Bioinformatics, 34(15), 2018, 2673–2675
doi: 10.1093/bioinformatics/bty167
Advance Access Publication Date: 15 March 2018
Applications Note

Sequence analysis

Arioc: GPU-accelerated alignment of short bisulfite-treated reads

Richard Wilton[1,*], Xin Li[2], Andrew P. Feinberg[3,4,5] and Alexander S. Szalay[1,6]

[1]Department of Physics and Astronomy, Johns Hopkins University, Baltimore, MD, USA, [2]School of Medicine, Sun Yat-sen University, Guangdong, China, [3]Department of Medicine, Johns Hopkins University School of Medicine, Baltimore, MD, USA, [4]Department of Mental Health, Johns Hopkins Bloomberg School of Public Health, Baltimore, MD, USA, [5]Department of Biomedical Engineering, Johns Hopkins Whitehead School of Engineering, Baltimore, MD, USA and [6]Department of Computer Science, Johns Hopkins University, Baltimore, MD, USA

*To whom correspondence should be addressed.

Associate Editor: Bonnie Berger

Received on December 18, 2017; revised on March 6, 2018; editorial decision on March 12, 2018; accepted on March 14, 2018

Abstract

Motivation: The alignment of bisulfite-treated DNA sequences (BS-seq reads) to a large genome involves a significant computational burden beyond that required to align non-bisulfite-treated reads. In the analysis of BS-seq data, this can present an important performance bottleneck that can be mitigated by appropriate algorithmic and software-engineering improvements. One strategy is to modify the read-alignment algorithms by integrating the logic related to BS-seq alignment, with the goal of making the software implementation amenable to optimizations that lead to higher speed and greater sensitivity than might otherwise be attainable.

Results: We evaluated this strategy using Arioc, a short-read aligner that uses GPU (general-purpose graphics processing unit) hardware to accelerate computationally-expensive programming logic. We integrated the BS-seq computational logic into both GPU and CPU code throughout the Arioc implementation. We then carried out a read-by-read comparison of Arioc's reported alignments with the alignments reported by well-known CPU-based BS-seq read aligners. With simulated reads, Arioc's accuracy is equal to or better than the other read aligners we evaluated. With human sequencing reads, Arioc's throughput is at least 10 times faster than existing BS-seq aligners across a wide range of sensitivity settings.

Availability and implementation: The Arioc software is available for download at https://github.com/RWilton/Arioc. It is released under a BSD open-source license.

Contact: richard.wilton@jhu.edu

Supplementary information: Supplementary data are available at *Bioinformatics* online.

1 Introduction

Detailed analysis of DNA cytosine methylation patterns is central to the understanding of epigenetic regulation mechanisms. Next-generation DNA sequencing methods for the direct sequencing of sodium bisulfite-treated DNA (BS-seq) have become the 'gold standard' for studying genome-wide DNA methylation with single-base resolution (Adusumalli *et al.*, 2015). As the use of this whole genome bisulfite sequencing (WGBS) technology becomes increasingly widespread, the cost of sequencing a single human genome at 30-fold coverage continues to decrease toward $1000 (van Nimwegen *et al.*, 2016), and the number of public multi-gigabyte BS-seq datasets is rapidly growing.

The first step in analyzing the data generated by a WGBS sequencing run is read alignment, the process of determining the

point of origin of each sequencing read with respect to a reference genome. BS-seq read alignment is algorithmically complex and time consuming (Supplementary Appendix A1), to the point where the time spent in executing read-alignment software approaches that of the DNA sequencing run itself.

To address the need for time- and resource-efficient read alignment, a number of attempts have been made to develop software that exploits the parallel processing capability of general-purpose graphics processing units, or GPUs, including SOAP3-dp (Luo *et al.*, 2013) and nvBowtie (http://nvlabs.github.io/nvbio). GPUs are video display devices whose hardware and system-software architecture also supports general purpose computing. They are well suited to software implementations where independent computations on many thousands of data items can be carried out in parallel. This was the primary motivation for the development of the first version of Arioc (Wilton *et al.*, 2015), a high-throughput GPU-based read aligner.

There are additional computational challenges in aligning DNA sequencing reads when the DNA has been treated with bisulfite so as to differentiate methylcytosine from cytosine residues in the DNA sequences. After bisulfite-treated DNA is sequenced, the resulting BS-seq short reads must be aligned to a reference genome in a manner that identifies each methylcytosine occurrence in the context of its neighboring bases. The new, extended version of Arioc presented here extracts this information from BS-seq reads using software techniques that provide for efficient GPU acceleration.

2 Implementation

The Arioc aligner is written in C++ and compiled for both Windows (with Microsoft Visual C++) and Linux (with the GNU C++ compiler). The implementation runs in a single computer on a user-configurable number of concurrent CPU threads and on one or more CUDA-capable Nvidia GPUs (Nvidia Corporation, 2017). The implementation pipeline uses 38 different CUDA kernels written in C++ (nongapped and gapped alignment computation, application-specific list processing) and about 150 calls to various CUDA Thrust APIs (sort, set reduction, set difference, string compaction).

Arioc aligns reads by first extracting short subsequences (seeds) from each read. It then uses lookup hash tables to identify reference-sequence locations at which each seed subsequence appears in the reference sequence (genome). For BS-seq alignments, the lookup tables are constructed by first converting all Cs in the reference sequence to Ts.

Seed-and-extend alignment. Alignment proceeds for each bisulfite-treated read by similarly converting Cs to Ts in the read sequence. Arioc then uses its CT-converted lookup tables to find reference-sequence locations at which to compute alignments. It scores alignments by performing a base-by-base comparison of the original (not CT-converted) bisulfite-treated read with the original reference sequence; a T in the read sequence may match either C or T in the reference. This ensures that the base mapping (as reported in the SAM CIGAR field) and alignment score are correctly computed, regardless of the presence of C_m (methylcytosine) in the read sequence.

Arioc performs read alignment in two passes. It first attempts nongapped spaced-seed alignment (Chen *et al.*, 2009) in order to quickly identify read sequences that differ from the reference by no more than a few mismatches, without gaps (insertions or deletions). For reads that have an insufficient number of nongapped mappings, Arioc computes alignments at each candidate location using the Smith-Waterman algorithm (Gotoh, 1982; Smith and Waterman, 1981). Because Arioc performs each alignment on a separate GPU thread, this computationally-intensive procedure benefits greatly from GPU acceleration.

Arioc implements other heuristics that improve throughput. In particular, for each read Arioc sorts its list of alignment locations and prioritizes locations where multiple seeds cluster together. Additionally, for paired-end reads, Arioc compares the sorted lists for both mates to identify locations that meet distance and orientation criteria for proper (concordant) mappings. These sort, scan, and reduction operations are well suited to GPU acceleration.

Identifying methylation sites. Once Arioc has identified the set of high-scoring mappings to report, it performs a base-by-base comparison of each mapped read sequence with the corresponding region in the reference sequence (genome). This procedure follows a model similar to the one implemented in Bismark (Krueger and Andrews, 2011): Arioc establishes a methylation context (CpG, CHG, CHH, CHN, CN) for each Cm by examining the two subsequent bases in the read sequence. It reports the position and methylation context of each identified Cm in a character-string map associated with the read sequence (emitted as an optional XM field in SAM-formatted alignment results).

3 Performance

We used the human reference genome release 38 (https://www.ncbi.nlm.nih.gov/assembly/GCF_000001405.32/) for all experiments. For experiments with simulated data, we used Sherman (http://www.bioinformatics.babraham.ac.uk/projects/Sherman/) to generate 100 nt paired-end reads. For experiments with BS-seq sequencer reads, we used data from an 'Omics catalogue of lung adenocarcinoma cell lines' (https://www.ebi.ac.uk/ena/data/view/DRR016653) and from a hepatocellular carcinoma cultivar (https://trace.ncbi.nlm.nih.gov/Traces/sra/?run=SRR6020687).

We used simulated reads to illustrate sensitivity and specificity. We plotted the cumulative number of correctly-mapped and incorrectly-mapped reads reported by each aligner, stratified by the MAPQ score (Li *et al.*, 2008) for each read. We used sequencer reads to measure throughput using both paired-end and unpaired reads across a range of speed versus sensitivity settings.

With simulated Illumina read data, Arioc mapped paired-end reads to their correct origin in the reference genome with sensitivity and specificity as good as or better than the best of the BS-seq aligners we evaluated (Supplementary Figs S1–S8). With sequencer reads, Arioc was 10–20 times faster across a wide range of sensitivity settings in comparison with CPU-based aligners (Fig. 1, Supplementary Fig. S9).

4 Discussion

Our results imply that Arioc's BS-seq implementation is at least an order of magnitude faster than the most widely used CPU-only aligners. This estimate is independent of the number of discoverable high-scoring mappings (Supplementary Fig. S10), although it varies with GPU and CPU clock speeds, the number of available GPU and CPU threads, whether the aligner is parameterized to favor speed or sensitivity, and (for large datasets) disk I/O bandwidth.

Significantly, these speed results are conservative because they were obtained with the same Nvidia K20c devices with which we previously reported results for general short-read alignment. With newer GPUs that have more compute cores, higher internal clock speeds, and more on-device memory, throughput is increased by an additional factor of two or more (Supplementary Fig. S11).

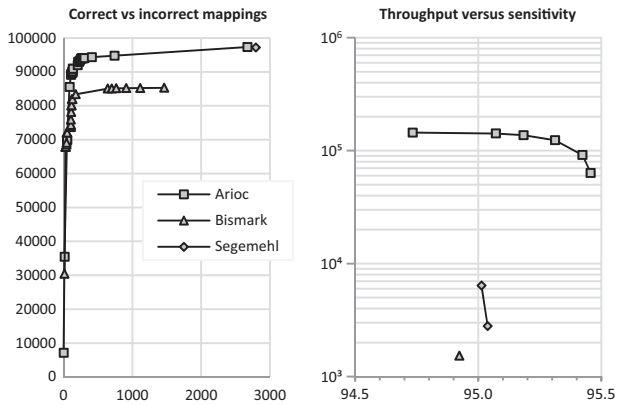

Fig. 1. Sensitivity is evaluated by plotting correctly mapped versus incorrectly mapped reads, ordered by decreasing MAPQ, for 100 000 simulated 100 nt paired-end Illumina reads (200 000 mates) with an error rate of 5%. Speed (reads aligned per second) versus sensitivity (percentage of paired-end reads mapped) is measured for Arioc, Bismark and Segemehl (Otto *et al.*, 2012) using 4 million 100 nt paired-end BS-seq reads (Suzuki *et al.*, 2014). Workstation hardware: 12 CPU cores (24 threads of execution) at 2.93 GHz, one Nvidia K20c GPU (Nvidia Corporation, 2012)

Throughput also scales almost linearly with multiple GPU devices (Supplementary Fig. S12).

Because Arioc integrates the logic for aligning BS-seq reads into the implementation of the alignment algorithms, much of the additional work involved in BS-seq alignment is performed on highly-parallel GPU threads. This contributes to a significant increase in throughput in comparison with a CPU-only implementation. This approach also avoids certain inaccuracies associated with the software architecture of BS-seq read aligners that wrap a general-purpose read aligner like Bowtie 2 (Langmead and Salzberg, 2012) or SOAP3-dp with logic that identifies methylation sites. Such aligners may underestimate overall DNA methylation levels (Xi *et al.*, 2012) and may underreport high-scoring alignments where multiple high-scoring mappings differ only incrementally (Supplementary Appendix A4).

Overall, Arioc provides a tangible increase in throughput in comparison with CPU-based BS-seq aligner implementations while maintaining high sensitivity and avoiding the most common potential inaccuracies associated with BS-seq read-alignment software. Arioc's speed also increases proportionally when additional hardware resources are available. These characteristics make Arioc a reasonable choice for aligning multi-gigabyte WGBS samples and other large datasets of bisulfite-treated short reads.

Acknowledgements

We are grateful to Andrea Manconi and to Felix Krueger for their help with software configuration and for their insights into the technical challenges of BS-seq alignment.

Funding

This work was supported by a National Institutes of Health grant, 'Center for the Epigenetics of Common Human Disease' [P50HG003233 to A.F.].

Conflict of Interest: none declared.

References

Adusumalli,S. *et al.* (2015) Methodological aspects of whole-genome bisulfite sequencing analysis. *Brief. Bioinf.*, **16**, 369–370.

Chen,Y. *et al.* (2009) PerM: efficient mapping of short sequencing reads with periodic full sensitive spaced seeds. *Bioinformatics*, **25**, 2514–2521.

Gotoh,O. (1982) An improved algorithm for matching biological sequences. *J. Mol. Biol.*, **162**, 705–708.

Krueger,F. and Andrews,S.R. (2011) Bismark: a flexible aligner and methylation caller for Bisulfite-Seq applications. *Bioinformatics*, **27**, 1571.

Langmead,B. and Salzberg,S. (2012) Fast gapped-read alignment with Bowtie 2. *Nat. Methods*, **9**, 357–359.

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

Luo,R. *et al.* (2013) SOAP3-dp: fast, accurate and sensitive GPU-based short read aligner. *PLoS One*, **8**, e65632.

Nvidia Corporation. (2012) Tesla K20 GPU Active Accelerator Board Specification. Nvidia document BD-06499-001_v02 (November 2012)

Nvidia Corporation. (2017) CUDA C Programming Guide. Nvidia document PG-02829-001_v9.1 (November 2017)

Otto,C. *et al.* (2012) Fast and sensitive mapping of bisulfite-treated sequencing data. *Bioinformatics*, **28**, 1698–1704.

Smith,T.F. and Waterman,M.S. (1981) Identification of common molecular subsequences. *J. Mol. Biol.*, **147**, 195–197.

Suzuki,A. *et al.* (2014) Aberrant transcriptional regulations in cancers: genome, transcriptome and epigenome analysis of lung adenocarcinoma cell lines. *Nucleic Acids Res.*, **42**, 13557–13572.

Xi,Y. *et al.* (2012) RRBSMAP: a fast, accurate and user-friendly alignment tool for reduced representation bisulfite sequencing. *Bioinformatics*, **28**, 430–432.

Van Nimwegen,K.J.M. *et al.* (2016) Is the $1000 genome as near as we think? A cost analysis of next-generation sequencing. *Clin. Chem.*, **62**, 1458.

Wilton,R. *et al.* (2015) Arioc: high-throughput read alignment with GPU-accelerated exploration of the seed-and-extend search space. *PeerJ*, **3**, e808.