ISTITUTO DI ANALISI DEI SISTEMI ED INFORMATICA
"Antonio Ruberti"
CONSIGLIO NAZIONALE DELLE RICERCHE

E. Weitschek, D. Santoni, M.C. De Cola, G. Felici

ABOUT SIMILARITY OF DNA READS

R. 13-17 2013

Emanuel Weitschek – Istituto di Analisi dei Sistemi ed Informatica del CNR, viale Manzoni 30 - 00185 Roma, Italy. Email: emanuel.weitschek@iasi.cnr.it.

Daniele Santoni – Istituto di Analisi dei Sistemi ed Informatica del CNR, viale Manzoni 30 - 00185 Roma, Italy. Email: daniele.santoni@iasi.cnr.it.

Maria Cristina De Cola – Istituto di Analisi dei Sistemi ed Informatica del CNR, viale Manzoni 30 - 00185 Roma, Italy. Email: cristina.decolo@gmail.com.

Giovanni Felici – Istituto di Analisi dei Sistemi ed Informatica del CNR, viale Manzoni 30 - 00185 Roma, Italy. Email: felici@iasi.cnr.it.

ISSN: 1128–3378

Collana dei Rapporti dell'Istituto di Analisi dei Sistemi ed Informatica "Antonio Ruberti", CNR

viale Manzoni 30, 00185 ROMA, Italy

tel. ++39-06-77161
fax ++39-06-7716461
email: iasi@iasi.cnr.it
URL: http://www.iasi.cnr.it

Abstract

The DNA assembly process consists in reconstructing a complete DNA sequence from a high number of reads - fragments of the complete original DNA sequence (the genome) - extracted in a sequencing procedure. The need for fast assembly methods has increased with next generation sequencing (NGS) machines, that extract a high number of short reads from a genomic source. A large class of DNA assembly methods rely on a read comparison step, where promising read pairs are separated from non-promising ones in order to reduce the computational burden of the main assembly algorithm and to speed up the reconstruction of sequenced DNA.
A reads comparison method based on an alignment-free distance is proposed, where the similarity of two reads is computed by calculating the substrings of fixed dimensions (k-mers) frequencies. The alignment-free distance is compared with the quality of the BLAST alignment and the Needleman-Wunsch edit distance. Additionally, an ideal distance is defined and considered for the comparisons: this distance is computed by knowing in advance the mapping of the reads on the reference genome and by extracting the reads overlapping positions.
The distances are evaluated on three organisms *Saccharomyces cerevisiae* (Yeast), *Escherichia coli* (E.coli), and *Homo sapiens* (Human) and the prediction power of the distances is assessed by analyzing training and test sets composed of different reads pairs: the alignment-free distance is able to compete with the more computationally demanding alignment based distances.
The effectiveness of the alignment-free distance for computing a sound read similarity is proven. The alignment-free read pairs comparison is successfully adopted for DNA reads classification.

1. Introduction

DNA assembly consists in the reconstruction of a genomic sequence, whose original primary structure is unknown, from a large number of small fragments obtained by a *sequencing* operation. The small fragments, called *reads*, come from random parts of the genomic sequence, and are partially overlapping. The number and length of the reads obtained from a given sequence is determined by the type of sequencing experiment. Recently developed Next Generation Sequencing (NGS) machines allow the extraction of an extremely large amount of reads at a significantly reduced cost. The size of such reads is very small when compared with the length of a genome: it may range from 40 to 300 base pairs (characters), while the length of a simple genome, e.g. bacteria, is in the order of millions base pairs (Mbp). The complexity of the assembly process is driven by the length of the reads and by their number: longer reads are easier to be assembled, while a larger number of reads, although providing more information, require a higher computational effort for their processing. Typically, the number of reads produced by NGS experiments reaches several millions.

Three major NGS technologies are currently used [15]: Roche 454, Illumina and Ion Torrent. At present, Illumina technology performances are 40 gigabase pairs per day at a low cost per base pair [`illumina.com`] with reads average length of 70; Roche 454 performances are 1 gigabase pairs per day at a higher cost with reads average length of 250 [`454.com`]; Ion Torrent machines produce reads of 200 bp with a throughput of 5 gigabase pairs per day at a low cost per base pair [23]. The use of NGS machines results in much larger sets of reads to be assembled, posing new problems for computer scientists and bioinformaticians, whose task is to design assembly algorithms that are capable of aligning and merging the reads for effectively reconstructing the genome, or large portions of it, with sufficient precision and speed [24].

Many competing algorithms have been developed for DNA assembly; a comprehensive comparison of recent and well established methods can be found in [8] and [13], where these methods are tested on common benchmarks. The assembly problem is proven to be NP-hard [25] and several heuristic algorithms have been proposed for effectively solving this problem. Algorithms for DNA assembly are based on two main approaches: overlap graphs (e.g., [12]) and De Bruijn Graphs [8]. In the overlap graphs approach each read correspond to a node, the overlaps between read pairs are usually computed with alignment methods, and the weight of the edges is determined; an assembly is derived from an hamiltonian path in this graph. In the De Bruijn Graphs approach, reads are represented on a graph whose nodes and arcs are nucleotides subsequences [14]; the assembly is found searching an eulerian cycle in this graph and is represented by a sequence of arcs. A large number of these algorithms - in particular, those using the overlap graph - are based on the evaluation of the similarity between two reads. Such similarity is in fact the main tool to assess whether two reads may be overlapped in the reconstruction process or not. In these approaches it is thus required to compare each read pair, generating a number of comparison that is potentially quadratic in the - indeed large - number of reads. In this setting methods that can quickly establish whether two reads are good candidates for the alignment are therefore important.

This step is referred to as *filtering*. Its effect is to quickly filter out from the candidate set of read pairs those that would not provide a good alignment in the following steps of the process. The computational cost of filtering is then traded-off by the speedup obtained when a smaller set of read pairs is considered for assembly.

The objective of this paper is to evaluate the appropriateness of the alignment-free distances for read pairs similarity assessment, and to compare it with other similarity measures (distances) that are based on complete or partial alignment of the two reads.

Alignment-free techniques have already been proven to be successful in sequence analysis [31], and are classified in two main groups: methods based on sequence compression and methods that rely on subsequences (oligomers) frequencies [31]. The aim of the methods in the first group is to find the shortest possible description of the sequence. They rely on computing the similarity of the sequences by analyzing their compressed representations. Currently available methods are based on the Kolmogorov complexity [22] and on Universal Sequence Maps [9]. An extensive review can be found in [17]. The methods based on oligomers frequencies rely on the computation of the substrings frequencies of a given length k in the original sequences, called k-mers. Here the similarity of two sequences is assessed based

only on the dictionary of subsequences that appear in the strings, irrespective of their relative position [11]. The alignment-free distance adopted in this study is the k-mer frequency count analysis [19], where the substring frequencies of length k of a sequence are represented in a real vector and are therefore easily tractable in a metric space. The sequences are compared according to their vector representations, by computing a distance between them. A simple and easy to compute distance measure is the Euclidean distance; although, others may be used, e.g. the $d2$ distance of [18].

We consider the very straightforward implementation of the alignment-free distance, based on the euclidean distance of the frequency distribution of k-mers (i.e., substrings composed of k consecutive bases) in the two reads. Such a distance, referred to as `AF` in the following, is very simple to compute and requires linear time in the dimension of the reads. As far as the choice of the length of the oligomers, we adopt $k = 4$ (`tetramers`) as in many references this value has been confirmed to provide an ideal balance between the length of the oligomers and their number, when the sequences are expressed in the (A,C,G,T) alphabet [29, 27, 30]. `AF` is compared with two frequently used methods to measure DNA string similarity that are based on alignment: the Needleman-Wunsch edit distance, and the BLAST alignment algorithm (`NW` and `BL` in the following). Both methods require quadratic time in the length of the reads. Their choice is motivated by the fact the first is a global alignment method, in the sense that it searches for the best alignment of the complete reads, while the second is a local alignment, that searches for the longest possible portion that is aligned well within the two reads. Their choice covers therefore the two main approaches used in computing alignment based distances.

To perform a proper comparison among the three different distances we adopt the following test. First, we assume the existence of an *ideal distance*, that is, the distance that is given by the degree of overlapping of reads that have been aligned on their known reference genomic sequence. Second, we verify the ability of the three distances in approximating this ideal (target) distance. Such an approximation is measured by the ability of predicting, given a pair of reads, the magnitude of the target distance using the magnitude of the predicting one. Such assumption is based on the fact that an assembly method that uses the target distance to evaluate the opportunity of overlapping two reads would result in a extremely satisfactory assembly. To align the reads over the original sequence we use the well established Bowtie algorithm [20]. Two reads receive a maximum distance value if they do not overlap over the reference sequence; else, they receive a distance inversely proportional to their degree of overlapping over the sequence (e.g., they would have minimum distance if they are aligned in the same position by the Bowtie algorithm). Given two reads, we define such a value their *Bowtie distance* (`BT` in the following). We refer to `BT` as the *target* distance and to `AF`, or `NW` or `BL` as the *predictor* distance. A *threshold predictor* is a mapping between values of the target distance and of the predictor distance, such that, given two reads, the target distance between the two reads is predicted to be below a given value when the predictor distance between the same two reads is below the mapped value. Such predictor would thus have, for each value of the target distance, a certain error measured in terms of false positive and false negative predictions. The quality of the threshold predictor is then given by the error distribution over the values of the target distance.

We consider DNA sequences coming from three different organisms: *Saccharomyces cerevisiae*, *Escherichia coli*, and *Homo sapiens*. Publicly available sets of NGS reads for the three reference sequences are used. Each experiment is based on a large sample of read pairs, from which the best possible threshold predictor (among `AF`, `NW` or `BL`) of the value of the `BT` distance is computed. The precision of the predictors is evaluated using ROC curves both on the samples of read pairs used to identify the best predictors, and on other samples from the same set of read pairs that were unseen before. The results of these experiments show that `AF` performs very well as a thresold predictor for `BT`; its performances are indeed superior to those of `NW` and comparable to those exhibited by `BL`. Both `NW` and `BL` are much more demanding in terms of computing time when compared with `AF`.

The paper is organized as follows.

In Section *Methods* we provide sufficient background for the main methods and techniques used in the paper: the different read pairs distances adopted (subsections *Bowtie distance*, *Needleman-Wunsch Edit distance*, *The Blast alignment algorithm* and *alignment-free Distance on Tetramer Frequencies*) are described. Following, we delineate the rationale of threshold predictors and the way they are computed from data. Section *Results* describes the experimental design. First, the data sets used for the experiments (subsections *Data sets for saccharomyces cerevisiae / escherichia coli / homo sapiens*)and the way

read pairs samples are created from them are illustrated. Then we consider how the different distances correlate among each other over the different datasets (subsection *Pearson correlation among distances*), analyze the performance of the predictors over the training sets with the support of ROC curves and AUC indicators (subsection *Performance analysis of threshold predictors*), and discuss the results of the predictors for a cross validation evaluation scheme (subsection *Cross Validation Performances of the AF threshold predictor*).

Section *Conclusions* provides further discussion of the results, and draws the paper's conclusions.

2. Methods

2.1. Bowtie distance

The Bowtie distance (`BT`) is obtained after computing the alignments of the reads with the Bowtie algorithm [20, 21] to the reference genome. Bowtie is a fast and efficient read aligner, that aligns reads to the reference genome at a very high speed (25 million 35-bp reads per hour). Before the alignments computation Bowtie builds an index of the reference genome with a Burrows-Wheeler approach. Two versions of Bowtie are available: Bowtie1 [20] and Bowtie2 [21]. The first is optimized for short genomes, the latter for longer ones and supports gapped, local, and paired-end alignment modes. For every considered read the alignment position in the reference genome is obtained after running the Bowtie algorithm. For two reads r_1, r_2 we define the Bowtie distance as follows:

$$BT = 1 \quad \frac{2 * (\nabla(r_1, r_2))}{\lambda_1 + \lambda_2}$$

where $\nabla(r_1, r_2)$ is the number of overlapped positions of r_1 and r_2, λ_1 is the length of r_1 and λ_2 is the length of r_2. We note that the Bowtie distance takes into account also multiple alignments of the two reads over the reference sequence.

2.2. Needleman-Wunsch Edit Distance

The Needleman and Wunsch algorithm
tt NW [26], based on dynamic programming, is commonly used to perform a global alignment on two sequences. Time complexity of the algorithm is quadratic with respect to the lengths of the two sequences (N and M) to be aligned ($O(n * m)$, where n and m are the number of bases in the two reads). The NeoBio [6] Java implementation of the `NW` algorithm was adopted for performing the distance evaluation experiments. The Needleman and Wunsch score was turned into the `NW` distance by reversing its order and normalizing between 0 and 1.

2.3. The Blast alignment algorithm

The Basic Local Alignment Search Tool (`Blast`) algorithm [10] is used to compare a query sequence with a library or database of sequences. `Blast` uses an heuristic approach that is less accurate than other methods, but much faster. Time complexity of `Blast` is also quadratic ($O(n * m)$ where n and m are the lengths of the two reads to be aligned). It is worth to highlight that this is the same time complexity as other algorithms, including the `NW` global alignment. However, given the heuristic nature of the algorithm, using the statistically significant elimination of high-scoring Segment Pairs (HSPs) and words, `Blast` significantly lowers the numbers of segments which need to be extended and letting the algorithm run much faster than its worst case time complexity. In this work we use `Blast2`, that is the `Blast` version to simply align two sequences. The `Blast` implementation available in [1] running under a linux 64bit environment (kernel 3.2.0-34-generic) was adopted for computing the `Blast` scores and the `Blast` expected values between the considered read pairs. As the final `BL` distance we adopted the `Blast` score reversed and normalized between 0 and 1 (given the fixed and equal size of the sequences, the `Blast` expected values resulted to be perfeclty log-correlated with the `Blast` scores).

2.4. Alignment-free Distance on Tetramer Frequencies

We provide a simple sketch of the alignment-free distance computation used in this paper, mainly based on [31]. The frequencies of each substring of length 4 (also called *tetramer*) are computed counting the occurrences of the substrings in the read with a sliding window of length 4, starting at position 1 and ending at position $n-4+1$, where n is the length of the read. For the alphabet composed of the four symbols `(A,C,G,T)` we have a total of $4 \times 4 = 256$ different tetramers and thus each read is represented by a vector of 256 real numbers between 0 and 1. The choice of tetramers is motivated by [29, 27, 30], which confirm the ideal balance between the length of the oligomers and their number. Given two reads, the Euclidean distance between their associated frequency vectors is an inverse measure of the similarity of the two reads, and we refer to it as the `AF` distance between the two reads. An efficient Java implementation of the alignment-free frequency vector computation and representation was developed for computing the `AF` distance between the available read pairs. The related algorithm is linear with respect to the length of the reads. The software was run under a 64 bit linux environment (kernel 3.2.0-34-generic) with a 64 bit Oracle Java Virtual Machine (version 1.7.0_09).

2.5. Threshold distance predictors

We recall that the scope of this paper is to show that, given two DNA reads, the `BT` distance can be approximated with a tolerable degree of accuracy by the `AF` distance; to be more precise, we want to show that `AF` approximates `BT` as well as `NW` or `BL` distances, although less computationally demanding.

For a formal definition of *threshold predictor* we need some additional notation. Let r_1 and r_2 be two generic reads coming from a DNA sequencing operation, and $d_1(r_1, r_2)$, $d_2(r_1, r_2)$ be two alternative read-to-read distance functions. Given a vector α of dimension m, $\alpha = (\alpha_1, \alpha_2, \ldots, \alpha_m)$, a *threshold predictor* of $d_1(\cdot, \cdot)$ by $d_2(\cdot, \cdot)$ is determined by a vector $\beta = (\beta_1, \beta_2, \ldots, \beta_m)$. Given two reads r_1, r_2, the prediction on $d_1(\cdot, \cdot)$ is obtained by the following rule:

$$\texttt{if } d_2(r_1, r_2) \leq \beta_i \texttt{ then } d_1(r_1, r_2) \leq \alpha_i, i = 1, \ldots m.$$

The definition of threshold predictor depends on the choice of the interesting value of $d_1(\cdot, \cdot)$ in the vector α. We are indeed interested in the prediction only if $d_1(r_1, r_2)$ is below a certain value based on the value of $d_2(r_1, r_2)$, and would like this prediction to be precise for a limited number of reference values (those contained in the vector α).

We have `BT` as *target* distance (i.e. $d_1(\cdot, \cdot)$) and the other distances as *predictors* (i.e. $d_2(\cdot, \cdot)$). We are interested in predicting if two reads are close according to `BT`. But in DNA assembly it is not possible to know the `BT` distance (the original sequence is unknown) and thus we want to predict it using `NW`, `BL` or `AF`. In this case a threshold predictor that is precise for small values in α will be useful; else, we are not interested in predicting whether two reads are far or very far from each other: we only want to know if they are close to each other or not.

A proper way to evaluate the quality of a threshold predictor is to measure its errors, over one or more samples of read pairs where all distances are known. For each value α_i we may infact measure the *true positive* (read pairs that are below α_i and are predicted to be below α_i by the threshold predictor) and *true negative* (read pairs that are above α_i and are predicted to be above α_i by the threshold predictor) rates associated with the above rule, and from this derive standard performance indicators such as ROC curves and AUC values [16].

Read pairs samples are used also to identify and test good threshold predictors. Given the interesting value $\alpha = (\alpha_1, \alpha_2, \ldots, \alpha_m)$, we compute for a sufficiently large set of candidate values $\beta = (\beta_1, \beta_2, \ldots)$ the true positive and the true negative rates, construct the associated ROC curve for each value α_i, and derive the corresponding AUC value. If the AUC value is good enough, we identify the value β_j that provides the largest combination of true positive and true negative rates, and adopt that for α_i. A complete threshold predictor is then obtained by repeating this operation for each $\alpha_i, i = 1, m$.

The measure of a distinct precision value for each level of the target function enables to evaluate the appropriateness of the predictors there where it is needed. Clearly, the validity of a *threshold predictor* depends on the quality and the representativeness of the samples used to train (e.g., to derive the

predicting vector β) and test the method. For the latter we adopt a standard cross-validation approach where the β values are derived on a training sample and are then tested on the other samples.

The set of reference values α (for the target distance) and β (for the predictors) that have been used for the experiments are the values that separate the *percentiles* of the read pair distance distribution. This allows to sample the whole variation range of the normalized distances, obtaining a finer granularity in the portions where the density is higher. According to this choice, both α and β are vectors of dimension 100 composed of real values between 0 and 1 in non decreasing order. Clearly this choice may be changed with equally spaced intervals without a significant effect on the results, once the proper granularity of the intervals has been identified.

3. Results

The main goal of this work is to provide evidence that the
tt AF distance (Section *alignment-free Distance on Tetramer Frequencies*) is a suitable approach to approximate `BT` distance. We applied `AF` to three different data sets (described in the next subsections) showing the performances with respect to other two alignment based algorithms (`NW`, `BL`). As a first step, we computed the Pearson correlation coefficients among the distances in the read pairs samples; then, we computed the ability of each measure to predict `BT` distance at given `BT` thresholds, by ROC curves and the corresponding AUC values. We verified and demonstrated the consistency of the predictions by a cross validation scheme, using, for each one of the three organisms considered, the six different training sets; then the prediction quality was tested on the other five sets.

3.1. Data sets for saccharomyces cerevisiae

A total of 3,551,079 reads consisting of 710.2M base pairs from the organism *saccharomyces cerevisiae*, commonly known as `yeast`, were downloaded from the NCBI Sequence Read Archive database [5] (accession number ERX191563 and run id ERR216898). These genomic source reads were obtained with an Illumina HiSeq 2000 sequence machine using a whole genome shotgun strategy, and their average length was 236 bases (standard deviation of 69). The reads were aligned to the *saccharomyces cerevisiae* reference genome available on [7] using the Bowtie2 algorithm [21]. From the resulting alignments 54,860 reads belonging to chromosome 1 were selected randomly, and their reverse complemented representations were computed, resulting in a total of 109,720 reads. Out of all the possible pairs of different reads from this set, we selected six subsets, each with $1,000,000$ read pairs. The random selection of these six subsets was controlled in order to have half of the set with non overlapping reads (e.g., maximum Bowtie distance) and the other half with a varying degree of overlap. The four distances were then computed for each pair in the set: Bowtie Distance `BT`, the Needleman and Wunsch alignment `NW`, the Blast alignment `BL` and the alignment-free distance `AF` over the tt tetramers, tetramers, i.e. substrings of length 4. These measures were all turned into proper distances ranging from 0 to 1 with 0 corresponding to equal reads and 1 corresponding to maximally different reads. The six datasets will be referred, in the following, as `YA`, `YB`, ..., `YF` (`Y` as in yeast).

3.2. Data sets for escherichia coli

A total of 436,142 reads from the organism *escherichia coli* were downloaded from Chang Gung University, Department of Parasitology, College of Medicine [3]. These reads were obtained with a Roche 454 sequencing machine, and their average length was 235 bases (with standard deviation of 4). The reads were aligned to the *escherichia coli* reference genome available on [2] using the Bowtie algorithm [20]. From the resulting alignments $100,000$ reads were selected randomly (with equal probability), and their reverse complemented representations were computed, resulting in a set of $200,000$ reads. Out of all the possible pairs of different reads from this set, we selected six subsets, each with $200,000$ read pairs. The random selection of these six subsets was controlled in order to have half of the set with non overlapping reads (e.g., maximum Bowtie distance) and the other half with a varying degree of overlap. The four distances were then computed for each pair in the set: Bowtie Distance `BT`, Needleman and

Wunsch alignment NW, Blast alignment BL and alignment-free AF over the tetramers. These measures were all turned into proper distances ranging from 0 to 1 with 0 corresponding to equal reads and 1 corresponding to maximally different reads. The six datasets will be referred, in the following, as EA, EB, ..., EF (E as in E.coli).

3.3. Data sets for homo sapiens

A total of 14,267,012 reads consisting of 1.1G bases from an ancient homo sapiens genome [28] with accession number SRX013970 and run id SRR031057 were downloaded from the NCBI Sequence Read Archive database [5]. These genomic source reads were obtained with an Illumina Genome Analyzer II high throughput sequencing machine using a whole genome shotgun strategy, their average length was 75 (standard deviation of 5). The reads were aligned to the human hg18 reference genome available on [4] using the Bowtie2 algorithm [21]. From the resulting alignments 183,672 reads belonging to chromosome 1 were selected randomly, and their reverse complemented representations were computed, resulting in a total of 367,344 reads. Out of all the possible pairs of different reads from this set, we selected six subsets, each with 1,000,000 read pairs. The random selection of these six subsets was controlled in order to have half of the set with non overlapping reads (e.g., maximum Bowtie distance) and the other half with a varying degree of overlap. The four distances were then computed for each pair in the set: Bowtie Distance BT, Needleman and Wunsch alignment NW, Blast alignment BL and alignment-free AF over the tetramers. These measures were all turned into proper distances ranging from 0 to 1 with 0 corresponding to equal reads and 1 corresponding to maximally different reads. The six datasets will be referred, in the following, as HA, HB, ..., HF (H as in human).
A compact summarized overview of the data sets is given in table 3.3.

Table 1: Compact overview of the datasets

The Datasets			
	human	**yeast**	**E.coli**
Genome length	3.2 Mb	12.1 Mb	4.6 Mb
Sequencing machine	Illumina GA II	Illumina HiSeq	Roche 454
# reads	14 M	3.5 M	0.4 M
avg.reads length	75	236	235

3.4. Pearson correlation among distances

An initial comparison among the four distances is based on the analysis of the correlation coefficients of one distance with the others, over a sufficiently large sample of read pairs. The matrices in tables 3.4, 3.4, 3.4 report the Pearson correlation values between the four read-to-read distances for the three organisms.

Table 2: Pearson correlation matrix between the four read-to-read distances for Yeast - YA

	BT	**NW**	**BL**	**AF**
BT	1.00	0.45	0.81	0.63
NW	0.45	1.00	0.48	0.52
BL	0.81	0.48	1.00	0.61
AF	0.63	0.52	0.61	1.00

For each organism, the correlations are computed in one of the six samples available. Similar results are obtained on the other samples (not shown). It is of course of interest to analyze the correlation of the predictor distances (NW, BL, AF) with the target distance BT. We first note that the correlations measures are significantly different in the three organisms; in yeast the NW distance is correlated extremely poorly with BT, while its correlation improves for E.coli and human; AF correlation with BT is also weaker

Table 3: Pearson correlation matrix between the four read-to-read distances for Human - HA

	BT	**NW**	**BL**	**AF**
BT	1.00	0.68	0.72	0.67
NW	0.68	1.00	0.73	0.72
BL	0.72	0.73	1.00	0.63
AF	0.67	0.72	0.63	1.00

Table 4: Pearson correlation matrix between the four read-to-read distances for E.Coli - EA

	BT	**NW**	**BL**	**AF**
BT	1.00	0.76	1.00	0.95
NW	0.76	1.00	0.76	0.82
BL	1.00	0.76	1.00	0.95
AF	0.95	0.82	0.95	1.00

in yeast with respect to the other two organisms. The BL correlation with BT appears to be the higher among the three predictors. It is moreover evident that BT is perfectly, or almost perfectly, reproduced from the predictors BL and AF in E.coli, then followed by yeast and human.

The analysis of the linear dependence between the distances, although of interest for our analysis, does not tell the whole story. The requirement of a linear dependence between target and prediction distance is indeed a biased condition for the existence of the BT threshold predictor that we are interested in; we know in fact that some (small) reference values of the target distance are more interesting than other and need to be predicted with higher accuracy, while correlation represent an average similarity over the whole scale of the target. More appropriate evaluations are then reported on the following sections.

3.5. Performance analysis of threshold predictors

In this section we analyze the three predictors prediction power of the target distance. As mentioned above, we consider 100 intervals of the target BT distance corresponding to the percentiles of its distribution, and identify by exhaustive inspection the percentiles of the predictor distance that minimize the prediction error. Such analysis is performed by means of ROC curves, where we plot the true positive rate against the false positive rate for a given percentile of the target distance, when the percentiles of the predictor distance vary from 1 to 100. We recall that an ideal ROC curve contains the point (0,1) and therefore the area under the ROC curve (AUC) has value 1. Smaller values of AUC represent poorer prediction peformances, and, in general, an AUC value is usually considered very good when in the proximity of 0.9.

We start presenting the ROC curves for four interesting values of the target distance percentiles, that correspond to particular values of BT: 0.10, 0.15, 0.20, and 0.25. In figures 1, 2, 3 the ROC curves for predictors NW, BL, and AF are reported for the four reference values (panels on the rows) and for the three organisms (panels on the columns). As one can observe, both AF (green, solid) and BL (blue, dotted) curves perform much better than NW (red, dashed). The ROC curves related to yeast are reported in Figure 1, as one can observe, both AF and BL perform much better than NW with AUC values higher than 0.9 while NW curves have AUC values close to 0.7. Figure 2, related to E. coli, shows a very stable scenario, all the three measures are able to precisely predict BT for all the considered thresholds reaching values of AUC close to 1. Figure 3, related to human, shows that AF performs slightly better than NW that in turn performs slightly better than BL. AUC values of AF are close to 0.95 while those of NW around 0.91 and those of BL range from 0.9 to 0.88.

A more comprehensive outlook of the performances of the three predictors can be glanced from the three panels in Figure 4.

Here we report the AUC values for all the 100 percentiles of the target distance, for three samples coming from yeast, E.coli, and human. Similar results are obtained when the other five samples from each organism are used. The charts show very clearly that, for all three predictors, the precision decreases for

higher percentiles (i.e., larger values of the target distance). Lower percentiles (i.e., lower BT) correspond to higher level of overlapping so it is reasonable that for these percentiles it is easier for all the three measures to predict BT. The higher the percentiles (i.e. the smaller the overlapping) the higher the noise in the prediction will be. AUC values are indeed very high for smaller percentiles with the exception of NW in human. In Figure 4 panel A, related to yeast, there is evidence that BL and AF have both good performances for all the percentiles in terms of AUC, showing values higher than 0.9, until percentile 30, and anyway values higher than 0.8 for the last percentiles. BL performs slightly better than AF until the percentile 20, then AF is better until the percentile 60 and again BL is better until the last percentiles. NW AUC values range from 0.72 until 0.68 showing again a light decreasing slope. Figure 4 panel B, related to E. coli, shows, as previously highlighted, that the three measures have very good performances for this organism, with AUC values close to 1 until percentile 33, meaning that all the three measures are able to correctly predict BT. From the percentile 33 NW AUC values significantly decrease, while AF and BL are still close to 1 until the percentile 80, when they slowly decrease keeping anyway values higher than 0.9. In Figure 4 panel C, related to human, it can be observed that the three curves start from an almost common AUC value, around 0.95, but diverge when the percentiles increase. AF has the best performance mantaining AUC values higher than 0.9, then NW decreasing until 0.85 and finally BL falling down to 0.55.

3.6. Cross Validation Performances of the AF threshold predictor

The results discussed above show that the BL and AF predictors perform well when they are evaluated on the same sample that has been used to train the predictors reference values. It is indeed more interesting to verify that the relation between the predictor and the target distance, derived from a read pairs sample, maintains its validity also on other samples that were not used to train the method.

We restrict this analysis to the AF predictor, an test the threshold predictor rules, derived from one sample, on the other five samples of the same organism, in a cross validation scheme. The results are summarized in Figure 5, where positive and negative precision rates are reported for the four reference values of the target distance already used in figures 1, 2, 3 (0.10, 0.15, 0.20, and 0.25), for all the combinations of the cross validation scheme, and in Figure 6, where we plot the same positive, negative and total precision rates for all the 100 reference values over a single sample (Panel A for yeast, Panel B for E.coli, and Panel C for human). Similar results have been obtained also for the other five samples of the three organisms.

Figure 6 shows that in yeast (panel A) positive precision rate ranges from a percentage of 88.28 (BT = 0.25) until 90.58 (BT = 0.105) while the negative precision rate ranges from 82.60 (BT = 0.105) until 85.18 (BT = 0.25). In E. coli (panel B) both positive and negative precision rates show values always higher than 96.34. In human (panel C) positive precision rate always is around a percentage of 90 and negative precision rate ranges from 83.07 (BT = 0.105) and 85.98 (BT = 0.25).

Figure 5 shows positive, negative and total precision rates for all the percentiles in the three organisms, revealing, as expected, that for E. coli (panel B) AF has very good performances also in the cross validation, with a slight decreasing slope along the higher percentiles, but anyway the total precision rate is always higher that 0.9. In human and yeast (panel A and C) we have global good performances, with a total precision rate ranging from 0.8 to 0.9.

The results described in Figure 5 and Figure 6 confirm indeed that the AF predictor performs very well also in the cross validation scheme, and exhibits good generalization capabilities. The parallel analysis conducted on the other two candidate predictors result in similar performances of BL and in much poored performances of NW (results not shown).

4. Conclusion

The results discussed in the previous section clearly show the efficacy of the alignment-free distance in estimating a good read-to-read distance measure. The performances of AF in predicting BT are better than NW and at least comparable to BL, but it is clear the advantage, in terms of computational time, of using AF (linear in the size of the input). Indeed, as already discussed in *Methods* Section, AF is much faster

than NW and BL. As reported above, the prediction power of the three measures depends on the organism we considered, this issue deserves a further analysis and discussion. We analyzed two eukariotic genomes (yeast and human) and one bacterial one (E. coli), there is evidence that in E. coli the performances of all the three predictors are globally much better. This may be due to the nature of bacterial genomes that are made, almost totally, of coding sequences, making easier to recognize overlapping regions and reducing the noise due to low complexity regions present in the intergenic eukariotic portions of genome.

An additional fact that deserves attention is that the distance based on global alignment (NW), in general, performs poorly with respect to the one based on local alignment (BL); the alignment-free distance (AF) seems to compare well with the local alignment one, despite it is based on the evaluation of the whole sequence, thus overcoming the bias that may derive from requiring the global alignment of the two reads. Such consideration is somehow strenghtened by the different performances obtained on reads of different sizes; we recall that reads from human are smaller (average size 75 bases) than yeast and E. coli (236 and 235 bases, respectively); such difference may explain the improved performances of the NW distance in human, as with shorter reads the advantage of local versus global alignment is reduced.

The results shown in this paper are to be considered as a starting point to derive more efficient sequences similarity assessments methods for DNA reads from NGS sequencing. Read pairs comparison based on alignment-free distances may be for example useful in assembly processes or reads classification.

12.

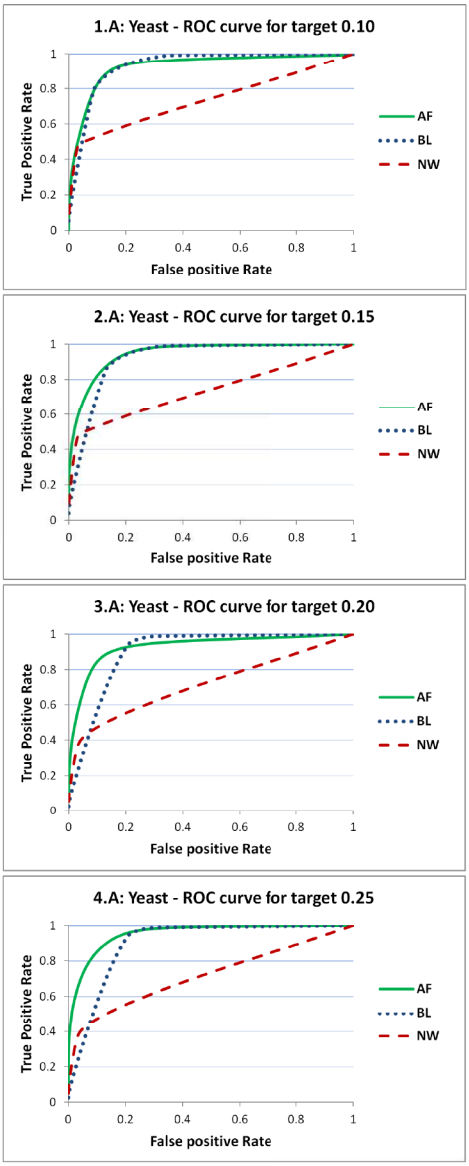

Figure 1: ROC curves for threshold predictors (yeast) ROC curves obtained from threshold predictors of four different reference values of the target `BT` distance $(0.10, 0.15, 0.20, 0.25)$. The charts report the ROC curves for the three predictor distances (`NW, BL, AF`). Results are provided on samples for yeast (`YA`).

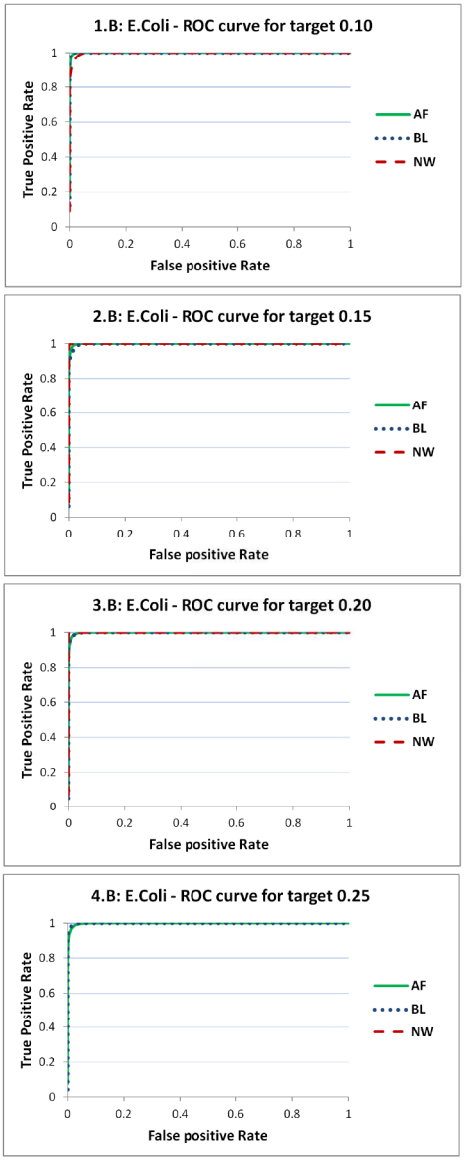

Figure 2: ROC curves for threshold predictors (E.Coli) ROC curves obtained from threshold predictors of four different reference values of the target `BT` distance $(0.10, 0.15, 0.20, 0.25)$. The charts report the ROC curves for the three predictor distances (`NW, BL, AF`). Results are provided on samples for E.coli (`EA`).

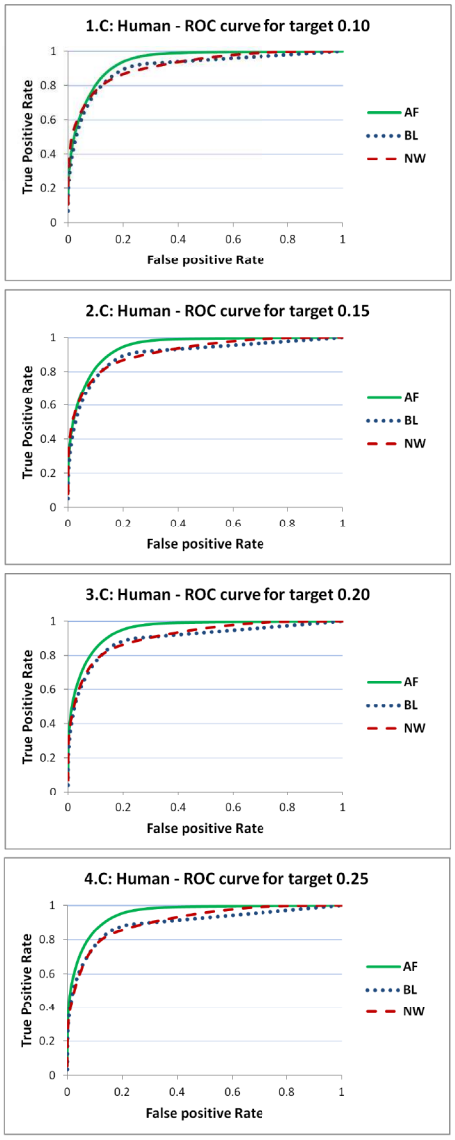

Figure 3: ROC curves for threshold predictors (human) ROC curves obtained from threshold predictors of four different reference values of the target `BT` distance $(0.10, 0.15, 0.20, 0.25)$. The charts report the ROC curves for the three predictor distances (`NW, BL, AF`). Results are provided on samples for human (`HA`).

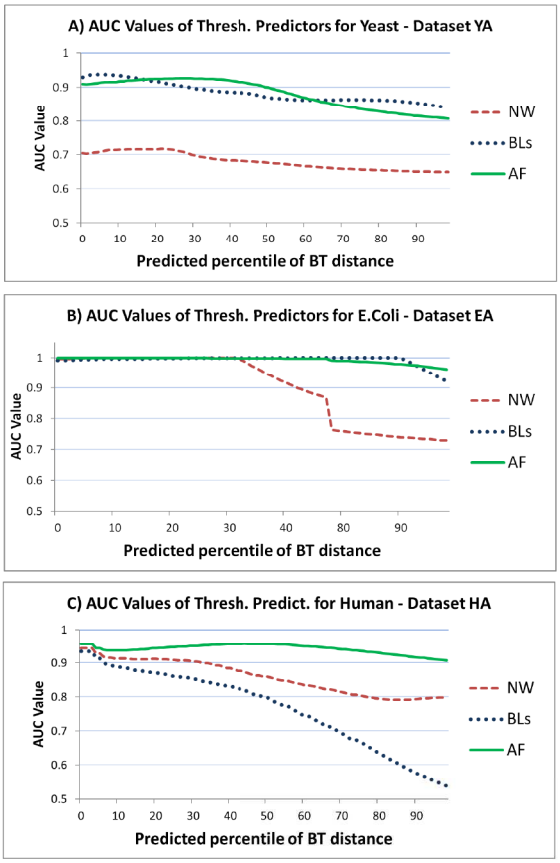

Figure 4: AUC Values for target percentile values, AUC values for each percentile values of the target `BT` distance. The three panels report AUC values for the three predictor distances (`NW`, `BL`, `AF`). Results are provided on samples for yeast (Chart A, `YA`), E.coli (Chart B, `EA`), and human (Chart C, `HA`).

16.

A) True Positive and True Negative Rates for reference values of Target Distance BT when predicted by AF, for Yeast. Predictor is trained on one set and tested over the other 5 sets

Set used for training the predictor	**Target diastance BT reference value for threshold**							
	0.105		**0.15**		**0.205**		**0.25**	
	True Pos %	True Neg %	True Pos %	True Neg %	True Pos %	True Neg %	True Pos %	True Neg %
YA	90.26%	85.05%	90.00%	85.30%	89.13%	86.76%	87.96%	87.57%
YB	90.03%	85.23%	89.75%	86.01%	89.69%	86.20%	87.94%	87.61%
YC	90.01%	85.28%	89.72%	86.06%	88.44%	87.34%	87.39%	88.09%
YD	90.23%	85.03%	89.74%	86.03%	89.12%	86.76%	88.58%	86.94%
YE	90.73%	84.52%	90.64%	85.11%	89.05%	86.77%	87.87%	87.57%
YF	92.22%	70.50%	91.41%	70.48%	91.38%	71.96%	89.95%	73.30%
Average	***90.58%***	***82.60%***	***90.21%***	***83.25%***	***89.47%***	***84.30%***	***88.28%***	***85.18%***

B) True Positive and True Negative Rates for reference values of Target Distance BT when predicted by AF, for E.Coli. Predictor is trained on one set and tested over the other 5 sets

Set used for training the predictor	**Target diastance BT reference value for threshold**							
	0.105		**0.15**		**0.205**		**0.25**	
	True Pos %	True Neg %	True Pos %	True Neg %	True Pos %	True Neg %	True Pos %	True Neg %
EA	97.00%	99.09%	96.38%	98.78%	95.43%	98.66%	94.61%	98.60%
EB	99.08%	98.14%	98.90%	97.43%	98.54%	96.92%	97.97%	96.78%
EC	100.00%	96.11%	99.98%	94.75%	99.98%	93.48%	99.96%	92.56%
ED	98.67%	98.47%	98.30%	97.93%	97.95%	97.49%	97.49%	97.21%
EE	98.31%	98.69%	97.42%	98.39%	97.15%	97.97%	96.57%	97.84%
EF	95.46%	99.45%	94.15%	99.35%	93.46%	99.19%	91.43%	99.35%
Average	***98.09%***	***98.32%***	***97.52%***	***97.77%***	***97.09%***	***97.29%***	***96.34%***	***97.06%***

C) True Positive and True Negative Rates for reference values of Target Distance BT when predicted by AF, for Human. Predictor is trained on one set and tested over the other 5 sets

Set used for training the predictor	**Target diastance BT reference value for threshold**							
	0.105		**0.15**		**0.205**		**0.25**	
	True Pos %	True Neg %	True Pos %	True Neg %	True Pos %	True Neg %	True Pos %	True Neg %
HA	91.05%	82.53%	91.24%	83.24%	90.96%	84.74%	90.19%	86.25%
HB	94.34%	78.82%	94.34%	79.58%	93.67%	81.78%	94.41%	81.35%
HC	89.27%	84.35%	90.15%	84.32%	88.92%	86.24%	89.77%	86.63%
HD	89.32%	84.28%	90.17%	84.27%	89.36%	86.24%	88.99%	87.24%
HE	89.12%	84.22%	89.57%	84.53%	89.02%	86.37%	87.99%	87.82%
HF	89.42%	84.23%	90.29%	84.22%	89.68%	86.05%	89.87%	86.59%
Average	***90.42%***	***83.07%***	***90.96%***	***83.36%***	***90.27%***	***85.24%***	***90.20%***	***85.98%***

Figure 5: True Positive and True Negative rates for reference target values, for yeast, E.coli, and human.

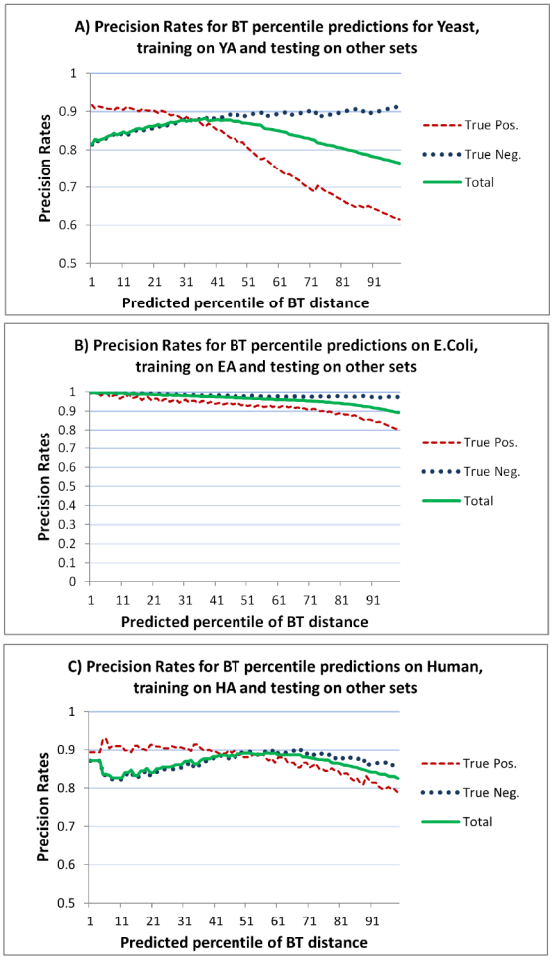

Figure 6: Cross Validation of threshold predictors Cross Validation precision rates of threshold predictors for each percentile values of the target `BT` distance; the Charts report precision rates on positive, negative and total. Results are provided on samples for yeast (Chart A, `YA` used for training), E.coli (Chart B, `EA` used for training), and human (Chart C, `HA` used for training).

18.

References

[1] "Blast package version 2.2.25-7."

[2] "E. coli bowtie index."

[3] "E. coli reads source."

[4] "Human bowtie index."

[5] "Ncbi sequence read archive."

[6] "Neobio: Bioinformatics algorithms in java."

[7] "yeast bowtie index."

[8] D. A. Earl, K. Bradnam, J. St. John, A. Darling, and o. others, "Assemblathon 1: A competitive assessment of de novo short read assembly methods," *Genome Research*, Sept. 2011. International competition of de novo genome assembly. The Symbiose team (IRISA/CNRS/ENS Cachan Brittany) participated to this competition.

[9] J. S. Almeida and S. Vinga, "Universal sequence map (usm) of arbitrary discrete sequences.," *BMC Bioinformatics*, vol. 3, p. 6, 2002.

[10] S. Altschul, W. Gish, W. Miller, E. Myers, and D. Lipman, "Basic local alignment search tool," *J Mol Biol*, vol. 215, pp. 403–410, 1990.

[11] A. Apostolico and F. Cunial, "Sequence similarity by gapped lzw," in *DCC*, pp. 343–352, 2011.

[12] J. Blazewicz, M. Bryja, M. Figlerowicz, P. Gawron, M. Kasprzak, E. Kirton, D. Platt, J. Przybytek, A. Swiercz, and L. Szajkowski, "Whole genome assembly from 454 sequencing output via modified dna graph concept," *Comput. Biol. Chem.*, vol. 33, pp. 224–230, June 2009.

[13] K. R. Bradnam, J. N. Fass, A. Alexandrov, and o. others, "Assemblathon 2: evaluating de novo methods of genome assembly in three vertebrate species," *arXiv preprint arXiv:1301.5406*, 2013.

[14] P. Compeau, P. Pevzner, and G. Tesler, "How to apply de bruijn graphs to genome assembly," *Nature biotechnology*, vol. 29, no. 11, pp. 987–991, 2011.

[15] M. Eisenstein, "The battle for sequencing supremacy," *Nature biotechnology*, vol. 30, no. 11, pp. 1023–1026, 2012.

[16] T. Fawcett, "An introduction to roc analysis," *Pattern Recognition Letters*, vol. 27, pp. 861–874, 2006.

[17] R. Giancarlo, D. Scaturro, and F. Utro, "Textual data compression in computational biology: a synopsis," *Bioinformatics*, vol. 25, no. 13, pp. 1575–1586, 2009.

[18] W. Hide, J. Burke, and D. B. DA VISON, "Biological evaluation of d2, an algorithm for high-performance sequence comparison," *Journal of Computational Biology*, vol. 1, no. 3, pp. 199–215, 1994.

[19] P. Kuksa and V. Pavlovic, "Efficient alignment-free dna barcode analytics.," *BMC Bioinformatics*, vol. 10 Suppl 14, p. S9, 2009.

[20] B. Langmead, C. Trapnell, M. Pop, and S. Salzberg, "Ultrafast and memory-efficient alignment of short dna sequences to the human genome," *Genome Biol*, vol. 10, no. 3, p. R25, 2009.

[21] B. Langmead and S. L. Salzberg, "Fast gapped-read alignment with bowtie 2," *Nature methods*, vol. 9, no. 4, pp. 357–359, 2012.

[22] M. Li and P. M. Vitnyi, *An Introduction to Kolmogorov Complexity and Its Applications*. Springer Publishing Company, Incorporated, 3 ed., 2008.

[23] L. Liu, Y. Li, S. Li, N. Hu, Y. He, R. Pong, D. Lin, L. Lu, and M. Law, "Comparison of next-generation sequencing systems," *Journal of Biomedicine and Biotechnology*, vol. 2012, 2012.

[24] M. L. Metzker, "Sequencing technologies - the next generation.," *Nat Rev Genet*, vol. 11, pp. 31–46, Jan 2010.

[25] N. Nagarajan and M. Pop, "Parametric complexity of sequence assembly: theory and applications to next generation sequencing.," *J Comput Biol*, vol. 16, pp. 897–908, Jul 2009.

[26] S. B. Needleman and C. D. Wunsch, "A general method applicable to the search for similarities in the amino acid sequence of two proteins.," *Journal of molecular biology*, vol. 48, pp. 443–453, Mar. 1970.

[27] D. T. Pride, R. J. Meinersmann, T. M. Wassenaar, and M. J. Blaser, "Evolutionary implications of microbial genome tetranucleotide frequency biases," *Genome Research*, vol. 13, pp. 145–158, 2003.

[28] M. Rasmussen, Y. Li, S. Lindgreen, J. S. Pedersen, A. Albrechtsen, I. Moltke, M. Metspalu, E. Metspalu, T. Kivisild, R. Gupta, *et al.*, "Ancient human genome sequence of an extinct palaeo-eskimo," *Nature*, vol. 463, no. 7282, pp. 757–762, 2010.

[29] H. Teeling, A. Meyerdiekers, M. Bauer, and F. O. Glckner, "Application of tetranucleotide frequencies for the assignment of genomic fragments," *Environmental Microbiology*, vol. 6, no. 9, pp. 938–947, 2004.

[30] H. Teeling, J. Waldmann, T. Lombardot, M. Bauer, and F. O. Glckner, "Tetra: a web-service and a stand-alone program for the analysis and comparison of tetranucleotide usage patterns in dna sequences," *BMC Bioinformatics*, vol. 5, p. 163, 2004.

[31] S. Vinga and J. Almeida, "Alignment-free sequence comparison-a review.," *Bioinformatics*, vol. 19, pp. 513–523, Mar 2003.