BRAT-BW: Efficient and accurate mapping of bisulfite-treated reads [Supplemental Material]

Elena Y. Harris[1], Nadia Ponts[2,3], Karine G. Le Roch[2] and Stefano Lonardi[1]

[1]Department of Computer Science and Engineering, University of California, Riverside, CA 92521

[2]Department of Cell Biology and Neuroscience, University of California, Riverside, CA 92521

[3]INRA, MycSA UR 1264, 71 Avenue Edouard Bourlaux, BP81, 33883 Villenave d'Ornon Cedex, France

1 FEATURE COMPARISON

Common indexes used in mapping short are hash tables or the FM-index (Ferragina and Manzini, 2000) that is based on the Burrows-Wheeler transform (Burrows and Wheeler, 1994). Table 1 shows a comparison of the main feature of the tools discussed in our manuscript, namely, BRAT-bw, BRAT (brat-large), Bismark and BS-seeker. BRAT (brat-large) is based on hash indexing, whereas the other tools use the FM-index. Table 2 summarizes the main differences between BRAT (brat-large) and BRAT-bw. Table 3 provides time complexity for the problem of mapping short reads to a reference genome for mapping tools that use hash indexing or the FM-index. The length of the seed used in hashing is fixed and usually shorter than the seed for BWT. Hence, the total number *Occ* of occurrences of a seed with the FM-index is usually much smaller than with hash indexing. Since we need to check a smaller number of full-length read alignments with the FM-index the mapping is more time-efficient.

2 BS-MAPPING WITH BRAT-BW

BRAT-BW uses the strategy proposed in (Lister *et al.,* 2009) and employed by both Bismark and BS-seeker. Two FM-indexes are built on the positive strand of the reference genome: in the first, Cs are converted to Ts, and in the second, Gs are converted to As. Refer to Figure 1 and Figure 2 below for an example of the discussion that follows.

There are two distinct types of bisulfite libraries: the first type yields sequenced reads that are bisulfite-converted versions of two original genomic strands (Lister *et al.,* 2009); the second type produces reads that correspond to four possible strands, as a byproduct of PCR step (Cokus *et al.,* 2008). To map reads from Lister *et al.* BS-library, BRAT-bw converts Cs to Ts in original reads and aligns the resulting reads to the first index (which is the original forward genomic strand with Cs converted to Ts); this corresponds to the alignment of reads from PCR_1^+ strand on Figure 1. Then BRAT-bw computes the reverse-complement of the original read, changes Gs to As in the reverse-complement, and then maps the resulting read to the second index (the original forward genomic strand with Gs converted to As); this corresponds to the alignment of reads from PCR_2^- strand in Figure 1.

For the library by Cokus *et al.,* we also align the reads only to the forward strand of the original genome (using the same two indexes with Cs converted to Ts and another one with Gs converted to As). For this type of library, in addition to two alignments described above, we perform two additional alignments. We convert Gs to As in an original read and align it to the second index (which is original genomic forward strand with Gs converted to As). This corresponds to the alignment of reads from PCR_2^+ strand on Figure 1. Next, we convert Cs to Ts in the reverse-complement of an original read and align it to the first index. This corresponds to the alignment of reads from PCR_1^- strand on Figure 1.

For any library, when there is a match for at least two out of two/four (for typeI/typeII BS-libraries) possible alignments for a read, then this read is considered to be ambiguous and is not reported in the output.

Methylation level is estimated for two strands separately: for cytocines on the forward strand we count the number of Cs and Ts in reads mapped to Cs in the genome using any alignment to the first index; and for cytocines on reverse strand we count the number of Gs and As in reads mapped to Gs in the genome using any alignments to the second index.

A read could be ambiguous due to repeats. If a read shows two or more best score alignments, it is regarded as ambiguous and not provided in the output. Best score alignments are those that have the same smallest number of mismatches.

3 MULTI-SEED MAPPING

BRAT-bw uses the FM-index to map reads. The FM-index is a data structure obtained by lexicographically sorting the list of all possible circular shifts of a given text (in our case, the reference genome). Let's us call M the sorted matrix. Each row of M corresponds a distinct suffix of the reference genome. In order to search for a read of length P using the FM-index the algorithm works backward from the end of the read. Two pointers sp and ep to the rows of M are maintained during the search. The values sp and ep for the current suffix $[i…P]$ of the read denote the first and the last rows of M that have prefixed matching exactly suffix $[i…P]$. If $sp < ep$, then there exist more than one genomic locations where this suffix $[i…P]$ occurs. If $sp = ep$, then suffix $[i…P]$ of the read occurs only once, and when $sp > ep$, there is no location within the genome where this suffix occurs and the search can be terminated.

BRAT-bw aligns longer reads using a multi-seed approach. It is easy to prove that in order for a read of length P to match a genomic location with k mismatches, there must be at least one substring in the read of length $P/(k+1)$ that matches exactly the reference genome. BRAT-bw makes ***K*** attempts (option ***-K***) to align read from different locations within the read starting from the end of the read and intending to find the substring (or a seed) within the read that aligns perfectly. In the first attempt, BRAT-bw starts from the end of the read, aligns the read base by base until there is a unique match (suffix of the read or entire read aligns uniquely to the genome), or until $sp > ep$ for some base at i-th position. Thus, the suffix $[i…P]$ of the read is the smallest suffix that does not align perfectly in a genome. If a suffix of the read aligns to one or more genomic locations, BRAT-bw performs full-length read alignments and disregards alignments with a number of mismatches greater than the threshold provided with option ***-m***. If suffix $[i…P]$ of the read cannot being mapped exactly, but is longer than 31 bp, BRAT-bw makes additional full-length read alignments to the genomic locations corresponding to the last valid values of sp and ep. In other words, if suffix $[i…P]$ has not been mapped exactly, BRAT-bw performs full-length alignment for the genomic locations to which the suffix $[i+1…P]$ of the read mapped exactly.

The number of attempts is controlled by option ***-K***. The next attempt is performed ***D*** bases to the left of the previous attempt (see Figure 3), and this process is repeated until all attempts have been completed. BRAT-bw attempts perfectly align the first ***f*** bases (this parameter is controlled by BRAT-bw, computed from the read length) in order to find a match or to identify ambiguous reads. If option ***F*** is set to 1, then BRAT-bw substitutes each base of the first ***f*** bases with the other two possible bases, and for each substitutions, aligns the first ***f*** bases (containing a substitution); if there is a match, it performs a full-length alignment to the corresponding genomic locations.

4 SIMULATED READS AND OPTIONS USED

Simulated reads can be downloaded from http://www.cs.ucr.edu/~stelo/pub/simulated_reads.tar.bz2

We tried our best to select parameters for each tool to ensure the best balance between mapping accuracy, number of mapped reads and speed whenever possible. We also tried to use similar parameters in different tools, to make the comparisons as fair as possible. Table 4 provides the full list of options used with all the tools discussed in our paper on simulated reads.

Bismark has options "best" and k=2 set as default options with Bowtie.

BRAT-bw has built-in options similar to D with Bowtie (interval between seeds in multi-seed alignment) and similar to most_valid_alignment (or "best" and k=2). D with BRAT-bw is set automatically for each reads (reads might be of variable length) for best performance in terms of mapping accuracy and time; and BRAT-bw automatically finds best alignments and outputs only the best alignment for each read.

Below we provide commands used to build indexes.

To build a genome index with Bismark and bowtie2:

bismark_v0.6.3/bismark_genome_preparation --bowtie2 --path_to_bowtie bowtie2-2.0.0-beta5 human_genome/ > out_bismark_build2

To build a genome index with Bismark and bowtie1:

bismark_v0.6.3/bismark_genome_preparation --path_to_bowtie bowtie-0.12.7 human_genome/ > out_bismark_build1

Please note that we made some changes to *bismark_genome_preparation* with bowtie-1 and bowtie-2, in particular, we set *offrate* option with bowtie to 4.

To build a genome index with BS-seeker:

time ./Preprocessing_genome.py -t N --p ../bowtie-0.12.7 --f human_genome.fa > log_hum_ref_build

Please note that we made changes to *Preprocessing_genome.py*, in particular, we set option *offrate* with bowtie to 4 and removed *-p* option with bowtie (multithreading)

To build a genome index with BRAT-bw:

./build_bw -P path_to_index -G 1 -r file_with_human_references.txt

./build_bw -P path_to_index -G 2 -r file_with_human_references.txt

Format of input files with simulated reads for BRAT-bw:
Read <int> <int>
<int> above is 0, these specify that 0 bases were trimmed from a read at the beginning and end of each read.
Format of files with genomic positions, pos_reads*.txt, where the reads were generated from:
<Read_ID>, <chrom_name>, <strand>, <position within the chromosome starting with 0>
Example:
0 chr1 + 77377303

5 SUPPLEMENTARY TABLES

Table 1. Feature comparison of BRAT-bw, BRAT-large, Bismark and BS Seeker (as of on March, 2012).

Feature	BRAT-bw	BRAT-large	Bismark	BS-seeker
Number of FM-instances (typeI/typeII bisulfite libraries)	2	NA	4	2/4
Paired-end (PE) support	Yes	Yes	Yes	No
Variable read length	Yes	Yes	Yes	Yes*
Adjustable insert size (PE)	Yes	Yes	Yes	NA
Uses basecall qualities for FastQ mapping	No	No	Yes	No
Supports typeI/typeII bisulfite libraries	Yes/Yes	Yes/No	Yes/Yes	Yes/Yes
Number of mismatches per read	Unlimited	Unlimited	Unlimited	3
Number of mismatches in a seed (bowtie-1/bowtie-2 if applicable)	1	0	3/1	3/NA
Supports indels	No	No	Yes**	No

* user has to provide the maximum read length with option e

** with bowtie-2 (we did not test this feature)

Table 2. Feature comparison between BRAT and BRAT-bw.

	BRAT, 2009	BRAT-BW, 2012
Method	Hash indexing	BWT and FM-index
Number of genome indexes	Single index on positive strand only	First index: positive strand with Cs converted to Ts Second index: positive strand with Gs converted to As
Number of read seeds	1 seed, the first 24-64bp	Multi-seeds of variable length
Number of mismatches in a seed	0 mismatches in the first 24bp	1 mismatch in the first 32-64bp
Number of mismatches per read	Unlimited	Unlimited
Read length	Unlimited	Unlimited
Advantages	Uses half as much space	Maps more reads, Higher mapping accuracy, Faster

Table 3. Time complexity for mapping tools based on hash indexing versus tools based on BWT and FM-index.

Method:	Hash indexing	BWT and FM-index
Seed length, k	fixed	variable
Time to find Occ genomic positions with occurrences of a seed of length k in genome of size N	O(1)	O(k + Occ·log(N))
Time to align entire read of length P to Occ genomic positions	O(Occ·P)	O(Occ·P)

Table 4. Options used with the tools on *in silico* reads.

Read length, bp	Mism	BRAT	BRAT-bw	Bismark with Bowtie-1	Bismark with bowtie-2	BS-seeker
36	1	m=1,S	m=1,F=1	n=1,l=36	most_valid_alignments=2,L=32,D=2,R=1,n=1	m=1,e=36
50	1	m=1,S	m=1,F=1	n=1,l=50	most_valid_alignments=2,L=32,D=2,R=1,n=1	m=1,e=50
75	2	m=2,S	m=2	n=2,l=75	most_valid_alignments=2,L=32,D=10,R=1, score_min=L,0,-0.2	m=2,e=75
100	2	m=2,S	m=2	n=2,l=100	most_valid_alignments=2,L=32,D=10,R=1, score_min=L,0,-0.2	m=2,e=100
250	5	m=5,S	m=5	n=3,l=100,e=150	most_valid_alignments=2,L=32,D=10,R=1, score_min=L,0,-0.2	m=3,e=150

6 SUPPLEMENTARY FIGURES

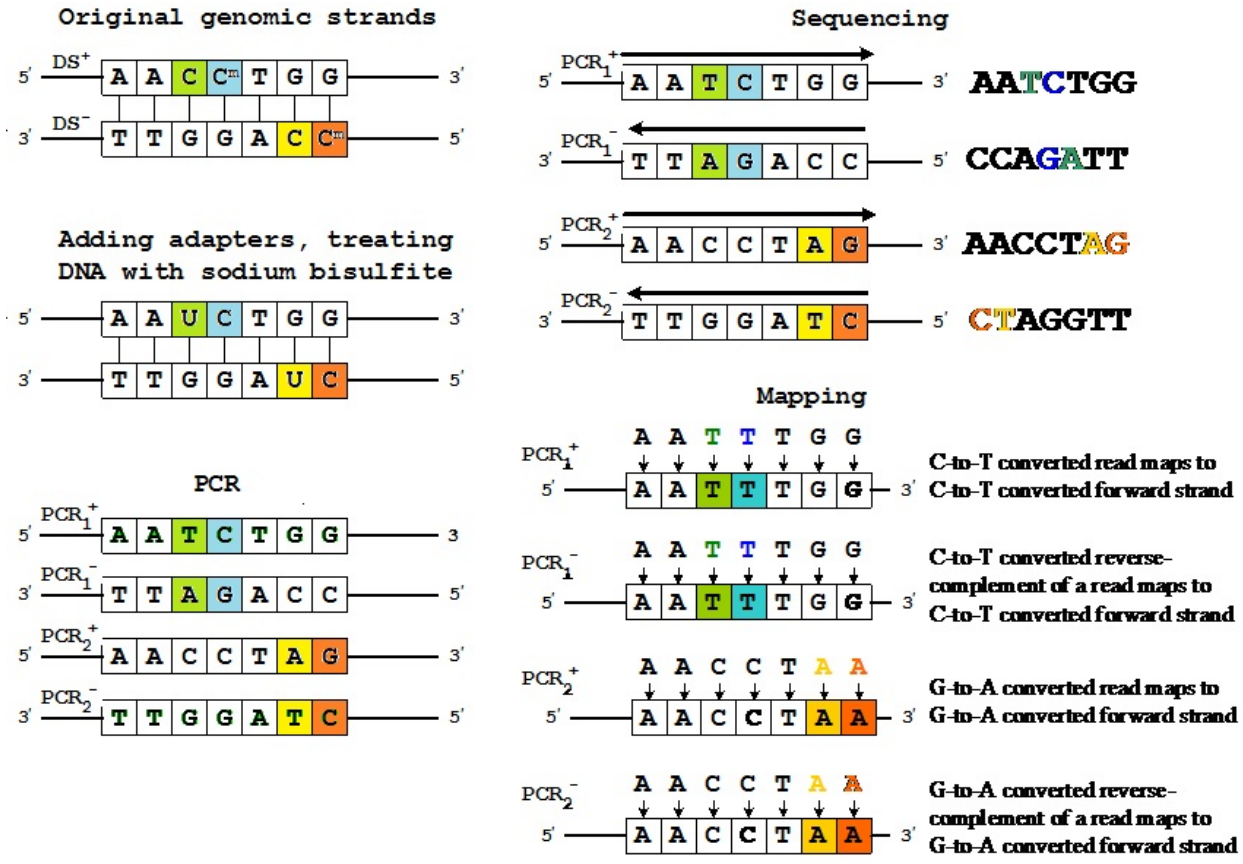

Figure 1. PCR_1^+ and PCR_2^- strands correspond to the original positive and negative genomic strands respectively (BS library by Lister et al. has only these two strands), and PCR_1^- and PCR_2^+ strands are reverse complements of PCR_1^+ and PCR_2^- respectively. Methylated and unmethylated cytosines on positive (forward) strand are shown in blue and green respectively. Methylated and unmethylated cytosines on negative strand are shown in orange and yellow respectively. In similar colors are shown converted versions (converted with sodium busilfite, further with PCR, and during alignment).

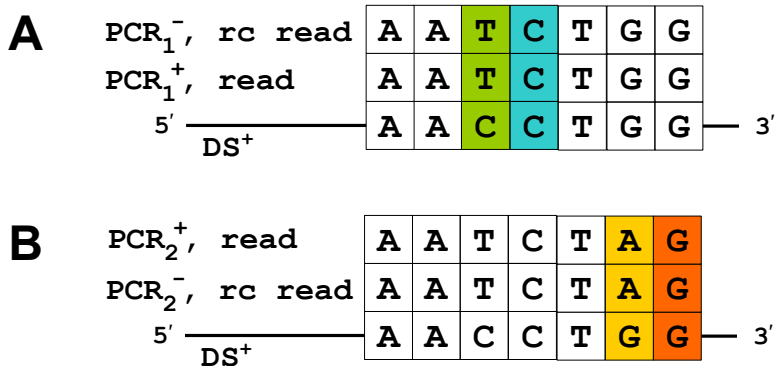

Figure 2. Methylation level is estimated for two strands separately. (A) For cytosines on forward strand we count the number of Cs and Ts in reads (or their reverse-complements) mapped to Cs in the genome using alignments to the first index. Here methylated cytosines are shown in blue and unmethylated in green. (B) For cytosines on reverse strand we count the number of Gs and As in reads (or reverse-complement of reads) mapped to Gs in the genome using alignments with the second index. Methylated cytosine on the reverse strand is shown in orange and unmethylated in yellow.

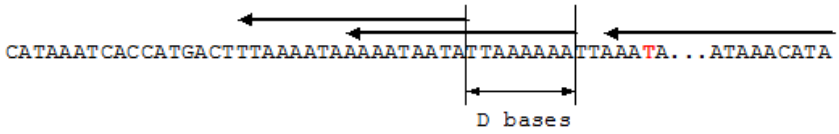

Figure 3. Multi-seed alignment with BRAT-bw. Here we show multiple attempts to align a read (each attempt is shown by an arrow). A sequenced error (or a mismatch) is shown in red. All attempts that cover this mismatch will either fail or result in wrong alignments, but other attempts that align substrings of the read without mismatches will identify correct alignment(s). ***D*** determines the interval in bases between consecutive alignments (8 bases here). D is set by BRAT-bw dynamically dependent on read length, but for the reads longer than 200bp this option is user defined.