A near-tight lower bound on the density of forward sampling schemes

Bryce Kille[1,*,†], Ragnar Groot Koerkamp[2,*,†],
Drake McAdams[1], Alan Liu[1], and Todd J. Treangen[1,3]

[1]Department of Computer Science, Rice University, Houston, TX, USA
[2]Department of Computer Science, ETH Zurich, Zurich, Switzerland
[3]Ken Kennedy Institute, Rice University, Houston, TX, USA
[*]These authors contributed equally to this work.
[†]To whom correspondence should be addressed: blk6@rice.edu and ragnar.grootkoerkamp@inf.ethz.ch

November 19, 2024

Abstract

Motivation: Sampling k-mers is a ubiquitous task in sequence analysis algorithms. Sampling schemes such as the often-used random minimizer scheme are particularly appealing as they guarantee at least one k-mer is selected out of every w consecutive k-mers. Sampling fewer k-mers often leads to an increase in efficiency of downstream methods. Thus, developing schemes that have low density, i.e., have a small proportion of sampled k-mers, is an active area of research. After over a decade of consistent efforts in both decreasing the density of practical schemes and increasing the lower bound on the best possible density, there is still a large gap between the two.

Results: We prove a near-tight lower bound on the density of forward sampling schemes, a class of schemes that generalizes minimizer schemes. For small w and k, we observe that our bound is tight when $k \equiv 1 \pmod{w}$. For large w and k, the bound can be approximated by $\frac{1}{w+k}\lceil\frac{w+k}{w}\rceil$. Importantly, our lower bound implies that existing schemes are much closer to achieving optimal density than previously known. For example, with the current default minimap2 HiFi settings $w = 19$ and $k = 19$, we show that the best known scheme for these parameters, the double decycling-set-based minimizer of Pellow et al., is at most 3% denser than optimal, compared to the previous gap of at most 50%. Furthermore, when $k \equiv 1 \pmod{w}$ and the alphabet size σ goes to ∞, we show that mod-minimizers introduced by Groot Koerkamp and Pibiri achieve optimal density matching our lower bound.

Availability and implementation:
Minimizer implementations: github.com/RagnarGrootKoerkamp/minimizers
ILP and analysis: github.com/treangenlab/sampling-scheme-analysis

1 Introduction

For over a decade, k-mer sampling schemes have served as a ubiquitous first step in many classes of bioinformatics tasks. By sampling k-mers in a way which ensures that two similar sequences will have similar sets of sampled k-mers, sampling schemes enable methods to bypass the need to compare entire sequences at the base level and instead allow them to work more efficiently using the sampled k-mers.

Local sampling schemes satisfy a *window guarantee* that at least one k-mer is selected out of every window of w consecutive k-mers. Most schemes used in practice, such as the random minimizer scheme (Schleimer et al., 2003; Roberts et al., 2004), are *forward schemes* that additionally guarantee that k-mers are sampled in the order in which they appear in the original sequence. These properties are particularly appealing since they guarantee that no region is left unsampled.

As the purpose of these schemes is to reduce the computational burden of downstream methods while upholding the window guarantee, the primary goal of most new schemes is to minimize the *density*, i.e., the expected proportion of sampled k-mers. Over the past decade, many new schemes have been proposed that obtain significantly lower densities than the original random minimizer scheme.

For example, there are schemes based on *hitting sets* (Orenstein et al., 2016; Marçais et al., 2017, 2018; DeBlasio et al., 2019; Ekim et al., 2020; Pellow et al., 2023; Golan et al., 2024), schemes that focus on sampling positions rather than k-mers (Loukides and Pissis, 2021; Loukides et al., 2023), schemes that use an ordering on t-mers ($t < k$) to decide which k-mer to sample (Zheng et al., 2020; Groot Koerkamp and Pibiri, 2024), and schemes that aim to minimize density on specific input sequences (Zheng et al., 2021b; Hoang et al., 2022). All of these improvements notwithstanding, it is still unknown how close these schemes are to achieving minimum density.

A trivial lower bound on density given by the window guarantee is $\frac{1}{w}$, and recently Groot Koerkamp and Pibiri (2024) improved the bound of Marçais et al. (2018) from $\frac{1.5+\frac{1}{2w}}{w+k}$ to $\frac{1.5}{w+k-0.5}$. However, for many practical values of w and k, there is a sizeable gap between these lower bounds and the density of existing schemes. This raises the question whether schemes with density much closer to $\frac{1}{w}$ exist, but have not been found yet, or whether existing schemes are already very close to optimal and it is the lower bound that needs improvement. Our new lower bound closes most of the gap, and thus answers this question: Indeed, especially for $k \geq w$, the best existing schemes have near-optimal density in many cases. This allows future research to focus on improving other sampling scheme metrics, such as the *conservation* described in Edgar (2021) and Shaw and Yu (2022).

1.1 Contributions

Main lower bound theorem. We prove a novel lower bound on the density of forward schemes that is strictly tighter than all previously established lower bounds for all w, k, and alphabet size σ:

Theorem 1. *Let f be a (w, k)-forward sampling scheme and $M_\sigma(p)$ count the number of aperiodic necklaces of length p over an alphabet of size σ. Then, the density of f is at least*

$$g_\sigma(w,k) := \frac{1}{\sigma^{w+k}} \sum_{p\,|(w+k)} M_\sigma(p) \left\lceil \frac{p}{w} \right\rceil \geq \frac{\left\lceil \frac{w+k}{w} \right\rceil}{w+k} \geq \frac{1}{w}, \tag{1}$$

where the middle inequality is strict for $w > 1$.

We prove that this bound can be extended to work for more general classes of sampling schemes, such as the local schemes described by Marçais et al. (2018) and the multi-local schemes described by Kille et al. (2023).

Comparison with optimal schemes for small parameters. We show that our lower bound is tight for some small w, k, and σ by using an integer linear program to construct schemes whose density matches our lower bound. This marks the first time that there is an analytical description of a tight minimum density of any forward scheme. We conjecture that when $k \equiv 1 \pmod{w}$, there exist schemes with density matching our lower bound.

Comparison with practical schemes for large parameters. To show that our bound is significantly closer to the density achieved by existing schemes compared to previous lower bounds, we replicate the benchmark from Groot Koerkamp and Pibiri (2024) for a selection of w and k (Fig. 3). For example, with the default minimap2 (Li, 2018) HiFi settings $w = 19$ and $k = 19$, the lower bound goes up from 50% of the density achieved by the double decycling based method to 97% of the achieved density (Table 1).

Analysis of the mod-minimizer. Finally, our new lower bound implies that the mod-minimizer scheme (Groot Koerkamp and Pibiri, 2024) is optimal when $k \equiv 1 \pmod{w}$ and σ is large. Indeed, for the ASCII alphabet ($\sigma = 256$), the mod-minimizer scheme density is consistently within 1% of the lower bound when $k \equiv 1 \pmod{w}$ (Fig. 4 in Supplement D).

2 Background

Notation. We begin by defining some necessary notation, as well as definitions of mathematical concepts that will be used throughout the work. We use $[n]$ to refer to the set $\{0, 1, \ldots, n-1\}$. The alphabet is denoted by Σ and has size $\sigma := |\Sigma|$, with $\sigma = 4$ for DNA. The expression $a \mid b$ indicates that a divides b. The summation $\sum_{a|b}$ is over all positive divisors a of b. We use $a \bmod m$ for the remainder (in $[m]$) of a after dividing by m and we use $a \equiv b \pmod{m}$ to indicate that a and b have the same remainder modulo m. Given a string W, $W[i, j)$ refers to the substring of W containing the characters at 0-based positions i up to $j-1$ inclusive. For two strings X and Y, XY represents the concatenation of X and Y.

Classes of sampling schemes. There are multiple established classes of sampling schemes. We begin by drawing a distinction between schemes with and without a *window guarantee* that guarantees that at least one every w k-mers is sampled. While schemes without a window guarantee, such as fracminhash (Irber et al., 2022), are often efficient to compute, the lack of a guarantee on the distance between sampled k-mers makes them ineffective or inefficient for certain tasks such as indexing and alignment. Indeed, we only consider schemes with a window guarantee:

Definition 1. *A* (w, k)-local scheme *with window guarantee w and k-mer size k on an alphabet Σ corresponds to a sampling function $f : \Sigma^{w+k-1} \to [w]$.*

In other words, given a window of $w + k - 1$ characters (w consecutive k-mers), the output of the sampling function $f(W)$ is an integer in $[w]$ which represents the index of the sampled k-mer in W. Recently, Kille et al. (2023) proposed a generalization of (w, k)-local schemes which samples at least s k-mers out of every w instead of at least 1 and we extend our results to these more general schemes in Supplement A.

Local schemes have no restrictions on which of the w k-mers can be selected for each window, but *forward schemes* are a subset of local schemes that enforce the restriction that they never select a k-mer which occurs before a previously selected k-mer.

Definition 2. *A (w, k)-local scheme is also* (w, k)-forward *if for all strings $W \in \Sigma^{w+k}$ representing two adjacent windows,*

$$f(W[0, w + k - 1)) \leq f(W[1, w + k)) + 1.$$

Definition 3. *The* density $\mathrm{d}(f)$ *of a sampling scheme f is defined as the expected proportion of sampled positions from an infinite, uniformly random string.*

For a further background on types of sampling schemes, we refer to Shaw and Yu (2022), Zheng et al. (2023), Groot Koerkamp and Pibiri (2024), and Ndiaye et al. (2024).

De Bruijn graphs. Let $B_{n,\sigma} = (V, E)$ denote the complete De Bruijn graph of order n, which has as vertices all strings of length n, $V = \Sigma^n$, and edges between vertices that overlap in $n - 1$ positions, $E = \{(X, X[1, n)c) \mid X \in V, c \in \Sigma\}$. When σ is clear from the context or irrelevant for a particular discussion, it is omitted. It is worth noting that the vertices of B_{n+1} correspond to edges of B_n.

For each string s of length n, the n rotations of s induce a *pure cycle* in B_n consisting of (up to) n vertices cyclically connected by edges. Note that when s is repetitive, e.g., a single repeated character or some other repeated string, the length of the cycle will be a divisor of n. These pure cycles are also called *necklaces*. The set of necklaces of length n corresponds to a partitioning of the vertices of B_n

into a vertex-disjoint set of pure cycles. We use $\mathcal{C}_n$ to refer to this set of pure cycles of B_n, and for $c \in \mathcal{C}_n$, we write $|c|$ for the number of vertices in the cycle.

When a string of length n has n unique rotations, the corresponding necklace is said to be *aperiodic*. The total number of necklaces and the number of aperiodic necklaces of length n are given by Moreau (1872) (and see also Riordan (1957)) as respectively

$$N_\sigma(n) = \frac{1}{n} \sum_{p|n} \varphi(n/p) \cdot \sigma^p, \qquad\qquad M_\sigma(n) = \frac{1}{n} \sum_{p|n} \mu(n/p) \cdot \sigma^p,$$

where $\varphi(p)$ is Euler's totient function that counts the number of integers in $[p]$ coprime to p. The formula $M_\sigma(n)$ counting *aperiodic* necklaces follows from the formula for $N_\sigma(n)$ via *Möbius inversion* (Möbius, 1832), where μ is the *Möbius function* defined to be 0 if n is divisible by a square (> 1) and $\mu(n) = (-1)^q$ otherwise, where q is the number of prime factors of n.

Charged contexts. The *context* of a window of length $w + k - 1$ in a sequence is the set of preceding windows that influences whether the current window samples a new position.

For a local scheme to select a new position, none of the previous $w - 1$ windows may have selected the same k-mer as the current window. As a result, the context for local schemes consists of $2w + k - 2$ characters: the current window of w k-mers as well as the $w - 1$ windows preceding the current window.

For a forward scheme, however, as soon as a window samples a different position than the preceding window, this position must be a new position. Thus, one needs only to consider the context of two consecutive windows of w k-mers, for a total of $w + k$ characters.

When a sampling scheme selects a new position for the last window in a context, the context is *charged*. Marçais et al. (2017) showed that the density of a scheme f can be defined as the proportion of contexts which are charged. In the case of forward schemes, each edge in B_{w+k-1} represents a context, and the charged contexts are the edges (u, v) for which $f(u) \neq f(v) + 1$.

Universal hitting sets. In 2021, Zheng et al. (2021a) related the density of forward and local schemes to the concept of universal hitting sets (UHS). A (w, ℓ)-UHS is defined as a set of ℓ-mers U such that *any* sequence of w adjacent ℓ-mers must contain at least one ℓ-mer from U. Theorem 1 of Zheng et al. (2021a) showed that when $k = 1$, one can use the minimum size of a $(w, \ell{=}w + k)$-UHS to bound the density of $(w, k{=}1)$-forward schemes, and the minimum size of a $(w, \ell{=}2w + k - 2)$-UHS to bound the density of a $(w, k{=}1)$-local schemes.

3 Theoretical results

In this section, we prove our main result: an improved lower bound on the density of forward sampling schemes. We first generalize some existing theorems to arbitrary w and k (Sections 3.1 and 3.2), after which our main theorem follows in Section 3.3.

3.1 A lower bound on the size of a (w, ℓ)-UHS

We begin by considering a $(w{=}2, \ell)$-UHS. A $(2, \ell)$-UHS is equivalent to a vertex cover in B_ℓ, i.e., a subset of vertices such that each edge in B_ℓ is adjacent to at least one vertex in the subset. Lichiardopol (2006) used the fact that for every cycle C, at least $\lceil |C|/2 \rceil$ of its vertices must be in a vertex cover, and obtained a lower bound on the size of a vertex cover by partitioning B_ℓ into its pure cycles. We naturally extend this argument to obtain a lower bound on the cardinality of a (w, ℓ)-UHS for any $w \geq 2$.

Proposition 4. *Let $M_\sigma(p)$ count the number of aperiodic necklaces of length p. For any (w, ℓ)-UHS U,*

$$|U| \geq \sum_{p|\ell} M_\sigma(p) \left\lceil \frac{p}{w} \right\rceil.$$

Proof. The pure cycles of $\mathcal{C}_\ell$ partition the vertices of B_ℓ. For any simple cycle of size p in B_ℓ, a (w,ℓ)-UHS must contain at least $\lceil p/w \rceil$ ℓ-mers. As there is a one-to-one correspondence between the pure cycles of length $p \mid \ell$ in B_ℓ and the $M_\sigma(p)$ aperiodic necklaces of length p, we have

$$|U| \geq \sum_{c \in \mathcal{C}_\ell} \left\lceil \frac{|c|}{w} \right\rceil = \sum_{p \mid \ell} M_\sigma(p) \left\lceil \frac{p}{w} \right\rceil. \qquad \square$$

Fig. 1b provides a depiction of a minimum $(2,4)$-UHS as well as the pure-cycle partitioning of B_4 on a binary alphabet. Notably, the pure cycle $(0011, 0110, 1100, 1001)$ has 3 vertices in the UHS, even though the lower bound given by Proposition 4 only requires it have 2. This is an example where the lower bound is not tight.

For certain values of w and ℓ, such as when ℓ is prime or $w = 2$ and ℓ is odd, Proposition 4 can be simplified to remove the summation and ceil function (Supplement B).

Proposition 4 is the core of the proof of Theorem 1 and already has the right structure. The remainder of this section translates this result on universal hitting sets to a result on the density of sampling schemes.

3.2 A connection between sampling scheme density and UHS size

Zheng et al. (2021a, Theorem 1) show a connection between universal hitting sets and the density of sampling schemes when $k = 1$. We naturally extend their result to $k \geq 1$ for both local schemes (Lemma 5) and forward schemes (Corollary 6).

Lemma 5. *Let f be a (w,k)-local scheme, and let C_f be its corresponding set of charged contexts defined as the set of strings W of length $2w+k-2$ for which the last window $W[w-1, 2w+k-2)$ selects a position $w-1+f(W[w-1,2w+k-1))$ not selected by any previous window:*

$$\begin{aligned} C_f := \{ &W \in \Sigma^{2w+k-2} \mid \forall\, 0 \leq i \leq w-2, \\ &f(W[w-1, 2w+k-2)) + (w-1) \neq f(W[i, i+w+k-1)) + i \}. \end{aligned}$$

Then, C_f is a $(w, 2w+k-2)$-UHS.

Proof. For the sake of a contradiction, suppose there is a walk of length w in the De Bruijn graph of order $(2w+k-2)$, say $(W_0, \ldots, W_{w-1})$, that avoids C_f. Let S be the *spelling* of the walk, i.e., the sequence of length $3w+k-3$ such that $S[i, i+2w+k-2) = W_i$. Since $W_{w-1} \notin C_f$ and S contains W_{w-1}, this implies that on the last $(w+k-1)$-mer of W_{w-1} (i.e., $S[2w-2, 3w+k-3)$), f selects an index $j \geq 2w-2$ in S which has already been picked.

Since $0 \leq f(\cdot) \leq w-1$ and $j \geq 2w-2$, the first window that selects position j must begin at an index $m \geq w-1$. Therefore, the context $W_{m-w+1} = S[m-(w-1), m+w+k-1)$ is charged, as f selects a previously unselected position when applied to its last $(w+k-1)$-mer. By definition, $W_{m-w+1} \in C_f$, contradicting our supposition and therefore C_f is a $(w, 2w+k-2)$-UHS. □

Identically, one can consider contexts for a (w,k)-forward scheme f, which requires only verifying that the selection for a window of length $w+k-1$ is distinct from the selection for the previous window. Therefore, the length of a context for forward f is only $w+k$. As above, every w contexts must have at least one charged context, leading to the following conclusion:

Corollary 6. *If f is a (w,k)-forward scheme and C_f is its corresponding set of charged contexts, defined as $C_f = \{W \in \Sigma^{w+k} \mid f(W[0, w+k-1)) \neq f(W[1, w+k)) + 1\}$, then C_f is a $(w, w+k)$-UHS.*

As all contexts of a particular length ℓ are equally likely to occur in an infinite, uniform random string, the proportion of charged contexts corresponds to the density of the sampling scheme (Marçais et al., 2017), i.e. $\mathrm{d}(f) = |C_f|/\sigma^\ell$, where $\ell = w+k$ for forward schemes and $\ell = 2w+k-2$ for local schemes. An example of the charged contexts of a $(2,2)$-forward scheme and the corresponding UHS is depicted in Fig. 1.

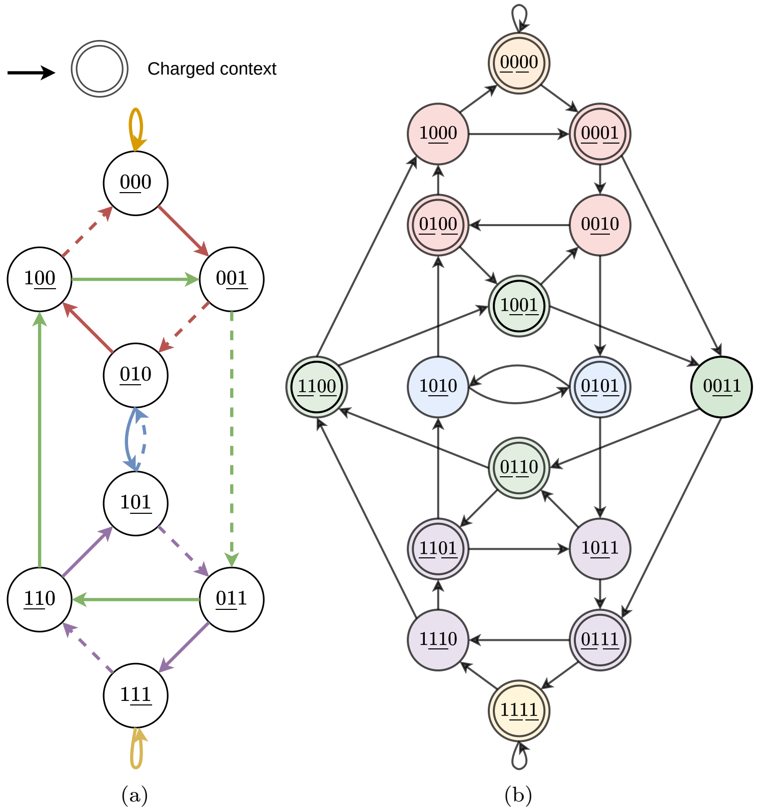

Figure 1: (a) A De Bruijn graph B_3 corresponding to a minimum density (w=2, k=2)-forward scheme. The underlined characters in each vertex represent the 2-mer that is selected for that window. The solid edges represent the charged contexts and the edge colors represent the pure cycles in B_4 (not in B_3 itself). For characters c_i, each edge $(c_0c_1c_2, c'_0c'_1c'_2)$ in B_3 corresponds to the vertex $c_0c_1c_2c'_2$ in B_4. (b) The corresponding (w=2, ℓ=4)-UHS in B_4. The vertices are partitioned by color, representing the pure-cycles. The 2-mer(s) selected in each context are underlined. The vertices with a double border represent the charged edges in B_3 in (a) and the corresponding $(2, 4)$-UHS. Each pure cycle c has at least $\lceil |c|/w \rceil$ vertices in the UHS.

3.3 Lower bounds on local and forward scheme density

We are now ready to state and prove our main theorem.

Theorem 1. *Let f be a (w, k)-forward sampling scheme and $M_\sigma(p)$ count the number of aperiodic necklaces of length p over an alphabet of size σ. Then, the density of f is at least*

$$g_\sigma(w,k) := \frac{1}{\sigma^{w+k}} \sum_{p\,|(w+k)} M_\sigma(p) \left\lceil \frac{p}{w} \right\rceil \geq \frac{\left\lceil \frac{w+k}{w} \right\rceil}{w+k} \geq \frac{1}{w}, \tag{1}$$

where the middle inequality is strict for $w > 1$.

Proof. Due to Corollary 6 and Marçais et al. (2017), we can see that a (w, k)-forward sampling scheme of density $\mathrm{d}(f)$ implies a (w, ℓ=$w + k$)-UHS of size $\sigma^{w+k} \cdot \mathrm{d}(f)$. By Proposition 4, this implies that every forward sampling scheme has a density of at least $g_\sigma(w, k)$, and hence $\mathrm{d}(f) \geq g_\sigma(w, k)$ follows.

For any p that divides $w + k$, we have $\lceil \frac{p}{w} \rceil \geq \frac{p}{w+k} \lceil \frac{w+k}{w} \rceil$, with strict inequality when $p = 1$ and $w > 1$. Substituting this in $g_\sigma(w, k)$, the middle inequality follows directly using the identity

$\sum_{p|w+k} p \cdot M_\sigma(p) = \sigma^{w+k}$ that counts the number of strings of length $w+k$ partitioned by their shortest period.

The last inequality follows directly from $\frac{1}{w+k}\lceil\frac{w+k}{w}\rceil \geq \frac{1}{w+k}\frac{w+k}{w} = 1/w$. □

As we will see in Section 4, $g_\sigma(w,k)$ is a tight bound for many small cases. Since its formula is somewhat unwieldy, $\frac{1}{w+k}\lceil\frac{w+k}{w}\rceil$ can be used as an approximation that quickly approaches $g_\sigma(w,k)$ (Fig. 2). Simple arithmetic shows that both $g_\sigma(w,k)$ and $\frac{1}{w+k}\lceil\frac{w+k}{w}\rceil$ improve the previous lower bound of $\frac{1.5}{w+k-0.5}$ of Groot Koerkamp and Pibiri (2024).

Given a (w,k)-local scheme f_k, we can construct a $(w, k' \geq k)$-local scheme $f_{k'}$ of the same density by ignoring the last $k'-k$ characters in each window, i.e. $f_{k'}(W) = f_k(W[0..(w+k))$. This directly implies $\mathrm{d}(f_k) = \mathrm{d}(f_{k'})$ (Zheng et al., 2021a). It follows that the minimum density of a (w,k)-local or forward scheme is monotonically decreasing as k increases. However, as can be seen in Fig. 2, $g_\sigma(w,k)$ is not a monotonically decreasing function. The local maxima appear to be at $k \equiv 1 \pmod{w}$, which motivates the following improved lower bound.

Theorem 2. *For any (w,k)-forward scheme f, an improved lower bound g' is given by*

$$\mathrm{d}(f) \geq g'_\sigma(w,k) := \max(g_\sigma(w,k), g_\sigma(w,k')) \geq \max\left(\frac{1}{w+k}\left\lceil\frac{w+k}{w}\right\rceil, \frac{1}{w+k'}\left\lceil\frac{w+k'}{w}\right\rceil\right),$$

where k' is the smallest integer $\geq k$ such that $k' \equiv 1 \pmod{w}$.

Remark 7. *Similar to Theorem 1, Lemma 5 implies that any (w,k)-local scheme f has density at least $\mathrm{d}(f) \geq g_\sigma(w, w+k-2)$. As this bound is in terms of g_σ, the improved bound in Theorem 2 can be applied to local schemes as well, i.e., for any (w,k)-local scheme f, an improved lower bound is given by*

$$\mathrm{d}(f) \geq g'_\sigma(w, w+k-2).$$

4 Empirical tightness of our bounds

Here, we compare our bounds g_σ and g'_σ to existing lower bounds. Further, we show how tight these bounds are for small w, k, and σ by searching for optimal schemes via an integer linear programming (ILP) formulation. We also show how close existing sampling scheme densities are to g'_σ for practical choices of w, k, and σ. Finally, we show when the recently described mod-minimizer scheme (Groot Koerkamp and Pibiri, 2024) achieves optimal density as $\sigma \to \infty$.

ILP description. We use an ILP to search for minimum density forward sampling schemes. In short, we use a single integer variable $x_W \in [w]$ for every window W of length $w+k-1$ (corresponding to a vertex in B_{w+k-1}) that indicates the position of the chosen k-mer, and a single boolean variable $y_{(W,W')}$ for each edge in B_{w+k-1} that indicates whether the corresponding context is charged. On each edge, we require that the scheme be forward. The objective function is to minimize the number of charged edges. To reduce the search space, we add an additional constraint corresponding to our lower bound g_σ by requiring that for each pure cycle of length $|c|$ in B_{w+k}, at least $\lceil |c|/w \rceil$ of the corresponding edges in B_{w+k-1} are charged. Further details, including the ILP formulation for local schemes, can be found in Supplement C.

Comparison against optimal schemes for small *k*. We used Gurobi (Gurobi Optimization, LLC, 2024) to solve the ILP for all combinations of w, k, and σ such that $1 \leq w \leq 12$, $1 \leq k \leq 12$, and $2 \leq \sigma \leq 4$ for both forward and local schemes and limited the runtime for each instance to 12 hours on 128 threads. All results are reported in Table 2 in Supplement D. While the additional constraint on pure cycles corresponding to g_σ significantly sped up the search, for most large w, k, and σ, the ILP failed to terminate with an optimal solution in the allotted time. As a result, we restrict most of our analysis to the following three cases: fixed alphabet size $\sigma = 2$, fixed window size $w = 2$, and fixed k-mer size $k = 1$ (Fig. 2).

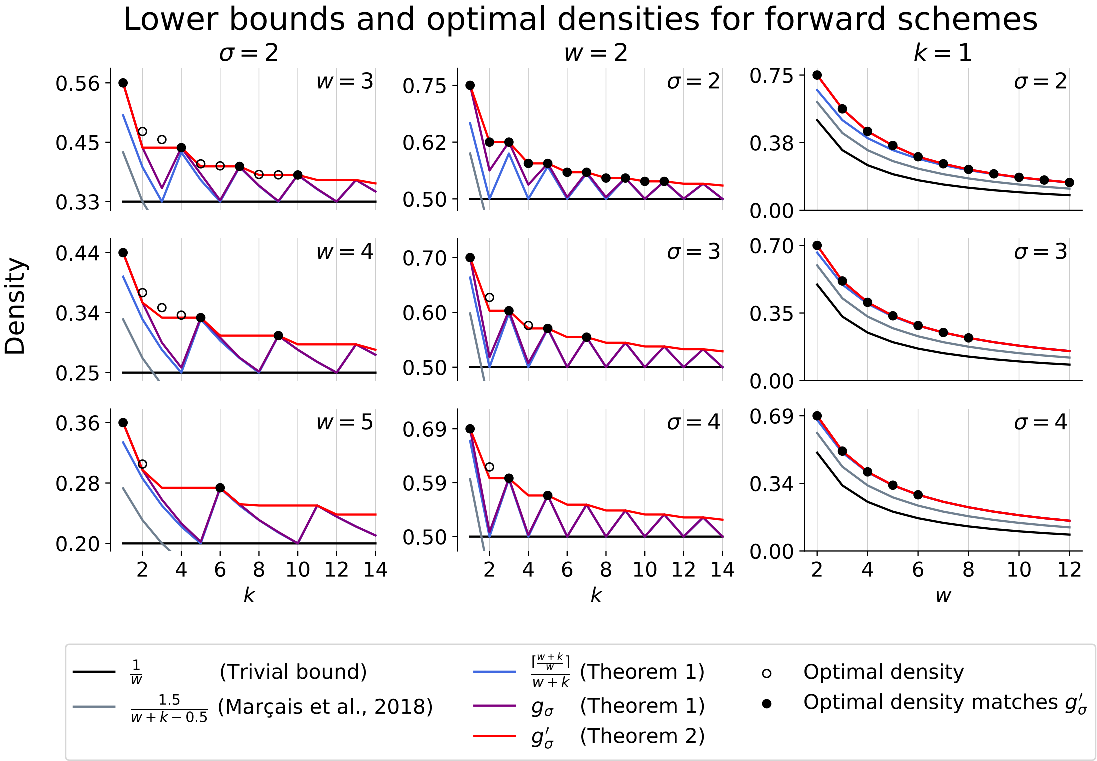

Figure 2: Comparison of forward scheme lower bounds and optimal densities for small w, k, and σ. Optimal densities were obtained via the ILP and are plotted as black circles that are solid when the optimal density matches our lower bound, g'_σ, and hollow otherwise. Each column corresponds to a parameter being fixed to the lowest non-trivial value, i.e., $\sigma = 2$ in the first column, $w = 2$ in the second column, and $k = 1$ in the third column. Note that the x-axis in the third column corresponds to w, not k.

For all (w, k, σ) where $k \equiv 1 \pmod{w}$ (including when $k = 1$), the minimum density exactly matches our lower bound $g_\sigma(w, k)$. Additionally, when $\sigma = 2$ and $w = 2$, the minimum density was equal to $g'_\sigma(w, k)$.

Comparison against existing schemes for large k. Using a sequence of 10 million random characters over alphabet size $\sigma = 4$, we approximated the density of recent sampling schemes using the benchmarking implementation from Groot Koerkamp and Pibiri (2024). To compare each density to the particular proportion of selected k-mers on a genomic sequence, we also ran all sampling schemes on the human Y chromosome (Rhie et al., 2023) after removing all non-ACTG characters. The densities of the best performing methods, Miniception (Zheng et al., 2020), double decycling-set-based minimizers (Pellow et al., 2023), and mod-minimizers (Groot Koerkamp and Pibiri, 2024) are plotted in Fig. 3 along with random minimizers and lower bounds.

The ratio between the minimum achieved densities and lower bounds for a selection of (w, k) pairs used by existing k-mer based methods are presented in Table 1. Additional results for $\sigma \in \{2, 256\}$ and $w \in \{2, 50\}$ are provided in Fig. 4 in Supplement D.

The mod-minimizer has optimal density for large σ when $w \equiv k \pmod{1}$. When w and k are constant and $\sigma \to \infty$, the probability of duplicate characters in a window goes to 0. This implies that we can use $t = 1$ for the mod-minimizer. When $k \equiv t = 1 \pmod{w}$, the density of the

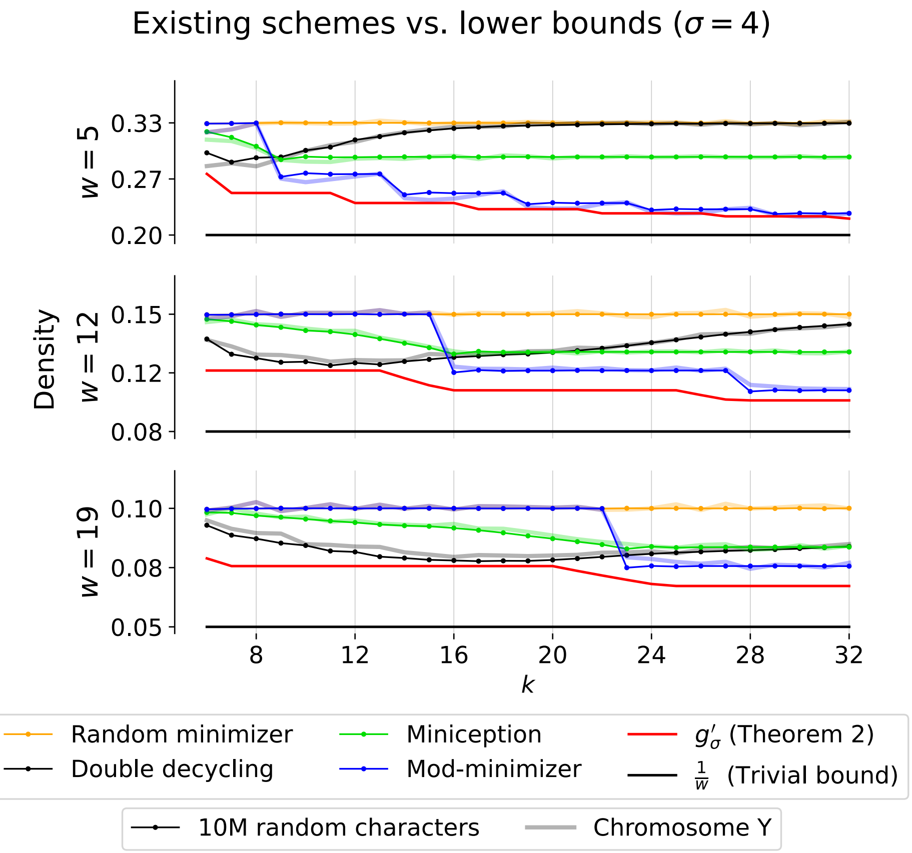
Existing schemes vs. lower bounds ($\sigma = 4$)

Figure 3: Comparison of existing schemes to lower bounds with practical parameters. Densities are calculated by applying each scheme to a random sequence of 10 million characters over an alphabet of size $\sigma = 4$ (dotted lines) and are compared with the corresponding proportion of sampled k-mers on the human Y chromosome (Rhie et al., 2023) (soft lines). The mod-minimizer uses parameter $r = 4$, and miniception uses parameter $\max(4, k - w)$. The window sizes 5 and 19 are the default window sizes for Kraken2 (Wood et al., 2019) and minimap2 (`-ax hifi`) (Li, 2018), respectively. For SSHash, $w = 12$ was the window size used when indexing the human genome (Pibiri, 2022).

mod-minimizer (Theorem 10 of Groot Koerkamp and Pibiri (2024)) is given by

$$\frac{\lfloor \frac{w+k-2}{w} \rfloor + 2}{w+k} + o(1/\ell).$$

The $o(1/\ell)$ term only accounts for duplicate t-mers, and hence disappears when $\sigma \to \infty$. We get

$$\frac{\lfloor \frac{w+k-2}{w} \rfloor + 2}{w+k} = \frac{\lfloor \frac{k+3w-2}{w} \rfloor}{w+k} = \frac{\lceil \frac{k+2w-1}{w} \rceil}{w+k} \overset{k\equiv 1 \pmod{w}}{=} \frac{\lceil \frac{w+k}{w} \rceil}{w+k}.$$

Thus, the mod-minimizer has density equal to the lower bound provided by Theorem 1 when σ goes to ∞ and w and $k \equiv 1 \pmod{w}$ are fixed.

In practice, for $\sigma = 256$ the mod-minimizer scheme is within 1% from optimal when $k \equiv 1 \pmod{w}$ (Fig. 4 in Supplement D). When $\sigma = 4$ (Fig. 3), a $t > 1$ must be used, causing the density plot to “shift right” compared to the lower bound. Because of that, the mod-minimizer does not quite match the lower bound for practical values of σ.

Table 1: Minimum densities achieved by existing sampling schemes for default parameters of frequently-used bioinformatics methods ($\sigma = 4$). The gap percentage describes the how much larger the lowest achieved density is than the lower bound and is calculated as $100 \cdot \frac{\mathrm{d}(f)-LB(w,k)}{LB(w,k)}$ where $LB(w,k) = \frac{1}{w}$ for the old gap and $LB(w,k) = g'_4(w,k)$ for the new gap. While Groot Koerkamp and Pibiri (2024) showed that $\frac{1.5}{w+k-0.5}$ is also a lower bound, $\frac{1}{w}$ is tighter for all of the parameter choices in the table. For SSHash (Pibiri, 2022), we show parameters used for indexing a single human genome.

Application	(w,k)	Random	Best		Lower bound		Gap (%)	
		$2/(w+1)$	Scheme	Density	$1/w$	g'	$1/w$	g'
Kraken2	$(5,31)$	0.333	Mod-mini	0.226	0.200	0.222	12.8	1.6
SSHash	$(12,20)$	0.154	Mod-mini	0.120	0.083	0.108	43.9	10.9
minimap2, `hifi`	$(19,19)$	0.100	dbl decycling	0.079	0.053	0.077	50.1	2.7

5 Discussion

5.1 Conjecture on when our lower bound is tight

Analytically, it is clear that $g'_\sigma(w,k)$ is much larger than $\frac{1}{w}$. In all cases, $g'_\sigma(w,k)$ is nearly tight, if not completely. In particular, our bound is tight for all 40 tested parameter sets where $k \equiv 1 \pmod{w}$, leading us to our conjecture:

Conjecture 1. *For any w and k satisfying $k \equiv 1 \pmod{w}$, there exists a (w,k)-forward sampling scheme f such that $\mathrm{d}(f) = g_\sigma(w,k)$.*

While the minimum size of a decycling set, i.e., a $(w{=}\infty, \ell)$-UHS, is well-known to be $N_\sigma(\ell)$ (Mykkeltveit, 1972), very little is known about the minimum size of a (w,ℓ)-UHS for finite w. In addition to providing the minimum density of a (w,k)-forward scheme for $k \equiv 1 \pmod{w}$, proving Conjecture 1 would also determine the minimum size of a $(w, \ell{=}w+k)$-UHS when $k \equiv 1 \pmod{w}$.

5.2 Existing schemes are nearly optimal when $k \geq w$ or σ is large

A natural investigation which follows our proposed lower bound is to determine the gap between $g'_\sigma(w,k)$ and current forward scheme densities. Previously, the gap between known densities and lower bound was rather large, making it unclear how much more the density could be reduced.

In Table 1, we observe that existing schemes are already within 11% from the optimal density for practical values of w and k across different applications, and in many cases are even within 3% of the optimal density. In Fig. 3, we see that this difference holds not just for the specific (w,k) in Table 1, but for most $k \geq w$. This is much more informative than the previous lower bound of $1/w$, which implied that most current schemes are at most 50% denser than optimal for many of the parameters in Fig. 3.

For alphabets much larger than DNA ($\sigma = 4$), such as the ASCII alphabet ($\sigma = 256$), we observe that when $k \equiv 1 \pmod{w}$, the mod-minimizer scheme recently proposed by Groot Koerkamp and Pibiri (2024) is at most 1% denser than optimal and furthermore, we show that it is optimal as $\sigma \to \infty$. This makes the mod-minimizer scheme the first practical scheme for which there exist finite parameters k and w for which it is close to optimal.

5.3 Tightening the bound for small k

Our new bound for forward schemes always improves over $1/w$ and appears tight when $k \equiv 1 \pmod{w}$. This leads to an increasingly close bound for $k \not\equiv 1 \pmod{w}$ as k increases, but leaves a large gap when $1 < k < w$. A better understanding of these small cases will be necessary to obtain a tight lower bound for all w and k. Looking at Fig. 3 and Fig. 4 in Supplement D, one might conjecture that the double decyling-set-based methods Pellow et al. (2023) are near-optimal, but subsequent work such as the greedy minimizer (Golan et al., 2024) has shown better schemes are possible. From Fig. 2, we already know that our lower bound is not always tight, so this leaves the question:

Open problem 1. *How close can practical sampling schemes get to the density given by our lower bound?*

5.4 Extending the bound to local schemes

For local schemes, though, our bound appears much less tight. We identified 8 sets of (w, k, σ) where local schemes can obtain lower densities than their forward counterparts. In all cases, however, the difference between the local and forward densities was minuscule, with the largest difference of being found for $(w = 4, k = 2, \sigma = 2)$ where the density decreased from 0.375 to 0.371 (Table 2 in Supplement D). Nevertheless, for some parameters, local schemes are able to achieve densities lower than our $g'_\sigma(w, k)$ lower bound for forward schemes. Given the trend observed in Table 2, we arrive at our final open problem:

Open problem 2. *How much lower can the density of a (w, k)-local scheme be compared to a (w, k)-forward scheme?*

Acknowledgements

We thank Giulio Ermanno Pibiri, Nicolae Sapoval, and our reviewers for their suggestions which helped improve the manuscript.

Conflict of interest

None declared.

Funding

This work was supported by the National Library of Medicine Training Program in Biomedical Informatics and Data Science [grant number T15LM007093 to B.K.]; in part, the National Institute of Allergy and Infectious Diseases [grant number P01-AI152999 to B.K. and T.J.T.], National Science Foundation (NSF) awards [IIS-2239114 and EF-2126387 to T.J.T]. R.G.K. is supported by ETH Research Grant ETH-1721-1 to Gunnar Rätsch.

References

D. DeBlasio, F. Gbosibo, C. Kingsford, and G. Marçais. Practical universal k-mer sets for minimizer schemes. In *Proceedings of the 10th ACM International Conference on Bioinformatics, Computational Biology and Health Informatics*, pages 167–176, 2019. doi: 10.1145/3307339.3342144.

R. Edgar. Syncmers are more sensitive than minimizers for selecting conserved k-mers in biological sequences. *PeerJ*, 9:e10805, 2021. doi: 10.7717/peerj.10805.

B. Ekim, B. Berger, and Y. Orenstein. A randomized parallel algorithm for efficiently finding near-optimal universal hitting sets. In *International Conference on Research in Computational Molecular Biology*, pages 37–53. Springer, 2020. doi: 10.1007/978-3-030-45257-5_3.

S. Golan, I. Tziony, M. Kraus, Y. Orenstein, and A. Shur. Generating low-density minimizers. *bioRxiv*, Nov. 2024. doi: 10.1101/2024.10.28.620726. URL `http://dx.doi.org/10.1101/2024.10.28.620726`.

R. Groot Koerkamp and G. E. Pibiri. The mod-minimizer: A simple and efficient sampling algorithm for long k-mers. In S. P. Pissis and W.-K. Sung, editors, *24th International Workshop on Algorithms in Bioinformatics (WABI 2024)*, volume 312 of *Leibniz International Proceedings in Informatics (LIPIcs)*, pages 11:1–11:23, Dagstuhl, Germany, 2024. Schloss Dagstuhl – Leibniz-Zentrum für Informatik. ISBN 978-3-95977-340-9. doi: 10.4230/LIPIcs.WABI.2024.11. URL `https://drops.dagstuhl.de/entities/document/10.4230/LIPIcs.WABI.2024.11`.

Gurobi Optimization, LLC. Gurobi optimizer reference manual, 2024. URL `https://www.gurobi.com`.

M. Hoang, H. Zheng, and C. Kingsford. Differentiable learning of sequence-specific minimizer schemes with DeepMinimizer. *Journal of Computational Biology*, 29(12):1288–1304, 2022. doi: 10.1089/cmb.2022.0275.

L. Irber, P. T. Brooks, T. Reiter, N. T. Pierce-Ward, M. R. Hera, D. Koslicki, and C. T. Brown. Lightweight compositional analysis of metagenomes with FracMinHash and minimum metagenome covers. *BioRxiv*, pages 2022–01, 2022. doi: 10.1101/2022.01.11.475838.

B. Kille, E. Garrison, T. J. Treangen, and A. M. Phillippy. Minmers are a generalization of minimizers that enable unbiased local Jaccard estimation. *Bioinformatics*, 39(9):btad512, 2023. doi: 10.1093/bioinformatics/btad512.

H. Li. Minimap2: pairwise alignment for nucleotide sequences. *Bioinformatics*, 34:3094–3100, 2018. doi: 10.1093/bioinformatics/bty191.

N. Lichiardopol. Independence number of de Bruijn graphs. *Discrete mathematics*, 306(12):1145–1160, 2006. doi: 10.1016/j.disc.2005.10.032.

G. Loukides and S. Pissis. Bidirectional string anchors: A new string sampling mechanism. In *ESA 2021-29th Annual European Symposium on Algorithms*, volume 204, pages 1–21, 2021. doi: 10.4230/lipics.esa.2021.64.

G. Loukides, S. P. Pissis, and M. Sweering. Bidirectional string anchors for improved text indexing and top-k similarity search. *IEEE Transactions on Knowledge and Data Engineering*, 35(11):11093–11111, 2023. doi: 10.1109/tkde.2022.3231780.

G. Marçais, D. Pellow, D. Bork, Y. Orenstein, R. Shamir, and C. Kingsford. Improving the performance of minimizers and winnowing schemes. *Bioinformatics*, 33(14):i110–i117, 2017. doi: 10.1093/bioinformatics/btx235.

G. Marçais, D. DeBlasio, and C. Kingsford. Asymptotically optimal minimizers schemes. *Bioinformatics*, 34(13):i13–i22, 2018. doi: 10.1093/bioinformatics/bty258.

C. Moreau. Sur les permutations circulaires distinctes. *Nouvelles annales de mathématiques: journal des candidats aux écoles polytechnique et normale*, 11:309–314, 1872.

J. Mykkeltveit. A proof of Golomb's conjecture for the de Bruijn graph. *Journal of Combinatorial Theory, Series B*, 13(1):40–45, 1972. doi: 10.1016/0095-8956(72)90006-8.

A. Möbius. Über eine besondere Art von Umkehrung der Reihen. *Journal für die reine und angewandte Mathematik*, 1832(9):105–123, 1832. doi: doi:10.1515/crll.1832.9.105. URL `https://doi.org/10.1515/crll.1832.9.105`.

M. Ndiaye, S. Prieto-Baños, L. M. Fitzgerald, A. Yazdizadeh Kharrazi, S. Oreshkov, C. Dessimoz, F. J. Sedlazeck, N. Glover, and S. Majidian. When less is more: sketching with minimizers in genomics. *Genome Biology*, 25(1):270, 2024. doi: 10.1186/s13059-024-03414-4.

Y. Orenstein, D. Pellow, G. Marçais, R. Shamir, and C. Kingsford. Compact universal k-mer hitting sets. In *Algorithms in Bioinformatics: 16th International Workshop, WABI 2016, Aarhus, Denmark, August 22-24, 2016. Proceedings 16*, pages 257–268. Springer, 2016. doi: 10.1007/978-3-319-43681-4_21.

D. Pellow, L. Pu, B. Ekim, L. Kotlar, B. Berger, R. Shamir, and Y. Orenstein. Efficient minimizer orders for large values of k using minimum decycling sets. *Genome Research*, 33(7):1154–1161, 2023. doi: 10.1101/gr.277644.123.

G. E. Pibiri. Sparse and skew hashing of k-mers. *Bioinformatics*, 38:i185–i194, June 2022. ISSN 1367-4811. doi: 10.1093/bioinformatics/btac245. URL `http://dx.doi.org/10.1093/bioinformatics/btac245`.

A. Rhie, S. Nurk, M. Cechova, S. J. Hoyt, D. J. Taylor, N. Altemose, P. W. Hook, S. Koren, M. Rautiainen, I. A. Alexandrov, et al. The complete sequence of a human Y chromosome. *Nature*, 621(7978):344–354, 2023. doi: 10.1038/s41586-023-06457-y.

J. Riordan. The combinatorial significance of a theorem of Pólya. *Journal of the Society for Industrial and Applied Mathematics*, 5(4):225–237, Dec. 1957. ISSN 2168-3484. doi: 10.1137/0105015. URL `http://dx.doi.org/10.1137/0105015`.

M. Roberts, B. R. Hunt, J. A. Yorke, R. A. Bolanos, and A. L. Delcher. A preprocessor for shotgun assembly of large genomes. *Journal of computational biology*, 11(4):734–752, 2004.

S. Schleimer, D. S. Wilkerson, and A. Aiken. Winnowing: local algorithms for document fingerprinting. In *Proceedings of the 2003 ACM SIGMOD international conference on Management of data*, pages 76–85, 2003. doi: 10.1145/872757.872770.

J. Shaw and Y. W. Yu. Theory of local k-mer selection with applications to long-read alignment. *Bioinformatics*, 38(20):4659–4669, 2022. doi: 10.1093/bioinformatics/btab790.

D. E. Wood, J. Lu, and B. Langmead. Improved metagenomic analysis with Kraken 2. *Genome biology*, 20:1–13, 2019. doi: 10.1186/s13059-019-1891-0.

H. Zheng, C. Kingsford, and G. Marçais. Improved design and analysis of practical minimizers. *Bioinformatics*, 36:i119–i127, 2020. doi: 10.1093/bioinformatics/btaa472.

H. Zheng, C. Kingsford, and G. Marçais. Lower density selection schemes via small universal hitting sets with short remaining path length. *Journal of Computational Biology*, 28(4):395–409, 2021a. doi: 10.1089/cmb.2020.0432.

H. Zheng, C. Kingsford, and G. Marçais. Sequence-specific minimizers via polar sets. *Bioinformatics*, 37:i187–i195, 2021b. doi: 10.1093/bioinformatics/btab313.

H. Zheng, G. Marçais, and C. Kingsford. Creating and using minimizer sketches in computational genomics. *Journal of Computational Biology*, 30(12):1251–1276, 2023. doi: 10.1089/cmb.2023.0094.

A (s, w, k)-multi-local schemes

Given a set A, we define $\binom{A}{s}$ as the set of all subsets of A of cardinality s.

Definition 8. *A (s, w, k)-multi-local scheme corresponds to a sampling function $f : \Sigma^{w+k-1} \to \binom{[w]}{s}$, where s is the sketch size, w is the window guarantee, k is the k-mer size, and $\sigma = |\Sigma|$ is the alphabet size.*

A quick substitution reveals that (s, w, k)-multi-local schemes are equivalent to (w, k)-local schemes when $s = 1$. These schemes enable a stricter window guarantee by ensuring that at least s k-mers are selected for every window of w k-mers. However, it was recently shown that this generalization can also be used to outline a more relaxed window guarantee (Kille et al., 2023).

Consider the case where instead of desiring at least one sampled k-mer for every w k-mers, the requirement is to have s k-mers sampled for every sw k-mers. This latter goal can be accomplished by a (w, k)-local scheme, but as was shown in Kille et al. (2023), a much lower density can be obtained with an (s, sw, k)-multi-local scheme.

Here, we will show how the bounds in the main text can be extended to yield a lower bound for (s, w, k)-multi-local schemes.

A (s, w, ℓ)-multi-UHS is defined as a mapping $\alpha : \Sigma^\ell \to [s + 1]$ that assigns weights to vertices in B_ℓ such that any sequence of w adjacent ℓ-mers must have a combined weight of at least s. Again, when $s = 1$, this corresponds to the (w, ℓ)-UHS described in the main text. We now provide an extension to Proposition 4 which provides a lower bound on the size of a (s, w, ℓ)-multi-UHS.

Proposition 9. *For any (s, w, ℓ)-multi-UHS α,*

$$\sum_{W \in \Sigma^\ell} \alpha(W) \geq \sum_{p|\ell} M_\sigma(p) \cdot \left\lceil \frac{sp}{w} \right\rceil.$$

Proof. We first show that for any simple cycle of size p in B_ℓ, the combined weight of the cycle must be at least $\left\lceil \frac{sp}{w} \right\rceil$. Let $W_0, \ldots, W_{p-1}$ be the vertices in a cycle of length p. Consider the walk of length w along the cycle starting at W_0. We have that $\sum_{i\in[w]} \alpha(W_i) \geq s$ where indices are taken modulo p. This same inequality holds for all p unique paths of length w along the cycle and summing over all of them yields $\sum_{i\in[p]} \alpha(W_i) \geq \frac{sp}{w}$. Finally, since $\sum_{i\in[p]} \alpha(W_i)$ must be an integer and each pure cycle in $\mathcal{C}_\ell$ of length p corresponds to an aperiodic necklace of length p, of which there are $M_\sigma(p)$, we arrive at the result:

$$\sum_{W \in \Sigma^\ell} \alpha(W) \geq \sum_{x \in \mathcal{C}_\ell} \left\lceil \frac{s \cdot |x|}{w} \right\rceil = \sum_{p|\ell} M_\sigma(p) \left\lceil \frac{sp}{w} \right\rceil. \qquad \square$$

The multi-local schemes also require that we generalize our definition of a charged context. As a multi-local scheme can select multiple new k-mers in a single window, we change our binary notion of a charged context to a mapping of weights to contexts, where a context W has weight $\alpha(W)$ if $\alpha(W)$ previously unsampled positions are selected in the final window of length $w + k - 1$ in W. Similar to local schemes, a multi-local scheme's necessary context is the current window of w k-mers as well as the previous $w - 1$ windows, leading to a context size of $2w + k - 2$.

Lemma 10. *If f is a (s, w, k)-multi-local scheme and $\alpha : \Sigma^{2w+k-2} \to [s + 1]$ is the corresponding mapping of weights to contexts, defined as $\alpha(W) = |A \setminus B|$, where A consists of all positions in the context selected by the final window and B consists of all positions in the context selected in all previous windows, i.e.*

$$A = \{j + (w - 1) \mid j \in f(W[w - 1, 2w + k - 2))\}$$

$$B = \left\{ \bigcup_{0 \leq i \leq w-2} \{j + i \mid j \in f(W[i, i + w + k - 1))\} \right\},$$

then α is a $(s, w, 2w + k - 2)$-multi-UHS.

Proof. Our proof follows a similar structure to that of Lemma 5. Let us show that the total weight of any path of length w in B_{2w+k-2} is at least s. Let a sequence of w consecutive contexts of length $2w + k - 2$ be given as $(W_0, \dots, W_{w-1})$. Take S to be the sequence of length $3w + k - 3$ such that $S[i, 2w + k - 2 + i) = W_i$. Then we have that f on the last $(w + k - 1)$-mer of W_{w-1} (which is $S[2w - 2, 3w + k - 3)$) selects s distinct indices $i_1, i_2, \dots, i_s$ in S where for each $1 \leq p \leq s$ we have $i_p \geq 2w - 2$. Suppose n of these indices are selected by f on any previous $(w + k - 1)$-mer of S, indexed $i_{q_1}, \dots, i_{q_n}$, while the other $s - n$ i_p are not (meaning $\alpha(W_{w-1}) = s - n$). Clearly, if $n = 0$, we are done. Therefore, assume instead $n > 0$.

Recall that for all $1 \leq j \leq n$, $i_{q_j} \geq 2w - 2$. Since $0 \leq f(\cdot) \leq w - 1$, we have that for each j, the first $(w + k - 1)$-mer $S[m_j, m_j + w + k - 1)$ in S such that f picks the index i_{q_j} satisfies $w - 1 \leq m_j \leq 2w - 2$. Then for each j, the choice of m_j yields that $W_{m_j - w+1}$ selects a new location i_{q_j} when f is applied to its last $(w + k - 1)$-mer. This means that $\sum\limits_{l \in [w-1]} \alpha(W_l) \geq n$, since the sum accounts for every context of the form $W_{m_j - w+1}$, and hence every chosen index $i_{q_1}, \dots, i_{q_n}$. As $\alpha(W_{w-1}) = s - n$, we have that the sum of the weights of the w contexts $W_0, \dots, W_{w-1}$ is at least s. □

Tying these results together using the same line of reasoning as we did in the main text for local schemes, we arrive at a lower bound for multi-local schemes:

Corollary 11. *If f is a (s, w, k)-multi-local scheme, then*

$$\mathrm{d}(f) \geq \frac{1}{\sigma^{2w+k-2}} \sum_{p | 2w+k-2} M_\sigma(p) \left\lceil \frac{sp}{w} \right\rceil.$$

Let us consider the use case of multi-local schemes described earlier, i.e., where the requirement is to have s k-mers sampled from every sw k-mers. We showed in the main text that this can be accomplished by a (w, k)-forward scheme with density at least $g_\sigma(w, k)$, or a (w, k)-local scheme with density at least $g_\sigma(w, w + k - 2)$. With our new bound, we can see that the requirement can be achieved with a (s, sw, k)-multi-local scheme with density at least $g_\sigma(w, k + (2s - 1)w - 2)$. In other words, the lower bound for local schemes is the same pattern as the forward bound, but “shifted left" by $w - 2$, and the bound for multi-local is the same as the local bound, but again shifted left by $(2s - 2)w$.

B An alternative form of $g_\sigma(w, k)$

When all divisors of ℓ apart from 1 have the same remainder modulo w, we can simplify $g_\sigma(w, k)$.

Corollary 12. *Let $N_\sigma(\ell)$ denote the number of cycles in the pure cycle partitioning of B_ℓ. Let w, ℓ be a pair of integers such that w does not divide ℓ and for all divisors $d \mid \ell$ excluding the unit divisor* 1*, $d \equiv z \pmod{w}$. Then for any (w, ℓ)-UHS U, $|U| \geq \frac{\sigma^\ell + N_\sigma(\ell)(w-z) + \sigma(z-1)}{w}$.*

Proof. There are σ singleton cycles in a De Bruijn graph on an alphabet of σ characters, and each of these must be included in any hitting set U. For all remaining cycles, we have that $\lceil |c|/w \rceil = (|c| + w - z)/w$.

$$
\begin{aligned}
|U| &\geq \sum_{c \in \mathcal{C}_\ell} \left\lceil \frac{|c|}{w} \right\rceil \\
&= \sigma + \left(\sum_{c \in \mathcal{C}_\ell} \frac{|c| + w - z}{w} \right) - \sigma \frac{1 + w - z}{w} \\
&= \sigma \left(1 - \frac{w - z + 1}{w} \right) + \sum_{c \in \mathcal{C}_\ell} \frac{|c|}{w} + \sum_{c \in \mathcal{C}_\ell} \frac{w - z}{w} \\
&= \sigma \left(\frac{z - 1}{w} \right) + \frac{\sigma^\ell}{w} + N_\sigma(\ell) \frac{w - z}{w} \\
&= \frac{\sigma^\ell + N_\sigma(\ell)(w - z) + \sigma(z - 1)}{w}. \qquad \square
\end{aligned}
$$

While the formula in Corollary 12 still includes a summation over divisors due to its use of N_σ, it no longer involves any ceiling calculations. Furthermore, it shows that $g_\sigma(w, k)$ can be written as the following when w and k satisfy the constraints of Corollary 12

$$
g_\sigma(w, k) = \frac{1}{w} + \frac{N_\sigma(w + k)(w - z) + \sigma(z - 1)}{w\sigma^{w+k}}.
$$

This form, compared to the general form of $g_\sigma(w, k)$, provides a more interpretable characterization of the gap between g_σ and $\frac{1}{w}$.

C ILP Model

Forward ILP definition. We used Gurobi (Gurobi Optimization, LLC, 2024) to implement our ILP. While the basic model described below is sufficient, we made multiple improvements which enabled identifying solutions for larger σ, w and k. We use x_u to model $f(u)$, i.e. the index of the k-mer selected in the window $u \in \Sigma^{w+k-1}$. The $y_{u,v}$ variables represent the charge of a context of two adjacent windows u and v.

Given a De Bruijn graph $B_{w+k-1} = (V, E)$, the windows correspond to the vertices of the graph, and the contexts correspond to the edges. An edge (u, v) is not charged if $x_u = x_v + 1$. For a forward scheme, $x_u \leq x_v + 1$ for every edge (u, v). Therefore, we can define an ILP which minimizes the density of a (w, k)-forward scheme as follows:

$$
\begin{aligned}
&\text{minimize} \sum_{(u,v) \in E} y_{(u,v)} \\
&\text{such that} \\
&\qquad x_u \in [w] && \forall u \in V, \\
&\qquad y_{(u,v)} \in \{0, 1\} && \forall (u, v) \in E, \\
&\qquad x_u \leq x_v + 1 && \forall (u, v) \in E, \\
&\qquad y_{(u,v)} = 0 \implies x_u = x_v + 1 && \forall (u, v) \in E.
\end{aligned}
$$

Local ILP definition. While we could define an ILP for local schemes similarly, the resulting model is inefficient due to the reliance on intermediate variables. Instead, we leverage the fact that the expected density on a random string (Marçais et al., 2017) is the same as the density of a (w, k)-local scheme on a circular De Bruijn sequence of order $2w + k - 2$.

Let S be circular De Bruijn sequence of order $2w + k - 2$, i.e. S is a circular sequence of length $L = \sigma^{2w+k-2}$ which contains every $(2w + k - 2)$-mer exactly once. As S is circular, we note that when

$i > j$, the substring $S[i..j)$ corresponds to $S[i..L)S[0..j)$, We use x_W to represent the value of $f(W)$ and y_i to be a binary variable which corresponds to whether or not position i was sampled. We define an ILP which minimizes the density of a (w, k)-local scheme as follows:

$$\begin{aligned} &\text{minimize} \sum_{i\in[L]} y_i \\ &\text{such that} \\ &x_W \in [w] && \forall W \in \Sigma^{w+k-1}, \\ &y_i \in \{0,1\} && \forall i \in [L], \\ &x_{S[i,(i+w+k-1) \bmod L)}{=}j \implies y_{(i+j) \bmod L}{=}1 && \forall i{\in}[L],\ j{\in}[w]. \end{aligned}$$

Improvements to the base forward ILP. First, we added the additional constraint that each pure cycle c in B_{w+k} must have at least $\lceil |c|/w \rceil$ nodes which correspond to charged edges in B_{w+k-1}. This additional constraint helps substantially when $k \equiv 1 \pmod w$. In cases where $k \not\equiv 1 \pmod w$, we add corresponding constraints for a subset of simple cycles in B_{w+k} that correspond to pure cycles in higher or lower order De Bruijn graphs. While this process adds many more constraints to the model, it narrows the search space substantially and also increases the objective of the ILP linear relaxation, leading to a smaller (and sometimes nonexistent) integrality gap.

As adding this constraint requires the model to have variables representing the charge of a context, it was more efficient to directly use the context charges as the objective variable as opposed to using selected positions in a De Bruijn sequence (as we do in the local ILP). For the local ILP, the g_σ bound is relatively loose and therefore the aforementioned cycle constraints do not help much. Furthermore, modeling context charges requires more intermediate variables in the local ILP, hence we directly used sampled positions in a De Bruijn sequence as the objective for the local ILP.

If $g_\sigma(w, k)$ is tight for some σ, w, k, then the objective value of the linear relaxation of the ILP is the same as the integer objective value. As a result, the objective bound is fixed from the start. We leveraged this fact by telling the optimizer to focus on identifying integral solutions as opposed to decreasing the gap between the objective bound and objective value in cases where we suspect that $g_\sigma(w, k)$ is tight. This was done through the `heuristics` parameter.

Finally, we used previously computed solutions to provide starting points for the ILP optimization. Let f be a (w, k)-local scheme. We constructed a $(w+1, k)$-local scheme or a $(w, k+1)$-local scheme f' through ignoring the last character in an input window W of length $w+k$, i.e. $f'(W) = f(W[0, w+k))$. This is similar to the "naive extensions" described in Marçais et al. (2018). If we had a minimum density $(w-1, k)$-forward scheme or a $(w, k-1)$-forward scheme, we seeded the optimizer with the "extended" sampling function which corresponded to whichever one of the precursors schemes yielded the lower density.

We ran our ILP model on a server with 128 threads and 128GB of RAM on all combinations of $2 \leq w \leq 12$, $1 \leq k \leq 12$, and $2 \leq \sigma \leq 4$. For each set of parameters, we limited the runtime to 12 hours. The ILP identified optimal forward schemes for 60 different sets of parameter and optimal local schemes for 11 different sets of parameters.

D Additional figures

See Fig. 4 and Table 2.

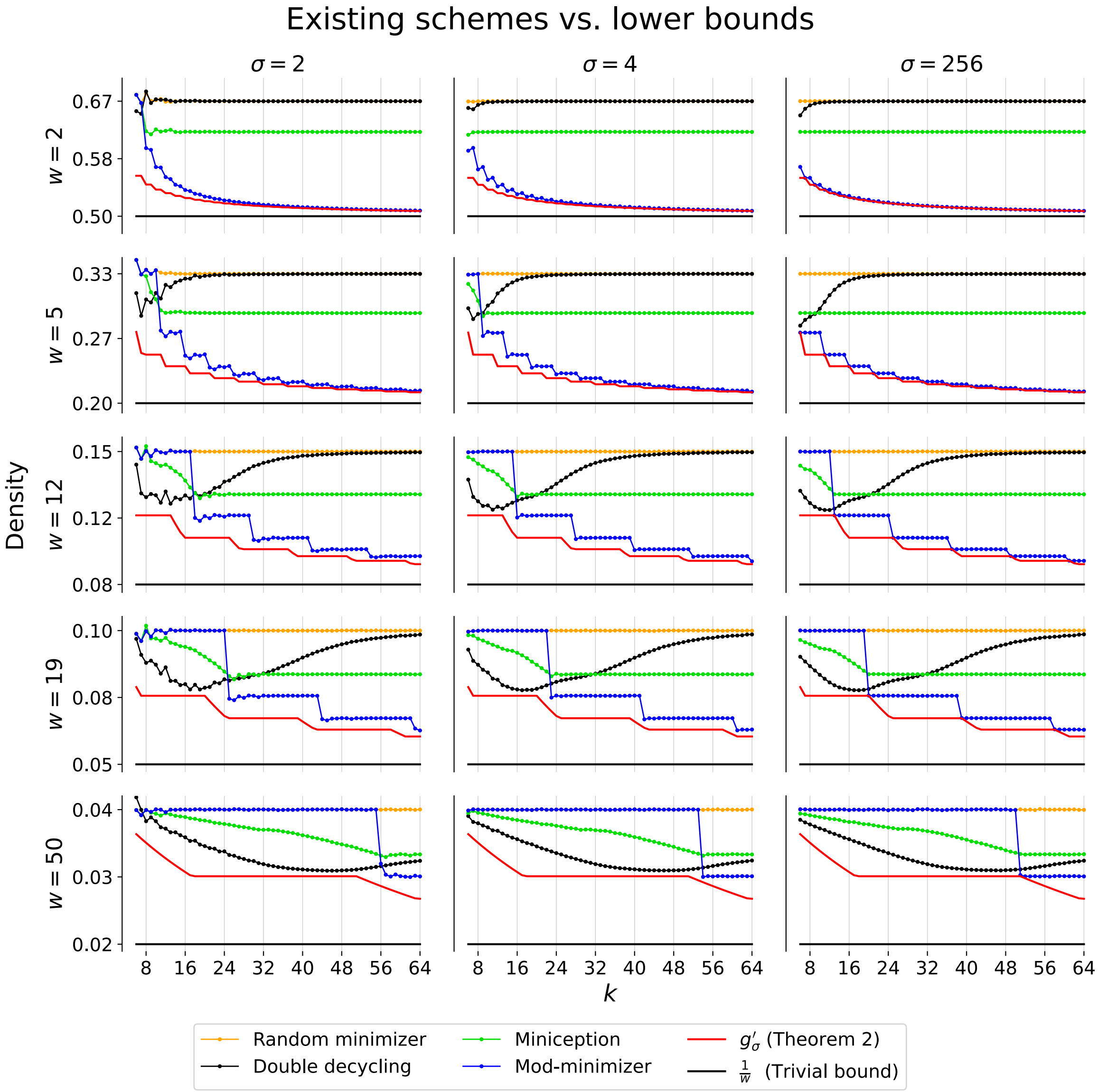

Figure 4: Comparison of existing schemes to lower bounds. Densities were calculated by applying each scheme to a random sequence of 10 million characters and are plotted as solid dotted lines. The mod-minimizer uses parameter $r = 6$ for $\sigma = 2$, $r = 4$ for $\sigma = 4$, and $r = 1$ for $\sigma = 256$. Miniception uses parameter $\max(r, k - w)$. Lower bounds are plotted as solid lines.

Table 2: Our $g'_\sigma(w,k)$ lower bound on the density of forward schemes, and minimum densities of forward and local obtained by the ILPs described in Supplement C. Entries with an equals sign (=) correspond to parameters where the minimum density is equal to the preceding column. Entries with the less-than-or-equal-to sign ($\leq$) correspond to parameters where the local ILP identified a solution with lower density than the optimal forward scheme but timed out before determining whether the identified solution was optimal. Bold entries indicate cases where the local scheme is better than the $g'(w,k)$ bound for forward schemes. For $w=2$, all local schemes are forward by definition and therefore the minimum $(2,k)$-local scheme density is equal to the minimum $(2,k)$-forward scheme density. Empty cells correspond to parameters where the local ILP timed out before it was able identify a local scheme with lower density than the optimal forward scheme or prove that no such scheme exists. In these cases, it is unknown whether or not there exists a local scheme with lower density than the optimal forward scheme.

(a) $\sigma = 2$

w,k	$g'_\sigma(w,k)$		Minimum d(f)			
	Fwd. bound		Forward		Local	
2, 1	0.750	3/4	=		=	
2, 2	0.625	5/8	=		=	
2, 3	0.625	5/8	=		=	
2, 4	0.578	37/64	=		=	
2, 5	0.578	37/64	=		=	
2, 6	0.559	143/256	=		=	
2, 7	0.559	143/256	=		=	
2, 8	0.546	559/1024	=		=	
2, 9	0.546	559/1024	=		=	
2, 10	0.539	1103/2048	=		=	
2, 11	0.539	1103/2048	=		=	
3, 1	0.562	9/16	=		=	
3, 2	0.438	7/16	0.469	15/32	=	
3, 3	0.438	7/16	0.453	29/64	=	
3, 4	0.438	7/16	=		=	
3, 5	0.401	411/1024	0.406	13/32	=	
3, 6	0.401	411/1024	0.402	103/256		
3, 7	0.401	411/1024	=			
3, 8	0.385	197/512	0.386	395/1024		
3, 9	0.385	197/512	0.385	1577/4096		
3, 10	0.385	197/512	=			
4, 1	0.438	7/16	=		=	
4, 2	0.359	23/64	0.375	3/8	0.371	95/256
4, 3	0.336	43/128	0.352	45/128	=	
4, 4	0.336	43/128	0.340	87/256		
4, 5	0.336	43/128	=		≤ **0.335**	**687/2048**
4, 9	0.308	1261/4096	=		≤ **0.308**	**10087/32768**
5, 1	0.359	23/64	=		**0.355**	**91/256**
5, 2	0.297	19/64	0.305	39/128	=	
5, 6	0.273	35/128	=		≤ **0.273**	**2239/8192**
6, 1	0.297	19/64	=			
6, 2	0.258	33/128	0.262	67/256	≤ 0.261	267/1024
7, 1	0.258	33/128	=		≤ **0.257**	**263/1024**
8, 1	0.227	29/128	=			
9, 1	0.202	207/1024	=			
10, 1	0.183	187/1024	=			
11, 1	0.168	687/4096	=			
12, 1	0.154	631/4096	=			

(b) $\sigma = 3$

w,k	$g'_\sigma(w,k)$		Minimum d(f)			
	Fwd. bound		Forward		Local	
2, 1	0.704	19/27	=		=	
2, 2	0.605	49/81	0.630	17/27	=	
2, 3	0.605	49/81	=		=	
2, 4	0.572	139/243	0.578	421/729	=	
2, 5	0.572	139/243	=		=	
2, 7	0.556	10939/19683	=		=	
3, 1	0.519	14/27	=		=	
3, 2	0.429	313/729	0.457	37/81		
3, 4	0.429	313/729	=		≤ **0.428**	**2810/6561**
4, 1	0.407	11/27	=			
5, 1	0.337	82/243	=			
6, 1	0.287	209/729	=			
7, 1	0.251	548/2187	=			
8, 1	0.222	4379/19683	=			

(c) $\sigma = 4$

w,k	$g'_\sigma(w,k)$		Minimum d(f)		
	Fwd. bound		Forward		Local
2, 1	0.688	11/16	=		=
2, 2	0.602	77/128	0.621	159/256	=
2, 3	0.602	77/128	=		=
2, 5	0.572	2341/4096	=		=
3, 1	0.508	65/128	=		
4, 1	0.402	103/256	=		
5, 1	0.334	685/2048	=		
6, 1	0.286	1171/4096	=		