NGS Short Read Alignment Algorithms and the role of Big Data and Cloud Computing

Rexie J A M, Kumudha Raimond, Mythily M, Kethsy Prabavathy A

Abstract*: *Next Generation Sequencing (NGS) raises opportunities to the computational field for fast and accurate methods for the various challenges associated with NGS data. NGS technology generates a large set of short reads of size 50 to 400 base pairs as a result of biological experiments done on the samples taken from species. Such raw reads are not directly ready for doing most of the analysis or comparative studies to figure out medical related solutions. Hence, the reads have to be assembled to form a complete genome sequence. During the assembly process, there is a high chance of erroneous positioning. Some strategy has to be applied to correct such errors. Once the error-free sequence data is prepared, it is ready for further analysis. The analysis may assist in identifying disease and its cause, similarity check, genetic issue, etc. All of these processes involve data of huge size (in terms of millions per day). To improve the performance of the algorithms working on such vast amount of data, the latest technologies such as Big Data and Cloud Computing can be incorporated. Here, in this paper the evolution of the algorithms for NGS data alignment and the role of Big Data and Cloud Computing technologies are discussed.

***Index Terms*: *NGS Short Read Alignment, Burrows Wheeler Transform, Suffix Array, Big Data, Cloud Computing*.**

I. INTRODUCTION

Human genome is made up of 23 chromosomes pairs within which DeoxyriboNucleic Acid (DNA) is encoded. DNA is identified as paired helical strands of chemicals referred as A - Adenine, T – Thymine, C – Cytosine and G – Guanine. Among these, A is complement for T and, C is complement for G. One of the strands is the complement of the other. Each of the pairs A-T and C-G is referred to be as base pairs. Hence it is to find the sequence of one of the strands which will in turn dictate the other and a DNA will be of 3-billion base pairs in size. The very first human DNA sequence was published by the project titled "Human Genome Project", which had taken 13 years to complete the project [1]. This ignited the Bio-medical field to focus on sequencing data and was considered as the First-Generation Sequencing. The sequence data can be used as reference sequence and based on which various medical analyses were done in the view of personalized medicine.

Soon later, the second-generation sequencing known as NGS, which is faster and cheaper than the first, has started booming. Presently, the NGS technology systems generate approximately 250 bases per week [2]. However, NGS technology will generate sequence in the form of millions of short reads of 50 to 400 base pairs, which have to be assembled to construct the whole genome. Assembling the short reads is achieved in two ways, alignment-based and assembly-based methods. The earlier generation needs the reference genome of the same species, which is being aligned, to regenerate the sequence, but the later doesn't need such.

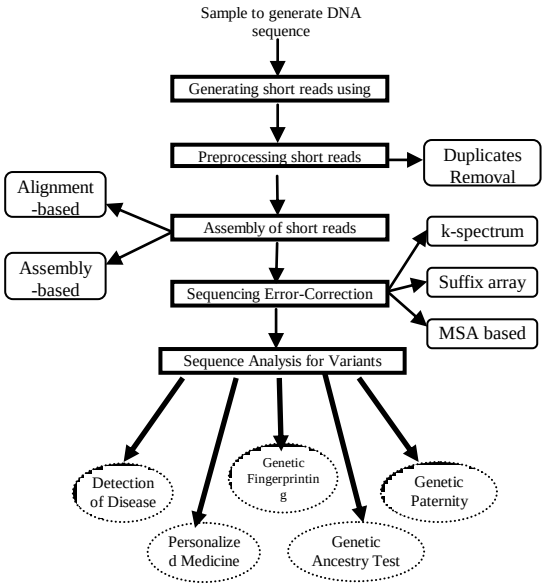

Fig. 1. Flow of NGS Data Processing and Analysis

Many methods have been developed for regenerating the whole sequence. But the common challenges faced by every approach are: (i) the enormous number of short reads as huge as 1TB [4] per sample generated by NGS, (ii) the duplicates present in the short reads and (iii) sequencing errors [3]. (i) The storage and processing of large data set can be overcome with the advent of Big Data and cloud computing platform, by doing the alignment process in distributed and parallel processing methods. (ii) For the removal of duplicate, many strategies are being followed. One of such tools, named ParDRe [5], employs clustering technique for eliminating duplicated reads. It forms groups of reads having similar prefix of length l. Among each group, in parallel, reads are examined for similar suffixes to identify duplicates. (iii) To address the issue of sequencing errors, many mechanisms have been deployed based on k-spectrum, suffix array and Multiple Sequence Alignment (MSA) methods [6].

Revised Manuscript Received on July 05, 2019.

Rexie J A M, Computer Science and Engineering, Karunya Institute of Technology and Sciences, Coimbatore, India.
Kumudha Raimond, Computer Science and Engineering, Karunya Institute of Technology and Sciences, Coimbatore, India.
Mythily M, Computer Science and Engineering, Karunya Institute of Technology and Sciences, Coimbatore, India.
Kethsy Prabavathy, Computer Science and Engineering, Karunya Institute of Technology and Sciences, Coimbatore, India.

Thus, the sample in the form of hair, blood or spit is the only source taken from the species, from which many biological and computational processes are done in order to equip medical field to give solutions to health-related issues. Each individual whole human genome would be having a size of 19 to 40 Mb unique DNA sequences compared to the reference gnome [7]. To identify such uniqueness, analysis on genome variation discovery has to be done. The flow diagram given in Fig. 1. depicts the mechanisms associated with the NGS sequenced data. This paper reviews the algorithms for the assembly of short reads and how the algorithm can be improved with the advantages of Big Data and Cloud Computing technologies.

II. ALIGNMENT-BASED ALGORITHMS FOR ASSEMBLY OF SHORT READS

In alignment-based approach, reference genome of the species, for which the short reads are to be assembled in order to get the complete sequence, is used to find the alignment of sequences. A sample alignment of the short reads with the reference genome is shown in Fig. 2.

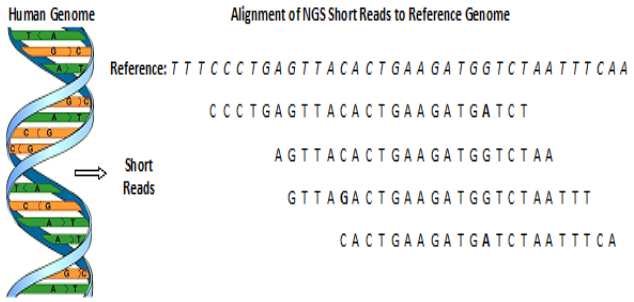

Fig. 2. Alignment of short reads against reference genome

Most of the alignment-based approaches for short read alignment apply the concept of Burrows Wheeler Transform (BWT) and Suffix Array (SA) construction [8], [9]. The BWT of a text T is all possible permutations of T. The pseudocode given in Algorithm 1 expresses the construction of BWT and SA of T[7]. BWT and SA are constructed for an example in Fig. 3.

Algorithm 1 Compute BWT and SA

```
Algorithm BWTSA(T)
//$ is lexicographically the least character
append $ at the end of T;
l = length(T$);
Let A[] be an array of l strings;
A[0]=T$;
for(i=1;i<l;i++) do
   //Cyclic left shift of A[i-1]
   A[i]= (<<A[i-1]);
end for
B[] = array of sorted strings of the array A[];
bwt = character at position l-1 of B[0] through B[l-1];
for(i=0;i<l;i++) do
   find the index of A, say j, such that B[i]=A[j];
   sa[i]=j;
end for
```

Index	Left rotated array
0	G A T G A C $
1	A T G A C $ G
2	T G A C $ G A
3	G A C $ G A T
4	A C $ G A T G
5	C $ G A T G A
6	$ G A T G A C

#	SA	Sorted Left rotated array
0	6	$ G A T G A C
1	4	A C $ G A T G
2	1	A T G A C $ G
3	5	C $ G A T G A
4	3	G A C $ G A T
5	0	G A T G A C $
6	2	T G A C $ G A

Fig. 3. Example showing construction of BWT and SA of the text T = "GATGAC" BWT(T) = "CGGAT$A and SA(T) = {6,4,1,5,3,0,2}

The algorithm BWTSA is applied to the reference sequence and BWT and SA are generated for the reference. To find whether any string is a substring of the reference sequence, it is enough to find the interval in the SA where the searching string is the prefix. This is because the SA is a sorted array having all possible permutations of left rotated sequence of strings. For example, to find the positions where the substring "GA" occurs in the string "GACGAT", the SA interval is found as [4,5]. The SA values with respect to the interval [4,5] are 3 and 0 which are the positions where the substring "GA" is found in the given string. If the search is for exact matching, there will be only one SA interval found. But for finding inexact matches, many such intervals can be found.

During alignment of short reads, exact matching will not be the objective and also not possible. Inexact matches should be allowed because there will be structural variation between the reference sequence and the short reads being aligned. Bowtie [10] permits the user to decide on the number of mismatches allowed. It considers the seed of the short reads, which is the high quality end of the reads, usually of 28 bases length. The seed is partitioned as two halves: the high quality half of the seed and the low quality half of the seed. The algorithm works in three phases:

Phase 1 - attempts to find the alignment allowing one or two mismatches in low quality half;

Phase 2 - allows permitted number of mismatches in high quality half;

Phase 3 - permits partly mismatches in high quality half and the remaining in the low quality half.

MAQ [18] is a program for short read alignment, first searches for the alignment without indels with least mismatch score at the alignment stage. MAQ takes only the positions with less than three mismatches into consideration to fasten the process of first stage i.e., alignment stage. Then, it searches for the alignment with indels in the areas of sequence that had comparatively higher mismatch score, but their mate pair is mapped in the previous stage. MAQ always align the reads across a single position in the reference sequence. Even if it finds a read matching similarly in more places of the reference, it selects one of them in random and assigns zero as mapping score. Hence, during the sequence alignment analysis phase, these positions will not be considered for structural variation. But still, the data on repeat count of the sequences and the amount of reads which can be aligned to the reference are given. If

needed, this information can be utilized for analysis. MAQ is unique from other methods in assigning mapping score and aligning all reads even if there are repeats.

SOAP2 [11] tool for short read alignment speed up the alignment process by creating hash table for the BWT index. To allow mismatches, it uses split-read method, where it allows one mismatch in either halves of the read at a time.

CUSHAW [12], short read aligner using parallel programming model, works on the basis of BWT. For inexact matches, it does not permit gapped alignment, i.e., insertions or deletions. Hence, inexact match can be considered as the exact matches of all permutations possible in a short read. To implement such a match, it constructs four-ary tree as in Fig. 4. for the permutations and applies depth-first-search (DFS) technique for matching.

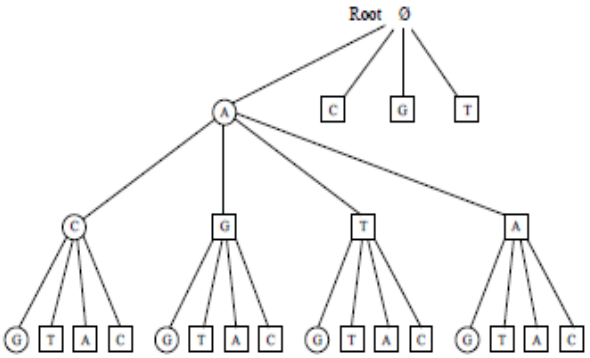

Fig. 4. A complete four-ary tree for inexact matches for the string "ACG" [12]

Kart [21] is a short read alignment program which follows divide and conquer strategy. Since the long reads are partitioned into smaller parts, it can be applied to align long reads and the alignment speed will be as same as the alignment of short reads. Kart takes read by read of different sizes from the NGS data and searches for all possible Locally Maximal Exact Matches (LMEMs) in the read to be aligned with reference sequence. For such aligning, it applies BWT search. Then the adjacent LMEMs are grouped based on mismatch score. To compose the clusters formed as a complete alignment, Kart packs the indels among the simple pairs with normal pairs. Once the size of the normal pair exceeds 30bp, it again applies divide and conquer technique and repeats sequence partitioning. The concatenation of simple pair and normal pair will be the end result.

Burrows Wheeler Alignment – Maximal Exact Match (BWA-MEM) [19] is a technique for short read alignment for which seed and extend strategy is the fundamental concept. Such substrings are identified and extended on both sides of the seeds by permitting a specified limit of mismatches. The mismatches can be either insertion or deletion. The implementation is based on Graphics Processing Unit (GPU). Parallelism is applied for processing multiple reads at the same time. But read extension is done serially, because one seed may be extended so that it can get overlapped with another. The scope for the extension of this work is to enable the seed extension process applicable for parallel processing.

Meta-aligner [20] is an efficient solution to align long reads against reference sequence. The available NGS read alignment programs are integrated in Meta-aligner as sub-modules and the estimation of statistics from reference sequence is used to improve the performance. It works in two phases:

First Phase (**Alignment**): Using existing aligners, do exact alignment portions of the NGS reads. Maximum portions of the reads will be processed here.

Second Phase (**Assignment**): The reads which are not aligned in the previous phase will be the challengeable reads which will be processed in this phase. Here, the remaining reads are clustered into groups based on the similarity of the substrings. Then the members of these groups are aligned.

III. THE ROLE OF BIG DATA AND CLOUD COMPUTING IN NGS ALIGNMENT & ANALYSIS

The NGS which tends to deliver crude information of size as substantial as 1TB for each example frequently postures trouble for the data mining and sensible elucidation. The huge amount of biological information created through NGS over the world in research laboratories is instances of Big Data. On an average, volume of space required for storing 1000 human genomes will be on an average 3TB; nevertheless, the storage space needed for raw sequences will be considerably of more size [13]. The current Big Data analytics, which are intended to deal with large-scale information through hardware affiliation, have been a bonus for NGS information analysis.

Cloud computing gives a versatile and cost productive solution for the Big Data challenge. A single machine can frequently have several Virtual Machines (VMs) by virtualization technology, guaranteeing most extreme use of the hardware and capital investment. A VM is a product application that copies a physical computing environment in which an Operating System (OS) and related applications can keep running with various VMs installed on a single machine. A Hypervisor, a virtualization administration layer, deciphers the requests from the VM to the underlying system (CPU, memory, hard disks and network) [14]. Software as a Service (SaaS) is a way of benefiting users to avail an application running on a remote cloud infrastructure, by means of the Internet. Amazon Cloud Services gives access to a few vast genomic informational indexes including the 1000 Genome projects, and in addition to NCBI, GenBank and Ensembl.

To enhance the performance of BWA, a tool named BigBWA [15] was established that takes advantage of Hadoop as Big Data technology. The main advantages of such a tool are the following: (a) the alignment process is done in parallel using a tested and scalable technology, which reduces the execution times dramatically; (b) BigBWA is fault tolerant, exploiting the fault tolerance capabilities of the underlying Big Data technology on which it is based; (c) There is no need of any alterations to BWA for making use of BigBWA. As a result, BigBWA will be compatible with any release of BWA.

StreamAligner [16] is a MapReduce based sequence alignment tool implemented on Apache Spark. It makes use of a SA index to map reads onto a reference genome. It follows three iterations to construct an index and plot a read onto a reference genome:

Iteration 1: The reference genome is cleaned and transformed so that it can further be processed using parallel computing. Iteration 2: A SA index is produced for reference genome which is processed in the previous iteration. Using a distributed algorithm, StreamAligner generates a SA index. Iteration 3: Streams are aligned onto the reference genome.

CloudBurst [17] is another MapReduce based algorithm for read mapping with reference sequence. It consists of three phases such as map, shuffle and reduce as shown in Fig. 5.

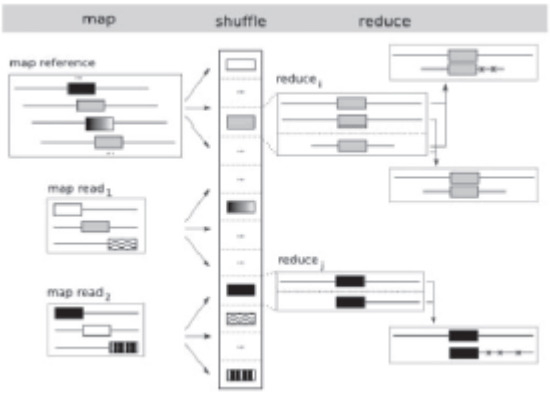

Fig. 5. CloudBurst – An overview [17]

The phases work as follows to do the alignment:

Phase 1 – *Map*: From the reads and reference sequence, k-mers (substrings of length k), called as seeds are produced.

Phase 2 – Shuffle: The k-mers (seeds), distributed among the reads and sequence, are assembled together.

Phase 3 – Reduce: The shared seeds are extended into end-to-end alignments permitting both mismatches and indels (insertions/deletions).

IV. CONCLUSION

The advancement of NGS technologies enables generating genome-wide sequence data over a short span of time, and this huge amount of data led to numerous important discoveries. However, this technology needs specific data analysis algorithms which are written for specific task, such as identification and cause of disease, personalized medicine, etc, to be performed. This paper focuses on the different short read alignment methods. Most of the alignment algorithms follow either BWT or seed and extend or both as the base for aligning the reads and work in stage by stage for pre-processing and aligning the NGS short reads. Also, it is discussed about how Big Data and Cloud Computing can contribute for enhancing and making ease the processes associated with NGS data.

REFERENCES

1. Brenda J Wilson, "The Human Genome Project, and recent advances in personalized genomics. Risk Management and Healthcare Policy", 9–20 (2015).
2. Momiao Xiong, "Next-Generation Sequencing" A Journal of Biomedicine and Biotechnology, (2010)
3. Lucian Ilie, "HiTEC: accurate error correction in high-throughput sequencing data" Bioinformatics, 27(3) 295–302 (2011)
4. Rashmi Tripathi, "Next-generation sequencing revolution through big data analytics" Frontiers in Life Science, 9(2) 119-149 (2016)
5. Jorge González-Domínguez, "ParDRe: faster parallel duplicated reads removal tool for sequencing studies" Bioinformatics, 32(10) 1562–1564 (2016)
6. Xiao Yang, "A survey of error-correction methods for next-generation sequencing" Briefings in Bioinformatics, 14(1) 56–66(2013)
7. Ruiqiang Li, "Building the sequence map of the human pan-genome" Nature Biotechnology, 28(1) 57-63 (2010)
8. Ross A. Lippert, "Space-Efficient Whole Genome Comparisons with Burrows–Wheeler Transforms" Journal Of Computational Biology, 12(4) 407-415 (2005)
9. Heng Li, "Fast and accurate short read alignment with Burrows–Wheeler transform" Bioinformatics, 25(14) 1754-1760 (2009)
10. Langmead B, "Ultrafast and memory-efficient alignment of short DNA sequences to the human genome" Genome Biology, 10(3) R25 (2009)
11. Ruiqiang Li (2009). SOAP2: an improved ultrafast tool for short read alignment. Bioinformatics, 25(15) 1966-1967 (2009)
12. Yongchao Liu, "CUSHAW: a CUDA compatible short read aligner to large genomes based on the Burrows–Wheeler transform" Bioinformatics, 28(14) 1830-1837 (2012)
13. Rashmi Tripathi, "Next-generation sequencing revolution through big data analytics" Frontiers in Life Science, 9(2) 119–149 (2016)
14. Aisling O'Driscoll (2013) 'Big data', Hadoop and cloud computing in genomics. Journal of Biomedical Informatics, 46(5) 774–781 (2013)
15. Jose M. Abu, "BigBWA: approaching the Burrows–Wheeler aligner to Big Data technologies" Bioinformatics, 31(24) 4003–4005 (2015)
16. Sanjay Rathee, "StreamAligner: a streaming based sequence aligner on Apache Spark. Journal of Big Data (2018) 5:8
17. Michael C. Schatz (2009). CloudBurst: highly sensitive read mapping with MapReduce" Bioinformatics, 25(11) 1363–1369 (2009)
18. Heng Li, "Mapping short DNA sequencing reads and calling variants using mapping quality scores" Genome Research, 18, 1851-1858 (2008)
19. Ernst Joachim Houtgast, "GPU-Accelerated BWA-MEM Genomic Mapping Algorithm Using Adaptive Load Balancing" Springer: ARCS 2016, LNCS 9637, pp. 130–142, 2016.
20. Damoon Nashta-ali, "Meta-aligner: long-read alignment based on genome statistics" BMC Bioinformatics (2017) 18:126
21. Hsin-Nan Lin, "Kart: a divide-and-conquer algorithm for NGS read alignment" Bioinformatics, 33(15) 2281–2287 (2017)

AUTHORS PROFILE

Rexie J A M has completed B.E. and M.E. in Computer Science and Engineering and is currently pursuing Ph.D. in Computer Science and Engineering in the area of Bio-Informatics on NGS short read alignment. She is working as an Assistant Professor in the Department of Computer Science and Engineering in Karunya Institute of Technology and Sciences, Coimbatore. Her area of interest is Data Science, Analysis of Algorithms, Data Structures and Theory of Computation.

Dr. Kumudha Raimond is currently working as a Professor in the Department of Computer Science and Engineering in Karunya Institute of Technology and Sciences, Coimbatore. She earned her Doctoral Degree from Indian Institute of Technology, Madras, India. Her research focus is on the development of efficient models using hybrid intelligent techniques for various applications in the area of biometrics, biomedical, bioinformatics, etc. Her other areas of interest are Big Data Analytics, Remote Sensing Image Processing, Biomedical Image Processing, Biomedical Text Analysis, Watermarking and Wireless Sensor Networks. She has a good number of research publications in peer reviewed National and International Journals, Proceedings of International Conferences and Book Chapters to her credit. She is also involved in conducting special sessions in International Conferences, reviewing Journal and International Conference Papers. Besides having 19 years of teaching experience, she also has 3 years of MNC experience at John F. Welch Technology Research Centre, a Research wing of General Electric (GE), Bangalore. She worked as an Energy System Analyst in the Remote Monitoring and Diagnostic Lab of GE. She is a Senior Member of International Association of Computer Science and Information Technology (IACSIT) and Member of Machine Intelligence Research Lab: Scientific Network for Innovation and Research Excellence.

M. Mythily has completed her bachelor and master degree with a specialization of computer science and engineering from Avinashilingam Deemed University and Government college of Technology, Coimbatore, Tamilnadu, India respectively. Currently, She is pursuing her Ph.D degree with a specialization of software engineering at Anna University, Coimbatore. At present working as an Assistant Professor at Karunya Institute of Technology and Sciences. As a wholesome she gained a decade experience in industry as well as in teaching. Her area of interest includes Software engineering, Machine learning and Problem solving techniques. She has published 10+ papers in refereed international journals and conference proceedings.

Dr. A. KethsyPrabavathy received her Bachelor of Engineering degree in 2004, Master of Engineering in 2008 and Ph.D(Computer Science) in 2018. She is currently working as an Assistant Professor in the department of Computer Sciences Technology in Karunya University, Coimbatore. Her research interest is mainly based on Image processing, Image segmentation and Video segmentation.