VACmap: An Accurate Long-Read Aligner for Unraveling Complex Genomic Rearrangements

Hongyu Ding[1], Fritz J Sedlazeck[2,3,4], Zhirui Liao[1], Lianrong Pu[5] 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.
[2]Human Genome Sequencing Center, Baylor College of Medicine, Houston, TX, USA
[3]Department of Molecular and Human Genetics, Baylor College of Medicine, TX, USA
[4]Department of Computer Science, Rice University, 6100 Main Street, Houston, TX, USA
[5]College of Computer Science and Technology, Shandong University, Qingdao, China
22110850043@m.fudan.edu.cn, zhusf@fudan.edu.cn

Abstract

Inversions, duplications, and other critical medically challenging variations are often ignored by routine genetic analyses. The repetitive nature and complexity make them hard to align accurately by state-of-the-art sequence alignment methods. We show the limitation stems from the fundamental approach of the current sequence alignment methodology, which attempts to align homologous sequences solely through linear edits (deletions, insertions, and substitutions). Here, we introduce VACmap, an innovative mapping method that integrates both linear and non-linear edits to directly align complex rearrangements without splitting the read and assembling linear subalignments. VACmap has demonstrated its ability to precisely identify complex inversions, even amidst large segmental duplications. It also accurately determines the breakpoints of the KIV-2 repeat in the LPA gene, which has been notoriously difficult to align using conventional methods. VACmap's novel non-linear mapping approach represents a paradigm shift in sequence alignment methodology and provides a comprehensive view of the intricate patterns of genomic variation.

Introduction

Third-generation sequencing technologies are at the forefront of structural variant (SV) discovery because they generate longer reads that can span entire SVs [1]. Traditional sequence alignment algorithms assume that homologous sequences can be aligned in a collinear fashion [2-11]. They use linear edits—deletions, insertions, and substitutions—to form alignments that reflect the evolutionary relationships between related sequences. However, their ability to accurately represent genomic rearrangements is limited. Complex genomic structures such as duplications, inversions, translocations, and nested events often elude straightforward linear alignment [12]. To overcome these limitations, a common strategy involves a post-processing phase, which involves identifying all possible linear subalignments in the read and reconstructing the complex genomic structure by selecting the most appropriate linear subalignments from a pool of possibilities. However, since these subalignments are aligned independently, choosing among them does not ensure the combined alignment is globally optimal for the entire sequence. Additionally, these linear subalignments are prone to error, mistaking duplications for insertions or wrongly characterizing inversions as block substitutions, further complicating the alignment selection process.

To discern inversions from these imprecise linear alignments, heuristic methods search for regions with a high number of mismatches. They then take the reverse complement of these regions and realign them. If realigning the reverse complement results in fewer mismatches compared to the original alignment, this provides evidence that the region likely contains an inversion [5-7, 9]. Unfortunately, inversions are often more complex than simple sequence reversals, frequently occurring with additional variations like inverted duplications [13-15]. Our findings indicate that the simplistic application of these heuristic strategies can lead to suboptimal results. Further, tandem duplications present an identical profile to insertions when viewed through the lens of linear mapping techniques. In the Genome in a Bottle (GIAB) SV benchmark set, tandem duplications are frequently misclassified as insertions [16]. GIAB has put extra effort to difference tandem duplications from insertions by looking at how similar the insertion sequence is to neighboring regions. However, when it comes to complex SVs (CSVs), heuristic approaches tend to be inadequate. CSVs consist of several clustered SV events that are simple in isolation but occur in close proximity (usually within a span of less than 1000 base pairs (bp)) [17]. These events may be interspersed with normal sequences or may be tightly linked [18-21]. Ideally, the elaborate architecture of SVs would be best represented by a series of linear subalignments connected by non-linear junctions at the SV breakpoints. Unfortunately, current methods struggle to manage such complexity [9], especially when faced with highly intricate SVs or those situated within repetitive regions [17].

To address these challenges, we introduce VACmap. VACmap diverges from traditional alignment methodology that depend solely on linear edit operations (insertions, deletions, and substitutions) for sequence alignment. Our approach utilizes the innovative Variant-aware Chaining (VAC) algorithm, which adeptly integrates both linear and non-linear edits to deliver a single, globally optimal non-linear alignment for each long read. This method transcends the limitations imposed by the complexity or type of genomic rearrangement and eliminates the need for a subsequent, error-prone alignment selection process.

Extensive empirical analysis was conducted to assess the performance of VACmap, utilizing both simulated and real long-read datasets and comparing it against leading long-read aligners. The results demonstrate that VACmap significantly improves detection accuracy by approximately 30% to 70% in the F1 score for downstream CSV detection tasks. Importantly, VACmap has unrevealed about 70% more tandem duplication events that are frequently misclassified as insertions by conventional algorithms. Furthermore,

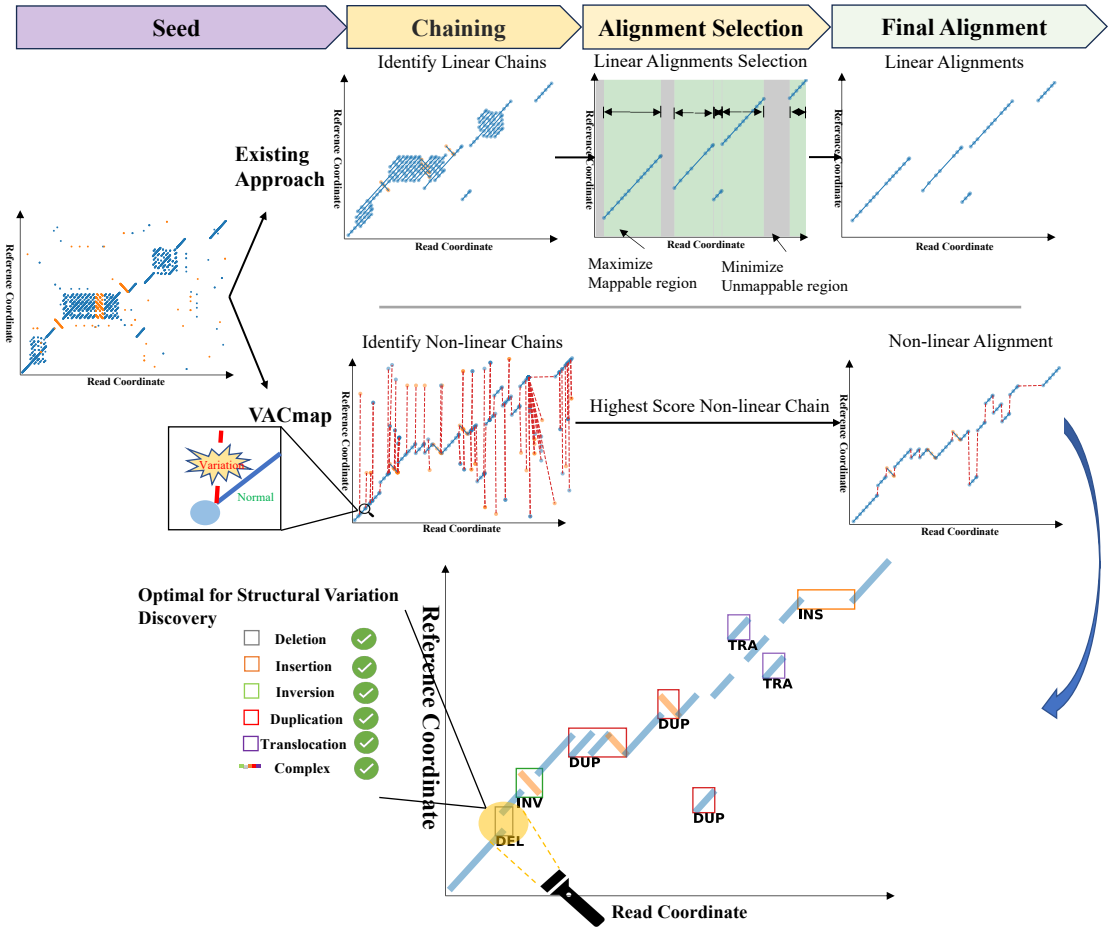

Figure 1 Comparison of traditional linear approach and VACmap non-linear approach on aligning a complex genomic rearrangement. This figure contrasts the workflows of the traditional approach and the VACmap approach for mapping complex rearrangements. Both start by matching seeds between the long read and reference (blue for the forward strand, orange for the reverse strand). Traditional methods then select appropriate collinear subalignments from a pool of candidates to reconstruct the rearrangement. Conversely, VACmap streamlines the process by directly representing the complex genomic rearrangement in a single non-linear alignment, eliminating the need for assembling linear subalignments.

VACmap's novel non-linear mapping strategy enables it to precisely map complex inversions accompanied by large segmental duplications, outperforming all other available methods. This is particularly evident in its enhanced characterization of critical medically challenging genes, such as the KIV-2 region in the LPA gene [22], a region where the copy number of 5.5kb segments is inversely correlated with the risk of cardiovascular disease and is especially challenging align to [23, 24]. With VACmap, we can more accurately pinpoint exact breakpoints within reads, thus providing a more reliable estimation of individual copy numbers per person. The promising outcomes of VACmap in mapping accuracy and its contribution to more precise SV detection in downstream analyses represent a significant advancement in the realm of genomics research.

Results

An Overview of the VACmap Algorithm.

VACmap is a non-linear mapping method, designed to leverage both linear and non-linear edits in the alignment process (Figure 1). First, VACmap adopts the implementation in minimap2 to construct the hash table of the reference genome with the minimizer k-mer sampling strategy [25]. For each long-read sequence, VACmap identifies minimizers that serve as seeds and searches for their corresponding positions in the reference genome. Matches between a minimizer from the long read and the reference genome are represented as quadruples called 'anchors,' which include the start positions in the long read and reference sequences, the strand match, and the anchor's length. After identifying all anchors, they are ordered by their start positions in the long read. The VAC algorithm then promotes the extension of the chain to subsequent anchors that preserve a strictly linear relationship with the preceding anchor, enhancing this connection with a positive score. Conversely, it penalizes extensions to anchors that disrupt this linearity by assigning negative scores to such connections. By selecting the chain with the highest aggregate score (the longest path), VACmap is able to determine the global optimal non-linear alignment for each long read, accommodating any genomic rearrangements. Identifying the highest-scoring non-linear chain for a set of N anchors can be achieved using dynamic programming in $O(hN)$ time. Where 'h' is determined by the average number of anchors whose scores surpass the specific threshold (Methods).

Complications arise when sequencing errors or single nucleotide polymorphisms (SNPs) are clustered, as these can hinder the effective collection of anchors in affected regions, especially when relying on larger k-mer sizes and a minimizer-based approach. Similar to the previous approach [8], VACmap creates a localized reference genome index and fetches a second set of anchors using smaller k-mers for an additional round of non-linear chaining. This dual-round approach does introduce a larger pool of anchors, which in turn, can increase the computational load. To manage this, VACmap utilizes a filtering mechanism based on the non-linear chain computed earlier, discarding any significantly divergent anchors. It also consolidates anchors that overlap to minimize the total number of anchors (Methods).

After establishing the optimal non-linear alignment, VACmap segments it at the points of non-collinearity into an array of linear subalignments. These subalignments are then packaged into the Sequence Alignment

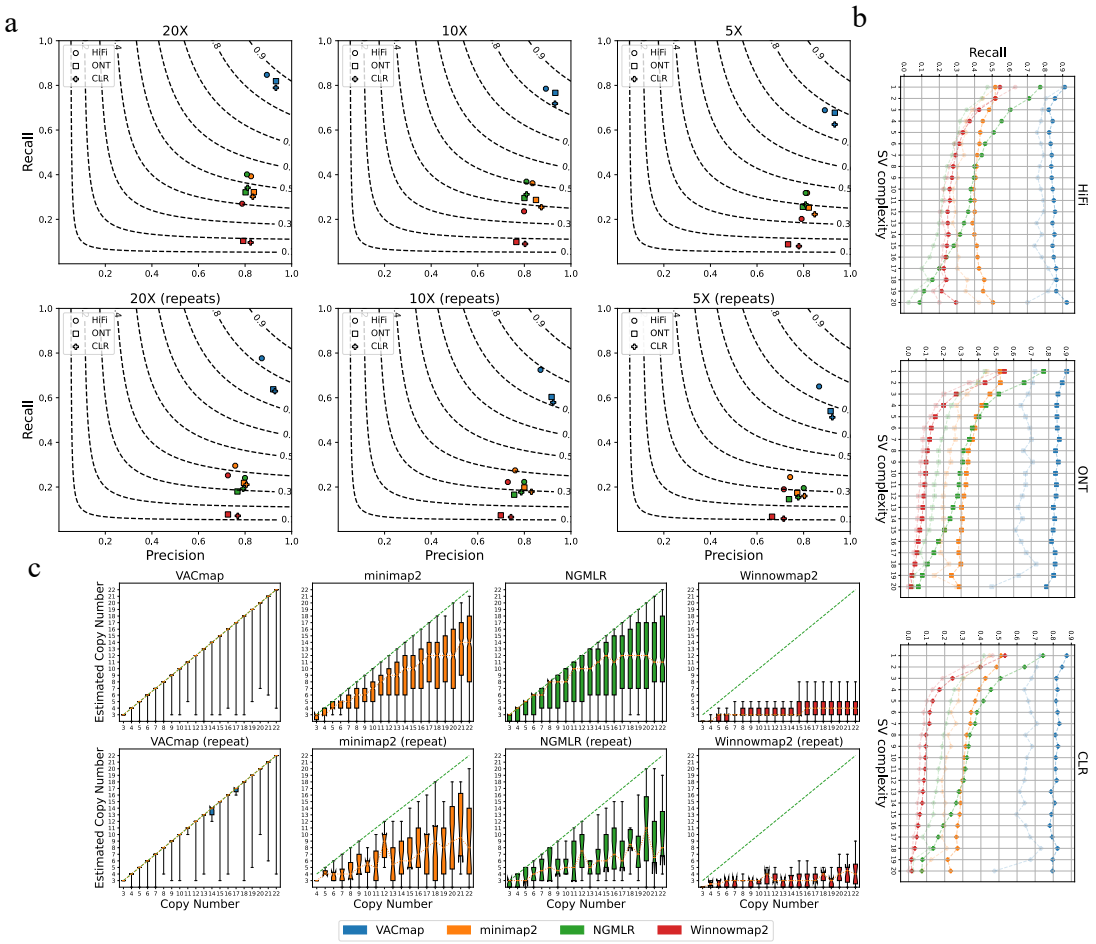

Figure 2 A comparison of four mapping algorithms in terms of downstream CSV detection. a. The precision, recall, and F1 scores of CSV detection performance in all T2T chromosomes and the repetitive region across different read depth and sequence technologies. **b.** The recall rates of CSV detection using alignments produced by four mapping methods under varying SV complexity and sequence technologies. The shaded color represents results in the repeat region. **c.** The box plots of the estimated tandem duplication copy numbers for each mapping method on ONT simulated data. Notably, VACmap demonstrates improved performance in downstream SV detection task across different SV complexity.

Map (SAM) format, effectively facilitating subsequent analytical procedures.

Assessing the Impact of VACmap on Detecting Complex Structural Variants in Synthetic Data

To evaluate the impact of VACmap on the detection of variants in downstream applications, a range of tests were performed on synthetic datasets. Initially, we produced synthetic long-read datasets laden with a spectrum of SVs, both simple and complex, using a custom tool we developed named VACsim, due to the lack of simulation tools for CSV. This tool introduced 30,000 SVs of varying complexity into the CHM13 T2T reference genome [26]. Each SV was constituted by multiple SV events, ranging from 1 to 20, chosen at random. The SV events were of five basic types—deletions, insertions, duplications, inversions, and translocations—with respective probabilities of 0.24, 0.24, 0.24, 0.24, and 0.04. The length of these SVs was randomly assigned between 100 and 1,000 base pairs. To facilitate detection by SVIM, two adjacent simple SV events are spaced

by a normal sequence of 200bp in size. For sequencing simulations, we utilized SURVIVOR [27] to place SNPs and generate artificial Pacific Biosciences (PacBio) and Oxford Nanopore Technologies (ONT) long-read data with coverage depths of 5, 10, and 20. The datasets were then aligned using VACmap, NGMLR, Winnowmap2, and minimap2, and the resulting alignment files were processed with SVIM [28] to identify SVs.

As illustrated in Figure 2, VACmap's alignment data markedly enhanced SVIM's ability to detect CSVs in simulated data from PacBio CLR, PacBio HiFi, and ONT, with F1 score improvements ranging between 31.9 to 68.9 percent (refer to Supplementary Table 1). Particularly for CSVs located within repetitive sequences, the use of VACmap-produced alignments provided significant gains in precision and recall, outperforming other methods by about 37.6 to 62 percent in F1 score (see Supplementary Table 1). Figure 2b shows the recall rates of SVs detection under different SV complexity and sequencing technology. NGMLR, minimap2, and Winnowmap2 were shown to be adequate only for identifying CSVs comprising up to two simple SV events. Beyond this complexity, the recall rates of SVIM plummeted when using alignments from these tools. Conversely, SVIM with VACmap alignments consistently displayed sensitive and reliable SV detection across the full spectrum of SV complexities.

For precise gene copy number quantification, particularly in tandem duplications that might influence protein levels, accurate mapping is essential. To investigate the performance of copy number estimations with alignments produced by different aligners, we generated 10,000 tandem duplications on chromosome 1 of the T2T genome using VACsim. These duplications had repeat unit sizes ranging from 100 to 500 base pairs and repeat counts between 1 to 20. SURVIVOR was then used to produce simulated PacBio and ONT datasets at 40-fold coverage. SVIM was employed to estimate the copy number for each tandem duplication from the various aligners' alignments. According to the outcomes depicted in Figure 2c, Supplementary Figure 1, and Supplementary Table 2, alignments from current mapping methods led to a notable bias in copy number estimation. There was a marked decline in the linear correlation between the actual and estimated copy numbers as the repeat count grew, especially within repetitive areas. In stark contrast, alignments from VACmap resulted in substantially more precise copy number estimates across diverse copy number intervals and within repetitive regions. This underscores VACmap's outstanding capability in accurately ascertaining the copy number of tandem duplications, emphasizing its effectiveness and accuracy in dealing with complex genomic structures.

Evaluation of the Genome in a Bottle Benchmark.

We further analyzed the extensively studied HG002 human sample, and three publicly available long-read datasets from different sequencing technologies, namely PacBio CLR, PacBio HiFi, and ONT. VACmap, NGMLR, Winnowmap2, and minimap2, were used to align long reads to the GRCh37 reference genome. SVIM was employed to detect SVs and Truvari [29] was used to assess the precision, recall, and F1 score of the call

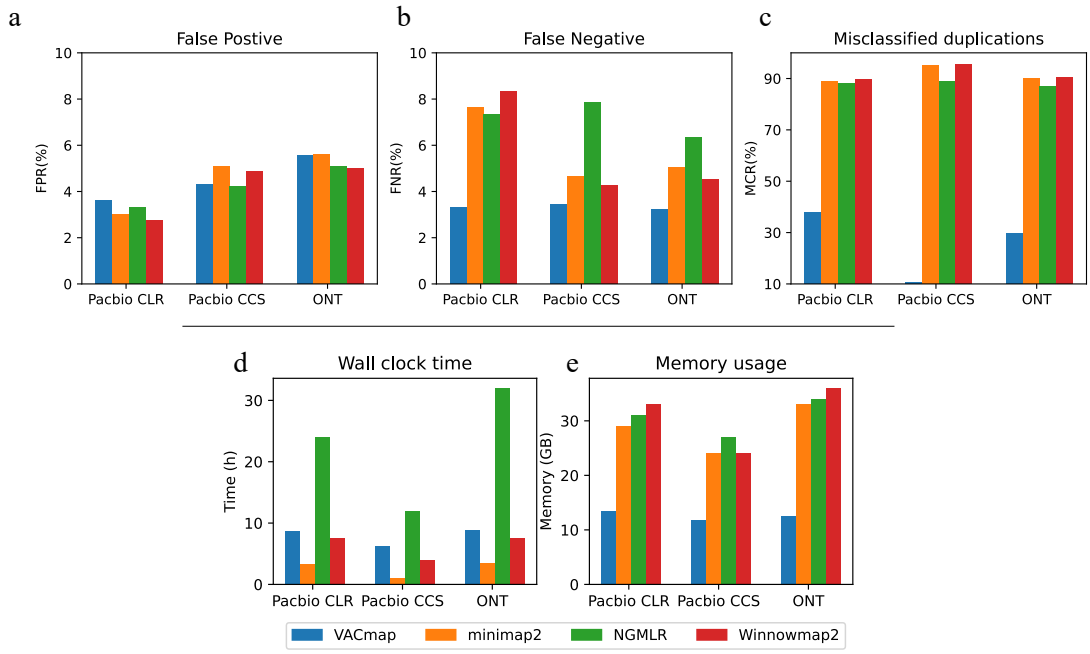

Figure 3 The GIAB benchmark results of four SV detection pipelines and computational requirements. a,b,c. The false positive rate (FPR), false negative rate (FNR), and misclassified rate (MCR) of SVIM call sets on alignments produced by four different mapping methods. **d,e.** The runtime and peak memory usage of the four mapping methods. Notably, VACmap demonstrates improved sensitivity and much lower misclassification rates in different datasets.

sets produced by SVIM. The SV benchmark set for the HG002 human sample was obtained from the GIAB [16]. However, we observed that many, insertion calls within of the GIAB call set are actually tandem duplications. We re-annotated the tandem duplication calls of SVIM as insertions. For reference, the results without re-annotation have been included in Supplementary Table 3. The results, which are displayed in Figure 3a, 3b, and Supplementary Table 4, clearly demonstrate that VACmap consistently outperformed the other methods in terms of false negative rates, recall rates, and F1 score across PacBio CLR, HiFi, and ONT data.

The existing mapping methods frequently misalign the duplication sequence as insertion and hinder subsequent analyses. To evaluate the accuracy of VACmap on mapping duplication sequences, we identified tandem duplication calls within the GIAB SV benchmark set via REPTYPE annotation [29]. Then we compared the insertion calls detected by SVIM against the tandem duplication calls in the GIAB benchmark set. The purpose of this comparison was to quantify the proportion of tandem duplications that were incorrectly identified as insertions and assess the efficacy of different mapping algorithms in aligning tandem duplication sequences. Results, as depicted in Figure 3c and Supplementary Table 5, show that existing methods misalign approximately 90% of tandem duplications as insertions. In contrast, VACmap demonstrated superior performance, significantly outperforming the other three mapping algorithms by correctly identifying about 73-74.6% more tandem duplications (Supplementary Table 6). The result indicates that by leveraging the non-linear alignment algorithm, VACmap-produced alignments are more accurate and enable better characteristics of the precise structure of SVs. Then, we evaluated the performance of four mapping methods on SNPs and indels detection with DeepVariant on chromosome 20 under GRCh38 reference. “hap.py” was used to compare the DeepVariant call sets with the GIAB small variant benchmark set [30] to yield precision, recall, and F1 scores. The F1 scores of VACmap, NGMLR, Winnowmap2 and minimap2 were all between 99.0-99.1% and similarly for indel detection ranging from 99.7-99.9% (Supplementary Table 7). All four alignment algorithms exhibit an exceptional level of accuracy in indel and SNP detection and minor variations in their performance.

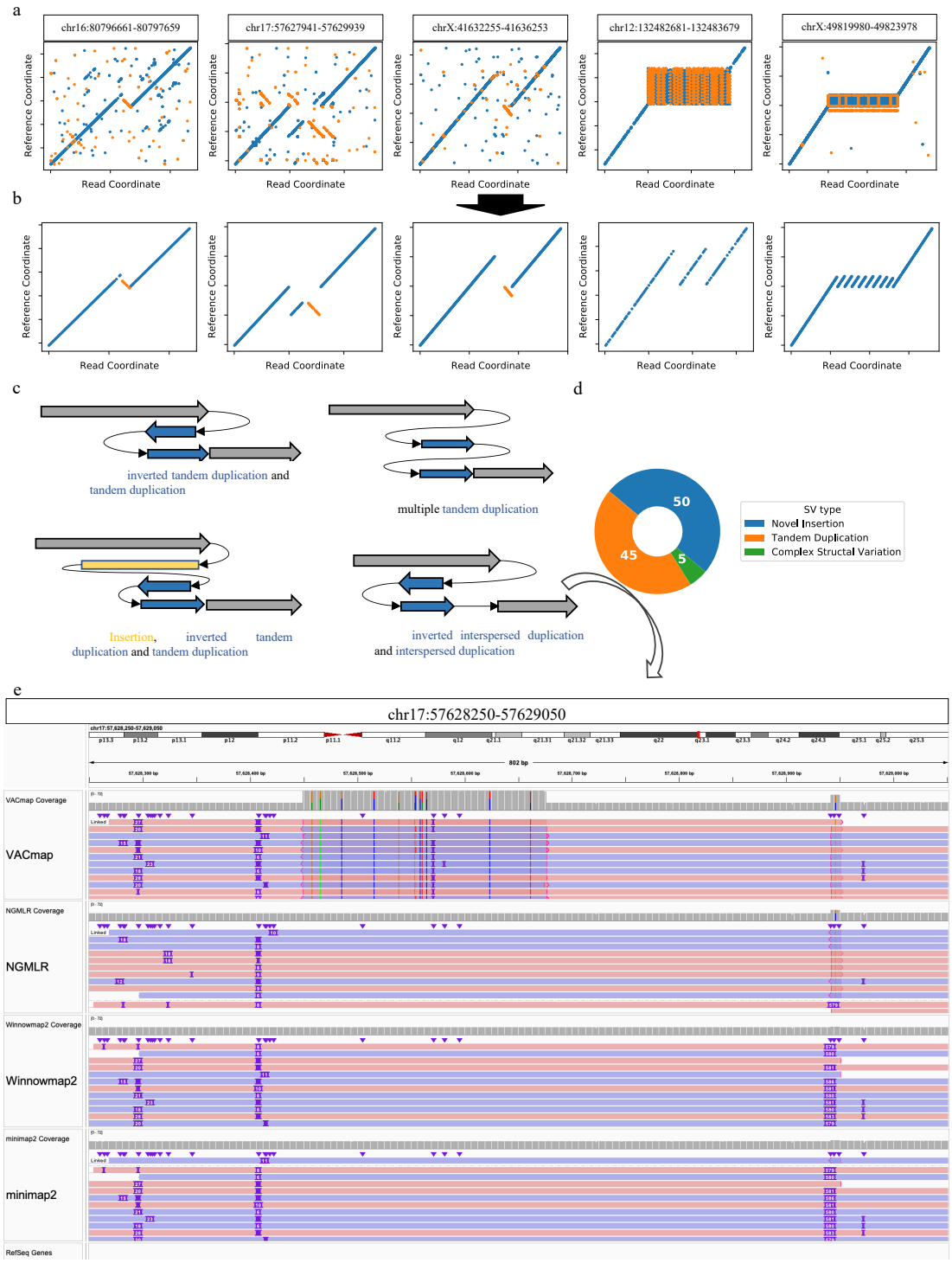

Figure 4 An analysis of the SV complexity of 100 randomly selected GIAB insertion calls. a. The dot plot of five CSV regions. **b.** The dot plot depicts non-linear alignments generated by the VAC algorithm of five CSV regions. **c.** A schematic representation of these five CSVs. **d.** A summary of the SV complexity derived from the 100 randomly selected GIAB insertion calls. **e.** The IGV visualization displays the alignments pileup image in an "inverted interspersed duplication" – "interspersed duplication" CSV region. The forward and reversed anchors are respectively marked in blue and orange in each dot plot.

These computation requirements of the four mapping methods were conducted on a 44-core intel CPU server using 40 CPU threads on three different datasets. As depicted in Figure 3d, and 3e, the runtime of VACmap is better with NGMLR, while it is slower compared to minimap2 and Winnowmap2. However,

173 VACmap required less memory compared to the other mapping algorithms.

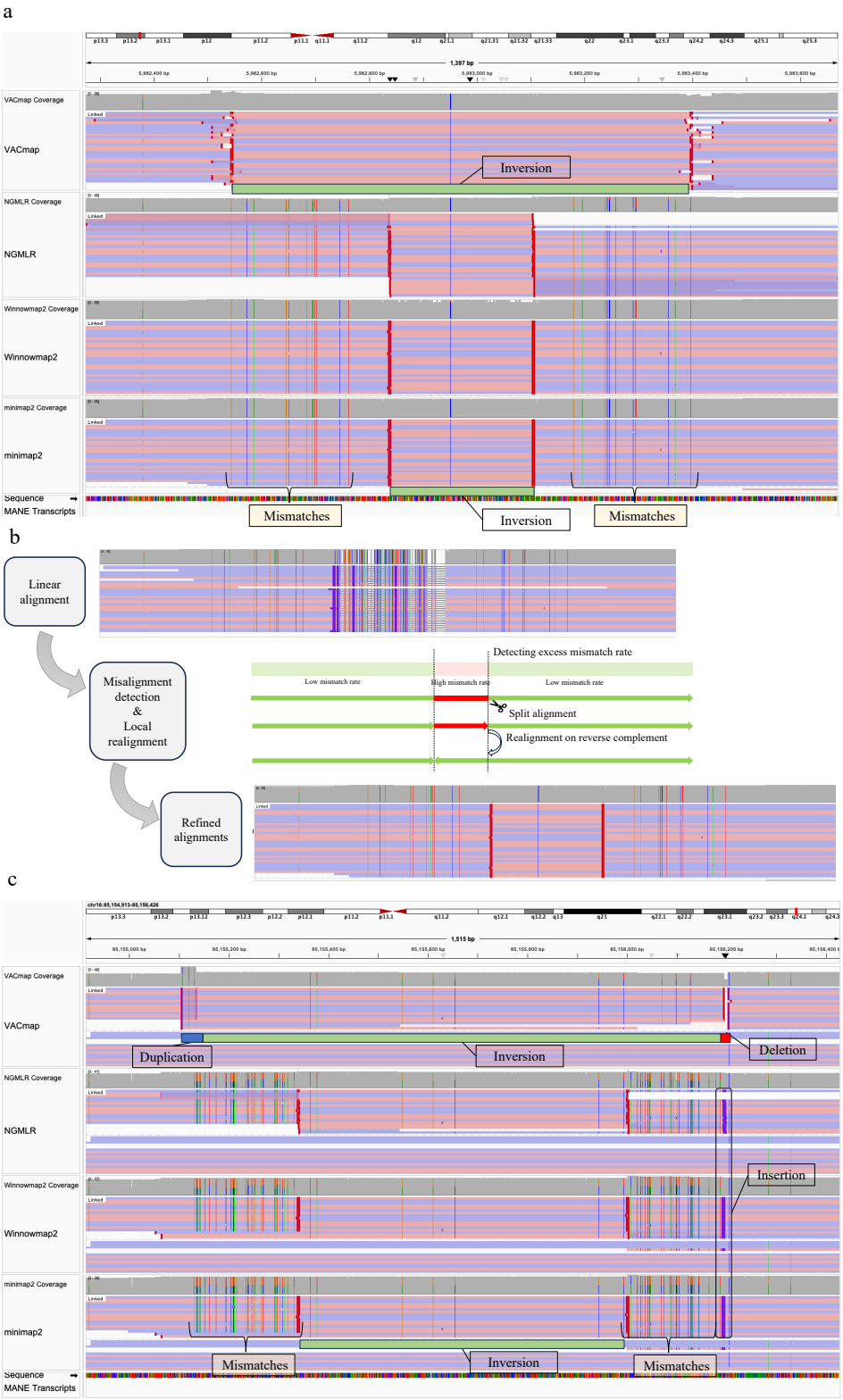

Figure 5 Mapping inverted sequence on HG002 human sample. a. The IGV visualization of four aligners produced alignments at a simple inversion region. **b.** The workflow of a heuristic approach used by minimap2 to recover misaligned inversion within linear alignments. **c.** The IGV visualization of four aligners produced alignments at a complex inversion region. VACmap-produced alignments successfully unveil the precise structure of both simple and complex inversion regions.

174 Next, we randomly selected 100 calls labeled as insertion from the GIAB benchmark set for manual

inspection. We found that 50, 45, and 5 of these calls were novel insertions, tandem duplications and CSVs, respectively (Figure 4d). These five CSVs are composed of duplication and insertion events, with the count of simple SV events ranging from 2 to 9 (Figure 4 a,b). Based on their patterns, we classify them into four types of CSVs: “inverted tandem duplication - tandem duplication”, “multiple tandem duplication”, “insertion - inverted tandem duplication - tandem duplications” and “inverted interspersed duplication - interspersed duplication” (Figure 5c). The Integrative Genomics Viewer (IGV) [31] visualization (Figure 4e, Supplementary Figure 2-5) shows that VACmap accurately aligns the sequences corresponding to each variation event, and thus accurately represents the structure of CSVs. In contrast, the other three mapping methods lack the ability to accurately identify the complex structure and leave these sequences unmapped or falsely align the corresponding sequence as insertion. Additionally, extra SNPs are found in four of these CSVs with an average occurrence of SNPs of 7 in every 1000 nucleotides (Figure 4e, Supplementary Figure 2-5). These SNPs nest in the sequences corresponding to CSVs further increasing the SV complexity.

Linear Mapping Obstruct Precise Inversion Discovery

Inversions stand as a crucial type of SV, yet they have not received as much attention as other SV forms, such as deletions and insertions [32]. Detecting inversions poses significant challenges due to their copy-neutral characteristic, which retains the collinearity of the up and down stream of the inversion region. This retention can confound traditional sequence alignment algorithms that are dependent on collinearity. To address this, heuristic methods have been developed that target subregions within a linear alignment, where a high frequency of mismatches may signal an inversion. By realigning the reverse complements of these suspect subregions, these methods strive to outline the inversions [5-7, 9]. However, their effectiveness is often constrained by the choice of parameters and an inability to fully capture the intricate reality of inversion events, which can encompass complex rearrangements beyond simple sequence reversals [13].

In evaluating the proficiency of VACmap and three other alignment tools in mapping inversions, we scrutinized regions within the HG002 human sample, drawing on a recent study [33]. In a segmental duplication flanked inversion site, the IGV visualizations (Figure 5a) revealed that while all algorithms flagged potential inversions, VACmap outperformed the others, particularly in the fidelity of its mapping. This was evidenced by a reduced number of mismatches near the inversion breakpoints in the VACmap alignments, suggesting the other tools were unable to pinpoint the precise boundaries of the inversions, leading to erroneous alignments. The limitations of conventional alignment methods in recovering inversion from linear alignment are exemplified in Figure 5b. These standard approaches typically scan for inversions by pinpointing regions within a linear alignment that exhibit anomalously high mismatch rates. In the case of the inversion under consideration, these methods discern the elevated mismatch rates at its center. However, they neglect the inversion's extremities, where mismatch rates are comparatively lower, and thus fail to trigger the detection algorithms. Moreover, for large inversions that extend beyond the length of a single sequencing read. Existing algorithms attempt to map the reads in fragments, hoping to reconstruct the full inversion from these partial alignments. This segmented approach maps each piece independently, leading to inconsistent and suboptimal results (Supplementary Figure 6).

Conventional alignment methods encounter significant difficulties when mapping more intricate inversion events. As depicted in Figure 5c, the IGV visualization of a mobile element insertion flanked inversion region reveals that alignments generated by NGMLR, Winnowmap2, and minimap2 are marred by extensive mismatched bases and erroneously inserted sequences, indicative of poor mapping quality. In stark contrast, the alignment produced by VACmap is markedly more accurate and coherent, clearly depicting a complex structural

configuration. VACmap delineates the inversion with precision, showing that the inversion's extremities are demarcated by an inverted duplication at one end and a deletion at the other. The challenge with existing aligners arises from their foundational design, which is based on a linear framework that struggles to accommodate the realities of complex, non-linear genomic structures. In this complex inversion region, the inverted duplication and deletion are incorrectly interpreted as insertion, and the failure to precisely identify the boundary of the inversion causes more mismatched bases. Further illustrating the capabilities of VACmap, Supplementary Figures 7 and 8 present two additional complex inversion regions. In these figures, VACmap's alignments effectively reveal the intricate architecture of the genomic segments, where inversions are flanked by deletions and inverted duplications. These examples underscore the advanced ability of VACmap to navigate and accurately represent the elaborate landscape of genomic inversions, surpassing the limitations of traditional linear-based aligners.

Analysis of the KIV-2 Repeat in the LPA Gene

The copy number of the KIV-2 repeat in the LPA gene is known to have an inverse relationship with human lipoprotein (a) levels, which are strongly linked to coronary heart disease [23]. However, accurately quantifying the KIV-2 copy number poses significant challenges due to the low mappability of both short and long sequencing reads in the LPA gene region [24]. In our study, we assessed the performance of four mapping methods by aligning PacBio HiFi and ONT sequencing data from human samples (CHM13 and HG002) against the GRCh38 reference genome. Our analysis, visualized using the IGV, revealed that NGMLR, Winnowmap2, and minimap2 produced alignments show excessive mismatches and less informative coverage information, in contrast to VACmap. VACmap demonstrated a superior ability to accurately represent KIV-2 repeats, showing clear and distinct coverage boundaries, as illustrated in Figure 6a and Supplementary Figures 9-11.

The GRCh38 reference contains six KIV-2 repeat units of two types: type A and type B. Directly mapping the KIV-2 repeat segment to the GRCh38 reference results in multiple mapping choices and complicates copy number determination. To simplify this, we removed the second to the sixth KIV-2 repeat units in the GRCh38

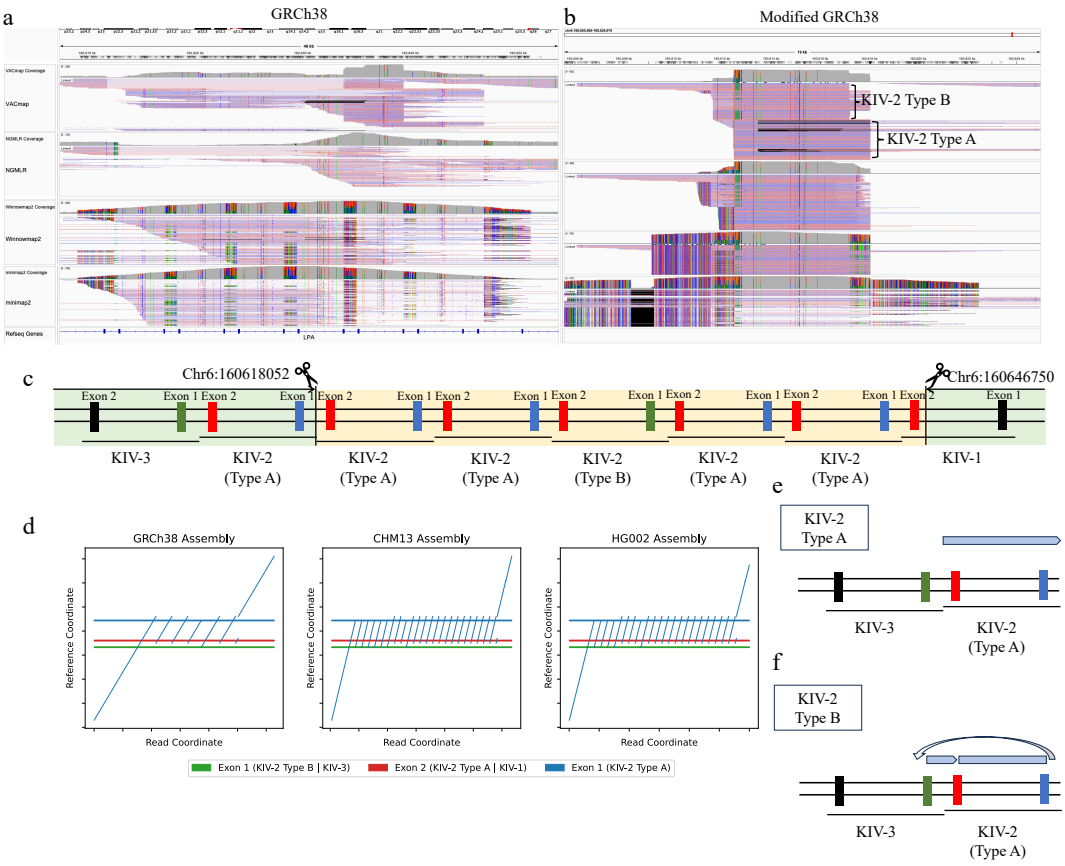

Figure 6 KIV-2 repeat in LPA region. a, b. The IGV visualization of alignments produced by four aligners using GRCh38 and modified GRCh38 reference in the KIV-2 region. **c.** The illustration of GRCh38 reference modification and the exon structure of the KIV-2 domain. The exon 2 (red) in the type A KIV-2 repeat unit, type B KIV-2 repeat unit, and KIV-1 repeat unit have 100% identical sequences. The exon 1 (green) in the KIV-3 repeat unit and type B KIV-2 have 100% identical sequences. The light green and light orange regions indicate the reserved and removed regions, respectively. This figure is modified from [23]. **d.** The dot plot depicting non-linear alignments generated by the VAC algorithm of GRCh38, CHM13, and HG002 assembly against modified GRCh38 reference. **e, f.** The alignments scheme of type A KIV-2 sequence and type B KIV-2 sequence against modified GRCh38 reference.

reference, along with the first 1000bp sequence of the follow-up KIV-1 unit, which includes an identical exon 2 sequence in the KIV-2 unit (Figure 6c). We then realigned the PacBio HiFi and ONT data to the modified reference. The KIV-3 unit precedes the KIV-2 unit in the LPA region (Figure 6c), and the KIV-3's exon 1 sequence is 100% identical to type B KIV-2's exon 1 sequence. When mapping the sequence segment corresponding to the type B KIV-2 unit, the exon 1 in KIV-3 serves as a decoy and forces the part of the type B KIV-2 unit sequence with exon 1 aligned to the KIV-3 unit region (Figure 6f). IGV visualization (Figure 6b, Supplementary Figure 12-14) indicated that VACmap-produced alignments showed the expected alignment scheme of both type A and type B KIV-2 units. However, the other three mapping methods struggled to produce correct alignments, despite the reduced complexity of the modified reference. Furthermore, the ONT read facilitated the resolution of all 23 copies of the KIV-2 repeat unit in the CHM13 sample (Supplementary Figure 15), due to its much longer read length compared to PacBio HiFi data.

To obtain a better complete understanding of the KIV-2 repeat region, we aligned the GRCh38 assembly, CHM13 assembly, and HG002 assembly to the modified GRCh38 reference [34]. The non-linear alignment of these three assemblies is shown in Figure 6d. Consistent with previous findings [23], we found the GRCh38 assembly consisted of six copies of KIV-2 repeat units with the scheme "AAABAA" where "A" indicates the

type A KIV-2 repeat unit, and "B" indicates the type B KIV-2 repeat unit. In the CHM13 assembly, 23 copies of KIV-2 repeat units were identified with the scheme "BBBBBBAABAAAAAAAAAAAAA", while in the HG002 assembly, 22 copies of KIV-2 repeat units were identified with the scheme "BBBBAAABAAAAAAAAAAAAAA".

Discussion and Conclusions

VACmap marks a transformative step in read alignment methodology and our ability to fully leverage the potential of long-read sequencing data. Since the advent of next-generation sequencing, read alignment algorithms have universally depended on assumptions of collinearity between the read and reference genome. This linear mapping approach suffices for small variants like SNPs and indels but fundamentally limits our capacity to interpret larger, more complex genomic rearrangements. Methods to date have attempted to improve SV detection through various heuristic tweaks to the linear alignment paradigm. However, these incremental efforts cannot overcome the core constraints of forcing complex, non-linear genomic architectures into a linear mold.

VACmap breaks through this long-standing barrier via an entirely new non-linear mapping approach designed explicitly to handle complex SVs. By moving beyond assumptions of collinearity, it can find optimal global non-linear alignments for reads spanning multiple interleaved SV events. This enables direct full-length mapping of reads irrespective of rearrangements. The empirical evidence presented here, drawn from both simulated and real datasets, provides convincing support for the utility of VACmap in genomic research. One of the most compelling facets of VACmap's performance is its ability to delineate tandem duplications with enhanced precision. This capability is of particular importance given the role of these duplications in gene dosage effects and their potential to influence phenotypic expression. By refining the accuracy of copy number estimations for tandem duplications, VACmap contributes a critical tool for genomic studies, particularly those focused on understanding the genetic underpinnings of complex diseases.

Moreover, VACmap is demonstrating the novelty and need of its method especially on inversions and other critical medically challenging genes such as *LPA* [22]. Indeed, inversions remain challenging to resolve especially due to their location often surrounded by large segmental duplications [33]. Furthermore, these regions often form more complex events than simple inversions. Neither of forms complex or simple inversions are routinely detectable with state of the art methods [33]. VACmap on the other side enables this detection with more precise placement of read segments than any other method available due to its novel non-linear mapping approach. This further improves the characterization of complex duplications such as shown in KIV-2 a region in LPA itself. The copy number of this 5.5kbp long region is reversely correlated with cardiovascular disease risk from birth on and is notoriously hard to align to [24]. VACmap can more precisely recapitulate the exact breakpoints within the reads and thus guarantee a better estimation of the individual copies per person. This is only one example of multiple medically important but challenging genes that VACmap can improve upon and thus deliver a more precise picture of the variants currently often missed by analytical methods [22].

While VACmap has shown a marked improvement in mapping accuracy for complex genomic rearrangements, it is noteworthy that its precision in mapping normal reads (Supplementary Table 8) and detecting SNPs and small indels is on par with established methods. This finding indicates that VACmap does not sacrifice the detection of smaller variants for the sake of resolving larger and more complex ones. Such balanced performance is crucial for ensuring that VACmap can serve as a comprehensive tool for genomic analysis.

The non-linear alignment algorithm underpinning VACmap is a conceptual leap that can redefine what read mappers can achieve. Its innovative approach to mapping long-read sequencing data provides a new lens through which we can interrogate the genome, revealing the intricate tapestry of structural variations that underlie genetic complexity. As such, VACmap is poised to become an indispensable tool in the genomicist's arsenal, with the potential to illuminate the genetic architecture of traits and diseases that have long eluded clear characterization.

Method

Variant-aware Chaining Algorithm

Current sequence alignment algorithms [2-11] attempt to align homology sequences via linear edits—deletions, insertions, and substitutions—that preserve the original sequence order and orientation. While effective for point mutations and small insertions or deletions (indels), this linear paradigm falls short when sequences exhibit complex genomic rearrangements such as duplications, and inversions. Under this linear framework, duplications are often erroneously interpreted as insertions due to their identical signatures from a linear alignment perspective. Inversions, which do not result in a net gain or loss of genetic material, are challenging to distinguish from block substitutions. For more complex rearrangements—where multiple rearrangement events are clustered closely—the linear mapping approach attempts to reconstruct the complex structure by identifying all potential linear subalignments and selecting the most suitable subalignments from the pool of possibilities. The ideal representation, however, should consist of a series of linear subalignments connected by non-linear junctions at the breakpoints of these rearrangements. Yet, accurately retrieving the optimal set of subalignments is particularly challenging, especially for sequences located in highly repetitive regions or with substantial rearrangement complexity [17]. Moreover, since each linear subalignment is aligned independently, there is no assurance that a selection from locally optimal subalignments will culminate in a globally optimal alignment.

We present a novel solution with VACmap, a method that significantly deviates from the linear alignment model. VACmap integrates the Variant-aware Chaining (VAC) algorithm, which deftly combines linear and non-linear edits to achieve alignment. This dual-operation approach empowers VACmap to deliver a unified, globally optimal alignment that accurately reflects the genomic rearrangement in its entirety, bypassing the flawed post-alignment selection process.

To elaborate, our approach involved identifying k-mers that matched exactly between a reference sequence and a long-read sequence. These exact matches are expressed as quadruples, denoted as (x, y, s, k), and are termed anchors. Here, 'x' represents the start position of the anchor within the long-read sequence, while 'y' indicates the corresponding start position within the reference sequence. The variable 's' signifies the orientation of the match, with a value of 1 indicating alignment with the forward strand and -1 indicating alignment with the reverse strand. The 'k' variable specifies the extent of the sequence encompassed by the anchor. Within the framework of the VAC algorithm, these anchors are conceptualized as nodes within a graph that is both directed and weighted. We apply specific rules to the linkage of these nodes: Only nodes that precede others in the long-read sequence can form connections to those that follow, and there must be no overlap in the long-read sequence coverage between any two connected nodes. To determine the weight of

the directed edge from node[i] to node[j], where i is less than j, symbolized as w(i, j), we utilize a set of predefined formulas.

$$diff < max\left(maxdiff - c_j\ , 10\right), 0 \leq\ s_j \times refgap < maxgap, s_i = s_j \tag{1}$$

$$w(i,j) = k_j - CP_{i,j} \tag{2}$$

$$CP_{i,j} = 0.01 \times k_j * diff + \alpha \times log_2(\max(diff,\ 1)) \tag{3}$$

$$w(i,j) = k_j - VP_{i,j} \tag{4}$$

$$VP_{i,j} = \beta + min\left(c_j, 20\right)\ + min\begin{cases} 30 + 0.5 \times \ln(\max\ (1, diff)) \\ \min\left(10, \frac{diff}{100}\right) + \min\left(30, \frac{diff}{1000}\right) \\ \gamma \end{cases} \tag{5}$$

$$if\begin{cases} a = 0.5 & diff < 10 \\ a = 2 & diff \geq 10 \end{cases} \tag{6}$$

$$diff = \left|readgap - |refgap|\right| \tag{7}$$

$$readgap = \ x_j - x_i - k_i \tag{8}$$

$$refgap = \begin{cases} y_j - y_i - k_i, \text{if } s_\text{j} = 1 \\ y_j - y_i + k_j, \text{if } s_\text{j} = -1 \end{cases} \tag{9}$$

When two nodes exhibit strict collinearity, as defined by Formula 1, we calculate their connecting edge weight using Formula 2. Edges that fall under this category are termed "normal" edges. Formula 1 accounts for small indels between nodes, which are regulated by the hyperparameter *maxdiff*, set by default to 30. To incorporate an additional penalty for small indels, we adopt the affine-convex gap penalty model from minimap2 as outlined in Formula 3, which we refer to as the *CP* (colinear penalty). Conversely, when nodes do not conform to the conditions of Formula 1, we determine their edge weight with Formula 4, designating these as "variation" edges. The variation penalty, *VP*, is established through Formula 5, with its maximum value set to the sum of β, γ, and 20. The default settings for β and γ are 30 and 36, respectively. Given that k-mers occurring with high frequency are typically less informative [5], we impose an additional penalty, c[j], when extending a chain to a node associated with a frequently occurring k-mer, thus enforcing a stricter collinearity criterion. Here, c[j] represents the count of occurrences in the k-mer index corresponding to node j. For further clarity regarding these principles, we provide illustrative examples in Supplementary Figure 16 to show the conditions for assigning normal and variation edges. The selection of hyperparameter values critically affects the quality of the non-linear alignment, with maxdiff and maxgap (default values of 30 and 100, respectively) being particularly influential. A reduced maxdiff value tends to yield a more strictly collinear subalignment, while a smaller maxgap value helps to prevent the erroneous alignment of unrelated sequences.

To find the optimal non-linear alignment path among *N* nodes can be computed in *O(hN)* time if by dynamic programming. For a given *node[j]*, we can determine its maximum score *S(j)* and its best predecessor *node[i]* by utilizing the following formulas:

$$S(j) = max\left(max_{i \in M}\left(S(i) + w(i,j)\right), k_j\right) \qquad i \in M \tag{10}$$

$$i \in M \qquad if\ S(i)\ \geq max_{k<j \wedge x_j \geq x_k + k_k}\left(S(k)\right) - (\beta + \gamma + 20) \tag{11}$$

The *node[j]* best predecessor score $S(i)$ has a low bound (Formula 10, 11). The low bound is equal to the current highest score *S[k]* minus $(\beta + \gamma + 20)$, and *h* is equal to the size of set M.

Once the maximum score of all nodes is computed, we can identify the highest scores non-colinear chain by backtracking. The optimal set of colinear subalignments is recovered by discarding variation edges in the highest score chain.

Local Index and Anchor Reduction

When sequence errors or SNPs 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 [8] 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 VAC 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_i, y_i, s_i, k_i) and (x_j, y_j, s_j, k_j) into a new anchor $(x_i, min(y_i, y_j), s_i, x_j + k_j - x_i)$ if they satisfy certain conditions (Formula 12). 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, 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.

$$\begin{cases} y_i - x_i = y_j - x_j, x_i + k_i \geq x_j, s_j = s_i = 1, x_j > x_i, x_j + k_j - x_i \leq 19 \\ \quad or \\ y_i + x_i + k_i = y_j + x_j + k_j, x_i + k_i \geq x_j, s_j = s_i = -1, x_j > x_i, x_j + k_j - x_i \leq 19 \end{cases} \tag{12}$$

Quality Control of Linear Subalignment

The presence of sequencing error and closely situated SNPs complicates the process of gathering sufficient anchors for the chaining process. To accommodate the occurrence of sequencing errors and clustered SNPs, VACmap employs a sizeable hyperparameter, maxgap, with a default setting of 100, within the VAC algorithm. However, a large maxgap value has the potential drawback of mistakenly aligning sequences that are not actually related. To address this problem, we assess the error ratio within each linear subalignment and exclude those that demonstrate a high propensity for errors. This error ratio is ascertained by employing edlib [35] to calculate the edit distance between the matched sequences in the reference genome and the long read. The edit distance obtained is then normalized by dividing the length of the shorter sequence involved in the comparison. A linear subalignment is deemed unreliable and is therefore rejected if its error ratio surpasses the predetermined threshold: 0.2 for sequences generated using PacBio CLR or ONT, and 0.1 for sequences generated using PacBio HiFi.

Synthetic Data Simulation and Evaluation

To assess the performance of different mapping methods in downstream SV detection tasks, we employed the CHM13 T2T (v2.0) human reference genome [26] to generate two sets of synthetic datasets. For both simple and complex SVs simulation, we utilized VACsim to randomly introduce 30,000 SVs into the CHM13 T2T reference and generated the altered genome sequence containing both simple and complex SVs. VACsim determined the SV complexity (measured by the number of simple structural variation events within) of each simulated SV by randomly sampling from the range 1 to 20. The types of simple structural variation events were sampled from five simple SV events, namely deletion, insertion, duplication, inversion, and translocation, with respective probabilities of 0.24, 0.24, 0.24, 0.24, and 0.04. The size of each simple SV event ranged from 100 to 1000 bps. Additionally, SVIM (v1.4.2) employed a pattern match strategy to detect SVs, requiring each simple SV event to conform to the "Normal+Variation+Normal" pattern. Therefore, we spaced adjacent simple SV events by a normal sequence of 200 base pairs in size. For the copy number estimation task, VACsim was used to simulate and implant 10000 tandem duplications with repeat unit sizes ranging from 100 to 500 bps and repeat numbers ranging from 1 to 20 into chromosome 1 of the CHM13 T2T reference genome sequence. The SNPs were introduced into the altered sequences and generated simulated PacBio and ONT long-read datasets by SURVIVOR. For the clean read datasets, SURVIVOR is used to generate PacBio and ONT simulated long reads from chromosome 1 of the CHM13 T2T reference genome sequence.

Four long-read mapping methods, namely VACmap, NGMLR, Winnowmap2, and minimap2, were utilized to align the simulated long reads to the T2T reference genome sequences. The SVIM (v1.4.2) SV detection tool was employed to detect SVs from the alignments generated by the different mapping methods. Following previous studies [6], we evaluated the performance of CSV calls by decomposing CSVs into individual simple SV events and evaluating each event separately.

Furthermore, following the procedure outlined in a previous study [9], we employed Mashmap [36] to identify repetitive regions in the reference genome sequence with sequence similarity exceeding 95% and a sequence length greater than 10,000 bps. Subsequently, we analyzed the performance of downstream SV detection within these repetitive regions using alignments produced by different mapping algorithms. Command line parameters provided to these tools are listed in Supplementary Table 9.

Evaluation with GIAB callset

We evaluated four mapping methods, namely VACmap, NGMLR, Winnowmap2, and minimap2, using the GIAB Tier1 (v0.6) SV benchmark set and small indels benchmark set benchmark set for the HG002, and HG003 human sample, respectively. These benchmark sets are available at the following links: https://ftp-trace.ncbi.nlm.nih.gov/ReferenceSamples/giab/release/AshkenazimTrio/HG002_NA24385_son/NIST_STRUCTURAL VARIANT_v0.6/ and ftp://ftp-trace.ncbi.nlm.nih.gov/giab/ftp/release/AshkenazimTrio/HG003_NA24149_father/NISTv4.2.1/GRCh38/. PacBio CLR, PacBio HiFi, and ONT sequencing data were accessible at s3://giab/data/AshkenazimTrio/. The SVIM (v1.4.2) and DeepVariant (v1.5.0) were used to detect SVs and small indels and SNPs, respectively. Subsequently, we evaluated the SVs and small indels and SNPs call sets against the GIAB benchmark set using Truvari (v2.0.0) and hap.py (v0.3.12), respectively. Command line parameters provided to these tools are listed in Supplementary Table 9.

Data availability

This study utilized four human reference genome sequences: GRCh37, GRCh38, CHM13 T2T (v2.0) and HG002 T2T (v1.1). 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/, https://github.com/marbl/CHM13, and https://s3-us-west-2.amazonaws.com/human-pangenomics/T2T/HG002/assemblies/hg002v1.0.1.fasta.gz, respectively.

For the Pacbio CLR, Pacbio HiFi, 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_HIFI_15kb/, and s3://giab/data/AshkenazimTrio/HG002_NA24385_son/UCSC_Ultralong_OxfordNanopore_Promethion/.

The Pacbio HiFi and ONT data for the human sample CHM13 were downloaded from SRA under accessions SRR11292120-SRR11292123 and s3://human-pangenomics/T2T/CHM13/nanopore/UW/chm13_UW_Guppy_3.6.0.fastq.gz, respectively.

Code availability

VACmap are available on GitHub: https://github.com/micahvista/VACmap.

Reference

[1] Mahmoud, M., Gobet, N., Cruz-Dávalos, D.I. et al. Structural variant calling: the long and the short of it. Genome Biol 20, 246 (2019). https://doi.org/10.1186/s13059-019-1828-7

[2] Faust GG, Hall IM. YAHA: fast and flexible long-read alignment with optimal breakpoint detection. Bioinformatics. 2012;28(19):2417–24.

[3] Chaisson MJ, Tesler G. Mapping single molecule sequencing reads using basic local alignment with successive refinement (BLASR): application and theory. BMC Bioinformatics. 2012;13(1):1–18.

[4] Li H. Minimap and miniasm: fast mapping and de novo assembly for noisy long sequences. Bioinformatics. 2016;32(14):2103–10.

[5] Li H. Minimap2: pairwise alignment for nucleotide sequences. Bioinformatics. 2018;34(18):3094–100.

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

[7] Fu Y, Mahmoud M, Muraliraman VV, Sedlazeck FJ, Treangen TJ. Vulcan: Improved long-read mapping and structural variant calling via dual-mode alignment. GigaScience. 2021;10(9):giab063.

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

[9] 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.

[10] Alser M, Rotman J, Deshpande D, Taraszka K, Shi H, Baykal PI, et al. Technology dictates algorithms: recent developments in read alignment. Genome Biol. 2021;22(1):1–34.

[11] Sahlin, K., Baudeau, T., Cazaux, B. et al. A survey of mapping algorithms in the long-reads era. Genome Biol 24, 133 (2023). https://doi.org/10.1186/s13059-023-02972-3

[12] Armstrong, J., Fiddes, I. T., Diekhans, M., & Paten, B. (2019). Whole-Genome Alignment and Comparative Annotation. Annual review of animal biosciences, 7, 41–64. https://doi.org/10.1146/annurev-animal-020518-115005

[13] 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.

[14] Newman, T. L., Tuzun, E., Morrison, V. A., Hayden, K. E., Ventura, M., McGrath, S. D., Rocchi, M., & Eichler, E. E. (2005). A genome-wide survey of structural variation between human and chimpanzee. Genome Research, 15(10), 1344–1356. https://doi.org/10.1101/gr.4338005.

[15] Giner-Delgado, C., Villatoro, S., Lerga-Jaso, J. et al. Evolutionary and functional impact of common polymorphic inversions in the human genome. Nat Commun 10, 4222 (2019). https://doi.org/10.1038/s41467-019-12173-x

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

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

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

[19] 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.

[20] 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.

[21] 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.

[22] Wagner, J., Olson, N.D., Harris, L. et al. Curated variation benchmarks for challenging medically relevant autosomal genes. Nat Biotechnol 40, 672–680 (2022). https://doi.org/10.1038/s41587-021-01158-1.

[23] Konrad Schmidt, Asma Noureen, Florian Kronenberg, Gerd Utermann, Structure, function, and genetics of lipoprotein (a), Journal of Lipid Research, Volume 57, Issue 8, 2016, https://doi.org/10.1194/jlr.R067314..

[24] Chen-Shan Chin, Sairam Behera, Ginger A. Metcalf, Richard A. Gibbs, Eric Boerwinkle, Fritz J. Sedlazeck. A pan-genome approach to decipher variants in the highly complex tandem repeat of LPA. bioRxiv 2022.06.08.495395;

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

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

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

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

[29] English, A.C., Menon, V.K., Gibbs, R.A. et al. Truvari: refined structural variant comparison preserves allelic diversity. Genome Biol 23, 271 (2022). https://doi.org/10.1186/s13059-022-02840-6.

[30] Zook, J., Catoe, D., McDaniel, J. et al. Extensive sequencing of seven human genomes to characterize benchmark reference materials. Sci Data 3, 160025 (2016). https://doi.org/10.1038/sdata.2016.25

[31] 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.

[32] Hanlon, V. C. T., Lansdorp, P. M., & Guryev, V. (2022). A survey of current methods to detect and genotype inversions. Human Mutation, 43(11), 1576–1589.

[33] Porubsky D., Höps W., Ashraf H., Hsieh P., Rodriguez-Martin B., Yilmaz F., et al. (2022). Recurrent inversion polymorphisms in humans associate with genetic instability and genomic disorders. Cell 185, 1986–2005.e26. 10.1016/j.cell.2022.04.017

[34] Rautiainen M, Nurk S, Walenz BP, Logsdon GA, Porubsky D, Rhie A, Eichler EE, Phillippy AM, Koren S. Telomere-to-telomere assembly of diploid chromosomes with Verkko. Nat Biotechnol. 2023 Oct;41(10):1474-1482.

[35] 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.

[36] 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).

Acknowledgment

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

Competing interests

The authors declare no competing interests.