Patterns, Volume *2*

Supplemental information

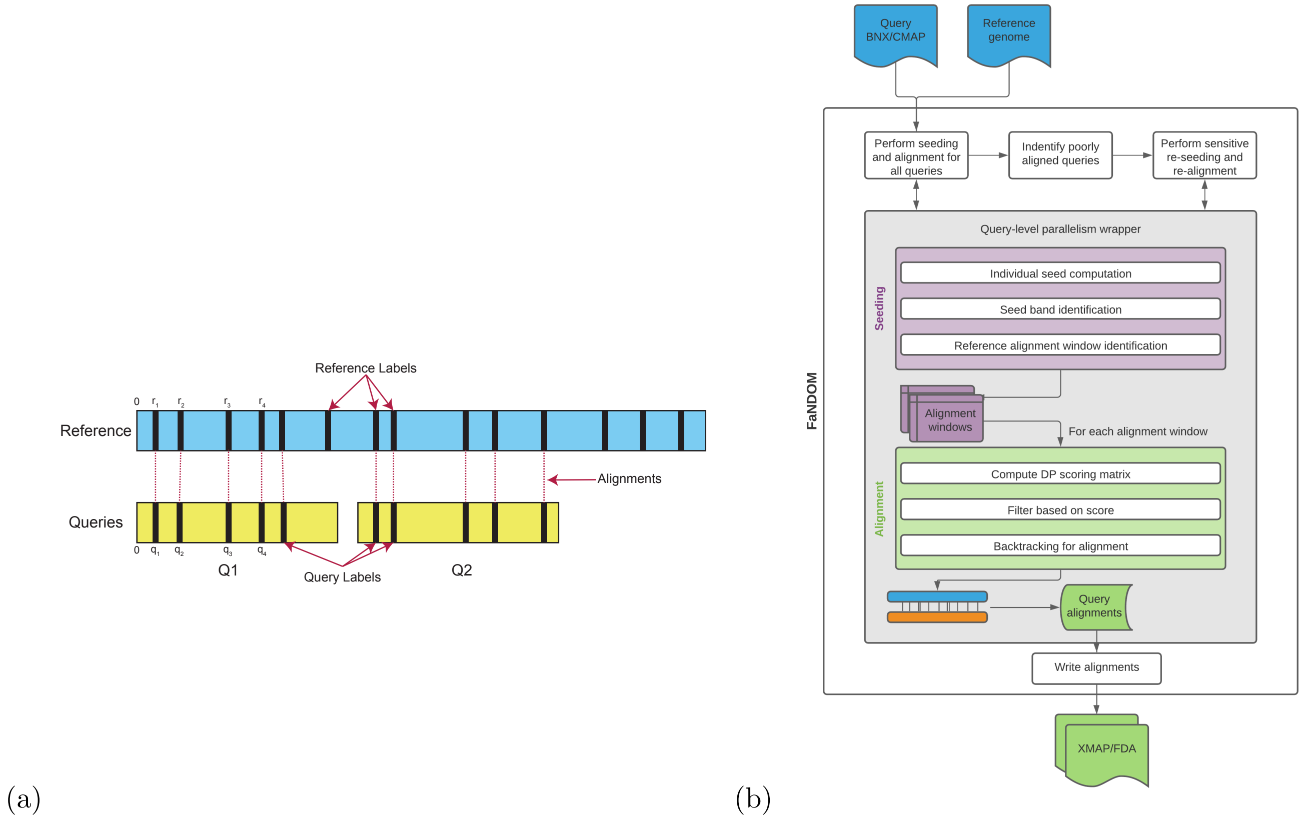
FaNDOM: Fast nested distance-based seeding of optical maps

Siavash Raeisi Dehkordi, Jens Luebeck, and Vineet Bafna

A Supplemental Experimental Procedures

Methods S1 A high-level overview of optical mapping and FaNDOM

FaNDOM utilizes optical mapping data which is converted to BNX or CMAP representations, and outputs optical map alignments in XMAP or its own FDA (**Fa**n**D**OM **A**lignment) file formats. Figure S1a provides a cartoon representation of multiple optical map 'queries' aligned to an optical map 'reference' segment. The schema of the FaNDOM aligner and its implementation is described in Figure S1b. The seeding and alignment modules are called by each parallel thread to produce and store alignments of query to reference.



Reference Labels
0
r_1
r_2
r_3
r_4
Reference
Alignments
Queries
0
q_1
q_2
q_3
q_4
Q1
Query Labels
Q2
(a)
Query BNX/CMAP
Reference genome
FaNDOM
Perform seeding and alignment for all queries
Indentify poorly aligned queries
Perform sensitive re-seeding and re-alignment
Query-level parallelism wrapper
Seeding
Individual seed computation
Seed band identification
Reference alignment window identification
Alignment windows
For each alignment window
Alignment
Compute DP scoring matrix
Filter based on score
Backtracking for alignment
Query alignments
Write alignments
XMAP/FDA
(b)

Figure S1: **Overview of OM alignments and FaNDOM software.** (a)Cartoon diagram of optical map queries aligned to *in-silico* reference map. $r1, r2, ...$ and $q_1, q_2, ...$ all represent numeric values in base-pair units of the expected or measured locations of labels in the reference and query, respectively. (b) Overview of FaNDOM software.

Methods S2 Molecule filtering and reference pre-processing

Lower bounds for molecule size filtering were selected using guidance from Bionano documentation available at the Bionano genomics website, page 8. Thus by default FaNDOM ignores molecules < 150 kbp or < 10 labels.

Given that the vast majority of distances between consecutive labels in the reference genome is $\ll 250$ kbp, molecules containing unlabeled stretches > 250 kbp are likely chimeric molecule fragments or incompletely labeled molecules and thus FaNDOM does not attempt to align molecules with unlabeled stretches exceeding 250 kbp.

Optical mapping data also suffers from an error modality in which labels located nearby on a molecule are indistinguishable and measured as a single fluorescent label[15]. FaNDOM pre-processes the reference genome to identify such sites. By default, FaNDOM selects a threshold of 800 bp for which to merge consecutive reference labels (creating an artificial label in the center). The basis for this choice is both theoretical and has some practical support as demonstrated in Luebeck et al., 2020[22]. The approximate length of a single basepair of DNA is approximatel 0.34 nm. The approximate wavelength of green light (used by the label fluorescence laser in the Bionano Saphyr) is approximately 550 nm. This suggests 500 (nm) $\div$ 0.34 (nm x bp^{-1}) $\approx$ 1600 bp of DNA are spanned inside the wavelength of green light. Applying the Abbe diffraction limit of $\lambda/2$ implies a theoretical resolution limit of 800 bp for a given label.

Methods S3 Determining scaling and stretch factors

We define 'scaling' and 'stretch' as two independent error modalities which we examined when benchmarking FaNDOM. *Scaling* refers to the calibration of measured basepairs per pixel in the imaging of optical map molecules by the instrument. If this calibration is not completely accurate, we observed that this error can lead to global lengthening or shortening of all molecules derived from the instrument. To ensure that optical map data has been properly scaled following the image processing performed by the Bionano instrument, we apply a grid-search method to try a range of re-scaling factors. *Stretch* on the other hand refers to the physical lengthening or shortening of individual DNA fragments traveling through the nanochannel array. It is accounted for after 'scaling' has been resolved.

A scaling correction (adjustment of base-pairs per pixel) may need to be applied to data-sets to improve alignment quality. To test how scaling profiles varied across samples, we obtained a a selection molecules from 38 human samples provided by Bionano Genomics. For each sample, we sampled 250 molecules over

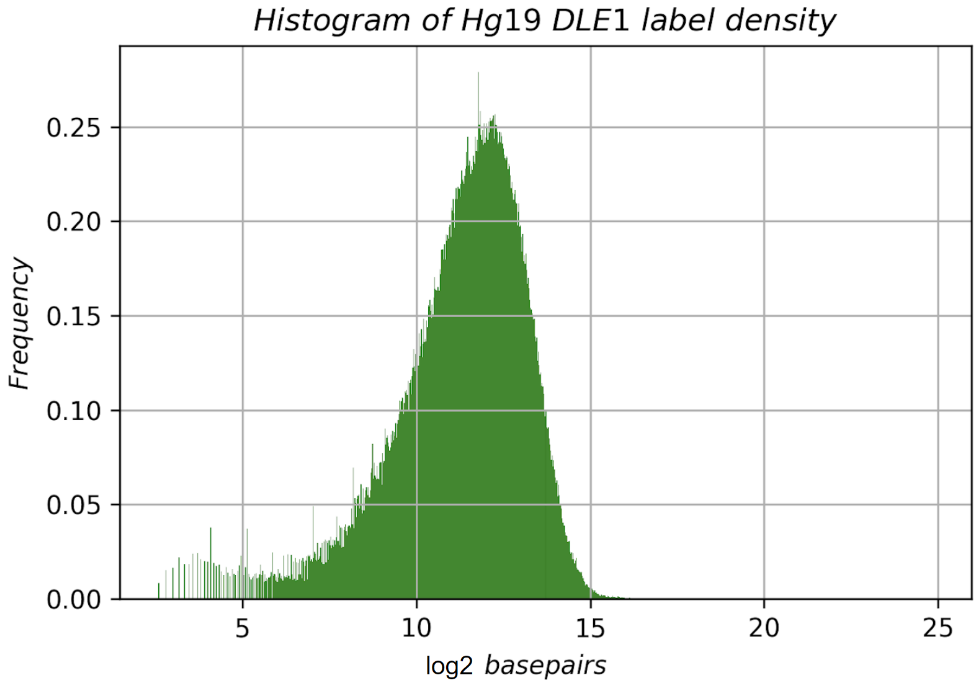


Figure S2: **Histogram of Hg19 DLE1 label density.** DLE1 label density (distance between consecutive label sites) in the hg19 reference genome. Note that the vast majority of the distances are lower than 250kbp $\simeq 2^{18}$bp.

a range of scaling factors from -0.96 to 1.2, and selected a single scaling factor that had the highest sum of alignment scores for all queries. Figure S3a shows the distribution of the best scaling-factors over the 38 samples.

To test for stretch, we selected 100,000 high confidence alignments of molecules. For each alignment of consecutive labels in the query to labels in the reference given by $(q_a, r_b), (q_c, r_d)$, we computed $\frac{|q_a - q_c|}{|r_b - r_d|}$ as the stretch factor. Fig. S3b, S3c shows the distribution of median of stretch factors for 100,000 molecules in two distinct cell-lines with different scaling factors, and suggests a low standard deviation of 0.02.

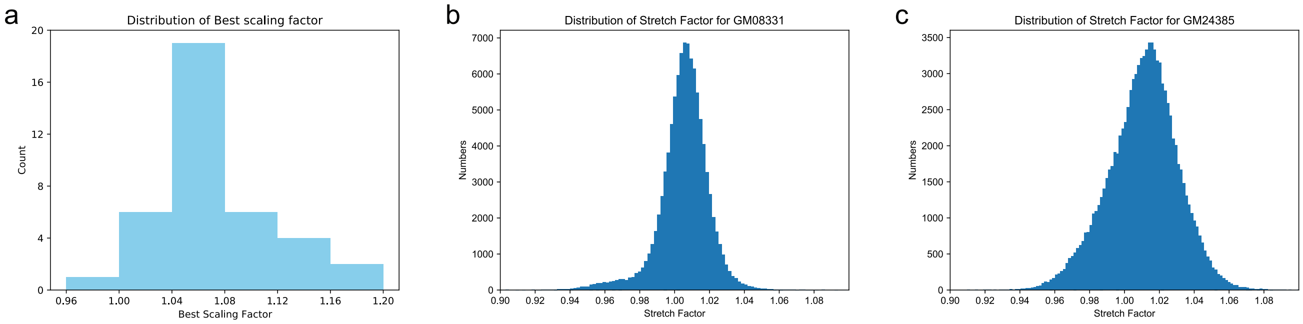
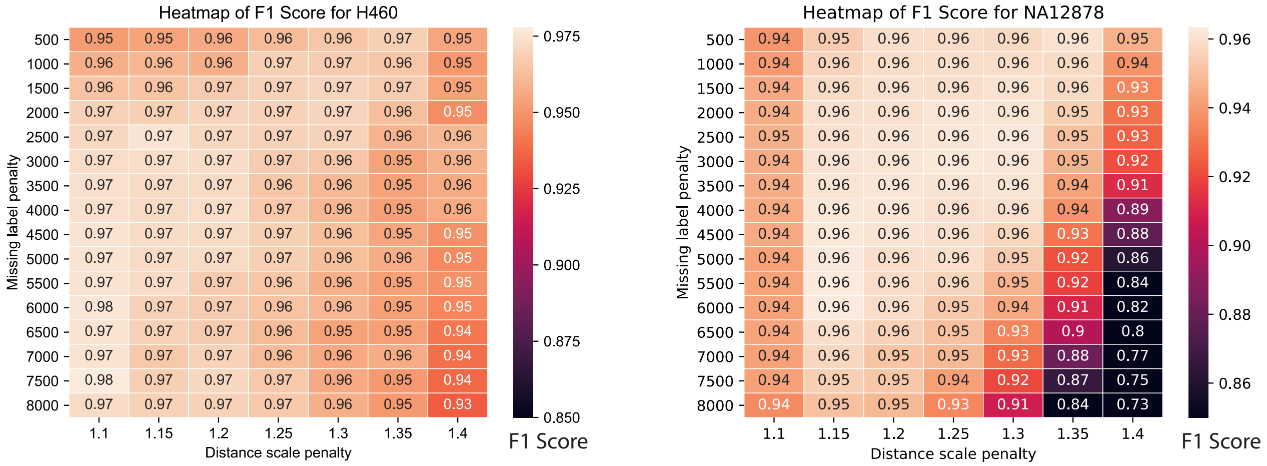


Figure S3: **Scaling factor variation**. (a) Distribution of the single scaling factors for each of 38 human samples achieving highest sum of alignment scores. (b-c) Distribution of estimated "stretch-factors" for samples GM08331 (b) and GM24385 (c) after applying scaling correction.

Methods S4 Setting default values for FaNDOM parameters



Heatmap of F1 Score for H460

Missing label penalty	1.1	1.15	1.2	1.25	1.3	1.35	1.4
500	0.95	0.95	0.96	0.96	0.96	0.97	0.95
1000	0.96	0.96	0.96	0.97	0.97	0.96	0.95
1500	0.96	0.96	0.97	0.97	0.97	0.97	0.95
2000	0.97	0.97	0.97	0.97	0.97	0.96	0.95
2500	0.97	0.97	0.97	0.97	0.97	0.96	0.96
3000	0.97	0.97	0.97	0.97	0.96	0.95	0.96
3500	0.97	0.97	0.97	0.96	0.96	0.95	0.96
4000	0.97	0.97	0.97	0.97	0.96	0.95	0.96
4500	0.97	0.97	0.97	0.97	0.96	0.95	0.95
5000	0.97	0.97	0.97	0.97	0.96	0.96	0.95
5500	0.97	0.97	0.97	0.96	0.96	0.95	0.95
6000	0.98	0.97	0.97	0.96	0.96	0.95	0.95
6500	0.97	0.97	0.97	0.96	0.95	0.95	0.94
7000	0.97	0.97	0.97	0.96	0.96	0.96	0.94
7500	0.98	0.97	0.97	0.97	0.96	0.95	0.94
8000	0.97	0.97	0.97	0.97	0.96	0.95	0.93

Heatmap of F1 Score for NA12878

Missing label penalty	1.1	1.15	1.2	1.25	1.3	1.35	1.4
500	0.94	0.95	0.96	0.96	0.96	0.96	0.95
1000	0.94	0.96	0.96	0.96	0.96	0.96	0.94
1500	0.94	0.96	0.96	0.96	0.96	0.96	0.93
2000	0.94	0.96	0.96	0.96	0.96	0.95	0.93
2500	0.95	0.96	0.96	0.96	0.96	0.95	0.93
3000	0.94	0.96	0.96	0.96	0.96	0.95	0.92
3500	0.94	0.96	0.96	0.96	0.96	0.94	0.91
4000	0.94	0.96	0.96	0.96	0.96	0.94	0.89
4500	0.94	0.96	0.96	0.96	0.96	0.93	0.88
5000	0.94	0.96	0.96	0.96	0.95	0.92	0.86
5500	0.94	0.96	0.96	0.96	0.95	0.92	0.84
6000	0.94	0.96	0.96	0.95	0.94	0.91	0.82
6500	0.94	0.96	0.96	0.95	0.93	0.9	0.8
7000	0.94	0.96	0.95	0.95	0.93	0.88	0.77
7500	0.94	0.95	0.95	0.94	0.92	0.87	0.75
8000	0.94	0.95	0.95	0.93	0.91	0.84	0.73

Figure S4: **Alignment score parameters.** Effect of missing label penalty (c) and distance scale penalty (k) parameters on alignment F1 score of (left) H460 OM assembled contigs and (right) NA12878 raw molecules. The performance remains robust for a wide range of parameters.

FanDOM uses a number of parameters that, while modifiable by the user, were optimized for Saphyr technology using a subset of 10,000 alignments computed by the Bionano Ref-Aligner. These alignments were chosen at random but with a small bias towards lower scoring alignments to ensure that we were not optimizing only for the high-quality alignments. The experiments leading to the default settings of the parameters are described below. We first experimented to get crude pre-optimized values for each parameter, and then optimized each parameter in turn, while keeping the others fixed. We note that FanDOM was robust to small changes in parameters and that the final testing using these parameters on a much larger set and across multiple cell-lines and simulations. Our choice of default parameters was based on achieving maximum speed-up while maintaining recall above 90% for this hard data-set.

Alignment score parameters. We considered the impact of simultaneously varying c (false-label penalty) and k (distance scale penalty) on the F1 score which combines precision and recall. We randomly selected 500 assembled OM contigs of H460 cell-line and 1000 raw molecule of NA12878 cell-line for the tests. Fig S4 indicates that a wide range of k, c showed identical performance. Increasing k, c resulted in the same alignments but with tighter boundaries. We chose the distance scale parameter $k = 1.15$ and false-label parameter $c = 3000$ as default values.

Tolerance. T represents tolerance for matching two genomic distances between query and reference. Increasing the tolerance would increase sensitivity, but would result in increased running time. To address this trade-off, we experimented with T ranging from 200 to 800, plotting running time versus recall. T ranging between 350 and 400 reached our target sensitivity but provided very high speedups. We chose $T = 350$ as default (Fig. S5a). Note that relaxing T to be very large can lead to false alignments and reduced sensitivity.

Width of alignment band. We experimented with $B_w \in [6, 18]$kbp (Fig. S5b). The choice of $B_w = 12$kbp reached our target sensitivity of 90% while maintaining speed of search.

Minimum number of seeds, T_h in a band. T_h represents the minimum number of seed matches within a band for it to be selected. Experimenting with parameters $T_h \in [3, 7]$ provided $T_h = 4$ as the value that achieves 90% recall with high speed (Fig. S5c).

Number of alignments computed. For each of the $10,000$ queries, and each band where a seed threshold was met, we computed a band-score as described in methods, and kept the top 150 band-scores for each query. when we computed the band-score ranks of the true alignments of each of these queries, we observed a range of ranks. For example, in 8417 of 10,000 queries, the true alignment also had the highest band-score and a rank of 1. Nevertheless, the ranks had a long-tailed distribution. We fit the band-scores to an exponential distribution with parameter λ. Fig. S5d plots the rank of the true alignment of a query versus the maximum value of (Band-score/λ) for that query. Note that when the maximum band-score exceeds 2.2λ, the correct alignment is ranked within top 10 in most cases(rectangle with x, y intercepts as $10, 2.2\lambda$ in Fig. S5d). The shaded region in Fig. S5d were chosen to set the cut-offs as below. The points in the un-shaded region corresponded to missed true alignments and represented 3% of the queries.

$$\max_a \text{score}(B_a) \begin{cases} > 2.2\lambda & \Rightarrow \text{Align top 10 bands} \\ > 1.7\lambda & \Rightarrow \text{Align top 50 bands} \\ > 1.5\lambda & \Rightarrow \text{Align top 100 bands} \\ \text{otherwise} & \Rightarrow \text{Align top 150 bands} \end{cases}$$

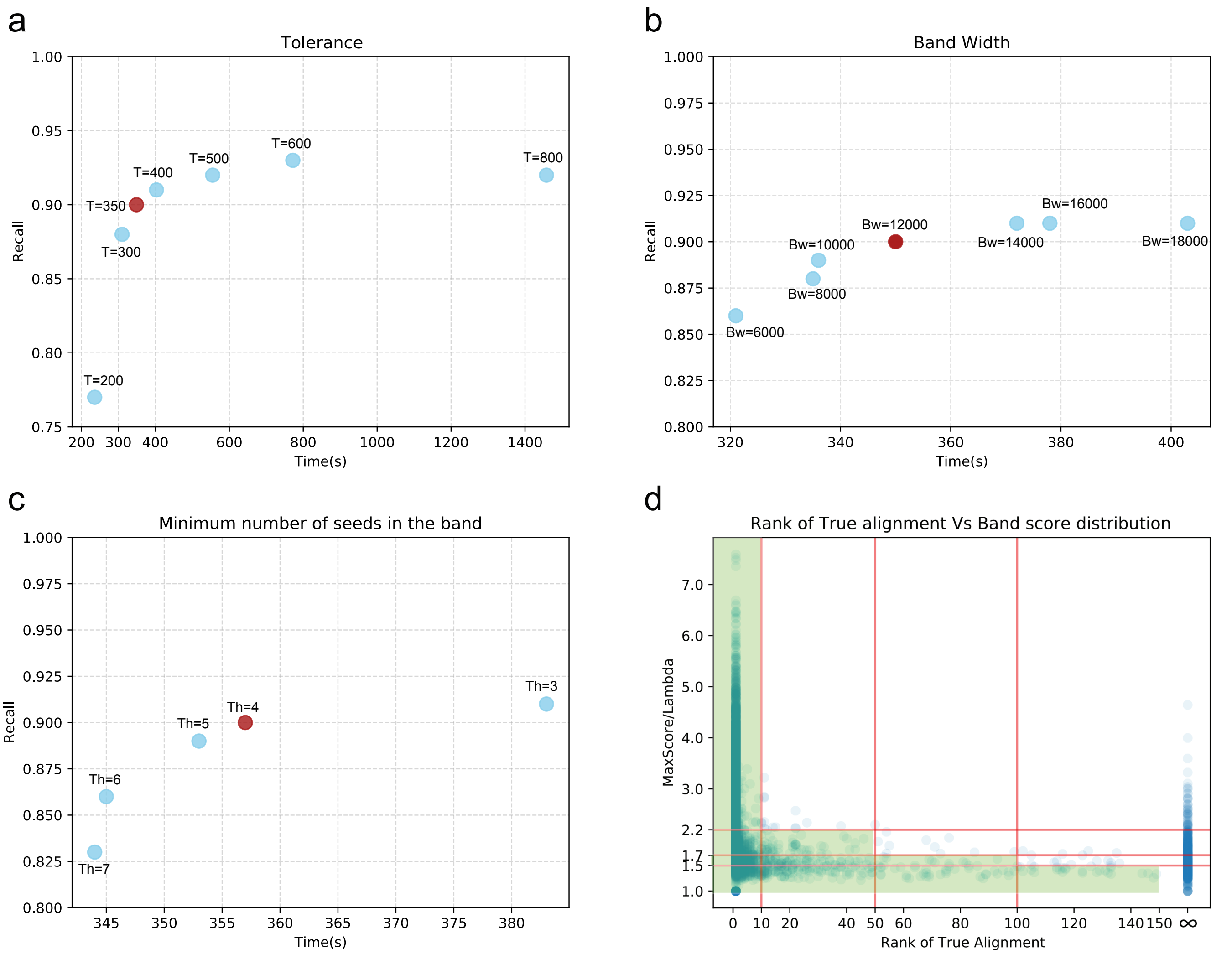


Figure S5: **Parameter tuning.** (a) Tolerance. (b) width of alignment band. (c) Number of seeds. (d) lambda

Methods S5 Experimental framework

All experiments were run on an Intel(R) Core(TM) i9 -9900 CPU @3.10GHz with 32 GB of main memory running Ubuntu 18.04.3 LTS (Bionic Beaver), using 10 threads. The executable file for OMBlast was downloaded directly from from GitHub page and executable file for RefAligner was used from Bionano Solve pipeline version Solve3.5.1_01142020. The following commands were used for each aligner. All aligners were run with default alignment parameters.

FaNDOM:

```
./FaNDOM -t=10 -no_partial -r=ref.cmap -q=query.bnx -sname=output/out
python PythonScript/filter_individual.py -i output/out.xmap
-o output/out_filtered -r ref.cmap
```

OMBlast:

```
java -jar OMTools.jar OMBlastMapper --refmapin ref.cmap --optmapin
query.bnx --optresout OMBlast.xmap --writeunmap false --multiple false
```

```
--thread 10 --minsig 10 --minsize 25000 --minconf 0
```

RefAligner:

```
RefAligner -i query.bnx -ref ref.cmap -o output/out  -maxthreads 10
```

Methods S6 RAM Usage and Running time

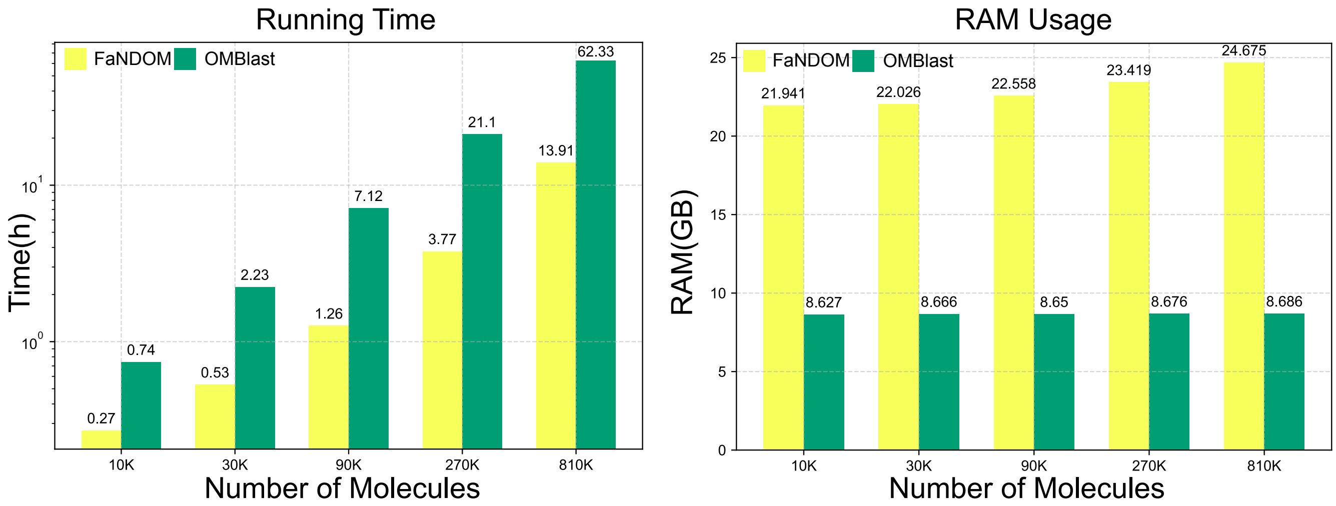


Figure S6: **Scalability of RAM usage and running time**. (left) Running time; (right) RAM usage.

RAM usage of FaNDOM was dependent on the density of labels in the input. Based on the empirical experiments results for Saphyr technology, FaNDOM needed ∼ 2.3 GB of RAM per thread on average irrespective of the query molecule length. The memory requirement of OMBlast increased with query size, requiring 200Gb for Saphyr contigs, as per their own documentation.

Methods S7 Simulation

The reference genome used for generating simulated data was hg19_DLE. Genome size was 3095.7(Mbp). The total number of labels was 527922 and average distances between labels was 5,704bp. From the reference genome, 10000 molecules were simulated and extracted by using OMTools simulation[17]. Based on the Bionano documentation for the Saphyr technology, the false positive label rate was 4 per each 100 Kbp and false negative label ratio was 0.1. The average molecule size was set to 250 Kbp. Also based on our calculation of the stretch factor variation (Fig. S3b, S3c), we set the stretch factor standard deviation to 0.02.

High error data:

```
java -jar OMTools.jar OptMapDataGenerator --refmapin hg19_DLE_mas
ked.cmap --flbound 150000 --moleno 10000 --optmapout q.cmap -fsize
250000 --rsln 100 --meas 100 --subound 1.02 --slbound 0.98
--scalesd 0.02 --fpr 0.00004
```

Low error data:

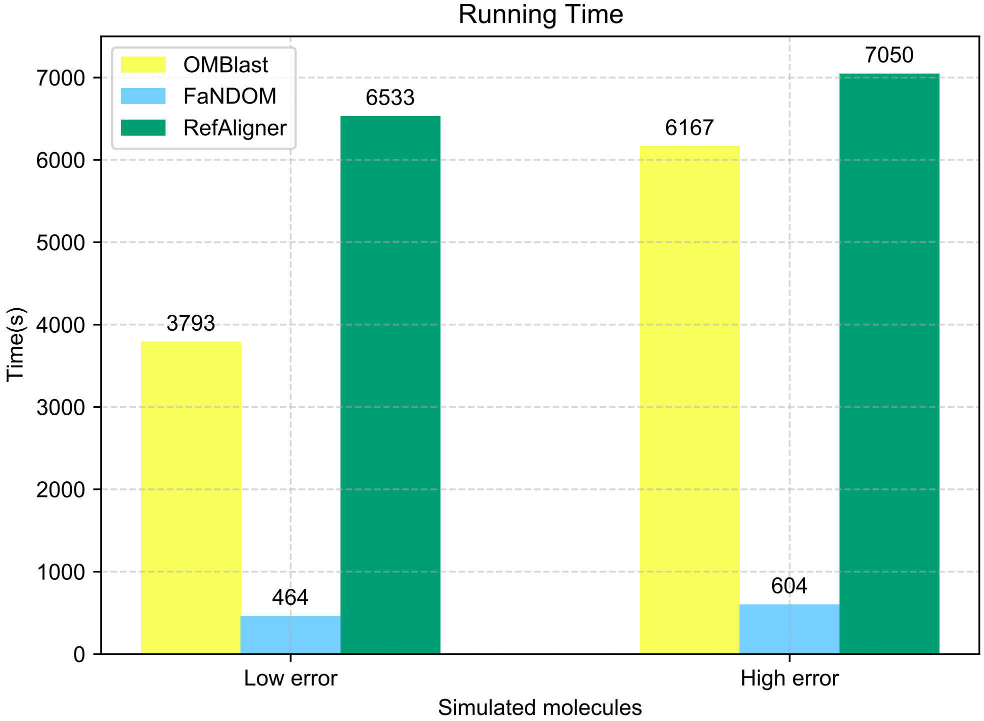


Figure S7: **FaNDOM performance on simulated data.**

```
java -jar OMTools.jar OptMapDataGenerator --refmapin hg19_DLE_mas
ked.cmap --flbound 150000 --moleno 10000 --optmapout q.cmap -fsize
250000 --rsln 100 --meas 100 --subound 1.02 --slbound 0.98
--scalesd 0.01 --fpr 0.00001
```

Methods S8 Comparison against Twin and Kohdista

Much algorithmic work on optical mapping data is related to aligning optical map reads to assemble large genomic optical map scaffolds. Computational tools have been designed for fast identification of overlapping pairs and assembly (e.g. MalignerIX[25], Kohdista[27]). The reference optical map scaffolds can be used for physical mapping of genomic sequences, by *in silico* digestion of the genomic sequence and mapping to the OM reference using tools such as TWIN[26].

While these tools were not explicitly designed to compare optical map fragments to an *in silico* digested genomic reference, we nevertheless benchmarked FaNDOM performance against TWIN[26], and Kohdista[27], which had previously shown fast, memory-efficient performance for the tasks they were designed to solve.

TWIN and Kohdista did not support the Bionano Saphyr technology, which is the dominant platform currently, and required for us to write customfile format converters to convert the modern .bnx, .cmap files into older file formats accepted by TWIN and Kohdista.

We took 10,000 OM molecules from NA12878 and used them as queries to align to the *in silico* digested human reference genome. However, it did not return any mappings. We did test that by using an identical sub-molecule from the *in silico* digestion, we were able to match, suggesting that TWIN showed poor tolerance for missing/false OM labels. TWIN results were previously demonstrated on simulated optical maps and optical map data with error profiles very different from Bionano Saphyr, so it is possible that a change of parameters could have changed the results. However, in personal communication, the authors did not recommend specific parameter settings appropriate for Bionano Saphyr.

We also attempted to use Kohdista for reference based-mapping. Its RAM usage was not optimized for the large human genome reference. Therefore, we tested a small group of 243 optical map molecules to the genomic reference chr10:0-46,272K (46 Mb). Kohdista used more than 130GB of RAM and did not return alignments after 12 hours. Also, When we tried to align this group of molecules to the entire chromosome 10, the RAM usage surpassed 200 GB. Since optical map data-sets frequently contain $>$ 1 million molecules, we concluded that Kohdista was not an appropriate tool for our problem.