Alignment-free sequence comparison with spaced k-mers

Marcus Boden[1], Martin Schöneich[1], Sebastian Horwege[1], Sebastian Lindner[1], Chris Leimeister[1], and Burkhard Morgenstern[1]

1 **University of Göttingen, Institute of Microbiology and Genetics, Department of Bioinformatics, Goldschmidtstr. 1, 37073 Göttingen, Germany**
bmorgen@gwdg.de

Abstract

Alignment-free methods are increasingly used for genome analysis and phylogeny reconstruction since they circumvent various difficulties of traditional approaches that rely on multiple sequence alignments. In particular, they are much faster than alignment-based methods. Most alignment-free approaches work by analyzing the k-mer composition of sequences. In this paper, we propose to use 'spaced k-mers', *i.e.* patterns of deterministic and 'don't care' positions instead of contiguous k-mers. Using simulated and real-world sequence data, we demonstrate that this approach produces better phylogenetic trees than alignment-free methods that rely on contiguous k-mers. In addition, distances calculated with spaced k-mers appear to be statistically more stable than distances based on contiguous k-mers.

1998 ACM Subject Classification J.3

Keywords and phrases Alignment-free sequence comparison, phylogeny reconstruction

Digital Object Identifier 10.4230/OASIcs.GCB.2012.21

1 Introduction

Traditional methods for comparative sequence analysis and phylogeny reconstruction are based on pairwise and multiple sequence alignment, see *e.g.* [14, 29] for an overview. During the last years, however, a large number of *alignment-free* methods have been proposed for sequence comparison, see [38] for a review. The main advantage of these methods is that they are much faster than alignment-based approaches. While aligning two sequences takes time proportional to the product of the sequence lengths, most alignment-free approaches work in *linear* time.

Consequently, alignment-free methods are increasingly used for genome comparison, in particular for genome-based phylogeny reconstruction [17, 10, 24]. In addition to being faster, alignment-free approaches circumvent some well-known problems such as finding ortholog genes [32] or aligning large genomic sequences [4]. Another advantage of alignment-free genome comparison is that these approaches can work with unassembled reads [34] and are not sensitive to genome rearrangements. Alignment-free methods have also been used for database searching [40] and to construct *guide trees* as a prerequisite for progressive multiple sequence alignment [22, 11, 3]. Here, alignment-free sequence comparison could crucially speed-up progressive multiple alignment, since the run time for alignment-based phylogeny reconstruction becomes prohibitive if the number of input sequences exceeds a few hundred or so.

Most alignment-free methods that have been proposed so far rely on some sort of k-mer statistics. That is, for a fixed integer k, they consider the (relative) frequencies of all possible

German Conference on Bioinformatics 2013 (GCB'13).
Editors: T. Beißbarth, M. Kollmar, A. Leha, B. Morgenstern, A.-K. Schultz, S. Waack, E. Wingender; pp. 21–31

OpenAccess Series in Informatics
Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany

k-mers for each of the input sequences and then define some distance measure based on these frequency vectors [20, 7, 39, 15]. Standard distance-based methods such as *UPGMA* [33] or *NeighbourJoining* [31] can then be used to calculate phylogenetic trees from the resulting distance matrices. Other approaches consider the local *context* of each sequence position in terms of overlapping k-mers [8]. Some alignment-free methods do not rely on a fixed k but allow for matches of variable length [37, 19, 9]. However, these methods are still based, in one way or the other, on *contiguous* exact matches.

Exact pattern matching is used in many areas of biological sequence comparison, see for example [16]. A traditional application of k-mer comparison in bioinformatics is *database searching*. Fast alignment programs such as *FASTA* [27] and *BLAST* [1] originally relied on identifying word matches of a fixed length. Such word matches, that are referred to as *seeds*, can be rapidly found in an initial phase of the algorithm, in a second phase these 'seeds' are then extended into both directions by slower but more accurate methods for local sequence alignment. The size of seeds is a trade-off between *sensitivity* and *speed*: short words are more sensitive, since more matches are found where local alignments are triggered and evaluated. This increases, however, the running time of these programs, since the local alignment step is the most time-consuming part of the algorithm. Longer word lengths lead to an increase in speed, but result in lower sensitivity.

In a pioneering paper, Ma *et al.* proposed to use *spaced seeds* instead of *contiguous* word matches as the first step in database searching [28]. That is, they proposed to use fixed patterns of *match* and *don't care* positions and to search for word pairs matching at the pre-defined *match* positions, with possible mismatches at the *don't care* positions. Their approach is implemented in the program *PatternHunter* [25]. The main advantage of *spaced seeds* is that hits at different positions are statistically less correlated with each other than contiguous word matches are. Also *spaced seeds* are better able to identify homolog sequence regions in the presence of mismatches. Ma *et al.* showed that for database searching, *spaced seeds* are superior to contiguous word matches in terms of *sensitivity* and *speed*. For this reason, the original contiguous *seeds* have been largely replaced by *spaced seeds* in rapid database search programs. Similarly, Burkhardt and Kärkkäinen [6] used *gapped q-grams* instead of contiguous $q-$grams (q-mers) in a filtering step for the well-studied k-differences problem.

In this paper, we propose to use *spaced k-mers*, *i.e.* k-mers with *don't care* characters at fixed, pre-defined positions, as a basis for alignment-free sequence comparison. Note that this approach is quite different form the above mentioned spaced-seeds approach to database searching: instead of using spaced k-mers to trigger *local alignments* for homology searching, we want to estimate the *global* degree of similarity between sequences by comparing their spaced k-mer composition. To do so, we use a generic distance measure on DNA and protein sequences based on their spaced k-mer frequencies. We use these distances to construct phylogenetic trees for simulated and real-world sequences, and we compare these results with trees constructed by the same method, but with *contiguous* k-mers that are traditionally used by alignment-free methods for sequence comparison. Our study shows that, for phylogeny reconstruction, *spaced k-mers* often outperform *contiguous* k-mers. In addition, we found that the *variance* of distances values calculated from spaced k-mers is lower than the variance calculated with contiguous k-mers.

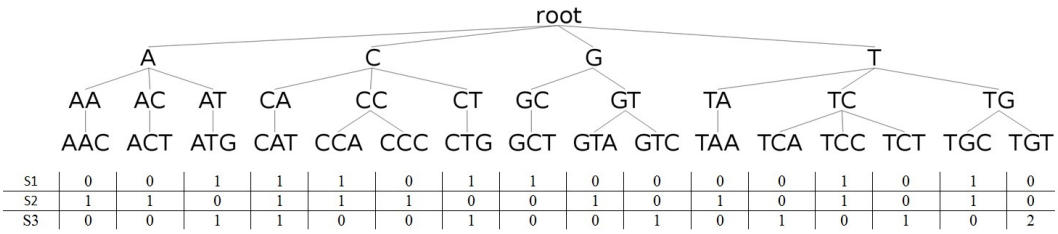

Figure 1 Tree representing the frequencies of *spaced k-mers* ($k = 3$) for an underlying pattern $P = X0X0X$ in a set of three sequences $S1 = ATTCGCCATTG$, $S2 = GTTCACACCATT$, $S3 = GTTCCATTGGTT$. For example, the spaced 3-mer corresponding to the word TGT and pattern P occurs twice in sequence S_3 and does not occur in sequences S_1 and S_2.

2 Calculating trees with spaced k-mers

As usual, for an alphabet $\mathcal{A}$ and $l \in \mathbb{N}$, $\mathcal{A}^l$ denotes the set of all sequences over $\mathcal{A}$ with length l. $S[i]$ denotes the i-th character of a sequence S. In our study, the alphabet $\mathcal{A}$ represents the set of nucleotides or amino acids, respectively. In analogy to the terminology introduced by Ma *et al.*, we define a *spaced k-mer* as a sequence $P \in \{0, X\}^l$, *i.e.* a sequence of '0' and 'X' characters (the underlying *pattern*), such that there are exactly k positions i in P with $P[i] = 1$, together with a finite sequence $w \in \mathcal{A}^k$ (the underlying *word*) with $l, k \in \mathbb{N}$ and $k < l$. In addition, we require that $P[1] = P[l] = X$ holds, *i.e.* the first and the last characters in P must be 'X'. The 'X' positions in the pattern P denote *match* positions while the '0' positions are the *don't care* positions. We call l the *length* of the spaced k-mer and k its *weight*. (One could also include the case $k = l$, but in order to distinguish 'spaced k-mers' from *words* or *k-mers* in the usual sense, we require k to be smaller than l.) We use the notation *length* and *weight* for the underlying pattern P accordingly.

Let α be a spaced k-mer with pattern P, word w, weight k and length l such that $1 = p_1, \ldots, p_k = l$ denote the positions of the 'X' characters in P. We say that α occurs in a sequence S at position i if $S[i + p_j - 1] = w[j]$ for all $1 \leq j \leq k$. For example, the spaced k-mer α consisting of the pattern $P = XX00X$ and the word $w = AGT$ occurs in the sequence $S = GGAGCTTCAGGATCC$ at positions 3 and 9.

In order to define a *distance function* on a set of sequences $S_1, \ldots, S_N$ over $\mathcal{A}$, we consider a fixed pattern P with length l and weight k. We then calculate for each sequence S_i the *relative frequencies* of all possible spaced k-mers that involve our pattern P – relative to the sequence length –, and we represent each sequence S_i by the $|\mathcal{A}|^k$-dimensional vector of the relative frequencies of the *spaced* k-mers with respect to the pattern P - similarly as sequences are represented as vectors of (relative) k-mer frequencies in standard alignment-free approaches.

The spaced k-mer frequencies of the input sequences $S_1, \ldots, S_N$ can be conveniently stored in a tree, as for the usual (contiguous) k-mers, see Figure 1. It is straight forward to calculate the spaced k-mer composition of a sequence of length n in $O(n \times k)$ time with a 'naive' algorithm. For *contiguous* k-mers, this can be reduced to $O(n)$ time, *e.g.* using a *rolling hash* approach [21]. This approach can be easily generalized to spaced k-mers by first considering (contiguous) l-mers and then correcting for the *don't-care* positions. This way, the spaced-k-mer frequencies can be calculated in $O(n \times d)$ where $d = l - k$ is the number of *don't-care* positions in the pattern P.

Once the spaced k-mer compositions are calculated for all input sequences, we proceed

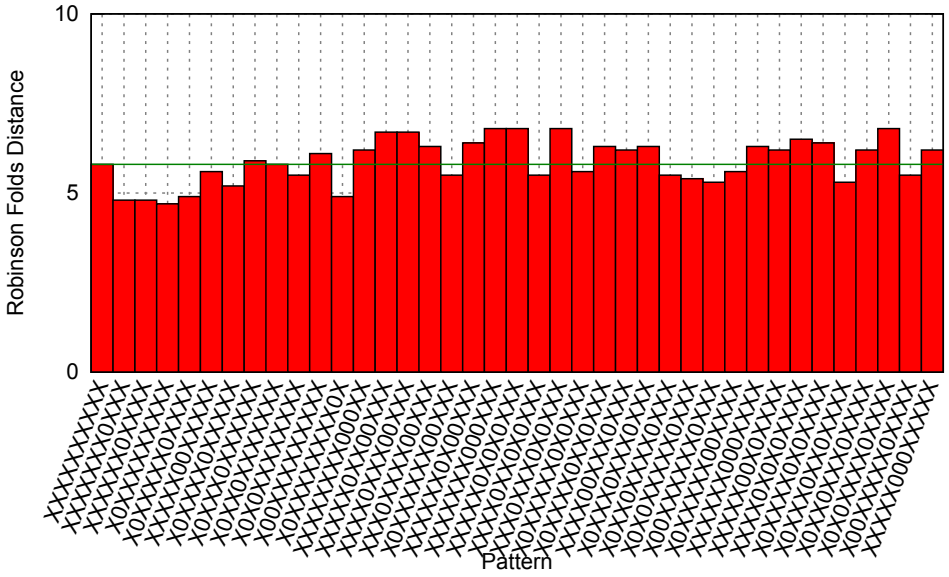

Figure 2 Performance of different patterns of weight $w = 10$ on simulated DNA sequences. We used *Rose* [35] to create 40 sequence sets, each containing 100 simulated DNA sequences of length 20,000. Tree topologies calculated with spaced and contiguous 10-mers were compared to the respective reference trees from *Rose* using the *Robinson-Foulds* metric. The horizontal green line is the RF distance obtained with the contiguous 10-mer.

as other alignment-free methods: we define the *distance* between sequences S_i and S_j as the distances of the corresponding frequency vectors. In the present study, we used the *Jensen-Shannon* distance [26] as this distance measure led to better results than alternative distances. Finally, we construct unrooted trees from the calculated distance matrix using the *Neighbour-Joining* program [31] from the *PHYLIP* package [13].

3 Benchmark data

To evaluate the k-mer and pattern-based methods, we used four different categories of benchmark data, consisting of simulated and real-world sets of DNA and protein sequences. For each sequence set, we used a *reference* tree that we consider to be reliable. We evaluated the methods under consideration by comparing the trees that they produce to the respective reference trees in our benchmark sequence sets.

To generate simulated sequences, we used the program *Rose* [35]. *Rose* mimics molecular evolution by producing a set of sequences along an evolutionary tree, starting with a common ancestral sequence. Mutations are randomly incorporated according to a pre-defined stochastic model of molecular evolution. As a result, one obtains a set of sequences with known evolutionary history that can be used to benchmark methods for phylogeny reconstruction. The parameter *relatedness* determines the *average* evolutionary distance, measured in *PAM* units, between the sequences produced by *Rose*.

For DNA sequence comparison, we created 40 sets of sequences, each of which containing 20 sequences of length 20,000 using *Rose* with a *relatedness* value of 70. In addition, to evaluate the *variance* of the distances defined with contiguous and spaced k-mers, we created pairs of DNA sequences with *Rose* using different values for *relatedness* (see below for details).

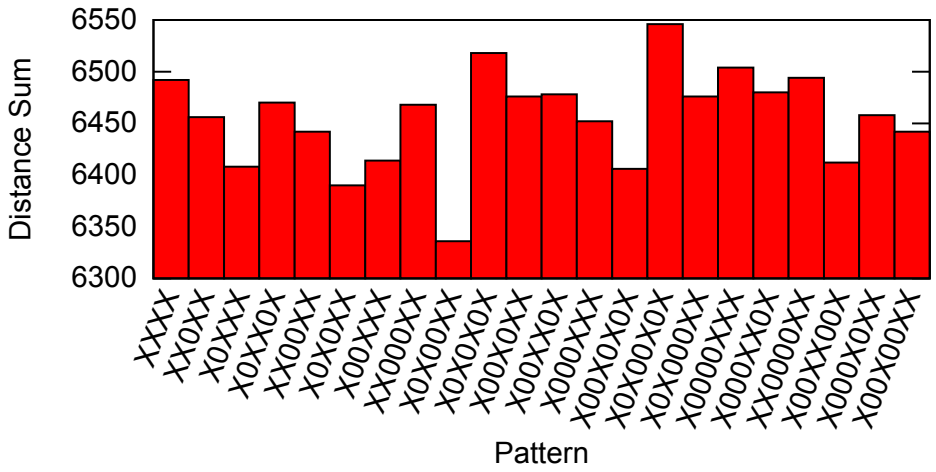

Figure 3 Performance of different patterns of weight $w = 4$ on BAliBASE. The total sum of Robinson-Foulds distances over the BAliBASE is shown for spaced k-mers with weight = 4 and length between 4 and 8.

As real-world DNA sequences, we used a set of 27 primate mitochondrial genomes that has already been used by Haubold *et al.* [18] as benchmark data for alignment-free sequence comparison.

To benchmark the various phylogeny-reconstruction methods on protein sequences, we simulated sets of 100 protein sequences with *Rose*, each sequence with a length of 300. Here, we used the *Rose* default values, together with *relatedness* values of 200, 350, 450 and 550. As real-world protein sequences, we used the *BAliBASE* benchmark database, a standard benchmark database for multiple alignment [2]. Since BAliBASE contains no information about the underlying phylogenetic trees, we applied *Maxium Likelihood* [12] approach to the reference multiple alignments from BAliBASE, and we used the resulting trees as the reference trees in our program evaluation.

To evaluate the various alignment-free methods described in the previous section and to compare them to the classical alignment-based approach, we compared the tree topologies generated by these methods to the topologies of the respective reference trees using the *Robinson-Foulds (RF)* metric [30].

4 Test results

To each category of benchmark data, we first applied the above outlined approach using *contiguous* k-mers with various values for k. This way, we identified for each category of benchmark sequences the k-mer length that gives the best result regarding the RF distances to the respective reference trees. For the this value of k, we then generated patterns with weight k and with varying lengths. Since the number of possible patterns grows rapidly with k, we randomly selected patterns for those data sets where the optimal k was too large to test all possible patterns exhaustively.

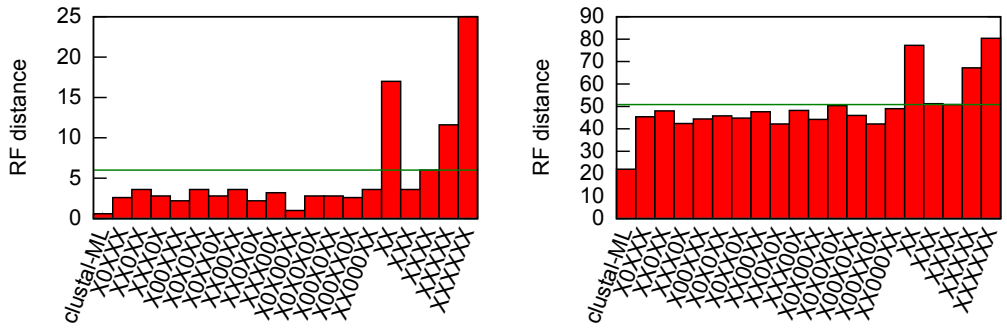

Figure 4 Performance of different approaches on protein sequences, simulated with *Rose* using *relatedness* values of 200 (left) and 350 (right). *Clustal W* and *Maximum Likelihood*

4.1 Genomic sequences

For the simulated DNA sequences of length 20,000, we found that k-mers with $k = 10$ gave the best results, with an average RF distance of 5 to the respective reference trees. Thus, we created patterns with a *weight* of 10; we varied their lengths between 11 and 13. That is, each pattern contained 10 'match' positions and between one and three 'don't care' positions. Overall, we used 32 different patterns of weight $w = 10$, namely 3 patterns of length 11, 8 patterns of length 12 and 21 patterns of length 13; the patterns were randomly selected. The test results with these patterns are shown in Figure 2. *All* patterns of length 11 and most patterns of length 12 produced better results than the corresponding approach with the contiguous 10-mer.

For the primate mitochondrial genomes, we obtained the best results with k-mers of length $k = 8$, so we generated patterns P with a weight of 8 and with varying lengths. In contrast to the simulated DNA sequences, our pattern-defined spaced 8-mers performed slightly worse than the contiguous 8-mer. The RF distance to the reference tree was 4 for the contiguous 8-mer, while the average RF distance for our spaced 8-mers was 4.68, with a range between 2 and 10.

4.2 Protein sequences

On our simulated and real-world protein sequences, contiguous k-mers with $k = 4$ produced the best results, with the exception of simulated *Rose* sequences with *relatedness* of 200 where $k = 3$ performed better, see Figures 3,4, 5. For protein sequences, we therefore generated patterns with weight $k = 4$ and varying lengths.

As shown in Figure 3, 4 and 5, *all* test runs with spaced 4-mers of length $l \leq 7$ gave better results than the corresponding test runs with contiguous 4-mers – with the notable exception of the *periodic* pattern $P = X0X0X0X$ which performed similar to the contiguous k-mers or even worse. On the simulated sequence sets with a *relatedness* value of 550, however, the difference was less clear. On these distantly related protein sequences, contiguous and spaced 4-mers performed almost equally.

We also applied the classical approach to phylogeny reconstruction to our sequence sets by calculating multiple alignments with *Clustal W* [36] and then applying the *Maximum Likelihood (ML)* software from the *PHYLIP* package [13]. The results of these test runs for the simulated protein sequences are inluded in Figures 4 and 5. On *BAliBASE*, the total RF

distance between the *Clustal/ML* trees and the reference trees was 4,478. On all sets of real and simulated protein sequences, this classical approach produced better results than our alignment-free methods, except for the simulated protein sequences with *relatedness* of 550 where both approaches performed comparably.

4.3 Variance of sequence distances calculated with spaced and contiguous k-mers

Finally, we investigated the *variance* of the distances that we used for the various alignment-free methods in this study. To do so, we simulated pairs of DNA sequences with pre-defined evolutionary distances with *Rose*, using *relatedness* values between 2 and 51. For each *relatedness* value we created 200 pairs of sequences of length 2,500 each. We then calculated the distance for each sequence pair as described in section 2 with a value of $k = 5$ which performed best on DNA sequences of this length. We measured the *variance* of these distance values for each value of *relatedness*, for all 200 sequence pairs and for each 5-mer. It turned out that for the contiguous pattern $XXXXX$ and for the periodic pattern $X0X0X0X0X$, the variance was considerably higher than for the non-periodic patterns. The results are summarized in Figure 6.

5 Discussion

The k-mer composition of DNA and protein sequences is frequently used to analyse evolutionary relationships and to construct phylogenetic trees. A certain disadvantage of this approach is the fact that k-mer occurrences at different sequence positions are far from independent from each other. For this reason, some authors corrected the k-mer statistics of sequences for the dependency of overlapping k-mer matches, *e.g.* Göke *et al.* [15]. For the same reason, k-mer matches have been replaced in homology searching by so-called *spaced seeds* where non-periodic patterns of 'match' and 'don't care' positions are used instead of contiguous word matches [28]. Motivated by this approach, we propose to use *spaced k-mers* instead of the traditionally used *contiguous k-mers* to define distances between sequences and to construct phylogenetic trees. While, under an *i.i.d.* Bernoulli model, the *expected* number of occurrences of a *spaced* k-mer in a random sequence is approximately the same as for a *contiguous* k-mer, occurrences of a spaced k-mers at different sequence positions are less dependent, provided that a non-periodic underlying pattern P is used.

To compare *spaced* and *contiguous* k-mers, we implemented a generic approach to phylogeny reconstruction based on the (spaced) k-mer composition of sequences and evaluated the resulting trees on various types of benchmark data. Figures 3, 4 and 5 show that distance matrices based on *spaced* 4-mer frequencies in protein sequences led to consistently better phylogenetic trees than the same approach with *contiguous* 4-mers. This is similar for simulated DNA sequences as shown in Figure 2, although here improvements could only be achieved with shorter patterns containing only up to two *don't care* positions. If the number of don't care positions - and thus the length of the pattern - was further increased, the results deteriorated. This is probably due to the frequency of insertions and deletions in *Rose* which make longer gap-free matches in 'homologous' regions less likely. Not surprisingly, the conventional approach for phylogeny reconstruction using *maximum likelihood* and *multiple alignments* performed better than the alignment-free approaches that we tested. Nevertheless, for very distantly related simulated protein sequences, the performance of k-mer and alignment-based phylogeny methods converged.

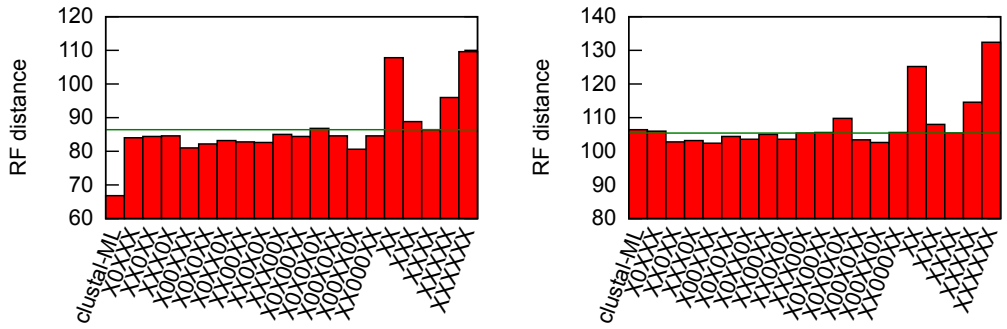

Figure 5 Performance of different approaches on protein sequences, simulated with *Rose* using *relatedness* values of 450 (left) and 550 (right).

As mentioned, a main advantage of spaced k-mers is that matches at different sequence positions are statistically less dependent on each other than matches of contiguous words are - as long as the underlying pattern is non-periodic (or 'irregular'). Distance measures using spaced k-mers can therefore be expected to be more stable than distances based on contiguous words. Figure 6 shows, for example, that the statistical variance of distances based on the contiguous pattern $XXXXX$ and the periodic ('regular') pattern $X0X0X0X0X$ is higher than for non-periodic patterns with the same number of 'match' positions.

Correspondingly, for the real-world and simulated protein sequences, *'non-regular'* patterns for which matches at different sequence positions have less overlap, often performed better than 'regular' patterns where matches at different positions are statistically more dependent. For *BAliBASE* and for the *Rose* sequence sets of relatedness 200 and 350, for example, the 'irregular' pattern $X0X00XX$ performed clearly better than the more 'regular' pattern $XX000XX$. Spaced 4-mers performed always better than contiguous words on these sequence sets – except for the periodic pattern $X0X0X0X$. Spaced k-mers with this periodic pattern often performed similar or even worse than the contiguous 4-mer.

A crucial question in our approach is how to select good patterns P. One approach would be to minimize the dependency of spaced k-mer matches at different sequence positions. First test runs indicate that the summed correlation coefficients for matches at different positions may be an indicator of how good a pattern is at distinguishing random similarities from true homologies.

The statistical properties of *spaced seeds* used in database searching have been studied extensively during the last ten years, and efforts have been made to identify optimal spaced seeds, see for example [23, 5]. Note, however, that these questions are quite different from the questions that are relevant in our approach. In database searching, one is interested in the probability of finding (at least) one hit between sequences with a certain degree of similarity, to trigger a local alignment. This probability determines the *sensitivity* of a spaced seed and has to be balanced against the number of random hits that slow down the program. By contrast, with our spaced k-mer approach, we want to study the *global* degree of similarity between two sequences by comparing their (spaced) k-mer compositions. Here, we are interested in the expected number of matching spaced k-mers and its *variance* in homologous vs. unrelated sequences. We are planning to study the statistical behaviour of spaced and contiguous k-mer matches in more detail to identify optimal patterns for alignment-free sequence comparison and phylogeny reconstruction.

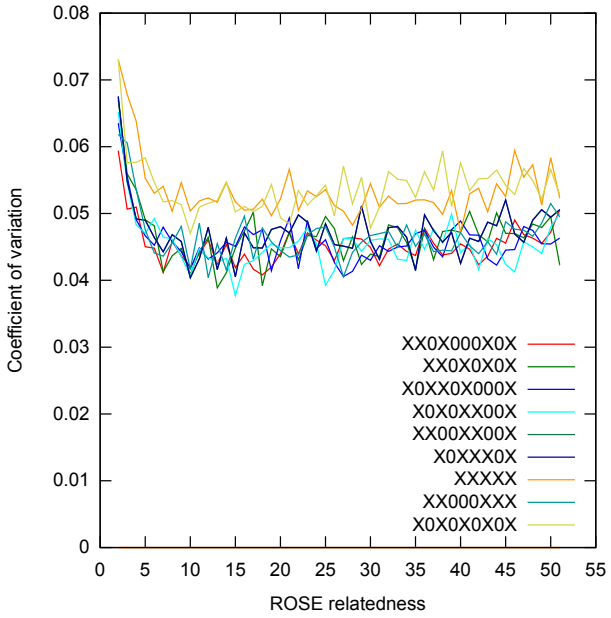

Figure 6 Variation coefficients for the distance values calculated with various *spaced* and *contiguous* 5-mers. We used *Rose* to simulate pairs of DNA sequences with different values of *relatedness*, *i.e.* evolutionary distances. For each value of *relatedness* between 2 and 51, we we generated 200 sequence pairs of length 2,500 and estimated the pairwise distances with the different k-mer based approaches. For each approach, the graphic shows the *variation coefficient* for the resulting 200 distance values.

Acknowledgements We want to thank Thomas Lingner, Anja Sturm, Anirban Mukhopadhyay and Susana Vinga for useful comments and discussions.

References

1 Stephen F. Altschul, Warren Gish, Webb Miller, Eugene M. Myers, and David J. Lipman. Basic local alignment search tool. *J. Mol. Biol.*, 215:403–410, 1990.

2 Anne Bahr, Julie Dawn Thompson, Jean-Claude Thierry, and Olivier Poch. BAliBASE (Benchmark Alignment dataBASE): enhancements for repeats, transmembrane sequences and circular permutations. *Nuc. Acids Research*, 29:323–326, 2001.

3 Gordon Blackshields, Fabian Sievers, Weifeng Shi, Andreas Wilm, and Desmond Higgins. Sequence embedding for fast construction of guide trees for multiple sequence alignment. *Algorithms for Molecular Biology*, 5:21, 2010.

4 Michael Brudno, Alexander Poliakov, Simon Minovitsky, Igor Ratnere, and Inna Dubchak. Multiple whole genome alignments and novel biomedical applications at the vista portal. *Nucleic Acids Research*, 35(Web-Server-Issue):669–674, 2007.

5 Jeremy Buhler, Uri Keich, and Yanni Sun. Designing seeds for similarity search in genomic dna. *J. Comput. Syst. Sci.*, 70:342–363, 2005.

6 Stefan Burkhardt and Juha Kärkkäinen. Better filtering with gapped q-grams. *Fundam. Inf.*, 56:51–70, 2003.

7 Benny Chor, David Horn, Yaron Levy, Nick Goldman, and Tim Massingham. Genomic DNA k-mer spectra: models and modalities. *Genome Biology*, 10, 2009.

8 Gilles Didier. Caractérisation des n-écritures et application à'l étude des suites de complexité ultimement n + cst. *Theor. Comp. Sci*, 215:31–49, 1999.

9 Gilles Didier, Eduardo Corel, Ivan Laprevotte, Alex Grossmann, and Claudine Landés-Devauchelle. Variable length local decoding and alignment-free sequence comparison. *Theoretical Computer Science*, 462:1 – 11, 2012.

10 Gilles Didier, Laurent Debomy, Maude Pupin, Ming Zhang, Alexander Grossmann, Claudine Devauchelle, and Ivan Laprevotte. Comparing sequences without using alignments: application to HIV/SIV subtyping. *BMC Bioinformatics*, 8:1, 2007.

11 Robert C. Edgar. MUSCLE: Multiple sequence alignment with high score accuracy and high throughput. *Nuc. Acids. Res.*, 32:1792–1797, 2004.

12 Joseph Felsenstein. Evolutionary trees from DNA sequences:a maximum likelihood approach. *Journal of Molecular Evolution*, 17:368–376, 1981.

13 Joseph Felsenstein. PHYLIP - Phylogeny Inference Package (Version 3.2). *Cladistics*, 5:164–166, 1989.

14 Joseph Felsenstein. *Inferring Phylogenies*. Sinauer Associates, Sunderland, MA, USA, 2003.

15 Jonathan Göke, Marcel H. Schulz, Julia Lasserre, and Martin Vingron. Estimation of pairwise sequence similarity of mammalian enhancers with word neighbourhood counts. *Bioinformatics*, 28(5):656–663, 2012.

16 Dan Gusfield. *Algorithms on Strings, Trees, and Sequences: Computer Science and Computational Biology*. Cambridge University Press, Cambridge, UK, 1997.

17 Klas Hatje and Martin Kollmar. A phylogenetic analysis of the brassicales clade based on an alignment-free sequence comparison method. *Front Plant Sci*, 3:192, 2012.

18 Bernhard Haubold, Peter Pfaffelhuber, Mirjana Domazet-Loso, and Thomas Wiehe. Estimating mutation distances from unaligned genomes. *Journal of Computational Biology*, 16(10):1487–1500, 2009.

19 Bernhard Haubold, Nora Pierstorff, Friedrich Möller, and Thomas Wiehe. Genome comparison without alignment using shortest unique substrings. *BMC Bioinformatics*, 6:123, 2005.

20 Michael Höhl, Isidore Rigoutsos, and Mark A. Ragan. Pattern-based phylogenetic distance estimation and tree reconstruction. *Evolutionary Bioinformatics Online*, 2:359–375, 2006.

21 Richard M. Karp and Michael O. Rabin. Efficient randomized pattern-matching algorithms. *IBM Journal of Research and Development*, 31(2):249–260, 1987.

22 Kazutaka Katoh, Kazuharu Misawa, Kei-Ichi Kuma, and Takashi Miyata. MAFFT: a novel method for rapid multiple sequence alignment based on fast fourier transform. *Nuc. Acids Research*, 30:3059 – 3066, 2002.

23 Uri Keich, Ming Li, Bin Ma, and John Tromp. On spaced seeds for similarity search. *Discrete Applied Mathematics*, 138:253 – 263, 2004.

24 Pandurang Kolekar, Mohan Kale, and Urmila Kulkarni-Kale. Alignment-free distance measure based on return time distribution for sequence analysis: Applications to clustering, molecular phylogeny and subtyping. *Molecular Phylogenetics and Evolution*, 65(2):510 – 522, 2012.

25 Ming Li, Bin Ma, Derek Kisman, and John Tromp. PatternHunter II: Highly sensitive and fast homology search. *Genome Informatics*, 14:164–175, 2003.

26 Jianhua Lin. Divergence measures based on the shannon entropy. *IEEE Transactions on Information theory*, 37:145–151, 1991.

27 David P Lipman and William R Pearson. Rapid and sensitive protein similarity searches. *Science*, 227(4693):1435–1441, 1985.

28 Bin Ma, John Tromp, and Ming Li. PatternHunter: faster and more sensitive homology search. *Bioinformatics*, 18:440–445, 2002.

29 David A. Morrison. Multiple sequence alignment for phylogenetic purposes. *Australian Systematic Botany*, 19:479–539, 2006.

30 DF Robinson and LR Foulds. Comparison of phylogenetic trees. *Mathematical Biosciences*, 53:131–147, 1981.

31 Naruya Saitou and Masatoshi Nei. The neighbor-joining method: a new method for reconstructing phylogenetic trees. *Molecular Biology and Evolution*, 4:406–425, 1987.

32 Fabian Schreiber, Kerstin Pick, Dirk Erpenbeck, Gert Wörheide, and Burkhard Morgenstern. Orthoselect: a protocol for selecting orthologous groups in phylogenomics. *BMC Bioinformatics*, 10:219, 2009.

33 Robert R Sokal and Charles D Michener. A statistical method for evaluating systematic relationships. *University of Kansas Science Bulletin*, 38:1409–1438, 1958.

34 Kai Song, Jie Ren, Zhiyuan Zhai, Xuemei Liu, Minghua Deng, and Fengzhu Sun. Alignment-free sequence comparison based on next generation sequencing reads: extended abstract. In *Proceedings of the 16th Annual international conference on Research in Computational Molecular Biology*, RECOMB'12, pages 272–285, Berlin, Heidelberg, 2012. Springer-Verlag.

35 Jens Stoye, Dirk Evers, and Folker Meyer. Rose: Generating sequence families. *Bioinformatics*, 14:157–163, 1998.

36 Julie D. Thompson, Desmond G. Higgins, and Toby J. Gibson. CLUSTAL W: improving the sensitivity of progressive multiple sequence alignment through sequence weighting, position-specific gap penalties and weight matrix choice. *Nucleic Acids Research*, 22:4673–4680, 1994.

37 Igor Ulitsky, David Burstein, Tamir Tuller, and Benny Chor. The average common substring approach to phylogenomic reconstruction. *Journal of Computational Biology*, 13:336–350, 2006.

38 Susana Vinga and Jonas Almeida. Alignment-free sequence comparison - a review. *Bioinformatics*, 19(4):513–523, 2003.

39 Susana Vinga, Alexandra M. Carvalho, Alexandre P. Francisco, Luís M. S. Russo, and Jonas S. Almeida. Pattern matching through chaos game representation: bridging numerical and discrete data structures for biological sequence analysis. *Algorithms for Molecular Biology*, 7:10, 2012.

40 John W. Wilbur and David J. Lipman. Rapid similarity searches of nucleic acid and protein data banks. *Proc. Natl. Acad. Sci. USA*, 80:726–730, 1983.