This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TCBB.2022.3177855, IEEE/ACM Transactions on Computational Biology and Bioinformatics

Improved DNA-versus-Protein Homology Search for Protein Fossils

Yin Yao and Martin C. Frith

Abstract—Protein fossils, i.e. noncoding DNA descended from coding DNA, arise frequently from transposable elements (TEs), decayed genes, and viral integrations. They can reveal, and mislead about, evolutionary history and relationships. They have been detected by comparing DNA to protein sequences, but current methods are not optimized for this task. We describe a powerful DNA-protein homology search method. We use a 64×21 substitution matrix, which is fitted to sequence data, automatically learning the genetic code. We detect subtly homologous regions by considering alternative possible alignments between them, and calculate significance (probability of occurring by chance between random sequences). Our method detects TE protein fossils much more sensitively than `blastx`, and $> 10\times$ faster. Of the ∼7 major categories of eukaryotic TE, three were long thought absent in mammals: we find two of them in the human genome, polinton and DIRS/Ngaro. This method increases our power to find ancient fossils, and perhaps to detect non-standard genetic codes. The alternative-alignments and significance paradigm is not specific to DNA-protein comparison, and could benefit homology search generally. This is an extended version of a conference paper [1].

Index Terms—Pseudogene, homology, alignment, probability, paleovirology

1 Introduction

GENOMES are littered with protein fossils, old and young. They can be found by comparing DNA to known proteins: new transposable element (TE) families have been discovered in this way [2], [3]. An interesting class of protein fossils comes from ancient integrations of viral DNA into genomes, enabling the field of paleovirology [4]. The DNA sequences of protein fossils often have similarity to distantly-related genomes (e.g. mammal versus fish), simply because the parent gene evolved slowly, so it is important to know that the DNA is a protein fossil in order to understand this similarity [5]. DNA-protein homology search is not only used for fossils. It is also used to classify DNA reads from unknown microbes, including nanopore and PacBio reads with many sequencing errors [6]. DNA-protein comparison can be used to find frameshifts during evolution of functional proteins [7], and programmed ribosomal frameshifts [8]. A more specialized and complex kind of DNA-protein comparison, outside this study's scope, considers introns and other gene features to identify genes.

DNA-protein homology search is a classical problem with many old solutions [9], [10], [11], [12], [13], [14], [15], [16], [17], [18], [19], [20]. A notable one is "three-frame alignment" [13], which we believe is the simplest and fastest reasonable way to do frameshifting DNA-protein alignment. Nevertheless, we can significantly improve DNA-protein homology search in these aspects:

- Better parameters for the (dis)favorability of substitutions, deletions, insertions, and frameshifts. Most previous methods use standard parameters such as the BLOSUM62 substitution matrix, which is designed for functional proteins, and likely completely inappropriate for protein fossils. We optimize these parameters by fitting them to sequence data.
- Instead of a 20×20 substitution matrix, use a 64×21 matrix (64 codons × 20 amino acids plus STOP). This allows e.g. preferred alignment of asparagine (which is encoded by `aac` and `aat`) to `agc` than to `tca`, which both encode serine.
- Incorporate frameshifts into affine gaps. Because gaps are somewhat rare but often long, it is standard to disfavor opening a gap more than extending a gap. However, most previous methods favor frameshifts equally whether isolated or contiguous with a longer gap.
- Detect homologous regions based on not just one alignment between them, but on many possible alternative alignments. This is expected to detect subtle homology more powerfully [21], [22].
- Calculate significance, i.e. the probability of such a strong similarity occurring by chance between random sequences. To this day, for ordinary alignment, BLAST can only calculate significance for a few hardcoded sets of substitution and gap parameters. We can do it for any parameters, for similarities based on many alternative alignments.

We also aimed for maximum simplicity and speed, inspired by three-frame alignment.

2 Methods

2.1 Alignment Elements

We define a DNA-protein alignment to consist of: matches (3 bases aligned to 1 amino acid), base insertions, and base deletions. To keep things simple, insertions are not allowed between bases aligned to one amino acid: if such an insertion really exists, we can approximate it with a nearby insertion. A deletion of length not divisible by 3 leaves "dangling"

- *Y. Yao was and M. Frith is with the Graduate School of Frontier Sciences, University of Tokyo, Chiba, Japan, and the Artificial Intelligence Research Center, AIST, Tokyo, Japan.*
- *M. Frith is with the Computational Bio Big-Data Open Innovation Laboratory (CBBD-OIL) AIST, Tokyo, Japan.*

(Corresponding author: Martin C. Frith.)

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TCBB.2022.3177855, IEEE/ACM Transactions on Computational Biology and Bioinformatics


```
Ser-TyrAlaThrMetLeuTrpAspGln--Leu***
tctCtat---acg--cctctga-atcagCAttctaa
```

Fig. 1. Example of a DNA-versus-protein alignment. `***` indicates a protein end from translation of a stop codon. Insertions are bold uppercase. "Dangling" bases, left by deletions of length not divisible by 3, are underlined gray.

bases (Fig. 1): for simplicity, we do not attempt to align these (equivalently, align them to the amino acid with score 0).

2.2 Scoring Scheme

An alignment's score is the sum of:

- Score for aligning amino acid u to base triplet V:

$$S_{uV}$$

- Score for an insertion of k bases:

$$a_I + b_I k + \begin{cases} 0 & \text{if } k \bmod 3 = 0 \\ f_I & \text{if } k \bmod 3 = 1 \\ g_I & \text{if } k \bmod 3 = 2 \end{cases}$$

- Score for a deletion of k bases:

$$a_D + b_D k + \begin{cases} 0 & \text{if } k \bmod 3 = 0 \\ f_D & \text{if } k \bmod 3 = 1 \\ g_D & \text{if } k \bmod 3 = 2 \end{cases}$$

This scheme extravagantly uses 4 frameshift parameters (f_I, g_I, f_D, g_D), because it's based on a probability model with 4 frameshift transitions (Fig. 2), and we can't think of a good way to simplify the model. Overall, our alignment scheme is similar to FramePlus [15] and especially to aln [16].

2.3 Finding a Maximum-Score Local Alignment

A basic approach is to find an alignment with maximum possible score, between any parts of a protein sequence $R_1 \dots R_M$ and a DNA sequence $q_1 \dots q_N$. Let Q_j mean the triplet q_{j-2}, q_{j-1}, q_j. We can calculate the maximum possible score X_{ij} for any alignment ending just after R_i and q_j, for $0 \le i \le M$ and $0 \le j \le N$ (with notation y/Y for deletion and z/Z for insertion):

$$\begin{aligned} y_1 &= Y_{i-1\ j-2} + [b_D + f_D] & z_1 &= Z_{i\ j-1} + [b_I + f_I] \\ y_2 &= Y_{i-1\ j-1} + [2b_D + g_D] & z_2 &= Z_{i\ j-2} + [2b_I + g_I] \\ y_3 &= Y_{i-1\ j} + [3b_D] & z_3 &= Z_{i\ j-3} + [3b_I] \\ X_{ij} &= \max(X_{i-1\ j-3} + S_{R_i Q_j},\ y_1,\ y_2,\ y_3,\ z_1,\ z_2,\ z_3,\ 0) \\ Y_{ij} &= \max(X_{ij} + a_D,\ y_3) & Z_{ij} &= \max(X_{ij} + a_I,\ z_3) \end{aligned}$$

The boundary condition is: if $i < 0$ or $j < 0$, $X_{ij} = Y_{ij} = Z_{ij} = -\infty$ (which takes care of $R_{i<1}$ and $Q_{j<3}$). The maximum possible alignment score is $\max(X_{ij})$, and an alignment with this score can be found by a standard traceback [23].

For each (i, j) this algorithm retrieves 7 previous results, and performs 9 pairwise maximizations and 9 additions (which could be reduced to 6 additions if each insertion cost equals its corresponding deletion cost). This is slightly slower than three-frame alignment, which retrieves 5 previous results and performs 7 pairwise maximizations and 6 additions.

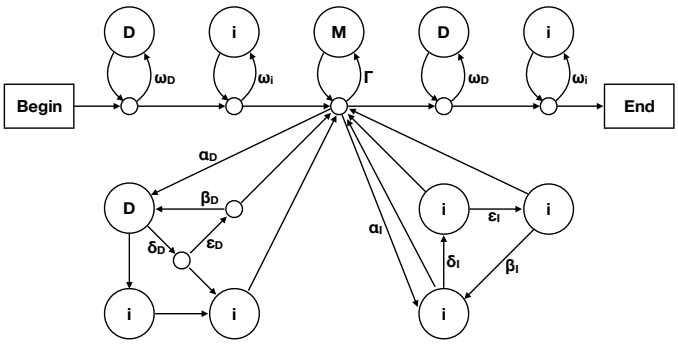

Fig. 2. A probability model for related DNA and protein sequences. The arrows are labeled with probabilities of traversing them. Each pass through an **i** state generates (emits) one base $v \in \{\mathtt{a}, \mathtt{c}, \mathtt{g}, \mathtt{t}\}$, with probabilities ψ_v. Each pass through a **D** state generates one amino acid u, with probabilities ϕ_u. Each pass through the **M** state generates one amino acid u aligned to three bases $V = v_1 v_2 v_3$, with probabilities π_{uV}. The two bottom-left **i** states correspond to "dangling" bases.

2.4 Probability Model

The preceding algorithm is equivalent to finding a maximum-probability path generating the sequences, through a probability model (Fig. 2). Such models are explained in [23], [24]. Briefly, the model is a scheme for generating sequences: starting at **Begin**, randomly traverse the arrows according to their probabilities (e.g. ω_D versus $1 - \omega_D$), generate letters when passing through circles labeled **D**, **i**, and **M** (see the figure legend), until hitting **End**.

The score and model parameters are related like this:

$$S'_{uV} = \exp\Big(\frac{S_{uV}}{t}\Big) = \frac{\Gamma}{\omega_D \omega_i^3} \cdot \frac{\pi_{uV}}{\phi_u \psi_V} \tag{1}$$

$$a'_I = \exp\Big(\frac{a_I}{t}\Big) = \frac{\alpha_I(1-\beta_I)}{\beta_I} \tag{2}$$

$$a'_D = \exp\Big(\frac{a_D}{t}\Big) = \frac{\alpha_D(1-\beta_D)}{\beta_D} \tag{3}$$

$$b'_I = \exp\Big(\frac{b_I}{t}\Big) = \frac{\sqrt[3]{\beta_I \delta_I \epsilon_I}}{\omega_i} \tag{4}$$

$$b'_D = \exp\Big(\frac{b_D}{t}\Big) = \sqrt[3]{\frac{\beta_D \delta_D \epsilon_D}{\omega_D}} \tag{5}$$

$$f'_I = \exp\Big(\frac{f_I}{t}\Big) = \frac{1-\delta_I}{1-\beta_I} \sqrt[3]{\frac{\beta_I^2}{\delta_I \epsilon_I}} \tag{6}$$

$$f'_D = \exp\Big(\frac{f_D}{t}\Big) = \frac{1-\delta_D}{1-\beta_D} \sqrt[3]{\frac{\beta_D^2}{\delta_D \epsilon_D \omega_D^2}} \Big/ \omega_i^2 \tag{7}$$

$$g'_I = \exp\Big(\frac{g_I}{t}\Big) = \frac{1-\epsilon_I}{1-\beta_I} \sqrt[3]{\frac{\beta_I \delta_I}{\epsilon_I^2}} \tag{8}$$

$$g'_D = \exp\Big(\frac{g_D}{t}\Big) = \frac{1-\epsilon_D}{1-\beta_D} \sqrt[3]{\frac{\beta_D \delta_D}{\epsilon_D^2 \omega_D}} \Big/ \omega_i \tag{9}$$

Here ψ_V is defined to be $\psi_{v_1} \psi_{v_2} \psi_{v_3}$, and t is an arbitrary positive constant (because multiplying all the score parameters by a constant makes no difference to alignment). An

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TCBB.2022.3177855, IEEE/ACM Transactions on Computational Biology and Bioinformatics

alignment score is then:

$$t \ln \left[\frac{\text{prob(path \& sequences)}}{\text{prob(null path \& sequences)}} \right], \tag{10}$$

where a "null path" is a path that never traverses the Γ, α_D, or α_I arrows [24].

2.5 Balanced Length Probability

A fundamental property of local alignment models is whether they are biased towards longer or shorter alignments [24]. If ω_D and ω_i are large (close to 1) and $\Gamma + \alpha_D + \alpha_I$ is small, there is a bias in favor of shorter alignments. In the converse situation, there is a bias towards longer alignments.

To assess this precisely, first note that we can vary ω_D and ω_i with no effect on the left hand sides of Equations 1–9, thus no effect on any of our homology search procedures, if we co-vary the other arrow probabilities in a suitable way. If we define

$$c' = \Gamma / (\omega_D \omega_i^3) \,, \tag{11}$$

we can co-vary Γ so as to keep c' fixed. We can also co-vary β_I, δ_I, and ϵ_I to keep b'_I, f'_I, and g'_I fixed: this requires that

$$\beta_I = \frac{b'_I \omega_i (f'_I + g'_I b'_I \omega_i + (b'_I \omega_i)^2)}{1 + f'_I b'_I \omega_i + g'_I (b'_I \omega_i)^2} \,. \tag{12}$$

We can also co-vary α_I to keep a'_I fixed:

$$\alpha_I = \frac{a'_I b'_I \omega_i (f'_I + g'_I b'_I \omega_i + (b'_I \omega_i)^2)}{1 - (b'_I \omega_i)^3} \,. \tag{13}$$

Likewise, we can co-vary the four deletion probabilities to keep a'_D, b'_D, f'_D and g'_D fixed, which requires that

$$\alpha_D = \frac{a'_D b'_D \omega_D (f'_D \omega_i^2 + g'_D b'_D \omega_i + b'^2_D)}{1 - b'^3_D \omega_D} \,. \tag{14}$$

These equations imply that, if we increase ω_D and ω_i, then Γ, α_I, and α_D increase. Now consider what happens if we increase ω_D and ω_i as much as possible. It may happen that we can increase ω_D and ω_i infinitessimally close to 1, at which point $\Gamma + \alpha_D + \alpha_I$ remains less than 1 by a finite amount: then the model is biased to shorter alignments. We may instead reach a point where $\Gamma + \alpha_D + \alpha_I$ becomes infinitessimally close to 1 but $\omega_D \omega_i$ does not: then the model is biased to longer alignments. Thus, the model is unbiased when they approach 1 together, which happens when

$$\begin{aligned} c' &+ \frac{a'_I b'_I (f'_I + g'_I b'_I + b'^2_I)}{1 - b'^3_I} \\ &+ \frac{a'_D b'_D (f'_D + g'_D b'_D + b'^2_D)}{1 - b'^3_D} = 1 \,. \end{aligned} \tag{15}$$

2.6 Sum over All Alignments Passing through (i, j)

To find subtly homologous regions, we should assess their homology without fixing an alignment [21], [22]. In other words, we should use a homology score like this:

$$t \ln \left[\frac{\sum_{\text{paths}} \text{prob(path \& sequences)}}{\text{prob(null path \& sequences)}} \right]. \tag{16}$$

However, if the sum is taken over all possible paths, we learn nothing about location of the homologous regions, which is important if e.g. the DNA sequence is a chromosome. There is a kind of uncertainty principle here: the more we pin down the alignment, the less power we have to detect homology. As a compromise, we sum over all paths passing through one (protein, DNA) coordinate pair (i, j). This has two further benefits: it is approximated by the seed-and-extend search used for big sequence data, and we can calculate significance.

To calculate this sum over paths, we first run a Forward algorithm for $0 \le i \le M$ and $0 \le j \le N$:

$$\begin{aligned} y_1 &= [b'_D f'_D] Y^F_{i-1\ j-2} & z_1 &= [b'_I f'_I] Z^F_{i\ j-1} \\ y_2 &= [b'^2_D g'_D] Y^F_{i-1\ j-1} & z_2 &= [b'^2_I g'_I] Z^F_{i\ j-2} \\ y_3 &= [b'^3_D] Y^F_{i-1\ j} & z_3 &= [b'^3_I] Z^F_{i\ j-3} \\ X^F_{ij} &= S'_{R_i Q_j} X^F_{i-1\ j-3} + y_1 + y_2 + y_3 + z_1 + z_2 + z_3 + 1 \\ Y^F_{ij} &= a'_D X^F_{ij} + y_3 & Z^F_{ij} &= a'_I X^F_{ij} + z_3 \end{aligned}$$

The boundary condition is: if $i < 0$ or $j < 0$, $X^F_{ij} = Y^F_{ij} = Z^F_{ij} = 0$. Note this is exactly the maximum-score algorithm with score-maximization replaced by summing exponentiated scores. We then run a Backward algorithm for $M \ge i \ge 0$ and $N \ge j \ge 0$:

$$\begin{aligned} y_1 &= [b'_D f'_D] Y^B_{i+1\ j+2} & z_1 &= [b'_I f'_I] Z^B_{i\ j+1} \\ y_2 &= [b'^2_D g'_D] Y^B_{i+1\ j+1} & z_2 &= [b'^2_I g'_I] Z^B_{i\ j+2} \\ y_3 &= [b'^3_D] Y^B_{i+1\ j} & z_3 &= [b'^3_I] Z^B_{i\ j+3} \\ X^B_{ij} &= S'_{R_{i+1} Q_{j+3}} X^B_{i+1\ j+3} + y_1 + y_2 + y_3 + z_1 + z_2 + z_3 + 1 \\ Y^B_{ij} &= a'_D X^B_{ij} + y_3 & Z^B_{ij} &= a'_I X^B_{ij} + z_3 \end{aligned}$$

The boundary condition is: if $i > M$ or $j > N$, $X^B_{ij} = Y^B_{ij} = Z^B_{ij} = 0$. Finally, $t \ln[X^F_{ij} X^B_{ij}]$ is the desired homology score, for all paths passing through (i, j).

2.7 Significance Calculation

The just-described homology score is similar to that of "hybrid alignment", which has a conjecture regarding significance [25]. (Hybrid alignment sums over paths ending at (i, j), instead of passing through (i, j).) We make a similar conjecture. Suppose we compare a random i.i.d. protein sequence of length M and letter probabilities Φ_u to a random i.i.d. DNA sequence of length N and triplet probabilities Ψ_V. We conjecture that the score $s_{\max} = t \ln[\max_{ij}(X^F_{ij} X^B_{ij})]$ follows a Gumbel distribution:

$$\text{prob}(s_{\max} < s) = \exp(-KMNe^{-s/t}) \,, \tag{17}$$

in the limit that M and N are large, provided that:

$$\begin{aligned} \Big(\sum_{u,V} \Phi_u \Psi_V S'_{uV} \Big) &+ \frac{a'_I b'_I (f'_I + g'_I b'_I + b'^2_I)}{1 - b'^3_I} \\ &+ \frac{a'_D b'_D (f'_D + g'_D b'_D + b'^2_D)}{1 - b'^3_D} = 1 \,. \end{aligned} \tag{18}$$

Equation 18 is analogous to Equation 27 or 28 in [25], see also [24]. In practice, we assume that $\Phi_u = \phi_u$ and $\Psi_V = \psi_V$, which makes Equation 18 equivalent to Equation 15.

This conjecture leaves one unknown Gumbel parameter K. We estimate it by generating 50 pairs of pseudorandom protein and codon sequences [26], with $\Phi_u = \phi_u$, $\Psi_V = \psi_V$, $M = 200$ and $N = 602$, and calculating

$$K = 1 / (MN \, \text{avg}[\exp(-s_{\max}/t)]) \,. \tag{19}$$

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TCBB.2022.3177855, IEEE/ACM Transactions on Computational Biology and Bioinformatics

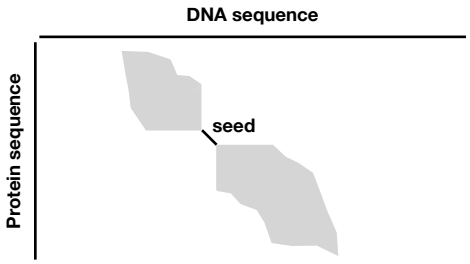

Fig. 3. Sketch of seed-and-extend heuristic for homology search.

This takes zero human-perceptible run time.

2.8 Seed-and-Extend Heuristic

To find homologous regions in big sequence data, we use a BLAST-like seed-and-extend heuristic (Fig. 3) [27]. We first find "seeds": we currently use exact-matches (via the genetic code), which can be sensitive if short, but we could likely get better sensitivity per run time with inexact seeds [28], [29]. Our seeds have variable length: starting from each DNA base, we get the shortest seed that occurs $\leq m$ times in the protein data [30]. These seeds have no score threshold. We then try a gapless X-drop extension in both directions, and if the score achieves a threshold d, we try a "Forward" extension in both directions.

We use our Forward algorithm, modified for semi-global instead of local alignment. In each direction, we sum over alignments starting at the seed and ending anywhere: thus the algorithm's $+1$ is done only at the first (i, j) next to the seed, and we accumulate the sum $W = \sum_{ij} X^F_{ij}$. We run this algorithm in increasing order of antidiagonal $(3i + j)$ on the seed's right side (decreasing order on the left side). If X^F_{ij} is less than a fraction f of W accumulated over previous antidiagonals, we stop extending, which defines the boundary of the gray region in Fig. 3. The final homology score is

$$t \ln[W_{\text{left}}] + \text{seed score} + t \ln[W_{\text{right}}] \,. \tag{20}$$

Sum-of-path algorithms are prone to numerical overflow [23]. To prevent that: once per 32 antidiagonals, we multiply all the X^F, Y^F, and Z^F values in the last six antidiagonals by a scaling factor of $1/W$.

A score with no alignment is of limited use, so we get a representative alignment by a similar semi-global modification of our maximum-score alignment algorithm. To avoid redundancy, we prioritize homology predictions by score (breaking ties arbitrarily), and discard any prediction whose representative alignment shares an (i, j) left or right end with a higher-priority prediction.

There are two options for further redundancy removal. The first (selected by `lastal` option `-K1`) omits a homology prediction if the DNA range of its representative alignment is contained in a higher-scoring prediction on the same DNA strand. The second (`lastal` option `-K0`) omits a homology prediction if the DNA range of its representative alignment overlaps a higher-scoring prediction on the same DNA strand. These aim to get the best homologs for each part of the DNA.

2.9 Fitting Substitution & Gap Parameters to Data

We can seek substitution and gap parameters with maximum-likelihood fit to some related (unaligned) DNA and proteins, by expectation-maximization [23]. Starting from some initial parameters, we calculate the expected count of each transition and emission (E step), then update the model probabilities to maximize the probability of the expected counts (M step), and repeat until convergence. We implemented two versions of this: an exact $O(MN)$ version, and a heuristic seed-and-extend version.

The seed-and-extend version is given some proteins and DNA (e.g. a genome): to save time, it just uses a sample of the DNA, by cutting it into 2000-base segments and pseudorandomly choosing 20 000 of them. In each E step, it finds significantly homologous regions (with `-K1` filtering) and gets expected counts from the seeds and extend regions (gray areas in Fig. 3). In the M step, it does not infer ϕ_u, ψ_v, ω_D, or ω_i in the usual way: it sets $\phi_u = \sum_V \pi_{uV}$, $\psi_v = \sum_{uij}(\pi_{u\,vij} + \pi_{u\,ivj} + \pi_{u\,ijv})/3$, and $\omega_D = \omega_i^3 =$ the value that satisfies Equation 15 (found by bisection with bounds $1 > \omega_i^3 > \beta_I\delta_I\epsilon_I$ and $1 > \omega_D > \beta_D\delta_D\epsilon_D$). We set $t = 3/\ln[2]$ to get scores in third-bit units.

It is likely that some of the 64×21 pairings are absent from the training data, so that naive fitting gives them $\pi_{uV} = 0$ and $S_{uV} = -\infty$. This can be avoided by adding pseudocounts to the expected counts [23]. We first tried constant pseudocounts, e.g. $P_{uV} \leftarrow P_{uV} + 0.5$, where P_{uV} is the expected count of $u{:}V$. However, we found it necessary to weight the pseudocounts by the letter frequencies $\hat{P}_u = 1 + \sum_V P_{uV}$ and $\check{P}_V = 1 + \sum_u P_{uV}$:

$$P_{uV} \leftarrow P_{uV} + 0.5(\hat{P}_u / \operatorname{avg} \hat{P})(\check{P}_V / \operatorname{avg} \check{P}) \,. \tag{21}$$

2.10 Fitting to Remote Homologs

Substitution and gap rates differ for old versus young fossils, whereas the sequence data used for parameter fitting may have a mixture of old and young fossils. Therefore, to focus on old fossils, our seed-and-extend fitting ignores homologous regions whose expected counts P_{uV} have percent identity greater than "pid", a BLOSUM-like threshold [31]. The percent identity of P_{uV} depends on the genetic code, which is inferred before each E step by $\operatorname{argmax}_u[\pi_{uV}/\phi_u]$.

3 Results

3.1 Software

The $O(MN)$ alignment and fitting code is available at https://github.com/Yao-Yin/protein-dna-align-EM. Seed-and-extend fitting and homology search, and estimation of K by full Forward-Backward algorithm, are available in LAST (https://gitlab.com/mcfrith/last).

3.2 Parameter Fitting

We applied our $O(MN)$ fitting to a set of human processed pseudogenes and their parent proteins from Pseudofam [32]. To avoid bias, we used uniform initial parameters $\pi_{uV} = 1/(21 \cdot 64)$. The fitting discovered the genetic code: for each codon V, its encoded amino acid has maximum S_{uV}.

Sometimes, our fitting had an undesirable feature: the S_{uV} values for some `cg`-containing codons were all negative.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TCBB.2022.3177855, IEEE/ACM Transactions on Computational Biology and Bioinformatics

TABLE 1
Parameter fitting to the human genome and TE proteins, with different pid thresholds

pid	alignments	expected π counts	$S_{m\,atg}$	$S_{m\,atc}$	a_D	a_I	b_D	b_I	f_D	g_D	f_I	g_I	ψ_a	ψ_c	ψ_g	ψ_t
100	13445	2543821	14	−2	−23	−29	−1	−1	+3	0	+4	0	0.4	0.19	0.17	0.23
90	12799	2356744	13	−2	−23	−29	−1	−1	+3	0	+4	0	0.4	0.19	0.17	0.24
80	11585	2109071	13	−1	−22	−28	−1	−1	+3	0	+4	0	0.41	0.18	0.17	0.24
70	9293	1680244	12	0	−22	−28	−1	−1	+3	0	+3	0	0.41	0.18	0.17	0.24
60	6657	1213898	12	0	−21	−27	−1	−1	+2	0	+3	0	0.4	0.18	0.17	0.25
50	3360	607464	11	1	−21	−27	−1	−1	+2	0	+3	0	0.39	0.18	0.17	0.26
40	1287	251719	10	1	−21	−25	−1	−1	+2	0	+2	0	0.37	0.19	0.17	0.26

This is presumably due to the well-known depletion of `cg` in human DNA, which can be captured in π_{uV} but not ψ_v. As an ad hoc fix, we set $\psi_V = \sum_u \pi_{uV}$ (after $O(MN)$ fitting, and at each iteration of seed-and-extend fitting).

Next, we applied our seed-and-extend fitting to the human genome (hg38) and transposable element (TE) proteins from RepeatMasker 4.1.0 [33], with various pid thresholds, using LAST v1250:

```
lastdb -q -c myDB RepeatPeps.lib
last-train --codon -X1 --pid=P myDB genome.fa > P.fit
```

Option `-q` appends ⋆ to each protein, `-c` requests simple-sequence masking (see below), and `-X1` sets substitution scores for letter `X` (which is frequent in these proteins) by

$$S'_{XV} = \sum_{u \in \text{20 aminos}} \phi_u S'_{uV} \Big/ \sum_{u \in \text{20 aminos}} \phi_u \,. \qquad (22)$$

As pid decreases, fewer alignments and expected counts are used for parameter fitting (Table 1, final E step), but the results do not change drastically, suggesting the amount of data remains adequate. As expected, when pid decreases, the parameters favor matches and disfavor mismatches and gaps less strongly (Table 1).

The fitted frameshift scores are surprising (Table 1). Firstly, frameshifts are not disfavored, perhaps because RepeatMasker's proteins are close to the fossils' most recent active ancestors. Secondly, mod-1 frameshifts are *favored*: this might be caused by the gap-length distribution not fitting the simple affine model, with an excess of length-1 and length-4 gaps [34]. Another surprise is that the fitted parameters are adenine-rich: $\psi_a \approx 0.4$ (Table 1). This matches the `a`-richness of L1 LINEs [35], the most abundant type of protein fossil.

The fitted substitution parameters include homology probabilities that would be lost in a 20×20 matrix (Fig. 4). For example, `acc` and `acg` both encode `T`, but `acg` is more favored to align with `M` (which is encoded by `atg`). Similarly, with pid=100, `N` (which is encoded by `aac` and `aat`) scores +4 with `agc` and −16 with `tca`, which both encode `S`. In general, single `a`↔`g` or `c`↔`t` mismatches tend to be favored.

3.3 Significance Calculation

To test the accuracy of our significance estimates, for the fitted genome-TE parameters, we calculated $s_{\max}$ by our full Forward-Backward algorithm for 10 000 pairs of random i.i.d. protein and codon sequences, with $\Phi_u = \phi_u$, $\Psi_V = \psi_V$, $M = 1000$, and $N = 3002$. After fitting K to these $s_{\max}$ values (Equation 19), the observed distribution of $s_{\max}$ is accurately predicted by Equation 17 (Fig. 5).

An interesting result is that K increases as pid decreases (Fig. 5). This is opposite to the behavior of K for ordinary gapless alignment [36], but consistent with previous results for hybrid alignment [25]. Perhaps insight into K could be gained by considering gapless sum-of-paths alignment.

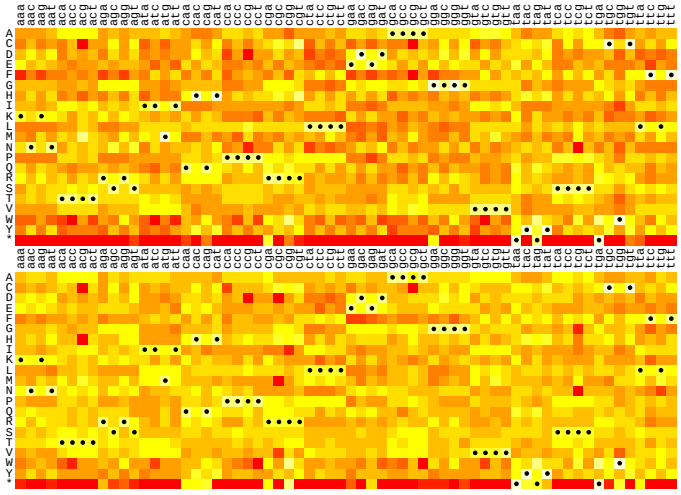

Fig. 4. Substitution matrices inferred from the human genome versus TE proteins, with pid=100 (top) or pid=40 (bottom). Darker red means more disfavored and paler yellow means more favored. Black dots indicate the standard genetic code.

To test whether our significance estimates apply to our seed-and-extend homology search, we compared one pair of random i.i.d. protein and DNA sequences, with $\Phi_u = \phi_u$, $\Psi_v = \psi_v$, and lengths equal to the number of unambiguous letters in the TE proteins and human chromosome 21 (chr21). Here, we used `lastal` with neither `-K` option. The search sensitivity depends on the seed parameter m: as m increases, sensitivity increases, and the distribution of homology scores approaches the Gumbel prediction (Fig. 6 top row).

Thus, we can accurately calculate probabilities of homology scores between random sequences with letter frequencies equal to the fitted ϕ and ψ, but unfortunately the genome has different base frequencies (`a`:`c`:`g`:`t` = 3:2:2:3). So we compared another pair of random i.i.d. protein and DNA sequences, with Φ_u and Ψ_v equal to the frequencies in the TE proteins and chr21. The number of homology predictions, for any score threshold, increased by a factor of about 3 (Fig. 6 top row, dashed versus solid blue lines), so the E-values (expected counts) are about 3× too low.

3.4 Simple Sequences

Homology search is confounded by "simple sequences", e.g. `ttttcttttttcctt`, which evolve frequently and independently. To assess this problem, we compared reversed

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TCBB.2022.3177855, IEEE/ACM Transactions on Computational Biology and Bioinformatics

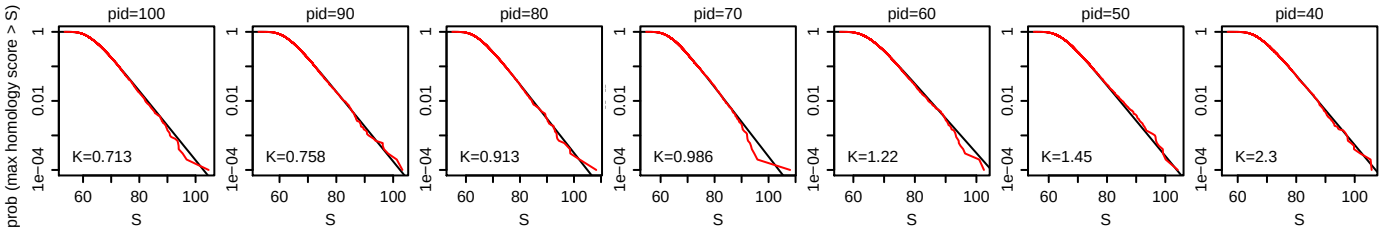

Fig. 5. Distributions of maximum homology score s_{max}, for 10 000 pairs of random protein and codon sequences, with $\Phi_u = \phi_u$ and $\Psi_V = \psi_V$. Red: observed, black: expected.

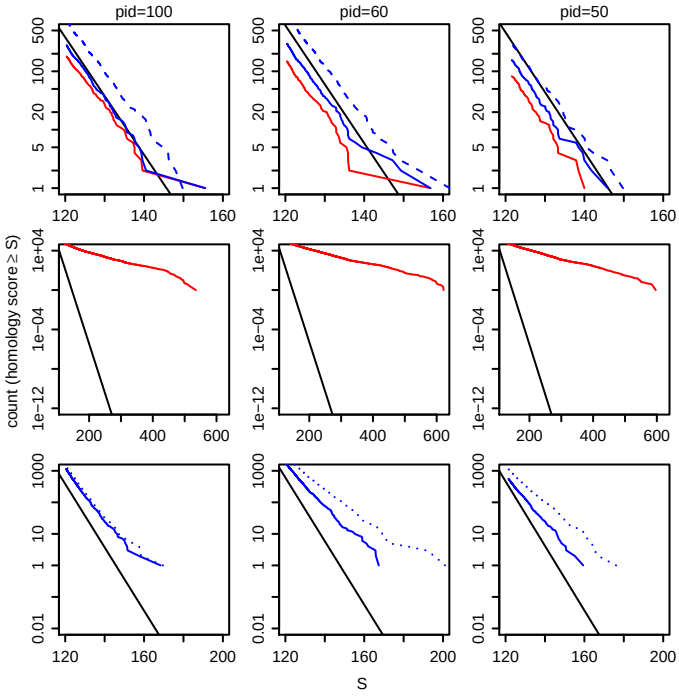

Fig. 6. Distributions of homology scores. Red: found with seed count limit $m = 100$. Blue: $m = 1000$. Black: expected number of homology scores $\geq S$ between random sequences. Top row: homology scores between a random DNA sequence with the same length as chromosome 21, and a random protein sequence with the same length as the TE protein database. Solid lines: $(\Phi, \Psi) = (\phi, \psi)$. Dashed lines: $(\Phi, \Psi) =$ letter frequencies in the TE proteins and chr21. Middle row: homology scores between TE proteins & reversed (but not complemented) chr21. Bottom row: same as middle row, with masking of simple sequences (dotted: $r = 0.005$, solid: $r = 0.02$).

(but not complemented) chr21 to the TE proteins: this test has no true homologies, but we found many highly-significant homology scores (Fig. 6 middle row), due to simple sequences (Fig. 7).

There are various methods to suppress such false homologies, but most do not fully succeed [37], [38]. Our solution is to identify simple sequences with `tantan` [37], and "mask" them: i.e. set their substitution scores $S \leq 0$ thus $S' \leq 1$ [38]. We use masking for homology scores but not for representative alignments.

`tantan` has a parameter `r` (simple-sequence probability): higher values mask a larger fraction of the sequence, i.e. they mask more-subtle simple sequences. Based on previous results [37], we tried masking the DNA with $r = 0.005$ and the proteins with $r =$ either 0.005 (which masks 3.7% of the TE proteins) or 0.02 (which masks 6.8%). The former retained

```
GlyGlyGlyGlyGlySerGlyGlyGlyGlySerGlyGlyGlyGlyGlyGlyGlyGlyGlySerGlyGly
GGTGGTGGTGGTAGTGGTAGTGGTGGTAGTGGTAGTAGTAGTGGTGGTAGTAGTGGTGGTGGTGGTAGTGGTGGT

GlyGlyPheGlyGlyGlyProGlyGlyGlyAsnGlyGlyGlyMetGlyIlePheGlyThrAlaGlyAlaGlyGly
AGTGGTAGTAGTGGTGGTGGTAGTGGTGGTGGTAGTGGTGGTGGTAGTGGTGGT---GGTAGTGGTGGTAGTGGTGGT

GlyGlyTyrGlyGlyGlyProGlyAspGlyGlyGlyHisGlyGlyGlyGlyGlyAspGlyGlyAlaGlyGly
GGTAGTGGTGGTAGTGGTGGTGGTGGTGGTGGTGTTGGTGGTGGTGGTAGTGGTGGTGGTAGTGGTGGT
```

Fig. 7. A similarity with E-value 3×10^{-38} between reversed (but not complemented) human chromosome 21 and a TE protein (found with pid=100).

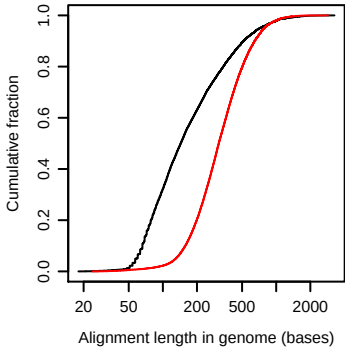

Fig. 8. Lengths of alignments between the human genome and TE proteins. Black: alignments found with pid=100 but not pid=50. Red: found with pid=50 but not pid=100.

false homologies with E-value $< 10^{-5}$ (Fig. 6 bottom row), so we use the latter.

3.5 Sensitivity of Different pid Thresholds

Next, we sought TE protein fossils in the human genome. To avoid TE/gene homologies, we compared the genome to TE proteins plus human proteins[1] from SwissProt version 2021_02 [39], then discarded the SwissProt hits. We used `lastal` parameters `-K0`, $m = 500$, and a significance level of one expected false homology per 10^9 random bp. This procedure found >500 000 hits (Table 2), but only 12–138 hits to the reversed genome, implying a false positive rate of 2–25 per 100 000 predictions.

Different pid thresholds find slightly different homologies. Some alignments found with one pid are completely missed (no alignment with any overlap on the same DNA strand) with another pid (Table 2). We expect high pid to be better at finding high-similarity homologies that are so short that their significance is borderline, and low pid to be better at low-similarity (but longer) homologies. The alignment lengths support this expectation (Fig. 8).

1. Non-LINE human proteins with PE (protein existence) level ≤ 3

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TCBB.2022.3177855, IEEE/ACM Transactions on Computational Biology and Bioinformatics

TABLE 2
Counts of alignments between the human genome and TE proteins

pid	alignments	(reversed genome)	of which not found (zero overlap) by pid=100	pid=90	pid=80	pid=70	pid=60	pid=50	pid=40
100	539769	12	0	4155	9048	12438	16882	25327	42444
90	544192	17	9003	0	6746	10307	14935	24074	42278
80	551880	18	23425	16212	0	6290	10832	21599	42608
70	553791	27	28875	21812	8308	0	7239	18610	40890
60	559627	138	39953	33064	19522	13849	0	16775	41415
50	547453	19	36934	30695	18726	13569	5081	0	28332
40	524215	90	32163	27002	17828	13973	7776	6225	0

3.6 Comparison to blastx

To test whether our homology search is more sensitive than standard methods, we compared chr21 to the TE proteins with NCBI BLAST 2.11.0:

```
makeblastdb -in RepeatPeps.lib -dbtype prot -out DB
blastx -query chr21.fa -db DB -evalue 0.1 -outfmt 7 > out
```

We repeated this comparison using our method with pid=70:

```
lastal -p 70.fit -D1e9 -m100 -K0 myDB chr21.fa > out
```

Option `-D1e9` sets the significance threshold to 1 random hit per 10^9 basepairs, and `-m100` sets $m = 100$.

This test indicated that our method has much better sensitivity and speed. The single-threaded runtimes were 193 min for `blastx` and 17 min for `lastal`. (We also tried `fasty` from FASTA version 36.3.8h [14], but it didn't finish within 30 hours.) `blastx` found alignments at 5267 non-overlapping sites on the two strands of chr21, of which all but 101 overlapped LAST alignments[2]. LAST found alignments at 6761 non-overlapping sites, of which 2450 did not overlap `blastx` alignments. All but 19 of LAST's sites overlapped same-strand annotations by RepeatMasker open-4.0.6 - Dfam 2.0 (excluding Simple_repeat and Low_complexity) [33], [40], suggesting they are not spurious. Note that RepeatMasker finds TEs by DNA models of tightly-defined TE families, which is likely superior to DNA-protein comparison when such models are available. Our approach cannot find the many TEs, such as Alu elements, that have never encoded proteins.

3.7 Polintons and YR Retrotransposons in Human

Eukaryotic TEs have immense diversity, but can be classified into ∼7 major orders: LTR, LINE, and tyrosine-recombinase (YR) retrotransposons, and DDE transposons, cryptons, helitrons, and polintons [41]. Three of them (YR retrotransposons, cryptons, polintons) were long thought absent in mammals [42], [43]. Recently, relics of non-autonomous cryptons were detected in human [44], and one polinton-related element was found in chromosome 7 [45]. No YR retrotransposon relics have been found in human [44], apart from two genes encoding protein domains derived from YR retrotransposons [46].

Among our DNA/protein homologies found with pid=50 are 37 of YR retrotransposon proteins and 30 of polinton proteins. The polinton homologies have E-values as low as 2.2×10^{-67}: these alignments covered 93–1339 bases, and seemingly-random parts of several polinton proteins (e.g. POLB, ATPase, PY, integrase). These 30 polinton hits lie in ∼20 clusters in the genome, presumably corresponding to 20 ancestral polinton elements, one of which matches the known element on chromosome 7 [1].

We found two superfamilies of YR retrotransposon: 30 DIRS alignments (covering 107–602 bases) with minimum E-value 1.1×10^{-44}, and seven Ngaro alignments (120–443 bases) with minimum E-value 1.7×10^{-13}. Two of these Ngaro alignments are near each other in chromosome 9, indicating that an ancient Ngaro was split by insertion of a LINE (Fig. 9). Whole-genome alignments (Fig. 9, green) indicate that the LINE inserted in an ancestor of African apes, and the Ngaro in an ancestor of all amniotes (mammals and reptiles).

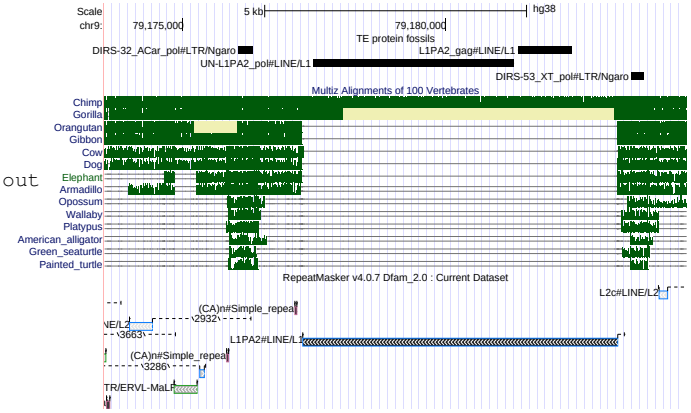

Fig. 9. Two ancient Ngaro fragments in human chromosome 9, separated by a recent L1 insertion. Each DNA-protein homology (black bars near the top) is labeled with the protein's name. Green: alignments of the human genome to other genomes [47]. Screen shot from http://genome.ucsc.edu [48].

4 Discussion

Our DNA-protein homology search method seems to be fast, specific, and highly sensitive for protein fossils. Specificity is achieved by (i) accurate calculation of significance (probability of chance similarity between random sequences), and (ii) suppressing non-homologous similarities of simple sequences. The latter is an under-appreciated problem [37], [38]: our solution could likely be improved, e.g. by tuning `tantan`'s parameters. It seems hard to be certain that a strong similarity is a true homology, but we can gain confidence from evidence other than sequence similarity.

2. The previous version of this study erroneously omitted reverse-strand `blastx` hits.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TCBB.2022.3177855, IEEE/ACM Transactions on Computational Biology and Bioinformatics

For example, the two Ngaro fragments in Fig. 9 are near each other, separated by a LINE, and lie in amniote-conserved DNA: this would all be a strong coincidence if they were spurious. Note that true homology may still lead to false-positive inference, e.g. some TEs carry fragments of non-related TEs [40]. With due care, our method enables discovery of more ancient and subtle fossils [49], such as the human polinton, DIRS and Ngaro elements found here.

Possible future improvements include better seeding, and using position-specific information on variability of a sequence family [23], [26]. Our significance calculation becomes inaccurate for short sequences, so a finite size correction would be useful [25]. Better parameter-fitting may be important, e.g. our parameters are a-rich and biased by many redundant L1 LINEs.

The sum-of-paths and significance paradigm is not specific to DNA-protein comparison, so could benefit homology search generally. A previous study made similar conjectures on significance of probabilistic homology scores [50]. That study considered the maximum-probability path and the sum over all paths; whereas we (following [25]) use a sum over *some* paths (emanating from an (i, j) point), with a balance condition (Equation 18). We suspect the conjectures in [50] may be too broad: e.g. one set of substitution and gap scores corresponds to a range of probability models with different values of t [24], but only one t can appear in the Gumbel formula (Equation 17).

Acknowledgments

We thank John Spouge, Yi-Kuo Yu, Osamu Gotoh, and the Frith and Asai labs' members for helpful advice.

References

[1] Y. Yao and M. C. Frith, "Improved DNA-versus-protein homology search for protein fossils," in *Int. Conf. Algorithms Comput. Biol.* Springer, 2021, pp. 146–158.

[2] T. J. Goodwin and R. T. Poulter, "The DIRS1 group of retrotransposons," *Mol. Biol. Evol.*, vol. 18, no. 11, pp. 2067–2082, 2001.

[3] E. J. Pritham and C. Feschotte, "Massive amplification of rolling-circle transposons in the lineage of the bat *Myotis lucifugus*," *Proc. Nat. Acad. Sci.*, vol. 104, no. 6, pp. 1895–1900, 2007.

[4] A. Katzourakis and R. J. Gifford, "Endogenous viral elements in animal genomes," *PLoS Genet.*, vol. 6, no. 11, p. e1001191, 2010.

[5] S. L. Sheetlin, Y. Park, M. C. Frith, and J. L. Spouge, "Frameshift alignment: statistics and post-genomic applications," *Bioinformatics*, vol. 30, no. 24, pp. 3575–3582, 2014.

[6] D. H. Huson, B. Albrecht, C. Bağcı, I. Bessarab, A. Gorska, D. Jolic, and R. B. Williams, "MEGAN-LR: new algorithms allow accurate binning and easy interactive exploration of metagenomic long reads and contigs," *Biol. Direct*, vol. 13, no. 1, p. 6, 2018.

[7] J. Raes and Y. Van de Peer, "Functional divergence of proteins through frameshift mutations," *Trends Genet.*, vol. 21, no. 8, pp. 428–431, 2005.

[8] R. Wang, J. Xiong, W. Wang, W. Miao, and A. Liang, "High frequency of +1 programmed ribosomal frameshifting in *Euplotes octocarinatus*," *Sci. Rep.*, vol. 6, p. 21139, 2016.

[9] H. Peltola, H. Söderlund, and E. Ukkonen, "Algorithms for the search of amino acid patterns in nucleic acid sequences," *Nucleic Acids Res.*, vol. 14, no. 1, pp. 99–107, 1986.

[10] D. States and D. Botstein, "Molecular sequence accuracy and the analysis of protein coding regions." *Proc. Nat. Acad. Sci.*, vol. 88, no. 13, p. 5518, 1991.

[11] X. Guan and E. C. Uberbacher, "Alignments of DNA and protein sequences containing frameshift errors," *Comput. Appl. Biosci.*, vol. 12, no. 1, pp. 31–40, Feb 1996.

[12] X. Huang and J. Zhang, "Methods for comparing a DNA sequence with a protein sequence," *Bioinformatics*, vol. 12, no. 6, pp. 497–506, 1996.

[13] Z. Zhang, W. R. Pearson, and W. Miller, "Aligning a DNA sequence with a protein sequence," *J. Comput. Biol.*, vol. 4, no. 3, pp. 339–349, 1997.

[14] W. R. Pearson, T. Wood, Z. Zhang, and W. Miller, "Comparison of DNA sequences with protein sequences," *Genomics*, vol. 46, no. 1, pp. 24–36, Nov. 1997.

[15] E. Halperin, S. Faigler, and R. Gill-More, "FramePlus: aligning DNA to protein sequences," *Bioinformatics*, vol. 15, no. 11, pp. 867–873, Nov. 1999.

[16] O. Gotoh, "Homology-based gene structure prediction: simplified matching algorithm using a translated codon (tron) and improved accuracy by allowing for long gaps," *Bioinformatics*, vol. 16, no. 3, pp. 190–202, Mar 2000.

[17] P. Ko, M. Narayanan, A. Kalyanaraman, and S. Aluru, "Space-conserving optimal DNA-protein alignment," in *Proc. 2004 IEEE Computational Systems Bioinf. Conf.*, pp. 80–88.

[18] M. Csűrös and I. Miklós, "Statistical alignment of retropseudogenes and their functional paralogs," *Mol. Biol. Evol.*, vol. 22, no. 12, pp. 2457–2471, 2005.

[19] F. Lysholm, "Highly improved homopolymer aware nucleotide-protein alignments with 454 data," *BMC Bioinf.*, vol. 13, no. 1, p. 230, 2012.

[20] P. L. Tzou, X. Huang, and R. W. Shafer, "NucAmino: a nucleotide to amino acid alignment optimized for virus gene sequences," *BMC Bioinf.*, vol. 18, no. 1, p. 138, 2017.

[21] L. Allison, C. S. Wallace, and C. N. Yee, "Finite-state models in the alignment of macromolecules," *J. Mol. Evol.*, vol. 35, no. 1, pp. 77–89, Jul 1992.

[22] S. R. Eddy, "A new generation of homology search tools based on probabilistic inference," *Genome Inform.*, vol. 23, no. 1, pp. 205–211, Oct 2009.

[23] R. Durbin, S. Eddy, A. Krogh, and G. Mitchison, *Biological sequence analysis: probabilistic models of proteins and nucleic acids*. Cambridge University Press, 1998.

[24] M. C. Frith, "How sequence alignment scores correspond to probability models," *Bioinformatics*, vol. 36, no. 2, pp. 408–415, 2020.

[25] Y. K. Yu and T. Hwa, "Statistical significance of probabilistic sequence alignment and related local hidden Markov models," *J. Comput. Biol.*, vol. 8, no. 3, pp. 249–282, 2001.

[26] Y.-K. Yu, R. Bundschuh, and T. Hwa, "Hybrid alignment: high-performance with universal statistics," *Bioinformatics*, vol. 18, no. 6, pp. 864–872, 2002.

[27] S. F. Altschul, T. L. Madden, A. A. Schäffer, J. Zhang, Z. Zhang, W. Miller, and D. J. Lipman, "Gapped BLAST and PSI-BLAST: a new generation of protein database search programs," *Nucleic Acids Res.*, vol. 25, no. 17, pp. 3389–3402, 1997.

[28] M. Roytberg, A. Gambin, L. Noé, S. Lasota, E. Furletova, E. Szczurek, and G. Kucherov, "On subset seeds for protein alignment," *IEEE/ACM Trans. Comput. Biol. Bioinf.*, vol. 6, no. 3, pp. 483–494, 2009.

[29] M. Steinegger and J. Söding, "MMseqs2 enables sensitive protein sequence searching for the analysis of massive data sets," *Nature Biotechnol.*, vol. 35, no. 11, pp. 1026–1028, 2017.

[30] S. M. Kiełbasa, R. Wan, K. Sato, P. Horton, and M. C. Frith, "Adaptive seeds tame genomic sequence comparison," *Genome Res.*, vol. 21, no. 3, pp. 487–493, 2011.

[31] S. Henikoff and J. G. Henikoff, "Amino acid substitution matrices from protein blocks," *Proc. Nat. Acad. Sci.*, vol. 89, no. 22, pp. 10 915–10 919, 1992.

[32] H. Y. Lam, E. Khurana, G. Fang, P. Cayting, N. Carriero, K.-H. Cheung, and M. B. Gerstein, "Pseudofam: the pseudogene families database," *Nucleic Acids Res.*, vol. 37, no. suppl_1, pp. D738–D743, 2009.

[33] A. Smit, R. Hubley, and P. Green, "RepeatMasker open-4.0," http://www.repeatmasker.org, 2013–2015.

[34] A. Tanay and E. D. Siggia, "Sequence context affects the rate of short insertions and deletions in flies and primates," *Genome Biol.*, vol. 9, no. 2, p. R37, 2008.

[35] A. F. Smit, G. Tóth, A. D. Riggs, and J. Jurka, "Ancestral, mammalian-wide subfamilies of LINE-1 repetitive sequences," *J. Mol. Biol.*, vol. 246, no. 3, pp. 401–417, 1995.

[36] S. F. Altschul, "A protein alignment scoring system sensitive at all evolutionary distances," *J. Mol. Evol.*, vol. 36, no. 3, pp. 290–300, 1993.
[37] M. C. Frith, "A new repeat-masking method enables specific detection of homologous sequences," *Nucleic Acids Res.*, vol. 39, no. 4, pp. e23–e23, 2011.
[38] ——, "Gentle masking of low-complexity sequences improves homology search," *PLoS One*, vol. 6, no. 12, p. e28819, 2011.
[39] The UniProt Consortium, "UniProt: the universal protein knowledgebase in 2021," *Nucleic Acids Res.*, vol. 49, no. D1, pp. D480–D489, 2021.
[40] J. Storer, R. Hubley, J. Rosen, T. J. Wheeler, and A. F. Smit, "The Dfam community resource of transposable element families, sequence models, and genome annotations," *Mobile DNA*, vol. 12, no. 1, pp. 1–14, 2021.
[41] J. N. Wells and C. Feschotte, "A field guide to eukaryotic transposable elements," *Annu. Rev. Genet.*, vol. 54, pp. 539–561, 2020.
[42] R. T. Poulter and M. I. Butler, "Tyrosine recombinase retrotransposons and transposons," *Mobile DNA III*, pp. 1271–1291, 2015.
[43] S. Campbell, A. Aswad, and A. Katzourakis, "Disentangling the origins of virophages and polintons," *Current Opinion Virol.*, vol. 25, pp. 59–65, 2017.
[44] K. K. Kojima, "Human transposable elements in Repbase: genomic footprints from fish to humans," *Mobile DNA*, vol. 9, no. 1, pp. 1–14, 2018.
[45] G. J. Starrett, M. J. Tisza, N. L. Welch, A. K. Belford, A. Peretti, D. V. Pastrana, and C. B. Buck, "Adintoviruses: A proposed animal-tropic family of midsize eukaryotic linear dsDNA (MELD) viruses," *Virus Evol.*, vol. 7, no. 1, p. veaa055, 2021.
[46] F. Abascal, M. L. Tress, and A. Valencia, "Alternative splicing and co-option of transposable elements: the case of TMPO/LAP2α and ZNF451 in mammals," *Bioinformatics*, vol. 31, no. 14, pp. 2257–2261, 2015.
[47] R. S. Harris, "Improved pairwise alignment of genomic DNA," Ph.D. dissertation, The Pennsylvania State University, 2007.
[48] W. J. Kent, C. W. Sugnet, T. S. Furey, K. M. Roskin, T. H. Pringle, A. M. Zahler, and D. Haussler, "The human genome browser at UCSC," *Genome Res.*, vol. 12, no. 6, pp. 996–1006, 2002.
[49] M. C. Frith, "Paleozoic protein fossils illuminate the evolution of vertebrate genomes and transposable elements," *Mol. Biol. Evol.*, vol. 39, no. 4, p. msac068, 2022.
[50] S. R. Eddy, "A probabilistic model of local sequence alignment that simplifies statistical significance estimation," *PLoS Comput. Biol.*, vol. 4, no. 5, p. e1000069, 2008.