Codes for Cost-Efficient DNA Synthesis

Andreas Lenz[*], **Yi Liu**[‡], **Cyrus Rashtchian**[†], **Paul H. Siegel**[‡],
Andrew Tan[‡], **Antonia Wachter-Zeh**[*], and **Eitan Yaakobi**[§]
[*]Institute for Communications Engineering, Technical University of Munich, Germany
[†]Computer Science and Engineering Department and the Qualcomm Institute, University of California, San Diego
[‡]Department of Electrical and Computer Engineering, CMRR, University of California, San Diego
[§]Computer Science Department, Technion – Israel Institute of Technology, Haifa, Israel
andreas.lenz@mytum.de, yil333@eng.ucsd.edu, crashtchian@eng.ucsd.edu, psiegel@ucsd.edu,
antonia.wachter-zeh@tum.de, yaakobi@cs.technion.ac.il

***Abstract*—As a step toward more efficient DNA data storage systems, we study the design of codes that minimize the time and number of required materials needed to synthesize the DNA strands. We consider a popular synthesis process that builds many strands in parallel in a step-by-step fashion using a fixed supersequence $\boldsymbol{S}$. The machine iterates through $\boldsymbol{S}$ one nucleotide at a time, and in each cycle, it adds the next nucleotide to a subset of the strands. We show that by introducing redundancy to the synthesized strands, we can significantly decrease the number of synthesis cycles required to produce the strands. We derive the maximum amount of information per synthesis cycle assuming $\boldsymbol{S}$ is an arbitrary periodic sequence. To prove our results, we exhibit new connections to cost-constrained codes.**

I. Introduction

In the past decade, DNA has emerged as a potentially viable storage technology [1], [2]. Compared to traditional storage media, DNA offers the possibility of significantly improved information density and durability [3], [4]. While much recent work has optimized many aspects of the DNA data storage pipeline [5], [6], we identify and address the goal of optimizing the synthesis process. Typically information is stored by first preprocessing the digital data and then encoding it in physical DNA molecules using a synthesis machine. Most experiments on DNA data storage use the same type of synthesis process [7], where a machine produces a large number of DNA strands in parallel using a nucleotide-by-nucleotide assembly. To append nucleotides to the strands, the synthesis machine follows a fixed supersequence of possible nucleotides and in each cycle the machine adds the nucleotide to a select subset of the DNA strands.

Within this setup, we aim to determine the maximum number of information bits that can be encoded into a set of strands, while given an overall budget of synthesis cycles. In particular, we derive the maximum *information rate*, measured in information bits per cycle, using a fixed periodic synthesis sequence. Figure 1 depicts the synthesis process of multiple strands from a fixed supersequence. This work has been published in [8].

II. Main Problem Statement

Consider a system, where digital data shall be encoded and synthesized into k DNA strands $\boldsymbol{x}_1, \ldots, \boldsymbol{x}_k \in \Sigma^*$, $\Sigma = \{\mathsf{A}, \mathsf{C}, \mathsf{G}, \mathsf{T}\}$ *in parallel*. The synthesis is performed by choosing a periodic synthesis sequence of nucleotides $\boldsymbol{S} = (S_1, S_2, \ldots) \in \Sigma^*$ and in each cycle $i = 1, 2, \ldots$, for each DNA strand $\boldsymbol{x}_j$, it is possible to either attach the symbol S_i to the strand $\boldsymbol{x}_j$ or to perform no action. Therefore, a DNA strand $\boldsymbol{x}$ can be synthesized in T cycles using the synthesis sequence $\boldsymbol{S}$, if and only if $\boldsymbol{x}$ is a subsequence of $\boldsymbol{S}_{1:T} \stackrel{\text{def}}{=} (S_1, \ldots, S_T)$. In this work we are aiming to characterize the maximum amount of information that can be synthesized in T synthesis cycles using the synthesis sequence $\boldsymbol{S}$. In other words, we are seeking to explore the maximum number of information bits that can injectively be encoded to DNA strands, such that each encoded DNA strand can be synthesized in T cycles using the synthesis sequence $\boldsymbol{S}$. Since the mapping from information words to DNA strands must naturally be injective, the number of possible information words is given by the number of sequences that can be synthesized in time T using $\boldsymbol{S}$.

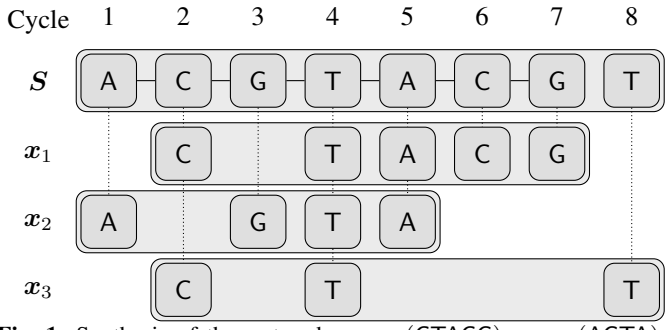

Fig. 1: Synthesis of three strands $\boldsymbol{x}_1 = (\mathsf{CTACG}), \boldsymbol{x}_2 = (\mathsf{AGTA})$, and $\boldsymbol{x}_3 = (\mathsf{CTT})$ using the synthesis sequence $\boldsymbol{S} = (\mathsf{ACGTACGT})$.

Definition 1. *We define $N(\boldsymbol{S}, T)$ to be the number of different DNA strands that can be synthesized using at most T cycles of the synthesis sequence $\boldsymbol{S}$.*

With this definition, $\log N(\boldsymbol{S}, T)$ is the number of information bits that can be synthesized in time T when using the synthesis sequence $\boldsymbol{S}$. This gives rise to our main objective.

Definition 2. *The asymptotic maximum **information rate**, measured in bits per synthesis cycle, is defined as*

$$\mathcal{R}^*(\boldsymbol{S}) = \limsup_{T \to \infty} \frac{\log(N(\boldsymbol{S}_{1:T}))}{T}.$$

III. Synthesis Codes via Constrained Codes

Interestingly, the maximum information rate is connected to the capacity of a cost-constrained system, which we will exhibit in the following. We will need the following notion of the *subsequence graph* $G(\boldsymbol{S})$ of a synthesis sequence $\boldsymbol{S}$.

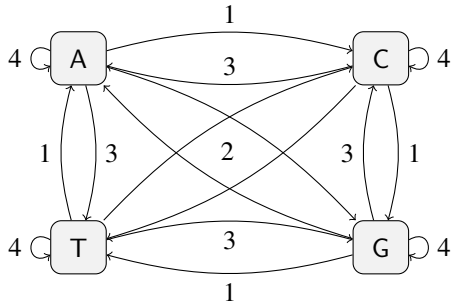

Fig. 2: Subsequence graph $G(\boldsymbol{S})$ for $\boldsymbol{S} = (\mathsf{ACGTACGT}\dots)$.

Definition 3. *The* subsequence graph $G(\boldsymbol{S})$ *of a periodic synthesis sequence* $\boldsymbol{S} = (\boldsymbol{s}\boldsymbol{s}\dots)$*, with the length-*$L$ *period* $\boldsymbol{s} \in \Sigma^L$ *is the directed graph with* L *vertices* $v_1, \dots, v_L$*. Each vertex* v_i *is labeled with the symbol* S_i*. Two vertices* v_i *and* v_j *are connected by a directed edge* e *of cost* $\tau(e) = j - i + L \cdot \mathbb{1}_{i \geq j}$*, if* $S_k \neq S_j$ *for all* $i < k < j + L \cdot \mathbb{1}_{i \geq j}$*, where* $\mathbb{1}_{i \geq j} = 1$ *if* $i \geq j$ *and* 0 *otherwise.*

Fig. 2 shows the subsequence graph $G(\boldsymbol{S})$ for $\boldsymbol{S} = (\mathsf{ACGTACGT}...)$. By construction of the graph $G(\boldsymbol{S})$ a sequence $\boldsymbol{x}$ that can be synthesized using $\boldsymbol{S}$ directly relates to a path through $G(\boldsymbol{S})$ whose traversed vertex labels generate $\boldsymbol{x}$. Further, the total weight of such a path constitutes the number of synthesis cycles that are required to synthesize $\boldsymbol{x}$. Note that by construction of the subsequence graph, the label sequences generated from all paths that start from the same vertex v_i are distinct. The subsequence graph induces a classical cost-constrained system, see e.g. [9], by the language generated by reading the labels of all paths through $G(\boldsymbol{S})$ of some maximum cost T. Defining by $C(G(\boldsymbol{S}))$ the capacity of the cost-constrained system induced by the subsequence graph $G(\boldsymbol{S})$, we can prove the following powerful correspondence.

Theorem 1. *For any periodic synthesis sequence* $\boldsymbol{S}$*, the maximum information rate is given by the capacity of the cost-constrained system* $G(\boldsymbol{S})$*, i.e.,*

$$\mathcal{R}^*(\boldsymbol{S}) = C(G(\boldsymbol{S})).$$

This universal equivalence allows to use the powerful machinery of cost-constrained systems for cost-efficient synthesis of DNA sequences. First, we can use the theory to compute the maximum information rate of a specific synthesis sequence $\boldsymbol{S}$. Second, we can use code constructions designed for cost-constrained systems [10] also to obtain codes that achieve efficient information rates.

Definition 4. *For an arbitrary periodic synthesis sequence* $\boldsymbol{S}$*, define by* $\mathbf{P}_{\boldsymbol{S}}(z)$ *the* $L \times L$ *matrix, with entries*

$$[\mathbf{P}_{\boldsymbol{S}}(z)]_{ij} = \begin{cases} 0, & \text{if there is no edge } e : i \to j \\ 2^{-z\tau(e)}, & \text{if edge } e : i \to j \text{ has cost } \tau(e) \end{cases}.$$

Using a well-known result about the capacity of constrained systems, we can deduce the following.

Corollary 2. *The maximum information rate of a periodic sequence* $\boldsymbol{S}$ *is given by the largest real solution to*

$$\det(\mathbf{I} - \mathbf{P}_{\boldsymbol{S}}(z)) = 0.$$

Example 1. *Consider the periodic synthesis sequence* $\boldsymbol{S} = (\mathsf{ACGTACGT}\dots)$*, which has been shown [8] to maximize the information rate. We directly obtain that* $\mathcal{R}^*(\boldsymbol{S}) = z_0$*, where* z_0 *is the largest real solution to* $2^{-z} + 2^{-2z} + 2^{-3z} + 2^{-4z} = 1$*, which evaluates to* $\mathcal{R}^*(\boldsymbol{S}) \approx 0.95$ bits/cycle.

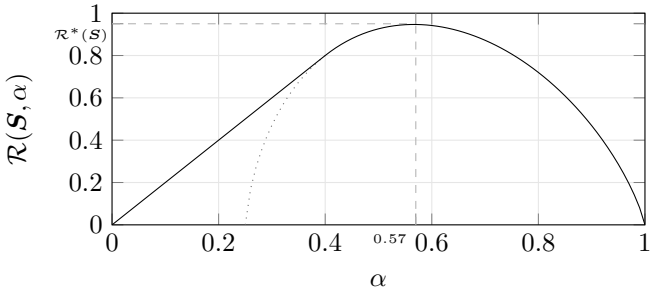

Fig. 3: Cost-constrained capacity for $\boldsymbol{S} = (\mathsf{ACGTACGT}\dots)$.

Note that our results can be applied to arbitrary periodic sequences over arbitrary alphabets.

IV. Further Results

In our publication [8] we have further been able to establish the equivalence of the synthesis cardinality $N(\boldsymbol{S}, T)$ and the number of subsequences of the synthesis sequence. This allows further to infer insights to the number of subsequences using results from cost-constrained systems. This is in particular interesting, as counting subsequences is an inherently challenging problem. Finally, we have also analyzed the case where the sequences $\boldsymbol{x}$ to be synthesized have a fixed length $\boldsymbol{x}$, corresponding to a constraint on the average cost per strand symbol. We have presented tools for computing the cost-constrained maximum information rate, $\mathcal{R}(\boldsymbol{S}, \alpha)$, where α is the inverse of the average symbol cost contraint. This is illustrated in Fig. 3 for the synthesis sequence $\boldsymbol{S} = (\mathsf{ACGTACGT}\dots)$. We also present a simple construction that achieves an information rate of 0.8 bits/cycle for $\boldsymbol{S} = (\mathsf{ACGTACGT}\dots)$. Using techniques in [10], we also constructed a fixed-to-variable length double-byte encoder with 4^8 codewords that achieves an information rate of 0.9281 bits/cycle.

References

[1] G. M. Church *et al.*, "Next-generation digital information storage in DNA," *Science*, vol. 337, no. 6102, pp. 1628–1628, Sep. 2012.

[2] N. Goldman *et al.*, "Towards practical, high-capacity, low-maintenance information storage in synthesized DNA," *Nature*, vol. 494, no. 7435, pp. 77–80, Feb. 2013.

[3] S. M. H. T. Yazdi *et al.*, "Portable and error-free DNA-based data storage," *Sci. Rep.*, vol. 7, no. 1, p. 5011, Dec. 2017.

[4] L. Organick *et al.*, "Random access in large-scale DNA data storage," *Nat. Biotechnol.*, vol. 36, no. 3, pp. 242–248, Mar. 2018.

[5] A. Lenz *et al.*, "Coding over sets for DNA storage," *IEEE Trans. Inf. Theory*, 2019.

[6] C. Rashtchian *et al.*, "Clustering billions of reads for DNA data storage," in *Proc. NeurIPS*, Long Beach, CA, Dec. 2017, p. 12.

[7] M. H. Caruthers, "The chemical synthesis of DNA/RNA: Our gift to science," *J. Biol. Chem.*, vol. 288, no. 2, pp. 1420–1427, Jan. 2013.

[8] A. Lenz *et al.*, "Coding for efficient DNA synthesis," in *Proc. Int. Symp. Inf. Theory*, Los Angeles, CA, USA, Jun. 2020, pp. 2885–2890.

[9] A. Khandekar *et al.*, "The discrete noiseless channel revisited," in *Coding, Communications, and Broadcasting*. Badlock: Research Studies Series Ltd., UK, 2000, pp. 115–137.

[10] Y. Liu *et al.*, "Rate-constrained shaping codes for structured sources," *IEEE Trans. Inf. Theory*, pp. 1–23, Apr. 2020, (early access).