Bioinformatics
doi.10.1093/bioinformatics/xxxxxx
Advance Access Publication Date: Day Month Year
Manuscript Category

Subject Section

The minimizer Jaccard estimator is biased and inconsistent*

Mahdi Belbasi [1], Antonio Blanca [1], Robert S. Harris [2], David Koslicki [1, 2, 3], and Paul Medvedev [1, 3, 4] *

[1]Department of Computer Science and Engineering, The Pennsylvania State University,
[2]Department of Biology, The Pennsylvania State University,
[3]Huck Institutes of the Life Sciences, The Pennsylvania State University, and
[4]Department of Biochemistry and Molecular Biology, The Pennsylvania State University

*Corresponding author, pzm11@psu.edu.

Associate Editor: XXXXXXX

Received on XXXXX; revised on XXXXX; accepted on XXXXX

Abstract

Motivation: Sketching is now widely used in bioinformatics to reduce data size and increase data processing speed. Sketching approaches entice with improved scalability but also carry the danger of decreased accuracy and added bias. In this paper, we investigate the minimizer sketch and its use to estimate the Jaccard similarity between two sequences.

Results: We show that the minimizer Jaccard estimator is *biased* and *inconsistent*, which means that the expected difference (i.e., the bias) between the estimator and the true value is not zero, even in the limit as the lengths of the sequences grow. We derive an analytical formula for the bias as a function of how the shared k-mers are laid out along the sequences. We show both theoretically and empirically that there are families of sequences where the bias can be substantial (e.g. the true Jaccard can be more than double the estimate). Finally, we demonstrate that this bias affects the accuracy of the widely used mashmap read mapping tool.

Availability: Scripts to reproduce our experiments are available on GitHub [26].

Contact: pzm11@psu.edu

1 Introduction

Sketching is a powerful technique to drastically reduce data size and increase data processing speed. Sketching techniques create a smaller representation of the full dataset, called a *sketch*, in a way that makes algorithms more efficient, ideally without much loss of accuracy. This property has led to sketching methods being increasingly used to meet the scalability challenges of modern bioinformatics datasets, though sometimes without understanding the detrimental effects on accuracy.

A thorough treatment of sketching in bioinformatics can be found in the excellent surveys of [29, 21], but we mention a few notable examples next. The seminal Mash paper [25] showed how estimating the Jaccard similarity of two sequences from their minhash sketches [3] enables clustering of sequence databases at unprecedented scale. The hyperloglog sketch [10] is used to compute genomic distances [1]; the modulo sketch [34] is used to search sequence databases [27]; strobemers [30] and minhash with optimal densification [36, 39] are used for sequence comparison; order minhash is used to estimate edit distance [19]; and count minsketch [5] is used for k-mer counting [6].

One of the most widely used sketches, which forms the basis of our work, is the minimizer sketch [34, 28], which selects, for each window of w consecutive k-mers, the k-mer with the smallest hash

*Authors are listed in alphabetical order.

value. Minimizer sketches are used for transcriptome clustering [31] and error correction [32], as well as for seed generation by the Peregrine genome assembler [4] and the widely used minimap [16, 17] and mashmap [12, 13] aligners.

Just as with other sketching techniques, in order for the minimizer sketch to be useful, it must come with theoretical (or at least empirical) bounds on the loss of accuracy that results from its use. For instance, the minhash Jaccard estimator used by Mash has the property of being *unbiased* [3], i.e. its expected value is equal to the true Jaccard. Such a theoretical guarantee, however, cannot be assumed for other sketches. Here, we will consider the example of the *minimizer Jaccard estimate* [12, 13, 15], which computes the Jaccard similarity using minimizer sketches and forms the basis of the widely used mashmap [12, 13] aligner. This estimator is useful for sequence alignment because the minimizer sketch has the nice property that, roughly speaking, the sketch of a long string contains the sketches of all its substrings. However, its theoretical accuracy has not been studied and empirical evaluations have been limited.

In this paper, we study the accuracy of the minimizer Jaccard estimator $\widehat{J}$, both theoretically and empirically. We prove that $\widehat{J}$ is in fact biased and inconsistent (i.e. the bias is not zero, and it remains so even as the sequences lengths grow). We derive an approximate formula for the bias that is accurate up to a vanishingly small additive error term, and give families of sequence pairs for which $\widehat{J}$ is expected to be only between 40% to 63% of the true Jaccard. We then empirically evaluate the extent of the bias and find that in some cases, when the true Jaccard similarity is 0.90, the estimator is only 0.44. We also study both theoretically and empirically the bias of $\widehat{J}$ for pairs of sequences generated by a simple mutation process and find that, while not as drastic, the bias remains substantial. Finally, we show that the bias affects the mashmap aligner by causing it to output incorrect sequence divergence estimates, with up to a 14% error. Our results serve as a cautionary tale on the necessity of understanding the theoretical and empirical properties of sketching techniques.

2 The minimizer sketch and minimizer Jaccard estimator

In this section, we will define the minimizer sketch [34, 28] and the Jaccard estimator derived from it [12]. Let $k > 2$ and $w > 2$ be two integers. This paper will assume that we are given two duplicate-free sequences A and B of L k-mers, with $L \geqslant 7(w+1)$. A sequence is *duplicate-free* if it has no duplicate k-mers, but A and B are allowed to share k-mers. These requirements on the sequences do not limit the general scope of our results. In particular, since we will show the existence of bias for these constrained cases, it immediately implies the existence of bias within broader families of sequences.

Let A_i denote the k-mer starting at position i of A, with A_0 and A_{L-1} being the first and last k-mers, respectively. Let $\mathrm{Sp}^k(A)$ be the set of all k-mers in A. We define $I(A,B)$ to be the number of k-mers shared between A and B, and $U(A,B)$ to be the number of k-mers appearing in either A or B. Formally,

$$I(A,B) \triangleq |\mathrm{Sp}^k(A) \cap \mathrm{Sp}^k(B)|$$
$$U(A,B) \triangleq |\mathrm{Sp}^k(A) \cup \mathrm{Sp}^k(B)|$$

The Jaccard similarity between the sequences A and B is defined as

$$J(A,B) \triangleq \frac{I(A,B)}{U(A,B)}.$$

Suppose we have a hash function h that takes an element from the set of all k-mers and maps it to a real number drawn uniformly at random from the unit interval $[0,1]$. Under this hash function, the probability of a collision is 0. We denote by a_i the hash value assigned to k-mer A_i and for integer $w \geqslant 2$ define the minimizer sketch of A as

$$\mathrm{MS}(A;w) \triangleq \bigcup_{i=0}^{L-w} \left\{ A_p : p = \underset{j \in [i, i+w-1]}{\arg\min}\ a_j \right\}.$$

An element in $\mathrm{MS}(A;w)$ is called a *minimizer* of A. The minimizer intersection and the minimizer union of A and B are defined, respectively, as

$$\widehat{I}(A,B;w) \triangleq |\mathrm{MS}(A;w) \cap \mathrm{MS}(B;w)|$$
$$\widehat{U}(A,B;w) \triangleq |\mathrm{MS}(A;w) \cup \mathrm{MS}(B;w)|.$$

The minimizer Jaccard estimator between A and B is defined as

$$\widehat{J}(A,B;w) \triangleq J(\mathrm{MS}(A;w), \mathrm{MS}(B;w)) = \frac{\widehat{I}(A,B;w)}{\widehat{U}(A,B;w)}.$$

3 Main theoretical results

In this section, we state our main theoretical results and give some intuition behind them. We can think of the relationship between the shared k-mers of A and B as the subset of $(A_0, \ldots, A_{L-1}) \times (B_0, \ldots, B_{L-1})$ that corresponds to pairs of equal elements; i.e., to pairs (A_i, B_j) with $A_i = B_j$. Because A and B are duplicate-free, this relationship is a matching. We call this the *k-mer-matching* between A and B. Our main result is stated in terms of a term denoted by $\mathcal{B}(A,B;w)$, which is a deterministic function of the window size w and of the k-mer-matching between A and B. We postpone the exact definition of $\mathcal{B}(A,B;w)$ until Appendix A.1, since it requires the introduction of cumbersome notation. The main technical result of this paper is:

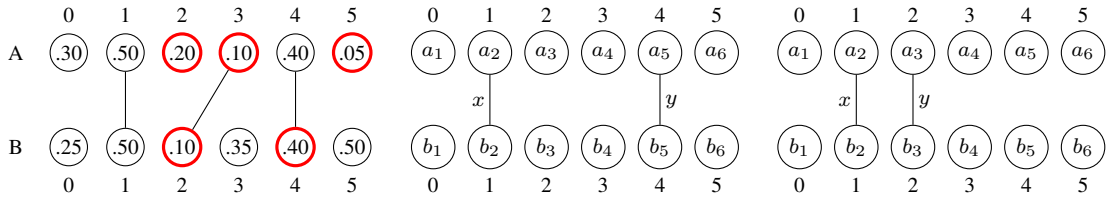
Theorem 1. *Let $w \geqslant 2$, $k \geqslant 2$, and $L \geqslant 7(w+1)$ be integers. Let A and B be two duplicate-free sequences, each consisting of L k-mers. Then there exists $\varepsilon \in [0, \frac{15w^2}{\sqrt[3]{L}}]$ such that*

$$\mathcal{B}(A,B,w) - \varepsilon \leqslant \mathbb{E}[\widehat{J}(A,B;w)] - J(A,B) \leqslant \mathcal{B}(A,B,w) + \varepsilon.$$

In other words, the difference between the expected value of the minimizer Jaccard estimator and the true Jaccard is $\mathcal{B}(A,B;w)$, up to a vanishingly small additive error. We now investigate the value of the term $\mathcal{B}$, which approximates the bias. First, we can show that for padded sequences, $\mathcal{B}(A,B;w) < 0$, except that $\mathcal{B}(A,B;w) = 0$ when $J(A,B) = 0$. We say two sequences are padded if they do not share any minimizers in the first or last w k-mers. (We note that the effect of padding becomes negligible for longer sequences.)

Theorem 2. *Let $w \geqslant 2$, $k \geqslant 2$, and $L \geqslant 7(w+1)$ be integers. Let A and B be two duplicate-free padded sequences, each consisting of L k-mers. Then $\mathcal{B}(A,B;w) < 0$ unless $J(A,B) = 0$; when $J(A,B) = 0$, we have $\mathcal{B}(A,B;w) = 0$.*

Moving forward, we may omit A, B, and w from our notation when they are obvious from the context. Theorems 1 and 2 state that $\widehat{J}$



Example 1 Example 2a Example 2b

Fig. 1: Examples of the Jaccard and the minimizer Jaccard estimator. Each example shows the k-mers of a sequence A on top, the k-mers of a sequence B on the bottom, and lines connecting k-mers show the *k-mer-matching* between A and B. Each k-mer is labeled by its hash value. In Example 1, $J(A,B) = 1/3$. The minimizers for $w = 3$ are circled in bold red. Here, $\widehat{I}(A,B;3) = 1$, $\widehat{U}(A,B;3) = 4$, and $\widehat{J}(A,B;3) = 1/4$. Examples 2a and 2b give intuition for why the minimizer Jaccard estimator is biased. Here, a_i refers to the hash value assigned to position i and x and y are k-mers shared between A and B. The expected minimizer Jaccard for $w = 2$ is different in the two examples but the Jaccard is not ($J = 0.2$); hence the expected minimizer Jaccard cannot be equal to the true Jaccard.

is biased for padded sequences as long as ε is sufficiently small (e.g. L is sufficiently large or w is sufficiently small). Here, we use "biased" in the statistical sense that $\mathbb{E}[\widehat{J}] \neq J$. Intuitively, $\widehat{J}$ is biased because it depends on the layout of the shared k-mers along the sequences (i.e. on the k-mer-matching), while J only depends on the number of shared k-mers but not on their layout. Note that our results hold for any duplicate-free choice of A and B and do not assume any background distribution, e.g. that A is generated uniformly at random.

We illustrate the point with Examples 2a and 2b in Figure 1. In both examples, the expected size of $\widehat{I}$ is the probability that x is a minimizer in A and in B plus the probability that y is a minimizer in A and in B. These two probabilities are equal to each other in these examples and $\mathbb{E}[\widehat{I}] = 2\tau$, for some τ. When $w = 2$, in Example 2a, τ is the probability that a_1 is the smallest hash value out of five independently chosen hash values a_0, a_1, a_2, b_0, and b_2. In Example 2b, however, $b_2 = a_2$, and τ is the probability that a_1 is the smallest hash value out of four independently chosen hash values (a_0, a_1, a_2, b_0). Hence, the values of τ are different in the two examples, and therefore $\mathbb{E}[\widehat{I}]$ is also different.

The discrepancy on $\mathbb{E}[\widehat{I}]$ turns out to be crucial since it induces a bias. Specifically, as part of the proof of Theorem 1, we will show that $\mathbb{E}[\widehat{J}] \approx \frac{\mathbb{E}[\widehat{I}]}{\frac{4L}{w+1} - \mathbb{E}[\widehat{I}]}$, and, since the difference between the expected sizes of the minimizer intersections varies for the two examples, we have that $\mathbb{E}[\widehat{J}]$ is also different; in particular, $\mathbb{E}[\widehat{J}]$ is affected by the layout of the k-mer-matching. Note, however, that the Jaccard similarity in both examples is the same, with $J = 0.2$, leading to the intuition that $\widehat{J}$ is biased when $w = 2$. Theorems 1 and 2 show that this bias extends beyond this contrived example and holds for most sequences of interest.

Next, we consider the value of $\mathcal{B}(A,B;w)$ for some more concrete families of sequence pairs. First, consider the case where any pair of k-mers that are shared between A and B are separated by at least w positions. This may approximately happen in practice when A and B are biologically unrelated and the k-mer matches are spurious. Formally, we say two padded sequences A and B are *sparsely-matched* if for all p and q such that $A_p = B_q$, $\{A_{p-w}, \ldots, A_{p-1}, A_{p+1}, \ldots, A_{p+w}\} \notin \mathrm{Sp}^k(B)$, and $\{B_{q-w}, \ldots, B_{q-1}, B_{q+1}, \ldots, B_{q+w}\} \notin \mathrm{Sp}^k(A)$. In such a case, one could imagine that since the shared k-mers do not interfere with each other's windows, the estimator might be unbiased. It turns out this is not the case.

Theorem 3. *Let $w \geqslant 2$, $k \geqslant 2$, and $L \geqslant 7(w+1)$ be integers. Let A and B be two duplicate-free, padded, sparsely-matched sequences, each consisting of L k-mers. Then $\mathcal{B}(A,B;w) \leqslant -J(A,B)\frac{3w^2-3w}{8w^2-2}$.*

A direct consequence of combining this with Theorem 1 is that for sparsely-matched sequences with $J(A,B) > 0$,

$$\frac{\mathbb{E}[\widehat{J}(A,B;w)]}{J(A,B)} \leqslant \frac{5w^2-3w-2}{8w^2-2} + \frac{\varepsilon}{J(A,B)}.$$

For example, for $w = 20$ and sufficiently long sequence pairs with a fixed (i.e. independent of L or w) Jaccard similarity, $\widehat{J}$ is at most 61% of the true Jaccard. The bias cannot be fixed by changing w, since at $w = 2$, $\widehat{J}$ is at most 40% of J, and, as w grows, $\widehat{J}$ is at most 63% of the true Jaccard. This example also shows that $\widehat{J}$ is not only biased but also *inconsistent*, i.e. $\mathbb{E}[\widehat{J}]$ does not converge to J even as the sequences grow long.

Let us now consider the opposite side of the spectrum, where instead of being sparsely-matched, A and B are related by the simple mutation model (i.e. every position is mutated with some constant probability [2]). Deriving the bias for this case proved challenging, since the mutation process adds another layer of randomness. Instead, we derive the bias in a simpler deterministic version of this process, where there is a mutation every g positions, for some $g > w + 2k$.

Theorem 4. *Let $2 \leqslant w < k$, $g > w + 2k$, and $L = \ell g + k$ for some integer $\ell \geqslant 1$. Let A and B be two duplicate-free sequences with L k-mers such that A and B are identical except that the nucleotides at positions $k - 1 + ig$, for $i = 0, \ldots, \ell$, are mutated. Then,*

$$\mathcal{B}(A,B;w) = \frac{2\ell(\ell g + k)h(w)}{(\ell(g+k) + 2k - \ell h(w))(\ell(g+k) + 2k)},$$

where $h(w) = \frac{(w+1)(1-2(H_{2w}-H_w))}{2}$ and $H_n = \sum_{j=1}^{n} \frac{1}{j}$ denotes the n-th Harmonic number.

We can use this theorem in combination with Theorem 1 to obtain a precise approximation of the bias of $\widehat{J}$ for this family of sequences. For instance, taking $k = 15$, $w = 10$, $L = 9992$, and $g = 43$ yields that $\widehat{J}$ is $\approx 10\%$ smaller than the true Jaccard. As g increases, the bias decreases, e.g. for $g = 100$ and $L = 10,016$, $\widehat{J}$ is 4% smaller than the true Jaccard.

4 Overview of Theorem 1 proof

Due to space constraints, we will focus only on the main theorem (Theorem 1) in the main text, providing the intuition and giving an overview of the technical highlights. The proofs of all the theorems, as well as all the building blocks, are deferred to the Appendix. Our main technical novelty is the derivation of a mathematical expression, $\mathcal{C}(A, B; w)$, that approximates the expected value of the size of the minimizer intersection $\widehat{I}(A, B; w)$ between two sequences A and B.

Lemma 1. $\mathcal{C}(A, B; w) \leqslant \mathbb{E}[\widehat{I}(A, B; w)] \leqslant \mathcal{C}(A, B; w) + 2$.

$\mathcal{C}(A, B; w)$ is function of w, L, and of the k-mer-matching between A and B. In particular, when these parameters are known, then $\mathcal{C}(A, B; w)$ can be easily computed. We define $\mathcal{C}(A, B; w)$ formally in Appendix A.1, since it requires the introduction of additional notation. In Section 4.1, we give a high level proof of overview of Lemma 1 that does not require the definition of $\mathcal{C}$.

To prove Theorem 1, we first use Lemma 1 to approximate the value of $\mathbb{E}[\widehat{J}(A, B; w)]$.

Lemma 2. *Let $w \geqslant 2$, $k \geqslant 2$, and $L \geqslant 7(w + 1)$ be integers. Let A and B be two duplicate-free sequences, each consisting of L k-mers. Then there exists $\varepsilon \in [0, \frac{15w^2}{\sqrt[3]{L}}]$ such that*

$$\frac{\mathcal{C}(A, B; w)}{dL - \mathcal{C}(A, B; w)} - \varepsilon \leqslant \mathbb{E}[\widehat{J}(A, B; w)] \leqslant \frac{\mathcal{C}(A, B, w)}{dL - \mathcal{C}(A, B; w)} + \varepsilon,$$

where $d = 4/w + 1$.

Section 4.2 provides a sketch of the proof. Finally, to prove Theorem 1, we show that

$$\mathcal{B}(A, B, w) \approx \frac{\mathcal{C}(A, B, w)}{dL - \mathcal{C}(A, B; w)} - J(A, B),$$

up an additive error that vanishes as the number of k-mers growths; when combined with Lemma 2 this approximation yields Theorem 1 immediately. In the following subsection, we will use $\widehat{I}$ as shorthand for $\widehat{I}(A, B; w)$; we will similarly use $\widehat{U}, \widehat{J}, \mathcal{C}$.

4.1 Lemma 1

In this section, we give an intuition for the proof of Lemma 1 and for where $\mathcal{C}(A, B; w)$ comes from. Let M_p^A be the indicator random variable for the event that A_p is a minimizer in A. The expected size of the minimizer intersection can then be written in terms of M_p^A as follows:

$$\widehat{I}(A, B; w) = \sum_{p=0}^{L-1} \sum_{q=0}^{L-1} M_p^A M_q^B \, \mathbb{1}(A_p = B_q) \tag{1}$$

Here, we use $\mathbb{1}$ in as an indicator function, i.e. $\mathbb{1}(A_p = B_q)$ is 1 if $A_p = B_q$ and 0 otherwise. Next, we use the notion of a charged window from [34, 20]. Given a position $p \in [0, L - 1]$ we say that p *charges* an index i if $i \in [\max\{-1, p - w\}, p - 1]$, $a_p = \min\{a_{i+1}, \ldots, a_{\min(L-w-1, i+w)}\}$ and either $i = \max\{-1, p-w\}$ or $a_i < a_p$. Figure 2 illustrates the definition. For $p \in [0, L - 1]$ and $i \in [-1, L - w - 1]$ we define $X_{i,p}^A$ as an indicator random variable for the event that index i is charged by position p.

The following fact was already shown in [34] and states that a minimizer charges exactly one window; Figure 2 shows the intuition behind it.

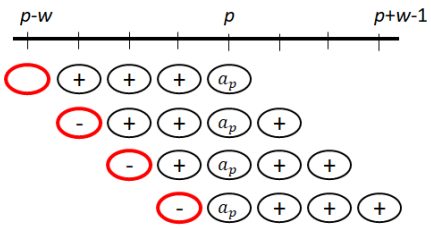


Fig. 2: Illustration of charging. Each row shows a possible way that position p can charge an index, with $w = 4$. A minus sign indicates the value is less than a_p, a plus sign indicates the value is larger than a_p, and no sign indicates that it does not matter. The circle at the index that is charged is shown in bold red. Note that no two rows are compatible with each other, i.e. every row pair contains a column with both a plus and a minus. As a result, the index that gets charged is unique.

Fact 1. *Let $p \in [0, L - 1]$. Position p is a minimizer in A iff there exists a unique $i \in [-1, L - w - 1]$ such that p charges index i. In other words, $M_p^A = \sum_{i=-1}^{L-w-1} X_{i,p}^A$.*

Let us assume for the sake of simplicity and for this section only that A and B are padded. This allows us to combine Eq. 1 with Fact 1 while avoiding edge cases and get:

$$\widehat{I} = \sum_{i=0}^{L-w-1} \sum_{j=0}^{L-w-1} \sum_{p=i+1}^{i+w} \sum_{q=j+1}^{j+w} X_{i,p}^A X_{j,q}^B \, \mathbb{1}(A_p = B_q)$$

Applying linearity of expectation, the law of total probability, and the uniformity of the hash value distribution, we can show that

$$\mathbb{E}[\widehat{I}] = \sum_{i=0}^{L-w-1} \sum_{j=0}^{L-w-1} \sum_{p=i+1}^{i+w} \sum_{q=j+1}^{j+w} \int_0^1 F \, dx, \tag{2}$$

where

$$F = \Pr[X_{i,p}^A = 1, X_{j,q}^B = 1 \mid a_p = b_q = x] \, \mathbb{1}(A_p = B_q).$$

To derive the value of the probability term F, let us fix p and q such that $A_p = B_q$ and fix a_p and b_q to be some value x. Observe that in order for $X_{i,p}^A$ and $X_{j,q}^B$ to both be one, there are certain positions that need to have a hash value less than x (which happens with probability x for each position) and certain positions that need to have a hash value more than x (which happens with probability $1 - x$ for each position). The hash values are pairwise independent, unless the two positions are in the k-mer-matching; in that case, the hash values are forced to be identical. If $X_{i,p}^A X_{j,q}^B = 1$ imply contradictory values for at least one position, then F is zero. Otherwise, let α be the number of hash values that need to be less than x, but counting matched pairs only once. Similarly, let β denote the number hash values that need to be more than x, counting the matched pairs only once. Then,

$$\Pr[X_{i,p}^A X_{j,q}^B = 1 \mid a_p = b_q = x] = x^\alpha (1 - x)^\beta;$$

Figure 3 gives some examples.

Observe that $0 \leqslant \alpha \leqslant 2$ and $0 \leqslant \beta \leqslant 2(L - 1)$. Therefore, the number of distinct terms in the summation of Eq. 2 is at most $6(L-1)$. The number of times each term is included in the summation is the

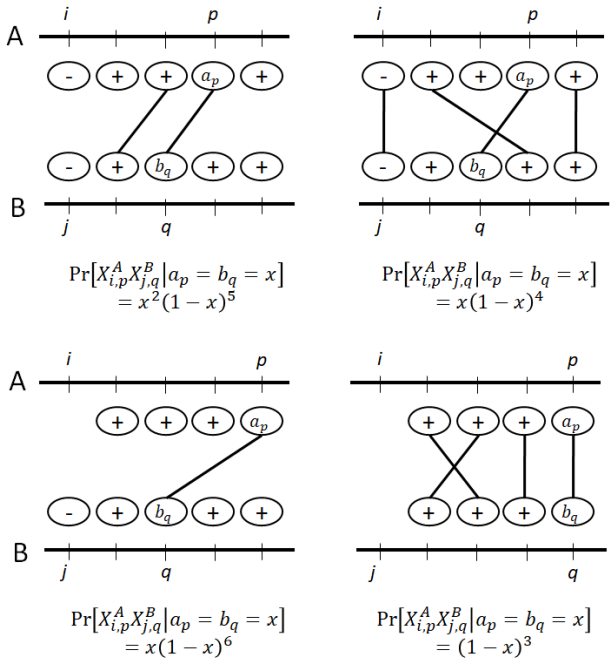


Fig. 3: Some examples of $\Pr[X^A_{i,p}X^B_{j,q} = 1 \mid a_p = b_q = x]$, with $w = 4$. The two horizontal lines correspond to sequences A and B, and a circle corresponds to a k-mer whose value is relevant to the probability. The lines between A and B show the k-mer-matching, i.e. they indicate that the corresponding k-mers are the same. A plus or minus sign at a position reflects that the hash value must be greater or less than x, respectively.

number of i, j, p, q that induce the corresponding values of α and β. In Appendix A.1, we formalize this notion using *configuration counts*; but, for the purposes of intuition, it suffices to observe that Eq. 2 reduces to a function of the k-mer-matching, w, and L. We call this function $\mathcal{C}(A, B; w)$ and then obtain Lemma 1.

4.2 Lemma 2

In this section, we will prove Lemma 2, though we defer the proofs of the building blocks to the Appendix. Lemma 1 gives a tight approximation of $\mathbb{E}[\hat{I}]$ in terms of $\mathcal{C}$. Now, we need to do the same for $\mathbb{E}[\hat{U}]$.

Lemma 3.

$$\frac{4L}{w+1} - \mathcal{C}(A, B; w) - 10 \leqslant \mathbb{E}[\hat{U}(A, B; w)] \leqslant \frac{4L}{w+1} - \mathcal{C}(A, B; w).$$

Now, with Lemmas 1 and 3, we can approximate $\frac{\mathbb{E}[\hat{I}]}{\mathbb{E}[\hat{U}]}$. The next step is to show that this ratio of expectations is a good approximation for the expectation of the ratio $\frac{\hat{I}}{\hat{U}}$, since $\hat{J} = \frac{\hat{I}}{\hat{U}}$. For this, we require asymptotically tight bounds on the variances of the random variables $\hat{I}$ and $\hat{U}$.

Lemma 4.

(i) $Var(\hat{I}(A, B; w)) \leqslant 8w^2\, I(A, B)$;
(ii) $Var(\hat{U}(A, B; w)) \leqslant 32w^2\, L$.

By isolating the central part of the distributions and bounding the effect of the tails using Chebyshev's inequality [22], we then obtain the following approximation for $\mathbb{E}\left[\frac{\hat{I}}{\hat{U}}\right]$.

Lemma 5. $\left|\mathbb{E}\left[\frac{\hat{I}}{\hat{U}}\right] - \frac{\mathbb{E}[\hat{I}]}{\mathbb{E}[\hat{U}]}\right| \leqslant \frac{11w^2}{\sqrt[3]{L}}$.

We now have the components to prove Lemma 2.

Proof (Lemma 2). For the lower bound, we note that

$$\mathbb{E}[\hat{J}] = \mathbb{E}\left[\frac{\hat{I}}{\hat{U}}\right] \geqslant \frac{\mathbb{E}[\hat{I}]}{\mathbb{E}[\hat{U}]} - \frac{11w^2}{\sqrt[3]{L}} \qquad \text{(Lemma 5)}$$

$$\geqslant \frac{\mathcal{C}}{\frac{4L}{w+1} - \mathcal{C}} - \frac{11w^2}{\sqrt[3]{L}} \qquad \text{(Lemmas 1 and 3)}$$

as claimed. For the upper bound, from Lemma 5, we know that

$$\mathbb{E}[\hat{J}] = \mathbb{E}\left[\frac{\hat{I}}{\hat{U}}\right] \leqslant \frac{\mathbb{E}[\hat{I}]}{\mathbb{E}[\hat{U}]} + \frac{11w^2}{\sqrt[3]{L}}.$$

The bounds from Lemmas 1 and 3 imply

$$\mathbb{E}[\hat{J}] \leqslant \frac{\mathcal{C} + 2}{\frac{4L}{w+1} - \mathcal{C} - 10} + \frac{11w^2}{\sqrt[3]{L}}.$$

To complete the proof, we require two additional (and straightforward) bounds.

Fact 2. $\mathcal{C}(A, B; w) \leqslant \frac{2L}{w+1}$.

Fact 3. *For all* $y > 20$ *and* $0 < x \leqslant y/2$, $\frac{x+2}{y-x-10} - \frac{x}{y-x} \leqslant \frac{12}{y-y}$.

Letting $x = \mathcal{C}$ and $y = \frac{4L}{w+1}$, we have $0 < x \leqslant y/2$ and $y > 20$ (since $L \geqslant 7(w+1)$) and so

$$\mathbb{E}[\hat{J}] \leqslant \frac{\mathcal{C}}{\frac{4L}{w+1} - \mathcal{C}} + \frac{12}{\frac{4L}{w+1} - 5} + \frac{11w^2}{\sqrt[3]{L}}$$

$$= \frac{\mathcal{C}}{\frac{4L}{w+1} - \mathcal{C}} + \frac{3(w+1)}{L - \frac{5(w+1)}{4}} + \frac{11w^2}{\sqrt[3]{L}}.$$

Plugging in $w + 1 \leqslant L/7$ and then using the fact that $w \geqslant 2$, we get

$$\mathbb{E}[\hat{J}] - J(A, B) \leqslant \frac{\mathcal{C}}{\frac{4L}{w+1} - \mathcal{C}} + \frac{84(w+1)}{23L} + \frac{11w^2}{\sqrt[3]{L}}$$

$$\leqslant \frac{\mathcal{C}}{\frac{4L}{w+1} - \mathcal{C}} + \frac{15w^2}{\sqrt[3]{L}}. \qquad \square$$

5 Empirical results

5.1 Experimental setup

We use two different models to generate sequence pairs. In the *unrelated pair* model, we take a desired Jaccard value j, set $L = \frac{2j4^k}{j+1}$, and independently and randomly generate two duplicate-free strings A and B with L k-mers. We chose L in this way so that under the assumption that A and B are uniformly chosen, j is the expected value of $J(A, B)$, over the randomness of the generative process. While such string pairs are unlikely to occur in practice for higher values of j, they allow us to observe the bias of unrelated pairs for whole range of Jaccard

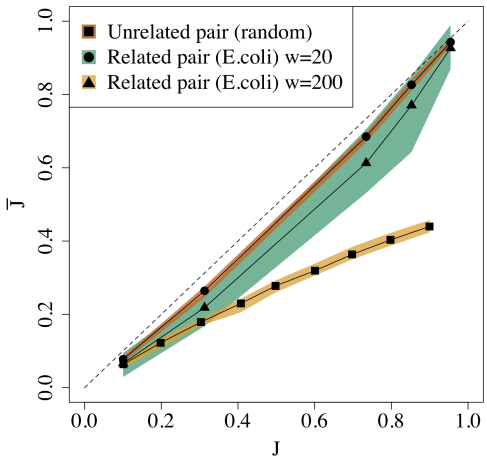


Fig. 4: Empirical bias for unrelated and related sequence pairs. For the unrelated pairs, we used $w = 20$ and $k = 8$ for $J \geqslant .4$ and $k = 7$ for $J \leqslant .3$. For related pairs, we set $k = 16$, $w \in \{20, 200\}$, $L = 10000$, and $r_1 \in \{.001, .005, .01, .05, .1\}$, with one mutation replicate. The colored bands show the 2.5$^{\text{th}}$ and the 97.5$^{\text{th}}$ percentiles. The dashed line shows the expected behavior of an unbiased estimator, with $\bar{J} = J$.

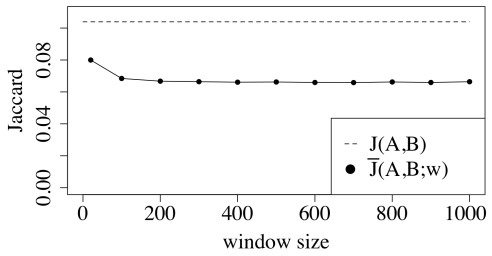


Fig. 5: The effect of w on the empirical bias for a pair of related sequences as a function of the window size. Here, $r_1 = 0.1$, $L = 10,000$, $k = 16$, $w \in \{20, 100, 200, \ldots, 1000\}$, and there are 50 mutation replicates.

similarities. In the *related pair* model, A is a randomly selected substring of *E. coli* [8] with L k-mers. String B is created by sweeping along A, at each position deciding with probability r_1 whether to mutate and then choosing a new nucleotide from those that would not create a duplicate k-mer. More details about the handling of special cases are in Appendix A.6

For each model, we generated 50 *hash replicates* hash function (unless otherwise noted) where each replicate uses a different seed for the hash function. We then report $\bar{J}$, which is the average of $\hat{J}$ over the hash replicates and is the empirical equivalent of $\mathbb{E}[\hat{J}]$. We used the hash function that is part of minimap2 [16], since the idealized hash function we assumed for the convenience of our theoretical proofs is not practical in software. For the mutation model, we also generated some number of *mutation replicates*, where each replicate is the result of re-running the random mutation process. In any experiment, the same set of hash seeds were used for every mutation replicate. Scripts to reproduce our experiments are available on GitHub [26].

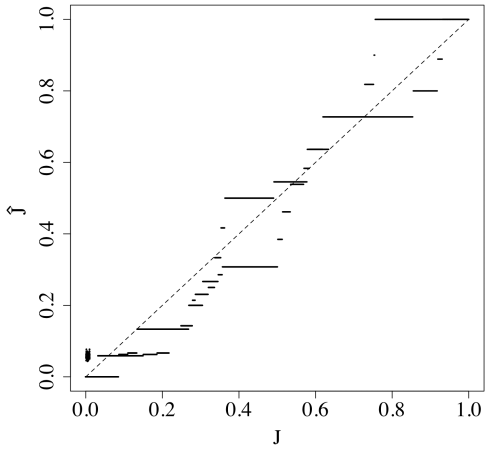


Fig. 6: The empirical bias that occurs during a mapping process. Each point represents a comparison of a read A against a putative mapping location B. Note that the points visually blur into lines. We used $k = 16$ and window size $w = 200$ to match the default of mashmap. One hash replicate was used.

5.2 The extent of the empirical bias on real sequences

Figure 4 shows that there is considerable bias across a wide range of Jaccard values, for both related and unrelated sequence pairs. There are pairs of sequences with a dramatic bias, e.g. for unrelated pair with a Jaccard of 90%, the estimator gives only 44%. In more practically relevant cases, the bias can remain substantial; e.g. when the true Jaccard of related pairs is 76%, the estimator gives only 65% (when $w = 200$). The extent to which this bias is detrimental to the biological interpretation of the result depends on the downstream application. For example, using $\hat{J}$ to estimate the average nucleotide identity in order to build phylogenies, in the style of Mash [25], may be inadvisable.

Figures 4 and 5 show the extent to which the empirical bias depends on the window size w. Figure 4 shows that the bias for related pairs can be twice as large for $w = 200$ compared to $w = 20$. Figure 5 gives a more fine-grained picture and shows how the absolute bias for a related sequence pair increases with w. We note that it plateaus for larger values of w.

We also wanted to understand the extent of the bias in a scenario where the sequences are being compared as part of a read mapping process. To that end, we mimicked the behavior of the mashmap mapper [12, 13] by taking one arbitrary substring A from *E.coli*, with $L = 1,000$, and comparing it to each substring B of *E.coli* with $L = 1,000$. Figure 6 show that during the alignment process, we encounter the whole range of true Jaccard values, and, for each one, there is a substantial but not drastic bias in $\hat{J}$. Unlike the prediction of Theorem 2, the bias is sometimes positive; after further investigation, this happens because the A and B in this experiment are not always padded, which is a condition of Theorem 2.

5.3 Effect of bias on mashmap sequence identity estimates

Mashmap is a read mapper that, for each mapped location, uses the Mash formula [25] to estimate the divergence (i.e. one minus the sequence identity) from $\hat{J}$. It was previously reported that the Mash formula's use of a Poisson approximation makes it inaccurate for

mashmap estimator	true divergence		
	10.00	5.00	1.00
unmodified	11.07	5.88	1.42
corrected	10.48	5.71	1.41
corrected + unbiased	10.05	4.99	1.00

Table 1. The median sequence divergence reported by mashmap, over 100 trials, for unmodified mashmap (first row), mashmap after Binomial-correction (second row) and, in addition, the removal of the $\hat{J}$ bias.

	Related pairs				
r_1	0.001	0.005	0.010	0.050	0.100
J	0.10	0.27	0.74	0.90	0.99
$\mathcal{B}$	-0.02	-0.05	-0.04	-0.02	-0.00
Error of $\mathcal{B}$ (mm2)	0.001	0.000	0.000	0.000	0.001
Error of $\mathcal{B}$ (mmh3)	0.001	0.000	0.001	0.001	0.000
Error of $\mathcal{B}$ (sm64)	0.000	0.000	0.002	0.000	0.000

Table 2. The empirical error of our theoretically predicted bias (Equation (3)) on the related pair sequences of Figure 4. The error is measured with respect to three different hash function families: the minimap2 hash function (mm2), the Murmurhash3 hash function (mmh3), and the SplitMix64 hash function (sm64).

	Unrelated pairs								
J	0.1	0.2	0.3	0.4	0.5	0.6	0.7	0.8	0.9
$\mathcal{B}$	-0.04	-0.08	-0.13	-0.17	-0.22	-0.28	-0.33	-0.39	-0.45
mm2	0.001	0.002	0.001	0.005	0.001	0.004	0.002	0.004	0.007
mmh3	0.001	0.000	0.000	0.002	0.003	0.002	0.001	0.003	0.003
sm64	0.001	0.000	0.001	0.002	0.002	0.003	0.002	0.001	-0.003

Table 3. The empirical error of our theoretically predicted bias (Equation (3)) on the unrelated pair sequences of Figure 4. The error is measured with respect to three different hash function families: the minimap2 hash function (mm2), the Murmurhash3 hash function (mmh3), and the SplitMix64 hash function (sm64).

higher divergence [33, 24], so before proceeding further, we modified mashmap to replace this approximation with the exact Binomial-based derivation (we derive the correction formula in Appendix A.6). We then simulated reads from *E.coli* with substitution errors to achieve a controlled divergence and mapped them back to the *E.coli* reference with mashmap (see Appendix A.6 for more details). We used $k = 16$ and mashmap automatically chose $w = 200$ as the window size.

Table 1 shows that even after our correction, the mashmap divergence had an error, e.g. for a true divergence of 5.00%, mashmap reported an average divergence of 5.71% – an error of 14%. To confirm that this remaining error was due to the minimizer sketch, we replaced the $\hat{J}$ estimator in mashmap with the true Jaccard. Table 1 shows that after this replacement, the remaining error was reduced by an order of magnitude, e.g. mashmap now reported an average divergence of 4.99%. We therefore conclude that the bias we observe in mashmap after the Binomial correction is dominated by the bias of $\hat{J}$. In absolute terms, the $\hat{J}$ bias (about half a percentage point of divergence) may be acceptable for applications such as read alignment. However, for other applications (e.g. a fine grained analysis of sequence divergence), this bias may lead to downstream problems.

5.4 Empirical accuracy of our $\mathcal{B}$ formula (Equation (3))

Theorem 1 predicts that our formula for $\mathcal{B}$ (Equation (3)) approximates the empirical bias. To empirically evaluate the quality of this approximation, we measured the empirical error of Equation (3), which we define to be the absolute difference between the empirically observed bias ($\bar{J} - J$) and $\mathcal{B}$. For the sequence pairs used in Figure 4, the empirical error is never more than 0.007 and roughly one to two orders of magnitude smaller than the bias itself (Tables 2 and 3). This held across three hash function families we tested: the one used by minimap2 [16], Murmurhash3 [23], and SplitMix64 [37]. Note that this robustness to different hash functions is not predicted by Theorem 1, which assumes an idealized version of a hash function which is collision free and maps uniformly to the real unit interval (in this case, none of the three functions map to the unit interval and Murmurhash3 is not collision free).

We measured the effect of increasing w and decreasing L on the empirical error for a related pair (Figure 7). The empirical error increases with w but remains almost two orders of magnitude smaller than the true Jaccard. For $L \geqslant 1000$, the empirical error is less than half a percent of the true Jaccard. Even for the smallest value of L (i.e. 100), the empirical error is only 2.6 percent of the true Jaccard. We conclude that Equation (3) is a high quality approximation for the empirically observed bias.

5.5 Accuracy of the ε bound to the approximation to Equation (3)

Theorem 1 states that the expected error of Equation (3) is at most $\varepsilon = \frac{100w^2}{\sqrt[3]{L}}$. Since this is only an upper bound, we wanted to check the tightness with respect to w and to L. For $w = 20$ and non-astronomical values of L, $\varepsilon > 1$ and thus Theorem 1 gives no guarantee on the accuracy of the $\mathcal{B}$ term. Empirically, however, the error is small (Figure 7A), indicating that, at least for related pairs, ε is likely not a tight bound. To understand if the dependence on L is accurate, we found the best fit of a function of the form αL^{β} to the observed error curve in Figure 7A. The best fit was $0.44L^{-0.74}$, which indicates that our dependence on L in ε is not tight. One possible way to achieve this may be to use tighter concentration bounds than Chebyshev's inequality inside the proof of Lemma 5 (leveraging the limited dependency between the events of k-mers being minimizers). Furthermore, Figure 7B suggests that the true error may be sub-linear in w, while ε has a w^2 dependence. Thus our empirical results indicate that ε could potentially be improved for related sequences, though it may still be tight in the worst-case.

6 Discussion

In this paper, we showed that the minimizer Jaccard estimator suffers from bias and inconsistency, using both theoretical and empirical approaches. The bias can be drastic in some fairly artificial cases (i.e. unrelated sequences with high Jaccard) but remains substantial even on more realistically related pairs of sequences. Our theoretical results indicate that the bias cannot be removed by decreasing the window size (except for the pathological case when $w = 1$, where effectively there is no sketching done). We showed how the bias manifests in the mashmap read mapper as error in the reported sequence divergence. A future direction would be to derive the expected value of the bias $\mathcal{B}$ in the simple mutation model of [2]; if $\mathcal{B}$ reduces to a function of w without depending on the k-mer layout, then it could potentially be used to correct the bias in mashmap. Even if that were not possible, one could still use the estimator provided that an experimental evaluation determines that the observed bias is tolerable for the downstream application. On the other hand, the bias problems can be sidestepped altogether by using a similar but unbiased sketch, e.g. the modulo sketch [34]. Finally, we

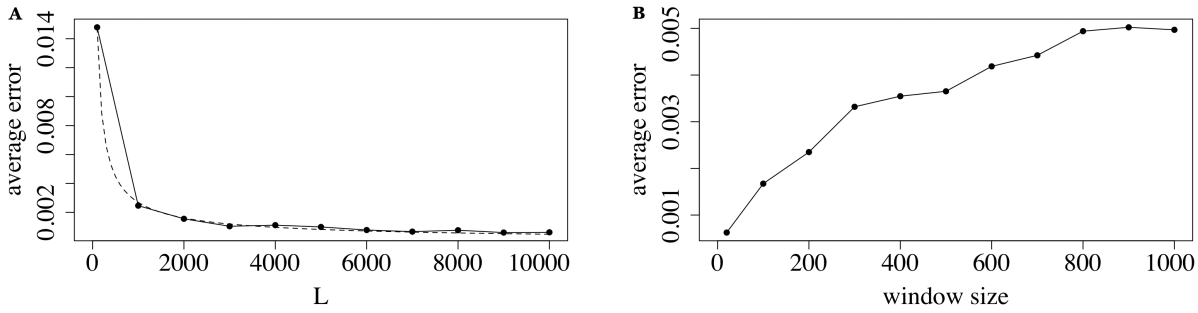


Fig. 7: The effect of the window size w and sequence length on the empirical error of Equation (3). In panel A, we use the related pair model with 50 mutation replicates, $k = 16$, $w = 20$, $r_1 = 0.1$, and $L \in \{100, 1000, 2000, \ldots, 10000\}$. The y-axis shows the error of $\mathcal{B}$, averaged over the mutation replicates. The dashed line shows the best fit function of the form αL^{β}, computed using the nls function in R. The average J, over the mutation replicates, is between .101 and .106, and the average empirical bias ranged between -0.023 and -0.027, depending on L. In panel B, we use the related pair model with 50 mutation replicates, $k = 16$, $L = 10,000$, and $w \in \{20, 100, 200, \ldots, 1000\}$. The average J is .104.

note that while we focus on bias in this paper, it is not the only theoretical property of importance for sketching; for example, there has been much exploration of different hash functions [20, 18, 7, 40, 9, 11, 14, 30] to reduce the density and/or to select k-mers that have desirable properties such as conservation or spread [35].

Our results also relate to the minhash minimizer Jaccard estimator ($\widehat{J}_{\text{minhash}}$) described by [12]. In this variant, the set of k-mers in a minimizer sketch is further reduced by taking the s smallest values (i.e. their minhash sketch); the Jaccard estimator is then computed between these reduced sets. If the minhash sketch is taken using a different hash function than was used for computing minimizers, then the classical result of [3] implies that $\mathbb{E}[\widehat{J}_{\text{minhash}}] = \mathbb{E}[\widehat{J}]$. This estimator would therefore suffer from the same bias that we have shown in this paper. If, on the other hand, the same hash values are reused, then the result of [3] is not applicable, because it assumes that the hash values being selected are uniformly random; in our case, the hash values being selected in the minhash step have already "won the competition" of being smallest in their window. Though we did not explore the bias of this variant of $\widehat{J}_{\text{minhash}}$, it would seem surprising if the minhash step somehow magically unbiased $\widehat{J}$.

References

[1]Daniel N Baker and Ben Langmead. Dashing: fast and accurate genomic distances with hyperloglog. *Genome biology*, 20(1):1–12, 2019.

[2]Antonio Blanca, Robert S Harris, David Koslicki, and Paul Medvedev. The statistics of k-mers from a sequence undergoing a simple mutation process without spurious matches. *bioRxiv*, 2021.

[3]Andrei Z Broder. On the resemblance and containment of documents. In *Proceedings. Compression and Complexity of SEQUENCES 1997 (Cat. No. 97TB100171)*, pages 21–29. IEEE, 1997.

[4]Chen-Shan Chin and Asif Khalak. Human genome assembly in 100 minutes. *bioRxiv*, page 705616, 2019.

[5]Graham Cormode and Shan Muthukrishnan. An improved data stream summary: The count-min sketch and its applications. In *Latin American Symposium on Theoretical Informatics*, pages 29–38. Springer, 2004.

[6]Michael R Crusoe, Hussien F Alameldin, Sherine Awad, Elmar Boucher, Adam Caldwell, Reed Cartwright, Amanda Charbonneau, Bede Constantinides, Greg Edvenson, Scott Fay, et al. The khmer software package: enabling efficient nucleotide sequence analysis. *F1000Research*, 4, 2015.

[7]Dan DeBlasio, Fiyinfoluwa Gbosibo, Carl Kingsford, and Guillaume 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.

[8]*Escherichia coli*, Strain K-12 substrain MG1655, GenBank accession number U00096.3. `https://www.ncbi.nlm.nih.gov/nuccore/U00096`.

[9]Robert Edgar. Syncmers are more sensitive than minimizers for selecting conserved k-mers in biological sequences. *PeerJ*, 9:e10805, 2021.

[10]Philippe Flajolet, Éric Fusy, Olivier Gandouet, and Frédéric Meunier. Hyperloglog: the analysis of a near-optimal cardinality estimation algorithm. In *Discrete Mathematics and Theoretical Computer Science*, pages 137–156, 2007.

[11]Martin C Frith, Laurent Noé, and Gregory Kucherov. Minimally overlapping words for sequence similarity search. *Bioinformatics*, 36(22-23):5344–5350, 2020.

[12]Chirag Jain, Alexander Dilthey, Sergey Koren, Srinivas Aluru, and Adam M Phillippy. A fast approximate algorithm for mapping long reads to large reference databases. In *International Conference on Research in Computational Molecular Biology*, pages 66–81. Springer, 2017.

[13]Chirag Jain, Sergey Koren, Alexander Dilthey, Adam M Phillippy, and Srinivas Aluru. A fast adaptive algorithm for computing whole-genome homology maps. *Bioinformatics*, 34(17):i748–i756, 2018.

[14]Chirag Jain, Arang Rhie, Haowen Zhang, Claudia Chu, Brian P Walenz, Sergey Koren, and Adam M Phillippy. Weighted minimizer sampling improves long read mapping. *Bioinformatics*, 36(Supplement_1):i111–i118, 2020.

[15]Chirag Jain, Luis M Rodriguez-R, Adam M Phillippy, Konstantinos T Konstantinidis, and Srinivas Aluru. High throughput ani analysis of 90k prokaryotic genomes reveals clear species boundaries. *Nature communications*, 9(1):1–8, 2018.

[16]Heng Li. Minimap and miniasm: fast mapping and de novo assembly for noisy long sequences. *Bioinformatics*, 32(14):2103–2110, 2016.

[17]Heng Li. Minimap2: pairwise alignment for nucleotide sequences. *Bioinformatics*, 34(18):3094–3100, 2018.

[18]Guillaume Marçais, Dan DeBlasio, and Carl Kingsford. Asymptotically optimal minimizers schemes. *Bioinformatics*, 34(13):i13–i22, 2018.

[19]Guillaume Marçais, Dan DeBlasio, Prashant Pandey, and Carl Kingsford. Locality-sensitive hashing for the edit distance. *Bioinformatics*, 35(14):i127–i135, 2019.

[20]Guillaume Marçais, David Pellow, Daniel Bork, Yaron Orenstein, Ron Shamir, and Carl Kingsford. Improving the performance of minimizers and winnowing schemes. *Bioinformatics*, 33(14):i110–i117, 2017.

[21]Guillaume Marçais, Brad Solomon, Rob Patro, and Carl Kingsford. Sketching and sublinear data structures in genomics. *Annual Review of Biomedical Data Science*, 2019.

[22]Michael Mitzenmacher and Eli Upfal. *Probability and computing: randomization and probabilistic techniques in algorithms and data analysis*. Cambridge university press, 2017.

[23]MurmurHash3. `https://en.wikipedia.org/wiki/MurmurHash`. Accessed: Oct, 2021.

[24]Brian D Ondov, Gabriel J Starrett, Anna Sappington, Aleksandra Kostic, Sergey Koren, Christopher B Buck, and Adam M Phillippy. Mash Screen: High-throughput sequence containment estimation for genome discovery. *Genome biology*, 20(1):232, 2019.

[25]Brian D Ondov, Todd J Treangen, Páll Melsted, Adam B Mallonee, Nicholas H Bergman, Sergey Koren, and Adam M Phillippy. Mash: fast genome and metagenome distance estimation using minhash. *Genome biology*, 17(1):132, 2016.

[26]Paper github repo. `https://github.com/medvedevgroup/minimizer-jaccard-estimator/tree/main/reproduce`.

[27]N Tessa Pierce, Luiz Irber, Taylor Reiter, Phillip Brooks, and C Titus Brown. Large-scale sequence comparisons with sourmash. *F1000Research*, 8, 2019.

[28]Michael Roberts, Wayne Hayes, Brian R Hunt, Stephen M Mount, and James A Yorke. Reducing storage requirements for biological sequence comparison. *Bioinformatics*, 20(18):3363–3369, 2004.

[29]Will PM Rowe. When the levee breaks: a practical guide to sketching algorithms for processing the flood of genomic data. *Genome biology*, 20(1):199, 2019.

[30]Kristoffer Sahlin. Strobemers: an alternative to k-mers for sequence comparison. *bioRxiv*, 2021.

[31]Kristoffer Sahlin and Paul Medvedev. De novo clustering of long-read transcriptome data using a greedy, quality value-based algorithm. *Journal of Computational Biology*, 27(4):472–484, 2020.

[32]Kristoffer Sahlin and Paul Medvedev. Error correction enables use of oxford nanopore technology for reference-free transcriptome analysis. *Nature Communications*, 12(1):1–13, 2021.

[33]Shahab Sarmashghi, Kristine Bohmann, M Thomas P Gilbert, Vineet Bafna, and Siavash Mirarab. Skmer: assembly-free and alignment-free sample identification using genome skims. *Genome biology*, 20(1):1–20, 2019.

[34]Saul Schleimer, Daniel S Wilkerson, and Alex Aiken. Winnowing: local algorithms for document fingerprinting. In *Proceedings of the 2003 ACM SIGMOD international conference on Management of data*, pages 76–85. ACM, 2003.

[35]Jim Shaw and Yun William Yu. Theory of local k-mer selection with applications to long-read alignment. *bioRxiv*, 2021.

[36]Anshumali Shrivastava. Optimal densification for fast and accurate minwise hashing. In *International Conference on Machine Learning*, pages 3154–3163. PMLR, 2017.

[37]Guy L Steele Jr, Doug Lea, and Christine H Flood. Fast splittable pseudorandom number generators. *ACM SIGPLAN Notices*, 49(10):453–472, 2014.

[38]Larry Wasserman. *All of statistics: a concise course in statistical inference*. Springer Science & Business Media, 2013.

[39]XiaoFei Zhao. BinDash, software for fast genome distance estimation on a typical personal laptop. *Bioinformatics*, 35(4):671–673, 2019.

[40]Hongyu Zheng, Carl Kingsford, and Guillaume Marçais. Improved design and analysis of practical minimizers. *Bioinformatics*, 36(Supplement_1):i119–i127, 2020.

A Appendix

In this appendix, we will prove the main theorems of the paper as well as provide experimental details to aid reproducibility.

A.1 Matching configurations and the definition of $\mathcal{C}(A, B; w)$ and $\mathcal{B}(A, B; w)$

In this section, we define the notion of matching configurations and then use them to define $\mathcal{C}(A, B; w)$ and $\mathcal{B}(A, B; w)$. As discussed in Section 3, the bias of $\hat{J}$ depends on the layout of the shared k-mers along the sequence. It turns out that the aspects of their sharedness that contribute to the bias are captured by the amount and location of k-mers that are shared between windows $\{A_i, \ldots, A_{i+w}\}$ and $\{B_j, \ldots, B_{j+w}\}$, for any i and j.

Let us define $S(i, j, \ell) \triangleq |\{A_i, \ldots, A_{i+\ell-1}\} \cap \{B_j, \ldots, B_{j+\ell-1}\}|$, i.e. the number of shared k-mers in the windows of length ℓ starting at positions i and j in A and B, respectively. We then define a *matching configuration* as a 5-tuple, written as

$$[\![C_{\text{a,left}}, C_{\text{a,right}}; C_{\text{b,left}}, C_{\text{b,right}}; s]\!],$$

where $s \in \{0, \ldots, w\}$ and $C_{\text{a,left}}, C_{\text{a,right}}, C_{\text{b,left}}, C_{\text{b,right}} \in \{0, 1, 2\}$. We then say that an index pair (i, j) with $i, j \in [0, L-w-1]$ has configuration $[\![C_{\text{a,left}}, C_{\text{a,right}}; C_{\text{b,left}}, C_{\text{b,right}}; s]\!]$ if the windows $\{A_{i+1}, \ldots, A_{i+w}\}$ and $\{B_{j+1}, \ldots, B_{j+w}\}$ share s k-mers (i.e., $s = S(i+1, j+1, w)$) and

$$C_{\text{a,left}} = \begin{cases} 0 \text{ if } A_i = B_j, \\ 1 \text{ if } A_i \in \{B_{j+1}, \ldots, B_{j+w}\}, \\ 2 \text{ otherwise;} \end{cases} \qquad C_{\text{a,right}} = \begin{cases} 0 \text{ if } A_{i+w} = B_{j+w}, \\ 1 \text{ if } A_{i+w} \in \{B_{j+1}, \ldots, B_{j+w-1}\}, \\ 2 \text{ otherwise;} \end{cases}$$

$$C_{\text{b,left}} = \begin{cases} 0 \text{ if } B_j = A_i, \\ 1 \text{ if } B_j \in \{A_{i+1}, \ldots, A_{i+w}\}, \\ 2 \text{ otherwise;} \end{cases} \qquad C_{\text{b,right}} = \begin{cases} 0 \text{ if } B_{j+w} = A_{i+w}, \\ 1 \text{ if } B_{j+w} \in \{A_{i+1}, \ldots, A_{i+w-1}\}, \\ 2 \text{ otherwise.} \end{cases}$$

An index pair (i, j) has exactly one configuration, and not all configurations are possible; in particular, configurations where exactly one of $C_{\text{a,left}}$ or $C_{\text{b,left}}$ is zero, or exactly one of $C_{\text{b,right}}$ and $C_{\text{a,right}}$ is zero, are impossible. Figure S1 shows some examples of configurations. We may label configuration elements as sets (e.g. $C_{\text{a,left}} = \{0, 2\}$) to indicate all the configurations that can be formed using values from that set, except for impossible configurations. We use $*$ as shorthand for the set $\{0, 1, 2\}$ of all possible values. For example, $[\![*, 0; *, 0; s]\!]$ refers to the configurations $[\![0, 0; 0, 0; s]\!]$, $[\![1, 0; 1, 0; s]\!]$, $[\![2, 0; 1, 0; s]\!]$, $[\![1, 0; 2, 0; s]\!]$, $[\![2, 0; 2, 0; s]\!]$. For a configuration C we use $N(C)$ to denote the number of pairs (i, j) such that the configuration of (i, j) is C.

In order to define $\mathcal{B}(A, B; w)$, we define first the quantity $\mathcal{C}(A, B; w)$. Let $t_0 = \frac{1}{2w-s}$, $t_1 = \frac{1}{(2w-s)(2w-s+1)}$, and $t_2 = \frac{1}{(2w-s)(2w-s+1)(2w-s+2)}$.

$$\begin{aligned} \mathcal{C}(A, B; w) \triangleq \sum_{s=0}^{w} & t_0 N([\![1, 0; 1, 0; s]\!]) + t_0 N([\![1, 0; 2, 0; s]\!]) + t_0 N([\![2, 0; 1, 0; s]\!])) \\ & + t_1 N([\![2, \{1, 2\}; 1, 1; s]\!]) + t_1 N([\![1, 1; 2, \{1, 2\}; s]\!]) + 2wt_1 N([\![0, 0; 0, 0; s]\!]) \\ & + t_1 s N([\![0, 1; 0, 1; s]\!]) + t_1 s N([\![0, 1; 0, 2; s]\!]) + t_1 s N([\![0, 2; 0, 1; s]\!]) \\ & + t_1 s N([\![0, 2; 0, 2; s]\!]) + 2t_2 s N([\![2, 2; 2, 2; s]\!]) + 4t_2 w N([\![2, 1; 2, 1; s]\!]) \\ & + t_2(s + 2w) N([\![2, 1; 2, 2; s]\!]) + t_2(s + 2w) N([\![2, 2; 2, 1; s]\!]) \\ & + t_2(6w - s + (2w - s)^2) N([\![2, 0; 2, 0; s]\!]) \end{aligned}$$

In particular, $\mathcal{C}(A, B; w)$ is a linear combination of configuration counts, where each count is weighted by some function of its s value and w. We also define $\mathcal{D}(A, B; w) = \sum_{s=0}^{w} N([\![*, 0; *, 0; s]\!])$. The term $\mathcal{B}(A, B; w)$, which essentially determines the bias of the Jaccard estimator (see Theorem 1), is defined as follows:

$$\mathcal{B}(A, B; w) \triangleq \frac{\mathcal{C}(A, B; w)}{\frac{4L}{w+1} - \mathcal{C}(A, B; w)} - \frac{\mathcal{D}(A, B; w)}{2L - \mathcal{D}(A, B; w)}. \tag{3}$$

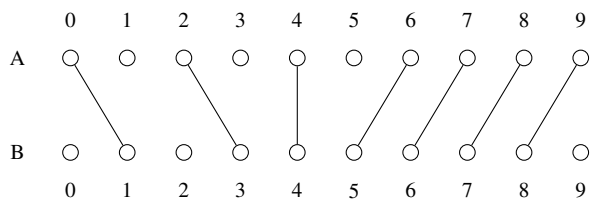


Fig. S1: Configuration examples with $w = 2$: the pair $(0, 1)$ has configuration $[\![0, 0; 0, 0; 1]\!]$; pair $(4, 4)$ has $[\![0, 1; 0, 2; 1]\!]$; pair $(7, 6)$ has $[\![0, 0; 0, 0; 2]\!]$.

A.2 Proof of Theorem 1

In all the following, we will assume that $L \geqslant 7(w+1)$.

A.2.1 Approximating the minimizer union and intersection (Lemmas 1 and 3)

In this section, we will prove Lemmas 1 and 3. First, we recapitulate the proof of Fact 1 in our notation:

Fact 1. *Let $p \in [0, L-1]$. Position p is a minimizer in A iff there exists a unique $i \in [-1, L-w-1]$ such that p charges index i. In other words, $M_p^A = \sum_{i=-1}^{L-w-1} X_{i,p}^A$.*

Proof. Figure 2 gives the intuition for the proof. For the only if direction, suppose that p charges index i. Then, by definition of charging, $a_p = \min\{a_{i+1}, \ldots, a_{i+w}\}$, and so p is a minimizer. For the if direction, suppose that p is a minimizer in A. Consider the leftmost window in which it is a minimizer, i.e. the smallest $i' \in [p-w+1, p]$ such that $a_p = \min\{a_{i'}, \ldots, a_{i'+w-1}\}$. Since i' is smallest, then either $i' = p-w+1$ or $a_{i'-1} < a_p$. This is the definition of p charging index $i'-1$. For uniqueness, consider all the possible windows that p can charge, shown in Figure 2. They are all pairwise incompatible, i.e. there is at least one position that is simultaneously required to be larger than a_p and smaller than a_p. □

The expected value of M_p^A is called the *density* of the minimizer scheme, and we compute it exactly in the following Fact. We note that similar derivations of the density also appeared in [34, 28], but our proof accounts also for the edge cases.

Fact 4. *For $p \in [0, L-1]$, we have $\mathbb{E}[M_p^A] \leqslant \frac{2}{w+1}$. More precisely,*

$$\mathbb{E}[M_p^A] = \begin{cases} \frac{2}{w+1} & \text{for } p \in [w, L-w]; \\ \frac{w+1+p}{w(w+1)} & \text{for } p \in [0, w-1]; \\ \frac{L-p+w}{w(w+1)} & \text{for } p \in [L-w+1, L-1]. \end{cases}$$

Proof. Let $\ell = \max(-1, p-w)$ and $u = \min(L-w-1, p-1)$. For $i \in [\ell+1, u]$, we have $\Pr[X_{i,p}^A] = \int_0^1 \Pr[X_{i,p}^A \mid a_p = x]\, dx = \int_0^1 x(1-x)^{w-1}\, dx = \frac{1}{w(w+1)}$. For $i = \ell$, we have $\Pr[X_{i,p}^A] = \int_0^1 (1-x)^{w-1}\, dx = 1/w$.

By Fact 1, $M_p^A = \sum_{i=-1}^{L-w-1} X_{i,p}^A$. When $p \in [0, w-1]$, we have

$$M_p^A = X_{-1,p}^A + \sum_{i=0}^{p-1} X_{i,p}^A = \frac{1}{w} + \frac{p}{w(w+1)}.$$

When $p \in [w, L-w]$, we have

$$M_p^A = X_{p-w,p}^A + \sum_{i=p-w+1}^{p-1} X_{i,p}^A = \frac{1}{w} + \frac{w-1}{w(w+1)} = \frac{2}{w+1}.$$

When $p \in [L-w+1, L-1]$, we have

$$M_p^A = X_{p-w,p}^A + \sum_{i=p-w+1}^{L-w-1} X_{i,p}^A = \frac{1}{w} + \frac{L-p-1}{w(w+1)} = \frac{L-p+w}{w(w+1)}.$$

□

We are now ready to prove Lemma 1.

Lemma 1. $\mathcal{C}(A, B; w) \leqslant \mathbb{E}[\widehat{I}(A, B; w)] \leqslant \mathcal{C}(A, B; w) + 2$.

Proof. From the definition of $\widehat{I}(A, B; w)$ and Fact 1, we have

$$\widehat{I}(A, B; w) = \sum_{i=-1}^{L-w-1} \sum_{p=0}^{L-1} \sum_{j=-1}^{L-w-1} \sum_{q=0}^{L-1} X_{i,p}^A X_{j,q}^B \mathbb{1}(A_p = B_q).$$

Observe that by definition of charging, $X_{i,p}^A = 0$ when $p \notin [i+1, i+w]$. Therefore,

$$\widehat{I}(A, B; w) = \sum_{i=-1}^{L-w-1} \sum_{p=i+1}^{i+w} \sum_{j=-1}^{L-w-1} \sum_{q=j+1}^{j+w} X_{i,p}^A X_{j,q}^B \mathbb{1}(A_p = B_q).$$

We can ignore some of the boundary terms associated with position -1 being charged without much loss in accuracy. Let

$$\widehat{I}_{\text{core}} = \sum_{i=0}^{L-w-1} \sum_{p=i+1}^{i+w} \sum_{j=0}^{L-w-1} \sum_{q=j+1}^{i+w} X_{i,p}^A X_{j,q}^B \mathbb{1}(A_p = B_q).$$

We claim that $\mathbb{E}[\widehat{I}_{\text{core}}] \leqslant \mathbb{E}[\widehat{I}(A,B;w)] \leqslant \mathbb{E}[\widehat{I}_{\text{core}}] + 2$. The lower bound is immediate. For the upper bound, let us first separate out the terms of $\widehat{I}$ with $i = -1$ or $j = -1$:

$$\widehat{I}(A,B;w) \leqslant \widehat{I}_{\text{core}} + \sum_{i=-1}^{L-w-1} \sum_{p=i+1}^{i+w} \sum_{q=0}^{w-1} X^A_{i,p} X^B_{-1,q} \mathbb{1}(A_p = B_q) + \sum_{p=0}^{w-1} \sum_{j=-1}^{L-w-1} \sum_{q=j+1}^{j+w} X^A_{-1,p} X^B_{j,q} \mathbb{1}(A_p = B_q)$$

For the second term, observe that, by definition of charging, there is at most one value of q for which $X^B_{-1,q} = 1$. Then, since there are no repeated k-mers in A or B, there is at most one value of p for which $A_p = B_q$. Finally, by definition of charging, there is at most one value of i for which $X^A_{i,p} = 1$. Hence the second term is at most one; by a symmetrical argument, the third term is at most one as well. This gives us the desired upper bound.

It now suffices to show that $\mathbb{E}[\widehat{I}_{\text{core}}] = \mathcal{C}(A,B;w)$.

$$\begin{aligned}
\mathbb{E}[\widehat{I}_{\text{core}}] &= \sum_{i=0}^{L-w-1} \sum_{j=0}^{L-w-1} \sum_{p=i+1}^{i+w} \sum_{q=j+1}^{j+w} \mathbb{E}[X^A_{i,p} X^B_{j,q}] \mathbb{1}(A_p = B_q) \\
&= \sum_{i=0}^{L-w-1} \sum_{j=0}^{L-w-1} \sum_{p=i+1}^{i+w} \sum_{q=j+1}^{j+w} \Pr[X^A_{i,p} = 1, X^B_{j,q} = 1] \mathbb{1}(A_p = B_q) \\
&= \sum_{i=0}^{L-w-1} \sum_{j=0}^{L-w-1} \sum_{p=i+1}^{i+w} \sum_{q=j+1}^{j+w} \int_0^1 \Pr[X^A_{i,p} = 1, X^B_{j,q} = 1 \mid a_p = b_q = x] \mathbb{1}(A_p = B_q)\, dx.
\end{aligned}$$

The probability $\Pr[X^B_{j,q} = 1, X^A_{i,p} = 1 \mid a_p = b_q = x]$ will depend on the configuration of the indices i and j and on whether $p = i + w$ or $q = j + w$. Therefore, we rearrange the sums as follows. For a configuration c, we say that $(i,j) \to c$ when the indices i and j are in configuration c, so that

$$\begin{aligned}
\mathbb{E}[\widehat{I}_{\text{core}}] &= \sum_c \sum_{(i,j)\to c} \sum_{p=i+1}^{i+w} \sum_{q=j+1}^{j+w} \int_0^1 \Pr[X^A_{i,p} = 1, X^B_{j,q} = 1 \mid a_p = b_q = x] \mathbb{1}(A_p = B_q)\, dx \\
&= \sum_c \sum_{(i,j)\to c} \sum_{p=i+1}^{i+w-1} \sum_{q=j+1}^{j+w-1} \int_0^1 \Pr[X^A_{i,p} = 1, X^B_{j,q} = 1 \mid a_p = b_q = x] \mathbb{1}(A_p = B_q)\, dx \qquad (4) \\
&\quad + \sum_c \sum_{(i,j)\to c} \sum_{p=i+1}^{i+w} \int_0^1 \Pr[X^A_{i,p} = 1, X^B_{j,j+w} = 1 \mid a_p = b_{j+w} = x] \mathbb{1}(A_p = B_q)\, dx \qquad (5) \\
&\quad + \sum_c \sum_{(i,j)\to c} \sum_{q=j+1}^{j+w-1} \int_0^1 \Pr[X^A_{i,i+w} = 1, X^B_{j,q} = 1 \mid a_{i+w} = b_q = x] \mathbb{1}(A_p = B_q)\, dx. \qquad (6)
\end{aligned}$$

Figure 3 gives some examples to develop the intuition for what the inner term can evaluate to. We consider next each summation Equation (4), Equation (5), and Equation (6) separately. We start with Equation (5). Note that in this case the value of q is fixed to $j + w$, and so there is at most one value of p in the summation that is not 0 (since $A_p = B_q$). We partition the space of all configurations into four possible cases: (i) $c = [\![*, 0; *, 0; s]\!]$, (ii) $[\![\{0,2\}, *; *, 1; s]\!]$, (iii) $c = [\![1, *; *, 1; s]\!]$, and (iv) $c = [\![*, *; *, 2; s]\!]$.

First note that for any c, we have $X^B_{j,j+w} = 1$ if and only if $b_{j+1}, \ldots, b_{j+w-1}$ are each greater than x. In case (i) when $c = [\![*, 0; *, 0; s]\!]$, the only value of p for which the probability in Equation (5) is not zero is $p = i + w$. From the definition of charging, we have $X^A_{i,i+w} = 1$ and $X^B_{j,j+w} = 1$ if and only if $a_{i+1}, \ldots, a_{i+w-1}, b_{j+1}, \ldots, b_{j+w-1}$ are each greater than x. The number of distinct k-mers in this sequence is $2w - 2 - S(i+1, j+1, w-1) = 2w - 2 - S(i+1, j+1, w) + 1 = 2w - 1 - s$. Therefore, $\Pr[X^A_{i,p} = 1, X^B_{j,j+w} = 1 \mid a_p = b_q = x] = (1-x)^{2w-1-s}$ and

$$\sum_{p=i+1}^{i+w} \int_0^1 \Pr[X^A_{i,p} = 1, X^B_{j,j+w} = 1 \mid a_p = b_{j+w} = x] \mathbb{1}(A_p = B_{w+j})\, dx = \int_0^1 (1-x)^{2w-1-s}\, dx = t_0,$$

recalling that $t_0 = \frac{1}{2w-s}$, $t_1 = \frac{1}{(2w-s)(2w-s+1)}$, and $t_2 = \frac{1}{(2w-s)(2w-s+1)(2w-s+2)}$. For case (ii) with $c = [\![\{0,2\}, *; *, 1; s]\!]$, because $C_{\text{b,right}} = 1$, the only value of p for which the probability in Equation (5) is not zero belongs to $[i+1, i+w-1]$. From the definition of charging, we have $X^A_{i,p} = 1$ iff $a_i < x$ and $a_{i+1}, \ldots, a_{i+w}$, with the exception of a_p, are all greater than x. As mentioned previously, we have that $X^B_{j,j+w} = 1$ iff $b_{j+1}, \ldots, b_{j+w-1}$ are each greater than x. Because $C_{\text{a,left}} \neq 1$, we have $A_i \notin \{B_{j+1}, \ldots, B_{j+w-1}\}$. Therefore, we have one hash value (i.e. a_i) that is less than x, and $2w - 2 - (S(i+1, j+1, w) - 1)$ distinct hash values that are more than x. As a result,

$$\sum_{p=i+1}^{i+w} \int_0^1 \Pr[X^A_{i,p} = 1, X^B_{j,j+w} = 1 \mid a_p = b_{j+w} = x] \mathbb{1}(A_p = B_{j+w})\, dx = \int_0^1 x(1-x)^{2w-1-s}\, dx \qquad = t_1.$$

For next two cases (i.e., case (iii) and (iv)) we show that the sum is 0. When $c = [\![1, *; *, 1; s]\!]$, the fact that $C_{\text{b,right}} = 1$ means that $C_{\text{a,right}} \neq 0$ which implies that $p < i + w$ and that, if $X^A_{i,p} = 1$, then $a_i < x$. The fact that $C_{\text{a,left}} = 1$ implies that $A_i \in \{B_{j+1}, \ldots, B_{j+w}\}$. Therefore, one of the values of $\{b_{j+1}, \ldots, b_{j+w}\}$ is less than x, which makes it impossible that $X^B_{j,q} = 1$. When $c = [\![*, *; *, 2; s]\!]$, there is no value of $p \in [i + 1, i + w]$ which satisfies $A_p = B_{j+w}$, so $\mathbb{1}(A_p = B_{j+w}) = 0$. Putting all the four cases together, we have shown that the inner summation in Equation (5) is:

$$\sum_c \sum_{(i,j)\to c} \sum_{p=i+1}^{i+w} \int_0^1 \Pr[X^A_{i,p} = 1, X^B_{j,j+w} = 1 \mid a_p = b_{j+w} = x]\, \mathbb{1}(A_p = B_q)\, dx$$
$$= \sum_{s=0}^{w} t_0 N([\![*, 0; *, 0; s]\!]) + t_1 N([\![\{0, 2\}, *; *, 1; s]\!]). \tag{7}$$

Deriving a closed form for Equation (6) is symmetric to Equation (5) with the exception that when $c = [\![*, 0; *, 0; s]\!]$, there is no value of q in the range of the sum (i.e. $q \in [j + 1, j + w - 1]$) such that $A_{i+w} = B_q$. Hence, for the inner summation in Equation (6), we obtain

$$\sum_c \sum_{(i,j)\to c} \sum_{q=j+1}^{j+w-1} \int_0^1 \Pr[X^A_{i,i+w} = 1, X^B_{j,q} = 1 \mid a_{i+w} = b_q = x]\, \mathbb{1}(A_p = B_q)\, dx$$
$$= \sum_{s=0}^{w} t_1 N([\![*, 1; \{0, 2\}, *; s]\!]) \tag{8}$$

With a similar but more delicate case-by-case analysis, we also derive a closed form for Equation (4), whose proof we postpone until later.

Fact 5. *Let*

$$T = \sum_c \sum_{(i,j)\to c} \sum_{p=i+1}^{i+w-1} \sum_{q=j+1}^{j+w-1} \int_0^1 \Pr[X^A_{i,p} = 1, X^B_{j,q} = 1 \mid a_p = b_q = x]\, \mathbb{1}(A_p = B_q)\, dx.$$

Then,

$$T = \sum_{s=0}^{w} s t_1 N([\![0, 2; 0, 2; s]\!]) + 2s t_2 N([\![2, 2; 2, 2; s]\!]) + 2(s-2) t_2 N([\![2, 1; 2, 1; s]\!])$$
$$+ (s-2) t_1 N([\![0, 1; 0, 1; s]\!]) + (s-1) t_1 (N([\![0, 1; 0, 2; s]\!]) + N([\![0, 2; 0, 1; s]\!]) + N([\![0, 0; 0, 0; s]\!]))$$
$$+ 2(s-1) t_2 (N([\![2, 1; 2, 2; s]\!]) + N([\![2, 2; 2, 1; s]\!]) + N([\![2, 0; 2, 0; s]\!])). \tag{9}$$

Finally, observe that summing Equation (7), Equation (8) and Equation (9) and then collecting the coefficients for each configuration, we obtain that $G = \mathcal{C}(A, B; w)$ as desired. □

We proceed with the proof of Fact 5.

Proof of Fact 5. For ease of notation, for a configuration c and a pair $(i, j) \to c$, let

$$H(c, i, j) = \sum_{p=i+1}^{i+w-1} \sum_{q=j+1}^{j+w-1} \int_0^1 \Pr[X^A_{i,p} = 1, X^B_{j,q} = 1 \mid a_p = b_q = x]\, \mathbb{1}(A_p = B_q)\, dx.$$

Since $p \neq i + w$ and $q \neq j + w$, we have that $X^A_{i,p} = 1$ and $X^B_{j,q} = 1$ iff $a_i < x$, $b_j < x$, and $a_{i+1}, \ldots, a_{i+w}$, $b_{j+1}, \ldots, b_{j+w}$, with the exception of a_p and b_q, are each greater than x. This corresponds to $2w - 1 - s$ hash values needing to be greater than x. What remains is to compute how many hash values need to be less than x.

We will partition the space of configurations into four possible cases: $[\![0, *; 0, *; s]\!]$, $[\![2, *; 2, *; s]\!]$, $[\![*, *; 1, *; s]\!]$, and $[\![1, *; *, *; s]\!]$. First, consider the case of $c = [\![0, *; 0, *; s]\!]$. In this case, $A_i = B_j$. Therefore,

$$H([\![0, *; 0, *; s]\!], i, j) = \sum_{p=i+1}^{i+w-1} \sum_{q=j+1}^{j+w-1} \int_0^1 x(1-x)^{2w-1-s}\, dx = \sum_{p=i+1}^{i+w-1} \sum_{q=j+1}^{j+w-1} t_1 = t_1 S(i+1, j+1, w-1).$$

Next, consider the case of $[\![2, *; 2, *; s]\!]$. This case is exactly the same as $c = [\![0, *; 0, *; s]\!]$, except that $A_i \neq B_j$ and so

$$H([\![2, *; 2, *; s]\!], i, j) = \sum_{p=i+1}^{i+w-1} \sum_{q=j+1}^{j+w-1} \int_0^1 x^2(1-x)^{2w-1-s}\, dx = 2t_2 S(i+1, j+1, w-1)$$

Next, observe that

$$S(i+1,j+1,w-1) = s - \begin{cases} 0 \text{ if } C_{\text{a,right}} = 2 \text{ and } C_{\text{b,right}} = 2 \\ 1 \text{ if } C_{\text{a,right}} = 0 \text{ and } C_{\text{b,right}} = 0 \\ 1 \text{ if } C_{\text{a,right}} = 1 \text{ and } C_{\text{b,right}} = 2 \\ 1 \text{ if } C_{\text{a,right}} = 2 \text{ and } C_{\text{b,right}} = 1 \\ 2 \text{ if } C_{\text{a,right}} = 1 \text{ and } C_{\text{b,right}} = 1 \,, \end{cases} \tag{10}$$

where recall that $s = S(i+1, j+1, w)$. Therefore,

$$\begin{aligned} H([\![0,2;0,2;s]\!],i,j) &= st_1, \\ H([\![0,1;0,2;s]\!],i,j) &= H([\![0,2;0,1;s]\!],i,j) = H([\![0,0;0,0;s]\!],i,j) = (s-1)t_1, \\ H([\![0,1;0,1;s]\!],i,j) &= (s-2)t_1, \\ H([\![2,2;2,2;s]\!],i,j) &= 2st_2, \\ H([\![2,1;2,2;s]\!],i,j) &= H([\![2,2;2,1;s]\!],i,j) = H([\![2,0;2,0;s]\!],i,j) = 2(s-1)t_2, \\ H([\![2,1;2,1;s]\!],i,j) &= 2(s-2)t_2. \end{aligned}$$

Now, when $c = [\![1,*;*,*;s]\!]$, $A_i \in \{B_{j+1}, \ldots, B_{j+w}\}$. However, we already argued that $a_i < x$ and that $b_{j+1}, \ldots, b_{j+w}$ are all at least x. Hence, we cannot have both $X^A_{i,p} = 1$ and $X^B_{j,q} = 1$, and this type of configuration does not contribute to the sum. The case of $c = [\![*,*;1,*;s]\!]$ is symmetric. Finally, observing that $T = \sum_c \sum_{(i,j)\to c} H(c,i,j)$, we combine all the cases to get the desired equality of the fact statement. □

We now restate Lemma 3, whose proof is a direct consequence of Lemma 1.

Lemma 3.
$$\frac{4L}{w+1} - \mathcal{C}(A,B;w) - 10 \leqslant \mathbb{E}[\widehat{U}(A,B;w)] \leqslant \frac{4L}{w+1} - \mathcal{C}(A,B;w).$$

Proof. Recall that M^A_p denotes the indicator random variable for A_p being a minimizer in A. Then

$$\mathbb{E}[\widehat{U}(A,B;w)] = \sum_{p=0}^{L-1} \mathbb{E}[M^A_p] + \sum_{q=0}^{L-1} \mathbb{E}[M^B_q] - \mathbb{E}[I(A,B;w)] = 2\sum_{p=0}^{L-1} \mathbb{E}[M^A_p] - \mathbb{E}[\widehat{I}(A,B;w)].$$

From Lemma 1, we know that $\mathbb{E}[\widehat{I}(A,B;w)] \geqslant \mathcal{C}(A,B;w)$, and from Fact 4 we get that $\sum_{p=0}^{L-1} \mathbb{E}[M^A_p] \leqslant \frac{2L}{w+1}$. Combining these two facts, we deduce

$$\mathbb{E}[\widehat{U}(A,B;w)] \leqslant \frac{4L}{w+1} - \mathcal{C}(A,B;w),$$

as desired. For the lower bound, from Fact 4 we can deduce that

$$\sum_{p=0}^{L-1} \mathbb{E}[M^A_p] \geqslant \sum_{p=w}^{L-w} \mathbb{E}[M^A_p] = \frac{2(L-2w+1)}{w+1} \geqslant \frac{2L}{w+1} - \frac{4w-2}{w+1} \geqslant \frac{2L}{w+1} - 4.$$

The lower bound then follows from Lemma 1. □

A.2.2 Approximating the ratio of the minimizer union and intersection (Lemmas 4 and 5)

We begin this section with the proof of Lemma 4, where we obtain bounds for the variances of $\widehat{I}(A,B;w)$ and $\widehat{U}(A,B;w)$.

Lemma 4.

(i) $Var(\widehat{I}(A,B;w)) \leqslant 8w^2\, I(A,B)$;
(ii) $Var(\widehat{U}(A,B;w)) \leqslant 32w^2\, L$.

Proof. For ease of notation, we let $I = I(A,B)$ and $U = U(A,B)$. If p is a position in A, then define $w_p = \{A_{\max\{0,p-w+1\}}, \ldots, A_{\min\{p+w-1,L-1\}}\}$ and, if $x = A_p$, we say that the k-mers in w_p are *nearby* x in A.

We begin with part (i). For ease of notation set $\widehat{I} = \widehat{I}(A,B;w)$ and recall that

$$\widehat{I} = \sum_{p=0}^{L-1}\sum_{q=0}^{L-1} M^A_p M^B_q\, \mathbb{1}(A_p = B_q).$$

Then,

$$\mathbb{E}[\widehat{I}^2] = \mathbb{E}\left[\left(\sum_{p=0}^{L-1}\sum_{q=0}^{L-1} M^A_p M^B_q\, \mathbb{1}(A_p = B_q)\right)\left(\sum_{p'=0}^{L-1}\sum_{q'=0}^{L-1} M^A_{p'} M^B_{q'}\, \mathbb{1}(A_{p'} = B_{q'})\right)\right]$$

$$= \sum_{p=0}^{L-1}\sum_{q=0}^{L-1}\sum_{p'=0}^{L-1}\sum_{q'=0}^{L-1} \mathbb{E}[M_p^A M_q^B M_{p'}^A M_{q'}^B]\,\mathbb{1}(A_p = B_q)\,\mathbb{1}(A_{p'} = B_{q'}).$$

Observe that $M_p^A M_q^B$ and $M_{p'}^A M_{q'}^B$ are independent if $|p-p'| > 2(w-1)$, $|q-q'| > 2(w-1)$, $w_p \cap w_{q'} = \varnothing$, and $w_{p'} \cap w_q = \varnothing$, since these four conditions guarantee that the two windows of size $2w-1$ centered at p and q (which determine $M_p^A M_q^B$) do not share k-mers with the two windows centered of size $2w-1$ at p' and q' (which determine $M_{p'}^A M_{q'}^B$).

Let D bet the set of tuples (p,q,p',q') such that $p,q,p',q' \in [0,L)$, $A_p = B_q$, $A_{p'} = B_{q'}$ and at least one of the following conditions hold: (i) $|p-p'| \leqslant 2(w-1)$, (ii) $|q-q'| \leqslant 2(w-1)$, (iii) $w_p \cap w_{q'} \neq \varnothing$, or (iv) $w_{p'} \cap w_q \neq \varnothing$. That is, D contains all tuples (p,q,p',q') for which $M_p^A M_q^B$ and $M_{p'}^A M_{q'}^B$ could be dependent, so that

$$\mathbb{E}[\hat{I}^2] \leqslant |D| + \left(\sum_{p=0}^{L-1}\sum_{q=0}^{L-1} \mathbb{E}[M_p^A M_q^B]\,\mathbb{1}(A_p = B_q)\right)\left(\sum_{p'=0}^{L-1}\sum_{q'=0}^{L-1} \mathbb{E}[M_{p'}^A M_{q'}^B]\,\mathbb{1}(A_{p'} = B_{q'})\right) = |D| + \mathbb{E}[\hat{I}]^2.$$

Then, $Var(\hat{I}) = \mathbb{E}[\hat{I}^2] - \mathbb{E}[\hat{I}]^2 \leqslant |D|$ and it thus suffices to derive an upper bound for $|D|$. To do so, we will count the number of tuples that satisfy each of the conditions on the definition of D and add them together together to get an upper bound on $|D|$. For condition (i), there are I values of (p,q) such that $A_p = B_q$, and for each one, there are $4w-3$ possible values of p' such that $|p-p'| \leqslant 2(w-1)$. Then, for a given value of p', there is at most one value of q' that would satisfy $A_{p'} = B_{q'}$. Therefore there are at most $(4w-3)I$ values of (p,q,p',q') that satisfy condition (i), i.e. $A_p = B_q$, $A_{p'} = B_{q'}$ and $|p-p'| \leqslant 2(w-1)$. By the same logic, there are at most $(4w-3)I$ values of (p,q,p',q') that satisfy condition (ii), i.e. $A_p = B_q$, $A_{p'} = B_{q'}$ and $|q-q'| \leqslant 2(w-1)$.

For condition (iii), again there are I values of (p,q) such that $A_p = B_q$. Then, each k-mer $x \in w_p$ can occur at most once in B, hence there are at most $2w-1$ values of q' such that $x \in w_{q'}$. Since $|w_p| = 2w-1$, there are at most $(2w-1)^2$ values of q' such that $w_p \cap w_{q'} \neq \varnothing$. For each value of q', there is at most one value of p' such that $B_{q'} = A_{p'}$. Therefore, there are at most $I(2w-1)^2$ values of (p,q,p',q') that satisfy condition (iii), i.e. $A_p = B_q$, $A_{p'} = B_{q'}$ and $w_p \cap w_{q'} \neq \varnothing$. By symmetric logic, the number of tuples that satisfy condition (iv) is also $I(2w-1)^2$.

Putting this all together, we get $Var(\hat{I}) \leqslant |D| \leqslant 2(4w-3+(2w-1)^2)I \leqslant 8w^2 I$, which completes the proof of part (i).

We prove part (ii) next. For a k-mer $x \in U$, let U_x be the indicator random variable for the event that $x \in \hat{U}(A,B;w)$. Let D be the set of all (x,y) pairs such that $x \in U$, $y \in U$, and U_x and U_y are dependent. Then,

$$\mathbb{E}[\hat{U}^2] = \mathbb{E}\left[\sum_{x\in U} U_x \sum_{y\in U} U_y\right] = \sum_{x\in U}\sum_{y\in U} \mathbb{E}[U_x U_y] \leqslant |D| + \sum_{x\in U}\sum_{y\in U} \mathbb{E}[U_x]\mathbb{E}[U_y] = |D| + \mathbb{E}[\hat{U}]^2,$$

and $Var(\hat{U}) = \mathbb{E}[\hat{U}^2] - \mathbb{E}[\hat{U}]^2 \leqslant |D|$. It thus suffices to derive an upper bound for $|D|$. Let x and y belong to U. If U_x and U_y are dependent, then at least one of the following holds:

(i) One of the sequences (i.e. either A or B) contains both x and y at a distance of at most $2(w-1)$.
(ii) A contains x, B contains y, and the nearby k-mers of x in A intersect with the nearby k-mers of y in B.
(iii) B contains x, A contains y, and the nearby k-mers of x in B intersect with the nearby k-mers of y in A.

We will count the possible number of (x,y) pairs that satisfy each of the conditions and use their sum as an upper bound on $|D|$. For (i), there are 2 choices for which sequence contains x and y, at most L choices for the position of x, and at most $4w-3$ choices for the position of y. Hence, there are at most $2L(4w-3)$ choices for x and y that satisfy (i). For (ii), there are at most L choices for the position of x. If y satisfies the condition, then there must exist a k-mer z which is nearby to x in A and also nearby to y in B. There are at most $4w-3$ choices for z, and, for each of those choices, there are at most $4w-3$ locations for y. Hence, there are at most $L(4w-3)^2$ choices for x and y that satisfy (ii). Case (iii) is symmetrical to case (ii). In total then, $|D| \leqslant 2L(4w-3) + 2L(4w-3)^2 \leqslant 32w^2 L$. □

With these bounds for the variances of $\hat{I}(A,B;w)$ and $\hat{U}(A,B;w)$ we can now prove Lemma 5.

Lemma 5. $\left|\mathbb{E}\left[\frac{\hat{I}}{\hat{U}}\right] - \frac{\mathbb{E}[\hat{I}]}{\mathbb{E}[\hat{U}]}\right| \leqslant \frac{11w^2}{\sqrt[3]{L}}$.

Proof. We start by introducing some convenient notation. Let $c = \sqrt[6]{L}$, $\sigma_{\mathrm{i}} = \sqrt{Var(\hat{I})}$ and $\sigma_{\mathrm{u}} = \sqrt{Var(\hat{U})}$. We say that $\hat{I}$ and $\hat{U}$ are *good* if their values lie in the range $\mathbb{E}[\hat{I}] \pm c\sigma_{\mathrm{i}}$ and $\mathbb{E}[\hat{U}] \pm c\,\sigma_{\mathrm{u}}$, respectively; otherwise we say they are *bad*. Let $\hat{R} = \hat{I}/\hat{U}$. Note that $\mathbb{E}[\hat{R}] = T_1 + T_2$, where

$$T_1 = \mathbb{E}\left[\hat{R} \mid \ \hat{I} \text{ and } \hat{U} \text{ are good}\right] \Pr[\hat{I} \text{ and } \hat{U} \text{ are good}],$$
$$T_2 = \mathbb{E}\left[\hat{R} \mid \ \hat{I} \text{ or } \hat{U} \text{ are bad}\right] \Pr[\hat{I} \text{ or } \hat{U} \text{ are bad}].$$

We will bound T_1 and T_2 separately. Observe that by Chebyshev's inequality [22], the probability that $\hat{I}$ is bad is at most c^{-2} and the same holds for $\hat{U}$. Hence, a union bound implies that $\Pr[\hat{I} \text{ or } \hat{U} \text{ are bad}] \leqslant 2c^{-2}$. Since $\hat{I} \leqslant \hat{U}$, $\hat{R} \leqslant 1$, and we obtain the following bounds for T_2:

$$0 \leqslant T_2 \leqslant \Pr\left[\hat{I} \text{ or } \hat{U} \text{ are bad}\right] \leqslant 2c^{-2}.$$

For T_1, observe that

$$\mathbb{E}\left[\hat{R} \mid \hat{I} \text{ and } \hat{U} \text{ are good}\right] \leqslant \mathbb{E}\left[\frac{\mathbb{E}[\hat{I}] + c\sigma_{\mathrm{i}}}{\mathbb{E}[\hat{U}] - c\,\sigma_{\mathrm{u}}}\right] \leqslant \frac{\mathbb{E}[\hat{I}] + c\sigma_{\mathrm{i}}}{\mathbb{E}[\hat{U}] - c\,\sigma_{\mathrm{u}}},$$
$$\mathbb{E}\left[\hat{R} \mid \hat{I} \text{ and } \hat{U} \text{ are good}\right] \geqslant \mathbb{E}\left[\frac{\mathbb{E}[\hat{I}] - c\sigma_{\mathrm{i}}}{\mathbb{E}[\hat{U}] + c\,\sigma_{\mathrm{u}}}\right] \geqslant \frac{\mathbb{E}[\hat{I}] - c\sigma_{\mathrm{i}}}{\mathbb{E}[\hat{U}] + c\,\sigma_{\mathrm{u}}}.$$

Also, since $\Pr[\hat{I} \text{ or } \hat{U} \text{ are bad}] \leqslant 2c^{-2}$, we have $\Pr[\hat{I} \text{ and } \hat{U} \text{ are good}] \geqslant 1 - 2c^{-2}$, and so

$$\frac{\mathbb{E}[\hat{I}] - c\,\sigma_{\mathrm{i}}}{\mathbb{E}[\hat{U}] + c\,\sigma_{\mathrm{u}}}(1 - 2c^{-2}) \leqslant T_1 \leqslant \frac{\mathbb{E}[\hat{I}] + c\,\sigma_{\mathrm{i}}}{\mathbb{E}[\hat{U}] - c\,\sigma_{\mathrm{u}}}.$$

Now, observe that $\frac{a}{b} \geqslant \frac{a-x}{b-x}$, for $0 < a \leqslant b$ and $0 \leqslant x < b$ and $\mathbb{E}[\hat{I}] - c\,\sigma_{\mathrm{i}} \leqslant \mathbb{E}[\hat{U}] + c\,\sigma_{\mathrm{u}}$, since $\mathbb{E}[\hat{I}] \leqslant \mathbb{E}[\hat{U}]$ and $c \geqslant 0$.

$$\mathbb{E}[\hat{R}] - \frac{\mathbb{E}[\hat{I}]}{\mathbb{E}[\hat{U}]} = T_1 + T_2 - \frac{\mathbb{E}[\hat{I}]}{\mathbb{E}[\hat{U}]} \geqslant T_1 - \frac{\mathbb{E}[\hat{I}]}{\mathbb{E}[\hat{U}]} \geqslant \frac{\mathbb{E}[\hat{I}] - c\,\sigma_{\mathrm{i}}}{\mathbb{E}[\hat{U}] + c\,\sigma_{\mathrm{u}}}\left(1 - 2c^{-2}\right) - \frac{\mathbb{E}[\hat{I}]}{\mathbb{E}[\hat{U}]} \geqslant \frac{\mathbb{E}[\hat{I}] - c(\sigma_{\mathrm{i}} + \sigma_{\mathrm{u}})}{\mathbb{E}[\hat{U}]}(1 - 2c^{-2}) - \frac{\mathbb{E}[\hat{I}]}{\mathbb{E}[\hat{U}]}. \quad (11)$$

Observe that for all $x > 0$ and $y > 0$, $\sqrt{x} + \sqrt{y} \leqslant \sqrt{x+y} + \sqrt{x+y} = \sqrt{2(x+y)}$. Then, using Lemma 4, we get:

$$\sigma_{\mathrm{i}} + \sigma_{\mathrm{u}} \leqslant \sqrt{2(Var(\hat{I}) + Var(\hat{U}))} \leqslant \sqrt{80w^2L}.$$

Furthermore, since every w consecutive k-mers have at least one minimizer, $\hat{U} \geqslant L/w$, and so

$$\frac{c(\sigma_{\mathrm{i}} + \sigma_{\mathrm{u}})}{\mathbb{E}[\hat{U}]} \leqslant \frac{L^{1/6}\sqrt{80w^2L}}{L/w} \leqslant \frac{\sqrt{80}w^2}{\sqrt[3]{L}}. \quad (12)$$

Plugging this bound into Equation (11) we get

$$\mathbb{E}[\hat{R}] - \frac{\mathbb{E}[\hat{I}]}{\mathbb{E}[\hat{U}]} \geqslant \left(\frac{\mathbb{E}[\hat{I}]}{\mathbb{E}[\hat{U}]} - \frac{\sqrt{80}w^2}{\sqrt[3]{L}}\right)\left(1 - \frac{2}{\sqrt[3]{L}}\right) - \frac{\mathbb{E}[\hat{I}]}{\mathbb{E}[\hat{U}]} = -\frac{\sqrt{80}w^2}{\sqrt[3]{L}}\left(1 - \frac{2}{\sqrt[3]{L}}\right) - \frac{\mathbb{E}[\hat{I}]}{\mathbb{E}[\hat{U}]}\frac{2}{\sqrt[3]{L}} \geqslant -\frac{\sqrt{80}w^2}{\sqrt[3]{L}} - \frac{2}{\sqrt[3]{L}} \geqslant -\frac{11w^2}{\sqrt[3]{L}}. \quad (13)$$

To derive the upper bound for $\mathbb{E}[\hat{R}] - \mathbb{E}[\hat{I}]/\mathbb{E}[\hat{U}]$, we first consider the case when $\mathbb{E}[\hat{U}] - \mathbb{E}[\hat{I}] < c\,(\sigma_{\mathrm{i}} + \sigma_{\mathrm{u}})$. Under this assumption,

$$\mathbb{E}[\hat{R}] - \frac{\mathbb{E}[\hat{I}]}{\mathbb{E}[\hat{U}]} \leqslant 1 - \frac{\mathbb{E}[\hat{I}]}{\mathbb{E}[\hat{U}]} = \frac{\mathbb{E}[\hat{U}] - \mathbb{E}[\hat{I}]}{\mathbb{E}[\hat{U}]} < \frac{c\,(\sigma_{\mathrm{i}} + \sigma_{\mathrm{u}})}{\mathbb{E}[\hat{U}]} \leqslant \frac{\sqrt{80}w^2}{\sqrt[3]{L}},$$

where the last inequality follows from Equation (12)

Now consider the case when $\mathbb{E}[\hat{U}] - \mathbb{E}[\hat{I}] \geqslant c\,(\sigma_{\mathrm{i}} + \sigma_{\mathrm{u}})$. Using the fact that $\frac{a}{b} \leqslant \frac{a+x}{b+x}$, for $0 < a \leqslant b$ and $x \geqslant 0$, we obtain

$$T_1 \leqslant \frac{\mathbb{E}[\hat{I}] + c\sigma_{\mathrm{i}}}{\mathbb{E}[\hat{U}] - c\,\sigma_{\mathrm{u}}} \leqslant \frac{\mathbb{E}[\hat{I}] + c(\sigma_{\mathrm{i}} + \sigma_{\mathrm{u}})}{\mathbb{E}[\hat{U}]} \leqslant \frac{\mathbb{E}[\hat{I}]}{\mathbb{E}[\hat{U}]} + \frac{\sqrt{80}w^2}{\sqrt[3]{L}},$$

where the last inequality follows from Equation (12).

Putting the upper bounds on T_1 and T_2 together we get

$$\mathbb{E}[\hat{R}] - \frac{\mathbb{E}[\hat{I}]}{\mathbb{E}[\hat{U}]} = T_1 + T_2 - \frac{\mathbb{E}[\hat{I}]}{\mathbb{E}[\hat{U}]} \leqslant \frac{\sqrt{80}w^2}{\sqrt[3]{L}} + 2c^{-2} \leqslant \frac{\sqrt{80}w^2 + 2}{\sqrt[3]{L}} \leqslant \frac{11w^2}{\sqrt[3]{L}}.$$

Combined with Equation (13) this implies the result. □

A.2.3 Proof of Theorem 1

To prove Theorem 1, we need to relate the bound on $\hat{J}(A, B; w)$ given by Lemma 2 to the values of $J(A, B)$. We first express $J(A, B)$ in terms of configuration numbers. Let $\mathcal{D}(A, B; w) = \sum_{s=0}^{w} N(\llbracket *, 0; *, 0; s \rrbracket)$. Note that, except near the start of the sequences, $A_i = B_j$ if and only if $(i - w, j - w)$ are in a configuration $\llbracket *, 0; *, 0; s \rrbracket$. Therefore, $\mathcal{D}(A, B; w)$ is approximately $I(A, B)$. Formally, we can prove:

Lemma 6. *If A and B are padded, then $\mathcal{D}(A, B; w) = I(A, B)$ and $J(A, B) = \frac{\mathcal{D}(A,B;w)}{2L - \mathcal{D}(A,B;w)}$. More generally,*

(i) $\mathcal{D}(A, B; w) \leqslant I(A, B) \leqslant \mathcal{D}(A, B; w) + 2w$;

(ii) $\frac{\mathcal{D}(A,B;w)}{2L-\mathcal{D}(A,B;w)} \leqslant J(A,B) \leqslant \frac{\mathcal{D}(A,B;w)}{2L-\mathcal{D}(A,B;w)} + \frac{4w}{L}$.

Proof. Observe that for $i \in [w, L-1]$ and $j \in [w, L-1]$, we have $A_i = B_j$ if and only if $(i-w, j-w)$ are in a configuration with $C_{\text{a,right}} = C_{\text{b,right}} = 0$. In the case that A and B are padded, then $I = \mathcal{D}$ and $J = \frac{I}{2L-I} = \frac{\mathcal{D}}{2L-\mathcal{D}}$. In general, the number of (i,j) pairs for which $A_i = B_j$ and either $i \in [0, w-1]$ or $j \in [0, w-1]$ is at most $2w$. Hence, $\mathcal{D} \leqslant I \leqslant \mathcal{D} + 2w$. For the J lower bound, $J = \frac{I}{2L-I} \geqslant \frac{\mathcal{D}}{2L-\mathcal{D}}$. For the J upper bound, $J \leqslant \frac{\mathcal{D}+2w}{2L-\mathcal{D}-2w}$. When $\mathcal{D} + 2w \leqslant L$, then

$$J(A,B) \leqslant \frac{\mathcal{D}+2w+2w}{2L-\mathcal{D}-2w+2w} = \frac{\mathcal{D}}{2L-\mathcal{D}} + \frac{4w}{2L-\mathcal{D}} \leqslant \frac{\mathcal{D}}{2L-\mathcal{D}} + \frac{4w}{L}.$$

When $\mathcal{D} + 2w > L$, then

$$\frac{\mathcal{D}}{2L-\mathcal{D}} + \frac{4w}{L} \geqslant \frac{L-2w}{L+2w} + \frac{4w}{L} \geqslant \frac{L-4w}{L} + \frac{4w}{L} = 1 \geqslant J.$$

□

We note that it is possible to derive exact expressions for $I(A,B;w)$ and $J(A,B;w)$ for the non-padded case as well; however, doing so is not necessary for our purposes and would just introduce (even more) burdensome notation. Next, we need to prove two facts:

Fact 2. $\mathcal{C}(A,B;w) \leqslant \frac{2L}{w+1}$.

Proof. By Lemma 1, the definition of $\widehat{I}$, and Fact 4, we have $\mathcal{C}(A,B;w) \leqslant \mathbb{E}[\widehat{I}(A,B;w)] = \sum_{p=0}^{L-1} \mathbb{E}[M_p^A] \leqslant \frac{2L}{w+1}$. □

Fact 3. *For all* $y > 20$ *and* $0 < x \leqslant y/2$, $\frac{x+2}{y-x-10} - \frac{x}{y-x} \leqslant \frac{12}{y-y}$.

Proof. Note that under the given assumptions, $y - x \geqslant y/2 > 0$ and $y - x - 10 \geqslant y/2 - 10 > 0$. Therefore,

$$\frac{x+2}{y-x-10} - \frac{x}{y-x} = \frac{2y+8x}{(y-x)(y-x-10)} \leqslant \frac{2y+\frac{8y}{2}}{\frac{y}{2}(\frac{y}{2}-10)} = \frac{12}{y-5}.$$

□

Now, we are ready to prove Theorem 1

Theorem 1. *Let* $w \geqslant 2$, $k \geqslant 2$, *and* $L \geqslant 7(w+1)$ *be integers. Let* A *and* B *be two duplicate-free sequences, each consisting of* L k*-mers. Then there exists* $\varepsilon \in [0, \frac{15w^2}{\sqrt[3]{L}}]$ *such that*

$$\mathcal{B}(A,B,w) - \varepsilon \leqslant \mathbb{E}[\widehat{J}(A,B;w)] - J(A,B) \leqslant \mathcal{B}(A,B,w) + \varepsilon.$$

Proof. We prove the upper bound first. From Lemmas 2 and 6, we know that

$$\mathbb{E}[\widehat{J}(A,B;w)] - J(A,B) \leqslant \frac{\mathcal{C}(A,B;w)}{\frac{4L}{w+1} - \mathcal{C}(A,B;w)} + \frac{15w^2}{\sqrt[3]{L}} - \frac{\mathcal{D}(A,B;w)}{2L-\mathcal{D}(A,B;w)} = \mathcal{B}(A,B;w) + \frac{15w^2}{\sqrt[3]{L}}.$$

For the lower bound, we have

$$\begin{aligned}
\mathbb{E}[\widehat{J}(A,B;w)] - J(A,B) &= \frac{\mathcal{C}(A,B;w)}{\frac{4L}{w+1} - \mathcal{C}(A,B;w)} - \frac{11w^2}{\sqrt[3]{L}} - J(A,B) && \text{(Lemma 2)} \\
&\geqslant \frac{\mathcal{C}(A,B;w)}{\frac{4L}{w+1} - \mathcal{C}(A,B;w)} - \frac{11w^2}{\sqrt[3]{L}} - \frac{\mathcal{D}}{2L-\mathcal{D}} - \frac{4w}{L} && \text{(Lemma 6)} \\
&= \mathcal{B}(A,B;w) - \frac{11w^2}{\sqrt[3]{L}} - \frac{4w}{L} \\
&\geqslant \mathcal{B}(A,B;w) - \frac{11w^2+4w}{\sqrt[3]{L}} \geqslant \mathcal{B}(A,B;w) - \frac{11w^2+2w^2}{\sqrt[3]{L}} \geqslant \mathcal{B}(A,B;w) - \frac{13w^2}{\sqrt[3]{L}},
\end{aligned}$$

as claimed. □

A.3 Proof of Theorem 2

Theorem 2. *Let $w \geqslant 2$, $k \geqslant 2$, and $L \geqslant 7(w+1)$ be integers. Let A and B be two duplicate-free padded sequences, each consisting of L k-mers. Then $\mathcal{B}(A,B;w) < 0$ unless $J(A,B) = 0$; when $J(A,B) = 0$, we have $\mathcal{B}(A,B;w) = 0$.*

Proof. We omit the parameters A, B and w from the following for conciseness. Let $d = \frac{2}{w+1}$. Observe that the following statements are equivalent:

$$\begin{aligned}
\mathcal{B} \leqslant 0 &\Leftrightarrow \frac{\mathcal{C}}{2dL - \mathcal{C}} \leqslant \frac{\mathcal{D}}{2L - \mathcal{D}} \\
&\Leftrightarrow \mathcal{C}(2L - \mathcal{D}) \leqslant \mathcal{D}(2dL - \mathcal{C}) \\
&\Leftrightarrow 2L\mathcal{C} - \mathcal{D}\mathcal{C} \leqslant 2dL\mathcal{D} - \mathcal{D}\mathcal{C} \\
&\Leftrightarrow 2L\mathcal{C} \leqslant 2dL\mathcal{D} \\
&\Leftrightarrow \mathcal{C} \leqslant d\mathcal{D}
\end{aligned}$$

Note that for the second equivalence, we rely on the fact $\mathcal{B}(A,B;w)$ is well defined and its denominators are not zero. In other words, 1) $2L - \mathcal{D} > 0$ because $\mathcal{D} \leqslant L$ (by definition) and 2) $2dL - \mathcal{C} > 0$ because $\mathcal{C} \leqslant dL$ (by Fact 2).

We now need to show that $\mathcal{C} \leqslant d\mathcal{D}$. We have

$$\begin{aligned}
\mathcal{C} &\leqslant \mathbb{E}[\widehat{I}] && \text{(by Lemma 1)} \\
&= \sum_{p=0}^{L-1}\sum_{q=0}^{L-1} \mathbb{1}(A_p = B_q)\Pr[M_p^A = 1, M_q^B = 1] \\
&= \sum_{p=0}^{L-1}\sum_{q=0}^{L-1} \mathbb{1}(A_p = B_q)\Pr[M_p^A = 1 \mid M_q^B = 1]\Pr[M_q^B = 1] \\
&= \sum_{p=0}^{L-1}\sum_{q=0}^{L-1} \mathbb{1}(A_p = B_q)\Pr[M_p^A = 1 \mid M_q^B = 1]d && (14) \\
&\leqslant Id \\
&= d\mathcal{D} && \text{(by Lemma 6)}
\end{aligned}$$

Note that Equation (14) follows because of the fact that A and B are padded and Fact 4. Next, observe that since all the terms in Equation (14) are positive, the only way to have equality with Id is if each term $\Pr[M_p^A = 1 \mid M_q^B = 1]$ is 1. We claim this can only happen if there are no shared k-mers between A and B, i.e. when $J(A,B) = 0$. Otherwise, take the leftmost shared k-mer in A. The window to its left in A will be assigned hash values that are independent of the hash values in B; therefore, $\Pr[M_p^A = 1 \mid M_q^B = 1]$ cannot be 1. Thus, if A and B share at least one k-mer, we get the stronger statement that $\mathbb{E}[\widehat{I}(A,B;w)] < Id$. This in turn implies that $\mathcal{C} < d\mathcal{D}$, which propagates to imply that $\mathcal{B} < 0$. □

A.4 Proof of Theorem 3

Theorem 3. *Let $w \geqslant 2$, $k \geqslant 2$, and $L \geqslant 7(w+1)$ be integers. Let A and B be two duplicate-free, padded, sparsely-matched sequences, each consisting of L k-mers. Then $\mathcal{B}(A,B;w) \leqslant -J(A,B)\frac{3w^2-3w}{8w^2-2}$.*

Proof. This proof simply counts the configuration numbers and then applies definitions and Theorem 1. We will first count the configuration numbers. Let us call $[\![2,2;2,2;0]\!]$ the *empty* configuration. Note that the terms involving the number of empty configurations cancel out in the equation for $\mathcal{C}$ and hence we do not need to count them. Observe, by the condition of the theorem, that a configuration (i,j) that is non-empty must contain exactly one pair $p \in [i, i+w]$ and $q \in [j, j+w]$ such that $A_p = B_q$. Therefore, to count the number of non-empty configurations, it suffices to count, for every $p \in [0, L-1]$ and $q \in [0, L-1]$ such that $A_p = B_q$, the types of configurations (i,j) for $i \in [p-w, p]$ and $j \in [q-w, q]$. Following a case analysis, we get one configuration of $[\![2,0;2,0;1]\!]$, $w-1$ configurations of $[\![2,1;2,2;1]\!]$, $w-1$ configurations of $[\![2,2;2,1;1]\!]$, $(w-1)^2$ configurations of $[\![2,2;2,2;1]\!]$, one configuration of $[\![0,2;0,2;0]\!]$, w configurations of $[\![1,2;2,2;0]\!]$, and w configurations of $[\![2,2;1,2;0]\!]$. Recall that $I = I(A,B)$ is the number of shared k-mers between A and B. Summing over all I values of p, we then get the non-zero configuration number of non-empty configurations are

$$\begin{aligned}
N([\![2,0;2,0;1]\!]) &= I \\
N([\![2,1;2,2;1]\!]) &= I(w-1) \\
N([\![2,2;2,1;1]\!]) &= I(w-1) \\
N([\![2,2;2,2;1]\!]) &= I(w-1)^2 \\
N([\![1,2;2,2;0]\!]) &= Iw
\end{aligned}$$

$$N(\llbracket 2,2;1,2;0\rrbracket) = Iw$$
$$N(\llbracket 0,2;0,2;0\rrbracket) = I.$$

We then plug these into the definition of $\mathcal{C}$ to get that $\mathcal{C}(A,B;w) = \beta I$, where $\beta = \frac{5w-2}{4w^2-1}$. By Lemma 6, $\mathcal{D}(A,B;w) = I$. Let $d \triangleq 2/(w+1)$. Note that $\beta - d = \frac{3w^2-3w}{-(w+1)(4w^2-1)} \leqslant 0$. Using these facts, we can now derive

$$\begin{aligned}\mathcal{B}(A,B;w) &\triangleq \frac{(w+1)\mathcal{C}(A,B;w)}{4L-(w+1)\mathcal{C}(A,B;w)} - \frac{\mathcal{D}(A,B;w)}{2L-\mathcal{D}(A,B;w)} = \frac{\mathcal{C}(A,B;w)}{2dL-\mathcal{C}(A,B;w)} - \frac{I}{2L-I} = \frac{\beta I}{2dL-\beta I} - \frac{I}{2L-I} \\ &= \frac{(2L-I)\beta I - 2dLI + \beta I^2}{(2dL-\beta I)(2L-I)} = \frac{2L\beta I - 2dLI}{(2dL-\beta I)(2L-I)} = J(A,B)\frac{2L\beta - 2dL}{2dL-\beta I} = J(A,B)\frac{2L(\beta-d)}{2dL-\beta I}.\end{aligned}$$

Note that because $\beta - d \leqslant 0$, $\mathcal{B}(A,B;w) \leqslant 0$. Then, using the fact that $\beta > 0$ and $I > 0$, we get

$$\mathcal{B}(A,B;w) \leqslant J(A,B)\frac{2L(\beta-d)}{2dL} = J(A,B)\frac{3w^2-3w}{-2(4w^2-1)}.$$

□

A.5 Proof of Theorem 4

Theorem 4. *Let $2 \leqslant w < k$, $g > w + 2k$, and $L = \ell g + k$ for some integer $\ell \geqslant 1$. Let A and B be two duplicate-free sequences with L k-mers such that A and B are identical except that the nucleotides at positions $k - 1 + ig$, for $i = 0, \ldots, \ell$, are mutated. Then,*

$$\mathcal{B}(A,B;w) = \frac{2\ell(\ell g+k)h(w)}{(\ell(g+k)+2k-\ell h(w))(\ell(g+k)+2k)},$$

where $h(w) = \frac{(w+1)(1-2(H_{2w}-H_w))}{2}$ and $H_n = \sum_{j=1}^{n}\frac{1}{j}$ denotes the n-th Harmonic number.

Proof. Let

$$\begin{aligned}W(s) = {} & t_0(N(\llbracket 1,0;1,0;s\rrbracket) + N(\llbracket 1,0;2,0;s\rrbracket) + N(\llbracket 2,0;1,0;s\rrbracket)) \\ & + t_1(N(\llbracket 2,\{1,2\};1,1;s\rrbracket) + N(\llbracket 1,1;2,\{1,2\};s\rrbracket) + 2wN(\llbracket 0,0;0,0;s\rrbracket)) \\ & + t_1 s(N(\llbracket 0,1;0,1;s\rrbracket) + N(\llbracket 0,1;0,2;s\rrbracket) + N(\llbracket 0,2;0,1;s\rrbracket) + N(\llbracket 0,2;0,2;s\rrbracket)) \\ & + t_2(2sN(\llbracket 2,2;2,2;s\rrbracket) + 4wN(\llbracket 2,1;2,1;s\rrbracket) + (6w-s+(2w-s)^2)N(\llbracket 2,0;2,0;s\rrbracket)) \\ & + t_2(s+2w)(N(\llbracket 2,1;2,2;s\rrbracket) + N(\llbracket 2,2;2,1;s\rrbracket))\end{aligned}$$

so that $\mathcal{C}(A,B;w) = \sum_{s=0}^{w} W(s)$. In our setting, the configuration counts are such that the following holds:

Fact 6.

$$W(s) = \begin{cases} 0 & \text{if } s = 0; \\ \frac{2\ell(g-w-k)}{w+1} + \frac{\ell(w+5)}{(w+1)(w+2)} & \text{if } s = w; \\ \ell s t_1 + \ell t_2(6w+8w^2-s(s+6w+1)) & \text{if } 1 \leqslant s \leqslant w-1. \end{cases}$$

From this fact, which we prove later, we get that $\mathcal{C}(A,B;w) = d\ell(g-k) + \ell f(w)$, where $d = 2/(w+1)$ and

$$f(w) = -\frac{2w}{w+1} + \frac{w+5}{(w+1)(w+2)} + \sum_{s=1}^{w-1} st_1 + t_2(6w+8w^2-s(s+6w+1)).$$

Note that since there are no matches in the first or the last k-mers and $k \geqslant w$, we have by Lemma 6 that $I = |A \cap B| = \mathcal{D}(A,B;w) = \ell(g-k)$ and so

$$\mathcal{C}(A,B;w) = dI + \ell f(w),$$

From the definition of $\mathcal{B}(A,B;w)$, we then have

$$\mathcal{B}(A,B;w) = \frac{\mathcal{C}}{2dL-\mathcal{C}} - \frac{I}{2L-I} = \frac{I + \frac{\ell f(w)}{d}}{2L - I - \frac{\ell f(w)}{d}} - \frac{I}{2L-I} = \frac{2L\frac{\ell f(w)}{d}}{(2L-I-\frac{\ell f(w)}{d})(2L-I)}.$$

We also have the following closed form for $f(w)$ (which we prove later).

Fact 7. *For $n \geqslant 1$, let $H_n = \sum_{k=1}^{n}\frac{1}{k}$. Then, $f(w) = 1 - 2(H_{2w} - H_w)$.*

configuration	count	reason for 0	configuration	count	reason for 0
$[\![0,2;0,2;<w]\!]$	ℓ	N/A	$[\![0,2;0,2;w]\!]$	0	TOO-FULL
$[\![2,2;2,2;>0]\!]$	0	see text	$[\![2,1;2,1;s]\!]$	0	CROSS
$[\![0,0;0,0;<w]\!]$	0	see text	$[\![0,0;0,0;w]\!]$	$\ell(g-w-k)$	N/A
$[\![1,0;1,0;s]\!]$	0	VERT	$[\![2,0;1,0;s]\!]$	0	VERT
$[\![1,0;2,0;s]\!]$	0	VERT	$[\![2,0;2,0;0]\!]$	0	TOO-EMPTY
$[\![2,0;2,0;>0]\!]$	ℓ	N/A	$[\![0,1;0,1;s]\!]$	0	VERT
$[\![0,2;0,1;s]\!]$	0	VERT	$[\![2,1;1,1;s]\!]$	0	CROSS
$[\![2,2;1,1;s]\!]$	0	CROSS	$[\![2,1;2,1;s]\!]$	0	CROSS
$[\![2,2;2,1;0]\!]$	0	TOO-EMPTY	$[\![2,2;2,1;1\cdots w-1]\!]$	$\ell(w-s)$	N/A
$[\![2,2;2,1;w]\!]$	0	TOO-FULL	$[\![0,1;0,2;s]\!]$	0	VERT
$[\![1,1;2,1;s]\!]$	0	CROSS	$[\![1,1;2,2;s]\!]$	0	CROSS
$[\![2,1;2,2;0]\!]$	0	TOO-EMPTY	$[\![2,1;2,2;1\cdots w-1]\!]$	$\ell(w-s)$	N/A
$[\![2,1;2,2;w]\!]$	0	TOO-FULL			

Table S1. Non-empty configurations appearing in the definition of $\mathcal{C}$, along with their counts in the context of Theorem 4 as well as why the counts are zero, if applicable. The reasons are explained in the proof of Fact 8.

From this, combined with the facts that $L = \ell g + k$ and $I = \ell(g-k)$, and letting $h(w) = \frac{(w+1)(1-2(H_{2w}-H_w))}{2}$, we get

$$\mathcal{B}(A,B;w) = \frac{2\ell(\ell g+k)h(w)}{(\ell(g+k)+2k-\ell h(w))(\ell(g+k)+2k)},$$

as claimed. □

It remains for use to provide the proofs of Facts 6 and 7. Fact 6 is a direct consequence of the following configuration counts.

Fact 8. *In the setting of Theorem* 4*, we have*

(i) $N([\![0,0;0,0;w]\!]) = l(g-w-k)$;
(ii) $N([\![0,2;0,2;\{0,\cdots,w-1\}]\!]) = l$;
(iii) $N([\![2,0;2,0;\{1,\cdots,w\}]\!]) = l$;
(iv) $N([\![2,1;2,2;\{1,\cdots,w-1\}]\!]) = l(w-s)$;
(v) $N([\![2,2;2,1;\{1,\cdots,w-1\}]\!]) = l(w-s)$.

For any other configuration c *that could contribute to* $\mathcal{C}(A,B;w)$*, we have* $N(c) = 0$ *or* $c = [\![2,2;2,2;0]\!]$.

Proof. We will refer to $[\![2,2;2,2;0]\!]$ as the *empty* configuration. Table S1 lists all non-empty configurations that appear in the definition of $\mathcal{C}$. Sometimes, a configuration type is further sub-divided according to different values of s. We will show that the counts in the table are correct, which will prove the Theorem.

The rows that whose reason is VERT have configurations that match $[\![*,*;1,0;s]\!]$, $[\![*,*;0,1;s]\!]$, $[\![1,0;*,*;s]\!]$, or $[\![0,1;*,*;s]\!]$. These configurations never occur because in our setting, all the matches are parallel to each other (i.e. if $A_i = B_j$ and $A_{i'} = B_{j'}$, then $j-i = j'-i'$), while these configurations contain a 0 in one place (indicating that the matches are vertical, i.e. $A_i = B_j$ implies $i = j$) and a 1 in another (indicated that the matching edges are angled, i.e. $A_i = B_j$ implies $i \neq j$). The rows whose reason is CROSS have a configuration that matches $[\![1,*;1,*;s]\!]$, $[\![*,1;*,1;s]\!]$, $[\![1,1;*,*;s]\!]$, or $[\![*,*;1,1;s]\!]$. These configurations never occur because the 1s indicate conflicting angles for the matches — they should either slant left (e.g. $i > j$) or right (e.g. $i < j$), but cannot do both. Note that for rows that could be categorized as both VERT and CROSS, the reason in the Table is arbitrarily chosen from those two. The rows whose reason is TOO-FULL have a configuration that matches $[\![*,2;*,*;w]\!]$ or $[\![*,*;*,2;w]\!]$. These configurations can never occur because the presence of the 2 indicates that either A_{i+w} or B_{j+w} is not involved in a match, making it impossible that $S(i+1,j+1,w) = w$. The rows whose reason is TOO-EMPTY have a configuration that matches $[\![*,*;*,\{0,1\};0]\!]$ or $[\![*,\{0,1\};*,*;0]\!]$. These configurations can never occur because the presence of the 0 or 1 indicates that either A_{i+w} or B_{j+w} is involved in a match, making it impossible that $S(i+1,j+1,w) = 0$.

By the definition of A and B from Theorem 4, we have alternating runs of k mismatches followed by $g-k$ matches, with k mismatches at the end. Therefore, we have $\ell+1$ blocks of k mismatches, at $i \in \{ig, ..., ig+k-1 | 0 \leqslant i \leqslant \ell\}$, and we have ℓ blocks of $g-k$ matches, at $i \in \{ig+k, ..., (i+1)g-1 | 0 \leqslant i < \ell\}$. We will refer to the latter as *match-blocks*.

Recall that configuration windows are of length $w+1$. Because $k > w$, no window can contain matches from more than one match-block. Moreover, any configurations involving an i or j in the first match-block will occur again in each other match-block, at the same coordinates modulo g. Thus it is enough to consider only the first match-block, and multiply the resulting counts by ℓ. We therefore restrict ourselves to the first match-block in the following discussion, and note that the leftmost match is at position k and the rightmost match is at $g-1$.

Let us consider the configurations that are $[\![2,2;2,2;>0]\!]$. In this case, $A_i \neq B_j$ and $A_{i+w} \neq B_{j+w}$, and there is some $i' \in [i+1, i+w-1]$ and $j' \in [j+1, j+w-1]$ such that $A_{i'} = B_{j'}$. This match must be part of match block, and in our setting, a match block has width $g-k$. This is more than w, making it impossible that $A_i \neq B_j$ and $A_{i+w} \neq B_{j+w}$. Hence $N([\![2,2;2,2;>0]\!]) = 0$.

Let us consider the configurations that are $[\![0,0;0,0;s]\!]$. In these configuration, $i = j$, $A_i = B_j$, and $A_{i+w} = B_{j+w}$. A configuration window of width $w+1$ cannot span more than one match block, since $g > w$. Therefore, $A_{i+\delta} = B_{j+\delta}$ for all $0 \leqslant \delta \leqslant w$. Hence, the number of

configurations with $s < w$ is 0. For $s = w$, Figure S2A shows all the configurations that are $[\![0,0;0,0;w]\!]$. We have that $i \in [k, g-w-1]$, resulting in $g-w-k$ possible windows with this configuration, in one match block

Let us consider the configurations that are $[\![0,2;0,2;s]\!]$ for $0 \leqslant s \leqslant w-1$. In this situation, $A_i = B_j$ and hence $i = j$. The match block containing this match ends before A_{i+w}, since $A_{i+w} \neq B_{j+w}$ in this configuration. Then the rightmost match, $A_{g-1} = B_{g-1}$, must be somewhere in the window, other than at $i+w$. To get s matches, $g-1 = i+s$ and thus $i = g-s-1$. Therefore, $N([\![0,2;0,2;s]\!]) = 1$ for each $s \in [0, w-1]$. Figure S2B shows how this configuration looks like. The top and bottom drawings show the two end cases, while the middle drawing demonstrates the general case.

Let us consider the configurations that are $[\![2,0;2,0;s]\!]$ for $1 \leqslant s \leqslant w$. The case is mostly symmetric to the previous one. In this situation, $A_{i+w} = B_{j+w}$ and hence $i = j$. The match block containing this match begins after A_i, since $A_i \neq B_j$ in this configuration. The leftmost match in the match-block, A_k, must be somewhere in the window other than at A_i. To get s matches, $k = (i+w) - (s-1)$ and thus $i = k-w+s-1$. Therefore $N([\![2,0;2,0;s]\!]) = 1$ for each $s \in [1, w]$. Figure S2C shows how this configurations looks like. The top and bottom drawings show the two end cases, while the middle drawing demonstrates the general case.

Let us consider the configurations that are $[\![2,1;2,2;s]\!]$ for $1 \leqslant s \leqslant w-1$. Figure S2D shows all the configurations. There are several possibilities for each s. For $s = 3$, the top and bottom drawings show the two end cases, while the middle drawing demonstrates the general case. Because $C_{\text{a,right}} = 1$, $A_{i+w} \in \{B_{j+1}, \ldots, B_{j+w-1}\}$ and $j > i$. Since $C_{\text{a,left}} = C_{\text{b,left}} = 2$, $A_i \neq B_j$, and the leftmost match in the match-block, A_k, must be somewhere in the window, other than at i. To get s matches, $k = (i+w) - (s-1)$ and thus $i = k-w+s-1$. The window for B can be positioned so that the leftmost match occurs in $\{j+1, \ldots, j+w-s\}$. Since this corresponds to A_k, we have $k \in \{j+1, \ldots, j+w-s\}$, which can be restated as $(i+w)-(s-1) \in \{j+1, \ldots, j+w-s\}$. We can in turn restate this as $i \in \{j-w+s, \ldots, j-1)\}$ and thus $j \in \{i+1, \ldots, i+w-s)\}$. Therefore, $N([\![2,1;2,2;s]\!]) = w-s$ for each $s \in [1, w-1]$.

Finally, we consider the configurations that are $[\![2,2;1,2;s]\!]$ for $1 \leqslant s \leqslant w-1$. This case is symmetrical to the above case, by swapping the roles of A and B in the definition of the configurations. Therefore, $N([\![2,2;1,2;s]\!]) = w-s$ for each $1 \leqslant s \leqslant w-1$. □

We are now ready to prove Fact 6.

Fact 6.

$$W(s) = \begin{cases} 0 & \text{if } s = 0; \\ \frac{2\ell(g-w-k)}{w+1} + \frac{\ell(w+5)}{(w+1)(w+2)} & \text{if } s = w; \\ \ell s t_1 + \ell t_2(6w + 8w^2 - s(s+6w+1)) & \text{if } 1 \leqslant s \leqslant w-1. \end{cases}$$

Proof. Let us consider first the $s = 0$ case. By Fact 8, the only two configurations with $s = 0$ and with non zero counts are $[\![2,2;2,2;0]\!]$ and $[\![0,2;0,2;0]\!]$. However, both of those terms are multiplied by s in $W(0)$, hence we have $W(0) = 0$.

Let us consider next the $s = w$ case. For this value of s, by Fact 8, we have $N([\![0,0;0,0;w]\!]) = l(g-w-k)$ and $N([\![2,0;2,0;w]\!]) = l$; all other configurations that may contribute to $\mathcal{C}(A, B; w)$ have zero counts.

At $s = w$, $[\![0,0;0,0;w]\!]$ has coefficient $\frac{2}{w+1}$ and $[\![2,0;2,0;w]\!]$ has coefficient $\frac{w+5}{(w+1)(w+2)}$. Hence

$$W(w) = \frac{2l(g-w-k)}{w+1} + \frac{l(w+5)}{(w+1)(w+2)}.$$

Finally, when $1 \leqslant s \leqslant w-1$, again by Fact 8, we have

$$N([\![0,2;0,2;s]\!]) = N([\![2,0;2,0;s]\!]) = l,$$
$$N([\![2,1;2,2;s]\!]) = N([\![2,2;2,1;s]\!]) = l(w-s),$$

and all other configurations do not contribute to W. Now, the coefficient of $N([\![0,2;0,2;s]\!])$ in W is st_1, the coefficient of $N([\![2,0;2,0;s]\!])$ in W is $t_2(6w - s + (2w-s)^2)$, and the coefficient of $N([\![2,1;2,2;s]\!])$ and $N([\![2,2;2,1;s]\!])$ in W is $t_2(s+2w)$. Combining this, we obtain

$$W(s) = \ell s t_1 + \ell t_2(6w - s + (2w-s)^2) + 2\ell(w-s)(s+2w)t_2 = \ell s t_1 + \ell t_2(6w + 8w^2 - s(s+6w+1))$$

as claimed. □

We conclude this section with the proof of Fact 7.

Fact 7. *For $n \geqslant 1$, let $H_n = \sum_{k=1}^{n} \frac{1}{k}$. Then, $f(w) = 1 - 2(H_{2w} - H_w)$.*

Proof. Recall that $f(w) \triangleq -\frac{2w}{w+1} + \frac{(w+5)}{(w+1)(w+2)} + \sum_{s=1}^{w-1} st_1 + t_2(6w + 8w^2 - s(s+6w+1))$. Let us rewrite $f(w)$ as

$$\begin{aligned} f(w) &= \frac{-2w(w+2) + w + 5}{(w+1)(w+2)} + \sum_{s=1}^{w-1} s(2w - s + 2)t_2 + t_2(6w + 8w^2 - s(s+6w+1)) \\ &= \frac{-2w^2 - 3w + 5}{(w+1)(w+2)} + \sum_{s=1}^{w-1} t_2(-s^2 + s(2w+2) + 6w + 8w^2 - s(s+6w+1)) \end{aligned}$$

$$= \frac{-2w^2 - 3w + 5}{(w+1)(w+2)} + \sum_{s=1}^{w-1} t_2(-2s^2 + s(-4w+1) + 6w + 8w^2)$$
$$= \frac{-2w^2 - 3w + 5}{(w+1)(w+2)} - 2S_4 + (-4w+1)S_2 + (6w+8w^2)S_1,$$

where $S_4 = \sum_{s=1}^{w-1} t_2 s^2$, $S_2 = \sum_{s=1}^{w-1} t_2 s$, and $S_1 = \sum_{s=1}^{w-1} t_2$. Let

$$T = -2S_4 + (-4w+1)S_2 + (6w+8w^2)S_1.$$

We will now reduce each of the sums.

$$S_1 = \sum_{s=1}^{w-1} t_2 = \sum_{s=1}^{w-1} \frac{1}{(2w-s)(2w-s+1)(2w-s+2)} = \sum_{i=w+1}^{2w-1} \frac{1}{i(i+1)(i+2)} = \sum_{i=1}^{2w-1} \frac{1}{i(i+1)(i+2)} - \sum_{i=1}^{w} \frac{1}{i(i+1)(i+2)}.$$

We now use the fact that $\sum_{k=1}^{n} \frac{1}{k(k+1)(k+2)} = \frac{n(n+3)}{4(n+1)(n+2)}$, which can be derived via partial fraction decomposition or induction. Then,

$$S_1 = \frac{(2w-1)(2w+2)}{4(2w)(2w+1)} - \frac{w(w+3)}{4(w+1)(w+2)} = \frac{(2w-1)(w+1)}{4w(2w+1)} - \frac{w(w+3)}{4(w+1)(w+2)}.$$

Proceeding similarly for the next component, we have:

$$S_2 = \sum_{s=1}^{w-1} \frac{s}{(2w-s)(2w-s+1)(2w-s+2)} = \sum_{i=w+1}^{2w-1} \frac{2w-i}{i(i+1)(i+2)} = 2wS_1 - \sum_{i=w+1}^{2w-1} \frac{1}{(i+1)(i+2)}.$$

Recalling that $\sum_{k=1}^{n} \frac{1}{(k+1)(k+2)} = \frac{n}{2(n+2)}$, we get

$$S_3 = \sum_{i=w+1}^{2w-1} \frac{1}{(i+1)(i+2)} = \sum_{i=1}^{2w-1} \frac{1}{(i+1)(i+2)} - \sum_{i=1}^{w} \frac{1}{(i+1)(i+2)} = \frac{2w-1}{2(2w+1)} - \frac{w}{2(w+2)}.$$

Hence

$$S_2 = 2wS_1 - S_3 = 2wS_1 - \frac{2w-1}{2(2w+1)} + \frac{w}{2(w+2)}.$$

Finally,

$$S_4 = \sum_{s=1}^{w-1} \frac{s^2}{(2w-s)(2w-s+1)(2w-s+2)} = \sum_{i=w+1}^{2w-1} \frac{(2w-i)^2}{i(i+1)(i+2)}$$
$$= 4w^2 \sum_{i=w+1}^{2w-1} \frac{1}{i(i+1)(i+2)} - 4w \sum_{i=w+1}^{2w-1} \frac{1}{(i+1)(i+2)} + \sum_{i=w+1}^{2w-1} \frac{i}{(i+1)(i+2)}$$
$$= 4w^2 S_1 - 4wS_3 + \sum_{i=w+1}^{2w-1} \frac{i}{(i+1)(i+2)}.$$

Using that $\sum_{k=1}^{n} \frac{k}{(k+1)(k+2)} = H_{n+1} + \frac{2}{n+2} - 2$ again via partial fraction decomposition or induction, we get

$$S_5 = \sum_{i=w+1}^{2w-1} \frac{i}{(i+1)(i+2)} = \sum_{i=1}^{2w-1} \frac{i}{(i+1)(i+2)} - \sum_{i=1}^{w} \frac{i}{(i+1)(i+2)} = H_{2w} - H_{w+1} + \frac{2}{2w+1} - \frac{2}{w+2}$$
$$= H_{2w} - H_w - \frac{1}{w+1} + \frac{2}{2w+1} - \frac{2}{w+2} = H_{2w} - H_w - \frac{3w+4}{(w+1)(w+2)} + \frac{2}{2w+1}.$$

Thus,

$$S_4 = 4w^2 S_1 - 4wS_3 + S_5.$$

Combining all of this, we get

$$T = -2S_4 + (-4w+1)S_2 + (6w+8w^2)S_1$$
$$= -2(4w^2 S_1 - 4wS_3 + S_5) + (-4w+1)(2wS_1 - S_3) + (6w+8w^2)S_1$$

$$= S_1(-8w^2 + 8w) + S_3(12w - 1) - 2S_5.$$

By using partial fraction decomposition, we can algebraically simplify each of the terms as follows:

$$S_1(-8w^2 + 8w) = -8w(w-1)\left(\frac{(2w-1)(w+1)}{4w(2w+1)} - \frac{w(w+3)}{4(w+1)(w+2)}\right) = \frac{24}{w+2} - \frac{3}{2w+1} - \frac{8}{w+1} - 3,$$

$$S_3(12w - 1) = (12w-1)\left(\frac{2w-1}{2(2w+1)} - \frac{w}{2(w+2)}\right) = \frac{7}{2w+1} - \frac{25}{w+2} + 6, -2S_5$$

$$= -2\left(H_{2w} - H_w - \frac{3w+4}{(w+1)(w+2)} + \frac{2}{2w+1}\right) = \frac{4}{w+2} - \frac{4}{2w+1} + \frac{2}{w+1} - 2\left(H_{2w} - H_w\right).$$

By plugging these expressions back into T, we get

$$T = \frac{3}{w+2} - \frac{6}{w+1} + 3 - 2\left(H_{2w} - H_w\right) = \frac{3(w^2+2w-1)}{(w+1)(w+2)} - 2\left(H_{2w} - H_w\right).$$

Now, we plug the value of T into $f(w)$ and it finishes the proof,

$$f(w) = \frac{-2w^2 - 3w + 5}{(w+1)(w+2)} + T = \frac{-2w^2 - 3w + 5}{(w+1)(w+2)} + \frac{3(w^2+2w-1)}{(w+1)(w+2)} - 2\left(H_{2w} - H_w\right) = 1 - 2\left(H_{2w} - H_w\right).$$

□

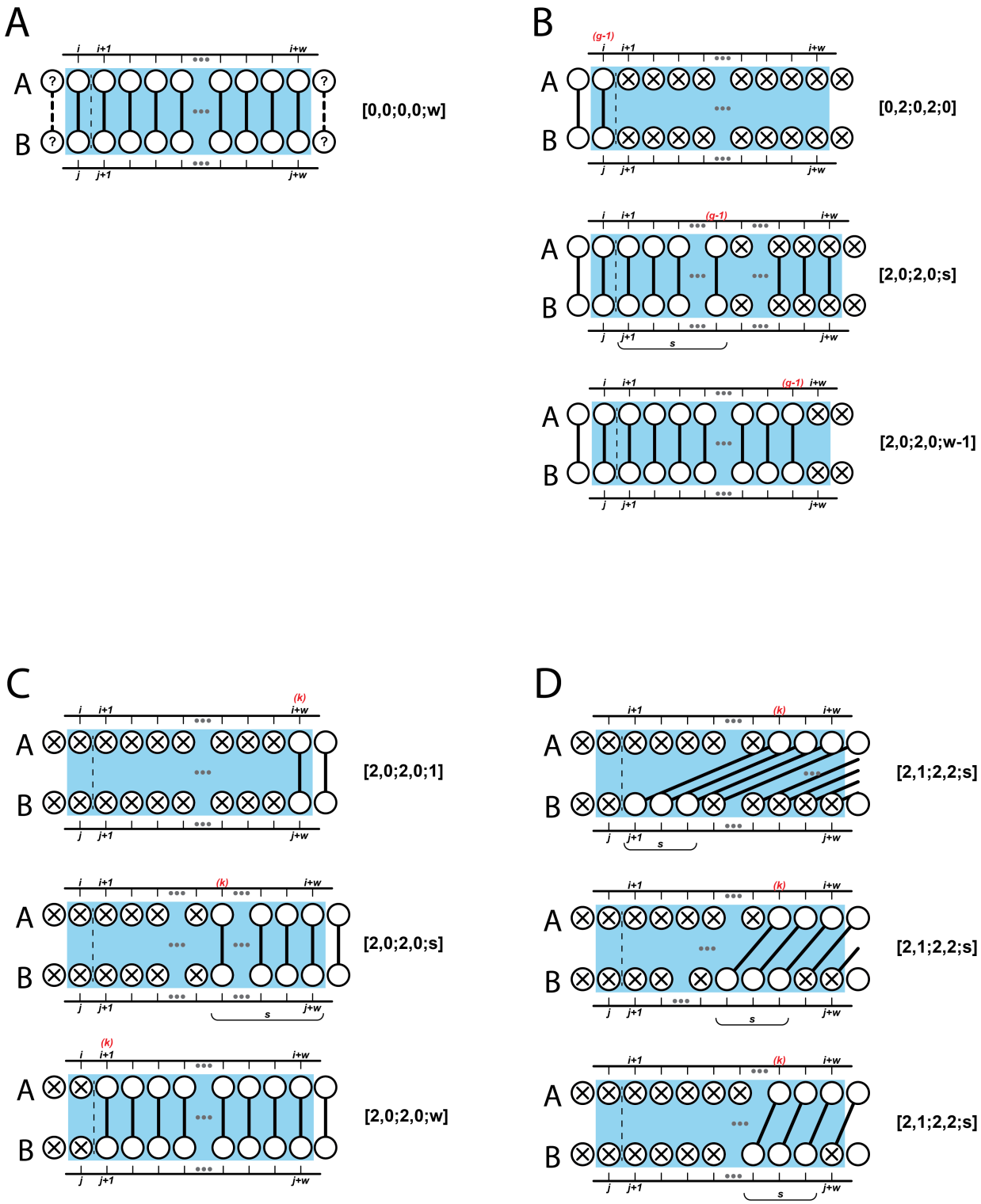


Fig. S2: Some of the configurations with non-zero counts in Fact 8.

A.6 Experimental details

In this section, we provide some experimental details to aid reproducibility. The scripts to reproduce our experiments are available on GitHub [26].

Generative models: When we generate an unrelated pair, we greedily extend each string from left to right. At each position, we choose, uniformly at random, one of the nucleotides that would not result in a k-mer we have already seen. If we get to a point where all the possible nucleotide extensions to a string are already present, we discard the string and start from the beginning. Though this sampling scheme is not guaranteed to terminate, we found that it always did in our experiments. We also verified that the Jaccard of the generated pair was close to the j that was used as a target. Under the assumptions that A and B are uniformly chosen, j should be the expected value under the generative process. Though it is not clear that the uniformity assumption holds in our generative process, we found that the true Jaccard was indeed very close to j in practice. In the related pair model, we also faced a possibility that after choosing to mutate a position, all the possible nucleotide substitutions would create a duplicate k-mer. In such a case, the position was left unchanged.

Mashmap divergence experiment: We sampled 100 substrings from the *E.coli* reference [8], each of length $L = 10,000$ and, for each substring and for each $r_1 \in \{0.90, 0.95, 0.99\}$, generated a "read" which was the substring with $r_1 L$ positions randomly picked and mutated. We then mapped it with mashmap, and discarded any read for which mashmap did not correctly identify a unique and correct mapping location. Mashmap was run with default parameters of $k = 16$ and $w = 200$.

Correction formula to remove Poisson-approximation from Mash distance Let j be the observed Jaccard. Let A and B be two sequences generated using a simple mutation process, i.e. a substitution is created at every nucleotide with a given probability r_1 [2]. The method of moments [38] estimator for the sequence identity is $\widehat{i}_{\text{mom}} = (1-n/L)^{1/k}$, where n is the observed number of mutated k-mers [2]. In the simple mutation model, the observed Jaccard j is related to n via $j = \frac{L-n}{L+n}$, or, equivalently, $n = \frac{L(1-j)}{1+j}$ [2]. Putting this together, we get that $\widehat{i}_{\text{mom}} = (1 - \frac{1-j}{1+j})^{1/k} = \frac{2j}{1+j}^{1/k}$. On the other hand, the Mash distance estimator is $-\frac{1}{k}\log(\frac{2j}{1+j})$ (Formula 1 in [12]), which equivalently translates to the identity estimator $\widehat{i}_{\text{mash}} = 1 + \frac{1}{k}\log(\frac{2j}{1+j})$. Combining the two, we get that $\widehat{i}_{\text{mash}} = 1 + \frac{1}{k}\log((\widehat{i}_{\text{mom}})^k)$. Solving for $\widehat{i}_{\text{mom}}$, we get the final correction formula: $\widehat{i}_{\text{mom}} = e^{\widehat{i}_{\text{mash}}-1}$.

Sliding read experiment: When choosing A, we avoided segments with any Ns or any duplicate k-mers. Any k-mers in B containing an N were hashed to the maximum hash value so as to avoid them being a minimizer. Also note that minimizers were computed separately for each B; thus, it is possible that the same k-mer might be a minimizer in one B but not a minimizer in a nearby B.