Bioinformatics Advance Access published September 23, 2016

Subject Section

FBB: A Fast Bayesian Bound tool to calibrate RNA-seq aligners

Irene Rodriguez-Lujan[1,2], Jeff Hasty[1,3,4], Ramón Huerta[1,*]

[1] BioCircuits Institute, University of California, San Diego, La Jolla, CA 92093-0328, USA
[2] Machine Learning Group, Escuela Politécnica Superior, Universidad Autónoma de Madrid, 28049 Madrid, Spain
[3] Department of Bioengineering, University of California, San Diego, La Jolla, CA 92093, USA
[4] Molecular Biology Section, Division of Biological Science, University of California, San Diego, La Jolla, CA 92093, USA.

*To whom correspondence should be addressed.

Associate Editor: Dr. Inanc Birol

Abstract

Motivation Despite RNA-seq reads provide quality scores that represent the probability of calling a correct base, these values are not probabilistically integrated in most alignment algorithms. Based on the quality scores of the reads, we propose to calculate a lower bound of the probability of alignment of any fast alignment algorithm that generates SAM files. This bound is called Fast Bayesian Bound (FBB) and serves as a canonical reference to compare alignment results across different algorithms. This Bayesian Bound intends to provide additional support to the current state-of-the-art aligners, not to replace them.
Results We propose a feasible Bayesian bound that uses quality scores of the reads to align them to a genome of reference. Two theorems are provided to efficiently calculate the Bayesian bound that under some conditions becomes the equality. The algorithm reads the SAM files generated by the alignment algorithms using multiple command option values. The program options are mapped into the FBB reference values, and all the aligners can be compared respect to the same accuracy values provided by the FBB. Stranded paired read RNA-seq data was used for evaluation purposes. The errors of the alignments can be calculated based on the information contained in the distance between the pairs given by Theorem 2, and the alignments to the incorrect strand. Most of the algorithms (Bowtie, Bowtie 2, SHRiMP2, Soap 2, Novoalign) provide similar results with subtle variations.
Availability: Current version of the FBB software provided at `https://bitbucket.org/irenerodriguez/fbb`.
Contact: rhuerta@ucsd.edu

1 Introduction

Goal. Quality scores of next-generation sequencing (NGS) reads are not usually integrated in the alignment algorithms. In some cases, if the quality scores of the reads are low the whole read may be dropped. In (Li *et al.*, 2008), the quality scores are summed over the mismatches, bypassing probability rules and ignoring the quality of the "correct" matches. These simplifications are a compromise between processing speed and reasonable theoretical approximations. Our goal is to provide theoretical basis for the fast estimation of the posterior probability of an alignment given a read and a genome that can be used to fairly compare NGS aligners.

On fast aligners. Most of the aligners typically rely on seed-and-extend algorithms (Fonseca *et al.*, 2012; Altschul *et al.*, 1997; Langmead *et al.*, 2009; Li and Durbin, 2009). A NGS read is aligned by first finding in the reference genome a short token of the sequence and then extending the alignment to the rest of the sequence. In general, these algorithms first find the alignments with the lowest mismatch score and then they use the quality scores in the mismatching bases to provide the mapping with a quality score. Aligners such as MAQ (Li *et al.*, 2008), Bowtie (Langmead *et al.*, 2009) or RMAP (Smith *et al.*, 2008) are implementations of this approach. A different strategy is proposed by the Slider algorithm (Malhis *et al.*, 2009). It uses the *prb* files containing the quality scores for every base in every position in the read. Slider creates for every read a list of possible candidate sequences based on the *prb* files. The number of candidate reads

is limited to a given minimal probability (20%) , but then the quality scores are not used for alignment purposes.

Aligner Comparisons. Given the large variety of aligners, there is a growing body of research comparing aligners. These approaches can be divided into three groups according to the type of data used for the comparison: simulated data (Břinda *et al.*, 2016; Caboche *et al.*, 2014; Engström *et al.*, 2013; Giese *et al.*, 2014), spike-in data (Minoche *et al.*, 2011), or real/non-simulated data (Holtgrewe *et al.*, 2011; Kumar *et al.*, 2015). Simulated data adds uncontrolled sources of errors, which makes comparisons unreliable due the additional layer of uncertainty (Giese *et al.*, 2014). In this work, we adopt a more pragmatic approach in line with previous methods such as RABEMA (Holtgrewe *et al.*, 2011) or CADBURE (Kumar *et al.*, 2015), which do not need synthetic data or spike-in experiments. RABEMA provides a *gold standard* for the read mapping problem that is based on the Hamming and Levenshtein distances; however, the incorporation of mate pair information and quality values are pointed out in (Holtgrewe *et al.*, 2011) as directions of further work. CADBURE compares aligners in pairs, and then analyzes the relative reliability and consistency with each other. Our suggestion is to compare aligners directly using RNA-seq reads using a natural metric based on the estimation of a Bayesian Bound. This work is also related to assembly quality control approaches based on Bayesian statistics and quality scores. On the one hand, Detonate (Li *et al.*, 2014) is a software package that implements model-based score methods for evaluating assemblies when the ground truth is unknown. The idea behind Detonate and FBB is similar. Detonate intends to compare several transcriptome assemblies of the same set of reads by considering the posterior probability of the assembly given the reads, and FBB maximizes the posterior probability of a position in the genome given the reads. However, Detonate's probabilistic model is different from the one proposed in this work as it has to account for assembly-specific problems, and it does not explicitly incorporates quality scores. On the other hand, ALE (Assembly Likelihood Evaluation) (Clark *et al.*, 2013) is an assembly quality control approach that also calculates the likelihood of observing a specific assembly given the reads, and it obtains the contribution to this likelihood from each position in the assembly. ALE's score is used to compare different assemblers, but not as an objective function itself. While ALE's score is obtained for the assembly position and insert size for a given read, FBB optimizes the posterior probability with respect to the genome position and insert size to provide the optimal score.

Why a Bayesian Bound. The posterior probability of an alignment at position i of the genome $\mathbf{g}$ of a NGS read $\mathbf{r}$, or $P_{\mathbf{g}}(i|\mathbf{r})$, can be directly estimated by using the Bayes Theorem as $P_{\mathbf{g}}(\mathbf{r}|i)P_{\mathbf{g}}(i)/P(\mathbf{r})$. The estimation of observing a read $\mathbf{r}$ given that the originating position is i requires the quality scores in the FASTQ files and non-trivial assumptions in the generation of the read from the genomic location. To speed up the calculation instead of fully estimating $P_{\mathbf{g}}(i|\mathbf{r})$, we propose to calculate a bound of the posterior probability. We call this bound Fast Bayesian Bound (FBB) because it intends to be calculated fast. This bound also allows aligning the reads using a selected threshold. If the FBB values are used for alignment purposes, we call it Fast Bayesian Bound Alignment (FBBA) algorithm and mappings are obtained as those maximizing the FBB value for each read.

On the results. The FBB estimate uses the quality scores to provide a bounded probability of having a correct output of the multi-valued function that will be used to be mapped against the genome of reference. We propose two Theorems, that provide two alignment bounds of the posterior probability for single-end and paired-end reads that may contain mismatches. Additionally, Lemma 3 extends these bounds for reads that may include mismatches and indels. Computational costs are simplified by avoiding the calculation of the normalization of the probability of the short reads because this is a common factor to all genome positions for a given experiment. The FBB's implementation reads the SAM files generated by any alignment algorithm and provides the average FBB values for all the reads. Those average values are then used to compare all the alignment algorithms in the same reference framework. We then compare the alignment results for several non-spliced aligners, namely: Bowtie (Langmead *et al.*, 2009), Bowtie2 (Langmead and Salzberg, 2012), BWA-MEM (Li, 2013), Novoalign (`http://www.novocraft.com`), Segemehl (Hoffmann *et al.*, 2009), SHRiMP2 (Rumble *et al.*, 2009), and Soap2 (Li *et al.*, 2009). Three organisms are considered: *Saccharomyces cerevisiae*, *Mus Musculus* and *E. Coli*. In real and simulated data, BWA-MEM and Segemehl produce different qualitative results than the others and tend to underperform. All the other methods perform very similarly.

2 Methods

2.1 Problem definition

Given a genome, $\mathbf{g}$, containing N base pairs and a NGS read, $\mathbf{r}$, containing $J = |\mathbf{r}|$ base pairs, we want to calculate the probability of the position i of the short read $\mathbf{r}$ in genome $\mathbf{g}$; or $P(i|\mathbf{r}, \mathbf{g})$. The posterior probability, $P(i|\mathbf{r}, \mathbf{g})$, can be used by an aligner algorithm to decide whether (1) $\mathbf{r}$ *aligns* to $\mathbf{g}$ or not. The aligner requires to have an arbitrary threshold θ to provide an alignment answer or not; (2) if the read *aligns* to the genome, the algorithm yields the optimal alignment location i in the genome. If the location i matches the true location, then we obtain a "true positive" answer.

In this framework, the alignment algorithms can have the following main sources of errors according to (Li *et al.*, 2008): (i) the error that an aligned read does not come from the reference genome (a "false positive" answer also called *type-1 mapping error*), (ii) the error associated with an unaligned read whose true position in the genome/chromosome is missed by the alignment algorithm (a "false negative" answer), and (iii) the error that the alignment is not the true one ("false positive" answer). The estimation of error (i) is out of the scope of this paper because that would require to calculate $P(i|\mathbf{r}, \mathbf{g}_j)$ where j runs over all possible samples of the genomes present in the experiment (Reinert *et al.*, 2015).

2.2 Mapping the genome

Given a genome, $\mathbf{g}$, containing N base pairs, we can map any K consecutive bases of $\mathbf{g}$ into a single positive integer s. The function that maps K consecutive bases into the positive integers is $f : (g_i, g_{i+1}, \ldots, g_{i+K-1}) \rightarrow s$. For a N-size genome we map at most $N - K + 1$ segments into words, and we can simplify the notation as $f(i) = s$, where i denotes the position of the beginning of the K consecutive bases and s is a positive integer in the interval $[0, 2^{64}]$.

Under this framework, let $H(\cdot)$ be a **multi-valued function** from a positive integer s, which represents a genome encoded sequence, to a genome position i. Since there can be multiple repetitions of short sequences in the genome, let us define the set $I(s)$ as the set of positions that contain all the genome locations of the same sequence s. Formally,

$$I(s) = \{i/H(s) = i\} \; . \tag{1}$$

That is, $H(s) = i \Rightarrow f(g_i, g_{i+1}, \ldots, g_{i+K-1}) = s$. The number of elements in this set is larger or equal than 0, *i.e.*, $|I(s)| \geq 0$.

2.3 Mapping a NGS read to the genome using multi-valued functions

The mapping will be defined by the leftmost position in the genome i that maximizes the posterior probability of the read. The short reads contain a quality measure that informs us on the probability of having an incorrect base call in that sequence. Our approach consists in creating a search policy

that takes into account the context of the sequence and the quality of the scores. We thus want to calculate the probability of the position i of the short read, $\mathbf{r}$ in genome $\mathbf{g}$; or $P(i|\mathbf{r},\mathbf{g})$.

We first map the genome $\mathbf{g}$ in its corresponding $s^g_1, s^g_2, \ldots, s^g_{N-K+1}$ positive integer values in the range $[0, 2^{64}]$ using the encoding function $f(\cdot)$ defined above. Then, the read $\mathbf{r}$ is also mapped into a sequence of $s_1, s_2, \ldots, s_{J-K+1}$ ($J = |\mathbf{r}|$) using the encoding function $f(\cdot)$. The set $I(s)$ in Equation (1) induces some properties that facilitate operating with all the genomic locations both in the algorithms and the calculation of the lower bound of the probability of alignment. Lemma 1 formalizes the fact of having a genome location of $\mathbf{r}$, under the assumption that the NGS reads originate from the genome $\mathbf{g}$ with no errors or mutations.

Lemma 1. *Given a short read* $\mathbf{r}$ *obtained from genome* $\mathbf{g}$ *at position* i *with no errors or mutations, the intersection of all the sets is*

$$I(s_1) \cap (I(s_2) - 1) \cap \ldots \cap (I(s_{J-K+1}) - J + K) = S\,. \quad (2)$$

Then, S *is not empty, and* $i \in S$.

Proof. Since the reads $r_1, r_2, \ldots, r_J$ are originated at the genome location $g_i, g_{i+1}, \ldots, g_{i+J-1}$ with $r_j = g_{i+j-1}$ for $j = 1, \ldots, J$, then $H(f\left(\sigma^K_j(\mathbf{r})\right)) = H(f\left(\sigma^K_{i+j-1}(\mathbf{g})\right))$ with the operator σ^K_j extracting the sequence of bases from j to $j + K - 1$. Thus, we have the following cases.

i) If there is a unique repetition of $\mathbf{r}$ in the genome, then $I(f\left(\sigma^K_{i+j-1}(\mathbf{g})\right)) = i + j - 1$ for all $j = 1, \ldots, J$. Thus, $\{I(f\left(\sigma^K_i(\mathbf{g})\right))\} \cap \{I(f\left(\sigma^K_{i+1}(\mathbf{g})\right) - 1\} \cap \ldots \cap \{I(f(\sigma^K_{i+J-K}(\mathbf{g}))) - J + K\} = \{i\} \cap \{i + 1 - 1\} \cap \{i + J - K - (J - K)\} = \{i\} = S$. Since there is at least one repetition in the genome then S is not empty and $i \in S$.

ii) If there are multiple repetitions of the the sequence $\mathbf{r}$ in the genome, then $I(f(\sigma^K_{i+j-1}(\mathbf{g}))) = \{i_k + j - 1\}$ for all $j = 1, \ldots, J$ and $k = 1, \ldots, NR$, with NR the number of repetitions in the genome with $i_1 = i$ representing the original copied location. Then,

$$\begin{aligned} I(f(\sigma^K_i(\mathbf{g}))) \cap \{I(f(\sigma^K_{i+1}(\mathbf{g}))) - 1\} \cap \ldots \\ \cap \{I(f(\sigma^K_{i+J-K}(\mathbf{g}))) - J + K\} = \\ = \{i_k\}^{NR}_{k=1} \cap \{i_k + 1 - 1\}^{NR}_{k=1} \cap \ldots \\ \cap \{i_k + J - K - (J - K)\}^{NR}_{k=1} = \{i_k\}^{NR}_{k=1} = S \end{aligned}$$

$$\begin{aligned} \{i_k\}^{NR}_{k=1} \cap \{i_k + 1 - 1\}^{NR}_{k=1} \cap \ldots \\ \cap \{i_k + J - K - (J - K)\}^{NR}_{k=1} = \{i_k\}^{NR}_{k=1} = S. \end{aligned}$$

Since $i = i_1 \in \{i_k\}^{NR}_{k=1}$, then $i \in S$ and S is not empty. ■

It follows from (Dembo *et al.*, 1994; Karlin and Altschul, 1990) that for increasing $|\mathbf{r}|$, S converges to i with probability almost one. It is easy to build a deterministic algorithm based on Lemma 1 that is essentially a special case of the Bayesian bound algorithm that we propose in the next section. A deterministic algorithm that does not use the quality scores is forced to evaluate all the possible locations when $|S| > 1$ and choose one of them based on other criteria.

2.4 Lower bound of the posterior probability

As in Section 2.3, we first map the genome $\mathbf{g}$ in its corresponding $s^g_1, s^g_2, \ldots, s^g_{N-K+1}$ positive integer values in the range $[0, 2^{64}]$ using function $f(\cdot)$ as shown above. Then, we apply $f(\cdot)$ to map each short read in its encoded sequence of positive values s_1 to s_{J-K+1}. When considering possible mismatches, we need to calculate the probability of $\mathbf{r}$, mapped as a sequence of numbers s_1 to s_{J-K+1}, to be located at position i as follows

$$P(i|\mathbf{r}) = P(i|s_{J-K+1}, \ldots, s_1)\,. \quad (3)$$

Since quality scores are available, and Lemma 1 provides the framework to map genome location i to read location j via the multi-valued function, we can estimate the probability of having a value of $s_j = s$ at location j in the read, given that we are at location i in the genome. Therefore, we use the Bayes Theorem (Bayes, 1764) to be able to express $P(i|\mathbf{r})$ in terms of probabilities that we can calculate from the data. The probability of matching a genome location is

$$P(i|\mathbf{r}) = P(i|s_{J-K+1}, \ldots, s_1) = \frac{P(s_{J-K+1}, \ldots, s_1|i)P(i)}{P(s_{J-K+1}, \ldots, s_1)},$$

where i denotes the genome/chromosome position that matches the beginning of the short read as stated in Lemma 1. The prior probabilities of the short reads, $P(s_{J-K+1}, \ldots, s_1)$, are the same regardless of the genome position i. Thus, this prior probability normalizes by a constant the conditional probability values, $P(\mathbf{r}|i)$, for any genome/chromosome alignment. We denote the prior probability of the reads with the constant A and simplify the above expression to a manageable format as

$$\log P(i|\mathbf{r}) + A = \log P(i) + \log P(s_{J-K+1}, \ldots, s_1|i). \quad (4)$$

The prior probability on the genome position, $P(i)$, depends on the chromosome and/or genome size. Since we may compare short reads against different chromosome sizes within a genome, we cannot assume that $P(i)$ is a constant.

The optimal choice of the genome location i given the data can be obtained from $\arg\max_i\{\log P(i|\mathbf{r}) + A\}$. In optimization terms, an aligner should be finding $\arg\max_i\{[\log P(i|\mathbf{r}) + A - \theta']_+\}$, where $[\cdot]_+$ is the positive clip function. Since A is constant, we merge A and θ' into one new single parameter $\theta = A - \theta'$. This problem may not have a solution because some of the reads in the FASTQ files may contain low quality base calls.

Let us apply the chain rule to $P(s_{J-K+1}, \ldots, s_1|i)$ to make it more amenable for analysis:

$$\begin{aligned} &P(s_{J-K+1}, \ldots, s_1|i) = P(s_{J-K+1}, \ldots, s_2|s_1, i)P(s_1|i) = \\ &P(s_{J-K+1}, \ldots, s_{K+1}|s_K, \ldots, s_2, i)P(s_K|s_{K-1}, \ldots, s_1, i) \cdots P(s_1|i) \\ &= P(s_{J-K+1}|\{s_{\pi(J-K+1)}\}, i) \cdot P(s_{J-K}|\{s_{\pi(J-K)}\}, i) \cdots P(s_1|i), \end{aligned}$$

where the set $\pi(k)$ is defined as $\pi(k) = \{k-1, k-2, \ldots, k-K+1\}$. The reason is that two encoded sequences s_i and s_j at distance lower or equal to K have bases in common, and thus, there exists a statistical dependence between them. We can write equation (4) as

$$\log P(i|\mathbf{r}) + A = \log P(i) + \sum_{j=1}^{J-K+1} \log P(s_j|\{s_{\pi(j)}\}, i). \quad (5)$$

The exact solution of equation (5) is not straightforward to calculate for all the positions i in the genome. We will circumvent it by proposing a tight lower limit in Theorem 1. To be able to introduce Theorem 1, we first need to explain how we obtain the probability of having an element of the short read, s_j, being generated at the genome position i.

Using quality scores to build $P(s_j|i)$. To calculate expression (5), we need to estimate how good the short read is given all the sources of errors.

The probability of having an encoded integer s_j given that the read comes from position i in the genome is

$$P(s_j|i) = \begin{cases} q_j & \text{if } i \in I(s_j) - j + 1, \\ \frac{1-q_j}{4^K-1} & \text{if } i \notin I(s_j) - j + 1. \end{cases} \quad (6)$$

In the absence of any other knowledge, we assume that the rest of the genome positions i that are not in $\{I(s_j) - j + 1\}$ are evenly distributed on the remaining possible values of short sequences of size K. $P(s_j|i)$ can be calculated based on the individual probability calls. Every base pair r_i in the short read has an estimated error probability, κ_i. So, the probability of having a correct read of K consecutive bases starting at position j of the read, q_j, is computed in a similar way as in the Slider algorithm (Malhis *et al.*, 2009)

$$q_j = \prod_{i=0}^{K-1} (1 - \kappa_{i+j}) \ . \quad (7)$$

The values of q_j corresponding to each s_j value can be simultaneously calculated as the FASTQ file is read. It can be typically seen that the 5' end of the read has fewer sequencing errors than the 3' end (Hillier *et al.*, 2008). We now can introduce Lemma 2 and Theorem 1.

Lemma 2. *Given a short read* $\mathbf{r}$ *with possible mismatches and obtained from genome* $\mathbf{g}$ *at position* i*, a lower bound of the probability that a non-overlapping subset of indices* $T(o)$ *of* $\mathbf{r}$ *is aligned to the genome/chromosome location* i *is given by the following equation*

$$\log P(i|\mathbf{r}, o) + A \geq \log P(i) + \sum_{\forall j \in T(o) \ s.t. \ i \in \{I(s_j)-j+1\}} \log q_j + N_{\hat{I}}(o) \log \frac{1-Q}{4^K - 1} \ , \quad (8)$$

where $Q = \max_{j=1}^{J-K+1} q_j$*, and* $N_{\hat{I}}(o) = |\{j \in T(o) \ s.t. \ i \notin I(s_j) - j + 1\}|$*. The set* $T(o)$ *defines the non-overlapping sets of indices for* $o = 1, \ldots, K$ *such that* $\sum_{o=1}^{K} P(T(o)) = 1$ *and* $T(1) \cap T(2) \cap \ldots \cap T(K) = \emptyset$*. These non-overlapping sets can be built as* $T(o) = \{k : k = o + n \cdot K; J - K + 1 \geq k \geq 1; n \in \mathbb{N}\}$*. The above lower bound becomes the equality if and only if either of these conditions are verified*

a) $q_j = \max_{k=1}^{J-K+1}\{q_k\}$ *for* $j \in T(o)$ *s.t.* $i \notin \{I(s_j) - j + 1\}$.
b) $N_{\hat{I}}(o) = 0$.

Proof. The conditional probability $P(i|\mathbf{r}, o)$ can be written as $P(i|\mathbf{r}, o) = P(\mathbf{r}|i, o)P(i \cap o)/P(\mathbf{r} \cap o)$. Since the read offset o is independent of both the genome location i and the actual read $\mathbf{r}$, we have $P(i|\mathbf{r}, o) = P(\mathbf{r}|i, o)P(i)/P(\mathbf{r})$. Equivalently, $\log P(i|\mathbf{r}, o) + \log P(\mathbf{r}) = \log P(\mathbf{r}|i, o) + \log P(i)$. As discussed before, $P(\mathbf{r})$ is a constant, and it is denoted as A. When restricted to the subset $T(o)$, $P(\mathbf{r}|i, o)$ can be factorized since the conditional dependencies $P(s_j|\{s_{\pi(j)}\}, i)$ in Equation (5) disappear as we sample the sequences s_1 to s_{J-K+1} every K positions. Then, the problem becomes

$$\log P(i|\mathbf{r}, o) + A = \log P(i) + \sum_{j \in T(o)} \log P(s_j|i), \quad (9)$$

$$\log P(i|\mathbf{r}, o) + A = \log P(i) + \sum_{\forall j \in T(o) \text{ s.t. } i \in \{I(s_j)-j+1\}} \log q_j + \sum_{\forall j \in T(o) \text{ s.t. } i \notin \{I(s_j)-j+1\}} \log \left(\frac{1-q_j}{4^K - 1} \right) \ . \quad (10)$$

Let us call $Q = \max_{k=1}^{J-K+1} q_k$, then $\log\left(\frac{1-q_j}{4^K-1}\right) \geq \log\left(\frac{1-Q}{4^K-1}\right)$ for any index j since the logarithm is a strictly increasing function. Thus, we can rewrite

$$\log P(i|\mathbf{r}, o) + A \geq \log P(i) + \sum_{\forall j \in T(o) \text{ s.t. } i \in I(s_j)-j+1} \log q_j + N_{\hat{I}}(o) \log \frac{1-Q}{4^K - 1} \ , \quad (11)$$

where $N_{\hat{I}}(o) = |\{j \in T(o) \text{ s.t. } i \notin I(s_j) - j + 1\}|$, and we recover the equality if only if $Q = q_k \ \forall j \in T(o)$ s.t. $i \notin \{I(s_j) - j + 1\}$ or $N_{\hat{I}}(o) = 0$. The condition $N_{\hat{I}}(o) = 0$ is satisfied when $\{j \in T(o) \text{ s.t. } i \notin I(s_j) - j + 1\} = \emptyset$, which implies that there exists an exact matching (no mismatches) between the reference genome $\mathbf{g}$ and the read $\mathbf{r}$. This is a desirable and intuitive property for the bound proposed in Lemma 2.

■

Now, we can formulate Theorem 1 whose proof is based on the lower bound presented in Lemma 2.

Theorem 1. *A lower bound of the probability that a short read,* $\mathbf{r}$*, is aligned to the chromosome/genome location* i *is*

$$\log P(i|\mathbf{r}) \geq -A + \log P(i) + \frac{1}{K} \sum_{o=1}^{K} L(o) \ , \quad (12)$$

with

$$L(o) = \sum_{\forall j \in T(o) \ s.t. \ i \in I(s_j)-j+1} \log q_j + N_{\hat{I}}(o) \log \frac{1-Q}{4^K - 1}$$

and $Q = \max_{j=1}^{J-K+1} q_j$*. The lower bound becomes an equality iff all the following conditions are satisfied*

a) $q_j = \max_{k=1}^{J-K+1}\{q_k\}$ *for all* $j = 1, 2, \ldots, J-K+1$ *or* $N_{\hat{I}}(o) = 0$.
b) $L(o) = L(o')$ *for all pairs* o *and* o'.

Proof. We use the total probability Theorem (Pfeiffer, 2013) to express $\log P(i|\mathbf{r})$ in the subset of non-overlapping sets, $T(o)$ as

$$\log P(i|\mathbf{r}) = \log \sum_{o=1}^{K} P(i|\mathbf{r}, o)P(o) = \log \frac{1}{K} \sum_{o=1}^{K} P(i|\mathbf{r}, o) \ . \quad (13)$$

Given that the logarithm is a strictly concave function, we can now apply the Jensen's inequality (Jensen, 1906) to move the logarithm inside the summatory, and we obtain

$$\log P(i|\mathbf{r}) \geq \frac{1}{K} \sum_{o=1}^{K} \log P(i|\mathbf{r}, o) \ , \quad (14)$$

which is the equality if $P(i|\mathbf{r}, o) = P(i|\mathbf{r}, o')$ for all pairs o, o'. Using equation (10) in Lemma 2 and replacing $\log P(i|\mathbf{r}, o)$ into equation (14) leads to equation (12). $P(i|\mathbf{r}, o) = P(i|\mathbf{r}, o')$ for all pairs o, o' is verified under the same conditions of Lemma 2 and $\sum_{j \in T(o) \text{ s.t. } i \in \{I(s_j)-j+1\}} \log q_j + N_{\hat{I}}(o) \log \frac{1-Q}{4^K-1} = \sum_{j \in T(o') \text{ s.t. } i \in \{I(s_j)-j+1\}} \log q_j + N_{\hat{I}}(o') \log \frac{1-Q}{4^K-1}$ for all pairs o and o'.

■

In terms of the practical implementation of the algorithms derived from Theorem 1 and making use of the notation employed in Lemma 1 for the sets $I(s)$, it is algorithmically easy to track the set of indices $\{i\}$ of the union of $I(s_1) \cup (I(s_2)-1) \cup \ldots \cup (I(s_J)-J+1)$. Since we may have some subsets $I(s_j)$ that can be empty, we should track all the possible indices of the union instead of the intersection. Moreover, we lack a fast operator that can access/store the indexes complementary to the union of the set because it is large. This is why we impose the lower bound in Lemma 2 by calculating Q. Another important aspect is that despite $P(i)$ depends on the number of base pairs in the chromosomes roughly as $-\log N_c$, where N_c is the size of chromosome c, the search can be reduced to the set of admissible genomic locations, i, by limiting the algorithm to the coding regions of the genome.

Finally, in terms of the conditions for equality, in most of the reads the q_j values do not vary more than 1% from each other which means that the inequality from equation (10) to (11) is fairly close to the equality. The condition b) for equality in Theorem 1 includes a broad range of cases. Interestingly, a sufficient but not necessary condition to have this equality is to have an exact matching (Lemma 1) and q_k constant for all $k = 1, \ldots, J-K+1$. For example, an exact matching with no errors or mutations in the reads (Lemma 1). This property shows the consistency of the bound as it shows that the better the quality of the read, the tighter the bound.

Using Theorem 1, we can now define the alignment algorithm based on the FBB as the following optimization problem

$$G(\theta) = \arg\max_i \{[\log P(i) + \frac{1}{K}\sum_{o=1}^{K}\left[\sum_{\forall j \in T(o) \text{ s.t. } i \in I(s_j)-j+1} \log q_j + N_{\hat{I}}(o)\log\frac{1-\max_j q_j}{4^K-1}\right] - \theta\Big]_+\Big\}.$$

Note that θ is a parameter that has to be set by data validation for a subsample of all the reads. The reason A vanishes from the optimization problem is because it is a constant. It follows from

$$\arg\max_i\{[\log P(i|\mathbf{r}) - \theta]_+\} = \arg\max_i\{[\log P(i|\mathbf{r}) + A - \theta']_+\}. \quad (15)$$

Whenever we use the FBB as an optimization algorithm based on the parameter θ, we will denote it by FBBA. Note that the FBBA algorithm implements a *best-hit* reporting policy.

2.5 Lower bound of the posterior for paired-end reads

The previous section allows us to obtain a lower bound of the posterior probability of the genome location for one single-end read. It is possible to extend Lemma 2 and Theorem 1 to evaluate the posterior probability of paired-end reads given the probability of the distance between the paired-end reads. In particular, we want to calculate the probability that the paired-end read given by $\mathbf{r}$ and $\mathbf{r}'$ aligns to the locations i and $i+d$, respectively, where d represents the distance between the leftmost (5') end of $\mathbf{r}$ and $\mathbf{r}'$. Thus,

$$P(i, i+d|\mathbf{r}, \mathbf{r}') = \frac{P(\mathbf{r}, \mathbf{r}'|i, i+d)P(i, i+d)}{P(\mathbf{r}, \mathbf{r}')}. \quad (16)$$

We can now formulate the next Theorem:

Theorem 2. *A lower bound of the probability that a paired-end read given by* $\mathbf{r}$ *and* $\mathbf{r}'$ *aligns to the chromosome/genome locations* i *and* $i+d$*, respectively, is*

$$\log P(i, i+d|\mathbf{r}, \mathbf{r}') + B \geq \log P(i) + \log P(d) + \frac{1}{K}\sum_{o=1}^{K}\left[\sum_{\forall j \in T(o) \text{ s.t. } i \in I(s_j)-j+1} \log q_j + N_{\hat{I}}(o)\log\frac{1-Q}{4^K-1}\right] + \frac{1}{K}\sum_{o=1}^{K}\left[\sum_{\forall j \in T(o) \text{ s.t. } i+d \in I(s'_j)-j+1} \log q'_j + N'_{\hat{I}}(o)\log\frac{1-Q'}{4^K-1}\right].$$

with $\mathbf{r}$ *mapping to* $s_1, \ldots, s_{J-K+1}$ *with the associated quality values* $q_1, \ldots, q_{J-K+1}$*, and* $\mathbf{r}'$ *mapping to* $s'_1, \ldots, s'_{J-K+1}$ *with the associated quality values* $q'_1, \ldots, q'_{J-K+1}$*. Moreover,* $Q = \max_{j=1}^{J-K+1} q_j$ *and* $Q' = \max_{j=1}^{J-K+1} q'_j$*.*

Proof. i) The prior probability $P(\mathbf{r}, \mathbf{r}')$ is independent of the genome location i and the genome/chromosome. Therefore, in terms of the optimization problem, we can assume $P(\mathbf{r}, \mathbf{r}') = B$ with B being a constant.

ii) $P(i+d, i) = P(i+d|i)P(i)$, but $P(i+d|i) = P(d)$ which is the probability of the distance between the paired-end reads. Thus, $\log P(i+d, i) = \log P(d) + \log P(i)$.

iii) Given that we are copying with some errors the location i and $i+d$ from the genome using the process provided in equation (6), we can write $P(\mathbf{r}, \mathbf{r}'|i, i+d) = P(\mathbf{r}|i)P(\mathbf{r}'|i+d)$.

We apply i), ii) and iii) into equation (16) to obtain

$$\log P(i, i+d|\mathbf{r}, \mathbf{r}') = -B + \log P(i) + \log P(d) + \log P(\mathbf{r}|i) + \log P(\mathbf{r}'|i+d) \quad (17)$$

If we apply Lemma 2 and the Jensen's Inequality of Theorem 1, we recover inequality (17) that becomes the equality if conditions a) and b) in Theorem 1 are satisfied for $\mathbf{r}$ and $\mathbf{r}'$.

■

We iteratively estimate $P(d)$ with one single pass over the data. More details can be found in Supplementary Material, Section 1. We can now formulate the following definitions to measure false paired-end alignments/mappings.

Definition 1 (**False positive paired-end alignments**). *A pair of reads* $\mathbf{r}$ *and* $\mathbf{r}'$ *are a false positive paired-end alignment/mapping, if they both are the best single-end alignment according to Equation (15) when considered independently, but they are not the optimum value for Equation (17) in Theorem 2 when considered as a pair.*

Definition 2 (**False positive paired-end alignments from SAM**). *A pair of reads* $\mathbf{r}$ *and* $\mathbf{r}'$ *are a false positive paired-end alignment/mapping, if they both are paired match in the SAM file, but they are not the optimum value for Equation (17) in Theorem 2 when considered as a pair.*

2.6 Extending FBB to indels

Insertions and/or deletions (indels) in NGS reads have a natural extension in our formalism. These are the two main changes with respect to the probabilistic framework presented in the preceding sections: i) the multi-valued function, $H(\cdot)$ can be expanded to contain indels on all the K consecutive bases of the genome, and ii) there will be an offset of the originating location of the genome due to the indel; the originating location of the genome does not match when an indel has occurred at some location in the NGS read according to our function from the read of K bps to the genomic location $I(s_j) - j + 1$. The probability of having a correct

read of size K shown in Equation 7 does not need to be modified as the q-scores provided with reads already give an estimate of the error probability, disregarding whether the source of error is either a mutation or an indel.

In terms of the Bayesian formalism the main modification is induced by ii), because the map from the extracted K-sequence read to the genome location $I(s_j) - j + 1$ becomes shifted if there exists indels at any of the coded K sequences s_1 to s_{j-1}. For example, if the NGS read contains δ_- deletions in the middle of the read, then the last encoded sequence of the read, s_{j^*} with $j^* = J - K + 1$, has δ_- base shifts with respect to the genuine genome location given by $I(s_1)$; that is, $I(s_{j^*}) - j^* + 1 = I(s_1) - \delta_-$. In general, let $\delta(j) = \delta_+(j) - \delta_-(j)$ be the difference between the accumulated insertions and deletions up to position j in the encoded read $\{s_1, s_2, \ldots, s_{j^*}\}$. Then, we can rewrite Lemma 2 as follows.

Lemma 3. *Given a short read* $\mathbf{r}$ *with possible mismatches and/or indels, and obtained from genome* $\mathbf{g}$ *at position i, a lower bound of the probability that a non-overlapping subset of indices* $T(o)$ *of* $\mathbf{r}$ *is aligned to the genome/chromosome location* i *is given by the following equation*

$$\log P(i|\mathbf{r}, o) + A \geq \log P(i) + \sum_{\substack{\forall j \in T(o) \\ s.t.\ i \in \{I(s_j) - j + 1 + \delta(j)\}}} \log q_j + N_{\hat{I}}(o) \log \frac{1-Q}{4^K - 1},$$

where $Q = \max_{j=1}^{J-K+1} q_j$, *and* $N_{\hat{I}}(o) = |\{j \in T(o)\ s.t.\ i \notin I(s_j) - j + 1 + \delta(j)\}|$. *The set* $T(o)$ *defines the non-overlapping sets of indices for* $o = 1, \ldots, K$ *such that* $\sum_{o=1}^{K} P(T(o)) = 1$ *and* $T(1) \cap T(2) \cap \ldots \cap T(K) = \emptyset$. *These non-overlapping sets can be built as* $T(o) = \{k : k = o + n \cdot K; J - K + 1 \geq k \geq 1; n \in \mathbb{N}\}$. *The scalar function* $\delta(j) = \delta_+(j) - \delta_-(j)$ *is the difference between the accumulated insertions and deletions up to* $j \in T(o)$. *The above lower bound becomes the equality if and only if either of these conditions are verified*

a) $q_j = \max_{k=1}^{J-K+1} \{q_k\}$ *for* $j \in T(o)$ *s.t.* $i \notin \{I(s_j) - j + 1 + \delta(j)\}$.
b) $N_{\hat{I}}(o) = 0$.

Proof. The proof of Lemma 3 follows from the proof of Lemma 2 by replacing the mapping function from $i \in \{I(s_j) - j + 1\}$ to $i \in \{I(s_j) - j + 1 + \delta(j)\}$.

■

Note that Theorem 2 and Lemma 3 build on Lemma 2. We only have to replace $i \in \{I(s_j) - j + 1 + \delta(j)\}$ by $i \in \{I(s_j) - j + 1\}$ to make them consistent with the indel mapping function. To calculate $\delta(j)$ note the following. Given two independent (non-overlapping) K-mers starting at the read positions j and j' and with encoding sequences s_j and $s_{j'}$, respectively, the probability that $s_{j'}$ aligns by chance within $\pm l$ bases from $I(s_j)$ is $P(I(s_{j'}) \in [I(s_j) - l, I(s_j) + l]) = \frac{2l+1}{N}$ with N the length of the reference genome. Note that any pair of indices j and j' in the partition $T(o) = \{k : k = o + n \cdot K; J \geq k \geq 1, n \in \mathbb{N}\}$ defined in Lemmas 2 and 3 satisfies this property for a given offset o. Each of the partitions $T(o)$ for $o = 1, 2, \ldots, K$ has at least $M = \lfloor \frac{J-K+1}{K} \rfloor$ non-overlapping $\{s_j\}_{j \in T(o)}$. Then, the probability that exactly $m \leq M$ indices in $T(o)$ fall within $\pm l$ of $I(s_1)$ by chance is $P(m, I(s_1)) = \binom{M}{m} \left(\frac{1+2l}{N}\right)^m \left(1 - \frac{1+2l}{N}\right)^{M-m}$, and the probability that at least $m \leq M$ indices in $T(o)$ fall within $\pm l$ of $I(s_1)$ by chance is $P'(m, I(s_1)) = \sum_{k=m}^{M} \binom{M}{k} \left(\frac{1+2l}{N}\right)^k \left(1 - \frac{1+2l}{N}\right)^{M-k}$. For example, if the NGS reads have $J = 100$, and we chose $K = 14$ the probability that 50% of the encoded $\{s_j\}$ are within ± 10 bps from the location of $I(s_1)$ is $2 \cdot 10^{-13}$ for a genome with $N \approx 10^6$. When extending the location interval to $l = 100$, the probability that 50% of the encoded $\{s_j\}$ are within ± 100 bps from the location of $I(s_1)$ is $2 \cdot 10^{-10}$. Thus, if we use Lemma 2 and we can find at least 50% of $T(o)$ within 10 bps, then it is very likely that there exists one or more indels in the NGS read. The method to correct for the indels is straightforward now. It consists in finding the nearest and existing $P(i|\mathbf{r}, o)$ and assign all the nearby contributions to the optimal location i.

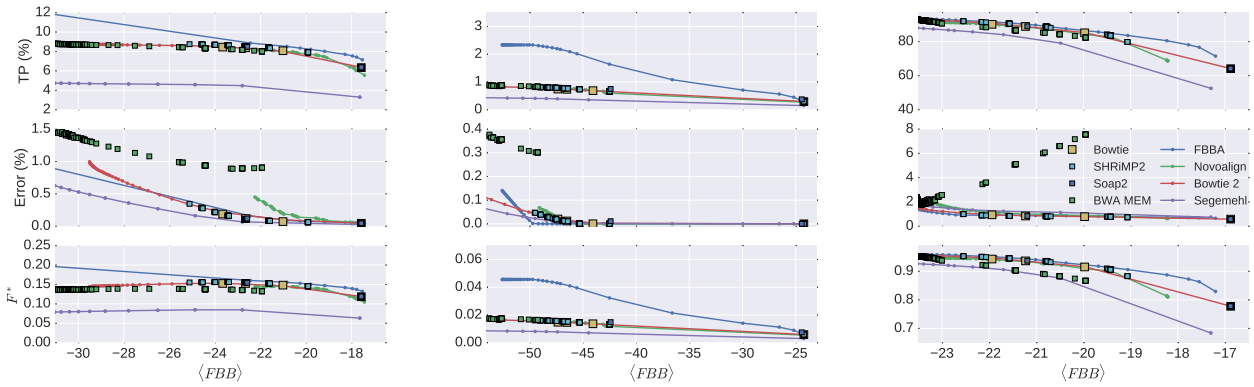
3 Results

We evaluate the performance of several non-spliced well-known aligners under different parameter settings (see Supplementary Material, Section 2) in real and simulated RNA-seq data and using the FBB quantity as reference. The aligners we compared are Bowtie (Langmead *et al.*, 2009), Bowtie2 (Langmead and Salzberg, 2012), BWA-MEM (Li, 2013), Novoalign, Segemehl (Hoffmann *et al.*, 2009), SHRiMP2 (Rumble *et al.*, 2009), and Soap2 (Li *et al.*, 2009). We considered non-spliced aligners as the multi-valued function is designed for contiguous nucleotides, so when working with RNA-seq reads, it is expected to provide better results for the transcriptome. Nevertheless, non-spliced aligners (and FBBA) can also be used to align reads against a reference genome, but reads that span an intron will not be aligned, which may lead to underestimate gene expression. However, in those cases where the reference genome is poorly annotated and splice variants are unknown, the use of FBBA or other non-spliced aligners may be useful. Finally, it should be remarked that the FBB score and the FBBA aligner are based on a *best-hit* strategy, and, thus, aligners' reporting modes have been configured accordingly.

3.1 Results on real data

We calculate the alignment on RNA-seq data from the *Saccharomyces cerevisiae* for wild type, RNA-seq experiments from heart cells of the *Mus Musculus*, and RNA-seq data from *E. Coli* exposed to Copper. We randomly subsampled one million reads from each of the experiments in each study. These RNA-seq data was selected because it contained paired-end and stranded reads. The stranded paired-end reads from these experiments allow us to determine: a) if the aligned pairs are correct according to Definition 1 (or 2), and b) if the mRNA fragment is aligned to the incorrect genome strand. If the paired-end reads align to the wrong genome strand, then the alignment might be considered incorrect or, at least, be flagged for further analysis. Based on these errors, we can obtain an estimate of the F1-score, an accuracy measure commonly used in information retrieval and defined as the harmonic mean of the precision and recall (Salton and McGill, 1986). If we denote by A the alignment as a percentage provided by each of the algorithms under comparison, and we denote by TP and E the percentage of true positive and false positive errors, respectively, provided by the FBB framework, then the precision of each algorithm is $P = TP/(TP + E)$. The recall measure R requires estimating the number of false negatives that is lower or equal than $100 - A$, thus the recall measure satisfies $R \geq A/100$. The F1-score is, therefore, lower-bounded by $F \geq 2\frac{P}{1 + \frac{100}{TP+E}} = F^*$, and we can use F^* as proxy for the F1-score. This estimated F1-score allows evaluating the performance of each alignment algorithm with respect to $\langle FBB \rangle$, the average over the FBB values provided by Theorem 2 and Lemma 3 for each read $\mathbf{r}$. From now on, we considered that two aligners map a read to the same genome position if their positions are at distance of 2 nucleotides at most. Computational times of FBBA and FBB on these datasets are provided in the Supplementary Material, Section 4.

Results for the Saccharomyces cerevisiae data (Fig. 1a). Results for *Saccharomyces cerevisiae* – chromosome XV, and data from Bioproject number PRJNA275812 are presented. In all cases, the percentage of false positive paired-end alignments, computed according to Definition 1 (or 2), decreases as $\langle FBB \rangle$ increases. The algorithm based on Theorem 2 filters them out because it is controlling for the distance between pairs during the



(a) *Saccharomyces cerevisiae*. (b) *Mus Musculus*. (c) *E. Coli*.

Fig. 1: Alignment comparison of the Bowtie, SHRiMP2, Soap 2, BWA MEM, FBBA, Novoalign, Bowtie 2, and Segemehl on the (a) *Saccharomyces cerevisiae* genome - Chromosome XV, (b) *Mus Musculus* transcriptome - Chromosome XVIII, and (c) *E. Coli* genome as a function of the average FBB values, $\langle FBB \rangle$. Results show the average values over all the experiments in each dataset. (top) Percentage of true positive pairs according to FBB framework. (middle) Percentage of false positive pairs according to Definition 1 (or 2). For genomic mappings (a and c), the percentage of alignments to genes on the opposite strand is also included in the error rate. (bottom) F^*, a proxy of the F1-score (see text for more details).

optimization. The percentage of strand-incorrect alignments is calculated following the annotations in the corresponding gff file.

After projecting the command parameters of all the alignment methods and parameter configurations into the average Bayesian bound $\langle FBB \rangle$, it is noticeable that most methods reach similar alignment percentage values for a given $\langle FBB \rangle$ except for Segemehl. Segemehl (Hoffmann *et al.*, 2009) was shown to be a robust method on simulated data (Caboche *et al.*, 2014); however, for this dataset it is not as competitive (see Supplementary Material, Section 2 for the options used). BWA MEM shows a competitive proxy F1-score (F^*), but it has the largest error rates. The stricter parameters of Soap 2 (`-M 0`), Bowtie 2 (`-score-min C,0`) and Bowtie (`-v 0`) lead to almost the same $\langle FBB \rangle$ location (-17.57) for the three panels.

Results for the Mus Musculus data (Fig. 1b). To compare with a larger genome, we present the results for *Mus Musculus* data (Bioproject number PRJNA244374, accession numbers SRR2032135-38), chromosome XVIII. In this case, alignment against *Mus Musculus* transcriptome was considered to show FBB and FBBA flexibility. Error rates in this case are only referred to the false positive pairs (Definitions 1 and 2), as locations in the annotation file are referred to positions in the genome. However, FBB and FBBA can also be applied when using *Mus Musculus* genome as reference given their computational efficiency. All the alignment methods except Segemehl and FBBA generate similar performance. FBBA generates lower errors because by construction it filters false positive pairs using Theorem 2, it may be obtaining larger alignment rates than the other methods as it does not impose any filter in terms of maximum number of substitutions and indels. Segemehl generates the lowest alignment rates and moderate error rates, which leads Segemehl to produce the lowest F^* as in the case of *Saccharomyces cerevisiae* data. BWA MEM does not yield large $\langle FBB \rangle$ values. Some parameter configurations produce competitive F^* scores, but high false positive rates.

Results for the E. Coli data (Fig. 1c). We compared the alignment methods with the densely annotated *E. coli* K-12 MG1655 genome using RNA-seq data obtained after *E. coli* was exposed to copper. BWA-MEM can perform similarly as the rest on the F^* metric for low FBB values and only for some command option parameters. In contrast, for the largest FBB values, its performance is behind most aligners due to its high false positive rate. On the other hand, Segemehl has similar error rates than most of the mappers, but its alignment rate is worse; therefore, its F1-score estimate F^* is the lowest one. All the rest of the algorithms are fairly close to each other in this case, perhaps because it is a prokaryote.

Overall, most aligners considered in this work, parametrized as described in the Supplementary Material, Section 2, yield similar results in terms of alignment, error rates, and, F^* for a given value of the average FBB score, $\langle FBB \rangle$. This shows that $\langle FBB \rangle$ is a good Bayesian-based *gold standard* to compare aligners and parameter settings. However, Segemehl's proxy F1-score is lower than that of other mappers in all cases, as it produces less alignments than other methods and similar error rates. BWA MEM also presents a different behavior when compared to other aligners. While it has been shown to be relatively close to other mappers in the *Saccharomyces cerevisiae* and *E. Coli* data in terms of $\langle FBB \rangle$ values, its $\langle FBB \rangle$ scores are generally much lower for *Mus musculus*. When providing comparable $\langle FBB \rangle$ scores, its error rates are always the largest.

Comparison with other studies. First of all, it should be noted that it is difficult to relate our results to those in other comparative studies in the literature mainly because of (i) different reporting modes, and (ii) different parameter settings. For example, results in (Caboche *et al.*, 2014) are not fully comparable with the ones here presented as they are based on *all-hits* reporting mode. Regarding differences in the parameter configuration, in this work we swept a broad spectrum of parameters and compare aligners based on the FBB score; however, in other works, mappers' parameters are generally set to the default values, which is likely to produce distant $\langle FBB \rangle$ scores for different aligners and hinder their comparison. This is the case of RABEMA's experimental setup (*any-best* reporting mode) (Holtgrewe *et al.*, 2011). Nevertheless, some of our results are in line with that obtained in (Holtgrewe *et al.*, 2011), where Bowtie and Soap2 provide similar performances with low error rates in general. This is also observed in our results. Though Shrimp2 is the worst method in (Holtgrewe *et al.*, 2011), we obtained rates very similar to Bowtie and Soap2. As already discussed, differences may be due to the different parameters settings used. The BWA aligner used in RABEMA is different from BWA MEM, making the results incomparable in this sense. Moreover, the comparison of BWA MEM against Bowtie 2 is consistent with (Břinda

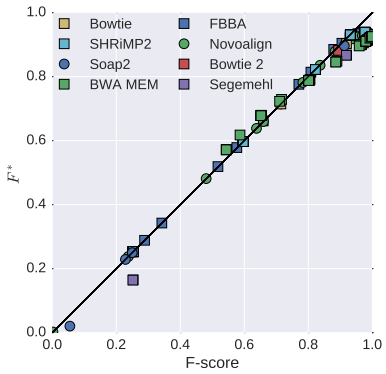


Fig. 2: Scatter plot of the real F1-score (F1-score) versus the proxy F1-score F^* for different aligners (Bowtie, SHRiMP2, Soap 2, BWA MEM, FBBA, Novoalign, Bowtie 2, and Segemehl) on simulated reads. Results correspond to ART paired-end read simulations (Huang *et al.*, 2012) on *E. Coli* transcriptome with default parameters.

et al., 2016), where BWA MEM produces larger False Discovery Rates than Bowtie 2 at the same level of alignment. Finally, in order to provide a fair comparison under the same experimental setup, we used simulated data to compare FBB to Cadbure (Kumar *et al.*, 2015), one of the state-of-the-art approaches to compare aligners that does not need simulated data either. As it is shown in Supplementary Material, Section 3.1, Cadbure and FBB yield similar results in terms of F1-score, but Cadbure does not provide a *gold standard* Bayesian measure for each aligner, and its computational cost is significantly higher than that of FBB.

3.2 Results on simulated data

Though using simulated data does not necessarily clarify the comparison across alignment algorithms(Yu *et al.*, 2012; Kumar *et al.*, 2015), simulations provide us with a ground truth to show that FBB indeed helps in identifying incorrect alignments, and to determine whether the F^* score estimate is accurate. We used the ART simulator for next-sequencing reads (Huang *et al.*, 2012). Simulated reads of *Escherichia coli str. K-12 substr. MG1655* transcriptome with length 100 were generated by using the Illumina's HiSeq 2000 sequencing system with a 20-fold read coverage. ART generates unstranded simulated reads and, therefore, only the FBB estimate for false positive reads (Definition 2) is considered as source of error. More details on the experimental setup and results on simulated data can be found in the Supplementary Material, Section 3. To summarize, results on simulated data show that FBBA algorithm provides a F1-score very close to 1 when using built-in, technology-specific read error models and base quality scores. This means that maximizing the FBB score is an effective strategy that leads to identify the vast majority of correct alignments, while having a very low false positive rates. Additionally, the analysis of the relationship between the real F1-score and the F1-score proxy F^* (Fig. 2) reveals that F^* is indeed a good proxy for the true F1-score. It is remarkable that F^* tends to underestimate the true F1-score for Segemehl and BWA MEM, which partially explicates the results obtained in Section 3.1.

4 Conclusions

We propose a new Bayesian bound for any NGS alignment algorithm that generates SAM files. The proposed bound called Fast Bayesian Bound (FBB) is presented in Lemmas 2-3 and Theorem 2, ant it fully integrates the quality scores to evaluate the alignment. It is designed to be as fast as possible for execution, and it is shown to be close to equality under certain conditions. This bound can be used as a canonical reference to evaluate and compare existing alignment methods, whose good performance usually depends on properly tuning several parameters. The FBB can be integrated in alignment algorithms that use indexing to map K-mers to genomic locations, and, as example, we propose a Fast Bayesian Bound Algorithm (FBBA) based on the maximization of the proposed bound.

The FBB reference values allow exploring many command options of the scoring functions of NGS aligners by projecting the scoring options of each algorithm into a single reference value in $\langle FBB \rangle$, representing the average over the FBB values provided by Theorem 2 and Lemma 3 for each short read. Moreover, FBB does not require to filter reads by quality scores to improve consistency, which is a common practice in many other algorithms. The FBB estimate can be used as an additional check of the alignment properties of different algorithms. If the FBB cannot be adapted to the particular requirements of the fast alignment methods, we provide the software to calculate the average FBB values from the aligned SAM files.

In our experiments on RNA-seq data to compare several aligners under different configuration parameters and using *best-hit* reporting mode, we used paired-end RNA-seq data because this type of data provides a natural way to detect errors by determining how erroneous two aligned pairs are based on a definition of false positive alignments (Definitions 1 and 2) that relies on the proposed Bayesian bound. For stranded data, we can also determine if the aligned pairs point at the incorrect strand of the genome. These error estimations allow us to calculate a proxy of the F1-score (F^*) that is used for comparison purposes across alignment algorithms. Though FBB estimates permits direct cross-comparisons without making assumptions on the generative process of the reads, we have also generated paired-end simulated reads in order to have a ground truth to show that (i) the FBB score is able to identify incorrect alignments, and (ii) the proxy F^* is a satisfactory estimate for the true F1-score. Simulated data also allowed us to compare FBB to Cadbure to conclude that both algorithms lead to similar F1-scores estimates in most cases. In the results section, we show that most alignment algorithms achieve similar performance results except for the BWA-MEM algorithm that produces higher level of errors, and Segemehl that provides lower alignment rates and fails to reach the same level of achievement than other mappers. Overall, most aligners yield very similar results for a given FBB score, which indicates that it is a very competent Bayesian-based gold standard to compare different aligners under different parameter settings.

Acknowledgements

The authors gratefully acknowledge the use of the facilities of Centro de Computación Científica (CCC) at Universidad Autónoma de Madrid.

Funding

This work has been supported by DARPA under grant HR0011-15-2-0046. I.R-L acknowledges partial support by Spain's grants TIN2013-42351-P (MINECO), S2013/ICE-2845 CASI-CAM-CM (Comunidad de Madrid).

References

Altschul,S.F., Madden,T.L., Schäffer,A.A., Zhang,J., Zhang,Z., Miller,W. and Lipman,D.J. (1997) Gapped LAST and PSI-BLAST: a new generation of protein database search programs. *Nucleic acids research,* **25** (17), 3389–3402.

Bayes,T. (1764) An Essay Toward Solving a Problem in the Doctrine of Chances. *Philosophical Transactions of the Royal Society of London,* **53**, 370–418.

Břinda,K., Boeva,V. and Kucherov,G. (2016) RNF: a general framework to evaluate NGS read mappers. *Bioinformatics,* **32** (1), 136–139.

Caboche,S., Audebert,C., Lemoine,Y. and Hot,D. (2014) Comparison of mapping algorithms used in high-throughput sequencing: application to Ion Torrent data. *BMC genomics,* **15** (1), 264.

Clark,S.C., Egan,R., Frazier,P.I. and Wang,Z. (2013) ALE: a generic assembly likelihood evaluation framework for assessing the accuracy of genome and metagenome assemblies. *Bioinformatics,* **29** (4), 435–443.

Dembo,A., Karlin,S. and Zeitouni,O. (1994) Limit distribution of maximal non-aligned two-sequence segmental score. *The Annals of Probability,* **22** (4), 2022–2039.

Engström,P.G., Steijger,T., Sipos,B., Grant,G.R., Kahles,A., Rätsch,G., Goldman,N., Hubbard,T.J., Harrow,J., Guigó,R. *et al.* (2013) Systematic evaluation of spliced alignment programs for RNA-seq data. *Nature methods,* **10** (12), 1185–1191.

Fonseca,N.A., Rung,J., Brazma,A. and Marioni,J.C. (2012) Tools for mapping high-throughput sequencing data. *Bioinformatics,* **28** (24), 3169–3177.

Giese,S.H., Zickmann,F. and Renard,B.Y. (2014) Specificity control for read alignments using an artificial reference genome-guided false discovery rate. *Bioinformatics,* **30** (1), 9–16.

Hillier,L.W., Marth,G.T., Quinlan,A.R., Dooling,D., Fewell,G., Barnett,D., Fox,P., Glasscock,J.I., Hickenbotham,M., Huang,W. *et al.* (2008) Whole-genome sequencing and variant discovery in C. elegans. *Nature methods,* **5** (2), 183–188.

Hoffmann,S., Otto,C., Kurtz,S., Sharma,C.M., Khaitovich,P., Vogel,J., Stadler,P.F. and Hackermüller,J. (2009) Fast mapping of short sequences with mismatches, insertions and deletions using index structures. *PLoS Comput Biol,* **5** (9), e1000502.

Holtgrewe,M., Emde,A.K., Weese,D. and Reinert,K. (2011) A novel and well-defined benchmarking method for second generation read mapping. *BMC bioinformatics,* **12** (1), 1.

Huang,W., Li,L., Myers,J.R. and Marth,G.T. (2012) ART: a next-generation sequencing read simulator. *Bioinformatics,* **28** (4), 593–594.

Jensen,J.L.W.V. (1906) Sur les fonctions convexes et les inégalités entre les valeurs moyennes. *Acta Mathematica,* **30** (1), 175–193.

Karlin,S. and Altschul,S.F. (1990) Methods for assessing the statistical significance of molecular sequence features by using general scoring schemes. *Proceedings of the National Academy of Sciences,* **87** (6), 2264–2268.

Kumar,P.K.R., Hoang,T.V., Robinson,M.L., Tsonis,P.A. and Liang,C. (2015) CADBURE: A generic tool to evaluate the performance of spliced aligners on RNA-Seq data. *Scientific reports,* **5**.

Langmead,B. and Salzberg,S.L. (2012) Fast gapped-read alignment with Bowtie 2. *Nature methods,* **9** (4), 357–359.

Langmead,B., Trapnell,C., Pop,M. and Salzberg,S.L. (2009) Ultrafast and memory-efficient alignment of short DNA sequences to the human genome. *Genome biol,* **10** (3), R25.

Li,B., Fillmore,N., Bai,Y., Collins,M., Thomson,J.A., Stewart,R. and Dewey,C.N. (2014) Evaluation of de novo transcriptome assemblies from RNA-Seq data. *Genome biology,* **15** (12), 1.

Li,H. (2013) Aligning sequence reads, clone sequences and assembly contigs with BWA-MEM. *arXiv preprint arXiv:1303.3997,* **1303**.

Li,H. and Durbin,R. (2009) Fast and accurate short read alignment with Burrows–Wheeler transform. *Bioinformatics,* **25** (14), 1754–1760.

Li,H., Ruan,J. and Durbin,R. (2008) Mapping short DNA sequencing reads and calling variants using mapping quality scores. *Genome research,* **18** (11), 1851–1858.

Li,R., Yu,C., Li,Y., Lam,T.W., Yiu,S.M., Kristiansen,K. and Wang,J. (2009) SOAP2: an improved ultrafast tool for short read alignment. *Bioinformatics,* **25** (15), 1966–1967.

Malhis,N., Butterfield,Y.S., Ester,M. and Jones,S.J. (2009) Slider—maximum use of probability information for alignment of short sequence reads and SNP detection. *Bioinformatics,* **25** (1), 6–13.

Minoche,A.E., Dohm,J.C. and Himmelbauer,H. (2011) Evaluation of genomic high-throughput sequencing data generated on Illumina HiSeq and genome analyzer systems. *Genome biology,* **12** (11), 1.

Pfeiffer,P.E. (2013) *Concepts of probability theory*. Courier Corporation.

Reinert,K., Langmead,B., Weese,D. and Evers,D.J. (2015) Alignment of Next-Generation Sequencing Reads. *Annual review of genomics and human genetics,* **16** (0), 133–151.

Rumble,S.M., Lacroute,P., Dalca,A.V., Fiume,M., Sidow,A. and Brudno,M. (2009) SHRiMP: accurate mapping of short color-space reads. *PLoS Comput Biol,* **5** (5), e1000386.

Salton,G. and McGill,M.J. (1986) *Introduction to modern information retrieval*. McGraw-Hill, Inc.

Smith,A.D., Xuan,Z. and Zhang,M.Q. (2008) Using quality scores and longer reads improves accuracy of Solexa read mapping. *BMC bioinformatics,* **9** (1), 128.

Yu,X., Guda,K., Willis,J., Veigl,M., Wang,Z., Markowitz,S., Adams,M.D. and Sun,S. (2012) How do alignment programs perform on sequencing data with varying qualities and from repetitive regions? *BioData mining,* **5** (1), 6.