Incorporating indel channels into average-case analysis of seed-chain-extend

Spencer Gibson
Department of Biomedical Engineering, Carnegie Mellon University, Pittsburgh, PA, USA

Yun William Yu[1]
Computational Biology Department, Carnegie Mellon University, Pittsburgh, PA, USA

Abstract

Given a sequence s_1 of length n and a mutated substring s_2 of length $m < n$, we often want to recover the mutation history that generated s_2 from s_1. Modern sequence aligners are widely used for this task, and many employ the seed-chain-extend heuristic with k-mer seeds. Previously, Shaw and Yu showed that optimal linear-gap cost chaining can produce a chain with $1 - O\left(\sqrt{\frac{1}{m}}\right)$ recoverability in $O\left(mn^{2.43\theta} \log n\right)$ expected time, where $\theta < 0.206$ is the mutation rate under a substitution-only channel and s_1 is assumed to be uniformly random. However, a gap remains between theory and practice, since real genomic data includes insertions and deletions (indels), and yet seed-chain-extend remains effective. In this paper, we make progress toward closing that gap by proving that the expected recoverability of an optimal chain is $\geq 1 - O\left((\log n)^2 n^{-C\alpha}\right)$, when $0 < \theta_T < 0.206$, $\theta_T = \theta_i + \theta_d + \theta_s$, and $\alpha = -\log_\sigma(1 - \theta_T)$, by introducing new mathematical tools to generalize those prior results under indel channels.

2012 ACM Subject Classification Applied computing → Computational biology; Applied computing → Bioinformatics

Keywords and phrases Sequence alignment, seed-chain-extend, indel channels

Digital Object Identifier 10.4230/LIPIcs.TBD.2025.23

Funding No funding or competing interests are declared.

1 Introduction

String alignment—i.e. determining the best way to match positions of two similar strings s_1 and s_2 under some cost function—has always been one of the central primitives in computational biology, essential for downstream biological analyses like comparing relatedness of genomes or mapping sequenced reads [4, 14, 19, 24]. It is closely related to the "edit distance" and longest common substring (LCS) problems [3, 7, 13, 21, 27], as the choice of matching positions in the alignment implies a series of insertions, deletions, and substitutions that would be needed to transform s_2 into s_1 or vice versa. To this end, Needleman-Wunsch [18] and Smith-Waterman [25] gave dynamic programming exact solutions to the global and local alignment problems in quadratic $O(mn)$ time, where $|s_1| = n$ and $|s_2| = m$. Unfortunately, it turns out that in the worst case, this is the best we can do—Backurs and Indyk showed in 2015 that "edit distance cannot be computed in strongly subquadratic time (unless SETH is false)" [2].

Of course, that's not the end of the story. Although in the earliest days of bioinformatics, we had the luxury of algorithms with mathematically provable guarantees on accuracy and speed, the rapid growth of biological data necessitated the development of faster heuristics that do not come with those strong guarantees. BLAST [1], one of the most highly cited papers of all time [28], gives a linear-time heuristic for local alignment, at the cost of

[1] Corresponding author

optimality, and is still to this day one of the primary workhorses of bioinformatics, whereas Smith-Waterman has been relegated to being just a subroutine within heuristic software. More broadly, there are many heuristics designed to optimize sequence alignment for specific tradeoffs [6, 10, 11, 15].

One heuristic of particular interest is "seed-chain-extend," which is used in modern software such as Minimap 2 [17]. The seed-chain-extend heuristic has three stages. In *seeding* k-mer 'seeds' are selected on both s_1 and s_2, and shared k-mers are marked as 'anchors' between the two strings. Afterwards, concordant anchors are *chained* together to form the skeleton of an alignment. Lastly, the space between anchors is filled in using standard quadratic-time dynamic programming in a process known as *extension*. Seed-chain-extend empirically showcases near quasilinear runtime on the similar genomic strings it is typically applied to, but is not guaranteed to find optimal alignments.

For a long time, bioinformaticians have contented ourselves to this gap between theory and practice. In the last five years, though, theoreticians have made several new breakthroughs by defining a generative model of string evolution and revisiting average-case analysis. Analysis of string algorithms [5, 16], particularly average-case analysis [26], historically made extensive use of generating functions; however, more recent approaches instead used tail-bounds to bound bad events, more akin to the analysis of some randomized probabilistic sketches [30]. Ganesh and Sy's 2020 breakthrough was to show that under a random mutation model with constant low mutation rate, a modified dynamic programming algorithm will with high probability compute edit distance in $O(n \log n)$ time between a random string s_1 and a mutated string s_2 of near equal length [9]. However, part of what made their analysis work was the concordance of their DP algorithm with their mutation model, which meant that extending their results to more practical but sophisticated heuristics like seed-chain-extend was nontrivial.

To that end, last year, our research group made substantial progress on proving similar results for seed-chain-extend [23]. Unfortunately, the machinery and techniques we introduced in that paper were insufficiently powerful to address insertions and deletions, so we had to restrict our theoretical results to a substitution-only mutation model. Indels create all kinds of weird dependencies, so most bioinformatics theoreticians either avoid directly working with them [8, 20, 22, 29], or adjust their algorithm and model to directly capture them [9]. We were able to run empirical benchmarks with indels that closely tracked our substitution-only theory (including accurate predictions of exponents), so we believed without proof the theorems were also morally correct for indels [23]. In this sequel, we make progress toward narrowing that remaining gap between theory and practice, generalizing our prior machinery and techniques to handle indels.

2 Strategy Overview

2.1 Challenges to analysis of seed-chain-extend

There are several difficulties in proving average-case results for seed-chain-extend. First, seed-chain-extend has three stages, and chaining and extension have different optimization objectives. When considering the overall performance of the heuristic, it is in theory possible for failures to happen at any stage. By failure here, we mean anything that leads to bad downstream events, which include most prominently not finding the correct string alignment or taking too long (e.g. quadratic time) to find that alignment. A bad chain can result from a failure in chaining, or just a bad selection of anchors in seeding. Similarly, extension failure can be a result of bad chaining, or because the extension procedure does not itself

find the right alignment, despite the chaining being "good". Here we should also note that the alignment problem under a mutation model actually diverges somewhat from the edit distance problem. The best scoring alignment corresponds to some edit distance between the strings, but arguably the "correct" alignment (at least from a biological perspective) is the one that reflects the mutations that happened to transform s_1 into s_2. Furthermore, seed-chain-extend is known to be an approximate heuristic, and it does not guarantee a correct alignment (under either definition of correct).

To resolve these difficulties, in the prequel, we introduced the concept of "recoverability", which decoupled chaining accuracy from extension accuracy. Extension is only performed in the gaps between anchors on the chain, so roughly speaking, recoverability measures how much of the correct alignment can possibly be recovered given a chain. By structuring our problem thus, we can focus on just the seeding and chaining—i.e. how good is the chain as a starting point for the extension phase. Importantly, recoverability of a chain is a theoretical measure of the goodness of the chain, as opposed to the optimization criterion used to generate the chain, such as linear-gap cost chaining.

2.2 Difficulty from indels

Unfortunately, our formal definition of recoverability in the prequel relied on the substitution-only error model. In the substitution-only regime, the correct alignment of a mutated substring s_2 to s_1 is always a diagonal line in the alignment matrix (which we termed the "homologous diagonal"). Thus, recoverability could be defined as the proportion of the homologous diagonal covered by anchors on the optimal chain and the dynamic programming (DP) extension blocks between anchors. In the presence of indels though, the correct alignment is no longer a straight diagonal. In this sequel, we thus must generalize the homologous diagonal to a "homologous path".

In the same vein, indels also mess up the matching of indices between s_1 and s_2. This is not only notationally very tricky to reconcile, but also make it hard to discuss the dependence structure of positions in anchors, which is necessary for the limited-dependence Chernoff bounds we used in the prequel.

Finally, having several different types of mutations raises questions of ordering and reversal that do not appear with only substitutions. A deletion or substitution may happen to a letter not originally in s_1, but that was inserted in a prior mutation step. Alternately, an insertion of the appropriate letter could reverse a prior insertion. This however creates a problem with defining recoverability of a homologous path. If a letter is inserted and then deleted, or vice versa, then the alignment specified by the homologous path will have a spurious kink that cannot be found via k-mer matching. For example, if $s_1 =$ `ACGT`, and it is mutated to $s_2 =$ `ACGT` by inserting another `G` before the original `G`, and then deleting the original `G`, then the "correct" alignment is

```
AC-GT
||  |
ACG-T
```

which naturally will not be found by any reasonable alignment method. This is not a problem for recoverability in extension regions, as it still could be an alignment produced by the extension block (given a very weird extension algorithm), but it cannot be found as an anchor, which would produce instead the "incorrect" (but lowest edit distance) alignment

```
ACGT
```

```
||||
ACGT
```

Still, in the interest of theoretical consistency, we will apply a recoverability penalty in the latter case, despite it not actually being a problem in practice, because the lower-edit-distance alignment that seed-chain-extend will find does not reflect the actual no-op mutation history.

2.3 Proof structure and motivation

The basic intuition behind the prequel [23] is that given reasonably low mutation rates, the optimal chain under linear-gap-cost chaining will be close to correct in the sense that most of the anchors will lie on the homologous diagonal. A gap between anchors can either be homologous if the anchors flanking it are both on the homologous diagonal, or non-homologous if at least one of the two anchors is off the homologous diagonal. Non-homologous gaps can lead to "breaks", which are regions of the string where extension through the gaps does not cover the homologous diagonal, leading to a decrease in recoverability. However, with high probability, all of the breaks have size $< \sqrt{m}$, so the recoverability will be high. Additionally, the runtime can be bounded by extension time through all the homologous and non-homologous gaps, which are small in an optimal chain.

Roughly speaking, the moral reason the above strategy works is that substitutions are much more likely to break anchors than they are to create spurious anchors. So long as we remain in the regime where sufficiently many anchors can still be found, the optimal chain is close to the correct chain, and the recoverability will be high. When indel channels are added to the mix, the above logic still basically holds (though we do suddenly have pseudo-spurious anchors from indel reversals we have to deal with), but we have to carefully redefine the model and generalize recoverability. Furthermore, because of the redefinition of the model, many of the theorems and lemmas from [23] need to be updated and proved from scratch. We detail the full proof for each theorem/lemma which requires new techniques and omit the proof only when it is exactly analogous to the corresponding proof in the prequel.

The major difference we handle is a new class of anchors, defined below, termed "clipping" anchors, which lie partially on the homologous path. Clipping anchors are a strange phenomenon of indels. They contribute to recoverability in a similar way as homologous anchors: extending through gaps flanked by clipping anchors still contains the bulk majority of the path in the gap. However, they behave like spurious anchors in that the anchors themselves may not recover points on the homologous path. Clipping anchors may seem to be an anomaly but, in fact, they are the most likely anchor type with indels. This is easily seen by calculating the probability that the path arising from a k-mer in the generative region has a single 'insert'-'delete' and otherwise contains no mutations. This scenario occurs with probability $\approx (1-\theta_T)^k$ and there are k such orientations – already pushing $\mathbb{E}(N_C) \geq \mathbb{E}(N_H)$ by linearity of expectation. Thus, clipping anchors must be dealt with carefully.

By generalizing all the theorems in the prequel and showing that the number of missed points in regions of clipping anchors is negligible, we conclude that the expected recoverability of an optimal chain is $\geq 1 - O\big((\log n)^2\, n^{-C\alpha}\big)$.

2.4 Preliminaries

Our work assumes a mutation model inspired by the work of Ganesh and Sy [9]. The mutation channel is the following: Let $S = x_1 x_2 \cdots x_{n+k-1}$ be a string where each letter x_i is sampled i.i.d. from an alphabet of size σ. The substring $S[p+1 : p+m'+k-1]$ is passed

through an indel channel acting independently at each position. For each $j \geq 1$, the input symbol $S[p+j-1]$ undergoes exactly one of:

- **Substitution** (probability θ_s): replaced by a different letter, chosen uniformly from the other $\sigma - 1$ symbols.
- **Deletion** (probability θ_d): the position is deleted
- **Insertion** (probability θ_i): a random string of length $L \sim \text{Geom}(1-\rho_i')$ is inserted immediately to its right.

Unlike the substitution-only case in [23] where the optimal alignment is the diagonal, i.e., the alignment matching S' to the generative substring of S, the optimal alignment is more complicated under indels. It is now a homologous *path*, which we define following Ganesh and Sy's notion of canonical alignment [9]. Intuitively, the homologous path follows the sequence of edits made to the generative substring of S in the alignment matrix of S and S'.

▶ **Definition 1** (Inspired by Ganesh and Sy). *Let $(S, S', \mathcal{E}) \sim M(n, a, b)$ be the uniformly random string S of length n, the mutated substring S' from $\mathcal{G}(S) = S[p+1 : p+m'+k-1]$, and the sequence of edits $\mathcal{E}$, sampled from the mutation process $M(n, a, b)$ where $a = p+1$ and $b = p+m'+1$. The homologous path between S and S' begins at $(a, 0)$, and at each reached position (i, j), we extend the path according to $\mathcal{E}$ as follows:*

- *If no insertion or deletion occurs at bit i, i.e., a substitution occurs or there is no mutation, include the points*

$$\{(i, j), (i+1, j+1)\}$$

- *If an insertion of I bits occurs at bit i and no deletion occurs, include the points*

$$\{(i, j), (i, j+1), \ldots, (i, j+I), (i+1, j+I+1)\}$$

- *If a deletion (and no insertion) occurs at bit i, include the points*

$$\{(i, j), (i+1, j)\}$$

- *If both an insertion of I bits and a deletion occur at bit i, include the points*

$$\{(i, j), (i, j+1), \ldots, (i, j+I), (i+1, j+I)\}$$

Throughout this work, we will assume that $\theta_T < 0.206$, $m' = |\mathcal{G}(S)| = \Omega(n^{2C\alpha+\epsilon}) < n$ since it is always possible to choose $C\alpha < \frac{1}{2}$, where $\alpha = -\log_\sigma(1-\theta_T)$ and $C > \frac{2}{1-2\alpha}$. The length of the seeds, k, is given by $k = C\log_\sigma(n)$. We will write $\log = \log_\sigma$ for short. Note that by our choice of variables, $\sigma^k = n^C$ and $(1-\theta_T)^k = n^{-C\alpha}$, which we will use repeatedly.

2.5 Tools

In this section, we go through definitions of tools and terms that simplify our analysis. A central concept of seed-chain-extend is an anchor, which we define formally below. An anchor is an exact k-mer match between S and S'.

▶ **Definition 2.** *An anchor occurs at position (i, j) if $S[p+i : p+i+k-1] = S'[p+j : p+j+k-1]$.*

We will also find it useful to define the random variable indicating if letters at a position in S and S' match, as well as the random variable indicating if there is an anchor starting at specified positions in S and S'.

▶ **Definition 3.** *Let $M(i,j)$ be the random variable defined by*

$$M(i,j) \;=\; \begin{cases} 1, & x_i = y_j, \\ 0, & \textit{otherwise.} \end{cases}$$

Define

$$A(i,j) \;=\; \prod_{\ell=0}^{k-1} M(i+\ell,\; j+\ell)$$

Then, $A(i,j)$ is the indicator random variable for the presence of an anchor at positions (i,j), and $M(i,j)$ denotes a match at positions i and j in S and S', respectively.

There are three types of anchors relevant to our analysis: homologous, clipping, and spurious anchors. Visually, anchors always appear as diagonals of length k in the alignment matrix. A homologous anchor lies entirely on the homologous path, which implies the homologous path contains no mutations at all throughout that anchor. Unlike the substitution-only case, under indel channels, anchors can touch the homologous path at a number of points without lying entirely on it– these are called clipping anchors. Spurious anchors remain the same as before: anchors that lie entirely off the homologous path. We now formally define each anchor type.

▶ **Definition 4.** *An anchor starting at position (i,j) is called homologous if $\{(i,j),\ldots,(i+k-1,j+k-1)\} = P_H[(i,j):(i+k-1,j+k-1)]$, spurious if $\{(i,j),\ldots,(i+k-1,j+k-1)\} \cap P_H = \emptyset$, and clipping otherwise.*

We'll now define the notion of a break, which we use throughout the paper to bound recoverability. Intuitively, a break is a maximal region of the chain containing only spurious anchors. Setting the recoverability of these regions to zero allows us to lower bound the recoverability of the entire chain. Later, we'll show that the total break length is short compared to m, so their impact can be ignored.

▶ **Definition 5.** *For a chain $((i_1,j_1),\ldots,(i_u,j_u))$, a break B is a maximal interval of spurious anchors $((i_p,j_p),\ldots,((i_q,j_q))$. The break length, $L(B)$ is the total number of points on the homologous path that belong to the break. Formally,*

$$L(B) = \begin{cases} \big|P_H\big[(i_a,j_a):(i_b,j_b)\big]\big|, & \textit{if } B \textit{ is flanked by anchors } (i_a,j_a) \textit{ and } (i_b,j_b), \\ \big|P_H\big[:(i_b,j_b)\big]\big|, & \textit{if } B \textit{ is only flanked on the right by } (i_b,j_b), \\ \big|P_H\big[(i_a,j_a):\big]\big|, & \textit{if } B \textit{ is only flanked on the left by } (i_a,j_a). \end{cases}$$

We borrow the Python convention of omitting the start and end of a list when splicing, i.e., $P_H[:(p_1,p_2)]$ refers to the portion of the homologous path between the start of P_H up to the point $(p_1,p_2) \in P_H$. The notation $P_H[(p_1,p_2):]$ is defined to be the portion of P_H starting at $(p_1,p_2) \in P_H$ to the end of the homologous path.

For a position $i \in \mathcal{G}(S)$, if i is not deleted when generating S', then there is a unique position $j_i \in S'$ whose letter distribution is dependent on i. For any other letter, $j \neq j_i$, we will repeatedly use the fact that $S[i]$ and $S'[j]$ are independent. It will be convenient to define a function f that maps each $i \in S$ to its unique corresponding position if one exists.

▶ **Definition 6.** *Define the function* $f : \{1, \ldots, |S|\} \longrightarrow \{1, \ldots, |S'|\} \cup \{\text{null}\}$ *such that*

$$f(i) = \begin{cases} \text{null}, & \text{if } i \notin \mathcal{G}(S) \text{ or } i \text{ is deleted,} \\ \min\{\, j : (i,j) \in P_H \,\}, & \text{otherwise.} \end{cases}$$

If $i \in \mathcal{G}(S)$ *and not deleted, then either no mutation occurs to* $S[i]$*, or it is substituted. In either case,* $\exists (i,j) \in P_H$*, so the minimum exists. Thus,* f *is well-defined.*

3 Methods and Bounds

3.1 Recoverability setup

The recoverability of a chain, $C = ((i_1, j_1), \ldots, (i_u, j_u))$, loosely speaking, is the fraction of the homologous path P_H, that lies on the anchors in the chain and in gap extensions. We define recoverability to include all *possible* alignments given by the chain, which means that we count any portion of the homologous path in a gap extension as 'recovered' since, in theory, it could be recovered by some extension algorithm. We formalize this intuition below.

▶ **Definition 7** ((Yu and Shaw) Recoverability)**.** *Given a chain* $C = ((i_1, j_1), \ldots, (i_u, j_u))$*, we define the set of all possible alignments,* $Align(C)$*, as:*

$$Align(C) = (\bigcup_{\ell=1}^{u} \{(i_\ell, j_\ell), \ldots, (i_\ell + k - 1, j_\ell + k - 1)\}) \;\cup\; (\bigcup_{\ell=1}^{u-1} Ext(\ell))$$

where $Ext(\ell) = [i_\ell + k, \ldots, i_{\ell+1} - 1] \times [j_\ell + k, \ldots, j_{\ell+1} - 1]$*. If* $i_\ell + k > i_{\ell+1} - 1$ *or* $j_\ell + k > j_{\ell+1} - 1$*, then* $Ext(\ell) = \emptyset$*.*

The recoverability of the chain, $R(C)$*, is defined to be:*

$$R(C) = \frac{|Align(C) \cap P_H|}{|P_H|}$$

3.2 Independence lemmas and the match graph

The match graph, defined below, captures the dependence structure between positions on S and S'. Visually, it is a bipartite graph with edges between positions in S and S'. An edge occurs between $i \in S$ and $j \in S'$ whenever j corresponds directly to i – formally, when $f(i) = j$ – or when (i, j) appears in a match variable $M(i_l j_l)$ being conditioned upon. We use the match graph to determine when variables are conditionally independent of others variables, which we use in the next section to bound the variance of the number of spurious anchors, N_S. As we will show, the independence lemma under indels requires similar but more general conditions than the substitution-only case. This manifests by scaling up the variance of N_S by a constant factor, which makes no asymptotic difference.

▶ **Definition 8.** *Let* $\mathcal{M} = \{\, M(i_1, j_1),\, M(i_2, j_2),\, \ldots \}$ *be a set of such matching variables. The match graph* $G(\mathcal{M}) = (V, E)$ *is defined as follows:*

$$V = \{x_1, \ldots, x_{n+k-1}\} \cup \{y_{p+1}, \ldots, y_{p+m+k-1}\}, \text{ and}$$

$$E = \big\{(x_i, f(x_i)) \mid i \in [p+1\,..\,p+m'+k-1] \text{ if } f(x_i) \neq null\big\} \cup \big\{(x_h, y_\ell) \mid M(h,\ell) \in \mathcal{M}\big\}.$$

Note $G(\mathcal{M})$ *is a bipartite graph with parts* $\{x_1, \ldots, x_{n+k-1}\}$ *and* $\{y_{p+1}, \ldots, y_{p+m+k-1}\}$*.*

▶ **Lemma 9.** *Consider a set of random variables of the form* $\mathcal{M} = \{ M(i_1, j_1), \dots M(i_\ell, j_\ell) \}$. *If two vertices* x, y *lie in separate connected components of* $G(\mathcal{M})$, *then they are conditionally independent of* $\mathcal{M}$.

In the substitution-only case, a sufficient condition for independence is for the match variables to be spurious and the graph to be acyclic. The condition is equivalent under indels, except spurious means that the position of the match variable in S does not map to the position of the match variable in S', i.e., $M(i, j)$ where $f(i) \neq j$, as detailed below.

▶ **Theorem 10.** *The random variables*

$$\mathcal{M} = \{ M(i_1, j_1),\ M(i_2, j_2),\ \dots \},$$

where $f(i_\ell) \neq j_\ell$ *for all* $M(i_\ell, j_\ell) \in \mathcal{M}$, *are independent if the induced match graph has no cycles.*

The above theorem implies that spurious anchors, (i, j) for which $f(x_l) \neq y_l, \forall (x_l, y_l) \in \{(i, j), \dots, (i + k - 1, j + k - 1)\}$ occur with probability $\frac{1}{\sigma^k}$, i.e., the probability that two random strings of length k drawn from an alphabet of size σ.

▶ **Corollary 11.** *If* $(i, j) \in S_p$, *then* $\Pr\big(A(i, j) = 1\big) = \frac{1}{\sigma^k}$.

Proof. Since $(i, j) \in S_p$, each $M(x, y) \in M = \{M(i, j), \dots, M(i+k-1, j+k-1\}$ is spurious. Corollary 1 in Yu and Shaw show that the induced match graph has no cycles. Applying Theorem 10, the M variables are independent, so $\Pr(A(i, j) = 1) = \prod_{l=0}^{k-1} \Pr(M(i+l, j+1) = 1) = \frac{1}{\sigma^k}$. ◀

The lemmas provided below establish the conditions for anchors to be independent. Intuitively, this occurs when they do not overlap too much on both S and S', and there is no 'twisting' that forms cycles. We provide the substitution-only independence lemma from Yu and Shaw [23], so that the reader can compare it with the indel analogue.

▶ **Lemma 12** ((Yu and Shaw) Independence lemma under substitutions). *For* $A(i, j)$ *and* $A(h, \ell)$, *if both of the following conditions hold:*
1. $|i - h| \geq k$ *or* $|j - \ell| \geq k$, *and*
2. $|i - \ell| \geq k$ *or* $|j - h| \geq k$,

then the induced match graph on the M *variables for* $A(i, j)$ *and* $A(h, \ell)$ *has no cycles.*

Below, we provide the general independence lemma that is valid under indel channels. The differences arise from the more complex way anchors can overlap under indels. In the substitution-only case, we required, e.g., $|i - l| \geq k$ (or $|j - h| \geq k$), which meant prevented cycles in awkward induced match graphs. With indels, this condition must be generalized to explicitly rule out any 'twisting' that can lead to cycles. This change captures the intuition that it is not sufficient to merely ensure a distance of length k between the start positions of one anchor on S and another on S' since distance regions on the two strings can be related through expansions and contractions. Although the general lemma is more notationally complex, the match graph in the indel case is not any more complicated because inserted and deleted positions do not have neighbors.

▶ **Lemma 13** (General independence lemma). *For* $A(i, j)$ *and* $A(h, l)$, *if both of the following conditions hold:*
1. $|i - h| \geq k$ *or* $|j - l| \geq k$, *and*
2. $[i : i + k - 1] \cap f^{-1}([l : l + k - 1]) = \emptyset$ *or* $[h : h + k - 1] \cap f^{-1}([j : j + k - 1]) = \emptyset$,

then the induced match graph on the M variables for $A(i,j)$ and $A(h,l)$ has no cycles.

The proof formalizes the intuition that the match graph in the indel case is simpler than in the substitution-only case since any deletion removes an edge and insertions do not add edges.

Proof. We will show that inserted positions in S' and deleted positions in S have degree at most 1, so they cannot be in any cycle. Removing all such positions from the match graph for our cycle analysis will result in the same conditional match graph (except with different k, which is unimportant) as Lemma 2 from Yu and Shaw [23]. Invoking their lemma will finish our proof.

First, note that in the unconditioned match graph, any deleted position in S has no neighbor in S', and any inserted position in S' has no neighbor in S. This follows from the fact that deleted positions have no dependents and inserted positions take values at random. There are eight total cases to consider, of which we detail one, since the remaining seven are entirely symmetric.

Let $|i-h| \geq k$ and $[i : i+k-1] \cap f^{-1}([l : l+k-1]) = \emptyset$. Take some $x \in \{i, \ldots, i+k-1\}$ that is deleted in the generation process. Since x is deleted, $x \notin f^{-1}([l : l+k-1])$. The spacing of the anchors on S is at least k apart, so $x \notin \{h, \ldots, h+k-1\}$, meaning it can have degree at most 1. Thus, it cannot be in a cycle. The same argument, applied verbatim, shows that any deleted position in $[h : h+k-1]$ cannot be in a cycle. Consider some $x \notin \{i, \ldots, i+k-1\} \cup \{h, \ldots, h+k-1\}$. This position has no additional edge from the anchors $A(i,j), A(h,l)$, and thus has degree at most 1. Thus, any position in S deleted during generation can be entirely removed from the match graph for cycle analysis.

Take some $y \in \{j, \ldots, j+k-1\}$ corresponding to an inserted position. It has no neighbors in the unconditioned match graph and so its degree can only be due to edges added by the anchors. Since $|i-h| \geq k$, there cannot be an edge between y and any $x \in \{h, \ldots, h+k-1\}$, so it has degree 1. Similarly, any inserted $y \in \{l, \ldots, l+k-1\}$ has degree at most 1. For any inserted $y \notin \{j, \ldots, j+k-1\} \cup \{l, \ldots, l+k-1\}$, it does not have any added edge from the anchors and can have degree at most 1. Thus, any inserted position in S' has degree at most 1 and cannot be in a cycle.

Thus, if a cycle exists, it must still exist if we remove all inserted positions in S' and deleted positions in S, which yields, up to extra positions with degree 1 which can also be removed, the same match graph as in the substitution-only case. The remainder of the proof is exactly the same. ◀

3.3 Anchor-count concentration bounds

The following lemma shows that any k-mer in $\mathcal{G}(S)$ cannot expand by more than ck when generating S' with high probability. Our approach will be to show that the probability of an arbitrary k-block expanding too much is very low, and then we will take a union bound over all k-blocks in $\mathcal{G}(S)$ to upper bound the global failure rate by $1/n$. We will find it useful to write k in terms of $\ln(n)$ in MGF/Chernoff style concentration inequalities. To that end, since $k = C\log_\sigma(n) = C\ln(n)\log_\sigma(e)$, let $\beta = \log_\sigma(e)$ so that $k = C\beta\ln(n)$. The constant β does not affect the analysis but allows us to be precise.

▶ **Lemma 14 (Bounded expansion (E)).** *With probability $\geq 1 - 1/n$, no k-mer in $\mathcal{G}(S)$ expands by more than $O(k)$, specifically, $\frac{1}{t_0}(\frac{2}{\beta}+1)k$, for a small fixed constant $t_0 > 0$.*

Proof. Without loss of generality, let the first k-block in $\mathcal{G}(S)$ start at position 1. Denote the random variable representing the total insertion length after the j-th coordinate as I_j

where

$$I_j = \begin{cases} 0, & \text{with probability } 1-\theta_i, \\ \text{Geom}(1-\rho_i'), & \text{with probability } \theta_i, \end{cases}$$

In particular, for $\ell > 0$, $\Pr(I_j = \ell) = \theta_i(1-\rho_i')(\rho_i')^{\ell-1}$. Define $Z_1 = \sum_{j=1}^{k} I_j$, which represents the total insertion length, the expansion, of the first k-block. A simple Chernoff bound shows that for any $t > 0$,

$$\Pr(Z_1 \geq c) \leq \frac{\mathbb{E}[e^{tZ_1}]}{e^{tc}} = \frac{\prod_{j=1}^{k}\mathbb{E}[e^{tI_j}]}{e^{tc}} = \frac{\mathbb{E}[e^{tI_1}]^k}{e^{tc}}$$

Where the first equality follows since the $\{I_j\}_{j=1}^{k}$ are i.i.d. The moment generating function of I_1 can be calculated directly:
$M_{I_1}(t) = \mathbb{E}[e^{tI_1}] = (1-\theta_i)+\sum_{j=1}^{\infty}\theta_i(1-\rho_i')(\rho_i')^{j-1}e^{tj} = (1-\theta_i)+\theta_i(1-\rho_i')e^t\sum_{j'=0}^{\infty}(\rho_i'e^t)^{j'}$. The last term is a geometric series which converges whenever $\rho_i'e^t < 1$. For an admissible t, we get: $M_{I_1}(t) = 1-\theta_i+\theta_i\frac{(1-\rho_i')e^t}{1-\rho_i'e^t}$. Since $\rho_i' \in (0,1)$, we can always pick $0 < t < -\ln(\rho_i')$ making $\rho_i'e^t < 1$. For such t, note that as $t \to 0$, $M_{I_1}(t) = 1+\theta_i(\frac{(1-\rho_i')e^t}{1-\rho_i'e^t}-1) \to 1+\theta_i(\frac{(1-\rho_i')}{1-\rho_i'}-1) = 1$. Thus, there exists some small, valid constant $t_0 > 0$ for which $M_{I_1}(t) < e$, and thus, $\Pr(Z \geq c) \leq e^{k-t_0c}$. Choosing $c = \frac{1}{t_0}(\frac{2}{\beta}+1)$ gives $\Pr(Z_1 \geq \frac{1}{t_0}(\frac{2}{\beta}+1)k) \leq e^{-(t_0\frac{1}{t_0}(\frac{2}{\beta}+1))C\beta\ln(n)} \leq e^{-2\ln(n)} = \frac{1}{n^2}$ since $C > 1$.

Define Z_i to be the random variable denoting the expansion of the i-th block in $\mathcal{G}(S)$. A simple union bound shows that $\Pr(\exists j : Z_j \geq \frac{1}{t_0}(\frac{2}{\beta}+1)k) \leq (m-k+1)\frac{1}{n^2} \leq n\frac{1}{n^2} = \frac{1}{n}$, and the result follows.

◀

We now show a similar lemma but for contraction: namely, any ℓ-block in $\mathcal{G}(S)$ has length $O(k)$ after deletions, where $\ell = \frac{20k}{\beta}$.

▶ **Lemma 15 (Bounded contraction (C)).** *With probability $\geq 1-1/n$, no ℓ-block in $\mathcal{G}(S)$ contracts to size $\leq \frac{(1-\theta_d)\ell}{2}$, where $\ell = \frac{21k}{\beta}$.*

Proof. The proof follows similarly to the previous lemma. Define

$$X_j = \begin{cases} 0, & \text{if the } j\text{-th index of } \mathcal{G}(S) \text{ is deleted (with probability } \theta_d), \\ 1, & \text{otherwise.} \end{cases}$$

Note if $X_j = 1$, then the j-th index of $\mathcal{G}(S)$ survives. Let $X = \sum_{j=1}^{\ell} X_j$ be the total number of surviving indices in the ℓ-block and set $q_d = 1-\theta_d$ to be the survival rate of an index. A classic Chernoff bound on the sum of i.i.d. Bernoulli random variables gives for any $0 < \delta < 1$: $\Pr(X \leq (1-\delta)q_d\ell) \leq \exp(-\frac{\delta^2 q_d\ell}{2})$. Since $\ell = ck = c\beta C\ln(n)$ where $c = \frac{21}{\beta}$ gives $\frac{\delta^2}{2}q_dc_0C\beta \geq \frac{1}{8}(.794)21 > 2$, when $\delta = 1/2$ and $\theta_d \leq \theta_T \leq 0.206$. Thus, $\Pr(X \leq \frac{1}{2}q_d\ell) \leq \exp(-2\ln(n)) = \frac{1}{n^2}$.

A union bound over all ℓ-blocks in $\mathcal{G}(S)$ gives:

$$\Pr(\exists \ell\text{-block shrunk to less than } \frac{(1-\theta_d)\ell}{2}) \leq n/n^2 = \frac{1}{n}$$

◀

We will refer to the space where the bounded contraction and expansion lemmas are jointly satisfied as **EC**.

▶ **Lemma 16** ((EC)). $\mathbb{E}(N_S^2) \leq \mathbb{E}(N_S)^2 + 2T_0k^2\frac{mn}{\sigma^k}$ *where* $T_0 = \max(\frac{1}{t_0}(\frac{2}{\beta}+1)\frac{2}{1-\theta_d}\frac{21}{\beta}, 4)$. *Thus,* $\text{var}(N_S) \leq T_0k^2\frac{mn}{\sigma^k}$

Proof. Let $S_p = \{(i,j) \in S \times S' \mid [(i,j),\ldots,(i+k-1,j+k-1)] \cap P_H = \emptyset\}$, i.e., the points where a starting k-mer at that position does not intersect the homologous path.

Define $B_k(i,j) = \{(h,l) \in S_p \mid |h-i| \leq k \ \cap \ |l-j| \leq k\}$, and

$$P_k(i,j) = \big\{(h,l) \in S_p : f^{-1}\big([l,l+k-1]\big) \cap [i,i+k-1] \neq \emptyset, \\ f^{-1}\big([j,j+k-1]\big) \cap [h,h+k-1] \neq \emptyset\big\}.$$

By Lemma 13, for any $(h,l) \notin B_k(i,j) \ \cup \ P_k(i,j)$, $A(i,j)$ and $A(h,l)$ are independent. Then, $N_S = \sum_{(i,j)\in S_p} A(i,j)$. We calculate the variance as follows:

$$\begin{aligned} N_S^2 &= \sum_{(h,l)\in S_p}\sum_{(i,j)\in S_p} A(i,j)A(h,l) \\ &= \underbrace{\sum_{(h,l)\in S_p\setminus\left(B_k(i,j)\cup P_k(i,j)\right)}\sum_{(i,j)\in S_p} A(i,j)A(h,l)}_{S_1} + \underbrace{\sum_{(h,l)\in S_p\cap B_k(i,j)}\sum_{(i,j)\in S_p} A(i,j)A(h,l)}_{S_2} \\ &\quad + \underbrace{\sum_{(h,l)\in\left(S_p\cap P_k(i,j)\right)\setminus B_k(i,j)}\sum_{(i,j)\in S_p} A(i,j)A(h,l)}_{S_3} \end{aligned}$$

Dealing first with S_1: by the independence lemma, $A(h,l), A(i,j)$ are independent, so

$$\begin{aligned} \mathbb{E}(S_1) &= \sum_{(h,l)\in S_p\setminus\left(B_k(i,j)\cup P_k(i,j)\right)}\sum_{(i,j)\in S_p} \mathbb{E}\big(A(i,j)\big)\,\mathbb{E}\big(A(h,l)\big) \\ &\leq \sum_{(h,l)\in S_p}\sum_{(i,j)\in S_p} \mathbb{E}\big(A(i,j)\big)\,\mathbb{E}\big(A(h,l)\big) = \mathbb{E}(N_S)^2. \end{aligned}$$

Note $\mathbb{E}(A(h,l)A(i,j)) \leq \mathbb{E}(A(i,j))$ since the anchor random variables take values in $\{0,1\}$. Thus,

$$\mathbb{E}(S_2) \leq \sum_{(h,l)\in S_p\cap B_k(i,j)}\sum_{(i,j)\in S_p} \mathbb{E}(A(i,j)) \leq 4k^2\frac{mn}{\sigma^k}$$

Lastly, we handle the S_3 term. Working under EC, $|P_k(i,j)| \leq \frac{1}{t_0}(\frac{2}{\beta}+1)\frac{2}{1-\theta_d}\frac{20}{\beta}k^2 = Tk^2$, since the region of S projecting onto $S'[j,\ldots,j+k-1]$ has size at most $\frac{2}{1-\theta_d}\frac{21}{\beta}k$, and the projection of $S[i,\ldots,i+k-1]$ onto S' has size at most $\frac{1}{t_0}(\frac{2}{\beta}+1)$. For simplicity, let $T_0 = 2\max(\frac{1}{t_0}(\frac{2}{\beta}+1)\frac{2}{1-\theta_d}\frac{21}{\beta}, 4)$. This yields,

$$\mathbb{E}(N_S^2) \leq \mathbb{E}(N_S)^2 + T_0k^2\frac{mn}{\sigma^k}$$

From which it immediately follows that $\text{var}(N_S) \leq T_0k^2\frac{mn}{\sigma^k}$.

◀

The previous lemma shows that allowing indels changes the variance of N_S, the number of spurious anchors, by at most a constant compared to the substitution-only case. The below lemma uses the conditional bounded variance of N_S along with the high likelihood of being in the EC space, to bound the number of spurious anchors w.h.p. The spurious

anchor bound below is exactly the same as in the prequel up to a constant, which makes no asymptotic difference.

▶ **Lemma 17** (F1). *With probability at least* $1 - \frac{3}{n}$*, the number of spurious anchors is*

$$\leq n^{2-C} + \sqrt{T_0\, m}\, C \log(n)\, n^{1-C/2}$$

Mathematically,

$$\Pr\Big(N_S \geq n^{2-C} + \sqrt{T_0\, m}\, C \log(n)\, n^{1-C/2}\Big) \;\leq\; \frac{3}{n}$$

Proof. Call the event $X = \{N_S \geq n^{2-C} + \sqrt{T_0\, m}\, C \log(n)\, n^{1-C/2}\}$, i.e., the event that there are more spurious anchors than the amount given by the expression. By the law of total probability:

$$\begin{aligned}\Pr(X) &= \Pr\big(X \mid EC\big)\ \Pr(EC) + \Pr\big(X \mid EC^c\big)\ \Pr(EC^c)\\ &\leq \Pr\big(X \mid EC\big) + \Pr(EC^c).\end{aligned}$$

The first term, $\Pr(N_S \geq n^{2-C} + \sqrt{T_0\, m}\, C \log(n)\, n^{1-C/2} \mid EC) \leq \frac{1}{n}$, since under EC, the variance of N_S is bounded as we showed, and we can use the same Chebyshev argument as in Lemma 4 of Yu and Shaw [23].

For the second term, $\Pr(EC^c) \leq \frac{2}{n}$, by a simple union bound from Lemma 14 and Lemma 15.

Combining terms, $\Pr\Big(N_S \geq n^{2-C} + \sqrt{T_0\, m}\, C \log(n)\, n^{1-C/2}\Big) \leq \frac{3}{n}$. ◀

We finish this section by upper bounding the expected number of clipping anchors, which allows us to upper bound the expected number of missed points due to clipping anchors in the chain.

▶ **Lemma 18.** *The expected number of clipping anchors,* N_C*, is at most* $O(|P_H| k (1-\theta_T)^k)$

Proof. Consider any point $(i, j) \in P_H$. There are exactly k possible anchors containing that point – anchors that contain it at k different distances from the start of the anchor. Let $N_C(i, j)$ be the number of clipping anchors contributed for the point (i, j). Then $\mathbb{E}(N_C(i, j)) \leq \sum_{l=1}^{k} \Pr(A(i - l + 1, j - l + 1) = 1) \leq k(1 - \theta_T)^k$. Thus, $\mathbb{E}(N_C) = \sum_{(i,j)\in P_H} \mathbb{E}(N_C(i, j)) \leq |P_H| k (1 - \theta_T)^k$. ◀

▶ **Corollary 19** ((EC)). *The expected number of clipping anchors is at most* $O(mk(1-\theta_T)^k)$.

Proof. Under EC, the number of points on the homologous path is at most some constant times more than $|S'| = m$. The result follows. ◀

In the next section, we show that there are no long homologous gaps in the generative region of S, $\mathcal{G}(S)$, and use this fact to bound break lengths with the help of the expansion-contraction lemma.

3.4 Bounding break lengths

A homologous gap is defined to be a region $[a, b]$ in the generative portion of S, $\mathcal{G}(S)$, for which there are no homologous anchors. Equivalently, a homologous gap is a region $\mathcal{G}(S)$ for which every k-mer contains a mutation. We begin by bounding the length of any homologous gap by establishing concentration bounds on homologous anchors in this k-dependence case

similar to the prequel [12, 23]. The lemmas in this substring make use of the generative region of S, $\mathcal{G}(S)$, which has length m' but the final inequalities are in terms of $|S'| = m$. This is fine, since under EC, $c_a|\mathcal{G}(S)| \leq |S'| \leq c_b|\mathcal{G}(S)|$, so the two lengths are equivalent up to a constant. We will represent fixed constants with variants of c; the exact values do not make a difference in the analysis, except that they remain fixed.

▶ **Theorem 20.** *(Yu and Shaw) Suppose we have* $X = \sum_{a \in A} Bernoulli_a(q)$ *for some* $0 < q < 1$. *A proper cover of* A *is a family of subsets* $\{A_i\}_{i \in I}$ *such that all random variables in* $A_i \subset A$ *are independent and* $\bigcup_{i \in I} A_i = A$. *Let* $\chi(\mathcal{A})$ *be the minimum size of the cover,* $|I|$, *over all possible proper covers. Then for* $t \geq 0$,

$$\Pr\left(X \leq \mathbb{E}X - t\right) \leq \exp\left(-\frac{8t^2}{25|A|\chi(\mathcal{A})q}\right).$$

▶ **Lemma 21.** $Pr(N_H \leq m'(1-\theta_T)^k - t) \leq exp(-\frac{8t^2}{25mk(1-\theta_T)^k})$

Proof. We use the previous theorem with $q = (1-\theta_T)^k$. Let $A(i)$ denote the random variable taking on the value of 1 if there is a homologous k-mer match starting from index i of S. Then each set $A_j = \{A(j), A(j+k), \dots\}$ contains mutually independent random variables. Note $A = \bigcup_{i \in I} A_i$, and thus the A_i form a partition of A. This implies that $\chi(\mathcal{A}) \leq k$ and the result follows. ◀

We can now apply this lemma to bound homologous gaps in the generative region of S exactly as in the prequel.

▶ **Lemma 22.** *For any interval consisting of* l k*-mers in* $\mathcal{G}(S)$, *the probability that all* l *homologous anchors are 0 is upper bounded by*

$$\exp\left(-\frac{8l(1-\theta_T)^k}{25k}\right).$$

Proof. For any interval on $\mathcal{G}(S)$ the letters and mutation processes occur with an identical distribution as a $l+k-1$ version of $\mathcal{G}(S)$. Thus, using the previous lemma with $t = l(1-\theta_T)^k$ yields the result. ◀

Working in $\mathcal{G}(S)$, we can now apply the previous lemma to bound homologous gaps w.h.p. exactly as in the prequel.

▶ **Lemma 23** (F2). *With probability* $\geq 1 - \frac{1}{n}$, *no homologous gap in* $\mathcal{G}(S)$ *has size greater than*

$$g(n) = \frac{50k}{8(1-\theta_T)^k}\ln(n) = \frac{C \cdot 50}{8}\log(n)\ln(n) \cdot n^{C\alpha}$$

plus a small $C \log n$ *term we will ignore because it is small asymptotically.*

We now turn to bounding the length of any break in the chain w.h.p. The logic will be to show that w.h.p. an optimal chain cannot be completely spurious, breaks flanked on both sides have length at most $O(\sqrt{m})$, and breaks flanked on a single side - breaks at the start or end of the chain - have length at most $O(\sqrt{m})$. Combining each piece, we will conclude that w.h.p. any break has length at most $O(\sqrt{m})$.

▶ **Lemma 24** ((EC+F1+F2)). *With probability at least* $1 - 6/n$, *every optimal chain* $\big((i_1, j_1), \dots, (i_u, j_u)\big)$ *contains at least one homologous anchor or clipping anchor, provided* n *is sufficiently large.*

Proof. It suffices to show that for large n, the chain with only homologous anchors is higher scoring than any completely spurious chain. This is shown in Supplemental Lemma S4 of Yu and Shaw. ◀

The following lemma shows that any break flanked on both sides has length $< cm^{1/2}$ with high probability.

▶ **Lemma 25** ((EC+F1+F2)). *A break flanked by two at least clipping anchors (clipping or homologous), has length $< c_0\sqrt{m}$ with probability at least $1 - 6/n$ (for sufficiently large n)*

Proof. We first show that the portion of the homologous path in a break that lies below the left anchor is at most $O(k)$. Call the left flanking anchor (i_l, j_l). Since the left flanking anchor is at least clipping, it contains some point on the homologous path. In the EC space, the homologous path must move $O(k)$ indices in S' for every k indices traversed in S and vice-versa, i.e., the homologous path has average slope $O(1)$. Thus, there is some index $i \in S$ for which $i - i_l \leq ck$ and $f(i) \geq j_l$. Applying the same argument to the right flanking anchor shows that, again, at most $O(k)$ points are missed in the break region above the right flank on the homologous path. The remainder of the proof ignores these terms since they contribute $O(k)$ to the break length, which is asymptotically negligible.

As in the previous lemma, the exact proof from Yu and Shaw shows that if the break corresponded to a region of length $\geq m^{1/2}$ in $\mathcal{G}(S)$, then all spurious anchors could be replaced by homologous anchors to obtain a higher scoring chain. Thus, the break in the generative region of S cannot be more than $m^{1/2}$. Under EC, any region of length $m^{1/2}$ cannot correspond to a region of length more than $cm^{1/2}$ in S' and so the total number of missed points is at most $2(m^{1/2} + cm^{1/2}) = c_0 m^{1/2}$, absorbing the constants into c_0. ◀

The following lemma from the prequel holds with the same exact proof and shows that a break flanked on a single side, i.e. a break at the very start or very end of the chain, has length $< cm^{1/2}$ w.h.p. for a fixed constant $c > 0$.

▶ **Lemma 26** ((EC+F1+F2)). *A break flanked on one side by an at least clipping anchor has length $< c\sqrt{m}$ with probability at least $1 - 6/n$ (for sufficiently large n)*

We now conclude the main result of this section: the break length lemma. By combining the previous cases of where a break occurs, we show that with probability $\geq 1 - \frac{6}{n}$, any break length is $< cm^{1/2}$ in an optimal chain.

▶ **Lemma 27** (Break Length Lemma (EC+F1+F2)). *Let $g(n) = \frac{C\,50}{8}\log(n)\ln(n)\,n^{C\alpha}$, and set $\zeta = \frac{1}{6\,g(n)}$. Suppose $C > \min\left(3, \frac{2}{1-2\alpha}\right)$, and that $m = \Omega\left(n^{2C\alpha+\varepsilon}\right)$ for some $\varepsilon > 0$. Then, for all sufficiently large n, with probability at least $1 - 6/n$ no optimal chain contains a break of length $\geq c\sqrt{m}$.*

Proof. Under (EC+F1+F2), any optimal chain contains at least one clipping anchor or homologous anchor meaning that any break is flanked on one or both sides. Consider a break flanked on both sides. Applying Lemma 25 shows that this break has length $< cm^{1/2}$. If the break is flanked on one side, applying Lemma 26, shows the break has length $< cm^{1/2}$. Since (EC + F1 + F2) holds with probability $\geq 1 - 6/n$, the result follows. ◀

4 Recoverability Theorem

We will now move on to the main result: proving that the expected recoverability of seed-chain-extend with indels is $\geq 1 - O\left((\log n)^2\, n^{-C\alpha}\right)$ for large enough n under the conditions

stated in Lemma 27. To show this bound, we will first prove that the recoverability of the chain can be lower bounded as the contribution from three pieces: the aligned fraction, points missed by clipping anchors in the chain, and points missed in breaks. Working in a 'good' space, $EC \cap F_1 \cap F_2$, which occurs with probability $\geq 1 - 6/n$, we have shown that with probability $\geq 1 - 6/n$, break lengths are $< cm^{1/2}$, the total number of clipping anchors is small so they cannot miss many points, and the aligned fraction is $\geq 1 - O(\frac{1}{\sqrt{m}})$. Combining these terms, we will conclude that the expected recoverability of an optimal chain is high.

▶ **Definition 28.** *Let $C = ((i_1, j_1), \ldots, (i_u, j_u))$ be a chain. If (i_l, j_l) is a clipping anchor, we define the set of points it misses as:*

$$Cl(i_\ell, j_\ell) = \big\{(i,j) \in P_H : i_\ell \leq i \leq i_\ell + k - 1,\ (i,j) \notin \{(i_\ell + t,\ j_\ell + t) \colon t = 0, \ldots, k-1\}\big\}.$$

The set of all missed points from clipping anchors is: $Cl(C) = \bigcup_{\substack{(i,j) \\ (i,j)\ clips\ C}} Cl(i,j)$.

▶ **Lemma 29** ((EC+F1+F2)). *Let $C = ((i_1, j_1), \ldots, (i_u, j_u))$ be a chain. With probability $\geq 1 - 6/n$, the recoverability of C can be lower bounded as*

$$R(C) \geq \frac{\min f^{-1}(j_u) - \max f^{-1}(j_1) - L(C) - |Cl(C)| - O(|C|k)}{|P_H|}$$

Proof. We will lower bound the recoverability of P_H between $(\max f^{-1}(j_1),\ j_1)$ and $(\min f^{-1}(j_u),\ j_u)$, which is clearly a lower bound on $R(C)$. Consider any point $(x, y) \in P_H$ that lies between two consecutive homologous or clipping anchors $(i_l, j_l), (i_{l+1}, j_{l+1})$, i.e. $i_1 \leq x \leq i_{l+1}$ and $j_1 \leq x \leq j_{l+1}$ then $(x, y) \in Ext(l)$ for all $(x, y) \in P_H$ unless $x_l \in \{i_l, i_{l+1}\}$ or $y \in \{j_l, j_{l+1}\}$. In other words, (x, y) is recovered unless it lies on the edges of the extension 'box'. Note that any point on P_H on the edges of the extension box must be due to an insertion or deletion at the last position of the (i_l, j_l) anchor or directly before the start of the (i_{l+1}, j_{l+1}) anchor. Under EC, all insertions and deletions have $O(k)$ length, so there are at most $O(k)$ missing points on the path in $Ext(l)$. Applying the same argument shows that at most $O(|C|k)$ points are missed on the edges of extension boxes throughout the entire chain. Exactly $|Cl(C)|$ points on P_H that are covered, but not recovered by clipping chain anchors, are missed. Consider a point $(x, y) \in P_H$ in the break B flanked by $(i_l, j_l), (i_{l+1}, j_{l+1})$, that does not lie on the edge of an extension box and is not covered by a clipping anchor. The anchors $(i_1, j_l), (i_{l+1}, j_{l+1})$ contain points on the homologous path, call them $(x_l, y_l), (x_{l+1}, y_{l+1})$. We have that $x_l \leq i_l < x < i_{l+1} \leq x_{l+1}$ and similarly for y. Thus, (x, y) belongs to the region of the path between those two points, and is counted in $L(B)$.

Let (i_a, j_a) be the first anchor in C that is clipping or homologous, which must exist under our space. Under EC, $|\min f^{-1}(j_a) - i_a| \leq ck$. Call the first break in the chain B. Every point in the break B is counted in $L(C)$. Thus, considering the path between $(\max f^{-1}(j_1), j_1)$ and $(\min f^{-1}(j_u), j_u)$, our expression misses at most $O(k)$ points at the start. By the same logic, at most $O(k)$ missed points at the end of the path are undercounted. Combining terms gives the final inequality.

◀

▶ **Theorem 30.** *The expected recoverability of any chain is $\geq 1 - O\big((\log n)^2\, n^{-C\alpha}\big)$ for large enough n under the conditions of Lemma 27.*

Proof. Recall that the recoverability of a chain C is defined as $R(C) = \frac{|Align(C) \cap P_H|}{|P_H|}$. We will work under $\mathcal{F} = \text{EC} \cap \text{F1} \cap \text{F2}$, so all breaks have length $< cm^{1/2}$, by Lemma 27. Note $\Pr(\mathcal{F}) \geq 1 - \frac{2}{n} - \frac{3}{n} - \frac{1}{n} = 1 - \frac{6}{n}$. Writing $R = R(C)$ for shorthand, we have that

$$\mathbb{E}(R \mid \mathcal{F}) \geq \mathbb{E}(\frac{\min f^{-1}(j_u) - \max f^{-1}(j_1)}{|P_H|} \mid \mathcal{F}) - \mathbb{E}(\frac{L(C) + Cl(C)}{|P_H|} \mid \mathcal{F}) - c_0\mathbb{E}(\frac{|C|}{|P_H|} \mid \mathcal{F})$$

From Lemma 26, $\min f^{-1}(j_u) - \max f^{-1}(j_1) \geq |P_H| - c\sqrt{m}$, so the bound becomes

$$\mathbb{E}(R \mid \mathcal{F}) \geq 1 - \frac{4c_1}{\sqrt{m}} - \mathbb{E}(\frac{L(C) + Cl(C)}{|P_H|} \mid \mathcal{F}) - c_0\mathbb{E}(\frac{|C|}{|P_H|} \mid \mathcal{F})$$

We have $L(C) = \sum_{B \in \text{Breaks}} L(B) \leq C_S m^{1/2} \leq N_S m^{1/2}$, since each break contains some spurious anchor in the chain, and that total is at most the total number of spurious anchors. Under $\mathcal{F}$, $\mathbb{E}(L(C) \mid \mathcal{F}) \leq \mathbb{E}(N_S m^{1/2} \mid \mathcal{F}) \leq \frac{mn^{1-C}}{(1-6/n)}$, using that $\mathbb{E}(N_S) \leq mn^{1-C}$ and the law of total expectation.

Moving on to the second term, note that $Cl(C) \leq kN_C$, so $\mathbb{E}(Cl(C) \mid \mathcal{F}) \leq k\mathbb{E}(N_C \mid \mathcal{F}) \leq c_1 mk^2(1-\theta_T)^k$ by Corollary 19. Under $\mathcal{F}$, $|P_H| \geq c_2 m$ for some constant $c_2 > 0$, so $\mathbb{E}(\frac{Cl(C)}{|P_H|} \mid \mathcal{F}) \leq ck^2(1-\theta_T)^k = ck^2 n^{-C\alpha}$ for some positive constant c. We can bound the third term in the same way, since $|C| = C_H + C_S + C_c \leq N_H + N_S + N_C$ and $\mathbb{E}(N_H + N_S + N_C \mid \mathcal{F}) \leq c' mkn^{-C\alpha}$, because $\mathbb{E}(N_C)$ dominates, so $\mathbb{E}(c_0 k\frac{|C|}{|P_H|} \mid \mathcal{F}) \leq c' k^2 n^{-C\alpha}$.

Thus,

$$\mathbb{E}(R \mid \mathcal{F}) \geq 1 - \frac{4c_1}{\sqrt{m}} - \frac{mn^{1-C}}{(1-6/n)} - ck^2 n^{-C\alpha}$$

Since $\Pr(\mathcal{F}) \geq 1 - 6/n$, we get

$$\begin{aligned}
\mathbb{E}(R) &\geq \mathbb{E}(R \mid \mathcal{F}) \Pr(\mathcal{F}) \\
&\geq \big(1 - \tfrac{4c_1}{\sqrt{m}}\big)\,(1 - \tfrac{6}{n}) \;-\; m\,n^{1-C} \;-\; c\,k^2\,n^{-C\alpha}\,(1 - \tfrac{6}{n}) \\
&= 1 - O\big(m^{-1/2}\big) - O\big((\log n)^2\, n^{-C\alpha}\big) \\
&= 1 - O\big((\log n)^2\, n^{-C\alpha}\big).
\end{aligned}$$

◀

5 Conclusion

In this work, we have shown that under the assumptions of Lemma 27 and $0 < \theta_T < 0.206$, the expected recoverability of an optimal chain under indels is $\geq 1 - O\big((\log n)^2\, n^{-C\alpha}\big)$. This result is weaker than the substitution-only case, in which Yu and Shaw [23] proved that the expected recoverability of an optimal chain is $\geq 1 - O(\frac{1}{\sqrt{m}})$. The weaker bound is due to the existence of clipping anchors, which uniquely arise with indels. In the substitution-only case, an anchor either lies entirely on or off the path. However, when the path is kinked, as it is with indels, this is no longer true. In the prequel, unrecovered points arise from breaks or at the start/end of the chain – no points are missed in regions covered by homologous anchors or on the 'sides' of extension boxes. We handled unrecovered points due to clipping anchors and the sides of extension boxes by naively upper bounding the size of the chain C by the total number of anchors $N_H + N_S + N_C$ and the number of clipping anchors in the chain by N_C. In future work, we aim to bridge the gap between recoverability results under indels and only substitutions by making better use of the match graph's dependence structure, and further analyzing clipping anchor contributions to the chaining score.

References

1 Stephen F Altschul, Warren Gish, Webb Miller, Eugene W Myers, and David J Lipman. Basic local alignment search tool. *Journal of molecular biology*, 215(3):403–410, 1990.

2 Arturs Backurs and Piotr Indyk. Edit distance cannot be computed in strongly subquadratic time (unless seth is false). In *Proceedings of the forty-seventh annual ACM symposium on Theory of computing*, pages 51–58, 2015.

3 Bonnie Berger, Michael S Waterman, and Yun William Yu. Levenshtein distance, sequence comparison and biological database search. *IEEE transactions on information theory*, 67(6):3287–3294, 2020.

4 Roy J Britten. Divergence between samples of chimpanzee and human dna sequences is 5%, counting indels. *Proceedings of the National Academy of Sciences*, 99(21):13633–13635, 2002.

5 Boris Bukh and Raymond Hogenson. Length of the longest common subsequence between overlapping words. *SIAM Journal on Discrete Mathematics*, 34(1):721–729, 2020.

6 Mark J Chaisson and Glenn Tesler. Mapping single molecule sequencing reads using basic local alignment with successive refinement (blasr): application and theory. *BMC bioinformatics*, 13:1–18, 2012.

7 Vacláv Chvátal and David Sankoff. Longest common subsequences of two random sequences. *Journal of Applied Probability*, 12(2):306–315, 1975. doi:10.2307/3212444.

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

9 Arun Ganesh and Aaron Sy. Near-linear time edit distance for indel channels. In *20th International Workshop on Algorithms in Bioinformatics*, page 17, 2020.

10 Ragnar Groot Koerkamp and Pesho Ivanov. Exact global alignment using a* with chaining seed heuristic and match pruning. *bioRxiv*, pages 2022–09, 2022.

11 Pesho Ivanov, Benjamin Bichsel, and Martin Vechev. Fast and optimal sequence-to-graph alignment guided by seeds. In *International Conference on Research in Computational Molecular Biology*, pages 306–325. Springer, 2022.

12 Svante Janson. Large deviations for sums of partly dependent random variables. *Random Structures & Algorithms*, 24(3):234–248, 2004.

13 Marcos Kiwi, Martin Loebl, and Jiří Matoušek. Expected length of the longest common subsequence for large alphabets. *Advances in Mathematics*, 197(2):480–498, 2005.

14 Eugene V Koonin, L Aravind, and Alexey S Kondrashov. The impact of comparative genomics on our understanding of evolution. *Cell*, 101(6):573–576, 2000.

15 Ben Langmead and Steven L Salzberg. Fast gapped-read alignment with bowtie 2. *Nature methods*, 9(4):357–359, 2012.

16 Jüri Lember and Heinrich Matzinger. Standard deviation of the longest common subsequence. *The Annals of Probability*, 37(3):1192 – 1235, 2009. doi:10.1214/08-AOP436.

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

18 Saul B Needleman and Christian D Wunsch. A general method applicable to the search for similarities in the amino acid sequence of two proteins. *Journal of molecular biology*, 48(3):443–453, 1970.

19 Sergey Nurk, Sergey Koren, Arang Rhie, Mikko Rautiainen, Andrey V Bzikadze, Alla Mikheenko, Mitchell R Vollger, Nicolas Altemose, Lev Uralsky, Ariel Gershman, et al. The complete sequence of a human genome. *Science*, 376(6588):44–53, 2022.

20 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*, 17:1–14, 2016.

21 Gesine Reinert, Sophie Schbath, and Michael S Waterman. Probabilistic and statistical properties of words: an overview. *Journal of Computational Biology*, 7(1-2):1–46, 2000.

22 Jim Shaw and Yun William Yu. Theory of local k-mer selection with applications to long-read alignment. *Bioinformatics*, 38(20):4659–4669, 2022.

23 Jim Shaw and Yun William Yu. Proving sequence aligners can guarantee accuracy in almost o (m log n) time through an average-case analysis of the seed-chain-extend heuristic. *Genome Research*, 33(7):1175–1187, 2023.

24 Jouni Sirén, Jean Monlong, Xian Chang, Adam M Novak, Jordan M Eizenga, Charles Markello, Jonas A Sibbesen, Glenn Hickey, Pi-Chuan Chang, Andrew Carroll, et al. Pangenomics enables genotyping of known structural variants in 5202 diverse genomes. *Science*, 374(6574):abg8871, 2021.

25 Temple F Smith, Michael S Waterman, et al. Identification of common molecular subsequences. *Journal of molecular biology*, 147(1):195–197, 1981.

26 Wojciech Szpankowski. *Average case analysis of algorithms on sequences*. John Wiley & Sons, 2011.

27 Esko Ukkonen. On approximate string matching. In *International Conference on Fundamentals of Computation Theory*, pages 487–495. Springer, 1983.

28 Richard Van Noorden, Brendan Maher, and Regina Nuzzo. The top 100 papers. *Nature News*, 514(7524):550, 2014.

29 Y William Yu, Deniz Yorukoglu, Jian Peng, and Bonnie Berger. Quality score compression improves genotyping accuracy. *Nature biotechnology*, 33(3):240–243, 2015.

30 Yun William Yu and Griffin M Weber. Hyperminhash: Minhash in loglog space. *IEEE Transactions on Knowledge and Data Engineering*, 34(1):328–339, 2022.