An evaluation of the genome alignment landscape

Alexandre Fonseca
KTH Royal Institute of Technology
Stockholm, Sweden
afonseca@kth.se

ABSTRACT

Genetic research has seen considerable advancements in recent years with new sequencing techniques achieving a 400-fold improvement in throughput and reducing the cost of genome sequencing from $100 million to just below $10,000 in the span of a decade. However, despite these developments in the sequencing process (the first in the genome analysis pipeline), the alignment process has remained rather stale and still relies in centralized, non-scalable implementations. Being the second step in the genome analysis pipeline, as well as the most complex and time consuming, alignment has thus become a bottleneck in genome analysis. In this paper, five different sequence aligners are evaluated: three centralized (Bowtie1, BWA and Bowtie2) and two distributed (Crossbow and Seal). The aligners are compared to one another according to alignment accuracy, duration and scalability. Our results indicate that significant improvements to the alignment process can be achieved by using distributed aligners although there is still room for improvement.

Keywords

Genome, sequence alignment, distributed, scalability

1. INTRODUCTION

Recent and continuous advancements in the area of genetic research have enabled scientists to link specific sequences of an organism's genome to particular traits and characteristics of that organism, along with the identification of anomalous mutations. This ability is of paramount importance as it enables a greater comprehension of the biological structure of organisms and its consequent optimization, with applications in a plethora of areas such as agriculture or medicine.

The process of genome analysis of an individual starts with sequencing. During this step, the ordering of the sub-units in an individual's DNA or RNA (also called nucleotides) is determined. The cost and time requirements of this process have been decreasing rapidly over the years with the appearance of next generation sequencing techniques relying on extreme parallelization and short sequence reads. From a total cost of $100 million in 2001, full human genome sequencing now costs less than $10,000 [15]. Regarding throughput, the use of next generation sequencing technologies provides a 400-fold increase when compared to the most commonly used method 10 years ago: Sanger Sequencing [11].

The sequencing process simply provides the raw data regarding reads from an individual's genome. This raw data has to be assembled and processed into a complete genome file. With reference-based analysis, obtained reads have to be aligned with matching portions of a reference genome in a process called sequence alignment. In the genome analysis pipeline, this sequence alignment process turns out to be the most time-consuming task as billions of reads have to be matched against billions of possible subsequences in the human genome. Given the advances in the area of genome sequencing, one would expect that sequence alignment technologies would have evolved accordingly. Unfortunately, this is not the case as many of the sequence alignment tools used nowadays are still too centralized, requiring very powerful computers to perform the alignment. While the alignment problem is well suited for parallelization, most tools restrict it to the thread or processor level which imposes a limit on scalability. Scaling the alignment process is of paramount importance to handle the ever increasing amount of raw data in useful time.

With the advent of cloud computing, individuals and companies can obtain new computing nodes with never before seen ease, cost and deployment speed. By leveraging the power of the cloud and distributing the alignment algorithms through several less powerful nodes, one should be able to achieve significant speed ups in sequence alignment while keeping, or even reducing, the total cost by not having to invest in expensive, specialized hardware. While some work has been started on this area, evaluations are often limited in the scope of alignment tools tested.

In this paper, an evaluation is made between five of the most popular centralized and distributed (Hadoop-based) sequence alignment frameworks. The evaluation addresses such points as total alignment time, accuracy, and scalability. The remainder of the paper will be organized as follows: in section 2, relevant related work is highlighted and commented upon; in section 3, the considered alignment tools are categorized into centralized or distributed and their most relevant characteristics are described; in section 4, the environment, data and configurations under which the evaluation is performed are explained; in section 5, the evaluation results are presented and discussed; finally, in section 6, the main findings are summarized and comments are made regarding possible future work.

2. RELATED WORK

Most evaluations and comparisons between sequence aligners are made in the context of papers presenting a novel sequence aligner algorithm. For instance, [10] presents an evaluation of BWA against MAQ, SOAPv2 and Bowtie1 while [9] evaluates Bowtie1 against MAQ and SOAPv1. While the described experiments are sound, one would expect collected results to be slightly biased towards those novel algorithms introduced in the papers. Unfortunately, there seems to be a lack of independent comprehensive evaluations of these tools. [14] is an example of such an evaluation but, having been made in 1999, it is based on alignment software that is no longer in common use nowadays. In addition, we were not able to find a satisfactory evaluation of distributed sequence aligners against centralized ones.

3. SEQUENCE ALIGNERS

In our evaluation, we considered five of the most popular centralized and Hadoop-based distributed sequence aligner frameworks. In this section, we categorize the different aligners based on those two categories and provide brief descriptions of their main features.

3.1 Centralized

3.1.1 Bowtie1

Bowtie1 [9] is a sequence aligner developed at the John Hopkins University since 2009. Currently in its 1.0.0 version, released on 9 April 2013, Bowtie1 is optimized for small single-ended read sets (up to 1024 base pairs, i.e, sequence characters) and mammalian DNA sequences. Its alignment process is based on a quality-aware, greedy, randomized depth-first search (DFS) through a Burrows-Wheeler Transform (BWT) index [4] which uses slightly over 2GB of RAM.

3.1.2 BWA

BWA [10], also known as Burrows-Wheeler Aligner, is another sequence aligner relying on a BWT index but employing a more precise trie matching mechanism. It was created by the Wellcome Trust Sanger Institute in 2009 and is currently at its 0.7.5a version, released on 5 May 2013. Despite the existence of more recent versions, we evaluated version 0.5.10 of BWA, released on 13 November 2011, so as to make a fairer comparison with SEAL, the distributed version of BWA which is based on this version.

3.1.3 Bowtie2

Bowtie2 [3] is the next generation of the Bowtie1 aligner, also developed at the John Hopkins University. The first version of Bowtie2 was released in 2012 and the evaluated version, 2.1.0, was released on 23 February 2013. Bowtie2 features a series of improvements over its predecessor, including: a Ferragina Manzini (FM) [6] index; optimizations for reads larger than 50 base pairs; support for gapped alignment (matching ends of sequences if quality of the middle section is bad); end to end and local alignments (total or partial alignments); and better support for paired-end alignment.

3.2 Distributed

Regarding the considered distributed sequence aligner algorithms, this paper focuses on those implemented using the Hadoop computational framework due to its high resilience to failures and ease of distribution. However, other distributed implementations exist that do not rely on Hadoop. Gnumap [5], ERNE [13] and GotCloud [2] are examples of such implementations that rely on the MPI framework or more archaic distribution models that are not commonly used in cloud environments.

3.2.1 *Crossbow*

Crossbow [8] is a distributed sequence analysis framework containing tools for aligning and analysing genomes. For the alignment step, it relies on Bowtie1 and is also being developed by the creators of Bowtie1. The version considered in this evaluation was the latest one, 1.2.1, released on 30 May 2013. By comparing the results obtained with Bowtie1 and Crossbow, conclusions regarding the advantages of node-level distribution can be easily extracted.

3.2.2 *Seal*

Just like Crossbow, Seal [12] is a complete distributed sequence analysis framework employing custom-tailored alignment and analysis components. The aligner components, however, were built from BWA 0.5.10, thus providing an opportunity to evaluate the effects of distribution of this aligner. The current version of Seal is 0.3.2 released on 7 February 2013.

4. EVALUATION SETUP

4.1 Hardware

The evaluation of the sequence aligners was done in a 7-node cluster owned by the Swedish Institute of Computer Science (SICS). Each node of this cluster is equipped with 2 6-core AMD Opteron 24355 CPUs, 32GB of RAM and 1TB of disk. The nodes are interconnected with 1Gbps full duplex ethernet. Unfortunately, access to the hardware and software resources of this cluster was shared, which meant not having complete flexibility in terms of configuration and the possibility of some interferences in the execution of the alignment tasks by other processes. However, an attempt was made to minimize the effect of shared executions on the results by only performing the alignments in periods of apparent inactivity of the cluster.

4.2 Software

In this evaluation, the aligner versions considered were: Bowtie1 1.0.0, BWA 0.5.10, Bowtie2 2.1.0, Crossbow 1.2.1 and Seal 0.3.2.

Sequence aligners relying on the Hadoop framework were run on Hadoop version 2.2.0, with, unless otherwise stated, 5 NodeManagers on nodes 1-5 of the cluster, ResourceManager on node 6 and HDFS NameNode on node 7. Each NodeManager was given access to 16GB of RAM (as enforced by the shared configuration) and to all the 12 CPUs. Each container running in the NodeManager could use at most 8GB of physical memory thus limiting the number of containers per node to at least 2. Both Crossbow and Seal were configured so as to take full advantage of the provided hardware resources.

Centralized sequence aligners were given the entirety of the resources of the machine in which they ran. Consequently, Bowtie1, BWA and Bowtie2 used up to 12 CPU cores and 32GB of RAM in their executions.

4.3 Inputs

For the benchmarking of the sequence aligners, two types of reads were used: single-ended and paired-ended. Paired-ended reads are obtained in the sequencing step by reading the same sequence starting at the beginning and at the end (reversed). Since read quality decreases as the distance from the starting point of the reading increases, by collecting two sequences from opposite starting points, one is able to obtain a read sequence with better overall quality. This also means that paired-ended reads are approximately twice larger than their single-ended counterparts. Although, initially, this evaluation was intended to focus on just one of these types, the inability of Bowtie1 and Crossbow to process paired-ended reads as well as the inability of Seal to process single-ended reads have forced the inclusion of both types of reads in the evaluation.

For each type of read we have considered two different categories: simulated and real. Regarding the former type, three sets of 900k 100 base-pair reads were generated from the hg19 human reference genome using the *wgsim* tool [1] with a 0.09% SNP mutation rate, 0.01% indel mutation rate and different sequencing error rates: 2%, 5% and 10%. These values were based on previous aligner evaluations using simulated reads such as [16] [10]. Each of these datasets amounts to roughly 200MB in filesize for the single-ended version and twice that (400MB) for the double-ended version. The real reads were obtained from the datasets sequenced at the Broad Institute and made available at the 1000Genomes project pertaining to individual NA12878 [7]. These reads are composed of 101 base-pairs and have a coverage of over 60 times the individual's genome, i.e., a redundancy factor of over 60 to account for sequencing errors. Of the whole genome, we focused on the reads of a subset of chromosomes with varying size (twice the size in GB for paired-ended sets):

- Chromosome 4 with 156M reads and a size of 35.5GB for the single-ended sets.
- Chromosome 20 with 51M reads and a size of 12GB for the single-ended sets.
- Chromosome 21 with 31M reads and a size of 7GB for the single-ended sets.

5. RESULTS

5.1 Accuracy

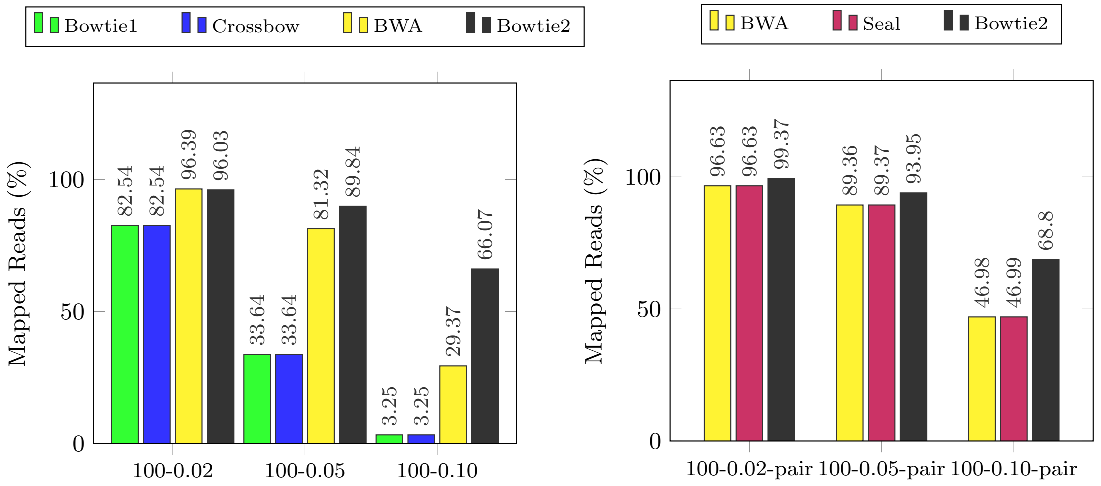
One of the most desirable characteristics for sequence aligners is good accuracy. Accuracy determines the quality of the resulting alignments and is therefore of vital importance for the achievement of a solid and adequate analysis of a genome. Accuracy can be measured in 2 ways regarding sequence alignment: percentage of aligned reads and percentage of errors in the aligned reads (alignment to wrong positions in the reference genome).

Figure 1 details the percentage of reads that each aligner was able to map to the reference genome from the simulated read sets. According to the figure, we can clearly see that the error rate of the sequencing has a direct and significant impact on the percentage of reads mapped in the alignment step, with higher error rates resulting in drastic reductions in this percentage. Among the aligners tested with the single-end read sets described in Figure 1a, it is clear that Bowtie1 and Crossbow are the worst performers, consistently below the values of the other aligners and achieving an all-time low of 3.25% with the highest input error rate. BWA, on the other hand, is the best aligner with very low input error rates but its accuracy decreases sharply as this error rate increases. Bowtie2, appears to be the best overall performer of the four, achieving over 50% of mapped reads even with the highest input error rate. This result is also reflected with the paired-end read sets as shown in Figure 1b where Bowtie2 consistently achieves over 68% alignment percentages. In this case, both BWA and Seal achieve very satisfactory results with the 2% and 5% error read sets, just 3% to 5% below the mapping percentages of Bowtie2. However, with the input set with the highest error rate, the percentage of alignment plummets to just under 47%.

Of particular interest in these results is the fact that aligners tend to obtain better results with paired-end read sets and that the distribution of the alignment process (Bowtie1 to Crossbow and BWA to Seal) does not appear to have an impact on the percentage of mapped reads.

Figure 2 details the other viewpoint of accuracy: the percentage of reads that were mapped to the wrong position. Since these read sets are artificially generated from the reference genome, they are tagged with their original location. By comparing the tag with the location determined by the aligner, one is able to verify the correctness of the alignment. Looking at the results for single-end read sets in Figure 2a, one can see that while the mapping error rate increases slightly with the input error rate, all aligners manage to achieve error percentages below or very close to 5% and, thus, compatible with 95% confidence intervals. Among these, BWA achieves the lowest error rate with Bowtie2 achieving the highest (perhaps motivated by the riskier mappings that allow Bowtie2 to achieve such good mapping percentages). With paired-end read sets, as shown in Figure 2b, there appears to be a slight increase in the overall error rate suggesting that while paired-end reads might have better quality, they are harder to map to the correct positions due to higher algorithmic complexity. Just as with the single-end read sets, Bowtie2 is the worst performer achieving a maximum error percentage of 6.72%. On the other hand, Both BWA and Seal manage to keep their error percentage below 5% for all three read sets.

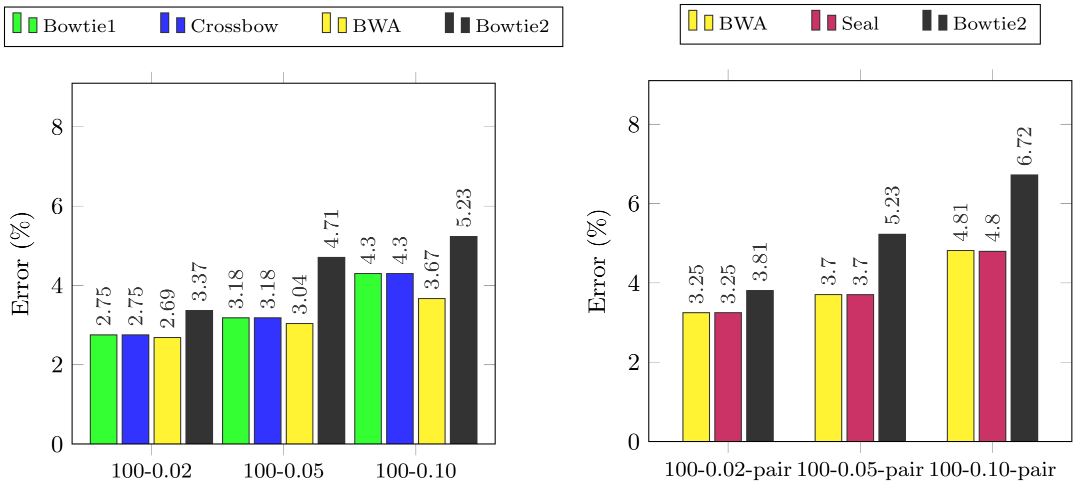
Just as what happened with the percentage of mapping results, one can also see that the distribution of the alignment does not affect the error percentage with both Bowtie1 and Crossbow, and BWA and Seal achieving exactly the same error percentages.



(a) Single-end read sets

(b) Paired-end read sets

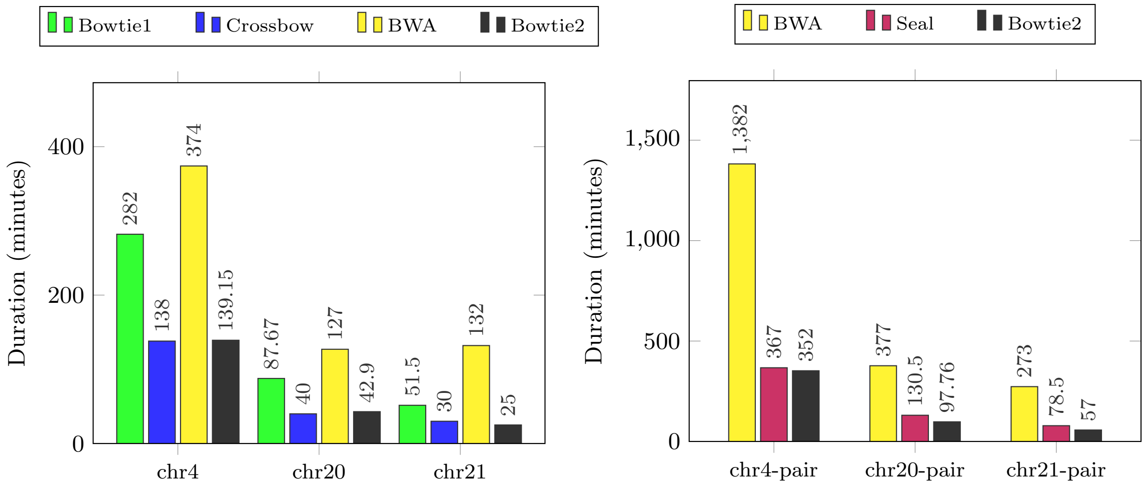
Figure 1: Percentage of mapped reads



(a) Single-end read sets

(b) Paired-end read sets

Figure 2: Percentage of alignment errors



(a) Single-end read sets

(b) Paired-end read sets

Figure 3: Alignment duration

5.2 Alignment duration

The speed with which a sequence aligner processes reads from a data set is one of the most important characteristics in these tools. With an increase in reads processed per second, the latency of the alignment step and, consequently, of the whole genome analysis pipeline, suffers a proportional decrease. While this might be of little consequence for scientific research where throughput, not latency, is the main objective, faster genome analysis is fundamental in medicine where a quick diagnosis might be the difference between life and death.

Alignment durations with real single and paired-end read sets can be seen in Figure 3. The read sets are presented in decreasing order of size with chromosome 4 being the largest read set and chromosome 21 being the smallest as described in subsection 4.3. Looking at the data, it is clear that the old generation centralized aligners offer the slowest alignment. In particular, for the single-end read sets shown in Figure 3a and, specifically, for chromosome 4, BWA takes 374 minutes and Bowtie1 282 minutes against the 138 minutes taken by Crossbow, the fastest aligner of the four. Crossbow, in fact, is able to obtain an alignment in less than half the time needed by Bowtie1, its centralized counterpart. This distribution-induced speedup, while decreasing slightly as smaller input sets are considered due to increased overhead, is still significant with chromosome 21 where Crossbow achieves a 20 minute reduction in alignment time. A surprising result from these experiments is that Bowtie2 produces alignments in roughly the same time as Crossbow, needing between 1 and 2 extra minutes for chromosomes 4 and 20 and being faster than Crossbow by 5 minutes with chromosome 21. This, in conjunction with the accuracy results shown in the previous section, is a testament to the efficiency of new alignment algorithms. Similar results can be seen in Figure 3b regarding paired-end read sets where Bowtie2 obtains the fastest alignments closely followed by Seal which required between 10 and 30 extra minutes. Even so, Seal represents a significant improvement over the alignment duration achieved by BWA, obtaining a 4- to 3-fold speedup over the alignment durations of the latter.

5.3 Scalability

The experiments described in the previous section show the obtained speedup for distributed aligners in a 5-node cluster. In this section, the relation between the number of nodes in the cluster and the alignment duration was examined.

Figure 4 shows the duration of the alignments of the chromosome 20 read set as performed by the two considered distributed aligners under a varying number of nodes (between 1 and 7). Also in-

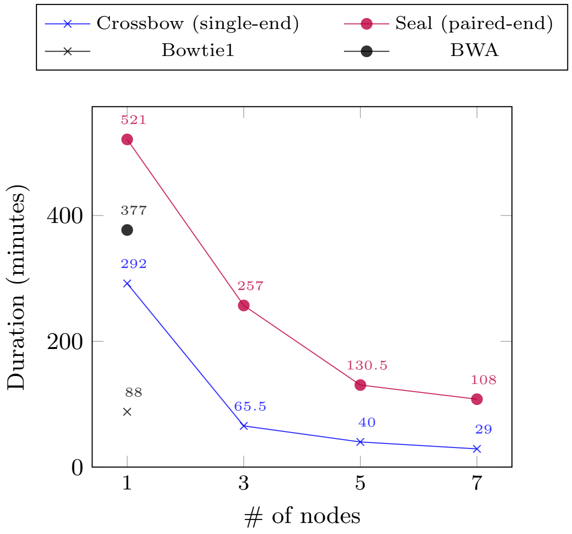


Figure 4: Effect of number of nodes in alignment duration with chromosome 20

cluded are reference points for the alignment durations achieved by the respective centralized aligners in a single-node scenario. By looking at the alignment times of Crossbow and Seal with 1 node, the impact of the distribution overhead introduced by the Hadoop framework becomes quite apparent: Crossbow takes a little over 3 times as much time as Bowtie1, needing 292 minutes to produce an alignment while Seal fares better with an alignment duration of just over 1.3 times that achieved by BWA. However, as one begins to add more nodes to the cluster, the alignment runtime is greatly reduced. With 3 nodes, Crossbow is already able to perform an alignment 20 minutes faster than Bowtie1 (and 230 minutes faster than in a single node). Similar results can be seen for Seal where, with 3 nodes, it achieved a runtime reduction of 120 minutes over BWA and of just over 260 minutes compared to a single node execution. As more nodes are added to the cluster, the alignment duration is further reduced although improvements start becoming progressively smaller. Nevertheless, from 5 nodes to 7, Crossbow still experiences a speedup of 11 minutes and Seal of approximately 20 minutes.

6. CONCLUSIONS

In this paper, the most popular offerings in terms of sequence aligners, both centralized and decentralized (based on the Hadoop framework), have been evaluated. This evaluation encompassed not only the duration of the alignment process for each of these tools, but also the accuracy of the final result and the scalability of the distributed aligners.

Based on this evaluation, the initial hypothesis regarding distributed aligners (better scalability and speed up) has been corroborated for bigger sets of reads. While there is still considerable overhead involved in the distribution of this computation, this overhead can be amortized over the speed up gains obtained by adding more hosts.

However, there is still room for improvement. Although distributed implementations fare well compared to their centralized counterparts (Crossbow vs Bowtie1; Seal vs BWA), a new generation of more efficient centralized tools such as Bowtie2 have achieved outstanding results even in a centralized setting. To the best of our knowledge, there is still no distributed aligner based on these more advanced algorithms. Such an aligner would, undoubtedly, perform significantly better than current distributed implementations and offer the possibility to further speed up the genome analysis process.

7. REFERENCES

[1] Github - lh3/wgsim. `https://github.com/lh3/wgsim`. Accessed: 2013-12-26.

[2] Gotcloud - genome analysis wiki. `http://genome.sph.umich.edu/wiki/GotCloud`. Accessed: 2013-12-26.

[3] S. L. S. Ben Langmead. Fast gapped-read alignment with bowtie 2, 2012.

[4] M. Burrows, D. J. Wheeler, M. Burrows, and D. J. Wheeler. A block-sorting lossless data compression algorithm. Technical report, 1994.

[5] N. L. Clement, Q. Snell, M. J. Clement, P. C. Hollenhorst, J. Purwar, B. J. Graves, B. R. Cairns, and W. E. Johnson. The gnumap algorithm: unbiased probabilistic mapping of oligonucleotides from next-generation sequencing. *Bioinformatics*, 26(1):38–45, 2010.

[6] P. Ferragina and G. Manzini. Opportunistic data structures with applications. In *Foundations of Computer Science, 2000. Proceedings. 41st Annual Symposium on*, pages 390–398, 2000.

[7] B. Institute. Gatk | best practices. `http://www.broadinstitute.org/gatk/guide/best-practices#bp_1292`.

[8] B. Langmead, M. Schatz, J. Lin, M. Pop, and S. Salzberg. Searching for snps with cloud computing. *Genome Biology*, 10(11):R134, 2009.

[9] B. Langmead, C. Trapnell, M. Pop, and S. Salzberg. Ultrafast and memory-efficient alignment of short dna sequences to the human genome. *Genome Biology*, 10(3):R25, 2009.

[10] H. Li and R. Durbin. Fast and accurate short read alignment with burrowswheeler transform. *Bioinformatics*, 25(14):1754–1760, 2009.

[11] O. Morozova and M. A. Marra. Applications of next-generation sequencing technologies in functional genomics. *Genomics*, 92(5):255 – 264, 2008.

[12] L. Pireddu, S. Leo, and G. Zanetti. Mapreducing a genomic sequencing workflow. In *Proceedings of the second international workshop on MapReduce and its applications*, MapReduce '11, pages 67–74, New York, NY, USA, 2011. ACM.

[13] N. Prezza, C. Del Fabbro, F. Vezzi, E. De Paoli, and A. Policriti. Erne-bs5: Aligning bs-treated sequences by multiple hits on a 5-letters alphabet. In *Proceedings of the ACM Conference on Bioinformatics, Computational Biology and Biomedicine*, BCB '12, pages 12–19, New York, NY, USA, 2012. ACM.

[14] J. D. Thompson, F. Plewniak, and O. Poch. A comprehensive comparison of multiple sequence alignment programs. *Nucleic Acids Research*, 27(13):2682–2690, 1999.

[15] K. Wetterstrand. Dna sequencing costs: Data from the nhgri genome sequencing program (gsp). `http://www.genome.gov/sequencingcosts`, Oct. 2013. Accessed: 2013-11-17.

[16] M. Zaharia, W. J. Bolosky, K. Curtis, A. Fox, D. A. Patterson, S. Shenker, I. Stoica, R. M. Karp, and T. Sittler. Faster and more accurate sequence alignment with snap. *CoRR*, abs/1111.5572, 2011.