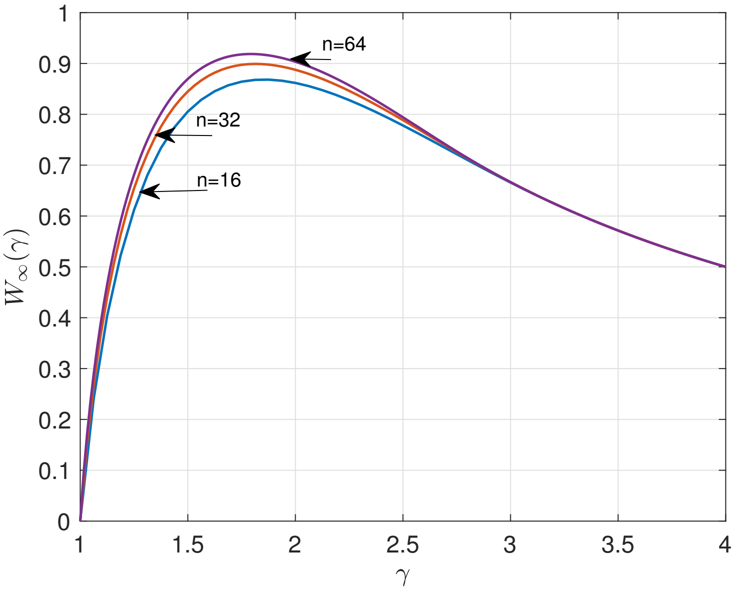

Figure 2. Information rate $W_n(\gamma)$ versus γ for $n = 16, 32, 64$, and $q = 4$.

1.3. Contributions and Overview of the Paper

Besides introducing the framework as just described, Lenz et al. [10] also conducted a brief performance analysis of DNA synthesis codes, mainly based on tools from the theory of cost-constrained channels. Constructions of efficient DNA synthesis codes were further explored in [13]. Here, in this paper, Section 2 deals with an extensive asymptotic analysis, with a focus on the trade-off between redundancy and information rate. The results are derived using Jaynes' maximum entropy principle. In Section 3, we compare the obtained theoretical optima with the performance of practical nibble replacement codes. Finally, we extend the analysis to codes with a runlength constraint in Section 4 and conclude the paper in Section 5.

2. Asymptotic Analysis of Low-Weight Codes

In order to evaluate the sizes of large low-weight codes, we use the following approach. Let $C_n(w)$ be the set of compositions $\boldsymbol{c} = (n_1, \ldots, n_q)$ of n, where n_i are nonnegative integers such that $\sum_{i=1}^{q} n_i = n$ subject to the constraint $\sum_{i=1}^{q} in_i = w$. The number of q-ary words of length n with n_i symbols equal to i, $1 \le i \le q$, denoted by $N_{\boldsymbol{c}}$, equals

$$N_{\boldsymbol{c}} = \frac{n!}{n_1! \ldots n_q!}. \tag{8}$$

The constant-weight code size, $|S_n(w)|$, is found by summing $N_{\boldsymbol{c}}$ for all possible compositions $\boldsymbol{c} \in C_n(w)$ so that

$$|S_n(w)| = \sum_{\boldsymbol{c} \in C_n(w)} N_{\boldsymbol{c}}. \tag{9}$$

2.1. Asymptotic Analysis of $R_n(\gamma)$

We are specifically interested in $R_n(\gamma)$ for asymptotically large n. So, let $n \to \infty$ and $n_i \to \infty$ for all i, while keeping $p_i = n_i/n$, $1 \le i \le q$, the distribution of the symbol values, fixed. It then follows (see Wallis argument in Section 11.4 of [16]), using Stirling's approximation, that

$$\frac{1}{n} \log_2 N_{\boldsymbol{c}} \to H_{\boldsymbol{c}}, \tag{10}$$

and thus

$$\frac{1}{n} \log_2 |S_n(w)| \to \max_{\boldsymbol{c} \in C_n(w)} H_{\boldsymbol{c}}, \tag{11}$$

where

$$H_{\boldsymbol{c}} = -\sum_{i=1}^{q} p_i \log_2 p_i. \tag{12}$$

In a similar vein, we find

$$R_n(\gamma) \to \max_{n \le w \le \gamma n} \frac{1}{n} \log_2 |S_n(w)| \to \max_{n \le w \le \gamma n} \max_{\boldsymbol{c} \in C_n(w)} H_{\boldsymbol{c}}. \tag{13}$$

Since $\frac{1}{n} \log_2 |S_n(w)|$ is monotonically increasing with w, $n \le w \le \gamma n$, if $1 \le \gamma \le (q+1)/2$, with a maximum $\log_2(q)$ at $w = n(q+1)/2$, we infer

$$R_n(\gamma) \to \begin{cases} \max_{\boldsymbol{c} \in C_n(\gamma n)} H_{\boldsymbol{c}}, & 1 \le \gamma < (q+1)/2, \\ \log_2(q), & (q+1)/2 \le \gamma \le q. \end{cases} \tag{14}$$

The problem of determining $R_\infty(\gamma) = \lim_{n \to \infty} R_n(\gamma)$, and thus the asymptotic redundancy $\rho_\infty(\gamma) = \log_2(q) - R_\infty(\gamma)$ and the asymptotic information rate $W_\infty(\gamma) = \frac{1}{\gamma} R_\infty(\gamma)$, is now a matter of finding, for asymptotically large n, a $\boldsymbol{c}$ in $C_n(\gamma n)$ that maximizes $H_{\boldsymbol{c}}$. The composition $\boldsymbol{c} = (n_1, \ldots, n_q)$ in $C_n(\gamma n)$ is characterized by

$$\sum_{i=1}^{q} n_i = n \text{ and } \sum_{i=1}^{q} in_i = \gamma n, \tag{15}$$

which can be conveniently rewritten as

$$\sum_{i=1}^{q} p_i = 1 \text{ and } \sum_{i=1}^{q} ip_i = \gamma. \tag{16}$$

In the next subsection, we maximize $H_{\boldsymbol{c}}$ by a judicious choice of the distribution of the symbol values, p_i, under these conditions.

2.2. Principle of Maximum Entropy

We change the above setting of finite-length codewords and now assume a stationary information source that transmits symbols of (integer) magnitude i, $i \in \{1, \ldots, q\}$, with probability distribution $\boldsymbol{p} = (p_1, \ldots, p_q)$, where $p_i \in \mathbb{R}$ and $\sum p_i = 1$. The information content per symbol sent, or *entropy*, denoted by H, defined by Shannon [17], is

$$H = -\sum_{i=1}^{q} p_i \log_2 p_i. \tag{17}$$

Although the variable H_c in (12) and Shannon's entropy H share the same expression in $\boldsymbol{p}$, the background of the expressions is different [16]. Note that in (12), the p_i's are rational numbers, while in (17) the p_i's are assumed to be real numbers.

We are interested in maximizing the entropy H. Define

$$\hat{H}(\gamma) = \max_{p_1,\ldots,p_q} H, \tag{18}$$

$1 \le \gamma \le q$, where the maximization over the p_i is under the conditions (16). Jaynes [18] concluded that the entropy, H, is maximized subject to these constraints by the maxentropic probability distribution

$$\hat{p}_i = 2^{\alpha - \beta i},\ 1 \le i \le q, \tag{19}$$

where the parameters α and β, $\alpha, \beta \in \mathbb{R}$, satisfy the conditions

$$\alpha = -\log_2 \sum_{i=1}^{q} 2^{-\beta i} \tag{20}$$

and

$$\sum_{i=1}^{q} i 2^{\alpha - \beta i} = \gamma. \tag{21}$$

After substituting (19) to (21) into (17), we find

$$\hat{H}(\gamma) = \beta\gamma - \alpha. \tag{22}$$

For the case $q = 2$, we may easily find that $p_1 + p_2 = 1$ and $p_1 + 2p_2 = \gamma$, so that $p_1 = 2 - \gamma$, $p_2 = \gamma - 1$, and

$$\hat{H}(\gamma) = -(2-\gamma)\log_2(2-\gamma) - (\gamma - 1)\log_2(\gamma - 1), \tag{23}$$

$1 \le \gamma \le 2$. For $q > 2$, no simple closed-form expression could be found, and we use numerical methods for solving (20) and (21).

2.3. Asymptotic Redundancy and Information Rate

From (14), we obtain

$$R_\infty(\gamma) = \begin{cases} \hat{H}(\gamma), & 1 \le \gamma < (q+1)/2, \\ \log_2(q), & (q+1)/2 \le \gamma \le q. \end{cases} \tag{24}$$

As a result, the asymptotic redundancy is

$$\rho_\infty(\gamma) = \begin{cases} \log_2(q) - \hat{H}(\gamma), & 1 \le \gamma < (q+1)/2, \\ 0, & (q+1)/2 \le \gamma \le q. \end{cases} \tag{25}$$

Figure 3 depicts, for $q = 2, 4$, and 6, the relationship between the asymptotic redundancy $\rho_\infty(\gamma)$ and γ.

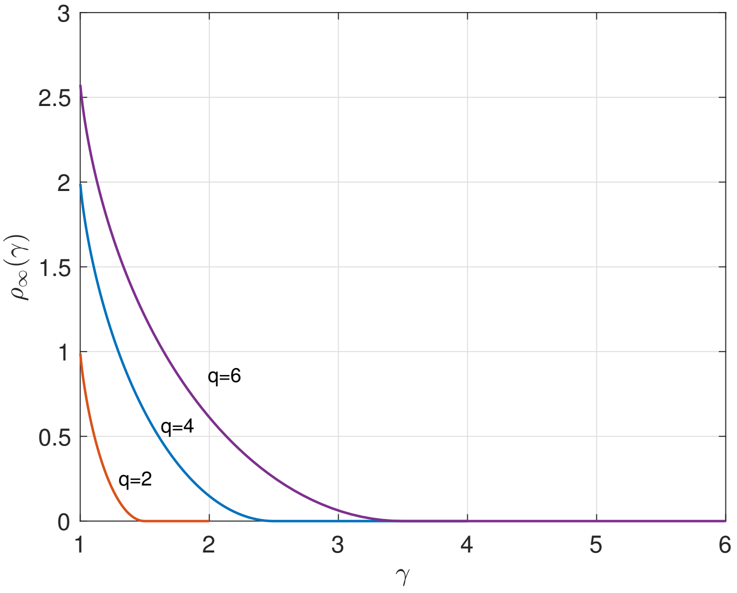

Figure 3. Redundancy $\rho_\infty(\gamma)$ versus γ, $q = 2, 4$, and 6.

The asymptotic information rate, $W_\infty(\gamma)$, equals

$$W_\infty(\gamma) = \begin{cases} \hat{H}(\gamma)/\gamma, & 1 \le \gamma < (q+1)/2, \\ \log_2(q)/\gamma, & (q+1)/2 \le \gamma \le q. \end{cases} \tag{26}$$

Figure 4 shows $W_\infty(\gamma)$ versus γ for $q = 2, 4$, and 6.

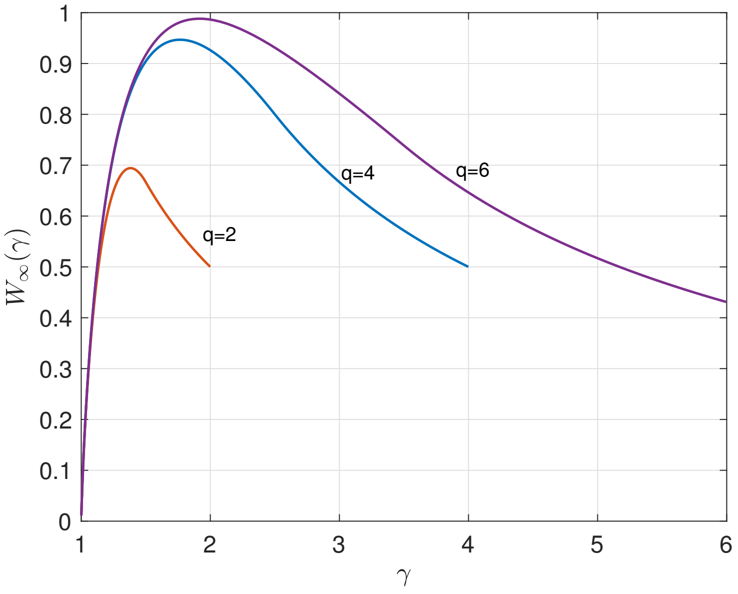

Figure 4. Information rate $W_\infty(\gamma)$ versus γ, $q = 2, 4$, and 6.

The maximum asymptotic information rate, denoted by

$$\hat{W}_\infty = \max_\gamma W_\infty(\gamma),$$

can be found after an analysis of (22). We write (22), using (20) and (21), as a function of β and conclude that the largest (real) root of

$$\sum_{i=1}^{q} 2^{-i\beta} = 1, \tag{27}$$

denoted by $\hat{\beta}$, maximizes $W_\infty(\gamma)$. We obtain, see (20), $\alpha = 0$ and hence, see (22), we infer that

$$\hat{H}_\infty = \hat{H}(\hat{\gamma}) = \hat{\beta}\hat{\gamma}, \tag{28}$$

where

$$\hat{\gamma} = \sum_{i=1}^{q} i2^{-\hat{\beta}i}, \tag{29}$$

and

$$\hat{W}_\infty = \hat{\beta}. \tag{30}$$

Note that Equation (27) is equivalent to the characteristic equation $\sum_{i=1}^{q} z^{-i} = 1$ of a binary source under the constraint that the maximum runlength is q [19]. The capacity of binary sequences with a maximum runlength constraint of q equals $\log_2(\hat{z})$, where $\hat{z}$ is largest (real) root of the characteristic equation [17]. Hence, the maximum asymptotic information rate $\hat{W}_\infty = \hat{\beta}$ of q-ary low-weight codes is equal to this capacity. Numerical values of the latter have been listed for selected values of q in [19]. Since the capacity approaches unity for increasing values of q, the same holds for the information rate $\hat{W}_\infty$, which is achieved for $\hat{\gamma} \to 2$. In other words, for large values of n and q, the maximum information rate is achieved by setting the maximum weight of the low-weight code equal to (roughly) $2n$. The corresponding redundancy is $\log_2(q) - \hat{\gamma}\hat{\beta} \to \log_2(q) - 2$. For any q, the asymptotic redundancy can be lowered from $\log_2(q) - \hat{\gamma}\hat{\beta}$ to zero by increasing γ from $\hat{\gamma}$ to $(q+1)/2$, which implies that the asymptotic information rate decreases from $\hat{\beta}$ to $2\log_2(q)/(q+1)$. This trade-off between redundancy and information rate is further explored for the case $q = 4$ in the next subsection.

2.4. Case Study for $q = 4$

In this subsection, we consider the case $q = 4$, which is of particular interest since it is the alphabet size for DNA synthesis codes. For $q = 4$, we find using numerical methods that $\hat{W}_\infty = \hat{\beta} = 0.947$, $\hat{\gamma} = 1.766$, and $\hat{H}_\infty = 1.672$. The probability distribution at maximum entropy is $\hat{\boldsymbol{p}} = (0.519, 0.269, 0.140, 0.072)$. Figure 5 shows the parametric representation of $W_\infty(\gamma)$ versus $\rho_\infty(\gamma)$ with γ as a parameter for the case $q = 4$. The curve is a typical price/performance curve, where we may observe that a higher $W_\infty(\gamma)$ comes with a higher penalty in redundancy $\rho_\infty(\gamma)$.

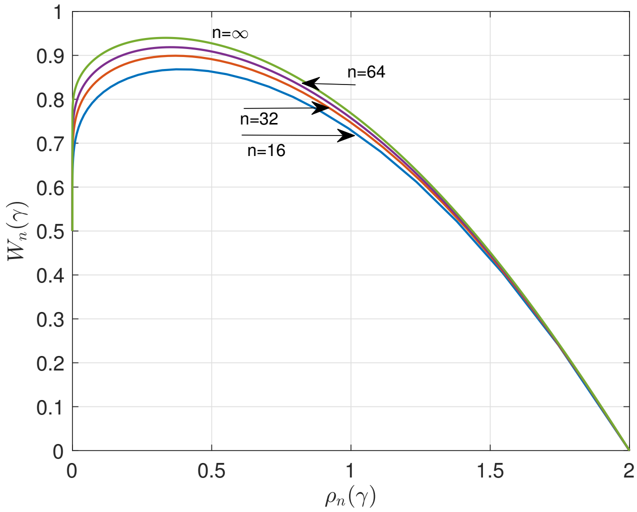

Figure 5. Parametric relationship between maxentropic information rate $W_n(\gamma)$ versus redundancy $\rho_n(\gamma)$, $q = 4$. As a comparison, we plotted $W_n(\gamma)$ versus $\rho_n(\gamma)$ for $n = 16, 32, 64$.

It is the difficult task of a system designer to trade the costs and benefits of the conflicting parameters. Note that in the range $\gamma \geq 5/2$ we have $\rho_\infty(\gamma) = 0$, a zero-redundant system, while in the range $\gamma < \hat{\gamma}$ we may achieve the same information rate $W_\infty(\gamma)$ with a smaller redundancy. For example, we may notice that we can achieve $W_\infty(\gamma) = 0.8$ for zero redundancy cost or for roughly 0.9. In practice, we prefer the smaller redundancy alternative so that in this range of practical interest, we have $4/5 \leq W_\infty(\gamma) \leq \hat{W}_\infty = 0.947$ and $0 \leq \rho_\infty(\gamma) \leq 2 - \hat{H}_\infty = 0.328$. Figure 6 displays $W_\infty(\gamma)$ versus $\rho_\infty(\gamma)$ in the range of practical interest $\hat{\gamma} \leq \gamma < 5/2$.

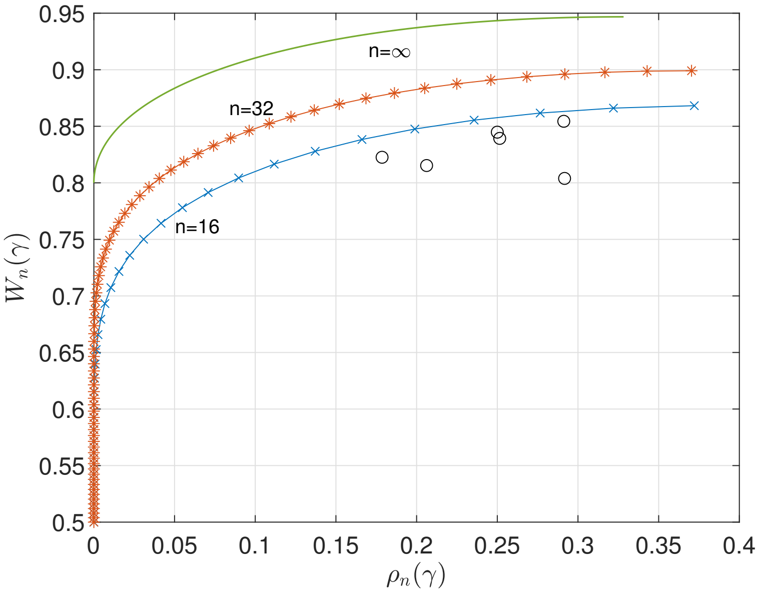

Figure 6. Parametric relationship between (maxentropic) information rate $W_n(\gamma)$ versus redundancy $\rho_n(\gamma)$, $n = 16, 32, \infty$, $q = 4$, in the range of practical interest. The black circles refer to the NR codes compiled in Table 1.

Table 1. Results of the NR coding method for selected values of subword length m and maximum subword cycle count, t_m, $q = 4$.

m	t_m	m_h	L	$\rho(t)$	$W(t)$
8	17	14	3	0.2917	0.8039
10	22	18	16	0.2062	0.8153
12	25	21	56	0.2515	0.8393
14	31	26	2	0.1786	0.8226
14	29	25	2	0.2500	0.8448
14	28	24	13	0.2912	0.8544

3. Comparison with Implemented Codes

In this section, we compare the performance of implemented codes with that of maxentropic low-weight codes. In [13], various code implementations have been assessed. Here, we focus on the *nibble replacement (NR) algorithm* [13,20], which is an efficient method for encoding/decoding with small complexity and redundancy.

In the NR format, an n-symbol strand is divided into L subwords of length m, so that $n = Lm$. Let t_m be the maximum allowed cycle count of an m-symbol q-ary word, then the maximum cycle count of the n-symbol q-ary word is $t = Lt_m$. Let

$$M = \sum_{w=m}^{t_m} |S_m(w)| \tag{31}$$

denote the number of low-weight m-symbol codewords. Define $m_h = \lceil \log_2 M \rceil$ and

$$L = \left\lfloor \frac{2^{m_h - 1}}{2^{m_h} - M} \right\rfloor. \tag{32}$$

The NR algorithm translates $Lm_h - 1$ source bits into L m_h-bit words. Each m_h-bit word is translated, using a look-up table, into a q-ary m-symbol word that satisfies the t_m-cycle count constraint. The NR encoding method requires data storage of L m_h-bit words, the execution of the encoding algorithm [13], and a look-up table for translating an m_h-bit wide word into a word of m q-ary symbols so that very large, n-symbol wide, look-up tables are avoided. The overall redundancy per symbol, $\rho(t)$, and information rate, $W(t)$, of the n-symbol word are

$$\rho(t) = \frac{L(\log_2(q)m - m_h) + 1}{n} \tag{33}$$

and

$$W(t) = \frac{Lm_h - 1}{t}. \tag{34}$$

Table 1 shows numerical results selected from Table I in [13].

The scattered points (black circles) in Figure 6 are found by plotting the redundancy, $\rho(t)$, and information rate, $W(t)$, of the NR codes shown in Table 1.

4. Runlength Limitation

It is known that homopolymer runs, i.e., adjacent repetitions of the same nucleotide, make DNA-based data storage more error prone [12]. Therefore, it could be advantageous to use strands in which long runs are avoided. Of course, this comes at the expense of an increased redundancy. In this section, we perform an asymptotic analysis of codes aiming at (i) small redundancy, (ii) high information rate, and (iii) small maximum runlength. These are conflicting goals resulting into trade-off considerations. Again, we start by investigating q-ary codes and then focus on the $q = 4$ case.

We say that a code is r-RLL (runlength limited) if within any codeword any run of identical symbols is of length at most r, where $1 \le r \le n$. When $r = n$, there is actually no constraint with respect to the runlength. Here, we focus on the other extreme, $r = 1$; i.e., we consider codewords in which any two adjacent symbols are different. We investigate the asymptotic redundancy and information rate of q-ary 1-RLL codes. The same notation as before is used, where we indicate with a tilde that the $r = 1$ constraint is in place.

Let $\tilde{X}_n(\gamma)$ denote the q-ary code consisting of all 1-RLL sequences that can be synthesized in at most $t = \gamma n$ cycles. The codewords $\tilde{\boldsymbol{y}} = (\tilde{y}_1, \ldots, \tilde{y}_n)$ of the associated low-weight code $\tilde{Y}_n(\gamma)$ are obtained from the codewords $\tilde{\boldsymbol{x}} = (\tilde{x}_1, \ldots, \tilde{x}_n)$ of $\tilde{X}_n(\gamma)$ by the bijective mapping

$$\tilde{y}_i = \tilde{x}_i - \tilde{x}_{i-1} \bmod q \tag{35}$$

with $\tilde{x}_0 = q$. Note that due to the 1-RLL property of $\tilde{\boldsymbol{x}}$, it holds that $\tilde{x}_i \neq \tilde{x}_{i-1}$ and thus $\tilde{y}_i \neq q$ for all $2 \le i \le n$. Hence, $\tilde{Y}_n(\gamma) = \cup_{w=n}^{\gamma n} \tilde{S}_n(w)$, where

$$\tilde{S}_n(w) = \{\tilde{\boldsymbol{y}} \in \{1, \ldots, q\}^n : w(\tilde{\boldsymbol{y}}) = w \wedge \tilde{y}_i \neq q \forall i \ge 2\} \tag{36}$$

and the range for γ is in this case $1 \le \gamma \le q - 1 + 1/n$, since the maximum number of cycles is $(q-1)n + 1$ rather than qn due to the runlength constraint.

Similarly to what we did before, we next evaluate

$$\tilde{R}_n(\gamma) = \frac{1}{n} \log_2 |\tilde{Y}_n(\gamma)|. \tag{37}$$

Since the symbol distribution $(\tilde{p}_1, \ldots, \tilde{p}_q)$ satisfies, for any codeword in the low-weight code,

$$\tilde{p}_q \leq \frac{1}{n} \to 0 \tag{38}$$

as $n \to \infty$, we can conclude that the value of $\tilde{R}_\infty(\gamma)$ in the q-ary case is equal to the value of $R_\infty(\gamma)$ in the $(q-1)$-ary case. Hence, it easily follows that

- The asymptotic redundancy $\tilde{\rho}_\infty(\gamma)$ in the q-ary 1-RLL case equals $\log_2(q/(q-1))$ plus the asymptotic redundancy $\rho_\infty(\gamma)$ in the $(q-1)$-ary case without runlength restriction;
- The asymptotic information rate $\tilde{W}_\infty(\gamma)$ in the q-ary 1-RLL case equals the asymptotic information rate $W_\infty(\gamma)$ in the $(q-1)$-ary case without runlength restriction.

As an illustration, we consider the case $q = 4$. By applying the results from (25) and (26) for $q = 3$, we find $\tilde{\rho}_\infty(\gamma)$ and $\tilde{W}_\infty(\gamma)$ for $q = 4$. These 1-RLL results are compared to the corresponding results without runlength limitation from Section 2 in Figures 7–9. Results for r-RLL codes, $1 < r < \infty$, will be in between the lower and the upper curves in these figures. Various trade-off possibilities can be considered. Note that, for small values of γ, imposing the runlength limitation comes at hardly any price, but that for larger values of γ we considerably pay in terms of redundancy and information rate. Fixing the asymptotic redundancy at, e.g., 0.5, it follows from Figure 9 that the asymptotic information rate drops from about 0.93 (∞-RLL, i.e., no runlength limitation) to about 0.87 (1-RLL).

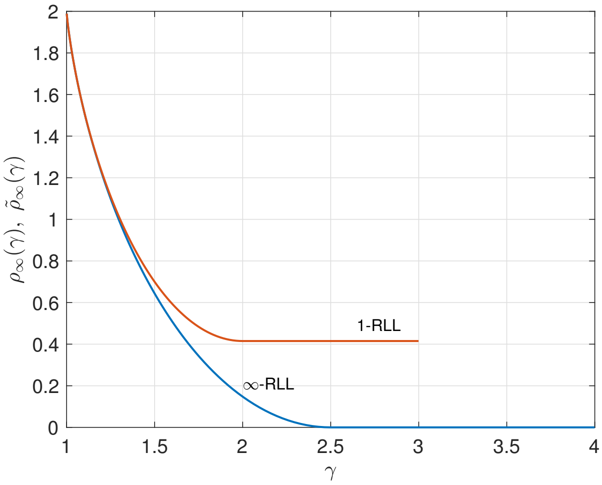

Figure 7. Asymptotic redundancies $\rho_\infty(\gamma)$ (∞-RLL) and $\tilde{\rho}_\infty(\gamma)$ (1-RLL) versus γ for the case $q = 4$.

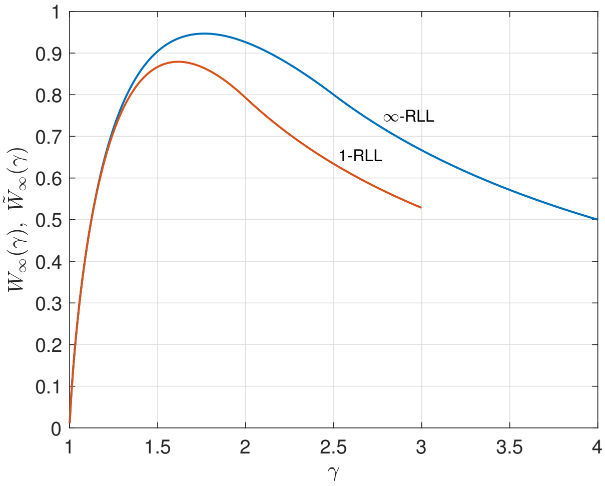

Figure 8. Asymptotic information rate $W_\infty(\gamma)$ (∞-RLL) and $\tilde{W}_\infty(\gamma)$ (1-RLL) versus γ for the case $q = 4$.

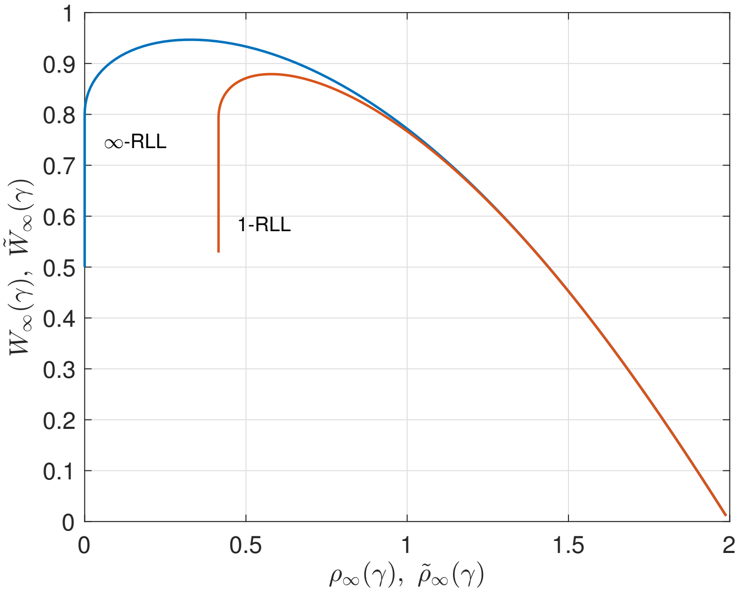

Figure 9. Parametric relationship between the asymptotic information rate and asymptotic redundancy for the ∞-RLL and 1-RLL cases, $q = 4$.

5. Conclusions

We have analyzed coding techniques for efficiently synthesizing multiple parallel DNA strands. We have computed the maxentropic redundancy and information rate of low-weight codes, $\rho_n(\gamma)$ and $W_n(\gamma)$, for asymptotically large codeword length n using Jaynes' maximum entropy principle. We have compared the performance of low-complexity NR codes, which are designed to minimize the synthesis time, with the performance of maxentropic low-weight codes. Finally, the performance of codes with a runlength limitation has been evaluated. All the presented results allow for making trade-offs between synthesis time and redundancy for long codes.

Author Contributions: Writing – review & editing, K.S.I., J.H.W. and K.C. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by SUTD Kickstarter grant number 2021_04_05.

Data Availability Statement: The original contributions presented in the study are included in the article, further inquiries can be directed to the corresponding author

Conflicts of Interest: Author Kees Schouhamer Immink was employed by the company Turing Machines Inc. The remaining authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

References

1. Church, G.M.; Gao, Y.; Kosuri, S. Next-generation digital information storage in DNA. *Science* **2012**, *337*, 1628–1628. https://doi.org/10.1126/science.1226355.
2. Lenz, A.; Siegel, P.H.; Wachter-Zeh, A.; Yaakobi, E. Coding Over Sets for DNA Storage. *IEEE Trans. Inf. Theory* **2020**, *66*, 2331–2351. https://doi.org/10.1109/TIT.2019.2961265.
3. Nguyen, T.T.; Cai, K.; Immink, K.A.S.; Kiah, H.M. Capacity-Approaching Constrained Codes with Error Correction for DNA-Based Data Storage. *IEEE Trans. Inf. Theory* **2021**, *67*, 5602–5613. https://doi.org/10.1109/TIT.2021.3066430.
4. Milenkovic, O.; Pan, C. DNA-Based Data Storage Systems: A Review of Implementations and Code Constructions. *IEEE Trans. Commun.* **2024**, *72*, 3803–3828. https://doi.org/10.1109/TCOMM.2024.3367748.
5. Blawat, M.; Gaedke, K.; Hutter, I.; Cheng, X.; Turczyk, B.; Inverso, S.; Pruitt, B.W.; Church, G.M. Forward Error Correction for DNA Data Storage. *Procedia Comput. Sci.* **2016**, *80*, 1011–1022. https://doi.org/10.1016/j.procs.2016.05.398.
6. Benerjee, K.G.; Banerjee, A. On DNA Codes With Multiple Constraints. *IEEE Commun. Lett.* **2021**, *25*, 365–368. https://doi.org/10.1109/LCOMM.2020.3029071.

7. Milenkovic, O.; Kashyap, N. DNA codes that avoid secondary structures. In Proceedings of the International Symposium on Information Theory, ISIT 2005, Adelaide, SA, Australia, 4–9 September 2005; pp. 288–292. https://doi.org/10.1109/ISIT.2005.1523340.
8. Benerjee, K.G.; Banerjee, A. On Homopolymers and Secondary Structures Avoiding, Reversible, Reversible-Complement and GC-balanced DNA Codes. In Proceedings of the 2022 IEEE International Symposium on Information Theory (ISIT), Espoo, Finland, 26 June–1 July 2022; pp. 204–209. https://doi.org/10.1109/ISIT50566.2022.9834744.
9. Kosuri, S.; Church, G.M. Large-scale de novo DNA synthesis: Technologies and applications. *Nat. Methods* **2014**, *11*, 499–507. https://doi.org/10.1038/nmeth.2918.
10. Lenz, A.; Liu, Y.; Rashtchian, C.; Siegel, P.H.; Wachter-Zeh, A.; Yaakobi, E. Coding for Efficient DNA Synthesis. In Proceedings of the 2020 IEEE International Symposium on Information Theory (ISIT), Los Angeles, CA, USA, 21–26 June 2020; pp. 2885–2890. https://doi.org/10.1109/ISIT44484.2020.9174272.
11. Makarychev, K.; Racz, M.Z.; Rashtchian, C.; Yekhanin, S. Batch Optimization for DNA Synthesis. *IEEE Trans. Inf. Theory* **2022**, *68*, 7454–7470. https://doi.org/10.1109/TIT.2022.3184903.
12. Elishco, O.; Huleihel, W. Optimal Reference for DNA Synthesis. *IEEE Trans. Inf. Theory* **2023**, *69*, 6941–6955. https://doi.org/10.1109/TIT.2023.3286694.
13. Immink, K.A.S.; Cai, K.; Nguyen, T.T.; Weber, J.H. Constructions and properties of efficient DNA synthesis codes. *IEEE Trans. Mol. Biol.-Multi-Scale Commun.* **2024**, *10*, 289–296. https://doi.org/10.1109/TMBMC.2024.3401583.
14. Nguyen, T.T.; Cai, K.; Immink, K.A.S. Efficient DNA Synthesis Codes with Error Correction and Runlength Limited Constraint. In Proceedings of the 2024 IEEE International Symposium on Information Theory (ISIT), Athens, Greece, 7–12 July 2024, pp. 669–674. https://doi.org/10.1109/ISIT57864.2024.10619131.
15. Flajolet, P.; Sedgewick, R. *Analytic Combinatorics*; Cambridge University Press: Cambridge, UK, 2009; ISBN 978-0-521-89806-5.
16. Jaynes, E.T. *Probability Theory: The Logic of Science*; Cambridge University Press: Cambridge, MA, USA, 2003.
17. Shannon, C.E. A Mathematical Theory of Communication. *Bell Syst. Tech. J.* **1948**, *27*, 379–423. https://doi.org/10.1145/584091.584093.
18. Jaynes, E.T. *Probability, Statistics and Statistical Physics*; Rosenkrantz, R., Ed.; Springer: Berlin/Heidelberg, Germany, 1989.
19. Tang, D.T.; Bahl, L.R. Block Codes for a Class of Constrained Noiseless Channels. *Inf. Control* **1970**, *17*, 436–461. https://doi.org/10.1016/S0019-9958(70)90369-4.
20. Immink, K.A.S.; Cai, K. Efficient encoding of constrained block codes. *IEEE Commun. Lett.* **2021**, *25*, 3468–3472. https://doi.org/10.1109/LCOMM.2021.3105327.

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.