

Bridging Chaos Game Representations and k -mer Frequencies of DNA Sequences

Haoze He¹, Lila Kari², and Pablo Millan Arias²

¹ École Polytechnique Fédérale de Lausanne (EPFL)

`haoze.he@epfl.ch`

² University of Waterloo

`{lila,pmillana}@uwaterloo.ca`

Abstract. This paper establishes formal mathematical foundations linking Chaos Game Representations (CGR) of DNA sequences to their underlying k -mer frequencies. We prove that the Frequency CGR (FCGR) of order k is mathematically equivalent to a discretization of CGR at resolution $2^k \times 2^k$, and its vectorization corresponds to the k -mer frequencies of the sequence. Additionally, we characterize how symmetry transformations of CGR images correspond to specific nucleotide permutations in the originating sequences. Leveraging these insights, we introduce an algorithm that generates synthetic DNA sequences from prescribed k -mer distributions by constructing Eulerian paths on De Bruijn multigraphs. This enables reconstruction of sequences matching target k -mer profiles with arbitrarily high precision, facilitating the creation of synthetic CGR images for applications such as data augmentation for machine learning-based taxonomic classification of DNA sequences. Numerical experiments validate the effectiveness of our method across both real genomic data and artificially sampled distributions. To our knowledge, this is the first comprehensive framework that unifies CGR geometry, k -mer statistics, and sequence reconstruction, offering new tools for genomic analysis and visualization. The web application implementing the reconstruction algorithm is available at <https://tinyurl.com/kmer2cgr>.

Keywords: DNA sequence, Genomic signature, Chaos Game Representation CGR, Frequency Chaos Game Representation FCGR, k -mer frequency vector, synthetic DNA.

1 Introduction

The increased availability of complete genome sequences has motivated a paradigm shift in comparative genomics, from homology-based to whole-genome analyses based on sequence composition patterns [33, 8]. The observation of different structural patterns in DNA sequences dates back to 1990, when Jeffrey applied concepts from chaotic dynamics to DNA sequences and introduced Chaos Game Representations (CGR) of DNA sequences [13]. A CGR is visualized within a unit square, with each of the four vertices labelled by one of the nucleotides

(*A*, *C*, *G*, and *T*). The plotting process follows a simple iterative procedure: the first nucleotide in the sequence is plotted at the midpoint between the center of the square and the vertex corresponding to that nucleotide. Each subsequent nucleotide is then plotted at the midpoint between the previously plotted point and the vertex representing the current nucleotide.

The appearance of interesting geometric patterns in CGRs of real DNA sequences, such as fractals and parallel lines, motivated further research in the field [14, 7, 11, 9]. Notably, Oliver *et al.* [25] computed the first discretization of CGRs. In these representations, referred to hereafter as Frequency Chaos Game Representations (FCGR), the unit square is divided into a $2^k \times 2^k$ grid, and the CGR is discretized by counting the number of points in a given cell. Each cell is associated with a particular subword of length k (k -mer), and in [25] it was suggested that the count in each cell must be equal to the frequency of the corresponding k -mer in the DNA sequence. Similarly, Hao introduced k -frames [10], a class of self-similar and self-overlapping fractals mapping each k -mer to a cell inside the unit square using the Kronecker product.

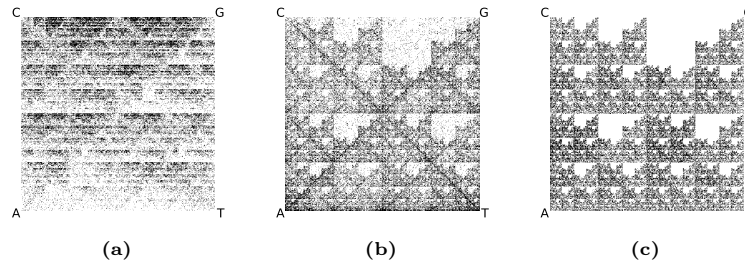


Fig. 1: (a) CGR of a 100,000 bp DNA sequence randomly extracted from the complete genome of *Pseudomonas aeruginosa* strain PAO1 (RefSeq NC_002516.2); (b) CGR of a 100,000 bp DNA sequence randomly selected from human chromosome 4 (GRCh38.p14 primary assembly, RefSeq NC_000004.12); (c) CGR of a computer-generated DNA sequence that is random in all other aspects except that the dinucleotide GC is absent.

Independent of CGR research, in 1995, Karlin and Burge introduced the notion of a *genomic signature* [17], as an umbrella term for any numerical quantity that shows greater similarity among DNA sequences of closely related organisms compared to those of more distantly related organisms. In that work, dinucleotide relative abundance profiles (DRAP) were proposed as genomic signatures, as they were effective in capturing expected variations between some species and similarities within the genome of a single species. Building on this concept, in 1999, Deschavanne *et al.* [4], characterized both CGR and FCGR as genomic signatures and showed that the variation between FCGR images of sequences along a genome was smaller than the variation between FCGR images of sequences taken from different genomes. For example, as seen in Figures 1a and 1b, the CGR of a DNA sequence from the genome of *P. aeruginosa* exhibits visual patterns that are significantly different from patterns in the CGR of a DNA sequence from human chromosome 4. Subsequently, these CGR studies lead to

the generalization of genomic signatures across various orders k , introduced in 2005 by Wang *et al.* [31] where, e.g., $k = 2$ corresponds to DRAP. In [31], it was shown that higher-order FCGRs capture sequence features that are not encoded by the DRAPs that had initially been proposed as genomic signatures.

The concepts of CGR, FCGR of order k , and k -mer frequency vector of a sequence (the latter comprising the counts of all its k -mers) collectively referred to here as *genomic signature*, have been widely used in comparative genomics as an alternative or a complement to alignment-based methods. For example, CGRs have inspired numerous alignment-free methods for taxonomic classification [27, 16, 15, 20, 28], clustering [22, 1], and phylogenetic analyses [12, 29, 19]. Similarly, distance measures between k -mer frequency profiles serve as robust proxies for evolutionary relatedness, making them the core component in similar applications [33, 30, 32, 5].

Despite the interchangeable use of these representations of DNA sequences in many applications [2, 22, 8], a formalization of their interconnections is still missing. In this work, we introduce a theoretical and algorithmic framework that rigorously links these genomic signatures. The main contributions of this paper are that it: (i) formally establishes two-way connections between a symmetry transformation of the CGR of a sequence and a morphism applied to the underlying sequence (Section 2), (ii) demonstrates the mathematical equivalence between a CGR of resolution $2^k \times 2^k$ of a DNA sequence s , and the Frequency Chaos Game Representation (FCGR) of order k of that sequence, as well as k -mer frequency vector of the sequence (Section 3), (iii) provides an algorithm and a software tool that computes a CGR and the corresponding synthetic DNA sequence, from a target k -mer frequency vector (Section 4).

1.1 Notation

Throughout this paper, Σ will denote the DNA alphabet, namely the set $\Sigma = \{A, C, G, T\}$. The cardinality of a set A will be denoted by $\text{card}(A)$. A non-empty word (string) over Σ , $w = a_1a_2\dots a_n$, $n \geq 1$, is a concatenation of letters $a_i \in \Sigma$, $1 \leq i \leq n$. The empty word is denoted by λ , by Σ^+ we denote the set of all non-empty words over Σ , and $\Sigma^* = \Sigma^+ \cup \{\lambda\}$, while for $k \geq 1$ we have that Σ^k is the set of words of length k over Σ . The length of a word w is denoted by $|w|$, and $|\lambda| = 0$. For a given $k \geq 1$, a word $w \in \Sigma^k$ will be called a k -mer. The number of occurrences of the k -mer w in the DNA sequence s will be denoted by $\text{occ}(s, w)$. For a word $w \in \Sigma^+$, we will denote by $\text{sub}_k(w)$ the set of all subwords of length k in the word w . We use Δ^{n-1} to denote the standard simplex $\{(x_1, \dots, x_n) : x_i \geq 0 \text{ for all } i = 1, \dots, n, \sum_{i=1}^n x_i = 1\}$ in $\mathbb{R}^n$.

2 CGR Symmetries and DNA Letter Permutations

This section first recalls the formal definition of the Chaos Game Representation (CGR) of a DNA sequence [13] (Definition 2) and introduces several other definitions and notations. Subsequently, Theorem 1 shows that, if a CGR is the

image of another CGR via a symmetry transformation of the square, then their originating DNA sequences are connected by a certain letter permutation, and viceversa. In addition, Theorem 2 establishes the correspondence between the k -mers avoided in each of the sequences underlying two CGRs that are obtained from one another by a symmetry transformation of the square.

Definition 1 (CGR square). *A CGR square is the square centred at the origin*

$$\{(x, y) \mid -1 < x < 1, -1 < y < 1\}$$

with corners $(-1, -1), (-1, 1), (1, 1), (1, -1)$, each labeled by the labeling function $label : \Sigma \rightarrow \mathbb{Z}^2$

$$label(A) = (-1, -1), label(C) = (-1, 1), label(G) = (1, 1), label(T) = (1, -1)$$

Note that the CGR square defined above has a side length of 2, whereas the original CGR square defined in [14] is a unit square with side length 1.

Definition 2 (CGR representation of a DNA sequence). *Let $n \geq 1$ and let $s = a_1 a_2 \dots a_n$ be a sequence of length n over the DNA alphabet Σ . The CGR representation of the sequence s is the set of points $CGR(s) = \{p_0, p_1, \dots, p_n\} \subseteq \mathbb{Q}^2$ whose coordinates are defined recursively by*

$$p_0 = (x_0, y_0) = (0, 0), \text{ and } p_i = \frac{p_{i-1} + label(a_i)}{2} \text{ for all } 1 \leq i \leq n.$$

The *dihedral group* [6] of degree 4 and order 8 is the symmetry group of a square $D_8 = \{e, r, r^2, r^3, s, sr, sr^2, sr^3\}$, and it comprises rotations, reflections across the horizontal and vertical axis, as well as reflections across the diagonals. The symmetries of the axis-aligned CGR square, centered at the origin, can be represented by 2×2 permutation matrices, acting on the plane by multiplication on column vectors of coordinates $\begin{bmatrix} x \\ y \end{bmatrix}$. The group composition operation is represented as matrix multiplication.

Definition 3. *For two words $u, w \in \Sigma^n$ and a symmetry $h \in D_8$, we will say that $CGR(u) = h \cdot CGR(w)$ if and only if, for all $1 \leq i \leq n$, we have that $q_{u,i} = h \cdot p_{w,i}$ where $q_{u,i}$ is the i th point in the generation of $CGR(u)$ and $p_{w,i}$ is the i th point in the generation of $CGR(w)$.*

Definition 4. *Let $CGR(w)$ be the CGR of a DNA sequence $w \in \Sigma^*$. The image of a $CGR(w)$ via a transformation $h \in D_8$ is defined as $\{h \cdot x \mid x \in CGR(w)\}$, the set obtained by multiplying the matrix h with each point in $CGR(w)$.*

Consider a permutation $\sigma : \Sigma \rightarrow \Sigma$ of the letters in the DNA alphabet Σ , and consider that $\sigma(\lambda) = \lambda$, where λ denotes the empty word. Such a permutation σ can be extended to a morphism $\sigma : \Sigma^* \rightarrow \Sigma^*$ by the morphism property whereby $\sigma(uv) = \sigma(u)\sigma(v)$, for all $u, v \in \Sigma^*$. We now explore the relationships between such morphisms applied to a DNA sequence and symmetry transformations in D_8 applied to its CGR.

Definition 5. Let S be the following set of permutations extended to morphisms on Σ^* :

$$S = \{(), (A, T, G, C), (A, G)(C, T), \\ (A, C, G, T), (A, C)(G, T), (C, T), (A, T)(C, G), (A, G)\}$$

In Definition 5, in the customary cyclic notation, $()$ denotes the identity permutation on $\{A, C, G, T\}$, and, e.g., (A, T, G, C) denotes a circular permutation that maps A to T , T to G , G to C , and C to A . Note also that S is a subset of the S_4 , set of all 24 permutations of $\{A, C, G, T\}$.

Recall that, [6, Section 1.2], the dihedral group D_8 is connected to a subgroup of the permutation group S_4 of all permutations of four elements, by the mapping $f : D_8 \rightarrow S_4$ defined as $f(e) = ()$, $f(r) = (A, T, G, C)$, $f(r^2) = (A, G)(C, T)$, $f(r^3) = (A, C, G, T)$, $f(s) = (A, C)(G, T)$, $f(sr) = (A, G)$, $f(sr^2) = (A, T)(C, G)$, $f(sr^3) = (C, T)$.

The following theorem proves that, if a CGR is observed to be the image of another CGR via one of the symmetry transformations in D_8 , then their originating DNA sequences are connected by one of the letter permutations in S (see Definition 5).

Theorem 1. Let $u, w \in \Sigma^n$ be two DNA sequences of length n , and let $\sigma \in S$ be one of the morphisms in S . Then $u = \sigma(w)$ if and only if $CGR(u) = f^{-1}(\sigma) \cdot CGR(w)$.

In other words, Theorem 1 states that, given two DNA sequences u and w and a letter permutation $\sigma \in S$, then $u = \sigma(w)$ iff $CGR(u)$ can be obtained from $CGR(w)$ via the symmetry $f^{-1}(\sigma)$ in D_8 . A direct consequence of this result is that the image of a CGR of a DNA sequence under some symmetry transformation $h \in D_8$ is still a CGR of another DNA sequence, i.e., the set of CGRs of DNA sequences is closed under transformations in D_8 .

Corollary 1. Given a word $u \in \Sigma^n$ and a symmetry transformation $h \in D_8$, there exists a word $w \in \Sigma^n$ such that $CGR(w) = h \cdot CGR(u)$. Constructively, this word can be computed as $w = f(h) \cdot u$, with $f : D_8 \rightarrow S_4$ as defined above.

A DNA sequence s is said to *avoid a word w* if s does not contain any occurrence of w as a subword. We now investigate the mathematical connection between k -mer avoidance in DNA sequences and the family of CGR images related by the symmetry group D_8 . This question is of interest because the CGR of a DNA sequence that avoids a k -mer w has a similar visual appearance as the CGR of a real DNA sequence in which the word w is under-represented, but not completely absent. For instance, Figures 1b and 1c illustrate that avoiding the dinucleotide GC produces the same “double-scoop” CGR pattern as that observed in CGRs of human DNA sequences (characterized by an under-representation of GC). Given the CGR of a DNA sequence that avoids a certain k -mer, and another CGR obtained from the first through a symmetry transformation in D_8 , the following theorem identifies the k -mer avoided by the underlying sequence of the second CGR.

Theorem 2. *Let $w, u \in \Sigma^n$ be two DNA words of length n , and let $\alpha, \beta \in \Sigma^k$ be two k -mers, $k < n$, such that $\text{sub}_k(w) = \Sigma^k \setminus \{\alpha\}$ and $\text{sub}_k(u) = \Sigma^k \setminus \{\beta\}$. Then, for any permutation morphism $\sigma \in S$, we have that $\text{CGR}(u) = f^{-1}(\sigma) \cdot \text{CGR}(w)$ implies $\beta = \sigma(\alpha)$.*

3 Frequency CGR, and Equivalence to k -mer Frequencies

In this section, we first recall the definition of $\text{FCGR}_k(s)$ [4], the frequency CGR of order k , of a sequence s , which is discretized version of $\text{CGR}(s)$ at resolution $2^k \times 2^k$. We formalize the discretization process of $\text{CGR}(s)$ (Proposition 1), and establish several properties of this discretized version of $\text{CGR}(s)$. Lastly, we prove that $\text{FCGR}_k(s)$ can be equivalently computed either by discretizing $\text{CGR}(s)$ at resolution $2^k \times 2^k$, or by directly counting the occurrences of all k -mers in the sequence s (Theorem 3 and its Corollary 2).

We start by describing the discretizing process of the CGR of a DNA sequence, by subdividing the CGR square (of size 2×2) into $2^k \times 2^k$ equal sub-squares of size $1/2^{k-1}$.

Definition 6 (grid cell of order k). *Given a CGR square, $k \geq 1$, and indices $0 \leq i, j \leq 2^k - 1$, the grid cell (i, j) of order k , denoted by $\text{cell}_k(i, j)$, is the region of the square defined by*

$$\text{cell}_k(i, j) = \{(x, y) : x_i - \frac{1}{2^k} < x < x_i + \frac{1}{2^k}, y_j - \frac{1}{2^k} < y < y_j + \frac{1}{2^k}\}$$

where the center of the grid cell $\text{cell}_k(i, j)$ is the point

$$(x_j, y_i) = \left(-\frac{2^k - 1}{2^k} + \frac{j}{2^{k-1}}, \frac{2^k - 1}{2^k} - \frac{i}{2^{k-1}} \right). \quad (1)$$

Note that the size of each grid cell $\text{cell}_k(i, j)$ of order k is $\frac{1}{2^{k-1}} \times \frac{1}{2^{k-1}}$. Observe also that each grid cell $\text{cell}_k(i, j)$ (without boundary) $i = 0, \dots, 2^k - 1, j = 0, \dots, 2^k - 1$, is uniquely determined by its center (x_j, y_i) , where the indices i and j are swapped so as to match the indexing convention of matrices. Lastly, note that grid cells of order k correspond to an image resolution of $2^k \times 2^k$.

We can now define the Frequency Chaos Game Representation of order k of a DNA sequence.

Definition 7 (FCGR of order k , of a sequence s). *Let $k \geq 1$ and $n \geq k$. A Frequency Chaos Game Representation of order k , of a sequence $s \in \Sigma^n$, is a matrix $\text{FCGR}_k(s) \in \mathbb{N}^{2^k \times 2^k}$ whose entries are defined as*

$$\text{FCGR}_k(s)(i, j) = \text{card}(\text{CGR}(s) \cap \text{cell}_k(i, j)) \text{ for all } 0 \leq i, j \leq 2^k - 1.$$

The $\text{FCGR}_k(s)$ matrix provides a compressed representation of the sequence s . Definition 7 can be viewed as a discretization of $\text{CGR}(s)$ into $2^k \times 2^k$ cells, each counting the number of points of $\text{CGR}(s)$ that fall inside that cell. We will show later that counting the points in a grid cell of order k is equivalent to calculating the number of occurrences of a specific k -mer in the sequence s . For that, we need to define the notion of a *CGR cell associated with a k -mer*.

Definition 8. Let $n \geq 1$ and $s \in \Sigma^n$. The last point of the CGR representation of s , $CGR(s) = \{p_0, p_1, \dots, p_n\}$, is defined as $p_{last}(s) = p_n$.

Let $p_{last}(w) = (x_w, y_w)$ be the last point in $CGR(w)$ of a word $w = a_1 a_2 \dots a_k$, where $k \geq 1$, and assume that $label(a_l) = (x_l, y_l)$ for all $1 \leq l \leq k$. By the definition of $CGR(s)$, it is easy to see that

$$x_w = \frac{\sum_{l=1}^k x_l \cdot 2^{l-1}}{2^k}, \quad y_w = \frac{\sum_{l=1}^k y_l \cdot 2^{l-1}}{2^k}. \quad (2)$$

As observed in [12], distinct sequences $s \in \Sigma^+$ have distinct last points $p_{last}(s)$ in their respective CGR representations.

Definition 9 (CGR cell associated with a k -mer). Let $k \geq 1$, let w be a k -mer in Σ^k , and let $p_{last}(w) = (x_w, y_w)$ be the last point of $CGR(w)$. The CGR cell associated with the k -mer w is

$$c(w) = \{(x, y) \mid x_w - \frac{1}{2^{|w|}} < x < x_w + \frac{1}{2^{|w|}}, y_w - \frac{1}{2^{|w|}} < y < y_w + \frac{1}{2^{|w|}}\}.$$

Figure 2a illustrates $c(ACG)$, the CGR cell associated with the 3-mer ACG. It is easy to see that, for a k -mer w , the last point $p_{last}(w)$ of $CGR(w)$ is the center of the sub-square $c(w)$. Note also that for two different k -mers w_1 and w_2 , the cells $c(w_1)$ and $c(w_2)$ do not intersect.

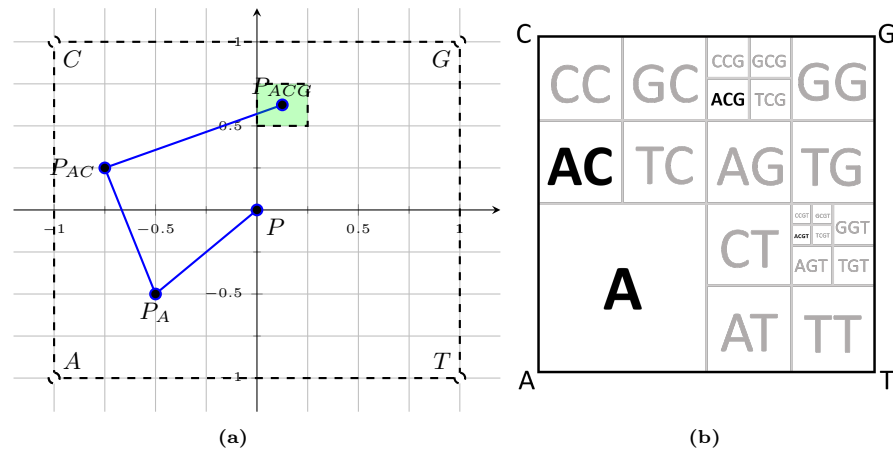


Fig. 2: (a) The cell $c(ACG)$ associated with the 3-mer ACG is marked in green. Its center P_{ACG} coincides with $p_{last}(ACG)$, the last point in the CGR representation of ACG. (b) Illustration of the hierarchical structure of CGR cells described by Lemma 2 and Proposition 3 (adapted from [20]).

The following result now establishes the correspondence between the grid cells of order k (Definition 6) and the CGR cells associated with a k -mer (Definition 9).

Proposition 1. *Let $k \geq 1$, let $w = a_1 \cdots a_k$ be a k -mer, and assume that $\text{label}(a_l) = (x_l, y_l)$ where $x_l, y_l \in \{-1, 1\}$, for all $1 \leq l \leq k$. Then $c(w) = \text{cell}_k(i, j)$, that is, the CGR cell associated with the k -mer w equals the grid cell (i, j) of order k with indices*

$$j = \frac{2^k - 1 + \sum_{l=1}^k x_l \cdot 2^{l-1}}{2}, \quad i = \frac{2^k - 1 - \sum_{l=1}^k y_l \cdot 2^{l-1}}{2}. \quad (3)$$

For a given k , Proposition 1 establishes a bijection between the set of CGR cells associated with k -mers (Definition 9) and the set of grid cells of order k (Definition 6). Observe that, in both Definition 6 and Definition 9, the respective cells do not include their boundaries. The next proposition, aided by the following lemma, proves that no point of $\text{CGR}(s)$ falls on the boundary of a cell associated with a k -mer, for any k with $1 \leq k \leq n$, where $|s| = n$ (ensuring thus that the $\text{FCGR}_k(s)$ as defined in Definition 7 does not miscount).

Lemma 1. *Let $n \geq 1, k \geq 1$, and let $u \in \Sigma^n, w \in \Sigma^k$ be two DNA sequences. Let $p_{\text{last}}(u), p_{\text{last}}(w)$ be the last points of $\text{CGR}(u), \text{CGR}(w)$ respectively. Then, the last point of $\text{CGR}(uw)$ is*

$$p_{\text{last}}(uw) = \frac{p_{\text{last}}(u)}{2^{|w|}} + p_{\text{last}}(w).$$

Proposition 2. *Let s be a sequence of length n over Σ , and let $1 \leq k \leq n$. The last point $p_{\text{last}}(s)$ of $\text{CGR}(s)$ is not on the boundary of a cell $c(w)$ associated with any k -mer w in Σ^k .*

Lemma 1 also serves as an aide to showing the hierarchical nested structure of the CGR cells associated with a k -mer w , as defined in Definition 9. The next result shows that the CGR cell associated with a given k -mer w includes all the CGR cells associated with words ending in w .

Lemma 2. *Let w_1, w_2 be words in Σ^+ . Then $c(w_1 w_2) \subseteq c(w_2)$.*

The closed CGR cell associated with a k -mer w is denoted by $\overline{c(w)}$ and is defined as the union between $c(w)$ and its boundaries (sides of the square). The next result proves that, given $n \geq 1$, the closed CGR cell associated with a k -mer w equals the union of all closed CGR cells associated with words $w'w$ with $|w'| = n$. This hierarchical structure of the closed cells associated with k -mers is illustrated in Figure 2b.

Proposition 3. *Let $k \geq 1$ and $n \geq 1$. For a given k -mer $w \in \Sigma^k$ we have that $\overline{c(w)} = \bigcup_{w' \in \Sigma^n} \overline{c(w'w)}$.*

Lemma 1 can now be used to show that the $\text{FCGR}_k(s)$ matrix defined in Definition 7 through discretizing $\text{CGR}(s)$ into grid cells of order k (i.e., at resolution $2^k \times 2^k$), can also be obtained directly by counting the number of occurrences of all k -mers in the sequence s .

Theorem 3. *Given a DNA sequence $s \in \Sigma^n$, and a k -mer $w \in \Sigma^k$, where $1 \leq k \leq n$, we have that*

$$\text{occ}(s, w) = \text{card}(\text{CGR}(s) \cap c(w))$$

where $\text{occ}(s, w)$ denotes the number of occurrences of the k -mer w in s .

Corollary 2. *Let $s \in \Sigma^n$ be a sequence, and let $1 \leq k \leq n$ be a k -mer length. Computing $\text{FCGR}_k(s)$ by discretizing $\text{CGR}(s)$ into grid cells of order k as defined in Definition 7 (i.e., at resolution $2^k \times 2^k$) is equivalent to counting the number of occurrences of k -mers in the sequence s .*

Importantly, we now observe that in [10] one finds a definition of “FCGR of order k of a sequence s ” via Kronecker products that is *different* from our Definition 7. The advantage of the concept as defined in Definition 7 is that, besides being equivalent to computing the counts of k -mers, $\text{FCGR}_k(s)$ in Definition 7 is a discretization of $\text{CGR}(s)$, at resolution $2^k \times 2^k$, and it is thus both visually similar and obtainable from $\text{CGR}(s)$. In contrast, the FCGR of order k of a sequence s as defined in [10] is not connected to $\text{CGR}(s)$, and it is visually different from $\text{CGR}(s)$. (see Appendix B.5 for a detailed comparison).

4 From k -mer Distributions to Synthetic DNA

In this section, we explore the concept of empirical k -mer distribution of a sequence s , and formalize the correspondence with its $\text{FCGR}_k(s)$. We exploit this correspondence to generate synthetic DNA sequences and their respective CGRs by tracing an Eulerian path on the De Bruijn multigraph built from a target k -mer distribution. The main result (Theorem 5) shows that, for any target distribution on the standard probability simplex satisfying the structural constraints imposed by the linear nature of DNA sequences, we can construct a synthetic sequence s whose empirical k -mer distribution matches the target with arbitrarily high precision. A computational tool implementing the algorithm is available at <https://tinyurl.com/kmer2cgr>, and is used to empirically evaluate the algorithm’s practical reconstruction accuracy through computational experiments using both real and synthetic target distributions.

Definition 10. *Given the DNA alphabet $\Sigma = \{A, C, G, T\}$ and a k -mer $w = a_1 \dots a_k \in \Sigma^k$, we define the index function $\text{idx} : \Sigma^k \rightarrow \{0, 4^k - 1\}$ as:*

$$\text{idx}(w) = \sum_{t=1}^{|w|} 4^{|w|-t} \times \xi(a_t), \quad (4)$$

where the labelling function $\xi : \Sigma \rightarrow \{0, 1, 2, 3\}$ assigns each nucleotide to an integer as follows:

$$\xi(A) = 0, \xi(C) = 1, \xi(G) = 2, \xi(T) = 3.$$

The previous definition interprets each k -mer w as a base-4 numeral via the digit assignment $\xi(\cdot)$. Note that $\text{idx}(w)$ is a bijection between Σ^k and $\{0, 4^k - 1\}$, with $k = |w|$. This bijection induces a natural ordering on Σ^k , where for $u, v \in \Sigma^k$, we write $u < v$ if and only if $\text{idx}(u) < \text{idx}(v)$. Based on the label assignment $\xi(\cdot)$, this ordering corresponds to the lexicographic order given by the nucleotide ordering $A < C < G < T$.

Definition 11 (k -mer frequency vector). *Given a DNA sequence s , its k -mer frequency vector $F_k(s)$ is a vector in $\mathbb{N}^{4^k}$ defined as:*

$$F_k(s) = (\text{occ}(s, w_0), \text{occ}(s, w_1), \dots, \text{occ}(s, w_{4^k-1}))$$

where $w_i = \text{idx}^{-1}(i)$ is the k -mer corresponding to index i under the mapping in equation 4.

We now formalize the intuition that the k -mer frequency vector $F_k(s)$ corresponds to a vectorization of the $FCGR_k(s)$ matrix and compute the mapping between the matrix coordinates and the vector positions. For that purpose, we use the $\text{label}(\cdot)$ function (Definition 1) that links the coordinates of each corner in the CGR square with each nucleotide. Recall that for each k -mer $w = a_1 \cdots a_k$, $\text{label}(a_l) = (x_l, y_l)$, with $x_l, y_l \in \{-1, 1\}$ for all $1 \leq l \leq k$. Consequently, $\text{label}^{-1}(x_l, y_l) = a_l$.

Lemma 3. *Let $k \geq 1$, and $0 \leq i, j \leq 2^k - 1$. If $i = \sum_{l=0}^{k-1} \alpha_l 2^l$ and $j = \sum_{l=0}^{k-1} \beta_l 2^l$, with $\alpha_l, \beta_l \in \{0, 1\}$ are the binary expansions of i and j , then $\text{cell}_k(i, j) = c(a_1 \cdots a_k)$ where $c(a_1 \cdots a_k)$ is the cell associated with $a_1 \cdots a_k$ constructed as*

$$a_l = \text{label}^{-1}(1 - 2\alpha_{l-1}, 2\beta_{l-1} - 1), \text{ for } 1 \leq l \leq k \quad (5)$$

Proposition 4. *Let $k \geq 1$, $n \geq 1$, $s \in \Sigma^n$ and $0 \leq i, j \leq 2^k - 1$. If $i = \sum_{l=0}^{k-1} \alpha_l 2^l$ and $j = \sum_{l=0}^{k-1} \beta_l 2^l$, with $\alpha_l, \beta_l \in \{0, 1\}$ (their binary expansions), then $FCGR_k(s)(i, j) = \text{occ}(s, w_\tau)$, where*

$$\tau = \sum_{l=0}^{k-1} 4^{k-l-1} \times \xi(\text{label}^{-1}(1 - 2\alpha_{l-1}, 2\beta_{l-1} - 1)) \quad (6)$$

Proposition 4 confirms the intuition that the k -mer frequency vector $F_k(s)$ corresponds to a vectorization of the $FCGR_k(s)$ matrix. Furthermore, we now have a connection between each word w , its corresponding point in the CGR ($p_{\text{last}}(w)$), the indices (i, j) in the FCGR matrix (Definition 7) and its position in the k -mer frequency vector given by the lexicographic order, $\text{idx}(w)$.

We will now proceed to define a probabilistic framework for comparing the k -mer composition of sequences of different lengths, with the ultimate purpose of generating synthetic DNA sequences from given k -mer frequency vectors.

Definition 12 (k -mer distribution). *For a sequence s of length n , the empirical k -mer distribution of s is defined as the probability vector*

$$\theta_s = (\theta_0^s, \theta_1^s, \dots, \theta_{4^k-1}^s) = \frac{1}{n - k + 1} \cdot F_k(s).$$

In other words, $\theta_i^s = \frac{\text{occ}(s, w_i)}{n-k+1}$ represents the normalized frequency of the k -mer w_i in s , where w_i denotes the i -th k -mer in the lexicographic order. Given $\theta \in \Delta^{4^k-1}$ and a k -mer w , we denote the component $\theta_{\text{idx}(w)}$ by θ_w .

Note that any θ_s is a point in the 4^k -dimensional probability simplex Δ^{4^k-1} and represents the empirical distribution of k -mers in s . However, not all points $\theta \in \Delta^{4^k-1}$ correspond to an empirical k -mer distribution; in particular, the points with at least one irrational coordinate do not correspond to an empirical k -mer distribution.

Besides the normalization constraint, if $\theta_s \in \Delta^{4^k-1}$ is a valid k -mer distribution, every internal $(k-1)$ -mer is shared by two overlapping k -mers. For example, every occurrence of a k -mer $w = a_1a_2 \dots a_k$ must be followed sequentially by a k -mer $w' = a_2a_3 \dots a_{k+1}$. This can be formalized by the requirement $\text{SUFFIX}(w) = \text{PREFIX}(w')$, where $\text{PREFIX}(w)$ (respectively $\text{SUFFIX}(w)$) is defined as being the word consisting of the first (respectively last) $|w| - 1$ letters in w . This condition imposes a marginalization constraint on all the k -mers, e.g.

$$\theta_{AXY} + \theta_{CXY} + \theta_{GXY} + \theta_{TXY} = \theta_{XYA} + \theta_{XYC} + \theta_{XYG} + \theta_{XYT} = \theta_{XY}.$$

Thus, for each $(k-1)$ -mer $v \in \Sigma^{k-1}$ (except edge cases in finite sequences), the sum of the frequencies of k -mers with v as a prefix must equal the sum of the frequencies of k -mers with v as a suffix. When considered across all k -mers, this condition imposes a set of additional marginal constraints that imply that θ_s does not occupy the full 4^k -dimensional simplex but rather a lower-dimensional sub-polytope [24, 18]. Formally, for every $v \in \Sigma^{k-1}$,

$$\sum_{a \in \Sigma} \theta_{va}^s = \sum_{a \in \Sigma} \theta_{av}^s = \theta_v^s. \quad (7)$$

Note that the constraints in equation 7 represent necessary conditions for a point $\theta \in \Delta^{4^k-1}$ to be the k -mer distribution of a valid DNA sequence. We will later show that these constraints are a sufficient condition for sequence reconstruction.

Given a k -mer distribution θ , our objective is to reconstruct a sequence s , whose empirical k -mer frequency vector approximates the target distribution θ as closely as possible. Then, we can compute the CGR according to Definition 2 to visualize the patterns corresponding to specific k -mer over- or under-representation. A similar problem has been studied in the field of genome assembly [23], where De Bruijn-graph-based methods have proven effective for assembling genomes from fragmented reads [26, 3]. We will show that a suitable traversal of an appropriately constructed De Bruijn graph, combined with corrections for rounding error and violation of equation 7, produces a sequence that satisfies an approximation criterion.

Definition 13. A directed multigraph M is a pair (V, E) , where V is a set of nodes, and E is a multiset of ordered pairs from $V \times V$, called edges. Loops and parallel edges are permitted.

Definition 14. Let $k \geq 2$, $n > k$, and $s \in \Sigma^n$. The De Bruijn multigraph, denoted $DE\ BRUIJN_k^+(s)$, is the directed multigraph $(V_{k-1}(s), E_k(s))$, where

$$V_{k-1}(s) = \{v \in \Sigma^{k-1} \mid v \text{ appears as a substring of } s\},$$

$$E_k(s) = \{(u, v) \in V_{k-1}(s)^2 \mid \text{there exists a } k\text{-mer } w \in \Sigma^k \text{ that appears in } s \\ \text{with } \text{PREFIX}(w) = u \text{ and } \text{SUFFIX}(w) = v\}.$$

In particular, for each $(u, v) \in V_{k-1}(s)^2$, the multiplicity of the edge (u, v) in $E_k(s)$ is exactly $\text{occ}(s, w)$.

Based on Definition 14, $DE\ BRUIJN_k^+(s)$ is completely determined by the collection of k -mers of s , so it can be constructed directly from a k -mer frequency vector without explicit knowledge of the sequence s . With this framework in place, assume a target distribution θ_s . It is then possible to rescale θ_s by $n - k + 1$ to obtain an array of pseudo counts c_s such that each component is defined by $c_w^s = (n - k + 1) \cdot \theta_w^s$. Consequently, the problem of string reconstruction reduces to finding a path in the De Bruijn multigraph (constructed from these pseudo-counts) that traverses every edge exactly once, i.e., an Eulerian path. Euler's theorem provides the necessary and sufficient conditions for the existence of such a path.

Theorem 4. (*Euler*) Let $G = (V, E)$ be a directed graph, and for any vertex $v \in V$ let $\text{in}(v)$ and $\text{out}(v)$ denote its indegree and outdegree, respectively. Then G contains an *Eulerian cycle* if and only if: (1) G is strongly connected, i.e., there exists a directed path between any pair of vertices in G , and (2) for every vertex $v \in V$, it holds that $\text{IN}(v) = \text{OUT}(v)$.

Thus, for a directed graph G to contain an Eulerian cycle, it must be strongly connected and balanced at every vertex. In our setting, this implies that the hypothetical counts c_w must satisfy the marginal constraint on the de Bruijn graph to yield a valid DNA sequence s . In particular, if we associate each $(k - 1)$ -mer v with the k -mers that have v as a prefix or suffix, then the following condition in equation 7 must hold for every vertex v , except for the unique vertices corresponding to the start and end of an Eulerian path. We now formalize these conditions in the following theorem:

Theorem 5. Let $1 < k < n \in \mathbb{Z}$, $\epsilon \in [0, 1]$, and let θ be a point in Δ^{4^k-1} satisfying the marginal constraint in equation 7 for every vertex v . If $n > \frac{2(k-1) \cdot 4^k}{\epsilon} + k - 1$, then there exists an algorithm that constructs a sequence s whose empirical k -mer distribution $\hat{\theta}_s = \frac{F_k(s)}{\|F_k(s)\|_1}$ satisfies $\|\hat{\theta}_s - \theta\|_1 \leq \epsilon$.

We validate the theoretical construction through numerical experiments, using two different approaches. First, a fragment of 100,000 base pairs (bp) is sampled from the reference genome of 100 different species, each representing a distinct kingdom of life. For each sampled fragment, empirical k -mer distributions are computed for $k \in \{2, \dots, 6\}$, and the average reconstruction error, measured by

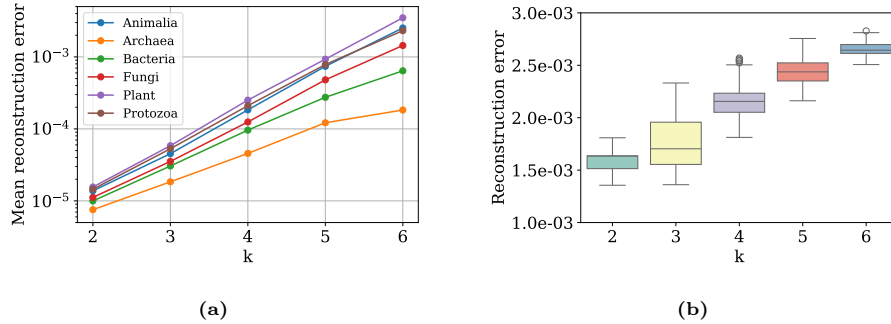


Fig. 3: (a) Reconstruction of sequences from k -mer distributions of real DNA sequences. The reconstruction is computed across different values of k , and sequences spanning species from each of the six kingdoms of life. For each kingdom, 100 genomes were selected, and a single 100,000-long DNA fragment was randomly selected from each genome. The mean error was computed for each pair $(k, \text{kingdom})$. **(b) Reconstruction of sequence from arbitrary k -mer distributions that satisfy the marginal constraints.** Each point is sampled from the standard probability simplex Δ^{4^k-1} subject to the constraints in equation 7 for different values of $k \in \{2, \dots, 6\}$. The sequence length $n_{\min}$, calculated for $\epsilon = 0.01$, ranges from 3,200 for $k = 2$, to 819,200 for $k = 6$. The reconstruction error is less than ϵ for all the values of k .

the total variation of k -mer distances, is determined for each kingdom. These results are summarized in Figure 3a. Second, we sample arbitrary distributions from the relevant sub-polytope of Δ^{4^k-1} . To this end, we employ a variation of the hit-and-run sampling procedure [21], which operates within the null-space defined by the linear constraints in equation 7. Here, the sequence length is fixed at $n_{\min} = 2 \cdot 4^k / \epsilon + k - 1$ for $\epsilon = 0.01$, and sampling is performed for $k \in \{2, \dots, 6\}$. Figure 3b illustrates the reconstruction error for different values of k over points sampled from the desired sub-polytope.

The results of our experiments indicate that the reconstruction algorithm is capable of recovering sequences whose k -mer distributions closely approximate the target distributions, regardless of whether the target is sampled from a reference DNA sequence or sampled from the simplex. A computational tool implementing the algorithm is available at <https://tinyurl.com/kmer2cgr> (see Figure 4).

5 Conclusions and Future Work

This work formalizes the connections between CGR, $FCGR_k$, and k -mer frequency vectors as genomic signatures. We note, however, that each genomic signature has distinct features and that each has proven effective in specific applications, depending on various factors such as sequence length or sequence dataset size. The aforementioned demonstrated connections between these types of genomic signatures could potentially aid in the determination of their suitability for particular practical applications. We also present an algorithm that generates synthetic DNA sequences and their corresponding CGRs from target k -mer

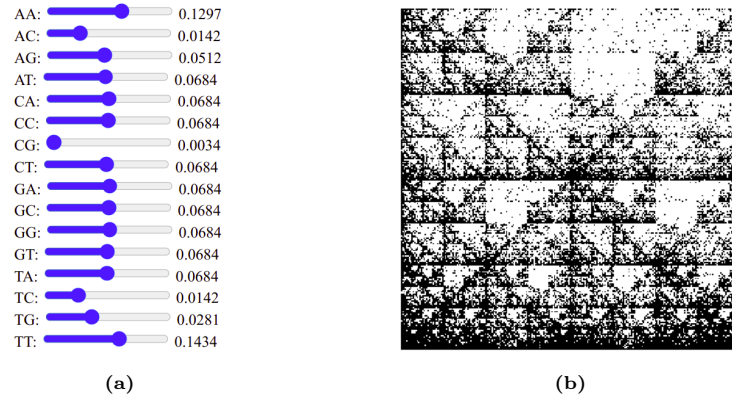


Fig. 4: Snapshot of the functionality of the computational tool: **(a)** Input: Dinucleotide ($k = 2$) distribution selected by a user via interactively adjusting 16 different sliders, each corresponding to one dinucleotide. **(b)** Output: CGR of a reconstructed DNA sequence whose k -mer distribution closely approximates the k -mer distribution in (a).

frequency vectors. This methodology opens a new avenue for synthetic DNA data generation in contrastive machine learning pipelines for taxonomic classification applications, whereby data augmentations (training samples artificially generated from existing data), are used to learn meaningful representations of the DNA sequences. Future work could explore the applicability of k -mer sampling for data augmentation and the effect of less stringent constraints, e.g., information-theoretic, on reconstruction guarantees.

Acknowledgments. The authors acknowledge the support of the Natural Sciences and Engineering Research Council of Canada (NSERC), [RGPIN-2023-03663] to L.K., and thank Niousha Sadjadi for generating Figure 2b.

References

- [1] Alipour, F., Hill, K.A., Kari, L.: CGRclust: Chaos game representation for twin contrastive clustering of unlabelled DNA sequences. *BMC Genomics* **25**, 1–17 (2024)
- [2] Avila Cartes, J. et al.: Accurate and fast clade assignment via deep learning and frequency chaos game representation. *GigaScience* **12**, giac119 (2023)
- [3] Bankevich, A. et al.: Spades: A new genome assembly algorithm and its applications to single-cell sequencing. *J. Comput. Biol.* **19**(5), 455–477 (2012)
- [4] Deschavanne, P.J. et al.: Genomic signature: characterization and classification of species assessed by chaos game representation of sequences. *Molecular Biology and Evolution* **16**(10), 1391–1399 (1999)
- [5] Dexter Dyer, B., Kahnand, M.J., LeBlanc, M.D.: Classification and regression tree (CART) analyses of genomic signatures reveal sets of tetramers that

- discriminate temperature optima of archaea and bacteria. *Archaea* **2**, 159–167 (2008)
- [6] Dummit, D.S., Foote, R.M.: Abstract algebra. John Wiley & Sons, Inc., Hoboken, NJ, third edn. (2004)
- [7] Dutta, C., Das, J.: Mathematical characterization of chaos game representation: New algorithms for nucleotide sequence analysis. *J. Molecular Biology* **228**(3), 715–719 (1992)
- [8] De la Fuente, R. et al.: Genomic signature in evolutionary biology: A review. *Biology* **12**(2) (2023)
- [9] Goldman, N.: Nucleotide, dinucleotide and trinucleotide frequencies explain patterns observed in chaos game representations of DNA sequences. *Nucleic Acids Research* **21**(10), 2487–2491 (1993)
- [10] Hao, B.L.: Fractals from genomes – exact solutions of a biology-inspired problem. *Physica A: Statistical Mechanics and its Applications* **282**(1), 225–246 (2000)
- [11] Hill, K.A., Schisler, N.J., Singh, S.M.: Chaos game representation of coding regions of human globin genes and alcohol dehydrogenase genes of phylogenetically divergent species. *J. Molecular Evolution* **35**, 261–269 (1992)
- [12] Hoang, T., Yin, C., Yau, S.S.T.: Numerical encoding of DNA sequences by chaos game representation with application in similarity comparison. *Genomics* **108**(3–4), 134–142 (2016)
- [13] Jeffrey, H.J.: Chaos game representation of gene structure. *Nucleic Acids Research* **18**(8), 2163–2170 (1990)
- [14] Jeffrey, H.: Chaos game visualization of sequences. *Computers & Graphics* **16**(1), 25–33 (1992)
- [15] Joseph, J., Sasikumar, R.: Chaos game representation for comparison of whole genomes. *BMC Bioinformatics* **7**, 1–10 (2006)
- [16] Karamichalis, R. et al.: An investigation into inter-and intragenomic variations of graphic genomic signatures. *BMC Bioinformatics* **16**(1) (2015)
- [17] Karlin, S., Burge, C.: Dinucleotide relative abundance extremes: a genomic signature. *Trends in Genetics* **11**(7), 283 – 290 (1995)
- [18] Kislyuk, A. et al.: Unsupervised statistical clustering of environmental shotgun sequences. *BMC Bioinformatics* **10**(1), 316 (2009)
- [19] Lichtblau, D.: Alignment-free genomic sequence comparison using FCGR and signal processing. *BMC Bioinformatics* **20**, 1–17 (2019)
- [20] Löchel, H.F., Heider, D.: Chaos game representation and its applications in bioinformatics. *Comput. Struct. Biotechnol. J.* **19**, 6263–6271 (2021)
- [21] Lovász, L., Vempala, S.: Hit-and-run from a corner. *SIAM J. on Computing* **35**(4), 985–1005 (2006)
- [22] Millán Arias, P.A. et al.: DeLUCS: Deep learning for unsupervised clustering of DNA sequences. *PLOS ONE* **17**(1), e0261531 (2022)
- [23] Miller, J.R., Koren, S., Sutton, G.: Assembly algorithms for next-generation sequencing data. *Genomics* **95**(6), 315–327 (2010)
- [24] Nissen, J.N. et al.: Improved metagenome binning and assembly using deep variational autoencoders. *Nature Biotechnology* **39**(5), 555–560 (2021)

- [25] Oliver, J. et al.: Entropic profiles of DNA sequences through chaos-game-derived images. *J. Theoretical Biology* **160**(4), 457–470 (1993)
- [26] Pevzner, P.A., Tang, H., Waterman, M.S.: An Eulerian path approach to DNA fragment assembly. *PNAS* **98**(17), 9748–9753 (2001)
- [27] Randhawa, G.S. et al.: Machine learning using intrinsic genomic signatures for rapid classification of novel pathogens: COVID-19 case study. *PLOS ONE* **15**(4), e0232391 (2020)
- [28] Rizzo, R. et al.: Classification experiments of DNA sequences by using a deep neural network and chaos game representation. In: *Proc. of the 17th Intl. Conference on Computer Systems and Technologies*. pp. 222–228 (2016)
- [29] Sengupta, D.C. et al.: Similarity studies of corona viruses through chaos game representation. *Computational Molecular Bioscience* **10**(3), 61 (2020)
- [30] Solis-Reyes, S. et al.: An open-source k-mer based machine learning tool for fast and accurate sub-typing of HIV-1 genomes. *PLOS ONE* **13**(11), e0206409 (2018)
- [31] Wang, Y. et al.: The spectrum of genomic signatures: from dinucleotides to chaos game representation. *Gene* **346**, 173–185 (2005)
- [32] Zielezinski, A. et al.: Benchmarking of alignment-free sequence comparison methods. *Genome Biology* **20**(1), 144 (2019)
- [33] Zielezinski, A. et al.: Alignment-free sequence comparison: benefits, applications, and tools. *Genome Biology* **18**(1), 186 (2017)

Appendix A: Technical Appendix

A.1 Proofs in Section 2

Theorem 1. *Let $u, w \in \Sigma^n$ be two DNA sequences of length n , and let $\sigma \in S$ be one of the morphisms in S . Then $u = \sigma(w)$ if and only if $CGR(u) = f^{-1}(\sigma) \cdot CGR(w)$.*

Proof. Recall the labelling function *label* that associates each letter of the DNA alphabet Σ to one of the corners of the CGR square. We start by proving the following claim.

Claim 1. Let $\sigma \in S$ be one of the letter permutations on Σ , extended to a morphism on Σ^* , and let $f : D_8 \rightarrow S$ be the mapping that associates to every symmetry of the square in D_8 one such letter permutation in S . Then, for any letters $a, b \in \Sigma$, we have that

$$a = \sigma(b) \iff label(a) = f^{-1}(\sigma) \cdot label(b).$$

Proof of Claim 1. Assume, without loss of generality, that $b = G$. This implies $label(b) = (1, 1)$. The following cases are possible for the morphism σ :

1. $\sigma = ()$, the identity permutation on Σ .
 “ $\implies$ ” $a = \sigma(b) = \sigma(G) = G$ implies $label(a) = label(G) = (1, 1)$ and $f^{-1}(\sigma) = e$. Thus, $f^{-1}(\sigma) \cdot label(b) = e \cdot label(G) = label(G) = label(a)$.
 “ $\Leftarrow$ ” $f^{-1}(\sigma) = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$, which implies $label(a) = f^{-1}(\sigma) \cdot label(b) = (1, 1)$, further implying that $a = G = \sigma(b)$.
2. $\sigma = (A \ C \ G \ T)$.
 “ $\implies$ ” $a = \sigma(G) = T$ implies $label(a) = label(T) = (1, -1)$ and $f^{-1}(\sigma) = r^3$.
 Thus,

$$f^{-1}(\sigma) \cdot label(b) = r^3 \cdot (1, 1) = \begin{bmatrix} 0 & 1 \\ -1 & 0 \end{bmatrix} \cdot (1, 1) = (1, -1) = label(a).$$

“ $\Leftarrow$ ” $f^{-1}(\sigma) = \begin{bmatrix} 0 & 1 \\ -1 & 0 \end{bmatrix}$, which implies $label(a) = f^{-1}(\sigma) \cdot label(b) = (1, -1)$, which further implies that $a = T = \sigma(b)$.

3. $\sigma = (A \ G)(C \ T)$.
 “ $\implies$ ” $a = \sigma(G) = A$ implies $label(a) = label(A) = (-1, -1)$ and $f^{-1}(\sigma) = r^2$.
 Thus,

$$f^{-1}(\sigma) \cdot label(b) = r^2 \cdot (1, 1) = \begin{bmatrix} -1 & 0 \\ 0 & -1 \end{bmatrix} \cdot (1, 1) = (-1, -1) = label(a).$$

“ $\Leftarrow$ ” $f^{-1}(\sigma) = \begin{bmatrix} -1 & 0 \\ 0 & -1 \end{bmatrix}$ implies $label(a) = f^{-1}(\sigma) \cdot label(b) = (-1, -1)$, which further implies that $a = A = \sigma(b)$.

4. $\sigma = (A \ T \ G \ C)$.
 “ $\implies$ ” $a = \sigma(G) = C$ implies $label(a) = label(C) = (-1, 1)$ and $f^{-1}(\sigma) = r$.
 Thus,

$$f^{-1}(\sigma) \cdot label(b) = r \cdot (1, 1) = \begin{bmatrix} 0 & -1 \\ 1 & 0 \end{bmatrix} \cdot (1, 1) = (-1, 1) = label(a).$$

“ $\Leftarrow$ ” $f^{-1}(\sigma) = \begin{bmatrix} 0 & -1 \\ 1 & 0 \end{bmatrix}$ implies $l(a) = f^{-1}(\sigma) \cdot label(b) = (-1, 1)$, which further implies that $a = C = \sigma(b)$.

5. $\sigma = (A \ C)(G \ T)$.
 “ $\implies$ ” $a = \sigma(G) = T$ implies $label(a) = label(T) = (1, -1)$ and $f^{-1}(\sigma) = s$.
 Thus,

$$f^{-1}(\sigma) \cdot label(b) = s \cdot (1, 1) = \begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix} \cdot (1, 1) = (1, -1) = label(a).$$

“ $\Leftarrow$ ” $f^{-1}(\sigma) = \begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix}$ implies $label(a) = f^{-1}(\sigma) \cdot label(b) = (1, -1)$, which further implies that $a = T = \sigma(b)$.

6. $\sigma = (A \ T)(C \ G)$.

" $\implies$ " $a = \sigma(G) = C$ implies $\text{label}(a) = \text{label}(C) = (-1, 1)$ and $f^{-1}(\sigma) = sr^2$.

Thus,

$$f^{-1}(\sigma) \cdot \text{label}(b) = sr^2 \cdot (1, 1) = \begin{bmatrix} -1 & 0 \\ 0 & 1 \end{bmatrix} \cdot (1, 1) = (-1, 1) = \text{label}(a).$$

" $\Leftarrow$ " $f^{-1}(\sigma) = \begin{bmatrix} -1 & 0 \\ 0 & 1 \end{bmatrix}$ implies $\text{label}(a) = f^{-1}(\sigma) \cdot \text{label}(b) = (-1, 1)$, which further implies that $a = C = \sigma(b)$.

7. $\sigma = (C \ T)$.

" $\implies$ " $a = \sigma(G) = G$ implies $\text{label}(a) = \text{label}(G) = (1, 1)$ and $f^{-1}(\sigma) = sr^3$.

Thus,

$$f^{-1}(\sigma) \cdot \text{label}(b) = sr^3 \cdot (1, 1) = \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix} \cdot (1, 1) = (1, 1) = \text{label}(a).$$

" $\Leftarrow$ " $f^{-1}(\sigma) = \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}$ implies $\text{label}(a) = f^{-1}(\sigma) \cdot \text{label}(b) = (1, 1)$, which further implies that $a = G = \sigma(b)$.

8. $\sigma = (A \ G)$.

" $\implies$ " $a = \sigma(G) = A$ implies $\text{label}(a) = \text{label}(A) = (-1, -1)$ and $f^{-1}(\sigma) = sr$.

Thus,

$$f^{-1}(\sigma) \cdot \text{label}(b) = sr \cdot (1, 1) = \begin{bmatrix} 0 & -1 \\ -1 & 0 \end{bmatrix} \cdot (1, 1) = (-1, -1) = \text{label}(a).$$

" $\Leftarrow$ " $f^{-1}(\sigma) = \begin{bmatrix} 0 & -1 \\ -1 & 0 \end{bmatrix}$ implies $\text{label}(a) = f^{-1}(\sigma) \cdot \text{label}(b) = (-1, -1)$, which further implies that $a = A = \sigma(b)$.

The cases when $b = T$, $b = A$, and $b = C$ can be proved similarly, and complete proof of Claim 1.

The proof is now by induction on n , the length of the input strings u and w . Assume $w = b_1 b_2 \dots b_n$ and $u = a_1 a_2 \dots a_n$, where $n \geq 1$ and a_i, b_i are letters in Σ for all $1 \leq i \leq n$. Recall that $CGR(u) = f^{-1}(\sigma) \cdot CGR(w)$ iff, for all $1 \leq i \leq n$, we have $q_i = f^{-1}(\sigma) \cdot p_i$ where q_i is the i -th point in the generation of $CGR(u)$, and p_i is the i -th point in the generation of $CGR(w)$.

Base Case. $n = 1$. Then $w = b \in \Sigma$, and $u = a \in \Sigma$.

" $\implies$ " Assume $a = \sigma(b)$. By the recursive definition of CGR , $CGR(a)$ consists of a single point $q_1 = \frac{\text{label}(a)}{2}$, and $CGR(b)$ consists of a single point $p_1 = \frac{\text{label}(b)}{2}$. By Claim 1, $\text{label}(a) = f^{-1}(\sigma) \cdot \text{label}(b)$. Thus,

$$q_1 = \frac{\text{label}(a)}{2} = \frac{f^{-1}(\sigma) \cdot \text{label}(b)}{2} = f^{-1}(\sigma) \cdot p_1$$

" $\Leftarrow$ " Assume $CGR(a) = f^{-1}(\sigma) \cdot CGR(b)$. Since $CGR(a)$ consists of one point $q_1 = \frac{label(a)}{2}$, and $CGR(b)$ consists of one point $p_1 = \frac{label(b)}{2}$, this implies

$$q_1 = \frac{label(a)}{2} = f^{-1}(\sigma) \cdot p_1 = f^{-1}(\sigma) \cdot \frac{label(b)}{2},$$

further implying $label(a) = f^{-1}(\sigma) \cdot label(b)$. By Claim 1 this now implies that $a = \sigma(b)$.

Inductive step. Assume that the iff statement of Theorem 1 holds for words of length $(m - 1)$, with $m \geq 2$ (Inductive Hypothesis, I.H.). Let w and u be two words of length m , that is, $w = b_1 b_2 \dots b_{m-1} b_m$ and $u = a_1 a_2 \dots a_{m-1} a_m$, where $a_i, b_i \in \Delta$ for all $1 \leq i \leq m$.

" $\Rightarrow$ " Assume $u = \sigma(w)$. By the definition of morphism, we have that $u = \sigma(w)$ iff $a_i = \sigma(b_i)$, for all $1 \leq i \leq m$.

Denote

$$w' = b_1 b_2 \dots b_{m-1}, \quad u' = a_1 a_2 \dots a_{m-1}.$$

By (I.H.), since $u' = \sigma(w')$, we have that $q_i = f^{-1}(\sigma) \cdot p_i$, for all $1 \leq i \leq m - 1$. By the definition of CGR, the m th point of $CGR(u)$ is q_m , and the m th point of $CGR(w)$ is p_m , where

$$q_m = \frac{q_{m-1} + label(a_m)}{2}, \text{ and } p_m = \frac{p_{m-1} + label(b_m)}{2}.$$

Since $a_m = \sigma(b_m)$, by Claim 1 we have that $label(a_m) = f^{-1}(\sigma) \cdot label(b_m)$. Thus, by I.H. and Claim 1, we have

$$\begin{aligned} q_m &= \frac{q_{m-1} + label(a_m)}{2} = \frac{f^{-1}(\sigma) \cdot p_{m-1} + f^{-1}(\sigma) \cdot label(b_m)}{2} = \\ &= f^{-1}(\sigma) \cdot \frac{p_{m-1} + label(b_m)}{2} = f^{-1}(\sigma) \cdot p_m. \end{aligned}$$

Since the equality $q_i = f^{-1}(\sigma) \cdot p_i$ holds now for all $1 \leq i \leq m$, it follows that $CGR(u) = f^{-1}(\sigma) \cdot CGR(w)$.

" $\Leftarrow$ " Assume that $CGR(u) = f^{-1}(\sigma) \cdot CGR(w)$. This implies that, for all $1 \leq i \leq m$ we have that $q_i = f^{-1}(\sigma) \cdot p_i$.

The recursive definition of $CGR(w)$ and the previous observation imply that:

$$q_m = f^{-1}(\sigma) \cdot p_m = \frac{q_{m-1} + label(a_m)}{2}, \quad (8)$$

In parallel, the recursive definition of CGR and the fact that $f^{-1}(\sigma)$ is a transformation in $\mathbb{R}^2$ imply that:

$$f^{-1}(\sigma) \cdot p_m = f^{-1}(\sigma) \cdot \left(\frac{p_{m-1} + label(b_m)}{2} \right) \quad (9)$$

By transitivity of equality applied to equations 8 and 9 we get that:

$$\frac{q_{m-1} + \text{label}(a_m)}{2} = f^{-1}(\sigma) \cdot \left(\frac{p_{m-1} + \text{label}(b_m)}{2} \right)$$

This can be further simplified using the distributivity property of matrix multiplication:

$$\begin{aligned} q_{m-1} + \text{label}(a_m) &= f^{-1}(\sigma) \cdot p_{m-1} + f^{-1}(\sigma) \cdot \text{label}(b_m) \\ \text{label}(a_m) &= f^{-1}(\sigma) \cdot \text{label}(b_m) \end{aligned}$$

which implies that $a_m = \sigma(b_m)$ by Claim 1.

By the (I.H.), since $CGR(u') = f^{-1}(\sigma) \cdot CGR(w')$ we have that $u' = \sigma(w')$. Thus, by the definition of a morphism and the previously derived relation $a_m = \sigma(b_m)$, we obtain that $\sigma(w) = \sigma(w')\sigma(b_m) = u'a_m = u$.

Corollary 1. *Given a word $u \in \Sigma^n$ and a symmetry transformation $h \in D_8$, there exists a word $w \in \Sigma^n$ such that $CGR(w) = h \cdot CGR(u)$. Constructively, this word can be computed as $w = f(h) \cdot u$, with $f : D_8 \rightarrow S_4$ as defined above.*

Proof. It directly follows from Theorem 1.

Theorem 2. *Let $w, u \in \Sigma^n$ be two DNA words of length n , and let $\alpha, \beta \in \Sigma^k$ be two k -mers, $k < n$, such that $\text{sub}_k(w) = \Sigma^k \setminus \{\alpha\}$ and $\text{sub}_k(u) = \Sigma^k \setminus \{\beta\}$. Then, for any permutation morphism $\sigma \in S$, we have that $CGR(u) = f^{-1}(\sigma) \cdot CGR(w)$ implies $\beta = \sigma(\alpha)$.*

Proof. Let $\sigma \in S$ be one of the permutation morphisms in S .

“ $\implies$ ” Since $CGR(u) = f^{-1}(\sigma) \cdot CGR(w)$, by Theorem 1 we have $u = \sigma(w)$.

Since σ is a morphism, it follows that $\text{sub}_k(u) = \text{sub}_k(f(w)) = f(\text{sub}_k(w))$. Now let us show $\beta = \sigma(\alpha)$. We have that

$$\text{sub}_k(u) = \Sigma^k \setminus \{\beta\} = \text{sub}_k(\sigma(w)) = \sigma(\text{sub}_k(w)) = \sigma(\Sigma^k \setminus \{\alpha\})$$

Since $\sigma \in S$ is a permutation of the letters of Σ , we have that $\sigma(\Sigma^k) = \Sigma^k$, and $\sigma(\Sigma^k \setminus \{\alpha\}) = \Sigma^k \setminus \{\sigma(\alpha)\}$. Together with the above equalities, this implies $\beta = \sigma(\alpha)$.

A.2 Proofs in Section 3

Proposition 1. *Let $k \geq 1$, let $w = a_1 \cdots a_k$ be a k -mer, and assume that $\text{label}(a_l) = (x_l, y_l)$ where $x_l, y_l \in \{-1, 1\}$, for all $1 \leq l \leq k$. Then $c(w) = \text{cell}_k(i, j)$, that is, the CGR cell associated with the k -mer w equals the grid cell (i, j) of order k with indices*

$$j = \frac{2^k - 1 + \sum_{l=1}^k x_l \cdot 2^{l-1}}{2}, \quad i = \frac{2^k - 1 - \sum_{l=1}^k y_l \cdot 2^{l-1}}{2}. \quad (10)$$

Proof. We start by first observing that, for a given value of k and k -mer w , the size of the CGR cell $c(w)$ is the same as the size of $cell_k(i, j)$, a grid cell of order k , where $0 \leq i, j \leq 2^{k-1}$. Therefore, for a given k and k -mer w , what remains is to find indices $0 \leq i, j \leq 2^{k-1}$ such that the center (x_j, y_i) of the grid cell $cell_k(i, j)$ coincides with the center $p_{last}(w) = (x_w, y_w)$ of $c(w)$, if such indices exist.

Let $w = a_1 a_2 \cdots a_k$ and let $p_{last}(w)$ be the last point in $CGR(w)$.

Recall that $p_{last}(w) = (x_w, y_w)$ in equation 2 is the center of the $c(w)$. For any indexing (i, j) , the center (x_j, y_i) of the $cell_k(i, j)$ is defined by equation 1. Letting $(x_w, y_w) = (x_j, y_i)$ leads to

$$x_w = \frac{\sum_{l=1}^k x_l \cdot 2^{l-1}}{2^k} = x_j = -\frac{2^k - 1}{2^k} + \frac{j}{2^{k-1}},$$

$$y_w = \frac{\sum_{l=1}^k y_l \cdot 2^{l-1}}{2^k} = y_i = \frac{2^k - 1}{2^k} - \frac{i}{2^{k-1}}.$$

Solving for j and i gives

$$j = \frac{2^k - 1 + \sum_{l=1}^k x_l \cdot 2^{l-1}}{2}, \quad i = \frac{2^k - 1 - \sum_{l=1}^k y_l \cdot 2^{l-1}}{2},$$

which is equation 10 as required. Since $x_l \in \{-1, 1\}$ for all $l = 1, \dots, k$, we have that j is an integer. Furthermore, the minimum value of j is achieved when $x_l = -1$ for all $l = 1, \dots, k$, and its maximum is achieved when $x_l = 1$ for all $l = 1, \dots, k$, which leads to $j \in \{0, \dots, 2^k - 1\}$. By a similar reasoning, $i \in \{0, \dots, 2^k - 1\}$. Thus, for these values of i and j , we have that $(x_w, y_w) = (x_j, y_i)$. Therefore, the grid cell $cell_k(i, j)$ equals the CGR cell $c(w)$ associated with the k -mer w , as having the same size and the same center.

Lemma 1. *Let $n \geq 1, k \geq 1$, and let $u \in \Sigma^n, w \in \Sigma^k$ be two DNA sequences. Let $p_{last}(u), p_{last}(w)$ be the last points of $CGR(u), CGR(w)$ respectively. Then, the last point of $CGR(uw)$ is*

$$p_{last}(uw) = \frac{p_{last}(u)}{2^{|w|}} + p_{last}(w).$$

Proof. By induction on k .

Base Case. Let $k = 1$, that is, $w = a$, where $a \in \Sigma$. It follows from the definition of CGR that

$$p_{last}(ua) = \frac{p_{last}(u)}{2} + \frac{label(a)}{2} = \frac{p_{last}(u)}{2} + p_{last}(a).$$

Inductive step. Assume that the statement holds for an arbitrary $k \geq 1$, and let $w \in \Sigma^{k+1}$. Then $w = w' a_{k+1}$ where $w' \in \Sigma^k, a_{k+1} \in \Sigma$. By I.H., for all $u \in \Sigma^{n_1}, n_1 \geq 0$, we have that the last point $p_{last}(uw')$ of $CGR(uw')$ is

$$p_{last}(uw') = \frac{p_{last}(u)}{2^k} + p_{last}(w').$$

The last point $p_{last}(uw)$ of $CGR(uw)$ is

$$\begin{aligned}
 p_{last}(uw) &= p_{last}(uw'a_{k+1}) = \frac{p_{last}(uw')}{2} + \frac{label(a_{k+1})}{2} && \text{(by definition of CGR)} \\
 &= \frac{p_{last}(u)}{2^{k+1}} + \frac{p_{last}(w')}{2} + \frac{label(a_{k+1})}{2} && \text{(by I.H.)} \\
 &= \frac{p_{last}(u)}{2^{k+1}} + p_{last}(w) && \text{(by definition of CGR)}
 \end{aligned}$$

Proposition 2. *Let s be a sequence of length n over Σ , and let $1 \leq k \leq n$. The last point $p_{last}(s)$ of $CGR(s)$ is not on the boundary of a cell $c(w)$ associated with any k -mer w in Σ^k .*

Proof. First note that any point in $CGR(s)$ of a sequence s is strictly inside the CGR square. Indeed, for any $s' \in \Sigma^n$, the x -coordinate of $p_{last}(s') = (x_{s'}, y_{s'})$, the last point of $CGR(s')$, is by definition of the form

$$x_{s'} = \frac{\pm 1}{2} + \frac{\pm 1}{2^2} + \frac{\pm 1}{2^3} + \cdots + \frac{\pm 1}{2^n}.$$

It follows that the absolute values of $x_{s'}$ and $y_{s'}$ are both strictly less than 1, therefore, $p_{last}(s')$ is strictly inside the CGR square.

Let $s = s'w$ where w is of length k . Then, by Lemma 1,

$$p_{last}(s) = \frac{p_{last}(s')}{2^k} + p_{last}(w).$$

The proof is completed by the fact that the absolute values of $x_{s'}$ and $y_{s'}$ are strictly less than 1.

Lemma 2. *Let w_1, w_2 be words in Σ^+ . Then $c(w_1w_2) \subseteq c(w_2)$.*

Proof. Let $(x, y) \in c(w_1w_2)$, let $p_{last}(w_1w_2) = (x_{w_1w_2}, y_{w_1w_2})$, $p_{last}(w_1) = (x_{w_1}, y_{w_1})$ and let $p_{last}(w_2) = (x_{w_2}, y_{w_2})$. Without loss of generality, we consider the x -coordinates of all points. By the definition of $c(w_1w_2)$, we have

$$-1/2^{|w_1w_2|} < x - x_{w_1w_2} < 1/2^{|w_1w_2|}.$$

By Lemma 1, we have that

$$x_{w_1w_2} = \frac{x_{w_1}}{2^{|w_2|}} + x_{w_2}.$$

By replacing $x_{w_1w_2}$ into the previous inequality we obtain

$$-1/2^{|w_1w_2|} + x_{w_1}/2^{|w_2|} < x - x_{w_2} < 1/2^{|w_1w_2|} + x_{w_1}/2^{|w_2|}.$$

Since $x_{w_1} \leq \sum_{l=1}^{|w_1|} 2^{l-1}/2^{|w_1|} = (2^{|w_1|} - 1)/2^{|w_1|}$, we have $x - x_{w_2} < 1/2^{|w_1w_2|} + x_{w_1}/2^{|w_2|} \leq 1/2^{|w_2|}$. Similarly, we have $-1/2^{|w_2|} \leq -1/2^{|w_1w_2|} + x_{w_1}/2^{|w_2|} < x - x_{w_2}$. Therefore, $-1/2^{|w_2|} < x - x_{w_2} < 1/2^{|w_2|}$.

Since an analogous argument can be made for the y -coordinate, it follows that the arbitrary point (x, y) from $c(w_1w_2)$ is situated in $c(w_2)$.

Proposition 3. *Let $k \geq 1$ and $n \geq 1$. For a given k -mer $w \in \Sigma^k$ we have that $\overline{c(w)} = \bigcup_{w' \in \Sigma^n} \overline{c(w'w)}$.*

Proof. By strong induction on n .

Base Case. For $n = 1$, we have to prove that

$$\overline{c(w)} = \bigcup_{a \in \Sigma} \overline{c(aw)}.$$

Let $p_{last}(w) = (x_w, y_w)$ and let $label(a) = (x_a, y_a)$. By definition of $\overline{c(w)}$,

$$\overline{c(w)} = \{(x, y) \mid -1/2^{|w|} \leq x - x_w \leq 1/2^{|w|}, -1/2^{|w|} \leq y - y_w \leq 1/2^{|w|}\}.$$

By Lemma 1 and the definition of $\overline{c(aw)}$, we have

$$\overline{c(aw)} = \{(x, y) \mid \frac{-1 + x_a}{2^{|w|+1}} \leq x - x_w \leq \frac{1 + x_a}{2^{|w|+1}}, \frac{-1 + y_a}{2^{|w|+1}} \leq y - y_w \leq \frac{1 + y_a}{2^{|w|+1}}\}.$$

Since $x_a, y_a \in \{-1, 1\}$, it suffices to consider all possible pairs in $\{-1, 1\} \times \{-1, 1\}$. Substituting each of these pairs into the preceding equation and taking their union, leads to the desired range for points in $\overline{c(w)}$, from which it follows that $\overline{c(w)} = \bigcup_{a \in \Sigma} \overline{c(aw)}$.

Inductive Step. Assume now that the statement holds for all $1 \leq i \leq j$, and show that it holds for $j + 1$. Note that

$$\bigcup_{w' \in \Sigma^{j+1}} \overline{c(w'w)} = \bigcup_{a \in \Sigma} \bigcup_{w'' \in \Sigma^j} \overline{c(w''aw)}.$$

Since by the induction hypothesis, we have $\bigcup_{w'' \in \Sigma^j} \overline{c(w''aw)} = \overline{c(aw)}$, and also that $\overline{c(w)} = \bigcup_{a \in \Sigma} \overline{c(aw)}$, the proof of the inductive step is complete.

Theorem 3. *Given a DNA sequence $s \in \Sigma^n$, and a k -mer $w \in \Sigma^k$, where $1 \leq k \leq n$, we have that*

$$occ(s, w) = \text{card}(CGR(s) \cap c(w))$$

where $occ(s, w)$ denotes the number of occurrences of the k -mer w in s .

Proof. Let $k \geq 1$ and let $w \in \Sigma^k$ be a k -mer. We first prove that $occ(s, w) \leq \text{card}(CGR(s) \cap c(w))$.

Consider an arbitrary occurrence of w in s . Then $s = s_1ws_2$ for some $s_1, s_2 \in \Sigma^*$, and let us denote $s' = s_1w$.

If $s_1 = \lambda$, then $p_{last}(s') = p_{last}(w)$ is the center of $c(w)$ and thus is located inside $c(w)$.

If $s_1 \in \Sigma^+$, by applying Lemma 1 to s' , we have that the last point $p_{last}(s')$ of $CGR(s')$ is

$$p_{last}(s') = p_{last}(s_1w) = \frac{p_{last}(s_1)}{2^k} + p_{last}(w).$$

From the proof of Proposition 2, we have that $p_{last}(s_1)$ is located inside the CGR square, that is, $p_{last}(s_1) \in \{(x, y) \mid -1 < x < 1, -1 < y < 1\}$. This implies that

$$p_{last}(s') = p_{last}(s_1 w) \in \{p_{last}(w) + (x, y) \mid -1/2^k < x < 1/2^k, -1/2^k < y < 1/2^k\}.$$

By comparing this with Definition 9 of $c(w)$, Definition 9, it follows that $p_{last}(s')$ is located inside $c(w)$.

Since we reached the same conclusion in both possible cases, it follows that any occurrence of w in $s = s_1 w s_2$ corresponds to a CGR point $p_{last}(s_1 w)$ located inside $c(w)$. From this, It follows that $occ(s, w) \leq \text{card}(CGR(s) \cap c(w))$.

We now prove that $occ(s, w) \geq \text{card}(CGR(s) \cap c(w))$.

Consider a point of $CGR(s)$ that is located inside $c(w)$. This point is $p_{last}(s')$, the last point of $CGR(s')$, for some prefix s' of s .

We first show that $|s'| \geq k$. Assume, for the sake of contradiction, that $|s'| = k' < k$.

If s' is not a proper suffix of w , then for any decomposition $w = w_1 w_2$ with $|w_2| = |s'|$, we have that $w_2 \neq s'$. Then $p_{last}(s')$ will be located inside $c(s')$, which does not overlap with $c(w_2)$, hence $p_{last}(s')$ is not located inside $c(w_2)$. By Lemma 2, $c(w) = c(w_1 w_2) \subseteq c(w_2)$. This implies that $p_{last}(s')$ is not located inside $c(w)$ - a contradiction. Thus, it must be the case that if $s' = k' < k$, then s' is a proper suffix of w , that is, $s' = w_2$ with $w = w_1 w_2$, and $w_1 \neq \lambda$.

However, this leads to a contradiction, as follows. From Lemma 1, we have $p_{last}(w_2) - p_{last}(w_1 w_2) = \frac{p_{last}(w_1)}{2^{k'}}$. For $p_{last}(s)$ to be located inside $c(w)$, we would need $2^{|w_1|} \cdot p_{last}(w_1) \in (-1, 1) \times (-1, 1)$, which cannot happen for any $w_1 \neq \lambda$. Indeed, for the aforementioned relation to hold, the x -coordinate of $p_{last}(w_1) = (x_{w_1}, y_{w_1})$ should satisfy

$$2^{|w_1|} \cdot x_{w_1} = \sum_{l=1}^{|w_1|} x_l \cdot 2^{l-1}.$$

If $x_{w_1} = 1$, then we have that $2^{|w_1|} \cdot x_{w_1} \geq 1$, and if $x_{w_1} = -1$, then we have that $2^{|w_1|} \cdot x_{w_1} \leq -1$. In both situations, the sum on the right-hand side cannot be in the interval $(-1, 1)$, while the quantity on the left-hand side is in $(-1, 1)$ - a contradiction.

Thus, it follows that $k \leq |s'| \leq n$, where s' is a prefix of s . Let $s' = s_1 w'$ where $s_1 \in \Sigma^*$, $w' \in \Sigma^k$. If $w' \neq w$, using the same argument as before, one can show that $p_{last}(s')$ will be in $c(w')$, which contradicts the fact that cells associated with different k -mers are non-overlapping. Therefore $s' = s_1 w$, which means that the point in $CGR(s)$ that we considered, $p_{last}(s')$, corresponds to the specific occurrence of w in s . Since distinct s' result in distinct $p_{last}(s')$, each point in $CGR(s)$ located inside $c(w)$ corresponds to a distinct occurrence of w in s . This further implies that $\text{card}(CGR(s) \cap c(w)) \leq occ(s, w)$, which completes the proof.

Corollary 2. *Let $s \in \Sigma^n$ be a sequence, and let $1 \leq k \leq n$ be a k -mer length. Computing $FCGR_k(s)$ by discretizing $CGR(s)$ into grid cells of order k as defined*

in Definition 7 (i.e., at resolution $2^k \times 2^k$) is equivalent to counting the number of occurrences of k -mers in the sequence s .

Proof. It follows from Theorem 3, Proposition 1 and the bijection that it establishes.

A.3 Proofs in Section 4

Lemma 3. Let $k \geq 1$, and $0 \leq i, j \leq 2^k - 1$. If $i = \sum_{l=0}^{k-1} \alpha_l 2^l$ and $j = \sum_{l=0}^{k-1} \beta_l 2^l$, with $\alpha_l, \beta_l \in \{0, 1\}$ are the binary expansions of i and j , then $\text{cell}_k(i, j) = c(a_1 \dots a_k)$ where $c(a_1 \dots a_k)$ is the cell associated with $a_1 \dots a_k$ constructed as

$$a_l = \text{label}^{-1}(1 - 2\alpha_{l-1}, 2\beta_{l-1} - 1), \text{ for } 1 \leq l \leq k \quad (11)$$

Proof. From Proposition 1, we know that

$$j = \frac{2^k + 1 + \sum_{l=1}^k x_l \cdot 2^{l-1}}{2}, \quad i = \frac{2^k + 1 - \sum_{l=1}^k y_l \cdot 2^{l-1}}{2},$$

which can be rewritten as:

$$j = \frac{1}{2} \left(\sum_{l=1}^k 2^{l-1} \cdot (1 + x_l) + 1 \right), \quad i = \frac{1}{2} \left(\sum_{l=1}^k 2^{l-1} \cdot (1 - y_l) + 1 \right).$$

Now, considering the binary expansion of the indices, we get the following equations:

$$j = \sum_{l=0}^{k-1} \beta_l \cdot 2^l = \sum_{l=1}^k 2^{l-1} \cdot \frac{(1 + x_l)}{2}, \quad i = \sum_{l=0}^{k-1} \alpha_l \cdot 2^l = \sum_{l=1}^k 2^{l-1} \cdot \frac{(1 - y_l)}{2},$$

which we solve for x_l and y_l to obtain the correspondence with the coefficients in the binary expansion:

$$x_l = 1 - 2\alpha_{l-1}, \quad y_l = 2\beta_{l-1} - 1 \quad (12)$$

Finally, by Definition 1 we get that: $a_l = \text{label}^{-1}(x_l, y_l) = \text{label}^{-1}(1 - 2\alpha_{l-1}, 2\beta_{l-1} - 1)$

Proposition 4. Let $k \geq 1$, $n \geq 1$, $s \in \Sigma^n$ and $0 \leq i, j \leq 2^k - 1$. If $i = \sum_{l=0}^{k-1} \alpha_l 2^l$ and $j = \sum_{l=0}^{k-1} \beta_l 2^l$, with $\alpha_l, \beta_l \in \{0, 1\}$ (their binary expansions), then $\text{FCGR}_k(s)(i, j) = \text{occ}(s, w_\tau)$, where

$$\tau = \sum_{l=0}^{k-1} 4^{k-l-1} \times \xi(\text{label}^{-1}(1 - 2\alpha_{l-1}, 2\beta_{l-1} - 1)) \quad (13)$$

Proof. First, note the correspondence between each k -mer and its position in the $FCGR_k$ matrix. In Lemma 3, we have found the correspondence between the indices in the matrix (equation 11) so that each position (i, j) defines a k -mer $w = a_1 a_2 \cdots a_k$. Replacing equation 11 into equation 4, yields the corresponding index τ of w in the lexicographic order.

Theorem 5. *Let $1 < k < n \in \mathbb{Z}$, $\epsilon \in [0, 1]$, and let θ be a point in Δ^{4^k-1} satisfying the marginal constraint in equation 7 for every vertex v . If $n > \frac{2(k-1) \cdot 4^k}{\epsilon} + k - 1$, then there exists an algorithm that constructs a sequence s whose empirical k -mer distribution $\hat{\theta}_s = \frac{F_k(s)}{\|F_k(s)\|_1}$ satisfies $\|\hat{\theta}_s - \theta\|_1 \leq \epsilon$.*

Proof. Let $\mathbf{c}$ be the rounded count vector obtained by rescaling θ by $n - k + 1$ and let $G = (V, E)$ be the de Bruijn graph constructed from $\mathbf{c}$. For each vertex $v \in V$ (corresponding to a $(k - 1)$ -mer), define the flow imbalance

$$\delta(v) = \sum_{a \in \Sigma} c_{va} - \sum_{a \in \Sigma} c_{av}.$$

Denote by $V^+ = \{v \in V : \delta(v) > 0\}$ the set of vertices with excess outgoing edges, and by $V^- = \{v \in V : \delta(v) < 0\}$ the set of vertices with excess incoming edges. By conservation of the total edge count, $\sum_{v \in V} \delta(v) = 0$; hence, the total excess outdegree equals the total excess indegree.

To balance the graph and satisfy Euler's condition, we add artificial edges. Specifically, for every $v \in V^+$ and every $w \in V^-$ there is a directed path $P_{v \rightarrow w}$ of length at most $k - 1$ edges (and hence $k - 2$ intermediate vertices $u_1, \dots, u_{k-2}$). This can be seen from the De Bruijn multigraph associated with the sequence $vw \in \Sigma^{2k-2}$, it starts with vertex v and ends with vertex w . To correct the imbalance of G at v and w , we add one edge connecting each node in $P_{v \rightarrow w}$, which increments the count of each edge in $P_{v \rightarrow w}$ by one. Doing this $\delta(v)$ times for each $v \in V^+$ restores in- and out-degree balance at every vertex (since intermediate nodes gain one in-edge and one out-edge, leaving their imbalance unchanged). Denote by $G' = (V, E')$ the resulting balanced graph. By construction, every vertex in G' now has equal indegree and outdegree. If G' is strongly connected, Euler's theorem guarantees the existence of an Eulerian cycle. If not, at most 4^{k-1} additional artificial edges are needed to connect the components. The sequence s' is reconstructed by traversing the Eulerian path in G' .

Error bound: Define $\delta_{max} = \max_{v \in V} |\delta(v)|$. Since the maximum rounding error for any count is $\frac{1}{2}$ and each $(k - 1)$ -mer has at most 4 incoming and 4 outgoing k -mers, we have $\delta_{max} \leq 4$. Let n_{art} denote the total number of artificial edges added; then

$$n_{art} = \sum_{v \in V^+} (k - 1) \cdot \delta(v) \leq (k - 1) \cdot \delta_{max} \cdot |V^+| \leq (k - 1) \cdot 4^k.$$

Denote $\mathbf{c} = (c_1, \dots, c_{4^k})$ the original k -mer counts and by $\mathbf{c}' = (c'_1, \dots, c'_{4^k})$ the k -mer counts in the reconstructed sequence s' . Each artificial edge increases the

corresponding k -mer count by 1, consequently, the error in the i -th component of the empirical distribution is

$$\left| \hat{\theta}_i^{s'} - \theta_i \right| = \left| \frac{c'_i}{|E| + n_{art}} - \frac{c_i}{|E|} \right|,$$

which can be rewritten as:

$$\begin{aligned} \left| \hat{\theta}_i^{s'} - \theta_i \right| &= \left| \frac{c'_i \cdot |E| - c_i \cdot (|E| + n_{art})}{|E| \cdot (|E| + n_{art})} \right| \\ &= \left| \frac{c'_i \cdot |E| - c_i \cdot |E| - c_i \cdot n_{art}}{|E| \cdot (|E| + n_{art})} \right| \\ &= \left| \frac{|E| \cdot (c'_i - c_i) - c_i \cdot n_{art}}{|E| \cdot (|E| + n_{art})} \right|. \end{aligned}$$

When we sum this over all k -mers, using the facts that $\sum_{i=1}^{4^k} |c'_i - c_i| = n_{art}$ and $\sum_{i=1}^{4^k} c_i = |E|$, we get:

$$\sum_{i=1}^{4^k} \left| \hat{\theta}_i^{s'} - \theta_i \right| \leq \frac{2n_{art}}{|E| + n_{art}},$$

$$\|\hat{\boldsymbol{\theta}}_{s'} - \boldsymbol{\theta}\|_1 \leq \frac{2n_{art}}{|E| + n_{art}}.$$

Since $|E| = n - k + 1$ and $n_{art} \leq (k - 1) \cdot 4^k$, it follows that

$$\|\hat{\boldsymbol{\theta}}_{s'} - \boldsymbol{\theta}\|_1 \leq \frac{2 \cdot (k - 1) \cdot 4^k}{(n - k + 1) + (k - 1) \cdot 4^k}.$$

Imposing the condition $n > \frac{2 \cdot (k-1) \cdot 4^k}{\epsilon} + k - 1$, ensures that $\|\hat{\boldsymbol{\theta}}_{s'} - \boldsymbol{\theta}\|_1 \leq \epsilon$.

Appendix B: Supplementary Information

B.4 Dihedral Group

The *dihedral group* [6] $D_8 = \{e, r, r^2, r^3, s, sr, sr^2, sr^3\}$ is the group of all symmetries of the square, a group comprising rotations and reflections about lines joining midpoints of opposite sides, respectively reflections about diagonals:

- e is the identity (no change)
- r (respectively r^2, r^3) is the rotation by $\frac{\pi}{2}$ (resp. $\pi, \frac{3\pi}{2}$) counterclockwise.
- s is the reflection about the line bisecting the left and right sides of the square (i.e. the line $y_1 = \{(x, 0) : x \in \mathbb{R}\}$ shown in Figure 5.)
- sr is the reflection about the diagonal from the bottom-right to the top-left vertex (i.e. the line $y_2 = \{(x_1, x_2) \in \mathbb{R}^2 : x_1 = -x_2\}$ shown in Figure 5.)
- sr^2 is the reflection about the line bisecting the top and bottom sides of the square (i.e. the line $y_3 = \{(0, y) : y \in \mathbb{R}\}$ shown in Figure 5.)
- sr^3 is the reflection about the diagonal from the top-right to the bottom-left vertex (i.e. the line $y_4 = \{(x_1, x_2) \in \mathbb{R}^2 : x_1 = x_2\}$ shown in Figure 5.)

The following representations, using matrices in $M_{2 \times 2}(R)$, will be used to implement these symmetries.

$$\begin{array}{llll} e \mapsto \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} & r \mapsto \begin{bmatrix} 0 & -1 \\ 1 & 0 \end{bmatrix} & r^2 \mapsto \begin{bmatrix} -1 & 0 \\ 0 & -1 \end{bmatrix} & r^3 \mapsto \begin{bmatrix} 0 & 1 \\ -1 & 0 \end{bmatrix} \\ s \mapsto \begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix} & sr \mapsto \begin{bmatrix} 0 & -1 \\ -1 & 0 \end{bmatrix} & sr^2 \mapsto \begin{bmatrix} -1 & 0 \\ 0 & 1 \end{bmatrix} & sr^3 \mapsto \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix} \end{array}$$

B.5 FCGR via Kronecker Products

Kronecker product $\otimes$ of A of size $m_1 \times n_1$ with B of size $m_2 \times n_2$ is a matrix of size $m_1 m_2 \times n_1 n_2$ defined as follows:

$$A \otimes B = \begin{bmatrix} A_{11}B & \cdots & A_{1n_1}B \\ \vdots & \ddots & \vdots \\ A_{m_1 1}B & \cdots & A_{m_1 n_1}B \end{bmatrix}.$$

In [10], FCGR is defined via Kronecker products of the 2×2 matrix as follows.

$$M = \begin{bmatrix} C & G \\ A & T \end{bmatrix}, \quad M^{(k)} = \underbrace{M \otimes M \otimes \cdots \otimes M}_{k \text{ times}}.$$

Definition 15 ([10]). A $FCGR_k$ of a sequence $s \in \Sigma^n$ with order k with $n \geq k$, is a matrix $FCGR_k(s) \in \mathbb{R}^{2^k \times 2^k}$ such that its (i, j) th entry $FCGR_k(s)(i, j)$ is the frequency of the k -mer $M_{ij}^{(k)}$.

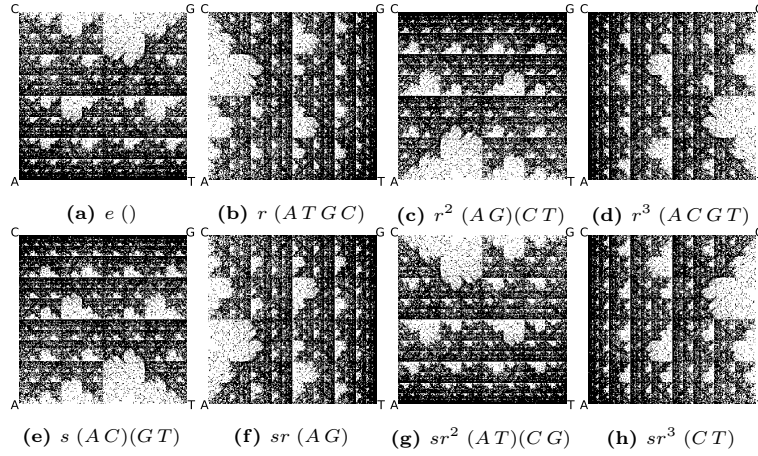


Fig. 5: Visual correspondence between the eight elements of the permutation group S_4 acting on $\{A, C, G, T\}$ and the dihedral group D_8 . Panels (a) to (h) show Chaos Game Representations of a 100 kb randomly selected fragment from human chromosome 4 (GRCh38.p14 primary assembly, RefSeq NC_000004.12) after applying the indicated nucleotide permutation.

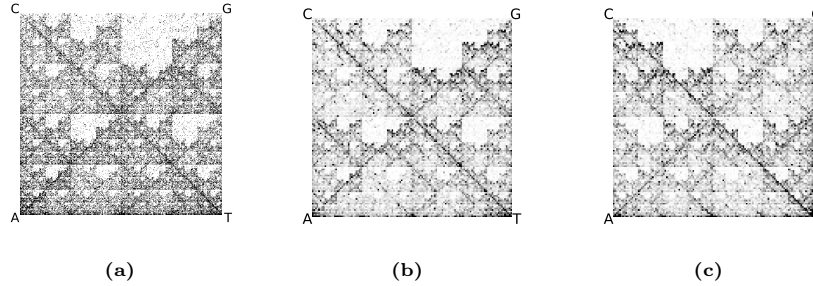


Fig. 6: A visual comparison of FCGRs of order $k = 7$ according to different definitions. The intensity of the pixel represents the frequencies of 7-mers. Darker pixels mean larger frequencies. **(a)** CGR of a 100,000 bp DNA sequence randomly selected from human chromosome 4 (GRCh38.p14 primary assembly, RefSeq NC_000004.12, used in Figure 1b). **(b)** FCGR of order 7 of the same sequence as in (a), according to our Definition 7. **(c)** FCGR of order 7 of the same sequence as in (a), according to [10].

Note that Definition 15 and Definition 7 in Section 3 are not the same. As a counterexample, consider the case $k = 2$. For a given $s \in \Sigma^n, n \geq k$, the FCGR entry $FCGR_k(s)(1, 2)$ will be the frequency of GC according to Definition 7 while according to Definition 15 from [10], $FCGR_k(s)(1, 2)$ will be the frequency of CG instead. See Figure 6 for a visual comparison of FCGRs of order $k = 7$ according to these two definitions for the 100,000 bp DNA sequence from human chromosome 4 (GRCh38.p14 primary assembly, RefSeq NC_000004.12, used in Figure 1b).

B.6 De Bruijn graphs and their applications in sequence reconstruction

Definition 16 (De Bruijn graph). Let $k \geq 2$, $n > k$ and $s \in \Sigma^n$. The De Bruijn graph $DE\ BRUIJN_k(s)$ is the directed graph $(V_{k-1}(s), E_k(s))$, defined as follows:

- $V_{k-1}(s) = \{v \in \Sigma^{k-1} \mid v \text{ appears as a substring of } s\}$.
- $E_k(s) = \{(u, v) \in V_{k-1}(s)^2 \mid \text{there exists a } k\text{-mer } w \in \Sigma^k \text{ that appears in } s \text{ with } \text{PREFIX}(w) = u \text{ and } \text{SUFFIX}(w) = v\}$.

An illustration of the graph $DE\ BRUIJN_2(\text{ATCGTATCCA})$ is presented in Figure 7a). In this construction, each edge uniquely corresponds to a specific k -mer in the original string, so that every path through $DE\ BRUIJN_k(s)$ represents a substring of s containing each k -mer exactly once. This characteristic, however, is insufficient for our purposes, as in our application, the goal is to reconstruct a sequence that preserves the target k -mer distribution; thus, it is necessary to employ a structure that accommodates multiple occurrences of the same k -mer. In other words, when a k -mer appears more than once in s , the standard De Bruijn graph is insufficient for recovering any substring corresponding to those repeated k -mers.

To address this issue, we introduce the notion of De Bruijn multigraph in Definition 14, a concept well established in the DNA sequence assembly literature. Note that an edge from vertex u to vertex v exists if the k -mer associated with that edge has u as its prefix and v as its suffix, with the edge's multiplicity corresponding to the frequency of the associated k -mer in s . Figure 7b) illustrates the multigraph $DE\ BRUIJN_2^+(\text{ATCGTATCCA})$.



Fig. 7: Comparison of De Bruijn graphs and multigraph of a sample string $s = \text{ATCGTATCCA}$, for $k = 3$. Each vertex represents a unique 2-mer from the string, and each edge represents a unique 3-mer. (a) De Bruijn graph: A walk in the graph corresponds to a substring of s with each k -mer occurring exactly once. (b) De Bruijn multigraph: The edge labels indicate the multiplicity of each 3-mer, thereby allowing the recovery of any substring of s via an appropriate Eulerian path.

B.7 Hit-and-run sampling from the standard probability simplex with marginalization constraints

Let $\theta_s = (\theta_w)_{w \in \Sigma^k}$, $\Sigma = \{A, C, G, T\}$, denote the empirical k -mer frequency vector computed from a (linear) DNA sequence s . By construction,

$$\theta_w \geq 0, \quad \sum_{w \in \Sigma^k} \theta_w = 1.$$

In addition to this normalization, θ_s satisfies a family of linear constraints induced by contiguous overlaps. This is commonly called *marginal consistency*: for each $v \in \Sigma^{k-1}$,

$$\sum_{a \in \Sigma} \theta_{va} = \sum_{a \in \Sigma} \theta_{av} = \theta_v, \quad (14)$$

where θ_v is the frequency of the $(k-1)$ -mer v . Combining these with the normalization yields a system

$$B\theta = b, \quad (15)$$

where:

- Row 0 of B is $\mathbf{1}^\top$, enforcing $\sum_i x_i = 1$.
- For each $(k-1)$ -mer v , one row enforces $\sum_{a \in \Sigma} x_{va} - \sum_{a \in \Sigma} x_{av} = 0$.
- $b = (1, 0, \dots, 0)$

Proposition 5. *The matrix B defined by equation 15 has full row rank.*

Proof. Let the rows of B be $r_0, r_{v_1}, \dots, r_{v_M}$, where $r_0 = \mathbf{1}^\top$ and each r_v has nonzero entries only on indices corresponding to k -mers with prefix or suffix v . For distinct v and u , their supports are disjoint, so r_v are mutually orthogonal and hence independent. Moreover, r_0 cannot be expressed as a linear combination of the zero-sum rows r_v . Therefore no nontrivial linear relation among $\{r_0, r_v\}$ exists, and B has full row rank.

Let $r = \text{rank}(B)$. The feasible set $\{x \geq 0 : Bx = b\}$ is an $(4^k - r)$ -dimensional convex polytope. We sample approximately uniformly from this polytope via a *hit-and-run* Markov chain as follows:

1. Compute an orthonormal basis $N \in \mathbb{R}^{4^k \times (4^k - r)}$ of $\ker(B)$ (e.g. via SVD).
2. Initialize $\theta^{(0)} = (1/4^k, \dots, 1/4^k)$.
3. For each iteration $t = 1, \dots, T$:
 - (a) Draw $\mathbf{z} \sim \mathcal{N}(0, I_{4^k - r})$ and set $\mathbf{d} = N\mathbf{z}/\|N\mathbf{z}\|$.
 - (b) Find $t_{\min} \leq t_{\max}$ so that $\theta^{(t-1)} + t\mathbf{d} \geq 0$ coordinate-wise.
 - (c) Sample $t \sim \text{Unif}[t_{\min}, t_{\max}]$.
 - (d) Update $\theta^{(t)} = \theta^{(t-1)} + t\mathbf{d}$.
4. Return $\theta^{(T)}$, which lies in the simplex and satisfies $B\theta = b$.

At each step, the direction d is drawn from a continuous distribution spanning all of $\ker(B)$, so any two interior points can be connected by a finite sequence of positive-probability moves. This procedure corresponds to a standard hit-and-run sampling strategy, common in MCMC theory (e.g. [21]).

B.8 Real data acquisition and preprocessing

All genome data were obtained on 24 April 2025 from the NCBI RefSeq FTP server (<ftp.ncbi.nlm.nih.gov/genomes/refseq>). We considered eight taxonomic groups: `Archaea`, `Bacteria`, `Fungi`, `Plant`, `Protozoa`, `Invertebrate`, `Vertebrate_Mammalian` and `Vertebrate_Other`. The last three correspond to the kingdom `Animalia`. For each taxonomic group, the summary file was downloaded and parsed into a table of assembly metadata. Entries were grouped by the field `organism_name`, and one assembly per species was selected by taking the first entry per group and then randomly sampling 100 species without replacement (random seed = 42). Assemblies lacking a valid FTP path or marked incomplete were omitted.

For each selected assembly, the corresponding FASTA file was downloaded via `urllib` (Python 3.11), decompressed with `gzip`, and parsed using `Biopython` (v1.79). From each complete genome sequence, a single 100,000 bp fragment was extracted by choosing a start position uniformly at random from (seed = 42). Fragments shorter than 100 kbp were discarded. Each fragment was saved in FASTA format and CGR images (256×256 px) were generated in `NumPy` (v1.23.5) and `Matplotlib` (v3.7.1).

All steps were automated by a Python 3.9 script (available in Supplementary Software) to ensure reproducibility.

B.9 Interactive synthetic CGR image generator

To facilitate visual exploration of compositional biases in synthetic DNA, the research presented in this work is accompanied by an open-source, browser-based tool that generates sequences whose k -mer frequency vector matches a user-specified or a sampled distribution. The application supports rapid, visual inspection of how local changes in dinucleotide or higher-order statistics alter global CGR patterns as the binary Chaos Game Representation (CGR) of the generated sequence is displayed to the user in each case.

The user may choose a value of $k \in \{2, \dots, 6\}$, a sequence length L and may choose to generate the sequence from real DNA or to sample a point from the standard simplex subject to the marginal constraints. Additionally, for $k=2$, 16 log-scaled sliders can be adjusted to select a target probability distribution (See Figure 4). The target frequency vector is converted to integer counts ($n - k + 1$ observations) and embedded in a directed multigraph whose nodes are $(k - 1)$ -mers. A depth-first Eulerian traversal yields a linear sequence that approximately produces the count multiset. The server returns a 256×256 binary CGR (PNG, base-64 embedded). For $k \leq 3$ it also returns a bar plot of the sampled probabilities.

The tool is implemented in Python 3.11 using the following libraries: `Flask` (v2.3), `NumPy` (1.23.5) and `Matplotlib` (v3.8). The typical runtime is < 1 s for $n \leq 10^5$ and peak memory is less than 200MB on a standard laptop. The full source code can be found on GitHub under the MIT license. see Figure 4.