OMKar: optical map based automated karyotyping of genomes to identify constitutional abnormalities

Siavash Raeisi Dehkordi[1, 2, 5], Zhaoyang Jia[1, 5], Joey Estabrook[2], Jen Hauenstein[2], Neil Miller[2], Naz Güleray-Lafci[4], Jürgen Neesen[4], Alex Hastie[2], Alka Chaubey[2], Andy Wing Chun Pang[2], Paul Dremsek[4], and Vineet Bafna[1, 3, *]

[1]Department of Computer Science & Engineering, UC San Diego, La Jolla, California, USA
[2]Bionano Genomics, Inc., 9540 Towne Centre Drive Suite 100, San Diego, CA 92121
[3]Halicioğlu Data Science Institute, UC San Diego, La Jolla, California, USA
[4]Institute of Medical Genetics, Center for Pathobiochemistry and Genetics, Medical University of Vienna, 1090 Vienna, Austria
[5]These authors contributed equally to this work.
[*]Correspondence: vbafna@ucsd.edu

Abstract

The whole genome karyotype refers to the sequence of large chromosomal segments that make up an individual's genotype. karyotype analysis, which includes descriptions of aneuploidies and other rearrangements is crucial for understanding genetic risk factors, for diagnosis, treatment decisions, and genetic counseling linked to constitutional disorders. The current karyotyping standard is based on microscopic examination of chromosomes, a complex process that requires high expertise and offers Mb scale resolution.

Optical Genome Mapping (OGM) technology can identify large DNA lesions in a cost-effective manner. In this paper, we developed OMKar, a method that uses OGM data to create a virtual karyotype. OMKar processes Structural (SV) and Copy Number (CN) Variants as inputs and encodes them into a compact breakpoint graph. It recomputes copy numbers using Integer Linear Programming to maintain CN balance and then identifies constrained Eulerian paths representing entire donor chromosomes. In tests using 38 whole genome simulations of constitutional disorders, OMKar reconstructed the karyotype with 88% precision and 95% recall on SV concordance and 95% Jaccard score on CN concordance. We applied OMKar to 50 prenatal, 41 postnatal, and 63 parental samples from ten different sites. OMKar reconstructed the correct karyotype in 144 out of 154 samples, covering 25 of 25 aneuploidies, 32 of 32 balanced translocations, and 72 of 82 unbalanced variations. Detected constitutional disorders included Cri-du-chat, Wolf-Hirschhorn, Prader-Willi deletions, Down, and Turner syndromes. Importantly, it identified a plausible genetic mechanism for five cases of constitutional disorder that were not detected by other technologies.

Together, these results demonstrate the robustness of OMKar for OGM-based karyotyping. OMKar is publicly available at `https://github.com/siavashre/OMKar`.

1 Introduction

Genomic structural variants involving the loss, amplification, or rearrangement of large genomic regions have been associated with many constitutional diseases[1]. The Decipher database lists over 2500 disorders, often caused by large structural changes in the genome, including trisomy, microdeletions and duplications, and other rearrangements[2]. For molecular diagnosis, affected individuals undergo standard of care (SOC) testing from drawn blood, where the extracted DNA is analyzed for genetic lesions. Genetic prenatal testing is also an important need, despite the recent advancements of non-invasive screening (NIPS) methods, typically utilizing maternal blood samples. Data from large studies suggest that while the negative predictive value (TN/(FN+TN)) was close to 100% for Down syndrome (trisomy 21) screening, the precision was in the 50-81% range, and the numbers were similar for other disorders[3,4]. Thus, a positive NIPS result is typically followed by a more invasive molecular diagnostic.

The current standard of care for genetic diagnostic tests includes (a) karyotyping, (b) chromosomal microarray (CMA)[5,6], (c) FISH screening[7,8], (d) panel sequencing, or (e) whole exome sequencing. Karyotyping methods require considerable manual expertise and have a low resolution of 3-10Mbp[9]. They can be combined with CMA or whole exome sequencing to improve resolution for detecting copy number changes. These high resolution methods (CMA, panel sequencing) do not easily detect copy number neutral rearrangements. FISH requires knowledge of probes and is therefore limited in detecting novel variations. In contrast, about 50% of all reciprocal translocations are *de novo*[10]. Balanced rearrangements are found in 0.2% of individuals (up to 2.2% of the individuals with a previous history of miscarriage). Individuals with a balanced translocation may not directly present with a phenotype/syndrome, but during meiosis, a gamete could carry an unbalanced copy number and result in fertility issues[11,12]. However, balanced translocations would be very likely missed by exome sequencing/CMA.

Optical Genome Mapping (OGM) provides an exciting alternative to diagnostic technologies that lie between cytogenetics and exome sequencing in terms of resolution. OGMs are large enough to span repetitive and low complexity regions, while still being able to capture smaller structural variations. While OGM technology cannot call single nucleotide substitutions, or small insertions and deletions, it is well-suited for calling aneuploidies, larger structural variations, balanced and unbalanced rearrangements, inversions and deletions[13,14], especially with the development of advanced tools[15,16].OGM has been used to successfully identify constitutional genomic lesions, despite some limitations[12,17–19]. In principle, OGMs can be supplanted by long read whole genome sequencing[20], but these methods are not yet readily available in a clinical setting. In fact, the demand for OGM based diagnostics is increasing[21–23].

Automated karyotyping using OGM. The Molecular karyotype of a donor can be described as a collection of genomic sequences, each sequence corresponding to one donor chromosome. Traditionally, the karyotype information was captured by cytogenetics, albeit at low resolution, and helped identify balanced and unbalanced rearrangements, aneuploidies, and other events that are

directly relevant to constitutional disorders. In moving from cytogenetics to CMA and exome sequencing, much of that important information was lost. Current methods for SV calling typically do not capture the larger karyotype making it harder to assign significance, for example, to a translocation event, or determining the locations of amplified genomic segments[24].

Here, we present a method, *OMKar*, for automatically identifying karyotypes using OGM data. We tested our method using extensive simulations as well as on OGM data acquired from over 100 prenatal and postnatal samples with constitutional disorders to gain an improved understanding of the power and limitations of the OGM technology for karyotyping.

2 Methods

Segment.	An oriented, continuous genomic interval from the reference genome, denoted by ⟨chromosome, start-coordinate, end-coordinate⟩. A **donor chromosome** is described as an ordered sequence of segments.
Breakpoint.	A breakpoint is described by a pair of non-adjacent coordinates denoting a transition from one segment to another in the donor.
Chromosome Group.	A set of all homologous donor chromosomes sharing the same chromosomal identity. The chromosomal identity is determined by the most represented centromere, and if the chromosome is acentric, by the most represented chromosomal origin of its composing segments.
Chromosome Cluster.	A pair of chromosome groups is denoted as dependent if there exist breakpoints connecting them. A **chromosome cluster** is a connected component of dependent chromosome groups. A chromosome cluster is often defined by a set of **canonical structural variants**, each with an ISCN nomenclature (International Standard of Cytogenetic Nomenclature).
Molecular karyotype.	A proposed file format that unambiguously describes the karyotype in terms of segments, with nucleotide-level resolution. This file format contains a dictionary of segments that span the entire reference genome, followed by a set of ordered sequences of segments, each representing a chromosome.

Table 1: **Terminology.**

OMKar processes the output of the Bionano Solve pipeline[25], which includes structural variation (SV), copy number variation (CNV), and contig alignment data. It generates a *Molecular karyotype* (Table 1) in a custom text format (Supplementary Section S1.5) and also presents the karyotype as chromosomal clusters using ISCN language, with reference genome coordinates instead of cytogenetic bands. This approach bridges karyotyping and SV calling. Additionally, OMKar provides a graphical karyotype display. The OMKar method (Fig. 1) is detailed in Section 2.1 below.

Prior to our work, formal measurements of karyotyping accuracy were lacking. To address this, we developed two additional tools: KarSim, which generates random karyotypes in Molecular karyotype and FASTA formats, and KarCheck, which compares two karyotypes by measuring their SV and CNV similarities. These tools (Sections 2.2, and 2.3) help improve method comparisons and enable cross-technology evaluations.

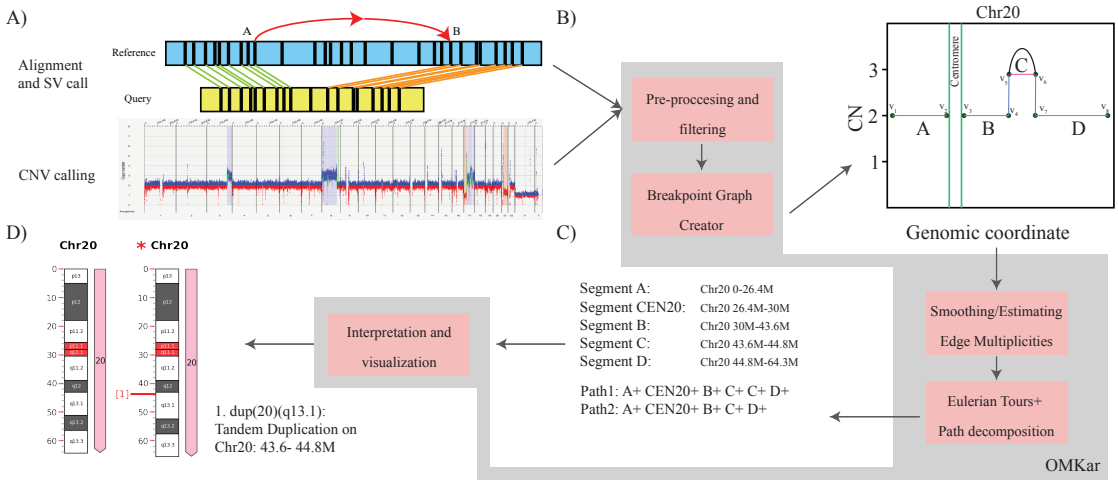


Figure 1: **Overview of the OMKar method**. (A) Input data: OMKar takes structural variant (SV) calls, copy number variation (CNV) calls, and sequence alignments as input. (B) Pre-processing and breakpoint graph construction after filtering SV and CNV calls to remove low-confidence variations. Chromosomes are segmented based on CNV boundaries and breakpoints, and a breakpoint graph is constructed, where vertices represent segment boundaries and edges represent segment continuity, reference adjacencies, and rearrangements. (C) Smoothing and path decomposition: Integer Linear Programming is used to estimate edge multiplicities, ensuring consistency of copy number constraints. Edge editing is used to generate a Euclidean graph, and paths extraction methods used to reconstruct chromosomes. (D) Interpretation and visualization: Structural variations are re-coded using ISCN, disrupted genes are identified, and results are compiled into an interactive HTML report with chromosome visualizations.

2.1 The OMKar method

The OMKar algorithm follows a multi-step process: (a) pre-processing of input data, (b) construction of a breakpoint graph, (c) smoothing of edge multiplicities to generate an Eulerian graph, (d) computation of an Eulerian tour, and (e) chromosomal segregation and identification to derive the Molecular karyotype.

Pre-processing. OMKar filters the SV and CNV calls to ensure data quality and relevance of the SV to karyotyping. OMKar utilizes the default Bionano pipeline thresholds for calling CNVs and SVs to ensure that only reliable variants are used in karyotype reconstruction. The Bionano pipeline maintains a database of regions that are observed with structural or copy number variation, and masks genomic regions that are frequently seen in normal samples. OMKar filters out CNV calls in the masked regions if they do not have supporting SVs. Finally, OMKar filters CNVs smaller than 200 Kbp, and those are reincorporated as local changes after karyotype reconstruction. (see Supplementary Section S1.1).

Breakpoints from SV calls are further processed, sorting them by chromosomal and genomic coordinates and merging adjacent breakpoints within a 50 Kbp window, while also ensuring precise representation of structural variations by splitting CNV segments when breakpoints occur within their boundaries. The result of this pre-processing is a partitioning of each chromosome into a

minimal number of segments, so that (a) each segment has a nearly uniform copy-number, (b) all breakpoints link only the end-coordinates of segments, (c) segments span all regions of reference chromosomes with copy number ≥ 1.

Breakpoint graph construction. We use the genome segment partitioning to generate a breakpoint graph[26] $G(V, E_s \cup E_r \cup E_b)$. Each vertex $v \in V$ corresponds to the end-coordinate of a segment. The set of *segment-edges* is defined using $E_s = \{(u, v)$ s.t. u and v are canonically the head and tail-node of the same segment$\}$. The *non-segment* edges include two sets. First, the set of *reference-edges*, $E_r = \{(v, u)\}$ where the head-node u of a segment is *adjacent in reference coordinates* to the tail-node v of another segment, and second, *breakpoint-edges* $(u, v) \in E_b$ joining vertex u to a *non-adjacent* vertex v. By definition, each vertex v is incident on exactly one segment-edge, at most one reference-edge, and possibly multiple breakpoint edges denoted by $E_b[v]$.

Smoothing Edge Multiplicities to generate an Eulerian graph. We use an Integer Linear Programming (ILP)[27] formulation to constrain the copy number of each genomic segment. Let c_v denote the copy number assigned to the segment-edge incident on vertex v. The ILP assigns copy numbers $r_v \geq 0$ to the reference edge, $s_e \geq 0$ to each edge in $E_b[v]$ and auxiliary values x_v while enforcing the following constraints:

$$r_v + \sum_{e \in E_b[v]} s_e - c_v \leq x_v, \quad |x_v| \leq \lceil \frac{c_v}{4} \rceil \quad \forall v \in V \tag{1}$$

These two constraints ensure that the sum of copy numbers of outgoing edges from a segment is not greater than the segment's assigned copy number. Moreover, if a reference-edge incident on v is supported by a contig alignment that spans the adjacent segments, then $r_v \geq 1$. In order to make the graph Eulerian, we need to reduce the number of vertices with odd degrees. For this purpose binary parameter o_v is defined as follows:

$$o_v = \begin{cases} 1 & \text{if the degree of } v \text{ is odd,} \\ 0 & \text{if the degree of } v \text{ is even} \end{cases} \quad \text{and} \quad o_v = c_v + x_v + r_v + \sum_{e \in SV(v)} s_e - 2o'_v$$

In which $o'_v \geq 0$ is an auxiliary variable added to the ILP. We minimize an objective function

$$\min \underbrace{\gamma \sum_{v \in V} (c_v + x_v - r_v - \sum_{e \in SV(v)} s_e)}_{(a)} + \underbrace{\sum_{v \in V} \beta_v |x_v|}_{(b)} - \underbrace{\sum_{s_e \in SV} \alpha_e \cdot \text{sgn}(s_e)}_{(c)} + \underbrace{\lambda \sum_{v \in V} o_v}_{(d)} \tag{2}$$

where (a) penalizes for the discrepancy between observed plus slack copy number of segment edges and the total copy number of adjacent reference and breakpoint edges; (b) penalizes for using non-zero slack; (c) provides a reward for using breakpoint edges at least once and (d) penalizes for having odd degree vertices. The actual implementation linearizes the non-linear terms (Supplementary Section S1.2). The objective includes the parameters $\gamma, \alpha_e, \beta_v, \lambda$ which were set

empirically(Supplementary Section S1.3).

Computing Eulerian tours. Following the estimation of edge multiplicities, we utilize a Breadth-First Search (BFS) algorithm to identify all connected components within the graph, each component representing a chromosome cluster. For each connected component, denoted as C, our approach initiates with the identification of vertices that represent the telomeric regions of chromosomes. If C contains odd-degree non-telomeric vertices, we connect such pairs using dummy edges to transform C into an Eulerian structure. Algorithm 1 (Supplementary Section S1.4) is used to compute Eulerian tours originating from one of the telomeric vertices.

Chromosomal Segregation and identification. In this formulation, each chromosome is a subpath with alternating segment and non-segment edges. A connected component may incorporate multiple chromosomes. Therefore, in Algorithm 1, we compute Eulerian paths that force an alternation between segment and non-segment edges. In case the only possible transition from segment edge (u, v) is to the segment edge (v, u), this represents a boundary between homologous chromosome pairs, and the subpaths are split at node v.

Eulerian decomposition is not unique, and multiple decompositions may exist. Let us assume that after path segregation, we have a set of sub-paths $P = P_1, P_2, \ldots$. Now, consider two paths P_i and P_j that share the same segment s. In that case, a 'crossover transition'

$$\begin{array}{ll} P_i = a_1, a_2, \ldots, s, b_1, b_2, \ldots & \xrightarrow{\text{crossover transition}} \quad P'_i = a_1, a_2, \ldots, s, b'_1, b'_2, \ldots \\ P_j = a'_1, a'_2, \ldots, s, b'_1, b'_2, \ldots & \qquad\qquad\quad P'_j = a'_1, a'_2, \ldots, s, b_1, b_2, \ldots \end{array}$$

would generate another valid Eulerian path decomposition. We use this idea to heuristically refine the chromosomes based on known biology. Specifically, OMKar counts the number of centromeres in each path. If there exists a pair of paths–one containing two centromeres with at least one segment s in between and a second chromosome containing segment s but zero centromeres–OMKar performs a crossover transition on s to ensure that both paths now contain a single centromere. Finally, to standardize orientation, we flip the chromosomes so that they are all oriented in the p-to-q direction. Each chromosome's orientation is identified by the centromeric segment, or, if it is acentric, by the majority orientation of all segments.

Event Interpretation. Structural variants have somewhat conflicting definitions within the Genomics and Cytogenetics community. We developed an Event Interpretation module to describe SVs using the International System for human Cytogenomics Nomenclature (ISCN), described in Supplementary Table S1. OMKar automates the interpretation as follows (See also, Supplementary Section S1.6): It aligns structural variations (SVs) in reconstructed chromosomes with their wild-type (WT) counterparts, identified by centromeric or segmental makeup. It uses the Longest Common Subsequence algorithm to create blocks and classify them as concordant, insertion, or deletion. Adjacent blocks with the same classification and contiguity are combined. Each insertion or deletion block is assigned an ISCN based on unique block-type signatures, where indel size

allowance determined via simulation (Supplementary Table S1, Supplementary Fig. S1). The system favors interpretations involving single, complex SVs over multiple simpler ones, providing a comprehensive explanation for the observed chromosomal deviations (Supplementary Section S1.6).

Report. Based on the interpreted SVs, disrupted genes that are present in the DDG2P database[28] are reported. For balanced SVs, we looked at the boundaries within a resolution of 5 Kbp for disrupted genes that might lead to a loss of function (resolution determined empirically with simulations, Supplementary Section S4). For unbalanced SVs, we looked at the entire affected regions, for either gain or loss of gene product. Lastly, the allelic (monoallelic/biallelic) and mutational (loss/gain/altered gene product) requirements are used to filter for disrupted gene output and phenotype prediction.

An HTML report is compiled for ease of reading. It includes the decomposed paths (chromosomes), the corresponding visualization of the chromosome in both cytoband and segment views (Supplementary Section S1.7), the interpreted SVs under the ISCN language, and the disrupted developmental genes.

2.2 KarSim module for simulating karyotypes

The KarSim module (Supplementary Section S2.1, Supplementary Fig. S2) generates a Molecular karyotype file, a FASTA file, and a history log of events for downstream use, including KarCheck comparisons, while also allowing for the simulation of different sequencing technologies.

Usage in Simulation tests. Random karyotypes were generated to simulate a common genetic disorder from Decipher Database's CNV syndromes[2], followed by 7 to 14 random de-novo SVs from Supplementary Table S2. Certain genomic regions, including centromeres and telomeres, were masked during the analysis to ensure accurate structural variant placement. More details on the masked regions can be found in the Supplementary Section S2.1. SVs were placed with breakpoints at least 50 Kbp from masked regions, ensuring no segments smaller than 50 Kbp were generated.

After generating the parameterized-random Molecular karyotypes, simulated data was processed with OMSim[29] to generate OGM molecules with added noise. The standard Bionano Solve pipeline (v3.7)[25] was applied to compute CNVs, SVs, and contig alignments, which were then used as input for OMKar to reconstruct the final virtual karyotype. Full details on the simulation process and parameters can be found in the Supplementary Section S2.3.

2.3 The KarCheck module for comparing karyotypes

karyotypes from the simulation (K_t) are compared with reconstructed karyotypes (K_r) using the KarCheck module (Supplementary Section S3, Supplementary Fig. S2). Preprocessing is initially applied to K_t and K_r to divide chromosome groups into clusters. To achieve comparability, segments are further divided so both karyotypes share the same set of segments (Supplementary Section S3.1).

For each chromosome cluster, three metrics are reported: 1) chromosome count concordance, 2) Jaccard similarity of SV edges, and 3) Jaccard similarity of CN.

SV similarity computation. SV similarity computation is performed by comparing non-segment edges in the pair of chromosome clusters. The edges are matched (allowing for some tolerance, analyzed in Supplementary Section S4), and a Jaccard similarity score (Intersection over Union) is computed to measure similarity (Supplementary Section S3.2). OGM reads do not map with high confidence in the telomere, acrocentric p-arm, and acrocentric centromere regions. Therefore, these prefix/suffix regions were excluded in simulations and during SV similarity computations.

Copy Number similarity comparison and metrics. CN similarity comparison is done by binning the whole genome (excluding prefix/suffix masked region) into spanning, non-overlapping bins of 50 Kbp, with a tolerance of +/- 100 bp (exact size chosen to maximize the size of the last bin on the chromosome). Each bin is used to store the average CN within that region, and a bin is termed "with CNV" if deviates more than 0.05 from diploid (chosen based on OGM's resolution). A Jaccard similarity is computed between the bins with CNV. (Supplementary Section S3.3).

3 Results

3.1 OMKar runs efficiently on a standard desktop

We tested the tool's performance on 154 clinical samples and 38 simulated datasets using a standard linux machine (Intel(R) Xeon(R) CPU X5680 @ 3.33GHz, 128 GB of RAM, and running Ubuntu 16.04.6 LTS.) OMKar was very efficient with a median runtime of 8.4s (range [6.2-26.1s]; Supplementary Fig. S3). Including the time for image generation required for the html output, the median runtime increased to 21.3s (range 15.0-48.5s). The runtime was correlated with the number of rearrangements (breakpoints).

3.2 OMKar reconstructs karyotypes in simulated data with high accuracy

A total of 552 structural variations were simulated on 38 karyotypes at 100X coverage. The 38 karyotypes could be grouped into 803 chromosome clusters. 299 of 803 clusters had at least one SV. Of the remaining 504 'non-event' clusters, only 6 were reconstructed with an SV (false-positive), yielding a true-negative rate of 98.8%. We observed that the Bionano variant calling pipeline had a lower accuracy of 42.7% in capturing terminal SVs, which were simulated in the peri-telomeric regions (Supplementary Section S5, Supplementary Table S3). Therefore, in the following, we focused on clusters that contain non-terminal SV, represented in 250 of 299 chromosomal clusters.

We first tested if OMKar could estimate the number of chromosomes correctly. 14 aneuploidies– a gain or loss of a chromosome–where simulated in 9 of the 250 chromosome clusters with events,

and OMKar correctly reconstructed 13 of them. OMKar reconstructed a normal number of chromosomes in 229 of the remaining 241 clusters (6 FP in the 208 clusters without terminal events, each with three or more balanced translocations; 6 FP in the 33 clusters containing terminal events, due to arm deletion).

For each cluster containing non-terminal SVs, we computed the Jaccard similarity (intersection over union) of the non-terminal SVs in the simulated and predicted clusters. The average Jaccard Similarity across the 250 clusters was 84.8%, (recall 94.7%; precision 87.5%), suggesting a high-quality karyotype reconstruction. To quantify performance according to the relative difficulty of the simulation, we estimated the *complexity* of each cluster as the number of breakpoint-edges from non-terminal SVs. We denoted clusters with complexity score ≤ 6 as being *low-complexity*, and *high-complexity* otherwise. As expected, the performance on the low-complexity clusters (Fig. 2a; Jaccard 89.9%; Recall 95.6%; Precision 92.2%) was much better than on high-complexity clusters (Jaccard 80.8%; Recall 89.9%, precision 85.8%). The CNV comparison metric behaved similarly with high average Jaccard Similarity across the 250 clusters at 96.0% and some degradation in performance from low to high complexity (Fig. 2b). We note that the clinical cases described later (Section 3.3) were overwhelmingly of low-complexity (Fig. 2c).

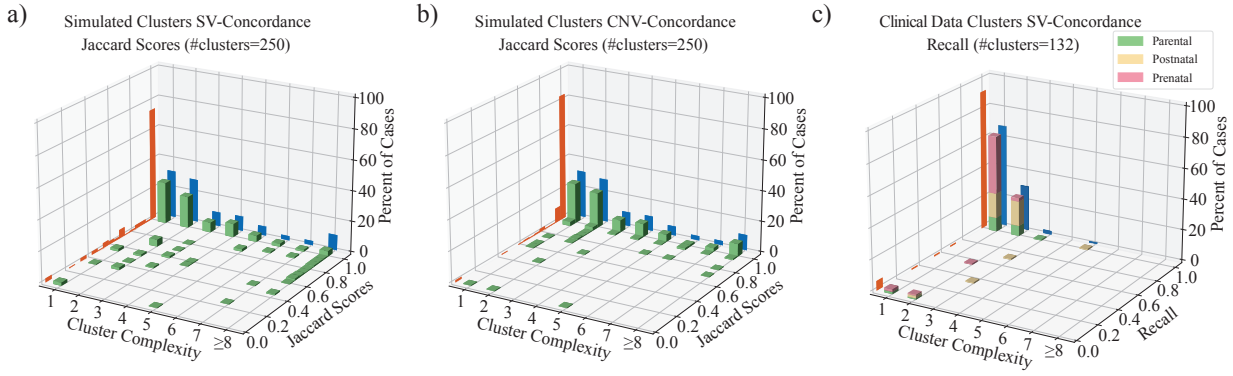


Figure 2: **Validation statistics on simulated and clinical karyotypes**. Each plot displays a 3-d histogram of Jaccard score (or Recall) of clusters in the sample group. The three axes separate clusters based on complexity, frequency of observations, and Jaccard score or Recall. The frequencies of specific Jaccard scores are displayed in the orange projection, showing that the vast majority of samples have high Jaccard score or Recall. The frequencies of cluster complexity are shown in blue and reveal that the cluster complexity of simulated cases is generally higher than clinical data. (a) Jaccard score of SV edges in simulations; (b) Jaccard score of CNV calls in simulations; (c) Recall of SV calls on 132 clusters from prenatal, postnatal and parental clinical samples.

In addition, we measured accuracy by directly investigating the SV edges (breakpoints) in the chromosome clusters (Table 2). A total of 839 SV edges were introduced in simulations, with 502 in low-complexity clusters. The overall accuracy was high: 96% for low-complexity and 89% for high-complexity clusters, where the occurrence of closely spaced SV edges created challenges. The accuracy also varied depending on the SV type. For example, OMKar was successful in catching balanced translocations but had lower accuracy for Tandem Duplications and Duplication Inver-

Simulated Non-Terminal SV Edge Type	Low Complexity Simulated/Identified	High Complexity Simulated/Identified
Deletion	74/70 (94.6%)	12/11 (91.7%)
Inversion	100/96 (96.0%)	18/18 (100.0%)
Duplication Inversion	94/93 (98.9%)	52/45 (86.5%)
Tandem Duplication	56/50 (89.3%)	18/14 (77.8%)
Transposition	90/85 (94.4%)	97/88 (90.7%)
Balanced Reciprocal Translocation	88/88 (100.0%)	140/125 (89.3%)

Table 2: **Simulated Non-Terminal SV Edge Recall by Event Types**

sions. These results support the conclusion that OMKar can accurately reconstruct the karyotypes, especially in samples with low rearrangement complexity.

3.3 OMKar reconstructs karyotypes from prenatal, postnatal, and parental screenings with high accuracy

We applied OMKar to OGM data acquired from 154 samples (50 prenatal, 41 postnatal, and 63 parental) prepared at ten different sites (Supplementary Table S4). Seven postnatal samples were biological replicates of 3 individuals, including samples that were mapped at different test sites. For each sample, a previous diagnosis of constitutional abnormality had been made using combinations of traditional cytogenetic methods of karyotyping, CMA, and FISH. These methods are not comprehensive, but they have high precision, so we first checked if OMKar could correctly reconstruct previously detected variations.

The union of calls from karyotyping, CMA, and FISH revealed 141 variations in 154 samples. OMKar was able to fully reconstruct 129 (91%) of the 141 variations including 25/25 (100%) aneuploidies, 32/32 (100%) balanced reciprocal translocations, 38/39 (97.4%) deletions, 32/38 (84.2%) amplifications, 1/3 (33.3%) unbalanced translocations. OMKar did not detect 1 inversion, 1 Robertsonian translocation, and 1 isodicentric chromosome. These 146 structural variations formed 132 chromosome clusters, where reconstructions of 121 (91.7%) were fully concordant. Some of the missed karyotypes were due to SVs being masked (3 clusters), duplication size below OMKar threshold (2 clusters), Robertsonian Translocation (1 cluster), and isodicentric chromosome (1 cluster; Fig. 2c, Supplementary Table S5).

Importantly, OMKar improved upon every other technology when considered in isolation (Supplementary Fig. S4). On the 65 samples where karyotyping was performed, it detected only 56 (64%) of the 87 SVs. Similarly, CMA was applied on 76 samples and detected 94 (88%) of the 107 SVs; FISH was applied on 16 samples, and detected 11 (58%) of the 19 SVs. Specifically, karyotyping mostly captured large rearrangements, whereas CMA and FISH mostly captured unbalanced events.

We tested OMKar consistency using biological replicates prepared and mapped at different test sites. Specifically, one postnatal sample was processed at six different test sites, and two postnatal samples were each processed at two test sites (Supplementary Table S4). In all cases, OMKar successfully reconstructed the correct karyotype.

Importantly, OMKar reconstructed additional SVs not caught by any of the other techniques. Specifically, after filtering out lower quality SVs (Section S1.1), OMKar detected 436 deletions, 506 amplifications and 67 inversions, averaging 2.8 deletions, 3.3 amplifications, and 0.44 inversions as novel events per sample. These discoveries need to be experimentally validated. However, coverage support for novel SVs was similar to simulations, where OMKar achieved a precision of 88%.

3.4 OMKar correctly reconstructs variations with partially missing calls

Ideally, unbalanced rearrangements are supported by both SV edges and by CNVs. However, the CNV call might be missed if the region is small, and SV call might be missed if the breakpoints lay in regions of low-complexity that are masked by the Bionano pipeline. OMKar reconstructs karyotypes with rearrangements that are supported only by SVs or only by CNVs. Specifically, it infers missing SVs by adding additional edges to make the breakpoint graph Eulerian (Methods). OMKar outputs inferred variants as lower confidence karyotype features.

We reanalyzed 40 deletions and 42 amplifications detected by OMKar that were cross-validated using complementary technologies. Of the deletions, 12 (30%) were reconstructed using only CNV calls, and 4 (10%) deletions were reconstructed using only SV calls. Among the amplifications, 15 (36%) amplifications were reconstructed using only CNV calls.

In prenatal sample 205, an inter-chromosomal duplicated insertion was previously reported using a combination of karyotyping and CMA: ins(14;2)(q32;q36.1q31.2) (105.159M; 221.205M-178.043M). OGM reported a high confidence inter-chromosomal SV call of Chr14: 105.159M to Chr2: 221.205M, with the correct orientation, but the other inter-chromosomal SV call was missing. It also reported a CN gain of the duplicated region of Chr2. OMKar correctly inferred the missing SV call using support from the other SV and CNV calls, and was able to automatically reconstruct this rearrangement.

3.5 OMKar identifies the genetic basis of previously diagnosed phenotype

The OGM samples were generated based on different usage modalities (Supplementary Table S4). Prenatal testing in 50 samples was performed, either because of an abnormality detected by non-invasive screening (44 samples) or because of elevated risk due to family history or advanced maternal age. In contrast, 28 of 34 unique postnatal samples presented with a clinical phenotype. The 63 parental samples contained individuals who had experienced miscarriage, had a higher probability of translocation, or suspicion of infertility.

Among the pre- and postnatal samples, 20 had a previously diagnosed Genotype-to-Phenotype (G2P) mechanism, largely obtained through a manual analysis of CMA and karyotyping. OMKar automatically identified all 20 G2P mechanisms through a correlation between reconstructed karyotypes and an intersection with the DDG2P database (Methods). Interestingly, these were all aneuploidies, with phenotypes including Triple-X, Jacobs, Turner Syndrome, and Down Syndrome. In contrast, for the (largely asymptomatic) parental samples, OMKar automatically and correctly reconstructed 9 of 10 translocations (missed one Robertsonian) to explain eight infertility cases

and one clinically remarkable child. The last case (ID:1999) highlighted the importance of parental karyotyping using OMKar. The sample was unremarkable in cytogenetic karyoptying but carried a balanced translocation between the p-arms of Chr4 and Chr7, t(4;7)(3,903,798;6,881,853). This balanced translocation resulted in the inheritance of an unbalanced translocation. The child carried a deletion of Chr4 0 - 3.9Mbp, causal for the Wolf-Hirschhorn Syndrome (deletion of Chr4:1.6Mbp - 2.1Mbp)[30,31].

Apart from the 10 cases, OMKar also identified a deletion, del(X) (31,614,556-31,831,572) in one parent (ID:19), which causes a monoallelic loss of the gene *DMD*, leading to Becker muscular dystrophy (BMD)[32]. BMD can be inherited, and while the deletion was likely not causal for the observed infertility, it may have contributed in combination with other un-diagnosed factors.

All of these confirmatory diagnoses either involved large CNVs that could be resolved by CMA, or large translocation events that could be detected by karyotyping. We next investigated OMKar's capacity to detect smaller SVs.

3.6 OMKar reconstruction explains genetic basis of postnatal phenotypes

In addition to the larger variants discussed previously, OMKar reported 188 balanced (copy-neutral) SVs and 454 unbalanced SVs in 21 postnatal samples with missing G2P explanations. Importantly, OMKar also provided novel G2P explanations for 5 of 21 samples(Table 3).

Two samples showed unbalanced events with a loss of genes important for neurodevelopment. For sample 2081, it was a deletion call (10.4 Mbp). The karyotype for sample 2280 was more complex, with a deletion (5.1 Mbp) in the middle of a translocated segment (segment 4c, Fig. 3A-C). Previously, only karyotyping had been performed for both cases, and neither deletion was detected.

Sample	Clinical Indication	Clinical Diagnosis	OMKar Genotype	OMKar G2P Prediction
2081	Absent teeth, ID	ID	(new) PAX8 deletion	Congenital Hypothyroidism non-goiterous type 2
2280	Mild dysmorphic features, hypotonia, DD	Kabuki Syndrome; Unknown	(new) deletion in the middle of translocation, WDFY3 + HNRNPD	Primary Microcephaly or macrocephaly with developmental delay; HNRNPD-related developmental disorder (monoallelic)
2276	Seizures, ID	Unknown	translocation interrupts TANC2	TANC2-related neurodevelopmental and psychiatric disorders
2281	DD	DD	transposition interrupts MBD5	EHMT1-like ID
2282	ID	ID	translocation interrupts SOX5	12P12.5 Intragenic deletions associated with ID

Table 3: **OMKar G2P explanations for undiagnosed postnatal cases.** ID: Intellectual Disability, DD: Developmental Delay.

Among 13 balanced events reconstructed by OMKar in these 5 samples, boundaries in 3 samples (2276, 2281, and 2282, see Table 3) interrupted a neurodevelopmental gene. Because precise coordinates of translocation were not reported via karyotyping or CMA, a genotypic basis was not previously diagnosed. Sample 2281 in particular, illustrates the power of OMKar's reconstruction

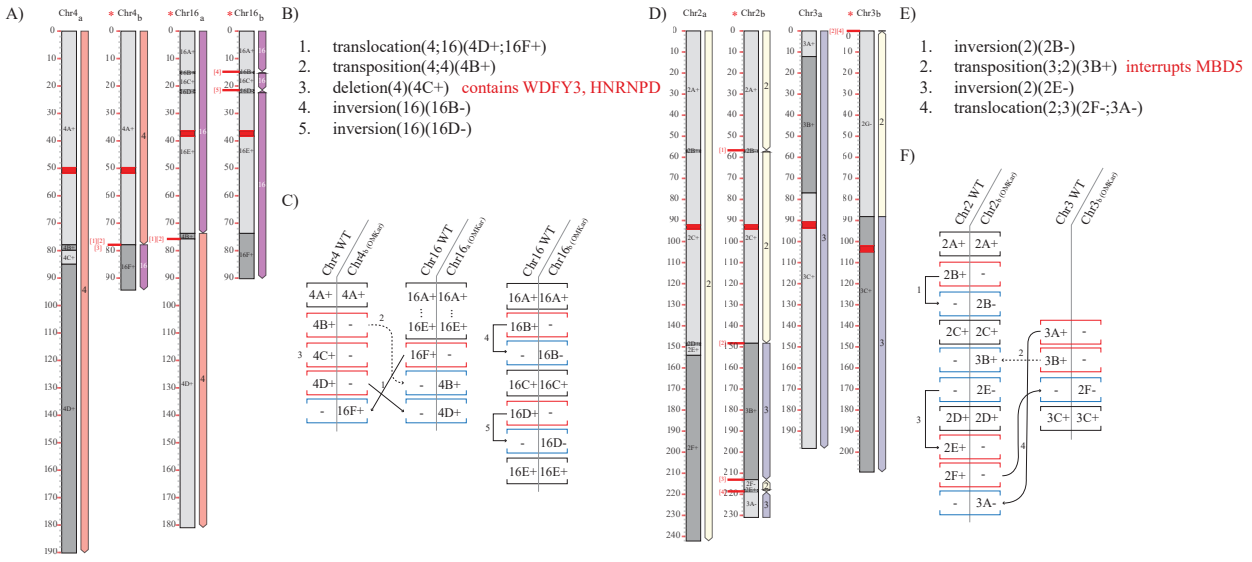


Figure 3: **OMKar reconstructions in two postnatal samples: 2280 (panels A-C) and 2281 (panels D-F)**. The karyograms (panels A, D) and the ISCN-formatted description (panels B, E; slightly altered for exposition) were automatically generated by OMKar. The karyograms displayed show the 'segment view' for easier referencing. Panels C, F describe the SV interpretation process after path decomposition, with black brackets indicating concordant blocks, red indicating deletion blocks, and blue indicating insertion blocks.

of a complex karyotype (Fig. 3D-F). The reconstruction revealed a transposition of a chr3 segment on to chr2, interupting the MBD5 gene between segments 2C and 2D. The karyotype additionally included two inversions and a translocation resulting in a highly rearranged chromosomal cluster with no change in copy number.

4 Discussion

Karyotyping remains an essential tool in the diagnosis of constitutional genetic disorders, particularly those arising from large chromosomal rearrangements such as aneuploidies, translocations, and complex structural variants. While traditional methods such as cytogenetic karyotyping, FISH, and microarray have long served as the standard for detecting these abnormalities, they are constrained by limited resolution, manual labor intensity, and an inability to detect balanced rearrangements and novel variations with high precision. In contrast, genomic technologies (whole exome/genome sequencing) are very precise, but do not easily provide chromosome level characterizations. Long-read technologies are currently too expensive for clinical use. Optical Genome Mapping (OGM) thus represents a happy medium, offering a medium-resolution, robust alternative capable of detecting a broader range of structural variations (SVs) in clinical settings[33,34]. Recent clinical studies have shown OGM have high concordance (99.5%) with standard-of-care (SOC) methods over 1,000 samples, with increased detection rate of pathogenic or likely-pathogenic variants[23,35]. OGM is also recently incorporated into the International System for Human Cytogenomic Nomenclature (ISCN)[36]. For these reasons, we developed our tool starting with OGM data.

Our method, OMKar, bridges the gap between low-resolution techniques like cytogenetics and high-resolution sequencing methods by capturing large-scale rearrangements, but also combining the information into an automated karyotype inference. It identifies key structural abnormalities such as balanced translocations, inversions, and duplications. The ability to automate karyotyping through OMKar not only reduces the manual workload but also enhances the speed and scalability of the analysis. This enables clinicians to analyze large datasets, improving diagnostic accuracy and potentially leading to faster treatment decisions.

In developing OMKar, we faced a significant challenge of resolving conflict between SV and CNV calls. Such conflicts were caused by either having one of the calls with high confidence while the other call was missed or masked or when SV and CNV call boundaries were not identical. OMKar is designed to infer variations with partially missing or conflicting calls and resolve boundaries. Future developments in OGM technology and in algorithmic reconstruction should reduce these conflicts, resulting in higher confidence calls.

OMKar (and OGM), while highly effective for most structural variants, shows reduced sensitivity in detecting mosaic chromosomal abnormalities, events occurring in regions of low complexity and segmental duplications that can lead to non-allelic recombination such as Robertsonian translocations. Mosaicism, characterized by variation in chromosomal numbers within different cell populations, also pose a challenge for OGM, which is primarily focused on large, stable genomic rearrangements. Further refinement of OGM technology and the tools will be needed to broaden its applicability in clinical settings.

OMKar showed a performance gap between terminal and non-terminal structural variation. Previous results from FISH screening suggest that a small number (5-10%) of developmental disorders that lead to intellectual disability are due to "cryptic telomeric rearrangements" [37]. However, this may be an underestimate because subtelomeric rearrangements are often seen as *de novo* variants[38]. Our future research will focus on algorithms for identifying telomeric abnormalities, including ring chromosomes[39].

OGM technologies cannot currently detect variations within centromeres or the short arms of acrocentric chromosomes[25,40]. In particular, Robertsonian translocations involve rearrangements of the short-arms of acrocentric chromosomes are not currently detected by OGM. Recent studies have suggested the use of long-read technologies like Oxford Nanopore for detecting them[39]. Because the core of OMKar algorithm, which includes the building of Eulerian graphs followed by path extraction, is agnostic of a specific sequencing technology, OMKar should be easily adapted to other technologies. As more datasets describing these events are made available on different sequencing platforms, we plan to develop karyotyping tools for those platforms.

In conclusion, OMKar has demonstrated significant potential in automating and improving the accuracy of karyotyping using OGM data. It offers a robust, scalable, and high-resolution approach to detecting constitutional genetic abnormalities, though ongoing improvements are necessary to fully address its limitations. As the tool continues to evolve, it could become an increasingly important method for research and clinical diagnostics, complementing and potentially surpassing

traditional methods in terms of accuracy and efficiency.

Ethics Review Statement

This study involved human participants and was approved by the Ethics Committees of the Medical University of Vienna (ethical code: 2229/2019) and Keçiören Teaching and Research Hospital (ethical code: 2012-KAEK-15/2083). The study adhered to the principles outlined in the Declaration of Helsinki. Informed consent was obtained from all participants prior to their inclusion.

Additionally, the study was conducted in accordance with the Declaration of Helsinki and received approval from the Institutional Review Boards of Western IRB – Copernicus Group (WCG) under study numbers 20203726 and 20212956. This approval included provisions for informed consent or waived authorization for the use of de-identified, banked samples for research purposes. All protected health information (PHI) was removed, and data were anonymized (coded and double-blinded) before accessioning for the study.

Acknowledgments

This work was supported by the National Institutes of Health (NIH) grant R01GM114362. VB is a co-founder and scientific advisory board member of Boundless Bio, Inc. (BBI) and Abterra Inc., holding equity in both companies. BBI and Abterra were not involved in this research.

We thank Gautam Kathir for developing the initial HTML report and Christopher Day for discussions on database-related aspects. We also acknowledge the authors of the two multicenter studies (PMID: 36828597, 38211722, 39032820) for providing 98 clinical OGM datasets used in the evaluation of OMKar. Some de-identified data used in this study originated from a study sponsored by Bionano Genomics, specifically from the "Validation of Optical Genome Mapping for the Identification of Constitutional Genomic Variants in a Postnatal Cohort" study (NCT05295277; `https://clinicaltrials.gov/study/NCT05295277?term=Optical%20Genome%20Mapping&rank=6`).

References

[1] Stankiewicz, P. & Lupski, J. R. Structural variation in the human genome and its role in disease. *Annual Review of Medicine* **61**, 437–455 (2010).

[2] Firth, H. V. *et al.* DECIPHER: Database of Chromosomal Imbalance and Phenotype in Humans Using Ensembl Resources. *Am J Hum Genet* **84**, 524–533 (2009).

[3] Bianchi, D. W. *et al.* Dna sequencing versus standard prenatal aneuploidy screening. *New England Journal of Medicine* **370**, 799–808 (2014).

[4] Taylor-Phillips, S. *et al.* Accuracy of non-invasive prenatal testing using cell-free dna for detection of down, edwards and patau syndromes: a systematic review and meta-analysis. *BMJ Open* **6**, e010002 (2016).

[5] Miller, D. T. *et al.* Consensus statement: chromosomal microarray is a first-tier clinical diagnostic test for individuals with developmental disabilities or congenital anomalies. *American Journal of Human Genetics* **86**, 749–764 (2010).

[6] Shinawi, M. & Cheung, S. W. The array cgh and its clinical applications. *Drug Discovery Today* **13**, 760–770 (2008).

[7] Pergament, E., Chen, P. X., Thangavelu, M. & Fiddler, M. The clinical application of interphase FISH in prenatal diagnosis. *Prenat Diagn* **20**, 215–220 (2000).

[8] de Moraex-Malinverni, A. C. *et al.* Application of fluorescence in situ hybridization (FISH) as a tool to aid cytogenetics in 1,409 fetal samples. *Clin Exp Obstet Gynecol* **43**, 685–690 (2016).

[9] Shaffer, L. G. & Bejjani, B. A. A cytogeneticist's perspective on genomic microarrays. *Hum Reprod Update* **10**, 221–226 (2004).

[10] Chang, Y. W. *et al.* Balanced and unbalanced reciprocal translocation: an overview of a 30-year experience in a single tertiary medical center in Taiwan. *J Chin Med Assoc* **76**, 153–157 (2013).

[11] Chantot-Bastaraud, S., Ravel, C. & Siffroi, J. P. Underlying karyotype abnormalities in IVF/ICSI patients. *Reprod Biomed Online* **16**, 514–522 (2008).

[12] Dai, P. *et al.* Evaluation of optical genome mapping for detecting chromosomal translocation in clinical cytogenetics. *Mol Genet Genomic Med* **10**, e1936 (2022).

[13] Nilius-Eliliwi, V., Gerding, W. M., Schroers, R., Nguyen, H. P. & Vangala, D. B. Optical Genome Mapping for Cytogenetic Diagnostics in AML. *Cancers (Basel)* **15** (2023).

[14] Balducci, E., Kaltenbach, S., Villarese, P. *et al.* Optical genome mapping refines cytogenetic diagnostics, prognostic stratification and provides new molecular insights in adult

mds/aml patients. *Blood Cancer Journal* **12** (2022). URL `https://doi.org/10.1038/s41408-022-00718-1`.

[15] Raeisi Dehkordi, S., Luebeck, J. & Bafna, V. Fandom: Fast nested distance-based seeding of optical maps. *Patterns* **2**, 100248 (2021). URL `https://doi.org/10.1016/j.patter.2021.100248`.

[16] Li, L., Leung, A. K., Kwok, T. P. *et al.* Omsv enables accurate and comprehensive identification of large structural variations from nanochannel-based single-molecule optical maps. *Genome Biology* **18**, 230 (2017). URL `https://doi.org/10.1186/s13059-017-1356-2`.

[17] Mantere, T. *et al.* Optical genome mapping enables constitutional chromosomal aberration detection. *Am J Hum Genet* **108**, 1409–1422 (2021).

[18] Dremsek, P. *et al.* Optical Genome Mapping in Routine Human Genetic Diagnostics-Its Advantages and Limitations. *Genes (Basel)* **12** (2021).

[19] Sahajpal, N. S., Barseghyan, H., Kolhe, R., Hastie, A. & Chaubey, A. Optical Genome Mapping as a Next-Generation Cytogenomic Tool for Detection of Structural and Copy Number Variations for Prenatal Genomic Analyses. *Genes (Basel)* **12** (2021).

[20] Goenka, S. D. *et al.* Accelerated identification of disease-causing variants with ultra-rapid nanopore genome sequencing. *Nat Biotechnol* **40**, 1035–1041 (2022).

[21] Levy, B. *et al.* A framework for the clinical implementation of optical genome mapping in hematologic malignancies. *American journal of hematology* **99**, 642–661 (2024).

[22] Ghabrial, J. *et al.* Diagnostic and prognostic/therapeutic significance of comprehensive analysis of bone and soft tissue tumors using optical genome mapping and next-generation sequencing. *Modern Pathology* 100684 (2024).

[23] Iqbal, M. *et al.* Multisite Assessment of Optical Genome Mapping for Analysis of Structural Variants in Constitutional Postnatal Cases. *The Journal of Molecular Diagnostics* **25**, 175–188 (2023).

[24] Xiao, B., Luo, X., Liu, Y. *et al.* Combining optical genome mapping and rna-seq for structural variants detection and interpretation in unsolved neurodevelopmental disorders. *Genome Medicine* **16** (2024). URL `https://doi.org/10.1186/s13073-024-01382-9`.

[25] BionanoGenomics. Bionano Solve Theory of Operation: Structural Variant Calling (2018). URL `https://bionanogenomics.com/wp-content/uploads/2018/04/30110-Bionano-Solve-Theory-of-Operation-Structural-Variant-Calling.pdf`.

[26] Alekseyev, M. A. & Pevzner, P. A. Breakpoint graphs and ancestral genome reconstructions. *Genome Research* **19**, 943–957 (2009).

[27] Schrijver, A. *Theory of linear and integer programming* (John Wiley & Sons, 1998).

[28] Thormann, A. *et al.* Flexible and scalable diagnostic filtering of genomic variants using G2P with Ensembl VEP. *Nat Commun* **10**, 2373 (2019).

[29] Miclotte, G. *et al.* OMSim: a simulator for optical map data. *Bioinformatics* **33**, 2740–2742 (2017).

[30] Wolf, U., Reinwein, H., Porsch, R., Schröter, R. & Baitsch, H. Deficiency on the short arms of a chromosome No. 4. *Humangenetik* **5**, 397–413 (1965).

[31] Zollino, M. *et al.* Mapping the Wolf-Hirschhorn syndrome phenotype outside the currently accepted WHS critical region and defining a new critical region, WHSCR-2. *Am J Hum Genet* **72**, 590–597 (2003).

[32] Ervasti, J. M., Ohlendieck, K., Kahl, S. D., Gaver, M. G. & Campbell, K. P. Deficiency of a glycoprotein component of the dystrophin complex in dystrophic muscle. *Nature* **345**, 315–319 (1990).

[33] Smith, A. C., Neveling, K. & Kanagal-Shamanna, R. Optical genome mapping for structural variation analysis in hematologic malignancies. *Am J Hematol* **97**, 975–982 (2022).

[34] Valkama, A. *et al.* Optical genome mapping as an alternative to fish-based cytogenetic assessment in chronic lymphocytic leukemia. *Cancers* **15** (2023). URL `https://www.mdpi.com/2072-6694/15/4/1294`.

[35] Broeckel, U. *et al.* Detection of Constitutional Structural Variants by Optical Genome Mapping: A Multisite Study of Postnatal Samples. *J Mol Diagn* **26**, 213–226 (2024).

[36] Hastings, R. J., Moore, S. & Chia, N. (eds.) *ISCN 2024: An International System for Human Cytogenomic Nomenclature (2024)* (Karger, Basel, 2024). URL `https://karger.com/books/book/6011/ISCN-2024An-International-System-for-Human`.

[37] Moeschler, J. B. & Shevell, M. Clinical genetic evaluation of the child with mental retardation or developmental delays. *Pediatrics* **117**, 2304–2316 (2006).

[38] Luo, Y. *et al.* Diverse mutational mechanisms cause pathogenic subtelomeric rearrangements. *Hum Mol Genet* **20**, 3769–3778 (2011).

[39] Mostovoy, Y. *et al.* Resolution of ring chromosomes, Robertsonian translocations, and complex structural variants from long-read sequencing and telomere-to-telomere assembly. *Am J Hum Genet* **111**, 2693–2706 (2024).

[40] BionanoGenomics. Bionano System Application Specifications (2024). URL `https://bionano.com/wp-content/uploads/2023/08/CG-00008-Bionano-Saphyr-System-Application-Specifications.pdf`.

[41] RaeisiDehkordi, S. Omkar. `https://github.com/siavashre/OMKar` (2024).

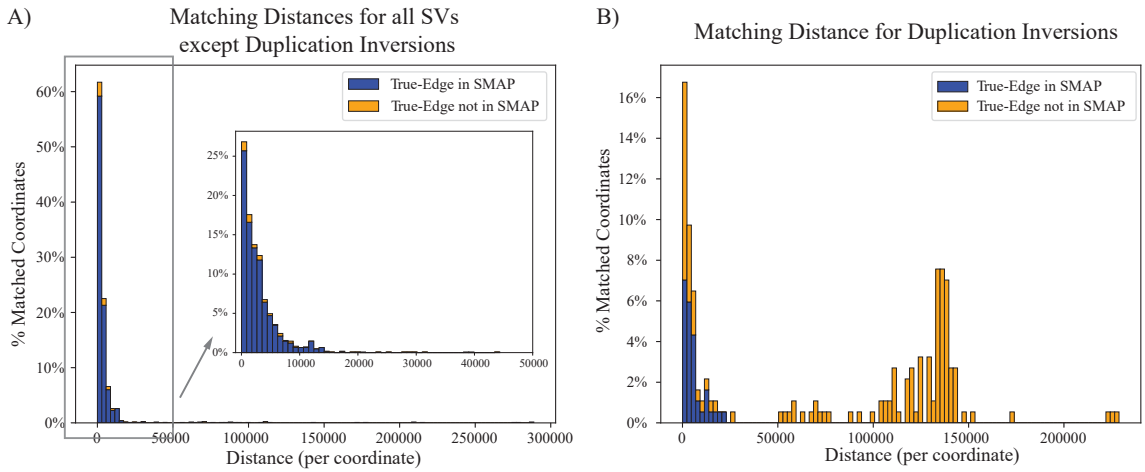
Supplementary Figure Captions

Supplementary Figure S1: **Matching distances per breakpoint coordinate**. The breakpoint accuracy of OMKar is analyzed by comparing OMKar reconstructions against the simulated truth karyotypes. The KarCheck SV-edge matching breakpoint distance allowance was relaxed to 300 Kbp to generate these plots. For each SV, the distance for each of the two breakpoint coordinates is computed separately. For better exposition, the SVs were partitioned into A) all SVs except Duplication inversions and B) duplication inversions.

Supplementary Figure S2: **KarSim and KarCheck modules**. A) Implementation of the KarSim for simulating karyotypes; B) Preprocessing module of KarCheck creates matching breakpoints between two karyotypes, allowing for direct comparisons. C) KarCheck Matching. Segments are annotated by size as being SM (small) or LG (large), and D) the complete simulation and validation pipeline for OMKar.

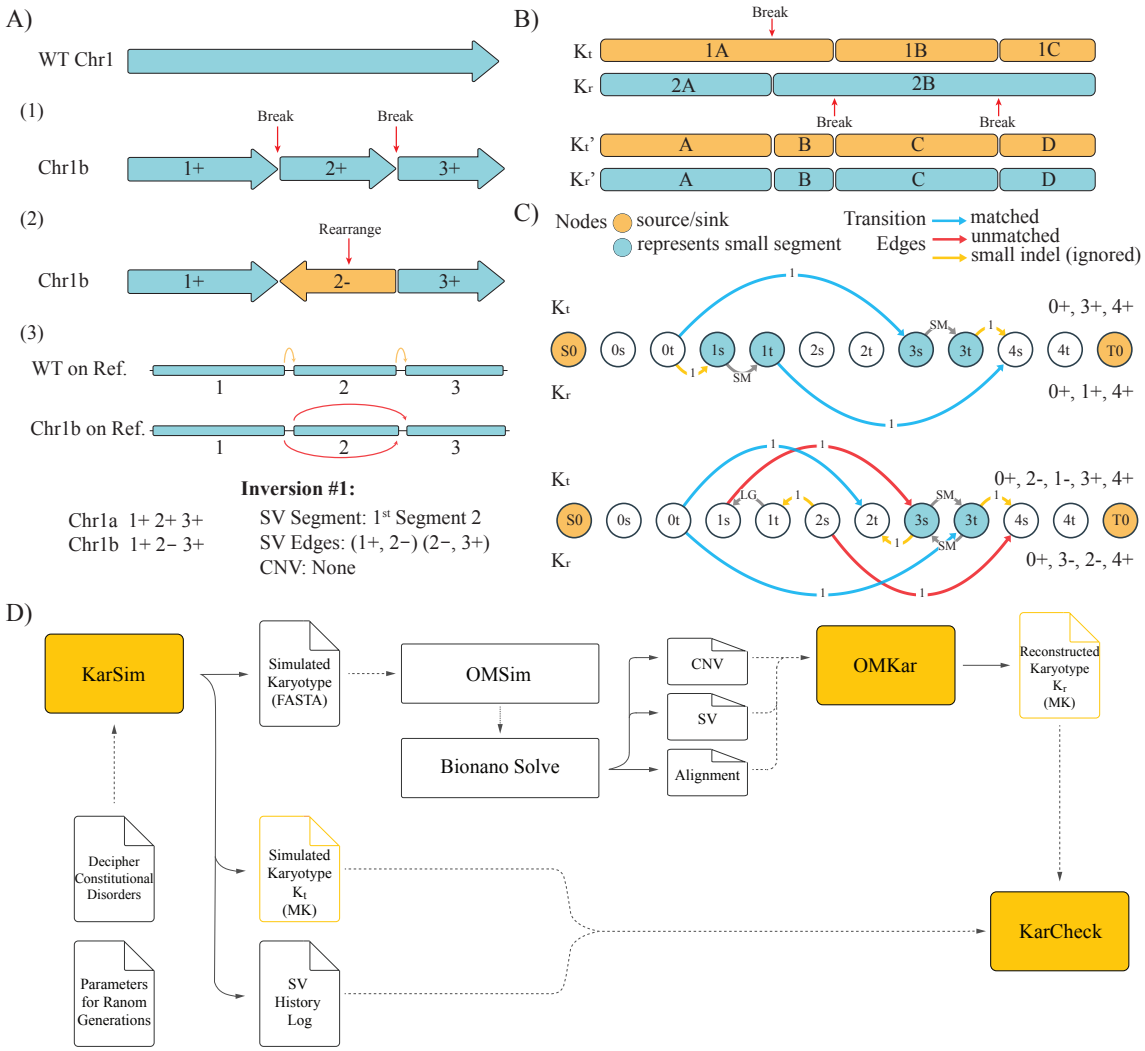
Supplementary Figure S3: **Runtime analysis**. Runtime of OMKar was collected on both simulation and clinical samples A) without rendering the visualization images in HTML report, and B) with rendering the report images.

Supplementary Figure S4: **Validation of OMKar reconstruction against clinical annotation by cytogeneticists using karyotyping, CMA, and/or FISH**. Note that OGM and OMKar was applied to all samples, but not all technology was applied on all samples. Each column represents a clinical sample. The color code in the cells explain the concordance of the applied technology. White: the technology was not applied; Green: concordance; Red: Missed. Variations were grouped into A) aneuploidies, B) balanced structural variations, and C) unbalanced structural variations. Note, the unbalanced structural variations take up two sets of rows.

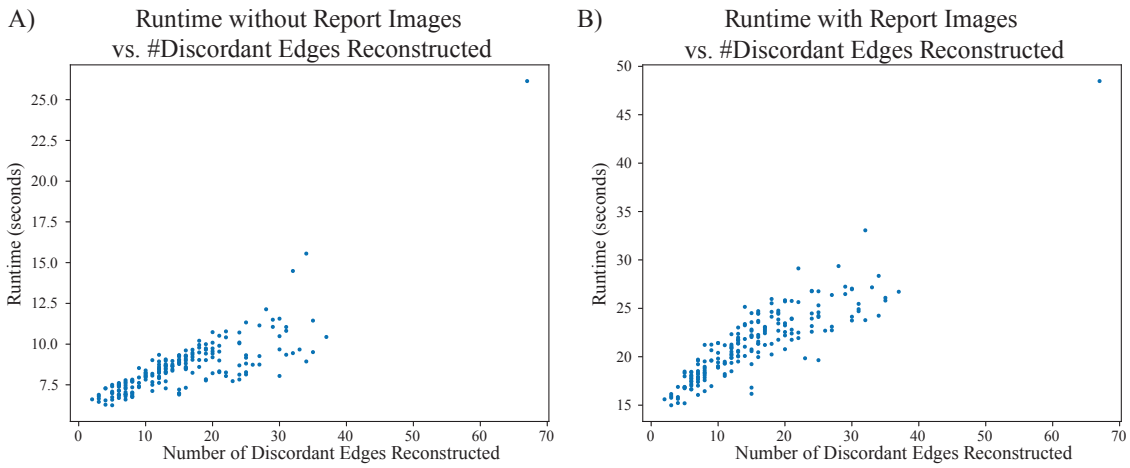


Supplementary Figure S1

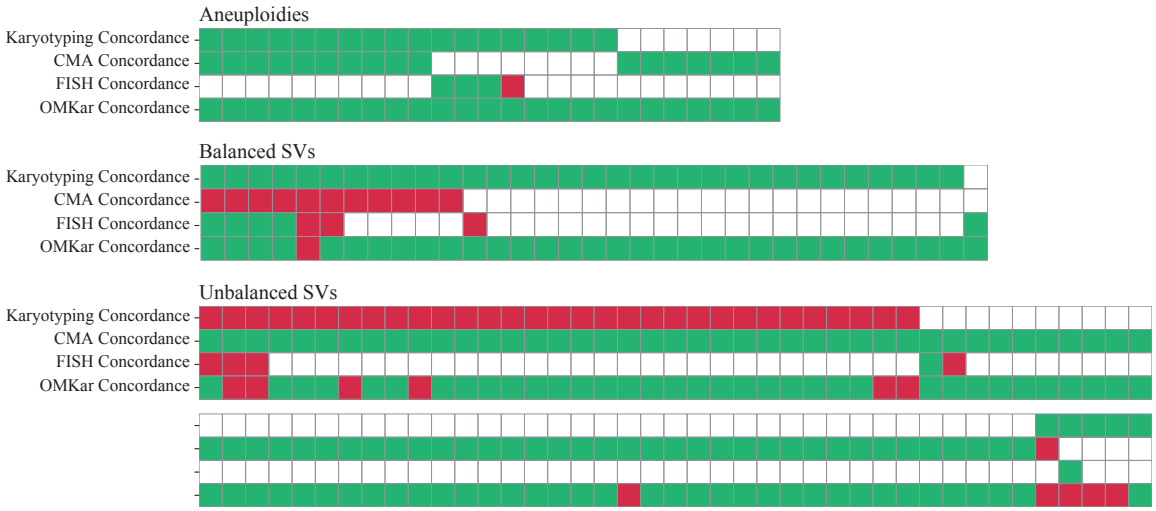
Supplementary Figures



Supplementary Figure S2



Supplementary Figure S3



Supplementary Figure S4

Supplementary Table Captions

Supplementary Table S1: **Block signature for each structural variation**. Variations are listed in the order of finding. INS: insertion, DEL: deletion, CONC: concordant.

Supplementary Table S2: **Simulated structural variations**. (Canonical Structural Variants), where the two WT chromosomes are [1+ 2+ 3+] and [4+ 5+ 6+].

Supplementary Table S3: **Simulated terminal SV edge recall by event types**. Recall of SVs with breakpoint in the peri-terminal region. Different matching allowance was used in the KarCheck module to observe the reconstruction breakpoint distance from the truth SV breakpoint.

Supplementary Table S4: **Clinical sample information**. Includes information on each clinical sample. Sample ID: consistent ID as addressed in the paper, and it also distinguishes sample cohort; Test Site: location the sample was sequenced; Replicate Info: biological replicates of the sample, if applicable; Clinical Phenotype/Risk: clinical documentation of the reason this individual was tested; SOC G2P Prediction: clinical G2P diagnosis based on SOC methods (excluding OGM and OMKar); OMKar G2P Prediction: G2P diagnosis based on OMKar reconstruction; "Method" Performed: wether an SOC testing procedure was performed on this individual.

Supplementary Table S5: **Clinical sample variant information**. Includes information on each structural variation as identified by different techniques. The information includes the following. Cluster ID of the chromosome cluster for the SV; Cluster Chroms lists the chromosome groups within this chromosome cluster; Structural Variation: clinical annotation of this SV, written in ISCN annotation, based on SOC data only. Columns E-H describe the results using Karyotyping, CMA, FISH, and OMKar. Columns I-L describe the concordance of the calls against the annotation. Column N describes reason why OMKar missed this particular variation, if applicable. Column O lists the G2P diagnosis based on OMKar reconstruction.

Supplementary Material

S1 OMKar details

S1.1 Filtering SV and CNV calls

OMKar utilizes the CNV and SV calls bro the Bionano pipeline. We applied several filters to improve data quality. First, we filtered out low-confidence CNV calls, defined as those with a confidence level of 0.95 or below, as well as those located in masked regions that could interfere with the analysis. Next, we excluded CNVs smaller than 200 Kbp. However, if an excluded CNV was supported by a corresponding SV, it was retrieved during later processing, ensuring no relevant variations were missed.

Following these steps, SV calls were filtered based on variation-specific confidence thresholds established by BioNano pipeline: translocations ($T_{trans} = 0.05$), inversions ($T_{inv} = 0.7$), and indels ($T_{indel} = 0$). Additionally, SV calls in masked regions were discarded. Breakpoints from SVs were processed by sorting them based on chromosomal and genomic coordinates and merging adjacent breakpoints within a 50 Kbp window to simplify the breakpoint graph. To ensure accurate representation, CNV segments were split if breakpoints occurred within their boundaries, guaranteeing that all breakpoints exclusively connect the terminal coordinates of segments.

S1.2 Converting to linear constraints

To tackle the inherent non-linearity of the absolute value and sign functions within the optimization problem, we introduce new variables and constraints to linearize these functions.

To convert the sign function into linear constraints, we employed the following approach: The ILP formulation for $s'_e = \text{sgn}(s_e)$ when $s_e \geq 0$ is:

$$\begin{aligned} s_e &\geq s'_e, \\ s_e &\leq 1000 \cdot s'_e \end{aligned}$$

where s'_e is a binary variable. This formulation ensures that $s'_e = 1$ when $s_e > 0$ and $s'_e = 0$ when $s_e = 0$, while maintaining linearity in the constraints. The constant 1000 is used as a sufficiently large value to approximate an upper bound on s_e.

The absolute value function can be linearized $y_v = |x_v|$ as follows:

$$\begin{aligned} y_v &\geq x_v, \\ y_v &\geq -x_v, \\ y_v &\geq 0, \end{aligned}$$

where y_v represents the absolute value of x_v. These constraints ensure that y_v is equal to $|x_v|$ by considering both the positive and negative cases for x_v. Since our objective is to minimize y_v, it naturally corresponds to $|x_v|$. Thus:

$$\text{Objective Function} = \gamma \sum_{v \in V} (c_v + x_v - r_v - \sum_{e \in SV(v)} s_e) + \sum_{v \in V} \beta_v |x_v| - \sum_{s_e \in SV} \alpha_e \text{sgn}(s_e) + \lambda \sum_{v \in V} o_v$$

$$= \gamma \sum_{v \in V} (c_v + x_v - r_v - \sum_{e \in SV(v)} s_e) + \sum_{v \in V} \beta_v y_v - \sum_{s_e \in SV} \alpha_e s'_e + \lambda \sum_{v \in V} o_v$$

S1.3 Parameter determination

In determining α_e, we consider the characteristics of structural variation e. If e signifies a high-confidence call and connects two distinct chromosomes, we set $\alpha_e = 72$ otherwise, $\alpha_e = 9$. A higher value is given to intra-chromosomal translocations to prioritize their preservation in the final karyotype. For the calculation of β_v, we introduce a penalty parameter that becomes more pronounced as segments grow in length. To achieve this, we define $\beta_v = 4 \times \lceil \frac{\text{length}(v)}{5 \times 10^6} \rceil$. This parameter β_v ensures that the penalty for modifying longer segments is appropriately weighted. The parameter γ is empirically determined and set to a value of 5, while λ is fixed at 1 to ensure that if two candidate decompositions are nearly identical, our preference is to select the one with fewer odd vertices. This empirical choice addresses the unique requirements of our approach, providing a balanced framework for the optimization process.

S1.4 OMKar Eulerian Path algorithm

To reconstruct chromosomal structures from breakpoint graphs, OMKar computes an Eulerian path, ensuring each edge is traversed exactly once while maintaining biologically meaningful constraints. It begins with a breakpoint graph, where vertices represent segment boundaries, and edges denote segment continuity, reference adjacencies, or breakpoint rearrangements. To enforce chromosomal structure, the algorithm prioritizes reference edges (minimizing deviation from the reference genome), followed by breakpoint edges (capturing structural variations), and lastly, segment edges. Using a recursive depth-first traversal, it starts from a telomeric vertex—representing a natural chromosomal endpoint—and evaluates edges with a validity function that prevents breaking segment integrity or disconnecting essential graph components. The pseudo-code is given in Algorithm 1.

Algorithm 1: Find Eulerian Tour Starting from Vertex v in Graph G

Data: Graph G

Result: Eulerian Tour

Function `ValidEdge`(*edge* $e(v, w)$, *edge* $e(u, v)$)**:**

- **if** *deg(v)==1* **then**
  - return True
- **else if** $e(v, w)$ *or* $e(u, v)$ *is segment edge* **then**
  - **if** $e(v, w)$ *is not bridge edge* **then**
    - return True
- return False

Function `FindEulerianTour`(*current vertex v, previous vertex u*)**:**

- Add v to the Eulerian tour list E;
- Initialize an empty list W;
- **for** *each edge* $e(v, w)$ **do**
  - **if** `ValidEdge`$(e(v, w), e(u, v))$ **then**
    - Add w to the valid edges list W
- Sort W based on the edge types (E_r, E_b, E_s);
- Pop w from list W;
- remove $e(v, w)$ from G
- `FindEulerianTour`(*vertex w, vertex v*)

Initialize an empty list E;

`FindEulerianTour`(*Telomeric vertex v , -1*);

S1.5 OMKar Report

For ease of use, OMKar optionally compiles a report that includes 1) the observed chromosomal segment lists with reconstructed karyotype, 2) a list of interpreted events using ISCN notation, 3) a visualization of each corresponding chromosome with cytoband and event labels, and 4) a table of important genes that are near the breakpoints of an event or have copy number (CN) alteration. An html version of the report is prepared for easy viewing.

By default, OMKar outputs a Molecular karyotype in a text format to unambiguously describe the karyotype as follows: It first lists all defined segments across the reference genome. For each segment, the following information is provided: segment number, chromosome, start and end coordinates, and the graph nodes representing the segment. Each segment is represented by two nodes, connected by a segment edge. All segments are forward-oriented (i.e., the end coordinate is greater than or equal to the start coordinate) and are sorted by chromosome groups (from 1 to Y) and increasing coordinates. The segments are non-overlapping, with no gaps between them, although telomeric regions of the reference genome may be excluded.

Following the segment definitions, OMKar reports the reconstructed paths that represent a karyotype. Each path consists of a list of segments in the format "Path number = segment number followed by direction." The segments are traversed either in the forward ('+') or reverse ('-') direction, where '+' indicates traversal from the start to the end of the segment, and '-' indicates

traversal from the end to the start of the segment. For example, "Path1 = 1+ 2+ 3-" means that the path traverses segment 1 in the forward direction, segment 2 in the forward direction, and segment 3 in the reverse direction. Additionally, the number of centromeres present in each path is reported, which should ideally be one, indicating a valid chromosome structure.

S1.6 OMKar Report: Event Interpretation

After segregating the chromosomes, OMKar interprets the structural variations in each chromosomal cluster using ISCN notation[36] as follows:

1. For each reconstructed chromosome in the chromosomal cluster, its chromosomal identity is assigned based on the highest represented centromere, and if it is acentric, by the overall highest-represented chromosome in the remaining segments.
2. In a pre-processing step, the Wild Type (WT) chromosome corresponding to the reconstructed chromosome is segmented prior to alignment, using the segments in the reconstruction.
3. Next, OMKar performs an alignment between each reconstructed chromosome and the corresponding Wild Type (WT) using an alignment that maximizes the longest common subsequence, with a linear penalty for indels.
4. After alignment, blocks of aligned segments are separated into three types: 1) concordant block represents matching between the reconstruction and the WT, 2) insertion block represents segments inserted in the reconstruction, and 3) deletion block represents segments deleted in the reconstruction. Adjacent blocks of the same type, and representing contiguous genomic coordinates are merged to minimize the total number of blocks.
5. The final interpretation assigns a representative SV name (using ISCN nomenclature) to each insertion or deletion block, reporting a potential cause of the deviation from the WT. Each SV has its unique signature in the combination of block types (Supplementary Table S1). For example, an inversion is always an insertion block of inverted segments next to a deletion block of non-inverted segments. In this example, it is more likely that a single inversion resulted in both the deletion and the insertion, instead of two separate events (inverted insertion and deletion). Similarly during the interpretation step, a compound SV is preferred over a sequence of simpler SVs (preference is given to reducing the total number of events).

To minimize the number of events, event types are resolved from the most complex to the least, using the following preference order: (1) inter-chromosomal balanced translocation and transposition; (2) intra-chromosomal balanced translocation and transposition; (3) other intra-chromosomal variations. Finally, if any deletions or insertions remain unaccounted for, they are marked as simple deletions or duplicated-insertions. During each variation type's resolution, each un-resolved block is iterated over, with the goal of being associated with the signature of that variation type. Balanced translocations attempt to associate anywhere in the cluster or chromosome, for inter- and intrachromosomal search, respectively. All other variations associate adjacent blocks.

For balanced translocations, the following step is performed after all insertion and deletion

blocks are resolved. All balanced translocations are initially denoted as *transposition*, associating a deletion block and a non-adjacent insertion block of an overlapping set of segments, with allowance for small indels. When transpositions form a cycle, they are interpreted as *balanced translocations*. This is implemented recursively, by jumping between each associated deletion-insertion pair, and then looking for nearby transposition blocks of the opposite type. When exhausted, if a cycle is formed, an n-break balanced translocation is interpreted, and otherwise, all transpositions are interpreted as individual transpositions.

When associating between different blocks, a procedure called "seed-matching" is applied. For each block that is currently being resolved, it searches for an associated block, such that the common subsequence between the blocks (in indivisible-unit of each segment) is sufficiently long (10 Kbp by default). The sizes of the flanking segments not matched may be limited by an "indel allowance". For example, an insertion block of $(B-, A-)$ next to a concordant block of $(B + C+)$ will be interpreted as a left-duplication-inversion only if $A-$ is less than 50 Kbp. On the other hand, the size of $C+$ is irrelevant for associating a duplication-inversion, as it is not between the two blocks. These allowances are determined empirically (Supplementary Section S4).

S1.7 OMKar Report: Visualization

The visualization is achieved by intersecting the coordinates in the observed chromosomal segment list with a given cytoband coordinate table. Each band pattern is then stacked and displayed using Matplotlib (v3.7.5). In an alternative plot, the bands are segregated by the segment boundaries in the OMKar output, instead of the cytoband. The event label is then applied to the corresponding segment location on the band stack.

By default, the table of important genes comes from genes that have CN alteration or within 20 Kbp of an event's breakpoint. This list of genes are further filtered to only include those in the Developmental Disorders panel in the Gene2Phenotype database (DDG2P)[28].

S2 KarSim

S2.1 KarSim module for simulating karyotypes

The KarSim module outputs a Molecular karyotype file, a corresponding FASTA file, and a history log with event segments and edges noted are outputted for downstream usage. The Molecular karyotype and history log can be used for KarCheck comparison, while the FASTA file can be used as input for simulating the sequencing technology of choice, given that many of such simulators are already available. KarSim is publicly available at `https://github.com/MolecularKaryotype/KarSimulator`.

Three steps are taken in order, with step 2 being optional:

1. A template karyotype is created given the counts of autosomes and sex chromosomes.
2. (Optional) a series of SVs are applied to the karyotype

3. The FASTA formatted sequence of the rearranged chromosomes is generated.

All intermediate files in steps 1 and 2 are Molecular karyotypes, which can be read-in for multiple parallel edits or outputting the FASTA file.

There are two methods to introduce additional SVs to a karyotype: 1) manual addition of SVs given the variation type and exact boundaries, and 2) using a parameter JSON file that contains the number of SVs, each SV type's likelihood, max/min size for each SV type, terminal occurrence likelihood etc. Both processes can be applied multiple times and in an interleaved fashion. The types of SVs supported can be found in Supplementary Table S2. In addition, the user has the option to include a masking file such that SVs are not generated within proximity to any of these regions. Another parameter can be passed in to prevent events that result in the formation of segments smaller than a certain size, useful for sequencing technologies which have a resolution limit.

S2.2 Implementation

KarSim is implemented using Python (version 3.9). A flowchart of the algorithm is in Supplementary Fig. S2A. Each chromosome in a karyotype is represented as a sequence of oriented segment objects of the hg38 genome, denoted by the chromosome, start index, and end index. To simulate an SV, left and right breakpoints are first introduced to the chromosome, breaking up the chromosome into segments. This results in each SV breakpoint being on the exact border of a segment. Then, the corresponding rearrangement is applied to the segments to simulate the intended SV.

For the parameterized-random SV selection, SVs are selected one at a time, for the number of SVs indicated on the parameter file as follows:

1. The SV type is randomly selectedbased on its likelihood.
2. The event size is selected from a uniform distribution between the maximum and minimum size indicated for the SV type.
3. The left breakpoint of the event location is selected uniformly among all chromosomes.
4. The right breakpoint is calculated based on the size of the event selected earlier. For a balanced reciprocal translocation, it is selected similar to the left breakpoint.
5. The breakpoints are applied to partition the segments, and if a masked region file or a smallest segment allowance is applied, the resulting breakpoints and segments are checked for legality. If the result is illegal under the parameters, steps three through five are recomputed.

S2.3 Generation and Processing of Simulated OGM Data

The FASTA files generated by the KarSim module were processed by OMSim [29] to simulate a BNX file containing OGM molecules with added noise. Parameters for the enzyme, OGM generation, and noise were sourced from publicly available repositories [41]. The Bionano Solve pipeline (v3.7) [25] was then used to compute CNVs, SV calls, and contig alignments, and these outputs were used as input for OMKar to reconstruct the final virtual karyotype.

When simulating structural variations, we used the masking files provided in Bionano Solve v3.7 (also included in repository[41]). The masking file is on reference hg38, with size of 423.1 Mbp (13.65%), including 128.1 Mbp (4.13%) in centromeres and telomeres, and 64.5 Mbp (2.15%) in acrocentric chromosomes' p-arm. None of the SVs was simulated with breakpoint boundary within 200 Kbp of any masking region. Events were simulated with size 50 Kbp to 2 Mbp.

S3 KarCheck

KarCheck is designed to be a symmetric comparator between two unphased karyotypes, denoted by K_r and K_t, by comparing their SV and CNV calls. In addition, to accommodate molecular techniques that do not have a nucleotide-level resolution, KarCheck has an adjustable tolerance for small breakpoint mis-matching for an SV that is present in both karyotypes. KarCheck is publicly available at `https://github.com/MolecularKaryotype/KarComparator`.

S3.1 KarCheck Preprocessing

Preprocessing is first applied to K_t and K_r to partition chromosome groups into *chromosome clusters*. Two chromosome groups are linked into the same cluster when there exists a breakpoint connecting them (signaling an inter-chromosomal SV). A maximal connected component of linked chromosome groups is denoted as a chromosome cluster.

Recall that we define each karyotype as an ordered list of segments. To make the two karyotypes comparable, we further partition their segments such that the two karyotypes share an identical set of segments (Supplementary Fig. S2B). To achieve this, all left/right endpoints of segments are collected, and if a segment has an endpoint internal to it, it is split into two, ensuring the left/right endpoint are on the boundaries (Supplementary Fig. S2B).

S3.2 SV similarity computation.

After pre-processing, the SVs between the two karyotypes are compared as a directed-multi-graph, which offers ease of checking the orientation of each SV edge (Supplementary Fig. S2C). On this graph, nodes represent either the start or the end of a segment (denoted as s and t), with the addition of a source node (S) and a sink node (T) for transition at the start and end of each chromosome. Edges represent the presence of a transition between two segments. For example, a chromosome of [k+, l-], where the WT is [k+, l+], will be represented by edges (S, k_s),(k_s, l_t), and (l_s, T).

Since the two karyotypes share the same set of segments, they also share the same set of nodes. Therefore, the comparison between the two sets of edges on this graph is equivalent to the comparison of the two sets of SVs. To allow tolerance in breakpoint matching, we define a linear distance function. Denote two non-segment edges as (a, b, o) and (c, d, o'), where o (and o') describe orientation. Define distance $D((a, b, o), (c, d, o'))$ as:

$$D((a,b,o),(c,d,o')) = \begin{cases} \infty & \text{Chr}(a) \neq \text{Chr}(c) \text{ or } \text{Chr}(b) \neq \text{Chr}(d) \\ \infty & o \neq o' \\ |\text{pos}(a) - \text{pos}(c)| + |\text{pos}(b) - \text{pos}(d)| & \text{otherwise} \end{cases}$$

Prior to distance computation, identical edges between the two graphs are pruned. Second, if a transition is s-to-t or t-to-s and is between two segments from the same chromosome with a small distance ($\leq$ 5 Kbp), it is pruned. This is justified by that these transitions represent small indels without change in orientation, which are not responsible if the technique has a minimum resolution threshold. Finally, minimum weight bipartite matching is performed for the remaining transition edges between the two karyotypes, with a maximum allowed matching distance 200 Kbp. Matching pairs are pruned, because they are considered similar SV edges. The matching distances are recorded for downstream analyses upon resolution. Finally, all residual non-segment edges are the differential SV edges between the two karyotypes. We use K_r to denote the reconstructed chromosome, and K_t as the WT or true chromosome. Therefore, residual edges in K_t represent false negatives, and residual edges in K_r represent false positives.

SV Similarity Metrics. The similarity of each individual SV edge is compared via the count of K_t edges after the two initial pruning procedures, and the residual edges in K_t and K_r after final matching. TP = |inital edge| $-$ |K_t residual|, FN = |K_t residual|, and FP = |K_r residual|. A Jaccard Similarity is computed for each cluster of compared simulated data to penalize both false positive and false negative events. For comparison against real data, only the recall is computed for each cluster, as the cytogenetic methods employed for the reference calls do not necessarily have small enough resolution or ability to catch de-novo balanced events.

S3.3 Copy Number similarity comparison and metrics

CN comparison is done by binning the whole genome (excluding prefix/suffix masked region) into spanning, non-intersecting bins of 50 Kbp +/- 100 bp (exact size chosen to maximize the size of the last bin on the chromosome). Each bin is used to store the average CN within that region, separately for the K_t and the K_r.

Then, for each cluster, the chromosome groups are determined by the union of all the chromosomal origins of the segments in the cluster. Each cluster's CN bins are the subset of the total CN bins, to only include the chromosome groups within the cluster. A WT expected count is determined to each chromosome group by rounding the average bin CN from K_t (diploid for autosomes and XX or XY for sex chromosomes).

The values of the CN bins' of K_t and K_r are computed with CNs from all corresponding segments. If a bin has a gain or loss of more than 0.05 CN from the WT expected count, it is a marked as "CN gain" or "CN loss". Otherwise, it is marked as "CN neutral". This forms a paired CNV array where the Jaccard Similarity can be computed. The denominator of the similarity is

the count of bins marked as CN gain or loss in either K_t or K_r, and the numerator is the count of bins where K_t and K_r agree on CN gain or loss.

S3.4 Addtional Functionalities for Downstream Analyses

Edges on both K_t and K_r can be labeled with additional input. For example, Table 2 was computed where each edge in K_t from the simulation was labeled with the Structural Variation event type, so a summary statistics was generated from the residual edge count of each event type.

When matching transition edges, the distances between the matched edges can be collected for analysis. This can be further categorized by having K_t's transition edge labeled with the causal event type. A detailed analysis on OMKar's reconstructions' distances against the simulation can be found in Supplementary Section S4.

S3.5 Usage in validating real data reconstructions with previous cytogenetic records

A function was implemented in KarCheck to allow efficient input of a karyotype for the purpose of validating a reconstruction against previous cytogenetic records on that karyotype. For each karyotype, its aneuploidy (if any), each event's induced SV-breakpoints, and CN changes are taken as input for the validation. This information is sufficient to populate a full Molecular karyotype, while KarCheck assumes the rest of the genome is WT.

For real data with cytogenetic records, the cytogenetic records were used as the "truth" karyptype (K_t) and compared against the reconstruction (K_r). CN gains called with microarray do not specify the structural breakpoints of the amplification. For these, we first assumed each amplification was a tandem duplication, and if the matching failed, we manually verified if the reconstruction contained the amplification as a segmental duplication. Additionally, previous cytogenetic techniques using microarray and staining did not fully capture every event on the genome, so the K_t was treated as incomplete, with potentially missed TP events. Therefore, for our final statistics of the comparison, we only computed the recall to verify if OMKar reconstruction included all the previously identified TPs.

S4 Analyses of Event Distances

The resolution of OGM was determined by the matching distances between the Truth SV edge and the reconstructed SV edge, using KarCheck. Observations from this analyses were used to estimate the sensitivity of OMKar event interpretation (Supplementary Table S1).

When applying KarCheck between the simulated and the reconstructed karyotypes, history logs from KarSim's output was used to label SV edges on K_t. This step marks each SV edge with its causal rearrangement. During KarCheck matching, the distance between a pair of matched edges were recorded with the causal SV type. This distance is further separated as the two distances of the endpoint.

Then, the matched edge from K_t was searched in OGM's SMAP output (contains all SV calls), with a proximity of 50 Kbp, equal to the SV call merging distance of OMKar. From Fig. S1, we observed duplication inversions tend to have much larger distances than all remaining structural variations simulated. Most of the non-duplication-inversion were reconstructed with less than 5 Kbp distance from the truth, much lower than the average gene size of 10-15 Kbp. Thus, if an SV contains or interrupts a gene, OGM and OMKar are likely to reconstruct the correct boundary to perform genotype-to-phenotype inference. Additionally, for all distances greater than 50 Kbp, the true SV edge was not captured with a high proximity in the SMAP, therefore, OMKar either had to reconstruct based on a distant SV call or had to infer the missing SV call based on CNV call.

S5 Terminal Event Simulation and Validation

We simulated a total of 117 terminal Structural Variations. During analyses, it was found that only 50 of the 117 SVs (42.7%) were reconstructed under the 0.2 Mbp matching distance (same distance used for non-terminal SVs; Supplementary Table S3). However, 78 of the 117 SVs (66.7%) were reconstructed under the 5 Mbp matching distance. Because one of the endpoint was in the terminal masking region, most terminal SV edge did not have an SV call. For unbalanced SVs, OMKar reconstructed these variations using the information from CNV calls, which had much greater error in the event boundaries. For inversion, as a balanced SV, missing the SV call meant OMKar had no other information on the SV, thus, resulting in a much lower recall rate at all distances. In addition, terminal events such as arm deletion may result in a false loss of centromere call, which result in a relatively higher false positive chromosomal-loss aneuploidy reconstruction from OMKar.

From the real data we received, we observed no terminal structural variation, hinting that terminal SVs were far less frequent than our simulation. For future reference, OMKar may incorporate additional information such as alignment contigs to improve the boundary accuracy and recall rates.