An Effective and Efficient Algorithm for Detecting Exact Deletion Breakpoints from Viral Next-Generation Sequencing Data

Ji-Hong Cheng[2], Wen-Chun Liu[3,4], Ting-Tsung Chang[3,4], Sun-Yuan Hsieh[2], Vincent S. Tseng[1,*]

[1]*Department of Computer Science, National Chiao Tung University,* Taiwan, ROC
[2]*Department of Computer Science and Information Engineering, National Cheng Kung University,* Taiwan, ROC
[3]*Department of Internal Medicine, College of Medicine, National Cheng Kung University, Tainan 701, Taiwan*
[4]*Infectious Disease and Signaling Research Center, National Cheng Kung University, Tainan, Taiwan*
**Email: vtseng@cs.nctu.edu.tw*

2020 International Computer Symposium (ICS) | 978-1-7281-9255-0/20/$31.00 ©2020 IEEE | DOI: 10.1109/ICS51289.2020.00038

***Abstract*— The COVID-19 pandemic has caused serious damage to the health, life and, economic stability of human beings all over the world. In order to combat this disease, researchers from all over the world, including computer scientists, are beginning to engage in cross-regional cooperation to conduct research on SARS-CoV-2. One of the latest reports pointed out that the sequence deletion of the specific region of the SARS-CoV-2 genomic is related to its viral infectivity. In addition, the sequence deletion of this specific region is also found in Hepatitis B Virus (HBV), and Hepatocellular carcinoma (HCC). Through next-generation sequencing (NGS) technology, the sequence data of biological genomes can be quickly obtained, but the number of short reads generated by NGS is often as high as one million big data. It is a challenge to detect the information necessary to provide the exact sequence deletion breakpoint from these NGS data, especially in the sequence data of highly variable viral genomes. In our previous research, we proposed VirDelect, a bioinformatics tool that can detect exact breakpoints in Viral NGS data. In this paper, a new method, One-base Alignment Plus (OAP), is proposed to enhance further the core VirDelect algorithm, in order to improve the sequence deletion detection correctness. We use the simulated data of SARS-CoV-2 and HBV with different deletion lengths and the real data of HBV to conduct experiments and evaluate the correctness. The experimental results showed that VirDelect+OAP was able to find deletions that VirDelect could not find in the simulation data, and in the real data, the correctness of VirDelect+OPA was raised effectively.**

Keywords*— *Next-generation sequencing, Big data, Viral deletion detection, COVID-19, Hepatitis B Virus

I. INTRODUCTION

The application of computer science and information technology to develop effective virus genome analysis methods is currently an important issue. SARS-CoV-2 caused the COVID-19 pandemic, which has caused serious harm to the health, economic stability, and lives of people all over the world. According to the latest statistics from the World Health Organization (WHO), more than 28 million people have been infected with SARS-CoV-2, causing more than 900,000 deaths. The number of SARS-CoV-2 infections is still increasing. Computer scientists and many cross-field researchers are actively cooperating in the development of technologies and products to fight COVID-19, such as the development of drugs and vaccines and the analysis of SARS-CoV-2 bioinformatics tools [1]. Current research has pointed out that the deletion of SARS-CoV-2 is related to viral infectivity [2]. Another recent study discovered the position of several smaller open reading frames (ORFS) in the genome sequence of SARS-CoV-2. The specific deletion fragments are approximately 3(nt) to 39(nt) in length [3][4]. In addition, in the case of patients infected with Hepatitis B Virus (HBV), if a deletion fragment is found in the pre-S region of the virus genome sequence, it can be linked with clinical presentations of chronic HBV infection and HCC [5]. It is important for virus researchers to have the ability to quickly obtain the virus genome sequence and apply computer science techniques to accurately detect and analyze and structural variations, such as deletions, as well as to obtain information about the relationship between a virus and the disease it causes, which is an important key to the success of the fight against the disease.

Next-Generation Sequencing (NGS) technology, also known as Massively Parallel Sequencing (MPS), is a hardware and software device that quickly completes biological sequence sequencing using optical or semiconductor technology. NGS provides ultra-high-speed, high-throughput sequencing technology for biological sequences. For example, approximately 10 years and 2.7 billion US dollars went into the Human Genome Project started in 1991 to complete a set of human genome reference sequences, but now, through NGS technology, it takes less than two hours to complete the sequencing of the human genome. NGS technology is also used in the analysis of virus sequences, such as the Hepatitis B Virus (HBV) deletion [5] and SARS-CoV-2 quasispecies [6]. NGS can generate and sequence a large number of short reads at the same time. However, because of the large number of reads, computer science and bioinformatics scientists must use effective information technology to design algorithms and application software in order to use these tools to analyze NGS reads and obtain useful information that can be applied clinically or in research. Common software tools are used for analyzing NGS, including De novo fragment assembly tools [7] used to assemble and splice reads, and reads mapping tools [8] used to align reads with reference genome sequences to obtain the most likely corresponding correct position in the reference sequence. In our previous study, we discussed several methods for detecting deletions, such as Breakdancer [9], PEMer [10] and Breakpointer [11], which can provide detection of approximate sites, and SoftSV [12] and Pindel [13], which can provide precise position points but are suitable only for analyzing pair-end NGS data. However, not all NGS platforms have pair-end reads data. For example, Ion Torrent sequencing platforms can generate single-end data but cannot generate pair-end reads data [14].

In our previous published study [15], we provided an algorithm, VirDelect, which can be used to detect the deletion fragments in the genome sequence of HBV in the NGS reads data of single-end and pair-end platforms. However, the viral genome sequence has a higher mutation rate than the human genome sequence [16] [17], so there is a greater difference between the reference sequence and the read, making the NGS read more difficult to accurately align in the right position. The disappearance of a long fragment such as a deletion in the genome sequence of the virus will make the structural variation analysis of the NGS virus sequence more

DOI 10.1109/ICS51289.2020.00038

complicated. This is a challenge for designing virus sequence structural variation analysis methods in NGS read data. Therefore, we give priority to an alignment method that can accurately process high-variation sequences so that it can provide more information about the read position, even if it may have a higher computational cost. In our previous study, we used a split alignment-based method to design our algorithm. The experimental results indicated that the proposed method, VirDelect, when compared with Pindel [14], indeed was more accurate.

In this research, we mainly propose a new method, One-base Alignment Plus (OAP), which is used to improve the correctness of VirDelect in order to detect the exact deletion position. In the VirDelect architecture, one base alignment is used as the method by which to calculate the sequence scores and improve the comparison efficiency, but it is impossible to obtain the continuous alignment information for the sequence base using this method. To increase the correctness of VirDelect alignment, the One-base Alignment Plus (OAP) method proposed in this study allows VirDelect to obtain continuous sequence alignment information while maintaining the same efficiency. We use the NGS simulation data of HBV and SARS-CoV-2 and real HBV data to compare the correctness of VirDelect and VirDelect+OAP at a precise location. It was found that VirDelect+OAP found deletion fragments that VirDelect could not find in the simulation data, and that for the real data VirDelect+OAP, VirDelect's correctness could indeed be improved.

The outline of this paper is as followings: In Section II, we review related work to this research. Section III, introduces the proposed method. Section IV, evaluates the experimental results of the proposed method. Section V provides the discussion and conclusion to this paper.

II. RELATED WORK

This chapter introduces research related work , including: (A)Viral Deletions in HBV and SARS-CoV-2 and (B) VirDelect. The following describes the related methods.

A. Viral Diversity in HBV and SARS-CoV-2

Chronic infection with hepatitis B virus (HBV) greatly increases the risk for liver cirrhosis and hepatocellular carcinoma (HCC). HBV is classified into ten genotypes (A–J) with an intergenotypic diversity of at least 8% in the full genome sequence 1. Over 40 related sub-genotypes are also identified based on a 4 to 7.5% divergence of HBV full genome[17][18][19]. The HBV genome has a highly compact organization, with four overlapping open reading frames (ORFs - preS/S, polymerase, preCore/core, and X) and regulatory elements[20]. The viral polymerase arises from the translation product of the 3.5 kb pre-core mRNA and pgRNA, that serves as template for reverse transcriptional synthesis of viral DNA. Due to the absence of proofreading activity, the HBV polymerases/RT leads to the introduction of random mutations into HBV genome[21]. HBV single nucleotide variants (SNVs) and deletion mutations linked with clinical presentations of chronic HBV infection and HCC. Some previous studies have indicated that deletion hotspots were located in the preC/C gene, preS region and BCP region in both genotypes B and C HBV. The preS region of HBV full genome, which comprises preS1 and preS2, had the greatest deletion frequencies and the most complex deletion patterns[22][5][23].

COVID-19 is a viral respiratory illness caused by a new coronavirus called SARS-CoV-2. The emergence of SARS-CoV-2 has led to the current global coronavirus pandemic and more than one million infections since December 2019[2]. The viral RNA genome of SARS-CoV-2 encodes several smaller open reading frames (ORFs) such as ORF1ab, ORF3a, ORF6, ORF7a, ORF7b, ORF8 and ORF10 located in the 3'region. These ORFs are predicted to encode for the replicase polyprotein, the spike (S) glycoprotein, envelope (E), membrane (M), nucleocapsid (N) proteins, accessory proteins, and other non-structural proteins (NSP)[2][24][25][26]. The rapid spread of SARS-CoV-2 raises intriguing questions such as whether its evolution is driven by mutations. Some variation sites in SARS-CoV-2 have been reported including ORF1ab, S, ORF3a, ORF8 and N regions, among which positions 28144 in ORF8 and 8782 in ORF1a. In addition, SARS-CoV-2 is rapidly moving across countries and genomes with new mutation hotspots are emerging[16]. Several reports have pointed out the deletions in throughout the viral genome, often producing deletion variants of non-structural and accessory proteins that may have direct implication upon viral infectivity [2].

B. Introduction of VirDelect

VirDelect [15] is a bioinformatics tool that can detect the of sequence deletion position in NGS virus data. The system architecture of VirDelect and the detection deletion process are shown in Fig. 1.

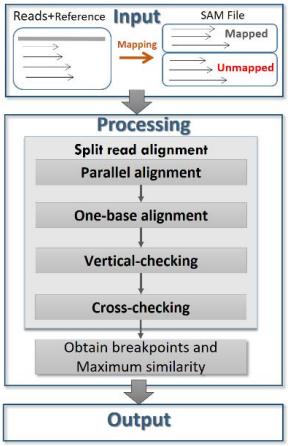

Fig. 1. Flow chart for detecting deletions using VirDelect.

1) Input: This part of the process is the data pre-processing stage. After the user obtains the NGS read data, the process by which the data enables VirDelect to perform the function of detecting the deletion position is explained as follows: First, the NGS short read data must be mapped with the reference sequence. Through this step, the read can obtain the reference position information. Bowtie2 [27] and BWA [28] were used as the NGS tool software in this step. After the user mapping is read, the SAM file can be obtained, which records the reference position for each read, including unmapped reads. Users can use another NGS tool software program, SAMtools, to take out the unmapped read in the SAM file and save it as an unmapped read SAM file.

2) Processing: After the user obtains the unmapped read SAM file, the specified reference is entered, and VirDelect can detect the deletion position. Split read alignment is the core architecture of VirDelect. It includes parallel alignment, one-base alignment, and checking. At this stage, each read is taken out of the SAM file in order. VirDelect cuts the read into prefixes and suffixes aligned so as to reference based on to the initial length specified by the user. The following explains the purpose and principle of the three algorithms in split read alignment:

a) *Parallel alignment:* Although the read is cut into a

$$Similarity(r) = \frac{number\ of\ the\ same\ bases\ in\ the\ read}{unmber\ of\ all\ bases\ in\ the\ read}, \quad (1)$$

$$deletion \notin any\ bases$$

pair of prefixes and suffixes, they can be aligned with the reference separately, and they do not have to be aligned along with the reference. For example, there is no need to fix the prefix and then only move the suffix to calculate the sorting similarity. The prefix only has to calculate the similarity with the reference and obtain the score matrix, after which the result can be checked with the suffix score matrix to determine whether the length is within the range specified by the user.

b) *One-Base alignment (OA):* When the read is cut into prefixes and suffixes, one-base alignment will be performed separately in parallel alignment. The similarity calculation formula is as shown in (1). OA is used to calculate the similarity for each position of the prefix and the reference sequence, and these data are compiled into the score matrix, as is also the case with the suffix. However, the computational cost of calculating similarity in this manner is too high. When the OA algorithm calculates the similarity of the prefix of length L and the suffix similarity, it retains the score matrix calculated at this time. When the prefix and suffix of L + 1 have to calculate the score matrix, the score matrix of L can be used to greatly reduce the number of calculations. Suffix can also use a similar principle to reduce the computational cost. This part will be explained in detail in the proposed method section. What we want to emphasize here is that this similarity score calculation formula cannot retain the information of sequential hits. We know that the sequential hits of the sequence are what we need to find. There should be another calculation formula to give it a higher score. However, this will also make the One-base alignment reduction algorithm more complicated.

c) Vert*ical-checking and Cross-checking*: After obtaining the score matrix through the one-base alignment algorithm, VirDelect can obtain the score matrix for the paired prefix and suffix in each reference position. If one takes the maximum value of the prefix and suffix score matrix and calculates whether the length of the corresponding position is within the range defined by the user, this is called the legal length, and the maximum value is the local maximal similarity (LMS). Currently, the largest LMS in all combinations is called the global maximal similarity (GMS). If all the prefix and suffix pairs for this read are lower than this GMS, then VirDelect will determine that the position corresponding to the GMS is the correct deletion position for this read. Another situation is where the similarity score of a prefix and suffix pair is greater than the current largest GMS, but its corresponding position is not a legal deletion length, so it cannot replace the current GMS. Nevertheless, the score matrix of this prefix and suffix pairing may hide a score greater than GMS. At this time, cross-checking will check the score and position of the two score matrices and use the current GMS as the upper limit of the search. This effectively reduces computational costs.

3) Output: VirDelect outputs a text file containing the exact breakpoint of deletion and the number of times it was detected. The deletions found are sorted according to the number of detections.

III. PROPOSED METHOD

A. Overview of One-base Alignment Plus

One-base Alignment Plus (OAP) redefines the formula for calculating a similarity score, and a score table and hit table are used to improve alignment efficiency. In this section, we first introduce the formula for the OAP algorithm used to calculate the similarity score, which can cause the sequence to have more sequential hits in the base and will in turn lead to a higher score and to improvements in the correctness of VirDelect. Scores can be obtained by using the score table in the OA prefix and suffix, but in OAP, a hit table must be used to obtain information on whether to use sequential hits. Therefore, after introducing the similarity score, we then define the hit table, score table, and initialize table. After clarifying the above definitions, the subsequent OAP prefixes and OAP suffixes can be more clearly understood.

B. Definition of One-base Alignment Plus

1) Similarity score: If there are two sequences of lengths X and Y, and i represents the position of the base in the sequence, then x[i], y[i] represent the base of the X and Y sequences in the same position. The similarity score is defined as shown in (2). Its purpose is to allow sequences with more sequential hits to obtain higher rewards and to increase the probability that VirDelect will find the exact deletion breakpoint.

If i was position, n was length of prefix or suffix:

$$Similarity\ score(i)\ of\ n = \sum_{k=i}^{n} award(k), denote\ Sn(i) \tag{2}$$

$$award(k) = \begin{cases} 2, if\ x[k] = y[k]\ and\ x[k-1] = y[k-1] \\ 1, if\ \ x[k] = y[k] \\ 0, otherwise \end{cases}$$

2) Score table and Hit table: As shown in Fig. 2 and Fig. 3, we use a matrix to represent the hit table and score table. The hit table records the information for each hit position. The score table records the similarity score for each position. The score table is similar to the score matrix, but the elements in the score table are not sorted based on similarity. The score ranges from small to large. In the hit table, if the result of the base aligned with the reference at a certain position is hit, it records the information as true, if not, it records the information as false. The score table records the score for each position alignment result. In OAP, every time a new hit table and score table are generated, they are stored for the use in the next OAP. For example, in the calculation of the prefix, after a prefix of length L is calculated by the OAP, it stores the hit table and the score table and prepares to use the prefix L+1. For a suffix of length L, the hit table and score table stored after OAP calculation are used for the L-1 suffix calculation OAP. Here, we define several symbols to represent the score table and hit table for prefixes or suffixes of different lengths:

When i was position, n was length of prefix or suffix:
HTn: *represent the prefix or suffix Hit table of length n*
HTn(i): *represents the similarity score of hit table with length n at position i*
STn: *represent the prefix or suffix Score table of length n*
STn(i): *represents the hit record of the Score table with length n at position i*

3) Initialize table: When the prefix generates the score matrix, if OAP finds that STn-1 is empty, then OAP calls the initialize table so that each position in HTn and STn has a corresponding value. This is because in the prefix, OAP uses HTn-1 and STn-1 to obtain HTn, but the first HTn and STn do not have any value. In VirDelect we define the initial length of the prefix, UL, which represents the length of the first prefix, and the initial length of the suffix is the read length minus UL. When the length of the prefix is equal to UL, the elements of HTn-1 and STn-1 are empty, and the similarity score of each position must be calculated according to Formula (2) to obtain STn. When the length of the suffix is equal to the length of the read minus UL, the elements of HTn+1 and STn+1 are empty, and the similarity score of each position must be calculated according to Formula (2) to obtain STn.

C. Prefix of One-base Alignment Plus

Fig. 2 shown below provides an example to illustrate the initial OAP prefix table calculation process and how it uses the hit table and score table to generate a new score table.

1) Initial table: Here we use |P|=n to indicate that the length of the prefix is n. Assuming that the initial length UL of the prefix specified by the user is 5, the initial table step will start when |P|=5. As described in the initialize table, we will obtain ST5 and HT5. Here, we use an example to illustrate the method for generating HT5. Assuming that the sequence of |P|=5 is CCTAC, when calculating HT5(i) every time, the last base C of |P|=5 is taken out and aligned with the reference. The comparison results in HT5(i) are then recorded, and if there is a hit, it is considered to be true; if there is no hit, it is considered to be false.

2) Generate new Score table: By querying STn, it is possible to determine the similarity score for each position in |P|=n. In this step, OAP uses STn-1, HTn-1, and HTn to obtain STn according to the rules established in Formula (3).

When i was position, n was length of prefix:

$$STn(i) = STn\text{-}1(i) + award(i).$$

$$award(i)\begin{cases} 2\,, if\ HT_{n-1}(i) = HT_n(i) = True \\ 1\,, if\ HT_{n-1}(i) = False,\ HT_n(i) = True \\ 0\,, HT_n(i) = False \end{cases} \quad (3)$$

Fig. 2 is used as an example to illustrate how OAP obtains the similarity score for position i=0, ST6(0) when |P|=6. First, OAP queries ST5(0) scores and HT5(0) records in ST5 to obtain ST5(0)=6 and True. Then, the last base C of |P|=6 is taken out and aligned with the position corresponding to the reference. The result of the alignment is hit, so HT6(0)=True. Now, the message obtained by OAP is HT5(0)=True, and HT6(0)=True represents a sequential hit, so award=2, and ST6(0)=6+2=8. Then we explain how to take the score of S6(1) when |P|=6. By querying ST5 and HT5, we know that ST5(1)=3 and HT5(1)=True, and the result of calculating HT6(1) is False, so award=0. ST(6) = ST(5) + award = 3 + 0 = 3.

Pos: 0123456789
Ref: CCTTCCTGTAA
|P|=5 CCTAC
|P|=6 CCTACC

Socre table

|P|=5, position	0	1	2	3	4	5
Score	6	3	0	1	5	1
|P|=6, position	**0**	**1**	**2**	**3**	**4**	**5**
Score	8	3	0	1	5	1

Hit table

|P|=5, position	0	1	2	3	4	5
Hit	T	T	F	F	F	F
|P|=6, position	**0**	**1**	**2**	**3**	**4**	**5**
Hit	T	F	F	F	F	F

Fig. 2. Prefix score matrix was generated by OAP

D. Suffix of One-base Alignment Plus

Fig. 3 below provides an example illustrating the initial table calculation process used to obtain the suffix of OAP and how it uses the hit and score tables to generate a new score table. It should be noted here that the suffix is different from the prefix, where STn+1 is used to generate STn.

1) Initial table: Here, we use |S|=n to indicate that the length of suffix is n. Assuming that the initial length of the suffix specified by the user is 6, the initial table steps will begin when |S|=6. As described in the initialize table, we obtain ST6 and HT6. Here, we use an example to illustrate the method used to generate HT6. Assuming that the sequence of |S|=6 is CATTTC, OAP takes the first base C aligned with the reference for each HT6(i) of |S|=6 and records the alignment result in HT6(i). If the result is a hit, it is considered to be true; if there is no hit, it is considered to be false.

When i was position, n was length of prefix:

$$STn(i) = ST_{n+1}(i\text{-}1) + award(i).$$

$$award(i)\begin{cases} -2\,, if\ HT_{n+1}(i-1) = HT_n(i) = True \\ -1\,, if\ HT_{n+1}(i-1) = False,\ HT_n(i) = True \\ 0\,, HT_n(i) = False \end{cases} \quad (4)$$

Pos: 0123456789
Ref: CATTACTGTAT
|S|=6, CATTTC
|S|=6, CATTTC
|S|=5 ATTTC
|S|=4 TTTC

Socre table

|S|=6, position	0	1	2	3	4	5
Score	8	1	1	2	2	2
|S|=5, position	**0**	**1**	**2**	**3**	**4**	**5**
Score	3	6	1	1	2	2

Hit table

|S|=6, position	0	1	2	3	4	5
Hit	T	F	F	F	F	T
|S|=5, position	**0**	**1**	**2**	**3**	**4**	**5**
Hit	F	T	F	F	T	F

Fig. 3. Suffix score matrix was generated by OAP

2) Generate new Score table: By querying STn, the similarity score of each position of |S|=n can be obtained. In this step, OAP uses STn+1, HTn+1, and HTn to obtain STn based on Formula (4). We use Fig. 3 as an example to illustrate how OAP obtains the similarity score of position i=1, ST5(1) when |S|=5. First, OAP obtains ST6(0)=8 and HT6(0)=True after querying the records with i=0 in ST6 and HT6. Then, the first base C of |S|=5 is taken out and aligned with the position corresponding to the reference to obtain hit, and the alignment result is recorded in HT5(1)=True. Now, the message obtained by OAP is HT6(0)=True, and HT5(1)=True represents a sequential hit, so award=-2, ST5(1)=8-2=6. Then, we explain how to get the score for ST5(2) when |S|=5. By querying ST6 and HT6, we know that ST6(1)=1 and HT6(1)=False. After calculating HT5(2) as False, award=0. ST5(2) = ST6(1) + award = 1 + 0 = 1.

3) Generate new Score table in position zero: In Fig. 3, the score of ST5(0) can be observed. It can be seen that the score of position i=0 makes it impossible to obtain the score from ST6, and the 5 bases of the suffix must be aligned with the reference to get ST5 (0). Therefore, for any length of the OAP suffix, STn(0) at the position of i=0, a score from STn+1 cannot be obtained, and each base of |S| = n must be aligned with the reference to obtain STn(0).

IV. RESULT

A. Data Set and Environment

In order to evaluate the correctness of the method proposed in this study, we use both simulated data and real data sets. We first introduce these materials and experimental environment. Here, we use the score function to evaluate the correctness of VirDelect + OPA.

1) Real data set and environment: The real dataset comprises HBV full genomes of Clone_N66 and Clone_H44 (KJ790200) extracted from chronic hepatitis B patients and sequenced using a direct Sanger sequencer. Clone_N66 and Clone_H44 (KJ790200)[29] were genotype B HBV strains. The reference is the full-length HBV genome sequences of AB602818 obtained from the GenBank database. The data is read using an Illumina HiSeq 2000 platform producing pair-end reads. The length of the read was 101 bp, for which 1,053,570 reads were obtained for H44 and 1,124,427 were obtained for N66. We use the HP Z620 Workstation, including CPU: Intel Xeon E5-2620, RAM: 56GB, OS: Windows 7, JAVA JDK1.8. As shown in Table 1, RD1–RD2: represent, respectively, real read data from patients H44 and N66.

2) Simulated data set: We use SAMtools wgsim[30] to generate the NGS simulation data, including four different mutation rates and three different HBV and SARS-Cov-2 deletion positions. The source of the HBV simulation data mix two H44 sequences, one without a deletion and another with a deletion. The length of the read was 100bp, and the number of reads totaled 900,000. The reference comprised a full-length HBV genome sequence of AB602818, with a length of 3215bp. SARS-Cov-2 is the complete genome of NC_045512.2 downloaded from GenBanK, for which the length was 29903bp. In the SARS-Cov-2 data, the length of read was 100bp, and the number of reads totaled 300,000. As shown in Table 1, SD1~SD3 are simulated data and contain the deletion breakpoint and the mutation rate.

Table 1. The simulated and real datasets in this work.

Dataset	Breakpoint of Deletion	Rate of mutation(%)
SD1	1610~1672	2, 5, 9, 16
SD2	1540~1560, 1610~1672	2, 5, 9, 16
SD3	424~494, 1145~1195	2, 5, 9
RD1	3210~3215~57	--
RD2	3132~3215~8	--

SD1–SD3: simulated read data by SAMtools wgsim. RD1–RD2: represent respectively real read data form patient H44 and N66.

B. Scoring function

As in previous research [15], we use formulas such as (5) to evaluate the correctness of the detected breakpoints. The scoring function uses a more rigorous scoring method to evaluate the ability of the program to detect the correct breakpoint. The W value was set at 5 based on a previous study. SP and EP represent the start and end positions found by the program. GSP and GEP represent the start and end positions of the ground truth.

$$Score = 100 - (|SP - GSP| + |EP - GEP|)^2 \times w \qquad (5)$$

C. Experimental results of VirDelect+OAP

1) Fig. 4 and Fig. 5 show the experimental results for the simulated data and real data, respectively. In the simulation data, shown in Fig. 1, we use Bowtie2 and SAMtools to take out the unmapped SAM read. Then we compare the correctness of the two methods obtained using VirDelect+OAP and VirDelect by recording the position of the exact detected breakpoint. The control ground truth was used to calculate the respective scores using the scoring function, where 100 points indicated where the VirDelect detected the same breakpoint position as the ground truth. As long as the missed positions were greater than a distance of 5 positions, the score was considered to be zero.

2) Fig. 4A-D presents the scores for the simulated dataset for the four diversities, 2%, 5%, 9%, 16%, respectively. Fig. 4A shows that program detected a difference of 2% in SD1, where SD2 and SD3 and SD2 and SD3 were two different deletion positions. The score for SD3 was zero, which means that the difference between this position and the correct position was more than 5bp. There was no SD3 score shown in Fig. 4D, and we did not analyze the SARS-Cov-2 simulation data with a diversity of 16% in this study. Fig. 4A-C shows that in different diversities, the scores for the VirDelect+OAP detection simulation data were higher than those for VirDelect. Fig. 5 shows the experimental results for the real data set. The VirDelect+OAP scores in the two different real data sets are also higher than the VirDelect scores.

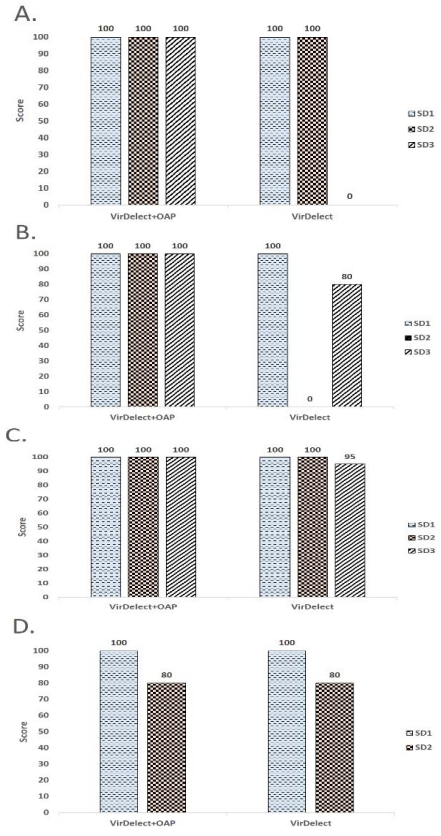

Fig. 4. Comparison of VirDelect+OAP and VirDelect with four different diversities, SD1, SD2 and SD3, in simulated datasets. (A)(B)(C)(D) represents, 2%, 5%, 9% and 16%, four different

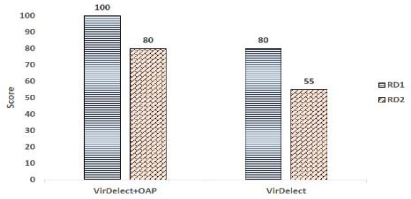

Fig. 5 Comparison of VirDelect+OAP and VirDelect with real dataset.

V. DISCUSSION AND CONCLUSION

This research was focused on proposing a new alignment algorithm intended to improve VirDelect's degree of correctness for virus sequence deletion detection. In order to achieve this goal, we proposed the One-base Alignment Plus (OAP) algorithm. We conducted experiments with both simulated data and real data. The experimental results are discussed below, and the conclusions drawn from this research are provided.

In SD1, we added one fragment deletion, and VirDelect's correctness was good for sequences with only one fragment deletion, as in our previous research[15]. We know from related work that the virus sequence is not only highly variable, but it may also have more than two deletions. Therefore, in this study, we added data with two fragment deletions, SD2 and SD3. In the results, we found that VirDelect is unstable for the exact breakpoint detection with two fragment deletions. For SD2, VirDelect can find the exact breakpoint of deletion in the 9% diversity, as shown in Fig. 4C, but it could not find the deletion in the 5% diversity, as shown in Fig. 4B. For SD3, VirDelect was able to find deletion at 9% diversity, as shown in Fig. 4C, but cannot find deletion at 2% diversity, as shown in Fig. 4A. In theory, a greater degree of diversity in the simulation data leads to a lower probability of finding the correct deletion. In the simulation data with less than 9% diversity, VirDelect+OAP was very stable and could accurately detect deletion breakpoints, while in the simulation data with 16% diversity, only some points were lost. In the real data results, VirDelect+OAP also obtained higher correctness.

In this study, we have proposed an effective and efficient method, called *One-base Alignment*, to improve VirDelect correctness. The main contributions included: (1) Providing an OAP algorithm that improves the correctness of VirDelect related to detecting exact deletion breakpoints in a virus sequence; (2) the experimental results show that VirDelect+OAP has high stability in detecting deletions in highly variable virus sequences; (3) VirDelect+OAP can detect the deletion points for two different fragments in a sequence among highly variable virus sequences.

Acknowledgement

This research was partially supported by Ministry of Science and Technology Taiwan under grant no. MOST 109-2224-E-009-003.

REFERENCES

[1] S.Cleemput *et al.*, "Genome Detective Coronavirus Typing Tool for rapid identification and characterization of novel coronavirus genomes," *Bioinformatics*, 2020, doi: 10.1093/bioinformatics/btaa145.

[2] T.Phan, "Genetic diversity and evolution of SARS-CoV-2," *Infect. Genet. Evol.*, 2020, doi: 10.1016/j.meegid.2020.104260.

[3] Y. C. F.Su *et al.*, "Discovery and genomic characterization of a 382-nucleotide deletion in ORF7B and orf8 during the early evolution of SARS-CoV-2," *MBio*, 2020, doi: 10.1128/mBio.01610-20.

[4] M. R.Islam *et al.*, "Genome-wide analysis of SARS-CoV-2 virus strains circulating worldwide implicates heterogeneity," *Sci. Rep.*, vol. 10, no. 1, pp. 1–9, 2020, doi: 10.1038/s41598-020-70812-6.

[5] W. C.Liu *et al.*, "Hepatocellular carcinoma-associated single-nucleotide variants and deletions identified by the use of genome-wide high-throughput analysis of hepatitis B virus," *J. Pathol.*, 2017, doi: 10.1002/path.4938.

[6] A.Jary *et al.*, "Evolution of viral quasispecies during SARS-CoV-2 infection," *Clin. Microbiol. Infect.*, no. xxxx, 2020, doi: 10.1016/j.cmi.2020.07.032.

[7] W.Zhang, J.Chen, Y.Yang, Y.Tang, J.Shang, andB.Shen, "A practical comparison of De Novo genome assembly software tools for next-generation sequencing technologies," *PLoS One*, 2011, doi: 10.1371/journal.pone.0017915.

[8] D.Yorukoglu, Y. W.Yu, J.Peng, andB.Berger, "Compressive mapping for next-generation sequencing," *Nature Biotechnology*. 2016, doi: 10.1038/nbt.3511.

[9] K.Chen *et al.*, "BreakDancer: An algorithm for high-resolution mapping of genomic structural variation," *Nat. Methods*, 2009, doi: 10.1038/nmeth.1363.

[10] J. O.Korbel *et al.*, "PEMer: A computational framework with simulation-based error models for inferring genomic structural variants from massive paired-end sequencing data," *Genome Biol.*, 2009, doi: 10.1186/gb-2009-10-2-r23.

[11] R.Sun *et al.*, "Breakpointer: Using local mapping artifacts to support sequence breakpoint discovery from single-end reads," *Bioinformatics*, 2012, doi: 10.1093/bioinformatics/bts064.

[12] C.Bartenhagen andM.Dugas, "Robust and exact structural variation detection with paired-end and soft-clipped alignments: SoftSV compared with eight algorithms," *Brief. Bioinform.*, 2016, doi: 10.1093/bib/bbv028.

[13] K.Ye, M. H.Schulz, Q.Long, R.Apweiler, andZ.Ning, "Pindel: A pattern growth approach to detect break points of large deletions and medium sized insertions from paired-end short reads," *Bioinformatics*, 2009, doi: 10.1093/bioinformatics/btp394.

[14] N. F.Lahens *et al.*, "A comparison of Illumina and Ion Torrent sequencing platforms in the context of differential gene expression," *BMC Genomics*, 2017, doi: 10.1186/s12864-017-4011-0.

[15] J. H.Cheng, W. C.Liu, T. T.Chang, S. Y.Hsieh, andV. S.Tseng, "Detecting exact breakpoints of deletions with diversity in hepatitis B viral genomic DNA from next-generation sequencing data," *Methods*, 2017, doi: 10.1016/j.ymeth.2017.08.005.

[16] M.Pachetti *et al.*, "Emerging SARS-CoV-2 mutation hot spots include a novel RNA-dependent-RNA polymerase variant," *J. Transl. Med.*, vol. 18, no. 1, pp. 1–9, 2020, doi: 10.1186/s12967-020-02344-6.

[17] W. C.Liu *et al.*, "Simultaneous quantification and genotyping of hepatitis B virus for genotypes A to G by real-time PCR and two-step melting curve analysis," *J. Clin. Microbiol.*, 2006, doi: 10.1128/JCM.01375-06.

[18] A.Kramvis, K.Arakawa, M. C.Yu, R.Nogueira, D. O.Stram, andM. C.Kew, "Relationship of serological subtype, basic core promoter and precore mutations to genotypes/subgenotypes of hepatitis B virus.," *J. Med. Virol.*, vol. 80, no. 1, pp. 27–46, Jan.2008, doi: 10.1002/jmv.21049.

[19] M.Sunbul, "Hepatitis B virus genotypes: global distribution and clinical importance.," *World J. Gastroenterol.*, vol. 20, no. 18, pp. 5427–5434, May2014, doi: 10.3748/wjg.v20.i18.5427.

[20] C.Seeger andW. S.Mason, "Molecular biology of hepatitis B virus infection," *Virology*. 2015, doi: 10.1016/j.virol.2015.02.031.

[21] D.Zhang *et al.*, "Whole genome HBV deletion profiles and the accumulation of preS deletion mutant during antiviral treatment.," *BMC Microbiol.*, 2012, doi: 10.1186/1471-2180-12-307.

[22] X.Li *et al.*, "PreS deletion profiles of hepatitis B virus (HBV) are associated with clinical presentations of chronic HBV infection," *J. Clin. Virol.*, 2016, doi: 10.1016/j.jcv.2016.06.018.

[23] I. C.Wu, W. C.Liu, andT. T.Chang, "Applications of next-generation sequencing analysis for the detection of hepatocellular carcinoma-associated hepatitis B virus mutations," *Journal of Biomedical Science*. 2018, doi: 10.1186/s12929-018-0442-4.

[24] S. F.Ahmed, A. A.Quadeer, andM. R.McKay, "Preliminary identification of potential vaccine targets for the COVID-19 Coronavirus (SARS-CoV-2) Based on SARS-CoV Immunological Studies," *Viruses*, 2020, doi: 10.3390/v12030254.

[25] T.Phan, "Novel coronavirus: From discovery to clinical diagnostics," *Infection, Genetics and Evolution*. 2020, doi: 10.1016/j.meegid.2020.104211.

[26] A. C.Walls, Y. J.Park, M. A.Tortorici, A.Wall, A. T.McGuire, andD.Veesler, "Structure, Function, and Antigenicity of the SARS-CoV-2 Spike Glycoprotein," *Cell*, 2020, doi: 10.1016/j.cell.2020.02.058.

[27] B.Langmead, C.Trapnell, M.Pop, andS. L.Salzberg, "Ultrafast and memory-efficient alignment of short DNA sequences to the human genome," *Genome Biol.*, 2009, doi: 10.1186/gb-2009-10-3-r25.

[28] H.Li andR.Durbin, "Fast and accurate long-read alignment with Burrows-Wheeler transform," *Bioinformatics*, 2010, doi: 10.1093/bioinformatics/btp698.

[29] W. C.Liu *et al.*, "Aligning to the sample-specific reference sequence to optimize the accuracy of next-generation sequencing analysis for hepatitis B virus," *Hepatol. Int.*, 2016, doi: 10.1007/s12072-015-9645-x.

[30] H.Li *et al.*, "The Sequence Alignment/Map format and SAMtools," *Bioinformatics*, 2009, doi: 10.1093/bioinformatics/btp352.