BIOINFORMATICS APPLICATIONS NOTE Vol. 25 no. 21 2009, pages 2841–2842
doi:10.1093/bioinformatics/btp533

Sequence analysis

Updates to the RMAP short-read mapping software

Andrew D. Smith[1,*], Wen-Yu Chung[2], Emily Hodges[2], Jude Kendall[2], Greg Hannon[2], James Hicks[2], Zhenyu Xuan[2] and Michael Q. Zhang[2,*]

[1]Molecular and Computational Biology, University of Southern California, Los Angeles, CA and [2]Cold Spring Harbor Laboratory, Cold Spring Harbor, NY, USA

Received on July 11, 2009; revised on August 19, 2009; accepted on September 3, 2009

Advance Access publication September 7, 2009

Associate Editor: Limsoon Wong

ABSTRACT

Summary: We report on a major new version of the RMAP software for mapping reads from short-read sequencing technology. General improvements to accuracy and space requirements are included, along with novel functionality. Included in the RMAP software package are tools for mapping paired-end reads, mapping using more sophisticated use of quality scores, collecting ambiguous mapping locations and mapping bisulfite-treated reads.

Availability: The applications described in this note are available for download at http://www.cmb.usc.edu/people/andrewds/rmap and are distributed as Open Source software under the GPLv3.0. The software has been tested on Linux and OS X platforms.

Contact: andrewds@usc.edu; mzhang@cshl.edu

The RMAP algorithm was introduced by (Smith *et al.*, 2008) as one of the earliest available programs for mapping reads from the Illumina second-generation sequencing technology. One important contribution of RMAP was to incorporate the use of quality scores directly into the mapping process: read positions with too low a quality score were not considered while mapping, and that quality score cutoff could be adjusted by the user. Subsequently, numerous mapping algorithm have appeared (Langmead *et al.*, 2009; Li,H. *et al.*, 2008; Li,R. *et al.*, 2008; Lin *et al.*, 2008; Schatz, 2009; Yanovsky *et al.*, 2008), with improvements in both efficiency and breadth of functionality (e.g. ability to map paired-end reads; integrated SNP calling). Investigators requiring solutions to mapping problems now have many options. As new applications of short-read sequencing emerge, many variations on the analysis task of read mapping emerge. Diversity in performance characteristics of existing mapping tools becomes potentially valuable.

We report the first major update to RMAP. The basic algorithmic framework in RMAP is still to preprocess reads and scan the genome, but several modifications have been made and much additional functionality has been included. Importantly, RMAP has a memory footprint that depends on the number of reads being mapped. This feature allows RMAP to be used effectively in cluster environments with commodity nodes, because partitioning the reads allows natural parallelizations with linear reduction in memory requirements per processor core used.

Included in this release of the RMAP software package is functionality for mapping paired-end reads, making more sophisticated use of quality scores, collecting mapping locations for ambiguously mapping reads and mapping bisulfite-treated reads.

1 GENERAL MAPPING ALGORITHM

1.1 Layered seeds

Originally RMAP used the filtration method of (Baeza-Yates and Perleberg, 1996). Filtration algorithms for approximate matching first identify locations in the genome where seeds (substrings of the reads) match exactly, which can eliminate many potential matches from consideration very rapidly. The updated version now uses the idea of layered seed structures, which is similar to multiple filtration (Pevzner and Waterman, 1995). Seed structures indicate sets of positions in the reads that are required to match the genome exactly at any location where the read can map. Two distinct sets of seed structures are obtained with the property that if two strings approximately match, then each set of seed structures will contain at least one structure indicating positions that match exactly between the two strings. The two distinct sets of seed structures are combined by taking the union of the positions they specify, creating a new set of seed structures corresponding to the cartesian product of the original sets. These layered seed structures are more numerous, leading to an increased number of scans of the genome. However, the layered seed structures are also more specific, and therefore each genome scan excludes more full comparisons and is more efficient.

Use of seeds in the filtration step proceeds by representing the seed as a bit-mask that selects a subset of the bases in the reads, which are also represented as bit-masks. The result of this operation is an unsigned integer value determined by the bases in the read at seed positions. As each seed structure is processed, a hash table is constructed to index all the reads based on the result of applying the structure to the read sequences. Collisions are resolved by chaining, and each chain indicates the set of reads with specific bases at the seed structure positions. To hash values resulting from applying a seed structure to a 2-bit sequence representation, we use the modulo function of the size of the hash table, which is maintained at sizes that are prime numbers to assist in balancing chain sizes.

1.2 Use of quality scores

Base-calling quality scores are generally derived from some probabilitistic model that describes probabilities for each base at each position in a read. Quality scores are usually assigned separately for each base at a given read position, but may be summarized as a score for the position itself (generally measuring confidence in the called base at that position). For example, the original Illumina pipeline produced quality scores in the range of -40 to 40, with at most one base at each position receiving a score of ≥ 0. The precise meaning of quality scores depends on the base-calling method, and it is not desirable for mapping methods to be too closely coupled to any particular base caller.

Weight-matrix matching: in this mode RMAP uses quality scores to weigh different possible mismatches at a given position so that mappings with

*To whom correspondence should be addressed.

non-consensus bases in the genomic sequence are penalized less if they are closer in score to the consensus base at the same position. In this way we generally penalize less for mismatches at positions with less confident base calls, but also are sensitive to situations in which the base caller has difficulty deciding between two bases at a position. Specified in the appropriate format, RMAP can use any values as quality scores, enabling it to work with novel base-calling methods.

More formally, let c denote the quality score of the consensus base at a particular position, and let m denote the least quality score at that position. Then for any base at that position, if the corresponding quality score is b then the penalty associated with that base is $(c-b)/(c-m)$. For example, assume the greatest (smallest) possible score in the dataset is 40 (-40). If the consensus base at a position receives a score of 40, and all others receive scores of -40, then a mismatch at that position will be equated with a score penalty of 1. So the penalty for the consensus base is always 0, and for a perfect consensus, the penalty for a mismatch is 1.

Wildcard matching: originally RMAP could use base-call quality scores through a user-specified cutoff, which designated read positions as either high- or low-quality. The low-quality positions always induce a match (acting as wildcards); mismatches are only counted at high-quality positions (and reads are prescreened to ensure that they have some reasonable number of high-quality positions). A similar 'wildcard' scoring option is available in the current release of RMAP. Using the notation introduced in the previous paragraph, at each position, if the probability for base b is less than a user-specified value, then that position will induce a mismatch when aligned to a genomic position with base b. In this way, positions with very low quality will never be penalized (similar to the original use of quality scores in RMAP), but positions with close calls between two bases will match either of the high-scoring bases.

2 READS FROM BISULFITE SEQUENCING

Functionality for mapping reads from bisulfite sequencing has been included. Bisulfite sequencing converts all Cs in reads to Ts, except those protected by methylation (which, in mammals, generally happens at CpGs), and is the gold-standard for interrogating CpG methylation status. The common strategy in mapping bisulfite-treated reads is to map to a converted genome, where all Cs are converted to Ts. Bisulfite treatment is harsh, and a balance must be struck between converting as many unmethylated cytosines as possible and retaining sufficient fragments at the appropriate sizes. Therefore, along with the methylated Cs at CpGs, anywhere from 1–5% of the remaining Cs may be unconverted by chance (depending on experimental parameters used). RMAP is able to use information in unconverted Cs to expand the portion of the genome that can be interrogated. Wildcard matching is used to allow Ts in reads to match either C or T in the genome. Cs in reads are not allowed to map over Ts in the reference *unless* the T is followed by a G. This last condition is critical for exploiting unconverted cytosines while ensuring that no bias is introduced toward increased mappability of reads showing more methylation.

3 PAIRED-END READS

Originally, RMAP did not have functionality for paired-end reads. The current version does map paired-end reads, either as read sequences or using full quality-score information. There are two modes for paired-end mapping. In the fully sensitive mode, RMAP simultaneously maps both read ends requiring that both ends map within a user-specified range and with appropriate relative orientation. A more efficient mode identifies candidate mappings independently for each end, and joins the ends with candidate mappings falling in the specified distance range.

4 PERFORMANCE EVALUATION

Similar to the evaluation by Smith *et al.* (2008) of the original version of RMAP, we profiled RMAP performance using data from resequencing bacterial artificial chromosomes (BACs) with well-known genomic origin (hg18). We measured *coverage* as the proportion of bases in the target region covered by at least 10 reads (using the first base of each read). *Enrichment* is the proportion of mappable reads mapped inside the target region. The dataset contained 6 721 851 reads of 36 bases each, and the target region size was 162 829 bases. Results are presented in Table 1 comparing RMAP using both wildcard matching (WC) and weight-matrix matching (Q) with the original version (orig) using mismatch counts and the original method of using quality scores (Q). The score column indicates number of allowed mismatches (fractional values arise from use of quality scores as described above). Only unique mappings were considered. The results show clear improvements in the coverage and reads mapped to target for a fixed value of enrichment, demonstrating the importance of using full quality score information at each read position.

Table 1. Performance of RMAP for different scoring modes

Program	Score cutoff	Coverage	Enrichment	Mapped in target
RMAP (orig)	1	0.699	0.966	2 490 598
	2	0.750	0.958	3 071 044
RMAP (orig; Q)	0	0.589	0.978	1 829 939
	1	0.742	0.975	2 871 745
RMAP (WC)	0	0.740	0.974	3 375 148
	1	0.766	0.965	3 791 773
RMAP (Q)	1.75	0.611	0.980	2 064 228
	2.0	0.652	0.976	2 273 794

In terms of speed, using the basic scoring method, RMAP is capable of mapping 8 M reads/h, fully sensitive to two mismatches (2 M/h fully sensitive to three mismatches) on a single Intel® Xeon (2.5 GHz) processor core when mapping 50 M total reads to the human genome.

Funding: National Institutes of Health (U01 ES017166 to M.Q.Z. and W-.Y.C.; T32 CA00917631 to E.H.); Department of the Army (W81XWH04-1-0477 to J.H.); The Breast Cancer Research Foundation (to J.H.); Howard Hughes Medical Institute (to G.H.).

Conflict of Interest: none declared.

REFERENCES

Baeza-Yates,R.A. and Perleberg,C.H. (1996) Fast and practical approximate string matching. *Information Processing Letters*, **59**, 21–27.

Langmead,B. *et al*. (2009) Ultrafast and memory-efficient alignment of short DNA sequences to the human genome. *Genome Biol.*, **10**, R25.

Li,H. *et al*. (2008) Mapping short dna sequencing reads and calling variants using mapping quality scores. *Genome Res.*, **18**, 1851–1858.

Li,R. *et al*. (2008) SOAP: short oligonucleotide alignment program. *Bioinformatics*, **24**, 713–714.

Lin,H. *et al*. (2008) Zoom! zillions of oligos mapped. *Bioinformatics*, **24**, 2431–2437.

Pevzner,P.A. and Waterman,M.S. (1995) Multiple filtration and approximate pattern matching. *Algorithmica*, **13**, 135–154.

Schatz,M.C. (2009) CloudBurst: highly sensitive read mapping with mapReduce. *Bioinformatics*, **25**, 1363–1369.

Smith,A. *et al*. (2008) Using quality scores and longer reads improves accuracy of Solexa read mapping. *BMC bioinformatics*, **9**, 128.

Yanovsky,V. *et al*. (2008) Read mapping algorithms for single molecule sequencing data. In *Proceedings of the 8th International Workshop on Algorithms in Bioinformatics*. Vol. 5251, LNCS, Springer, pp. 38–49.