Predictive Genome Analysis Using Partial DNA Sequencing Data

Nauman Ahmed, Koen Bertels and Zaid Al-Ars
Computer Engineering Lab, Delft University of Technology, Delft, The Netherlands
{n.ahmed, k.l.m.bertels, z.al-ars}@tudelft.nl

***Abstract*—Much research has been dedicated to reducing the computational time associated with the analysis of genome data, which resulted in shifting the bottleneck from the time needed for the computational analysis part to the actual time needed for sequencing of DNA information. DNA sequencing is a time consuming process, and all existing DNA analysis methods have to wait for the DNA sequencing to completely finish before starting the analysis. In this paper, we propose a new DNA analysis approach where we start the genome analysis before the DNA sequencing is completely finished. The genome analysis is started when the DNA reads are still in the process of being sequenced. We use algorithms to predict the unknown bases and their corresponding base quality scores of the incomplete read. Results show that our method of predicting the unknown bases and quality scores achieves more than 90% similarity with the full dataset for 50 unknown bases (slashing more than a day of sequencing time). We also show that our base quality value prediction scheme is highly accurate, only reducing the similarity of the detected variants by 0.45%. However, there is still room to introduce more accurate prediction schemes for the unknown bases to increase the effectiveness of the analysis by up to 5.8%.**

***Index Terms*—DNA Sequencing delay; Prediction; GATK;**

I. Introduction

The decreasing cost of DNA sequencing [1] has enabled scientists to perform genome analysis easily and with increasing resolution for applications ranging from research to clinical diagnostics.

In *Variant Calling* the sequenced DNA sample is compared against a reference genome to find the genetic variations in the sample as opposed to the reference. In this paper, we will use variant calling as case study for predictive genome analysis. Genome Analysis Toolkit (GATK) [2] is a widely-used variant calling pipeline. The stages in the GATK pipeline for detecting SNPs and INDELs are described in [3]

Both *DNA sequencing* as well as *DNA analysis* consume a lot of time before variants are available for further investigation (e.g., diagnostics). High-throughput Illumina DNA sequencing machines (such as the HiSeq 2500) require up to a week to fully sequence the DNA. Similarly, the processing of the large amounts of data by the genome analysis pipeline results in a huge computation time as well.

A lot of effort has been made in the past to accelerate the computation time of individual stages of the pipeline as well as accelerating the whole pipeline using cluster based computing. BWA-MEM is accelerated in [4]. A multithreaded version of Picard tools is presented in [5]. An FPGA acceleration of the PairHMM calculation is given in [6]. A cluster based Spark implementation of the whole GATK pipeline is presented in [7]. As a result of these efforts in reducing the computation time, the process of DNA sequencing is becoming the limiting-factor in the total time required for genome analysis. The process of DNA sequencing takes days to complete [8], while implementations of the genome analysis pipeline on computer clusters can process hundreds of gigabytes of DNA sequencing data in less than two hours [9].

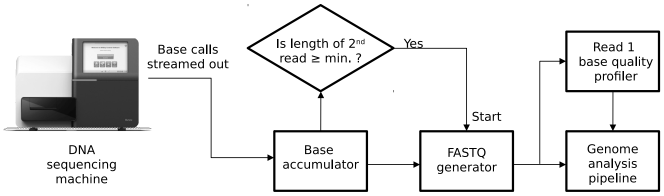

Fig. 1: The proposed scheme to hide DNA sequencing delay

In this paper, we overcome the problem of long DNA sequencing time by partially hiding its delay. This is achieved by starting the genome analysis while the sequencing of the DNA read data is still in progress. In this way, our scheme does not wait for the DNA sequencing process to completely finish before starting the analysis. As the genome analysis is started while the DNA read is still being sequenced, we do not know the values of the last bases of the read and their corresponding base quality scores. Therefore, we introduced an additional stage in the genome analysis pipeline that predicts the value of the unknown bases and their corresponding base quality scores. A patent based on the work in this paper is also filed in Europe [10].

The outline of the rest of the paper is as follows. Our approach to hide the DNA sequencing delay is presented in Section II. The method of predicting unknown bases, their corresponding base quality scores and some additional SAM fields is described in Section III. Experimental results of our proposed techniques are discussed in Section IV. We finally conclude the paper in Section V.

II. Approach

In this paper, we propose a scheme in which the large DNA sequencing time is partially hidden by starting the genome analysis before the DNA sequencing is completely finished. Current high throughput DNA sequencing machines (e.g., Illumina), sequence both ends of a DNA fragment to generate paired-end read data. Paired-end read data allows

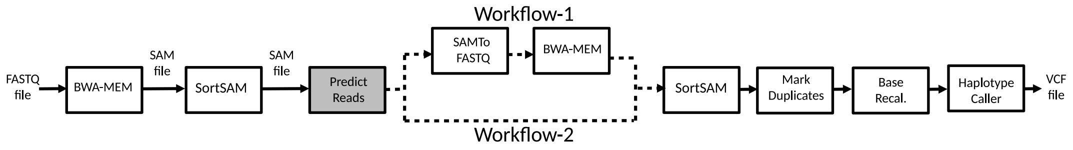

Fig. 2: (a) Workflow 1 and 2. Workflow 1 has two additional stages: `SAMToFASTQ` conversion and remapping with `BWA-MEM`.

more accurate genome analysis as compared to single-end reads. The DNA sequencing technology used by Illumina is known as *Sequencing by Synthesis* (SBS). In SBS, the first and the second read in the paired-end data is generated by sequencing the forward and reverse strand, respectively. The two reads in the pair are sequenced one after the other. Hence, the first read in the paired-end is completely sequenced followed by the second read. Moreover, one base is sequenced at a time. Sequencing a base, produces two values: 1) The actual value of the base (i.e., A, T, C, G or N (ambiguous base)) and 2) The *base quality score* which is the probability of error in the sequencing process.

In high-end Illumina DNA sequencing machines (e.g., HiSeq 2500), generating around 1 Terabyte of data, sequencing a base takes around 30 minutes [8]. Therefore, hiding the sequencing of even a few bases results in large saving in time. In this work, we have reduced the DNA sequencing latency by starting the genome analysis while the DNA sequencing of the second read in a paired-end read is still in progress. In this way, we can save a large amount of time, and the genome analysis can be completed much earlier as compared to the case in which the genome analysis is started after the DNA sequencing is completely finished. Figure 1 shows the proposed scheme. The sequenced bases are streamed out of the DNA sequencing machine while the sequencing is still in progress. The bases are stored in a base accumulator. After enough bases of the second read have been accumulated, the FASTQ file is generated. At the same time a base quality profile of the first read is also generated. The FASTQ file and the base quality profile of the first read are used to perform the genome analysis and complete the unknown part of the reads

III. Methods

In the rest of the paper we will use the following terminology:

1) The paired-end read dataset which would have been generated if the DNA sequencing is allowed to finish is called *original read dataset*.
2) The second read in the paired-end read dataset which would have been generated if the DNA sequencing is allowed to finish is called *original second read*.
3) The paired-end read dataset in which the second read has unknown bases due to incomplete DNA sequencing is called *incomplete read dataset*.
4) The first read in the incomplete read dataset is exactly the same as that in the original read dataset and is simply called *first read*.
5) The second read in the paired-end dataset with unknown bases due to incomplete DNA sequencing is called *incomplete second read*.
6) The number of unknown bases in the incomplete second read is called $n\_unknown$.
7) The paired-end read dataset in which the unknown bases and quality scores of the second read are completed by our read prediction schemes is called *completed read dataset*.
8) The second read in the paired-end read dataset with unknown bases and quality scores that have been completed by our read prediction schemes are called *completed second read.*

A. Input read dataset

In the experiments in this paper, we use the whole exome sequencing of NA12878 dataset. This dataset has 150x coverage with paired-end reads and a read length of 100 base pairs (bp) [11]. The 150x dataset is used to generate subsets of datasets with 50x and 100x coverage. Throughout the paper we will use these three read datasets as the original read datasets. Last tens of bases of the second reads of these datasets are clipped to form the incomplete read datasets.

B. Workflows

As described above we have reduced the DNA sequencing delay by starting the genome analysis while the second read in the paired-end DNA read data is still being sequenced. Therefore, the values of the last bases of the second read and their corresponding base quality scores are unknown to us. We have designed a *prediction stage* `PredictReads` that predicts the values of the unknown bases of the second read and their corresponding base quality scores. We have tested the efficacy of our prediction stage using two different workflows. Figure 2 shows Workflow-1 and Workflow-2, respectively, of our prediction scheme.

Workflow-1 (WF-1) starts with a FASTQ file in which the last bases of the second read and their corresponding base quality score values are unknown. It first maps the incomplete read dataset using `BWA-MEM`. The mapped reads are sorted (w.r.t. mapping position) using Picard's `SortSAM`. The unknown bases of the second read and their corresponding base quality score are predicted using our `PredictReads` stage. The predicted bases and their corresponding base quality scores are appended at the end of the second read. This SAM file is then converted into a FASTQ file using Picard's `SAMToFASTQ` utility. The output of the `SAMToFASTQ` is a

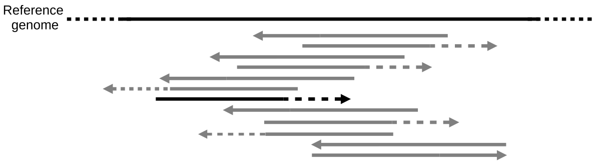

Fig. 3: Set of overlapping reads mapped to a reference genome. Lines with arrowheads are the reads

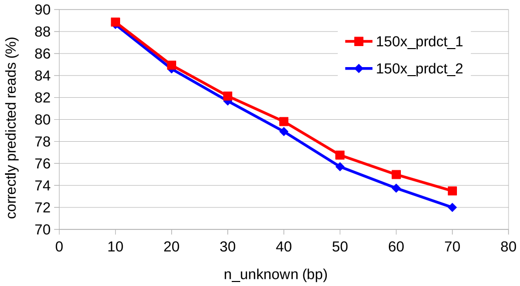

Fig. 4: Prediction accuracy of unknown bases

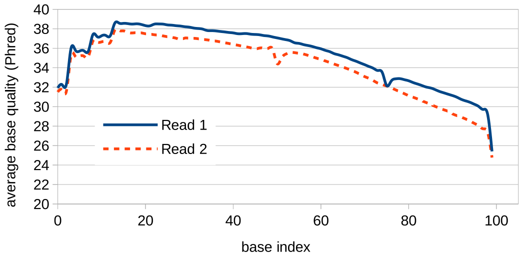

Fig. 5: Average base quality profile of WES 150x dataset.

FASTQ which is a completed read dataset. This FASTQ file is used as an input for a run of the whole GATK pipeline of.

Workflow-2 (WF-2) starts with the same first three stages (from `BWA-MEM` to `PredictReads`) of WF-1. WF-2 then sorts the SAM output file of the read prediction stage and continues to execute the remaining stages in the GATK pipeline after `SortSAM`. The read prediction stage of WF-2 is slightly different from the read prediction stage of WF-1. Apart from predicting the unknown bases of the second read and their corresponding base quality scores, the read prediction stage of WF-2 has to also predict/correct some additional fields of the input SAM file, as described in Section III-C3.

C. Read prediction

The core of the the proposed scheme is the read prediction stage. Read prediction has to perform the following: 1) Predict the values of the unknown bases of the second read, 2) Predict the base quality scores of the unknown bases of the second read, and 3) Predict some additional fields of the input SAM file in case of WF-2.

1) Predicting unknown bases: We predict the unknown bases by detecting the overlap between the incomplete second read and the reads that are mapped close by. Figure 3 shows a set of overlapping reads mapped to a reference genome. The black-colored read is an example of an incomplete second read, while the gray-colored reads are the overlapping reads. The reads containing dotted lines are the incomplete second reads, where the dots show the unknown bases. The arrow on the reads indicate the direction of mapping. We tested various prediction schemes. Here we will describe two prediction schemes that give the best results:

Scheme-1: For each unknown base we take a majority vote of the bases from the overlapping reads to predict the value of the unknown base. An unknown base having no overlap is substituted with the reference genome base located at the corresponding position. The number of bases having no overlap is quite low for high coverage data. For 150x coverage data with the last 30 bases unknown in the second read, there are only 2.22% unknown bases with no overlap. Moreover, only 3.64% of the unknown bases have less than 10 overlapping bases. Hence, in most of the cases the majority vote is taken among a large number of overlapping bases.

Scheme-2: In the second scheme, we predict the unknown bases of the incomplete second read by matching the known bases of the incomplete second read with the overlapping reads. The overlapping read that is matching most closely is used to predict the unknown bases. The bases of the overlapping read covering the unknown bases are used to complete the unknown bases of the incomplete second read. If the overlapping read does not cover all the unknown bases of the incomplete second read, then the remaining unknown bases are predicted using prediction scheme 1.

Figure 4 shows the effectiveness of scheme 1 and scheme 2 in completing the unknown bases in the incomplete reads. The figure shows percentage of the incomplete seconds reads that the scheme is able to complete perfectly (i.e., the completed second reads becoming exactly the same as the original second read). The original read data set has 150x coverage. The figure shows that prediction scheme 1 results in more accurate prediction of unknown bases as compared to prediction scheme 2. Therefore, in this work we will use prediction scheme 1 to predict the unknown bases.

2) Predicting unknown base quality scores: For predicting the base quality scores, we observe the fact that the slope of the average base quality score pattern generated by Illumina machines for the first read and the second read is nearly the same. We used FastQC [12] to plot the average base quality score of the first and second read across the read length. Figure 5 shows a plot of the average base quality score values across the entire read for the 150x original read dataset. It clearly shows that the slope of the average base quality score pattern of the first read and the second read is nearly the same.

To predict the base quality scores of the unknown bases of the second read, we modeled the base quality score pattern of the last $n\_unknown$ bases of the first read with a piece-wise linear function. The number of “pieces” in our piece-wise linear model is equal to $n\_unknown - 1$. Assuming that the slope of the base quality score pattern of the second read is same as the average of that of the first read, our piece-wise linear model is able to correctly predict the unknown base quality scores of the second read.

3) Predicting additional SAM fields: In the workflow WF-2 (Section III-B, apart from predicting the unknown bases and their corresponding base quality scores, we also need to predict some other SAM fields in the read prediction stage. These are: A. Mapping position of the completed second read B. CIGAR string of the completed second read.

A. Mapping position of the completed second read: In both workflows WF-1 and WF-2, mapping using BWA-MEM is the first stage. Therefore, if the incomplete second read is mapped on the reverse strand of the reference genome, then its mapping position in the SAM file will always be $n\_unknown$ positions ahead of the original second read, assuming that there are no deletions and soft clipping in the last $n\_unknown$ bases of the original second read. In our prediction of the mapping position, we assume that there are no deletions and soft clipping in the last $n\_unknown$ number of bases of the original second read, and hence, subtract $n\_unknown$ from the mapping position of the incomplete second read to form the mapping position of the completed second read. If the mapped incomplete second read has soft clipping at the beginning of the read we do not perform this operation. Figure 6 shows a plot of the correctly mapped reads in the completed read dataset. The original read dataset has 150x coverage. A read is regarded as *correctly mapped* if its mapping position is same as the mapping position of the read in the original read dataset. The percentage of correctly mapped reads are shown for two cases: 1) *150*x*_wf1* representing the case of WF-1, which is the percentage of correctly mapped reads after the completed read dataset is remapped using BWA-MEM, and 2) *150*x*_wf2* representing the case of WF-2, which is the percentage of correctly mapped reads in the completed read dataset, in which the mapping positions of the completed second reads are predicted using the method describe above. The plot shows that the reads in WF-2 have very high mapping accuracy. On the other hand, remapping the reads after the prediction of the unknown bases of the second read, as done in WF-1, greatly reduces mapping accuracy.

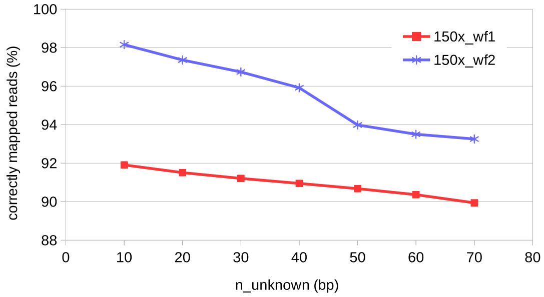

Fig. 6: Percentage of correctly mapped reads

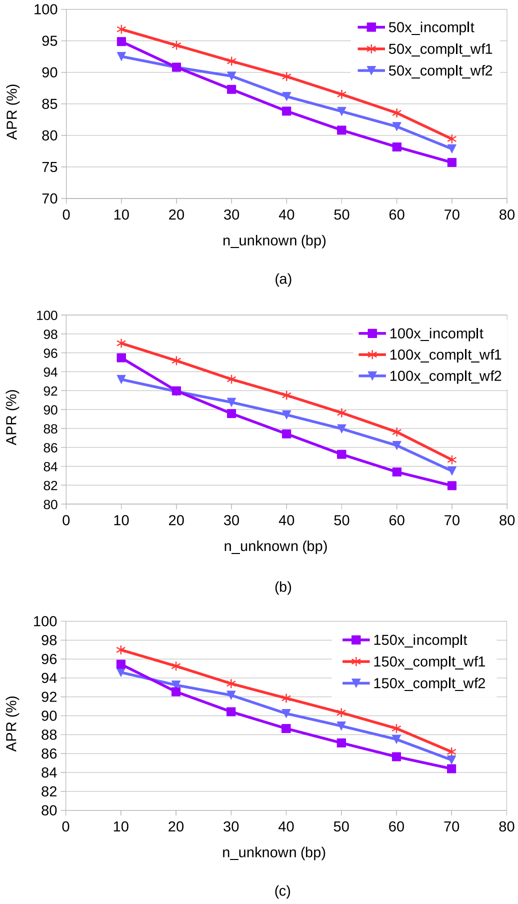

Fig. 7: (a) Effectiveness plot for 50x coverage data. (b) Effectiveness plot for 100x coverage data. (c) Effectiveness plot for 150x coverage data.

B. CIGAR string of the completed second read: In WF-2, we reevaluate the CIGAR string of the completed second read by performing a semi-global alignment between the completed second read and the substring of reference genome. Let T be the reference genome and $T[a,b]$ be its substring starting from reference position a and ending at position b. The substring of the reference genome used in the semi-global alignment is $T[p-10, p+qlen+10]$, where p is the predicted mapping position of the completed second read and the $qlen$ is the length of the completed second read. If the mapped incomplete read is soft clipped at either end, the CIGAR string is not reevaluated. Instead, the first or last operation in the CIGAR string of the incomplete second read is extended by $n\_unknown$ depending upon the mapping strand of the incomplete second read. If the incomplete second read is mapped on the reverse strand, the first CIGAR operation is extended and vice versa.

IV. Results

For evaluating our proposed scheme we implemented the read prediction stage in practice. Our read prediction stage is capable of predicting the unknown bases, unknown base quality scores and the additional SAM fields as described in Section III-C. The prediction of additional SAM fields is only required in WF-2. Our prediction stage requires the reference genome and base quality profile of the first read as input in addition to the SAM file of the DNA reads mapped and sorted w.r.t. mapping position. We used UCSC hg19 as the reference genome. We also used *1000G_phase1*, *dbsnp_138* and *Mills_and_1000G_gold_standard* as the known SNP and indel sites for the `BaseRecalibrator` stage. All the Picard and GATK tools are run with default settings. BWA-MEM also is run with default settings except the use of `-M` option, essential for Picard compatibility.

We first run the GATK pipeline with original read dataset to generate *orig* the set of variant calls (VCF file) that we compare our techniques with. We then clipped the last tens of bases of the original second reads to generate incomplete read dataset. The set of variant calls (VCF file output) of the GATK pipeline computed for the incomplete read dataset will be called as $incomplt$. To test our prediction scheme, we predict the unknown bases, unknown base quality scores and some additional SAM fields (only in WF-2). The sets of variant calls generated by running WF-1 and WF-2 of Figure 2 are called as $complt\_wf1$ and $complt\_wf2$, respectively. *Precision* and *recall* (sensitivity) can be defined as:

$$precision = \frac{TP}{TP + FP} \times 100\% \quad (1)$$

$$recall = \frac{TP}{TP + FN} \times 100\% \quad (2)$$

where TP, FP and FN are the true positives, false positives and false negatives, respectively, . In order to evaluate the effectiveness of the predictive analysis workflows defined in this paper, we use the area under the precision-recall (APR) curve as our metric. APR indicates the effectiveness of a pipeline in identifying as much as possible correct variants (high TP) while identifying as little as possible incorrect variants (low FP). In the ideal case, APR is equal to 100%, and the closer the APR is to 100%, the more effective the workflow is. This definition of APR is the same as the one used by [13] to evaluate the effectiveness of various variant calling pipelines. The APR is calculated for $complt\_wf1$, $complt\_wf2$ and $incomplt$, with respect to *orig*. We use `RTG` tools [14] to calculate the precision-recall graph. This is then further used to evaluate the APR of our workflows.

Figure 7 shows the effectiveness in terms of APR of $incomplt$, $complt\_wf1$ and $complt\_wf2$ with respect to $orig$, while clipping 10, 20, .. up to 70 bases of the original second read. Figures 7a, 7b and 7c show the APR plot for 50x, 100x and 150x coverage data, respectively. *Scheme-1* explained in Section III-C1 is used to predict the unknown bases in the second read. The figures show that the APR decreases almost linearly with increasing number of unknown bases. At the same time, the overall APR of all sets of variant calls increases as the data coverage is increased from 50x to 150x. The figures also show that WF-1 is the most effective workflow to accurately call variants of incomplete reads, consistently achieving a higher APR than WF-2 and $incomplt$ for all cases. For an increasing $n\_unknown$, the APR of WF-2 gets gradually closer to that of WF-1, but never actually reaching it. Although WF-2 has a much higher read mapping accuracy than WF-1 (according to Figure 6), Figure 7 shows that WF-1 has a better APR than WF-2 for all cases. This degradation in APR of WF-2 as compared to WF-1 can be attributed to the prediction scheme (scheme-1 of Section III-C1) that we used to predict the the unknown bases. In scheme-1, we take a majority vote of the bases from the overlapping reads to predict the value of the unknown base. This majority vote may cause a true variant to be overshadowed and hence, being missed in WF-2. On the other hand, in WF-1 the unknown bases of the incomplete second read are predicted, and then the completed read dataset is remapped to the reference genome. This causes some of the reads to be mapped to a different position than the initial mapping and hence, do not overshadow a true variant.

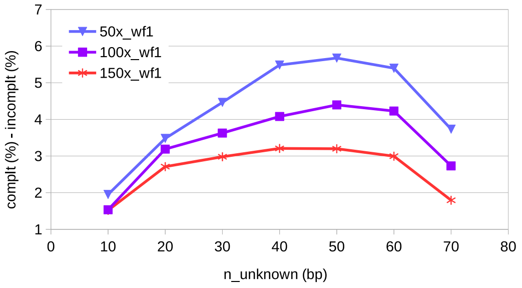

Fig. 8: The difference between the APR of WF-1 and $incomplt$ for a range of $n\_unknown$ and coverage values.

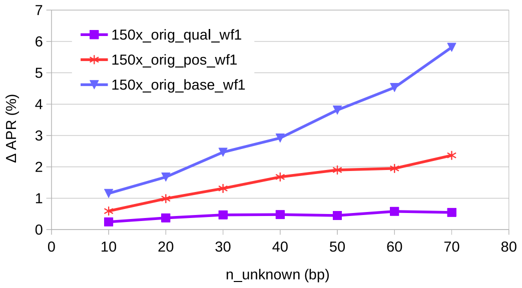

Fig. 9: Increase in the APR after assuming original values for mapping position, unknown base quality score or unknown base value.

Figure 8 shows a plot of the difference between the APR of WF-1 and $incomplt$ for a range of $n\_unknown$. As pointed earlier, the figure shows that WF-1 is always better than $incomplt$. The difference is initially small but increases with increasing $n\_unknown$, then peaks at around

$n\_unknown = 50$, and finally falls back down. The plot also shows that the difference in the APR of WF-1 and $incomplt$ is much higher at lower coverage than at higher coverage. For $n\_unknown = 50$, the difference is 5.7% and 3.2% for 50x and 150x coverage, respectively. This means that with increasing coverage the APR of $incomplt$ increases at a much higher rate than WF-1. Therefore, our method of predicting unknown bases and their corresponding base qualities has a bigger comparative impact on the APR with lower coverage as compared to $incomplt$.

We also studied the effect of accurate prediction of different parameters of the incomplete second read. We make three different assumptions in WF-1. 1) *orig_pos*: We assume that the mapping position of the reads going into the read prediction stage of WF-1 is exactly the same as mapping position of the reads in the original read dataset. The unknown bases and the corresponding base quality scores are predicted as described in Section III-C1 (scheme-1) and III-C2, respectively. 2) *orig_qual*: We assume that the unknown base quality scores of the incomplete second read are predicted with ideal accuracy (i.e., they are exactly the same as in original second read). The unknown bases of the incomplete second read are predicted as described in Section III-C1(scheme-1). 3) *orig_base*: We assume that the unknown bases of only those incomplete second reads in which the mapping positions are correct (same as that of original second read), are predicted with ideal accuracy (i.e., they are exactly the same as in original second read). The mapping positions of the incomplete second reads are predicted using the same method as described in Section III-C3. Unknown base quality scores of the incomplete second read are predicted as described in Section III-C2. Figure 9 shows the increase in APR for each of the three cases. $\Delta APR = 150\text{x}\_y\_wf1 - 150\text{x}\_complt\_wf1$, y is $orig\_pos$, $orig\_qual$ or $orig\_base$. Figure 9 shows that the assumption of accurate prediction of the unknown base values ($orig\_base$) causes a much higher increase in APR as compared to the other two assumptions. For $orig\_qual$, the increase in APR is quite small. Hence, there is a very little room for improvement in our method of predicting the unknown base quality scores. For $n\_unknown = 50$, accurate prediction of unknown base quality scores and read mapping positions cause an increase of 0.45% and 2% in the APR, respectively. On the other hand, accurately predicting the value of the unknown bases of only those incomplete second reads which have been mapped correctly can help to increase the APR by 3.8%. Therefore, we can conclude that the method of predicting unknown bases can be further improved to achieve more effectiveness.

V. Conclusion

In this paper, we proposed a predictive genome analysis approach based on the idea of starting the genome analysis before the DNA sequencing is completely finished. We introduced an additional stage in the GATK pipeline to predict the unknown bases and their corresponding base quality scores, due to incomplete DNA sequencing. Two workflows were proposed to achieve this purpose. The results showed that our method of predicting the unknown bases and quality scores achieves more than 90% similarity with the analysis performed on the full dataset for 50 unknown bases (slashing more than a day of sequencing time).

We also measured the impact of accurate prediction of unknown bases, unknown base quality scores and the read mapping position to improve the effectiveness of the workflows in identifying the variants. Results show that our base quality and read mapping position prediction scheme is highly accurate. with 50 unknown bases, ideal prediction of the value of the base quality scores and read mapping position gives only 0.45% and 2% higher similarity with analyzing the full dataset, respectively. However, accurately predicting the value of those unknown 50 bases can achieve a 3.8% higher similarity, meaning that more effective base prediction methods can achieve an even higher analysis accuracy.

Acknowledgment

This work is supported by the Faculty Development Program of the University of Engineering and Technology Lahore, Pakistan.

References

[1] K. Wetterstrand, "DNA Sequencing Costs: Data from the NHGRI Genome Sequencing Program (GSP)," Available at: www.genome.gov/sequencingcosts, 2016, Accessed [15 October, 2016].

[2] A. McKenna *et al.*, "The genome analysis toolkit: A mapreduce framework for analyzing next-generation dna sequencing data," *Genome Research*, vol. 20, no. 9, pp. 1297–1303, 2010.

[3] G. van der Auwera, "GATK Best Practices," https://software.broadinstitute.org/gatk/best-practices/, 2016.

[4] N. Ahmed, V. Sima, E. Houtgast, K. Bertels, and Z. Al-Ars, "Heterogeneous Hardware/Software Acceleration of the BWA-MEM DNA Alignment Algorithm," in *ICCAD'15*, 2015, pp. 240–246.

[5] A. Tarasov *et al.*, "Sambamba: fast processing of NGS alignment formats," *Bioinformatics*, 2015.

[6] J. Peltenburg, S. Ren, and Z. Al-Ars, "Maximizing Systolic Array Efficiency to Accelerate the PairHMM Forward Algorithm," in *BIBM'16*, 2016, pp. 758–762.

[7] H. Mushtaq and Z. Al-Ars, "Cluster-based apache spark implementation of the gatk dna analysis pipeline," in *BIBM*, Nov 2015, pp. 1471–1477.

[8] Illumina, "Illumina HiSeq-2500 System Specifications," https://www.illumina.com/documents/products/datasheets/datasheet_hiseq2500.pdf, 2015.

[9] H. Mushtaq, F. Liu, C. Costa, G. Liu, P. Hoftsee, and Z. Al-Ars, "SparkGA: A Spark Framework for Cost Effective, Fast and Accurate DNA Analysis at Scale," in *ACM-BCB*, August 2017, pp. 148–157.

[10] Z. Al-Ars, N. Ahmed, and K. Bertels, "Early DNA Analysis Using Incomplete DNA Datasets," Patent, European Patent no. NL 2017750 (pending), 2016.

[11] G. Highnam *et al.*, "An analytical framework for optimizing variant discovery from personal genomes," *Nature Communications*, vol. 6, no. 6275, 2015.

[12] S. Andrews, "FastQC: a quality control tool for high throughput sequence data," http://www.bioinformatics.babraham.ac.uk/projects/fastqc/, 2010.

[13] S. Hwang *et al.*, "Systematic comparison of variant calling pipelines using gold standard personal exome variants," *Scientific Reports*, vol. 5, no. 17875, 2015.

[14] J. Cleary *et al.*, "Comparing variant call files for performance benchmarking of next-generation sequencing variant calling pipelines," *bioRxiv*, 2015.