VACmap: an accurate long-read aligner for unraveling complex structural variations

Hongyu Ding[1] , Zhirui Liao[1] and Shanfeng Zhu[1,*]

[1]Institute of Science and Technology for Brain-Inspired Intelligence and MOE Frontiers Center for Brain Science, Fudan University, Shanghai 200433, China.

22110850043@m.fudan.edu.cn, zhusf@fudan.edu.cn

Abstract

Recent advancements in whole-genome sequencing have unveiled the remarkable diversity and abundance of complex structural variations, surpassing previous perceptions. Unfortunately, these complex variations often remain understudied and underappreciated in genetic research due to inherent technical challenges. Existing mapping methods often failed to align long reads containing complex structural variations, as they tend to align the long reads by maximizing a colinear score function between the long reads and the reference sequence. However, in the presence of structural variations, the long read can only represent in a non-linear manner and consequently colinear mapping is deemed to result in incorrect and incomplete alignments. To address this critical issue, we have developed VACmap—an accurate mapping method specialized in mapping long reads containing complex structural variations. VACmap incorporates a novel variant-aware chaining algorithm, which effectively identifies the globally optimal non-linear alignment for each long read. This algorithmic approach ensures the faithful representation of both simple and complex structural variations, thus producing correct alignments. Our experimental findings confirm that VACmap significantly enhances the performance of downstream simple and complex structural variation detection, allowing for reliable and precise use of the signals in long reads. Notably, VACmap greatly improves the discovery rate of duplication in downstream detection which existing methods often misalign them as insertion. Additionally, VACmap exhibits remarkable robustness in accurately identifying structural variations within repetitive genomic regions. These regions, which have traditionally posed challenges for mapping techniques, are effectively addressed by VACmap.VACmap overcomes a longstanding hurdles in aligning long-read with structural variations and we believe it will be widely embraced in various downstream analysis.

Introduction

Structural variation is a major source of genome variation and change more base pairs than single nucleotide polymorphisms (SNPs) [1-4]. Complex structural variation is a special form of structural variation and defined as gene rearrangement caused by single mutational events involving multiple breakpoints [5]. These breakpoints can be categorized as simple structural variation events such as deletion, insertion, inversion, duplication, or translocation [6]. Studies utilizing whole-genome sequencing have indicated that approximately 2-16% of structural variations are complex [7-10], but due to mapping and detection challenges, the actual proportion is likely higher [5]. The type of complex structural variation is determined by the combination of simple structural variation events, these simple structural variation events often occur closely (<1kb)[10] and adjacent simple structural variation events can be separated by normal sequence or closely connected [11-14]. However, accurately mapping these clustered events remains a challenge for existing long-read mapping algorithms [15].

While long-read sequencing holds promise for detecting structural variations [16-19], accurate detection relies on high-quality alignments produced by long-read mapping algorithms. Existing mapping algorithms align sequences by identifying co-linear matches between long reads and the reference genome [15, 16, 20]. However, if a long read contains structural variations, it cannot be linearly aligned to the reference genome in its entirety. In such cases, current mapping methods attempt to partially map long reads to maximize the alignable region. This results in multiple colinear alignments, potentially conflicting with each other (e.g., overlapping colinear alignments in the same regions of the long read), requiring subsequent alignment selection to filter out irrelevant colinear alignments. However, finding the optimal colinear alignment set is challenging, as it may not be within the candidate alignment set. Existing long-read mapping algorithm ms often yield incorrect alignments, particularly in the presence of balanced structural variations, small duplications and complex structural variations.

To address these limitations, we propose VACmap, a specialized long-read mapping method designed specifically for complex structural variations. VACmap employs a novel Variant-Aware Chaining algorithm to compute the single globally optimal non-linear alignment for long reads. This algorithm accurately represents both simple and complex structural variations while ensuring the output collinear alignment set is optimal and unique. When utilizing the alignment data generated by VACmap for downstream variation analysis tasks, it significantly improves the performance of existing structural variation detection algorithms for complex structural variations.

We conducted empirical analyses to evaluate the performance of VACmap using simulated and real long-read datasets, comparing it with existing long-read mapping algorithms. The alignment results obtained with VACmap demonstrated significantly superior performance in downstream structural variation detection tasks. VACmap exhibited excellent and consistent mapping performance for complex structural variations of varying complexity, providing reliable support for complex structural variation within repetitive regions. Moreover, compared to other long-read mapping algorithms, incorporating the VACmap approach for downstream complex structural variation detection tasks effectively improved detection accuracy by ~30% to ~60% in F1 score. And for simple structural variation detection, VACmap approach preserved ~30% more duplication information that is often misaligned as an insertion by existing mapping algorithms. VACmap shows promising results in mapping

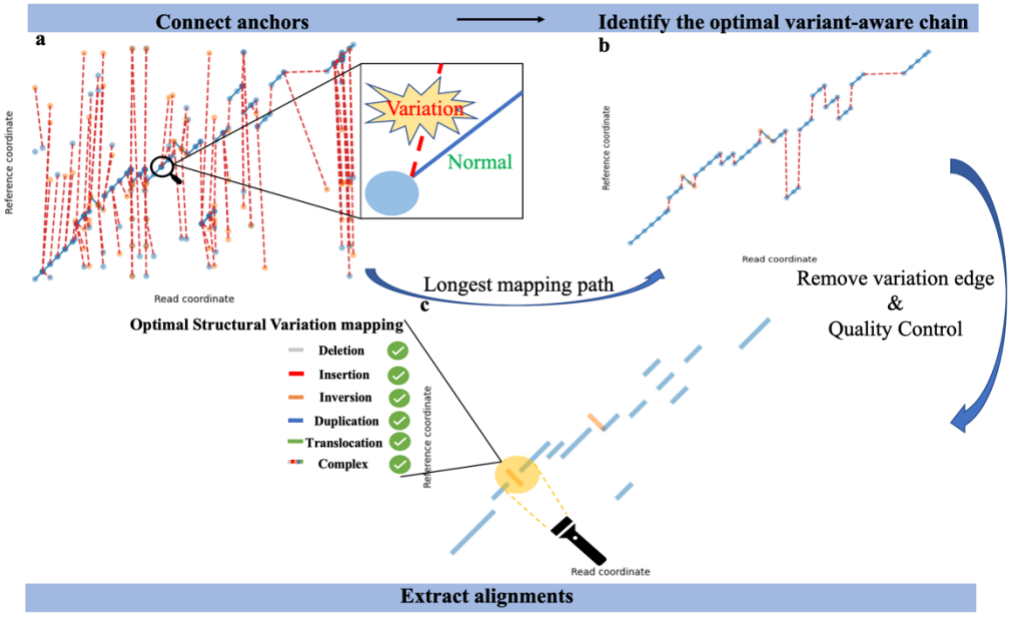

Figure 1 illustrates the workflow of VACmap. a. In the first step, VACmap connects anchors and assigns weights. When a pair of anchors maintains collinearity, VACmap assigns a positive weight to the edge connecting them, referred to as a normal edge. Conversely, when a pair of anchors does not maintain collinearity, VACmap assigns a negative weight to the connecting edge, known as a variation edge. The blue or yellow dot represent the anchor that are mapped to the forward or reverse strand, respectively. **b.** Next, VACmap identifies the optimal variant-aware chaining by searching for the longest path. **c.** Finally, VACmap removes the variation edges within the optimal variant-aware chain and extracts collinear alignments. The blue or yellow lines represent the alignments that are mapped to the forward or reverse strand, respectively. These output collinear alignments are optimized and facilitate sensitive downstream structural variation discovery.

tasks and enables more accurate detection of complex structural variations in downstream analyses. It represents a valuable contribution to the field of genomics research.

Result

An overview of the VACmap algorithm.

When dealing with a long read sequence that doesn't have any mutations, determining its position on the reference genome is straightforward, and the best alignment can be found by searching for the longest collinear matches between the reference sequence and the long read sequence. However, if there are structural variations within the long read, linearly mapping it to the reference genome becomes impossible. To tackle this problem, a commonly used method involves partially mapping the long read and minimizing the unmapped region. Nevertheless, this approach can generate a set of colinear alignments that may conflict with each other, necessitating subsequent alignment selection to identify the optimal set of collinear alignments. Studies have

demonstrated that a greedy strategy for alignment selection can handle complex structural variations in non-repetitive regions with moderate structural variations complexity. NGMLR[16] employs dynamic programming to optimize alignment selection, improving the mapping of complex structural variations. However, our experiments indicate that optimizing alignment selection alone does not ensure accurate mapping of complex structural variations, as the colinear alignments produced by existing long-read mapping algorithms can be incorrect.

VACmap adopts the classical seed-chain-align process for aligning long read sequences (Figure 1). Initially, we utilize the implementation in minimap2 [20] to collect the minimizers[21] of the reference sequence and build an index of the reference genome. For each long read sequence, we collect its minimizers as seeds and query their positions in the reference genome. If a minimizer in the long read finds a match in the reference genome, we represent the match with a quadruple **(*x*, *y*, strand, *k*)** and refer to it as an anchor. Here, *x* and *y* indicate the starting positions of the current anchor in the long read sequence and the reference sequence, respectively, while the strand denotes the matching strand (1 for forward strand and -1 for reverse strand). *k* indicates the length of the anchor. Once all anchors have been collected, we sort them in ascending order based on their starting positions in the long read sequence and utilize the variant-aware chaining algorithm to compute the globally optimal alignment path for the long read sequence.

The variant-aware chaining algorithm encourages the chain to extend to anchors that maintain collinearity with the last anchor in the chain and assigning a positive weight to the edge connecting these two anchors (**Methods**). Conversely, the algorithm penalizes the chain from extending to anchors that do not maintain collinearity with the last anchor and assigning a negative weight to the edge connecting them (**Methods**). By searching for the nonlinear chain with the highest score (longest path), VACmap can obtain the globally optimal nonlinear alignment for long read sequences with arbitrary structural variations. Finding the nonlinear chain with the highest score for a set of N anchors can be achieved using dynamic programming in *O(N^2)* time. However, since we are only interested in the nonlinear chain with the highest score, we can expedite the computation by testing only the chains with score higher than a specific threshold (**Methods**). This heuristic approach often yields the optimal nonlinear chain and significantly reduces the time complexity to *O(hN),* where *h* is determined by the average number of chains exceeding the specific threshold.

Sequencing errors frequently occur in clusters, and using a larger *k*-mer size and a minimizer-based *k*-mer sampling method can make it challenging to obtain sufficient matching information in specific regions of the long read. To address this issue, VACmap constructs a local index and collects anchors for the second round of nonlinear chaining by directly using smaller *k*-mers. However, employing smaller *k*-mers substantially increases the number of anchors, which in turn raises the computation time for finding the optimal nonlinear chain. To mitigate the computational cost, VACmap filters the anchors using the previously computed optimal nonlinear chain. Any anchor that deviates significantly from the previously computed optimal nonlinear chain (**Methods)** is discarded. Additionally, VACmap conducts pairwise analysis on the remaining anchors (**Methods**). If there is no variation between two overlapping anchors, VACmap merges them as one to further reduce the number of anchors.

After obtaining the optimal nonlinear chain, VACmap divides it into a set of colinear subchain and calculates the pairwise alignment for each colinear subchain. However, due to sequencing errors, the variant-aware algorithm may produce incorrect colinear alignments. VACmap filters out these incorrect colinear alignments by carefully

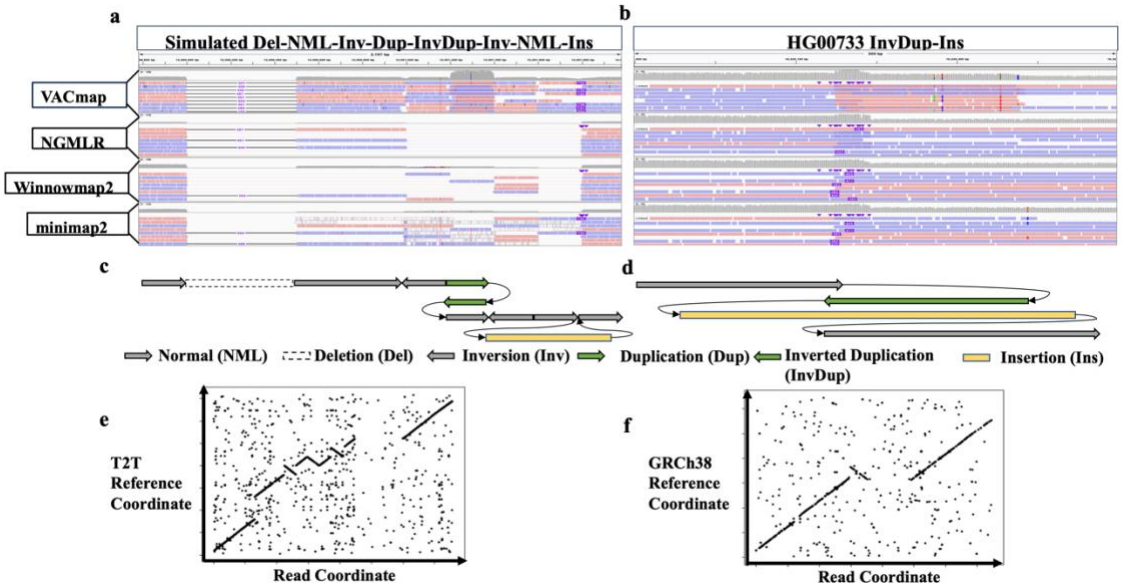

Figure 2 The visualization of a simulated complex structural variation and a complex structural variation found in the HG00733 human sample. a, b). These figures present the IGV visualization of the alignment pileup image in the mutation region. The grey track illustrates the mapping coverage of each alignment produced by the mapping algorithms. The red or purple line segments indicate alignments mapped to the forward or reverse strand, respectively. VACmap accurately aligns the mutation segment and achieves the expected read coverage for both balanced and imbalanced simple structural variation events. **c, d).** The schematics of the simulated and real complex structural variations, respectively. **e, f).** The dot plot of a randomly selected long read in the simulated and real complex structural variation regions. Please note that Figures a, c, and e correspond to the simulated complex structural variation, while Figures b, d, and f correspond to the real complex structural variation.

checking each alignment in the output colinear alignment set **(Method**). Finally, VACmap outputs those colinear alignments in the compressed BAM format, facilitating downstream analysis.

Visualization of aligned complex structural variation

The latest released T2T reference genome provides a complete blueprint of the human genome, filling the gaps in repetitive genomic regions [22]. These regions are prone to mutations and are challenging to align accurately. To validate VACmap's mapping performance on complex structural variations in repetitive regions, we followed previous research [15] and manually introduced a complex structural variation in the repetitive region of the T2T reference genome. This complex structural variation consists of events such as deletions, inversions, tandem duplications, inverted tandem duplications, and insertions (Supplementary Table 1). Then, using SURVIVOR [23], we introduced SNPs into the modified reference genome sequence and generated simulated Oxford Nanopore Technologies (ONT) long-read data with 40-fold read depth. VACmap, Winnowmap2 [15], NGMLR [16], and minimap2 [20] were used as alignment tools to align the simulated ONT long-read data to the T2T reference genome. The alignment results generated by different tools were visually analyzed using the

Integrative Genomics Viewer (IGV) [24]. The results (Figure 2a) show that VACmap accurately aligns most of the long reads covering this region and clearly displays the simple structural variation events in the complex structural variation. The other three long-read mapping algorithms exhibit significant drops in read depth within the variant region, especially in the Inversion-Duplication-Inverted Duplication-Inversion region, which indicates the other three mapping algorithms fail to align long reads with these complex SVs.

VACmap is specifically designed for complex structural variation mapping. Its accurate and complete alignments can also benefit SNP detection. Based on the IGV visualization results, VACmap effectively generated a comprehensive set of alignments that encompassed both the normal and variation segments, successfully capturing the simulated SNPs within the variation segment. On the contrary, other three mapping algorithms failed to map variation segment and also made it impossible to recover SNPs within.

Additionally, we investigated a reported [25] complex structural variation in the HG00733 human sample. This variation is located in chromosome 2 16255100-16255300 genetic region and comprised of two distinct events: an inverted duplication and an insertion. To assess the alignment accuracy of different tools (VACmap, NGMLR, Winnowmap2, and minimap2) in this region, we visually analyzed the alignment results using IGV (Figure 2b). Remarkably, only VACmap successfully aligned the variant segment associated with the inverted duplication. In contrast, the remaining three long-read alignment tools failed to identify this specific variation event and mistakenly classified it as an insertion event.

Evaluation of downstream complex and simple structural variation detection on simulated data

To evaluate the impact of VACmap on downstream variant detection, we conducted a series of experiments. Initially, we generated two set of simulated long-read data containing complex structural variations and simple structural variations respectively. Multiple alignment tools were then employed to generate alignment files, which were subsequently used with the SVIM [26] to detect structural variations.

Due to the lack of simulation tools specifically designed for complex structural variations, we developed VACsim to simulate both simple and complex structural variations. VACsim allows users to specify or randomly generate the type of complex structural variations. Due to the unpredictable nature of complex structural variation, the new form of complex structural variation is continually discovered. So, we randomly introduced different forms and complexities of structural variations on the T2T reference genome using VACsim. Total 30,000 structural variations were simulated and each simulated structural variations comprised a random number of simple structural variations events from range 1 to 20, where the type of each simple structural variations event was randomly chosen from deletions, insertions, duplications, inversions and translocations. The size of each simple structural variations event was randomly sampled from the range of 100 to 1000 base pairs. Similar to many model-based structural variation callers [27], SVIM has the capability to detect structural variations that match specific patterns. In this case, a simple structural variation event must be surrounded by normal sequence. To ensure simulated complex structural variations can be effective detected by SVIM, we separated these adjacent simple structural variations events with a normal sequence of 200 base pairs in length, except for adjacent duplication

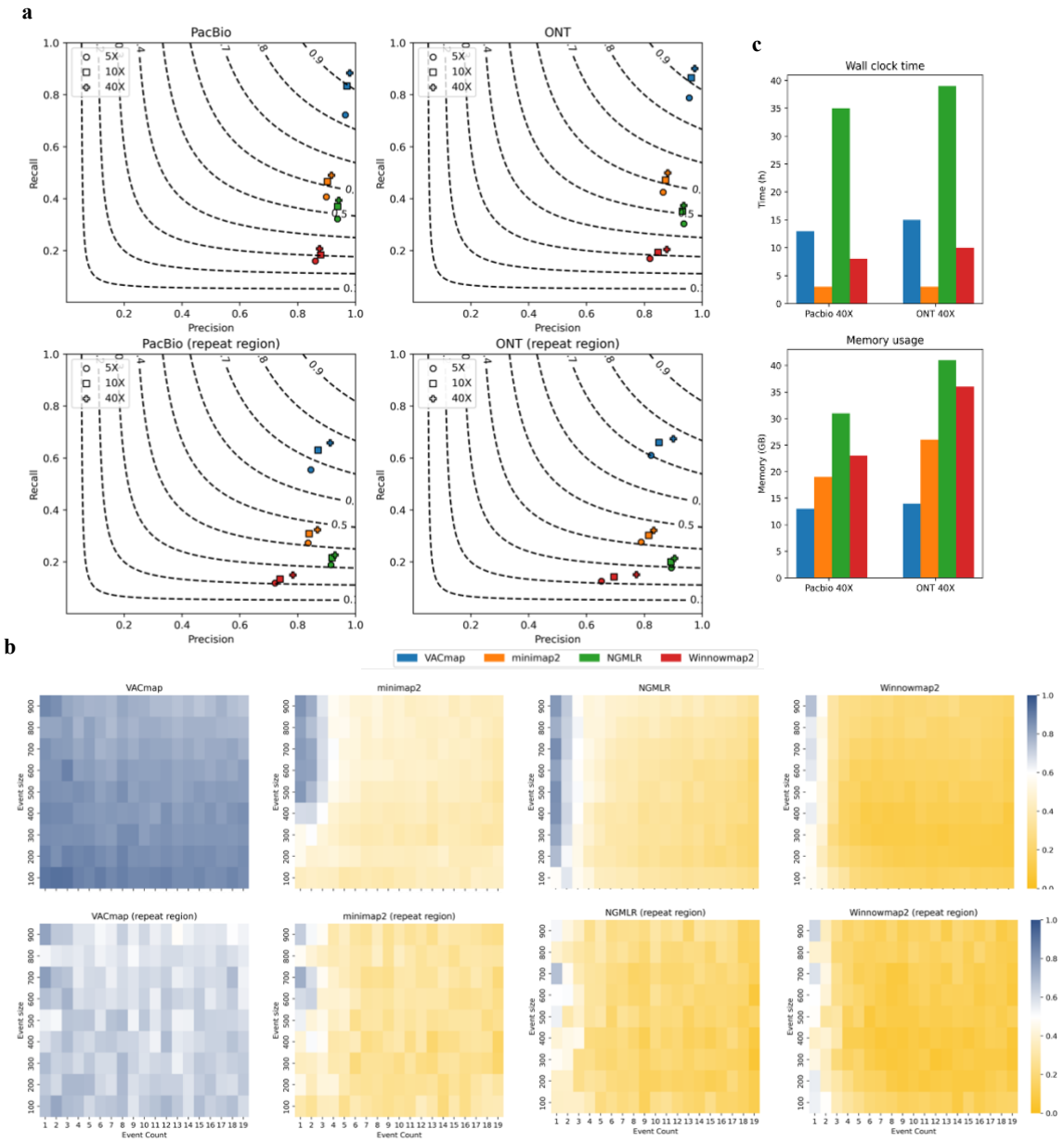

Figure 3 A comparison of four mapping algorithms in terms of downstream complex structural variation detection and computational requirements. a. The precision, recall, and F1 scores for all T2T chromosomes and specifically the repetitive region, considering different read depth and sequence technologies. **b.** The recall heat map displays the performance of the four mapping algorithms on 40-fold Pacbio simulated data across varying structural variation complexity and simple structural variation event length. **c.** The runtime and peak memory usage of the four mapping algorithms on 40-fold Pacbio and ONT simulated data. Notably, VACmap demonstrates improved sensitivity and specificity in downstream complex structural variation detection, with a comparable runtime and lower memory usage.

events (SVIM support detection of multiple repeat duplication). Then, SURVIVOR[23] was used to introduce SNPs, and simulated Pacbio and ONT long-read data with 5-fold, 10-fold, and 40-fold coverage. Different

alignment algorithms were employed to align the simulated long-read, and SVIM was used to detect structural

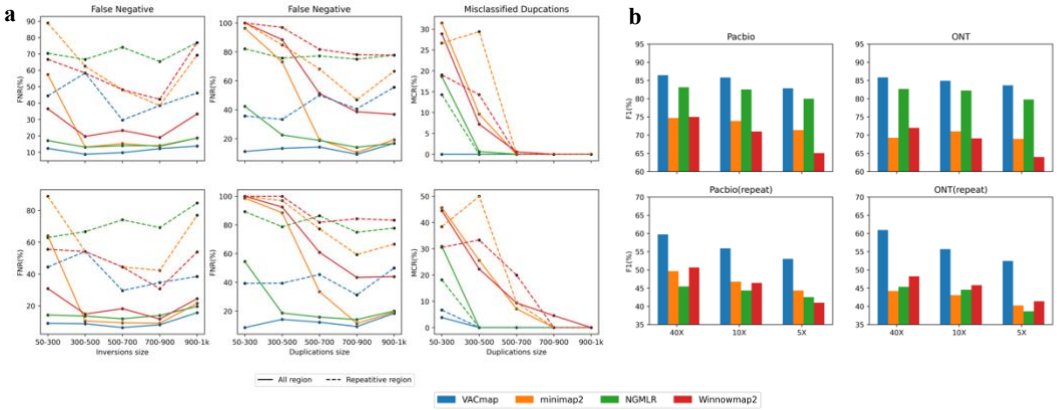

Figure 4 The comparison of four mapping algorithms in terms of downstream simple structural variation detection. a. The detection performance of different mapping algorithms for inversions (INVs) and duplications (DUPs) of various sizes at a 5-fold read depth. **b.** The performance of different mapping algorithms in simple structural variation detection at read depths of 5-fold, 10-fold, and 40-fold. The VACmap-SVIM pipeline achieved the highest F1 score across different datasets and consistently demonstrated reliable detection performance for inversions and duplications of different sizes.

variations from the alignment results. To evaluate SVIM's performance in detecting structural variations with different alignments in repetitive regions. We followed a previously established workflow [15] to identify repetitive regions greater than 10 kbp in length and with sequence similarity above 95% using Mashmap [28]. We then analyzed the complex structural variations mapping performance of different mapping tools on these repetitive regions.

As shown in Figure 3, the alignments output from VACmap significantly improved SVIM's performance in detecting complex structural variations on both PacBio CLR and ONT simulated long-read data across different read depths. In contrast, the F1 score were ~30 to ~60 precent lower when using NGMLR, Winnowmap2, and minimap2 alignment results for complex structural variations detection (Supplementary Table 2). Additionally, for the detection of complex structural variations in repetitive regions, the approach utilizing VACmap exhibited a significant advantage in both precision and recall rates, and outperformed other methods by ~20 to ~50 percent in F1 score (Supplementary Table 2). Figure 3b illustrates a heatmap displaying the recall rates for downstream structural variation detection tasks on PacBio CLR simulated data (see Supplementary Figure 1 for the results on ONT simulated data). The heatmap represents various combinations of event sizes and complexities of structural variations, employing the four mapping algorithms. NGMLR, minimap2 and Winnowmap2 are only tolerate for complex structural variations consisting of two simple structural variations events. For higher complexities, the mapping performance of NGMLR, minimap2, and Winnowmap2 significantly decreased. In contrast, VACmap's produced alignments enable consistent and sensitive detection of structural variations across varying levels of complexity, highlighting its robustness in accurately mapping complex structural variations.

Furthermore, we assessed the performance impact of different mapping algorithms on downstream simple structural variations detection. We simulated five types of simple structural variations (deletion, insertion, inversion, duplication, and translocation) on Chromosome 1 and Chromosome 2 of the T2T reference genome using SURVIVOR and generated PacBio and ONT simulated long read with read depths of 5-fold, 10-fold, and

40-fold. SVIM was then employed to detect structural variations from the alignments generated by different mapping algorithms. The simulated sequencing data consisted of 1000 insertions and deletions (ranging from 100 to 1000 base pairs), 1000 inversions (ranging from 100 to 1000 base pairs), 1000 duplications (ranging from 100 to 1000 base pairs), and 200 translocations (ranging from 100 to 1000 base pairs).

The results (Figure 4) indicate that when using the VACmap-SVIM pipeline for structural variation detection, VACmap approach outperformed the other three methods by ~2 to ~20 percent in F1 scores across different read depth and the most repetitive region. Furthermore, VACmap exhibited consistent performance in different inversions and duplications size. On the contrary, the false negative rates (FNR) of NGMLR, Winnowmap2, and minimap2 were increased for smaller inversion and duplication (<300bp). Then we accessed the proportion that misclassifies duplications as insertion using different mapping methods. The VACmap approach demonstrates a significantly lower misclassification rate for smaller duplications (<300bp), indicating its higher resolution in aligning duplication segments.

Evaluation of aligned mutations-free long read on simulated data

In addition, we conducted a comprehensive evaluation of four mapping algorithms to assess the accuracy and completeness of mapping mutations-free long reads. To create the simulated datasets, we extracted the chromosome 1 and chromosome 2 sequences from T2T, GRCh38, and GRCh37 reference genomes. Subsequently, SURVIVOR was employed to generate mutations-free simulated PacBio and ONT long read datasets. Each simulated long read dataset was aligned against its corresponding reference genome sequence using four different mapping algorithms.

To determine the correctness of a long-read mapping, we employed a similar evaluation method as described in a previous study [20]. Specifically, a long read was considered correctly mapped if its primary alignment corresponded to the true interval, and the overlap length was equal to or greater than 10% of the true interval's length. The results, presented in Supplementary Table 4, demonstrate that VACmap exhibited comparable performance in terms of mapping accuracy and completeness to NGMLR, Winnowmap2, and minimap2 for mutations-free long read mapping. These findings highlight the versatility of VACmap, which was originally designed for mapping long reads containing structural variations, but also performs well in general use cases.

Evaluation on genome in a bottle benchmark (HG002 cell line).

We further analyzed the extensively studied HG002 cell line, three publicly available datasets from different sequencing technologies were used, namely PacBio CLR, PacBio CCS, and ONT. The four mapping algorithms, namely VACmap, NGMLR, Winnowmap2, and minimap2, were used to align long reads to the GRCh37 reference genome. The SVIM structural variation detection tool was employed to detect structural variations in the alignments generated by different mapping algorithms. Truvari (https://github.com/spiralgenetics/truvari) was used to assess the precision, recall, and F1 score of the call sets produced by SVIM. The benchmark set for HG002

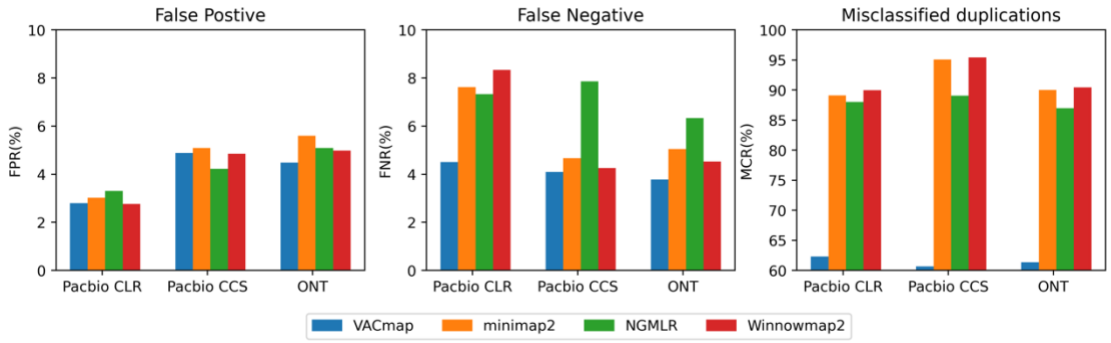

Figure 5 The GIAB benchmark results of four structural variation detection pipelines. a. The false positive rate of SVIM call sets obtained using four different mapping algorithms. **b.** The false negative rate of SVIM call sets under the four mapping algorithms. **c.** The misclassified rate of duplications as insertions under the four mapping algorithms. Notably, VACmap demonstrates improved sensitivity and much lower misclassification rates in both Pacbio and ONT datasets.

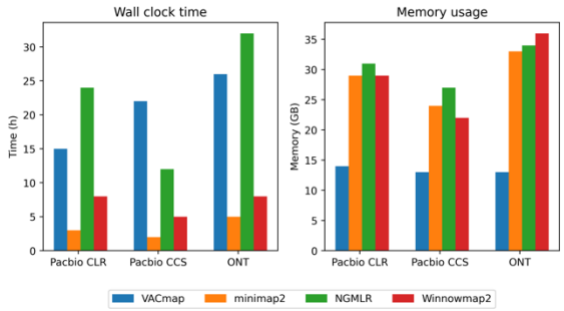

Figure 6 Wall clock and memory usage of four mapping method. VACmap shows comparable run time and lower memory usage comparing to state of art mapping methods.

human sample was obtained from Genome in a Bottle Consortium (GIAB) [29]. The existing GIAB benchmark set (v0.6) solely facilitates benchmarking insertion and deletion calls. Nevertheless, it is important to highlight that the insertion calls within the GIAB call set predominantly consist of duplications [29]. To accurately evaluate the performance of various structural variation pipelines, we conducted a re-annotation of the SVIM output, specifically focusing on duplications, which were reclassified as insertions. Subsequently, performance testing was carried out to assess the effectiveness of these pipelines. For reference, we have also included the results without the re-annotation in Supplementary Tables 5. The obtained results, as depicted in Figure 5a and 5b, along with Supplementary Tables 6, clearly illustrate that VACmap outperformed other methods in terms of recall rates and F1 score across PacBio CLR, CCS, and ONT data.

Long-read mapping algorithms commonly exhibit limitations in accurately aligning duplication sequences, often misaligning them as insertions during the alignment process. To thoroughly assess the mapping performance of diverse mapping algorithms specific to duplications, we implemented the REPTYPE annotation to identify and segregate duplication calls within the GIAB benchmark set. Subsequently, we meticulously compared the insertion calls detected by SVIM against the duplication calls in the GIAB benchmark set. By quantifying the proportion of duplications erroneously identified as insertions by SVIM, our objective was to ascertain the mapping efficacy of distinct mapping algorithms in handling duplication sequences. The outcomes, prominently illustrated in Figure 5c, unequivocally highlight the VACmap approach as surpassing the performance of the other three mapping

algorithms by revealing approximately 40% more duplications. Conversely, when employing the remaining three methods for structural variation analysis, an alarming 90% of duplications were inaccurately aligned as insertion sequences, impeding the identification of downstream structural variations.

Furthermore, we conducted runtime and peak memory usage tests on VACmap, minimap2, Winnowmap2, and NGMLR using 40 CPU threads on the three aforementioned datasets. The results (Figure 6) show that VACmap has similar runtime and lower peak memory usage compared to the other mapping algorithms.

Method

Variant-aware chaining algorithm

Existing long-read mapping algorithms[15, 16, 20] align long reads by identifying the highest-scoring colinear matches between a reference sequence and the long read, utilizing specific scoring functions. This approach is generally valid since the majority of genome regions do not exhibit structural variation. However, when a long read contains structural variations, these variations disrupt the collinearity between the reference and the long read, rendering it impossible to achieve alignment solely through colinear matches. To address this issue, we present the variant-aware chaining algorithm. This novel chaining algorithm offers native support for representing non-colinearity between long reads and the reference and avoiding the extra errorful alignment selection procedure, enabling accurate mapping for structural variations of any complexity.

In detail, We collected exact match k-mers between a reference sequence and a long-read sequence. These k-mers are represented as quadruples *(x, y, s, k)*, which we refer to as anchors. In this context, *x* and *y* indicate the starting positions of the current anchor in the long-read sequence and the reference sequence, respectively. The variable *s* denotes the matching strand (1 for the forward strand and -1 for the reverse strand), while *k* represents the length of the anchor. In our variant-aware chaining algorithm, each anchor is treated as a node in a directed weighted graph. We enforce certain constraints on the connections between nodes. Specifically, nodes with lower positions in the long read can only be connected to nodes with higher positions, and the regions covered by the two nodes in the long read sequence cannot overlap. To calculate the weight of the edge between *node[i]* and *node[j] (i<j)*, denoted as *w(i, j)*, we utilize the following formulas:

$$w(i,j) = \begin{cases} k_j + CP & if\ (diff < max(maxdiff - c_j, 10), 0 \leq s_j \times refgap < maxgap, s_i = s_j) \\ k_j + VP + c_j & else \end{cases} \quad (1)$$

$$CP = 0.01 \times k_j * diff + 0.5 log_2(\max(diff,\ 1)) \quad (2)$$

$$diff = |readgap - refgap| \quad (3)$$

$$readgap = x_j - x_i - k_i \quad (4)$$

$$refgap = \begin{cases} y_j - y_i - k_i, \text{if } s_\text{i} = s_\text{j} = 1 \\ y_j - y_i + k_j, \text{if } s_\text{i} = s_\text{j} = -1 \end{cases} \quad (5)$$

If two nodes satisfy collinearity, which is determined by the conditions specified in the if statement in Formula 1, we use the upper part of Formula 1 to calculate the weight. This type of edge is referred to as a "normal" edge. The collinear condition allows a small gap between two nodes, we employ the affine-convex gap penalty model (Formula 2) taken from minimap2[20] to add an extra penalty, which we refer to as the *CP (colinear penalty)*.

However, if two nodes do not satisfy collinearity, we utilize the lower part of Formula 1 to calculate the weight. This type of edge is referred to as a "variation" edge, indicating a potential variation. The high occurrence kmer is usually less informative[16], we assign an extra penalty *c[i]* when the chain tries to extend to a node that belongs to a high occurrence kmer and applies a stricter colinear condition to determine whether satisfy the collinearity. Where *c[i]* is the number of nodes that share the same start position in a long read. To aid in understanding these concepts, we provide several visual examples in Supplementary Figure 2, demonstrating when to assign a normal edge or a variation edge.

The quality of the non-linear alignment path is significantly influenced by the selection of hyperparameter values, with *maxdiff* and *maxgap* emerging as two pivotal hyperparameters to consider. A smaller *maxdiff* value tends to engender a more linear collinear subchain, whereas a smaller *maxgap* value aids in averting the erroneous alignment of unrelated sequences. Another vital hyperparameter is the *VP (variation penalty)*, which serves dual purposes during chain extension. Firstly, it encourages cautious chain extension by judiciously considering new variant edges. The variant-aware chaining algorithm endeavors to identify the longest path (highest-scoring chain) within a given set of anchors. To prioritize the discovery of the highest-scoring chain with only the most indispensable variation edge, we assign a negative weight to variation edges. Secondly, the variation penalty determines the minimum length of collinear subchains. For collinear subchains located at the extremities of a variant-aware chain, the minimum allowable length exceeds the value of the variation penalty. However, if the subchain is positioned within the middle of the variant-aware chain, the minimum allowable length is greater than twice the value of the variation penalty. It is noteworthy that longer collinear subchains may be unexpectedly discarded due to sequence errors in the variant-aware chaining algorithm. Moreover, the selection of the variation penalty hyperparameter directly impacts the performance of downstream tasks, such as structural variation detection, exerting an influence on both sensitivity and specificity (refer to Supplementary Figure 3).

To find the optimal non-linear alignment path among *N* nodes, dynamic programming can be employed, but this approach requires $O(N^2)$ time. However, when dealing with long-read sequence data that often generates a large number of nodes, executing VACmap using dynamic programming becomes time-consuming. To address this issue and expedite the runtime, we adopt a heuristic approach that approximates the calculation of the highest score non-linear alignment path. For a given *node[j]*, we can determine its maximum score *S(i)* and identify the best predecessor *node[i]* by utilizing the following formulas:

$$S(i) = max\left(max_{i \in M}\left(S(i) + w(i,j)\right), k_j\right) \qquad i \in M \mid s(i) > \max score - 2 * VP \tag{6}$$

The set M comprises nodes whose score surpass a dynamic update threshold, this threshold depends on presently highest score and the predefined *VP*.

Once the maximum score of all nodes is computed, we can identify the highest scores non-linear alignment path by backtracking. Unlike existing mapping methods[15, 16, 20] need to identify all possible colinear chains (colinear alignments) to make sure the entire long read is mapped, as the single colinear chain can't represent the entire long read containing structural variation. In VACmap, the highest score chain can well represent the nonlinearity between the reference sequence and long read. So we only need to identify the highest score chain and guarantee the entire long read is covered by the highest score chain. Finally, if the highest score chain contains variation edges, we discard all variation edges in the highest score chain and divide the highest score chain into multiple colinear subchains (colinear alignments). The approach avoids the selection of the alignments, a strategy widely used in existing mapping methods. The existing mapping methods often produce both correct and incorrect alignments when mapping long reads containing structural variation and need subsequence analysis to identify

only the correct alignments. However, this strategy often produces suboptimal results in aligning the long read containing complex structural variations, especially for complex structural variations with high complexity and within repetitive regions[5]. Furtherly, existing mapping methods are less sensitive to inversion, as the downstream and upstream genetic regions of the inversion site are remaining collinear[16, 20]. For duplication, existing mapping methods often misalign the duplication as an insertion in the colinear chain extension. The difficulty in aligning these structural variation segments will produce the incorrect alignments, and make the selection of the alignments even more challenging, as the correct alignments may not exist.

Local index and anchor reduction

When sequence errors occur in clusters, accurate mapping becomes challenging, especially when utilizing a large k-mer size setting and a minimizer k-mer sampling strategy. To mitigate this issue, VACmap adopts a similar approach used in a previous study [30] by constructing a local index with a smaller k-mer size, typically a 9-mer. The local index is built by collecting all possible k-mers from the previously computed highest-scoring chain's covered reference genome region. Subsequently, all the k-mers in the long read are employed to query the index, obtaining matching information known as anchors. The highest scoring chain is then recomputed based on these anchors.

However, the runtime of the variant-aware chaining algorithm depends on the number of anchors. Using a smaller k-mer size setting often results in an increased number of anchors, consequently escalating computational time. To address this issue, two strategies are employed to reduce the number of anchors and improve runtime efficiency. Firstly, VACmap iterates through the anchors and removes an anchor if the distance between the anchor and the previously computed highest-scoring chain exceeds a certain threshold (2000 bp). The distance between an anchor and the previous chain is defined as the distance between the anchor and its closest anchor in the chain under the reference sequence coordinate. Determining the closest anchor in the chain can be accomplished through a binary search, with a time complexity of $O(log^N)$, where N is the number of anchors. Secondly, VACmap merges two anchors (x_j, y_j, s_j, k_j) and (x_i, y_i, s_i, k_i) into a new anchor $(x_j, y_j, s_j, x_i + k_i - x_j)$ if they satisfy certain conditions. The sorting of N anchors in ascending order based on their position in the long read allows the merging process to be computed in $O(N)$ time using a hash table. The hash table utilizes an integer key and maintains a list of merged anchors as its value. For $anchor_i$, if s_i equals 1, then the key is set to $y_i - x_i$; otherwise, it is set to $-(y_i + x_i)$. VACmap tests the last $anchor_j$ in the list corresponding to a key. If the two anchors are overlapped or closely adjacent and the covered sequence in reference and long read is identical (formula 7), VACmap updates the last anchor in the list; otherwise, $anchor_i$ is appended to the list. If a key does not exist in the hash table, a new key-value pair is inserted. Finally, the merged anchors set can be obtained by traversing the hash table.

$$\begin{cases} y_j - x_j = y_i - x_i, x_j + k_j \geq x_i, s_j = s_i = 1, x_i > x_j \\ \quad or \\ y_j + x_j = y_i + x_i, x_j + k_j \geq x_i, s_j = s_i = -1, x_i > x_j \end{cases} \tag{7}$$

Quality control of colinear subchain

The variant-aware chaining algorithm has been specifically devised to facilitate the comprehensive alignment of long reads with the reference genome sequence. However, the presence of sequencing errors introduces the potential for erroneous alignment of unrelated sequences. To tackle this challenge, we have implemented a mechanism to evaluate the quality of each collinear subchain and remove those with low quality. To assess the accuracy of a collinear subchain, we employ the calculation of its error rate. This rate is determined by computing the edit distance [31] between the covered region of the collinear subchain in the reference sequence and the corresponding region in the long read sequence. The resulting edit distance is then normalized by dividing it by the minimum observed length between the covered regions of the reference sequence and the long read sequence.

In order to ensure alignment accuracy, VACmap incorporates a user-defined threshold (default: 0.2) for the error rate. If the error rate of a collinear subchain exceeds this defined threshold, it is considered unreliable and subsequently discarded from the alignment chain. Additionally, we offer the option to selectively discard collinear subchains that initiate with a deletion (insertion) and conclude with an insertion (deletion). However, it should be noted that this specific pattern may occasionally result in misplaced subchains or the unintended exclusion of genuine mutations.

Structural variation simulation and evaluation

To assess the performance of various mapping algorithms in downstream structural variant detection, we employed the T2T (v2.0) human reference genome sequences. Two altered genome sequences were generated, one containing simple structural variations and the other containing both simple and complex structural variations. For the altered sequence with both simple and complex variations, we utilized VACsim to randomly introduce 30,000 structural variations into the T2T reference. VACsim determined the number of simple structural variation events for each simulated structural variation, randomly selecting a value between 1 and 20 (1 for simulating simple structural variation). The types of simple structural variation events were determined by VACsim, sampling from basic structural variation types, namely deletion, insertion, duplication, inversion, and translocation events, with respective probabilities of 0.24, 0.24, 0.24, 0.24, and 0.04. The size of each type of simple structural variation event ranged from 100 to 1000 base pairs. Additionally, SVIM (v1.4.2) employed a model match strategy to detect structural variations, requiring each simple structural variation event to conform to the "Normal+Variation+Normal" pattern. Therefore, we preserved a 200 bp segment of normal sequence between each pair of adjacent simple structural variation events, except in the case of two adjacent simple duplication events of the same size, as SVIM supports the detection of multiple repeat duplications.

For the altered sequence containing simple structural variations, SURVIVOR (v1.0.6) was used to simulate 1000 insertions and deletions (ranging from 100 to 1000 base pairs), 1000 inversions (ranging from 100 to 1000 base pairs), 1000 duplications (ranging from 100 to 1000 base pairs), and 300 translocations (ranging from 100 to 1000 base pairs). These variations were randomly inserted into chromosomes 1 and 2 of the T2T reference genome sequence. Furthermore, SNPs were introduced into the two altered sequences using SURVIVOR and simulated Pacbio and ONT long-read datasets were generated.

The four long-read mapping algorithms, namely VACmap, NGMLR, Winnowmap2, and minimap2, were utilized to align the simulated long reads to the T2T reference genome sequences. The SVIM structural variant detection tool was employed to detect structural variations from the alignment files generated by the different mapping algorithms. Similar to previous studies [16], we evaluated the performance of complex structural variation detection by decomposing complex structural variations into individual simple structural variation events and evaluating each event separately.

Furthermore, following the procedure outlined in a previous study [15], we employed mashmap to identify repetitive regions in the reference genome sequence with sequence similarity exceeding 95% and a sequence length greater than 10,000 base pairs. Subsequently, we analyzed the performance of downstream structural variant detection within these repetitive regions using alignments produced by different mapping algorithms.

Evaluation with GIAB callset

We conducted an evaluation of four mapping algorithms, namely VACmap, NGMLR, Winnowmap2, and minimap2, using the GIAB Tier1 (v0.6) deletion and insertion benchmark set [29] for the HG002 human sample. The benchmark set is available relative to the GRCh37 human genome reference at the following URL: https://ftp-trace.ncbi.nlm.nih.gov/ReferenceSamples/giab/release/AshkenazimTrio/HG002_NA24385_son/NIST_SV_v0.6/. For this experiment, we utilized PacBio CLR, PacBio CCS, and ONT sequencing data, which were made accessible at s3://giab/data/AshkenazimTrio/HG002_NA24385_son/. To detect structural variations, we employed the SVIM (v1.4.2) structural variation detection tool, which generated call sets. Subsequently, we evaluated the SVIM call set against the GIAB benchmark set using Truvari (v2.0.0).

Data availability

This study utilized three human reference genome sequences: GRCh37, GRCh38, and T2T. They can be downloaded from the following links: http://ftp.1000genomes.ebi.ac.uk/vol1/ftp/technical/reference/phase2_reference_assembly_sequence/, https://ftp.1000genomes.ebi.ac.uk/vol1/ftp/technical/reference/GRCh38_reference_genome/, and https://github.com/marbl/CHM13.

For the Pacbio CLR, Pacbio CCS, and ONT sequencing data of the human sample HG002. They can be downloaded from the following links: s3://giab/data/AshkenazimTrio/HG002_NA24385_son/PacBio_MtSinai_NIST/PacBio_fasta/, s3://giab/data/AshkenazimTrio/HG002_NA24385_son/PacBio_CCS_15kb/, and s3://giab/data/AshkenazimTrio/HG002_NA24385_son/UCSC_Ultralong_OxfordNanopore_Promethion/.

The Pacbio CLR alignment data for the human sample HG00733 can be downloaded from the following link: https://ftp.1000genomes.ebi.ac.uk/vol1/ftp/data_collections/hgsv_sv_discovery/working/20160905_smithm_pacbio_aligns/. The raw reads can be extracted from the alignment files using samtools [32].

Discussion and Conclusions

VACmap is a specialized long-read mapping algorithm designed specifically for mapping long-read containing complex structural variations. Through extensive validation, we have demonstrated the exceptional performance of VACmap in downstream analysis tasks involving complex structural variations. In contrast, existing mapping algorithms struggle to provide reliable and complete alignments for long reads containing complex structural variations, even with optimized alignment selection strategies. This limitation also makes it impossible to carry out an effective downstream structural variation detection, existing structural variation detection methods often rely on modeling the relative position of alignments to determine the type of structural variation.

Currently, large-scale genomic analysis projects face difficulties in aligning and detecting complex structural variations, which has hindered focused analysis of these variations. However, we believe that VACmap can overcome these challenges by providing optimized sets of alignments for complex structural variation detection tasks. Previous study[5] suggested once accurate alignment sets are obtained, the detection of complex structural variations becomes a semantic problem as we analyze the relationships among alignments to identify the specific underlying simple structural variation events.

Notably, with the release of complete human genome sequences, we now have a more comprehensive view of the most repetitive and challenging-to-sequence regions, which often harbor structural variations. VACmap exhibits excellent mapping performance for complex structural variations, even in these repetitive regions of varying complexities. This capability enables us to gain a better understanding of structural variations in these challenging genomic areas. By leveraging VACmap, we anticipate significant advancements in our understanding of complex structural variations, fostering deeper research and exploration in both academic and clinical domains.

Acknowledgment

We would like to thank Dr. Mingfu Shao (The Pennsylvania State University) for the helpful suggestions and insightful comments on the manuscript.

Reference

[1] Sudmant PH, Rausch T, Gardner EJ, Handsaker RE, Abyzov A, Huddleston J, et al. An integrated map of structural variation in 2,504 human genomes. Nature. 2015;526(7571):75–81.

[2] Weischenfeldt J, Symmons O, Spitz F, Korbel JO. Phenotypic impact of genomic structural variation: insights from and for human disease. Nat Rev Genet. 2013;14:125–138. doi: 10.1038/nrg3373.

[3] Conrad DF, Bird C, Blackburne B, Lindsay S, Mamanova L, Lee C, et al. Mutation spectrum revealed by breakpoint sequencing of human germline CNVs. Nat Genet. 2010;42(5):385–91.

[4] Feuk, L., Carson, A. R., and Scherer, S. W. (2006). Structural variation in the human genome. Nat. Rev. Genet. 7, 85–97. doi: 10.1038/nrg1767.

[5] Quinlan AR, Hall IM. Characterizing complex structural variation in germline and somatic genomes. Trends Genet. 2012 Jan;28(1):43-53. doi: 10.1016/j.tig.2011.10.002. Epub 2011 Nov 15. PMID: 22094265; PMCID: PMC3249479.

[6] Hadi K, Yao X, Behr JM, et al. Distinct Classes of Complex Structural Variation Uncovered across Thousands of Cancer Genome Graphs. Cell. 2020 Oct 1;183(1):197-210.e32. doi: 10.1016/j.cell.2020.08.006. PMID: 33007263; PMCID: PMC7912537.

[7] Collins RL, Brand H, Redin CE, Hanscom C, Antolik C, Stone MR, et al. Defining the diverse spectrum of inversions, complex structural variation, and chromothripsis in the morbid human genome. Genome Biol. 2017;18(1):36.

[8] Conrad DF, et al. Mutation spectrum revealed by breakpoint sequencing of human germline CNVs. Nat Genet. 2010;42:385–391.

[9] Kidd JM, et al. A human genome structural variation sequencing resource reveals insights into mutational mechanisms. Cell. 2010;143:837–847.

[10] Quinlan AR, et al. Genome-wide mapping and assembly of structural variant breakpoints in the mouse genome. Genome Res. 2010;20:623–635.

[11] Perry GH, et al. The fine-scale and complex architecture of human copy-number variation. Am J Hum Genet. 2008;82:685–695.

[12] Zhang F, et al. The DNA replication FoSTeS/MMBIR mechanism can generate genomic, genic and exonic complex rearrangements in humans. Nat Genet. 2009;41:849–853.

[13] Carvalho CM, et al. Inverted genomic segments and complex triplication rearrangements are mediated by inverted repeats in the human genome. Nat Genet. 2011;43:1074–1081. doi: 10.1038/ng.944.

[14] Shimojima K, et al. Pelizaeus-Merzbacher disease caused by a duplication-inverted triplication-duplication in chromosomal segments including the PLP1 region. Eur J Med Genet. 2012;55:400–403. doi: 10.1016/j.ejmg.2012.02.013.

[15] Jain, C., Rhie, A., Hansen, N.F. et al. Long-read mapping to repetitive reference sequences using Winnowmap2. Nat Methods 19, 705–710 (2022). https://doi.org/10.1038/s41592-022-01457-8.

[16] Sedlazeck FJ, Rescheneder P, Smolka M, Fang H, Nattestad M, von Haeseler A, Schatz MC. Accurate detection of complex structural variations using single-molecule sequencing. Nat Methods. 2018;15:461–8.

[17] English AC, Salerno WJ, Reid JG. PBHoney: identifying genomic variants via long-read discordance and interrupted mapping. BMC Bioinformatics. 2014;15:180.

[18] Chaisson MJ, Tesler G. Mapping single molecule sequencing reads using basic local alignment with successive refinement (BLASR): application and theory. BMC Bioinformatics. 2012;13:238.

[19] Chaisson MJ, Huddleston J, Dennis MY, Sudmant PH, Malig M, Hormozdiari F, Antonacci F, Surti U, Sandstrom R, Boitano M, et al. Resolving the complexity of the human genome using single-molecule sequencing. Nature. 2015;517:608–11.

[20] Heng Li, Minimap2: pairwise alignment for nucleotide sequences, Bioinformatics, Volume 34, Issue 18, September 2018, Pages 3094–3100, https://doi.org/10.1093/bioinformatics/bty191.

[21] Roberts M. et al. (2004) Reducing storage requirements for biological sequence comparison. Bioinformatics, 20, 3363–3369.

[22] Sergey Nurk et al. ,The complete sequence of a human genome.Science376, 44-53(2022).

[23] Jeffares, D., Jolly, C., Hoti, M. et al. Transient structural variations have strong effects on quantitative traits and reproductive isolation in fission yeast. Nat Commun 8, 14061 (2017). https://doi.org/10.1038/ncomms14061

[24] Robinson JT, Thorvaldsdóttir H, Winckler W, Guttman M, Lander ES, Getz G, Mesirov JP. Integrative genomics viewer. Nat Biotechnol. 2011 Jan;29(1):24-6. doi: 10.1038/nbt.1754. PMID: 21221095; PMCID: PMC3346182.

[25] Lin J, Wang S, Audano PA, Meng D, Flores JI, Kosters W, Yang X, Jia P, Marschall T, Beck CR, Ye K. SVision: a deep learning approach to resolve complex structural variants. Nat Methods. 2022 Oct;19(10):1230-1233.

[26] David Heller, Martin Vingron, SVIM: structural variant identification using mapped long reads, Bioinformatics, Volume 35, Issue 17, 1 September 2019, Pages 2907–2915.

[27] Lin J, Yang X, Kosters W, Xu T, Jia Y, Wang S, Zhu Q, Ryan M, Guo L, Zhang C; Human Genome Structural Variation Consortium; Lee C, Devine SE, Eichler EE, Ye K. Mako: A Graph-based Pattern Growth Approach to Detect Complex Structural Variants. Genomics Proteomics Bioinformatics. 2022 Feb;20(1):205-218. doi: 10.1016/j.gpb.2021.03.007. Epub 2021 Jul 3.

[28] Jain, C., Koren, S., Dilthey, A., Phillippy, A. M. & Aluru, S. A fast adaptive algorithm for computing whole-genome homology maps. Bioinformatics 34, i748–i756 (2018).

[29] Zook, J. M. et al. A robust benchmark for detection of germline large deletions and insertions. Nat. Biotechnol. 38, 1347–1355 (2020).

[30] Ren J, Chaisson MJ. lra: A long read aligner for sequences and contigs. PLOS Comput Biol. 2021;17(6):e1009078.

[31] Šošic M, Šikic M. Edlib: a C/C ++ library for fast, exact sequence alignment using edit distance. Bioinformatics. 2017 May 1;33(9):1394-1395. doi: 10.1093/bioinformatics/btw753. PMID: 28453688; PMCID: PMC5408825.

[32] Heng Li and others, The Sequence Alignment/Map format and SAMtools, Bioinformatics, Volume 25, Issue 16, August 2009, Pages 2078–2079, https://doi.org/10.1093/bioinformatics/btp352