BIOINFORMATICS APPLICATIONS NOTE Vol. 29 no. 21 2013, pages 2790–2791
doi:10.1093/bioinformatics/btt468

Sequence analysis Advance Access publication August 23, 2013

# NextGenMap: fast and accurate read mapping in highly polymorphic genomes

Fritz J. Sedlazeck[1,*,†], Philipp Rescheneder[1,†] and Arndt von Haeseler[1,2]

[1]Center for Integrative Bioinformatics Vienna, Max F. Perutz Laboratories, University of Vienna, Medical University of Vienna, Dr. Bohrgasse 9, A-1030 Vienna, Austria and [2]Bioinformatics and Computational Biology, Faculty of Computer Science, University of Vienna, Waehringerstrasse 17, A-1090 Vienna, Austria

Associate Editor: Inanc Birol

**ABSTRACT**

**Summary:** When choosing a read mapper, one faces the trade off between speed and the ability to map reads in highly polymorphic regions. Here, we report *NextGenMap*, a fast and accurate read mapper, which reduces this dilemma. *NextGenMap* aligns reads reliably to a reference genome even when the sequence difference between target and reference genome is large, i.e. highly polymorphic genome. At the same time, *NextGenMap* outperforms current mapping methods with respect to runtime and to the number of correctly mapped reads. *NextGenMap* efficiently uses the available hardware by exploiting multi-core CPUs as well as graphic cards (GPUs), if available. In addition, *NextGenMap* handles automatically any read data independent of read length and sequencing technology.

**Availability**: *NextGenMap* source code and documentation are available at: http://cibiv.github.io/NextGenMap/

**Contact**: fritz.sedlazeck@univie.ac.at

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

Received on December 21, 2012; revised on May 17, 2013; accepted on August 7, 2013

## 1 INTRODUCTION

High-throughput sequencing technologies produce several million reads per run, with read lengths ranging from 36 to $>1000$ bp. The increasing read length and the advent of technologies like Ion Torrent and MiSeq drive the need for new efficient read mapping methods. In addition, the demand increases for methods that can cope with high sequence divergence and at the same time are user-friendly.

Two main groups of read mapping programs are distinguished based on their indexing methods (Nielsen *et al*., 2011). First, Burrows Wheeler transformation (BWT)-based methods, e.g. BWA (Li and Durbin, 2009), which are fast but optimized for short reads and genomes with low polymorphism. Second, hash-based methods like Stampy (Lunter and Goodson, 2011), which are slow but also suited for highly polymorphic genomes. Thus, a method that combines speed and accuracy provides a quantitative and qualitative improvement of current methods.

To this end, we introduce *NextGenMap*, a method that is faster than current BWT-based methods and at the same time handles short and long reads independent of the number of differences between reads and reference genomes. *NextGenMap* implements new techniques to efficiently identify genomic regions in the reference genome that share a sequence similarity with a read. To achieve a short runtime and to reliably map the reads, *NextGenMap* automatically estimates important parameters (like the number of alignment score computations required for each read). Additionally, *NextGenMap* makes use of multicore CPUs and, if available, any OpenCL-enabled graphic card (GPU).

*NextGenMap* supports fasta, fastq, SAM and BAM as input formats; and outputs SAM and BAM files. Furthermore, *NextGenMap* maps single-end and paired-end data, offers a local (Smith Waterman) and an end-to-end alignment mode and supports aligning bisulfite-treated reads (Dinh *et al*., 2012). The peak memory consumption of *NextGenMap* ranged from 5 to 6 GB, depending on the read length.

## 2 METHODS

*NextGenMap* comprises three steps. First, it splits the reference genome into overlapping *k*-mers and stores the positions in a hash-table. Second, *NextGenMap* identifies the genomic regions, where a read potentially maps to. To this end, the *k*-mers from each read are extracted and putative genomic locations are retrieved from the hash-table. Only regions on the genome where the number of *k*-mer matches exceeds a certain threshold are considered as candidate mapping regions. Unlike other methods, *NextGenMap* automatically determines a read-specific threshold, rather than one threshold for all reads. Third, *NextGenMap* computes the alignment score for the candidate mapping regions. For the candidate region(s) with the highest alignment score, the full alignment is computed and reported. This final step is performed using an extended implementation of the *MASon* library (Rescheneder *et al*., 2012). The Supplementary Section 1 provides a more detailed description on how *NextGenMap* aligns reads.

### 2.1 Benchmark data

To study *NextGenMap's* performance on re-sequencing projects, we used real datasets and simulated datasets (Supplementary Table S4). Three Illumina ($R_1$, $R_2$, and $R_3$), one Ion Torrent ($R_4$) and one 454 dataset ($R_5$) from the Sequence Read Archive (SRA-NCBI) served as benchmark data.

To assess the mapping accuracy, we simulated four read sets using Mason (Holtgrewe, 2010): $S_1$ (150 bp reads), $S_2$ (250 bp reads) derived from the human genome, $S_3$ (100 bp reads) from the *Arabidopsis thaliana*

*To whom correspondence should be addressed.

†The authors wish it to be known that, in their opinion, the first two authors should be regarded as joint First Authors.

genome and $S_4$ (100 bp reads) from the *Drosophila melanogaster* genome. In all datasets, we introduced a 1.2% sequencing error and assumed a genomic polymorphism (SNPs, insertion and deletions) of 0.1% ($S_1$, $S_2$) and 2% ($S_3$, $S_4$). Finally, we simulated 11 datasets ($A_0, \ldots, A_{10}$) based on the *A.thaliana* genome with an increasing degree of genomic polymorphisms (0–10%).

Based on the article by Nielsen *et al.* (2011), we compared *NextGenMap* with four popular mapping methods. Representatives of BWT-based methods were BWA (Li and Durbin, 2009), its extension for longer reads BWA-SW (Li and Durbin, 2010) and Bowtie2 (Langmead and Salzberg, 2012). As hash-based representative we selected Stampy (Lunter and Goodson, 2011). Similar to Fonseca *et al.* (2012), we executed all programs using default parameters. For further details, see Supplementary Sections 2 and 3.

## 3 RESULTS AND DISCUSSION

In terms of runtimes, *NextGenMap* outperformed all analyzed mappers (see Supplementary Table S5a and S6a) on all real ($R_1, \ldots, R_5$) and simulated datasets ($S_1, \ldots, S_4$). *NextGen Map's* CPU implementation was 1.1–2.3 times faster than Bowtie2, the fastest method so far. The GPU implementation further reduces the runtime (1.6–4.9 times faster). If we compare the runtimes of the methods that showed the highest accuracy, *NexGenMap* is between 2.9 and 5.8 times faster than BWA-SW. Stampy, although very accurate for highly polymorphic genomes (2.0%), shows the longest runtimes (maximum 37 h) compared with *NextGenMap* (65 min for the same dataset).

Supplementary Table S6b displays the mapping accuracies for the simulated data. For low genomic polymorphism (0.1%), BWA-SW shows 0.1% and 0.2% more correctly mapped reads compared with *NextGenMap* for $S_1$ and $S_2$, respectively. However, for genomic polymorphism of 2%, the hash-based mappers (*NextGenMap*, Stampy) are the best with a mapping accuracy of 97.6% and 85.5% for $S_3$ and $S_4$, respectively.

To further elucidate the influences of polymorphic genomes on the mapping accuracy, we varied the degree of genomic polymorphism from 0% to 10% for *A.thaliana* ($A_0, \ldots, A_{10}$). Figure 1a displays the decline in mapping accuracy for BWT-based methods and also shows that accuracy is unaffected by degree of genomic polymorphism for the hash-based mappers (Supplementary Table S7). However, Stampy can only retain the accuracy with the expense of an increased computing time (Fig. 1b), whereas *NextGenMap* shows no substantial increase in runtime (Supplementary Table S8). In summary, *NextGenMap* maps reads very accurately and independent of the amount of genomic polymorphism (up to 10%). At the same time, *NextGenMap* exhibits the fastest runtime. We also note that graphic cards provide additional speedup. However, a strong effect is only observed when the number of alignment computations is high, as is the case for highly polymorphic genomes ($\geq$ 8%).

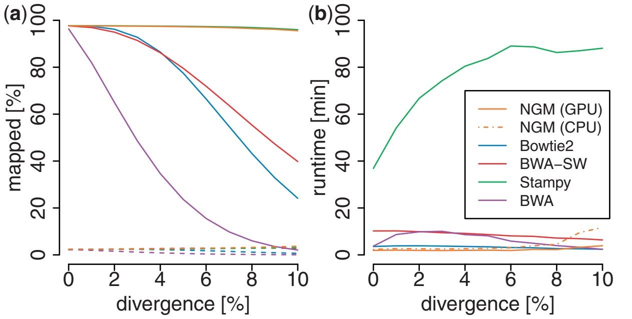

**Fig. 1.** (**a**) Percentage of correctly (solid) and incorrectly (dashed) mapped reads and (**b**) running times for different degree of genomic polymorphisms between read and reference genome for five million 100 bp *A.thaliana* reads ($A_0, \ldots, A_{10}$). NGM = *NextGenMap*

## 4 CONCLUSION

Here, we showed with real and simulated data that *NextGenMap* maps high-throughput sequencing reads accurately and fast at the same time. This fact was already exploited in studies of miRNA (Vesely *et al.*, 2012) and bisulfite-treated data (Dinh *et al.*, 2012).

Finally, the automatized estimation of the required number of alignment score computations per read and the automatic adaptation to the hardware result in a reliable and fast read mapper that requires minimal user interaction. This allows *NextGenMap* to reliably map reads even to highly polymorphic genomes (10%). Thus, it may also be used to map reads from non-standard organism to a phylogenetically close reference genome or to apply it to metagenomics data.

## ACKNOWLEDGEMENTS

The authors thank Miguel Gallach for help with the manuscript and Hermann Romanek for help with the implementation and helpful discussions.

*Funding*: This work was supported by Deutsche Forschungsgemeinschaft SPP1174 [HA1628-9-2 to A.v.H.].

*Conflict of Interest*: none declared.

## REFERENCES

Dinh,H.Q. *et al.* (2012) Advanced methylome analysis after bisulfite deep sequencing: an example in arabidopsis. *PLoS One*, **7**, e41528.

Fonseca,N.A. *et al.* (2012) Tools for mapping high-throughput sequencing data. *Bioinformatics*, **28**, 3169–3177.

Holtgrewe,M. (2010) Mason a read simulator for second generation sequencing data. *Technical Report TR-B-10-06, Institut für Mathematik und Informatik, Freie Universität Berlin*, Available at http://edocs.fu-berlin.de/docs/receive/FUDOCS_document_000000006687 (28 August 2013, date last accessed).

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

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

Li,H. and Durbin,R. (2010) Fast and accurate long-read alignment with BurrowsWheeler transform. *Bioinformatics*, **26**, 589–595.

Lunter,G. and Goodson,M. (2011) Stampy: a statistical algorithm for sensitive and fast mapping of illumina sequence reads. *Genome Res.*, **21**, 936–939.

Nielsen,R. *et al.* (2011) Genotype and SNP calling from next-generation sequencing data. *Nat. Rev. Genet.*, **12**, 443–451.

Rescheneder,P. *et al.* (2012) Mason: million alignments in seconds. In: *Proceedings of the International Conference on Bioinformatics: Models, Methods and Algorithms*. SciTePress, Setubal, Portugal, pp. 195–201.

Vesely,C. *et al.* (2012) Adenosine deaminases that act on RNA induce reproducible changes in abundance and sequence of embryonic miRNAs. *Genome Res.*, **22**, 1468–1476.