UNIVERSITY OF CALIFORNIA SAN DIEGO

Alignment Methods for Optical Maps

A Thesis submitted in partial satisfaction of the requirements for the degree Master of Science

in

Computer Science

by

Jens-Christian Luebeck

Committee in charge:

Professor Vineet Bafna, Chair
Professor Prashant Mali
Professor Debashis Sahoo

2019

The Thesis of Jens-Christian Luebeck is approved, and it is acceptable in quality and form for publication on microfilm and electronically:

Chair

University of California San Diego

2019

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

ABSTRACT OF THE THESIS

Alignment Methods for Optical Maps

by

Jens-Christian Luebeck

Master of Science in Computer Science

University of California San Diego, 2019

Professor Vineet Bafna, Chair

Optical mapping is a DNA physical mapping technique that can measure large-scale structural variation in genomes and enables more accurate completion of genome assemblies. Particularly, we focus on the case where optical mapping is used to complete genome assembly on pre-identified segments such as may be found in a breakpoint graph. In this work we study methods for aligning optical map contigs with reference genome segments. This work introduces a novel method for optical map alignment that outperforms existing methods for aligning breakpoint graph reference segments to assembled optical map contigs. Also discussed are some additional modifications that can be made to the method.

Introduction

The advent of DNA sequencing has enabled tremendous advances in the biological sciences over the past fifteen years and continues to do so today through innovations to sequencing technologies. New protocols for specialized DNA sequencing are being developed at a rapid pace. Additionally, new long-read sequencing techniques are emerging, enabling the interrogation of larger-scale genomic phenomena, such as complex structural variation. As next-generation sequencing (NGS) has been dominated by the short-read paradigm, it has left many interesting biological questions related to structural variation unanswered. Of paramount importance to understanding sequencing data is the production of high-quality alignments of sequencing reads with reference genomes.

This research deals with a technique orthogonal and complementary to DNA sequencing, called optical mapping. Optical mapping is a successor to older techniques of restriction enzyme mapping, which enabled the physical mapping of fragments of DNA, broken by restriction enzymes, using the fragment lengths. Instead, optical mapping uses a fluorescent labeling of restriction sites to keep large fragments of DNA intact. This enables the physical mapping of large fragments of DNA. The information provided by optical mapping does not contain base-level information like NGS, instead it provides a powerful orthogonal validation that can lend further support to structural variation suggested by NGS.

This thesis discusses the current technologies for optical mapping, primarily BioNano Genomics optical mapping, as well as current strategies for optical map alignment. A new algorithm for efficient alignment of optical maps given genomic reference segments is also developed. Our work will examine the performance of this method on aligning amplified regions of tumor genomes to assembled contigs, specifically where a breakpoint graph of the tumor genome is known.

1. BioNano Optical Mapping

BioNano Genomics optical maps are fluorescently labeled pieces of DNA, marked with sequence-specific enzymes (Hastie et al., 2013). The average fragment length varies based on sample preparation; however, for a typical mapping run, most maps are around 200kb long. The BioNano protocol involves using at least 3 micrograms of input DNA (about 500,000 genome equivalents). Typically, cells are fixed onto an agarose plug. DNA is extracted, labeled and pushed through a nanofluidic channel, on which it is stretched out and imaged. The sample preparation process for a BioNano sample takes one to two weeks, however it is relatively inexpensive at around $700/sample. BioNano has two models of optical mapping machines currently available, Irys and Saphyr (www.bionanogenomics.com). The newer Saphyr instrument is capable of higher throughput at reduced cost. In the chemistry used by the Irys machine, DNA is labeled using a restriction enzyme nickase and repaired with a fluorescent nucleotide. The Saphyr chemistry relies on direct modification of nucleotides, and a fluorescent label is attached directly to the DNA, without cutting. This prevents the phenomenon known as fragile sites, where DNA may break if too many nicks appear in close proximity. The absence of fragile sites allows Saphyr molecules to be assembled into ultralong contigs, which can span chromosome arms.

The output of imaging the molecules with the BioNano analysis pipeline is a file containing the positions, in base pair units of imaged sites on the molecules. The bioinformatic pipeline from BioNano performs image detection tasks related to sizing the molecules and identifying labelled positions. The detected optical maps can be assembled into a optical map contigs. Some properties of the BioNano molecules are listed in Table 1. While the Saphyr contigs tend to be much larger, there are still many small contigs produced, particularly in places where large repeated elements exist.

The measured molecules or assembled contigs can be represented as a sorted list of positive real numbers, indicating the relative position on the fragment of DNA where a fluorescent label was detected. Thus, in the context of this work we refer to optical maps not as images, but as the sorted list of positive real numbers, whose relative positions describe the presence of detected k-mer motifs from a reference genome. This representation of optical maps we also all to contain errors, such as the ones described in the next section.

1.1. Optical Mapping Limitations

We now describe some of the current limitations in optical mapping, such as intrinsic error associated with the technology and the scale at which it is useful for detecting variation. As described in Table 1, the average label density of BioNano indicates that for Irys data, the spacing of labels is about one per 10kb. This means that structural changes inside that region are often impossible to detect, and furthermore, small structural changes encompassing one or two labels are also hard to detect due to measurement errors.

Enzymatic labeling of BioNano molecules and contigs is not 100% efficient. Some issues arise with false-positive and false-negative labeling of sites on individual molecules. For instance, in the Irys technology, on NA12878 BioNano data (Pendleton et al., 2015), false positive label occurs with incidence 0.78 labels per 100Kb and false negative labels with an incidence ratio of 0.098. Fortunately, the incidence of false-positives is independent from across the genome, so when assembling molecules into contigs, the effective false-positive labeling rate on contigs is extremely low. For false-negative labeling, this is much more frequent and a systematic problem. False negatives can arise for two different reasons. The first being that the enzymatic labeling process may not be completely efficient and leave some sites unlabeled, in an independent fashion. For the same reasons as the independent false-positive labeling, this is not an issue on assembled contigs, where high coverage erases infrequent mislabeling. However, the other reason for false-negative labeling is due to limitations in imaging.

We briefly demonstrate why fluorescent labels within a certain distance are indistinguishable from each other given the current limitations of optics. Given that the wavelength of green light is about 540 nanometers, and a single of 10bp turn of a DNA double-helix is 34 angstroms, this implies

$$\frac{540\, nm}{.34\frac{nm}{bp}} \approx 1500bp.$$

There are approximately 1500bp of DNA contained in the wavelength of green light. When labels appear within this proximity, then they are nearly indistinguishable from one another, and thus are not detected. For this reason, false negative labels tend to cluster together tightly.

Our method leverages this property in scoring the alignments of optical maps to reference segments. In Figure 2, we can see that there is a sharp decline in label density for NA12878 assembled contigs at around ~1500bp, while the Hg19 *in silico* digested reference genome indicates many restriction sites closer than that cutoff.

2. Review of Current Methods for Optical Map Alignment

Among the earliest sequence alignment algorithms for optical mapping included Michael Waterman's 1984 paper which introduced a dynamic-programming strategy for aligning pairs of distances between optical maps (Waterman, Smith, & Katcher, 1984). It used a recurrence similar to classical Smith-Waterman alignment, however instead of aligning two strings, two ordered lists of numbers are aligned. They used a distance matrix to perform weighting of differences between aligned pairs of elements.

A separate strategy, produced by Thomas Anantharaman formulated a Bayesian maximum likelihood model for matching noisy mapping data, and coupled it with a dynamic-programming search method (Anantharaman, Mishra, & Schwartz, 1997). The method was designed primarily for smaller genomes. This method served as the underpinning for the BioNano Genomics RefAligner; a proprietary tool which performs alignment of BioNano Optical maps. RefAligner is very capable of aligning optical maps coming from larger genomes, though much of the work in the last 20 years has been done proprietarily.

Following the work by Anantharaman, a more recent updated algorithm from Valouev and Waterman in 2006 used the same dynamic-programming strategy as proposed in 1984, however with a more complex scoring function, which also maximizes the likelihood as a scoring function (Valouev et al., 2006). The process is simpler than that of Anantharaman, using a simple log-likelihood of two joint distributions, and scaled better for larger genomes. The maximum likelihood model suggested by Valouev et al. does not take in to account any properties of the optical mapping technology itself and makes simplistic assumptions about the rates of false-positive and false-negative labels.

A tool very similar to Valouev's method was released by Mihai Pop and others called SOMA (Nagarajan, Read, & Pop, 2008). SOMA is lacking in sensitivity for many of the same reasons discussed previously (such as general assumptions about the input data), and has been outperformed by more recent methods.

A 2014 approach from Muggli et al., uses an FM-Index approach on the distances between pairs of labels in the maps to identify near-exact matches between reference and query (Muggli, Puglisi, & Boucher, 2014). This method has been published as tool called TWIN. This method, however, does not tolerate missing data and the errors typically found in optical mapping data.

Given the matching and optimal global search, the previously described methods do not perform well with combining fragmented local alignments, a major consideration when using this technology for detecting structural variation with optical maps.

A more recent method was released based on the BLAST methodology (Altschul, Gish, Miller, Myers, & Lipman, 1990) called OMBlast. The OMBlast method which seeks to address these issues utilizes a "seed-and-extend" approach for identifying alignments, while seeding using k-mer matching, where a k-mer is a set of distances between pairs of points on a map (Leung et al., 2016). OMBlast shows high speed and ability to detect structural variants. Its performance on BioNano data is comparable to that of RefAligner in our findings, which we will discuss later on. OMBlast does have the potential to miss smaller local alignments, or error-containing alignments if a seed is not aligned. OMBlast also requires some user intervention in order to identify correct parameters when non-standard alignment tasks are desired.

3. SegAligner

We introduce a novel method for aligning optical maps, SegAligner, which uses a dynamic programming strategy, with contig recruitment and E-value filtering to sensitively identify alignments of optical maps. We note that SegAligner tends to outperform existing methods, especially when the reference segments being aligned are smaller, have fewer labels, or have overlapping alignments with contigs. SegAligner leverages a number of properties of the BioNano data to improve estimates of the true number of labels expected at a given matching region. SegAligner is implemented in C++ and has multithreading support. It is publicly available on GitHub (https://github.com/jluebeck/AmpliconReconstructor). We discuss the various stages for this method next.

3.1. SegAligner Algorithm

We utilize a dynamic programming (DP) strategy for aligning optical maps like Valouev, et al. and introduce a novel heuristic scoring function. Our scoring function accounts for the event that the labels predicted by multiple nearby *in-silico* reference appear as a single label on an optical map contig due to limitations of optics. That is, two labels on a BioNano molecule are measured as a single label given their proximity to each other. In order to determine the likelihood of observing any given local alignment by chance, SegAligner calculates an E-value in a similar fashion to BLAST (Karlin & Altschul, 1990), extracting only significantly high-scoring alignments.

SegAligner takes as input the assembled optical map contigs and a set of in-silico reference genome maps. It outputs local alignments of the reference segments to the contigs.

SegAligner creates an elementwise pairing of subsets of labels from a reference map and a query map. We define such an elementwise pairing of ordered subsets of labels to be an optical map alignment. To provide a score for an alignment, we first define a matching region. A matching region is defined as the region between two labels on a map. For example, j and i in Figure 3 constitute a matching region with size j - i and one unmatched label in-between. The alignment score for two matching regions depends on the size discrepancy of the matching regions and the number of unmatched labels in each matching region. SegAligner uses a dynamic programming strategy to identify alignments which maximize the sum of the alignment scores produced by the matching regions. Our optical map alignment tool implements the algorithm we designed shown in Figure 4. Algorithm 1 has complexity $O(mn\delta^2)$, where δ is the label lookback threshold (default 5). The lookback threshold causes the alignment to be performed like a banded alignment during the DP scoring matrix computation stage.

Let `S[(j,q)]` be the best score of aligning a subsequence of the first `j` labels on the contig with a subsequence of the first q labels on the segment. Where `j` and q must be included in the subsequences. For a semi-global alignment, which is the standard case, the DP scoring matrix is initialized as

```
S[(j,q)] = -inf if j_index != 0 && q_index != 0, else 0
```

The DP recurrence relationship for the alignment is given as: `(i < j, p < q)`:

```
S[(j,q)] = max(S[(j,q)], S[(i,p)] + Score(i,j,p,q))
```

We describe the SegAligner method in the following high-level workflow:

SegAligner workflow:

Inputs: Assembled optical map contigs, genomic reference segments.

1) All-vs-all DP-scoring of reference segments and contigs.
2) E-value calculation per segment to compute significant alignment score threshold, *S** (described in section 4).
3) Backtrack on contig-segment pairings having a best score above S*.
4) Re-align segment with contig, barring previously used label pairings, while the max score is above S*, to extract multiple alignments (described in section 4.2)
5) Contigs with significant alignments to reference segments undergo overlap alignment detection step (described in section 4.3)

In tests on real datasets using breakpoint graphs identified for cancer cell lines using AmpliconArchitect (Deshpande et al., 2019), RefAligner and OMBlast can complete the alignment of breakpoint graph segments with contigs in approximately the same amount of time, but at reduced sensitivity than SegAligner, as shown in Figure 6B. SegAligner is much slower, however (Figure 6A). Given that a run of the BioNano instrument, including sample preparation, can take a long amount of time, we do not see this as problematic. As SegAligner does not use k-mer seeding like RefAligner and OMBlast, it guarantees deterministic and globally optimal output.

3.2 Chained Alignment of Contigs with Connected Reference Segments

If a breakpoint graph is provided as input to this alignment algorithm, and an alignment which must obey the breakpoint graph is desired, one can increase the complexity of the algorithm to output what we define as a chained alignment. A chained alignment is here defined as an alignment of reference segments connected through a graph against an assembled optical map contig. We present an algorithm for this approach in Figure 5.

In this form of the dynamic programming alignment, instead of a single reference segment, it iterates over multiple reference segments to create matching regions not just inside a single segment but also spanning multiple segments. This enables for the direct chaining of reference segments together, helping to solve the problem created by multiple un-linked local alignments. Such a task is useful for segments of the genome which have been chained together through a structure like a breakpoint graph. Breakpoint graphs encode a set of rearrangements which can transform one genome into another. In the case of cancer genome rearrangements, the breakpoint graph contains intervals of the genome, many of which have been amplified. A current problem in bioinformatics is the reconstruction of amplified regions of cancer genomes. We created a tool to identify putative rearrangements using NGS data, called AmpliconArchitect (Deshpande et al., 2019). In order to validate or disambiguate suggested structures, we can use optical mapping.

3.3 Computation of BioNano Label Collapse Probability

We notice that labels less than 2000bp apart in the reference genome have a non-zero probability of appearing as a single label on BioNano contig covering that region (Figure 1), which rises the closer they appear, due to limitations in optics. For a given matching region on the *in-silico* reference the expected number of unmatched labels in a corresponding optical map contig is given by the sum of the probabilities that the unmatched labels have not collapsed in either direction.

$$P(l_k\ not\ collapse) = (1 - P(l_k \rightarrow l_{k+1}))(1 - P(l_k \rightarrow l_{k-1}))$$

$$E(labels) = \sum_{l_k \in (l_{i+1}, l_{j+1})} \left(1 - \frac{(l_{k+1} - l_k)^4}{2000^4}\right)\left(1 - \frac{(l_k - l_{k-1})^4}{2000^4}\right)$$

In the above equations, $P(l_k \rightarrow l_{k+1})$ refers to the probability that label k has merged with its right neighboring label. The equation above provides as sum of probabilities, which is an expected number of BioNano labels for a matching region in an *in-silico* digested genomic reference restriction map.

3.4 SegAligner Scoring Function

The scoring function used by SegAligner is based on heuristics and incorporates the BioNano label collapse probabilities we estimate above. The scoring function is

```
Score(x,i,j,p,q,e_labels(i,j)) = 10000 – 5000(e_labels) – 5000(j
– (i + 1)) – (|(j_pos – i_pos) – (q_pos – p_pos)|)^1.2
```

In this scoring function, matching regions start with a base score of 10000, then are penalized for the total number of unmatched labels in the matching regions, where the reference matching region unmatched label count has been reduced to the expected number of labels after label collapse (`e_labels`). The score is also penalized by the absolute difference between the sizes of the matching regions, raised to the power of 1.2. We determined this value experimentally, using a variety of test values to identify a scaling factor that performed best on real test data, where the true alignments were known beforehand. The choice to use 10000 as a base score was based on the approximate BspQI label density, so that the alignment score for true alignments and the total length of the alignment in base pairs would be approximately the same.

4. Assessing statistically significant alignments

As the dynamic programming strategy for aligning optical maps produces a scoring matrix, it is necessary to create a statistical model to describe whether alignments found in the scoring matrix are significantly better scoring than random alignments. The E-value model for identifying high scoring alignments, developed by Altschul & Karlin, provides a powerful solution. We created an approximation method for the metric based on linear regression. Altschul & Karlin define the notion of a “high scoring segment pair” (HSP), which we use to describe optical map alignments with scores above the 50th-percentile in a distribution of random best-scoring alignments. HSPs in this case are not necessarily statistically significant alignments. Then we derive a scoring threshold based on the E-value model.

4.1 Derivation of the P-value from E-value Model

Established by Altschul and Karlin, the E-value is defined as

$$E = Kmne^{-\lambda S}$$

where E is the number of high-scoring alignments with score ≥ S (Karlin & Altschul, 1990). K and λ are constants specific to the segments being aligned, and m and n are the sizes of the assembled optical map and the reference segment, respectively. It follows that

$$\log(E) = \log(Kmn) - \lambda S$$

This is a linear relationship, from which the values of $\log(K)$ and λ can be derived from the intercept and slope, respectively, of the linear regression. Similarly to BLAST(Altschul et al., 1990), the number of random high-scoring alignments, *a,* with scores ≥ *S* is given by a Poisson distribution. In this case, *P(a)* becomes

$$P(a) = \frac{e^{-a}E^{a}}{a!}$$

This implies the probability, *P*, of finding at least one HSP for a given *E* is

$$P = 1 - e^{-E}$$

Therefore, the score cutoff corresponding to a given p-value, *P*, for an individual segment S_i^* is

$$S_i^* = -\frac{\log\left(-\frac{\log(1-P)}{Kmn}\right)}{\lambda}$$

4.2. Identifying Multiple Significant Alignments

In the case that the DP scoring-matrix for a segment aligned with a contig contains an entry exceeding S_i^*, SegAligner backtracks to form an alignment. The pairings of the labels in the alignment are marked as used. The segment and the contig are aligned again (a new scoring matrix is generated), however no previously used pairings can be re-used (the scoring matrix values for these pairings are set to an extreme negative value while the rest of the matrix is re-computed). SegAligner continues to re-align the segment with the contig until the best alignment score falls below the significance threshold.

4.3 Identifying Overlapping Alignments

As high-scoring overlapping alignments will have lower total scores than high-scoring global alignments, we developed a strategy to recover significant overlapping alignments between segments and contigs.

After SegAligner first identifies high-scoring semi-global alignments with contigs, the same procedure is repeated for segment alignment and scoring, however the scoring matrix is initialized to only support overlapping alignments, and the scoring distribution is constructed again, for each segment against all contigs. SegAligner identifies all high-scoring overlapping alignments whose alignment score exceeds the p-value threshold.

In this case however, a small number of highest scoring alignments between relevant contigs (having alignments to one or more segments) are considered. If the overlapping alignment score scaled by the proportion of the segment aligned to the contig exceeds the significant alignment scoring threshold for a global alignment, the alignment is kept and reported as a “tip alignment”.

5. Identification of Unaligned Contig Regions

In many cases, assembled optical map contigs may have high-scoring alignments with segments in the input set (e.g. breakpoint graph segment set), however, they may still contain some large (>40kb) unaligned regions along the assembled contig. We wish to detect the identity of such unaligned regions, as they may represent missing segments from the breakpoint graph contained on the amplicon or integration sites of the amplicon.

In order to detect the identity of the unaligned regions, we extract the unaligned regions of contigs having two or more high scoring alignments with graph segments and convert each extracted region to an individual BioNano consensus map (CMAP). SegAligner then aligns the unknown regions to the Hg19 *in-silico* digested reference genome, producing a scoring matrix for each of the unknown segments with each entry in the reference genome. By default, SegAligner extracts a total of 500 scores from the scoring matrices, the number of scores for each entry in the reference proportional to its length divided by the total length of the reference genome, multiplied by the number of scores desired to build a scoring distribution (N = 500). That is, the number of scores to extract for a given scoring matrix n_i is given by

$$n_i = \frac{|c_i|}{\sum_j |c_j|} N$$

where $|c_i|$ is the size of the current reference genome entry (in bp), and *N* is the total number of scores we wish to have in our distribution of scoring matrices.

6. Parameterized Scoring Model

We next propose a parametrized scoring function based on a maximum likelihood approach, which would serve as a substitute for our fixed scoring model. We have yet to resolve problems related to numerical underflow which arise when this solution is implemented, however, it is a theoretically more sensitive model than our current approach. We begin by defining some variables.

- R_k be the *k*-th matching region on contig *R*. Its size is denoted s_r.

- Q_k is the *k*-the matching region on in-silico genomic segment Q. Its size is denoted s_q.
- λ_R indicates the per-base rate of observed labels on the assembled contigs.
- λ_Q indicates the per-base rate of observed labels in the reference genome. This quantity is determined empirically.

$$\lambda_R = mean(j - i, \forall\, i, j \in R, i < j, \forall R \in contigs)$$

$$\lambda_Q = mean(q - p, \forall\, p, q \in Q, p < q, \forall Q \in G)$$

- n_r is the number of unaligned labels on R_k ("false-positives" w.r.t. Q_k).
- n_q is the number of unaligned labels on Q_k ("false-negatives w.r.t R_k).
- f_n is the per-base false-negative label rate on the assembled contig.
- f_p is the per-base false-positive label rate on the assembled contig.
- σ is the standard deviation of the measurement error between labels on a matching region on an assembled contig and a matching region from the reference genome.

We can develop a likelihood ratio to describe the likelihoods that either a matching region constitutes a true alignment or is simple due to random chance. We define the true alignment hypothesis as θ_1 and the random alignment hypothesis as θ_0. For two matching regions, R_k, Q_k the following variables govern the likelihood of each hypothesis; s_r, s_q, n_r, n_q. The likelihood function is further conditioned upon $f_n, f_p, \lambda_R, \lambda_Q, \sigma$. Thus, we have

$$\mathcal{L}(\theta | s_r, s_q, n_r, n_q, f_n, f_p, \lambda_r, \lambda_q, \sigma) = f_\theta(s_r, s_q, n_r, n_q | f_n, f_p, \lambda_r, \lambda_q, \sigma)$$

Thus the likelihood ratio is

$$\Lambda(s_r, s_q, n_r, n_q, f_r, f_p, \lambda_r, \lambda_q, \sigma) = \frac{\mathcal{L}(\theta_0 | s_r, s_q, n_r, n_q, f_n, f_p, \lambda_r, \lambda_q, \sigma)}{\mathcal{L}(\theta_1 | s_r, s_q, n_r, n_q, f_n, f_p, \lambda_r, \lambda_q, \sigma)}$$

$$= \frac{f_{\theta_0}(s_r, s_q | n_r, n_q, f_n, f_p, \lambda_r, \lambda_q, \sigma)}{f_{\theta_1}(s_r, s_q, n_r, n_q | f_n, f_p, \lambda_r, \lambda_q, \sigma)}$$

We re-write the likelihood function f_θ as a probability density function. Using the following derivation, we can show that for each hypothesis the likelihood ratio can be written as

$$\frac{f_{\theta_0}}{f_{\theta_1}} = \frac{P(s_r | n_r, \lambda_r) P(s_q | n_q, \lambda_q)}{P(s_q | \lambda_q) P(s_r | s_q, \sigma) P(n_r | s_r, f_p) P(n_q | s_r, f_n)}$$

In an implementation of this method, the scoring function would become

$$Score(R_K, Q_K) = -\log(\frac{f_{\theta_0}}{f_{\theta_1}})$$

Derivation of f_{θ_0}

Under the null-hypothesis, the matching regions R_K, Q_K, do not correspond genomically, which implies that their sizes are independent.

$$P(s_r, s_q) = P(s_q) P(s_r)$$

Furthermore, all interior labels (n_q & n_r) in the matching regions are assumed to be real under the null hypothesis.

We assume the distances between labels in both reference and assembled matching regions are exponentially distributed. The probability of observing a matching region of size s_r composed of the sum of $n_r + 1$ exponential random variables is given by a special case of the gamma distribution; the Erlang distribution.

$$P(s_r | n_r + 1, \lambda_r) = \frac{e^{-\lambda_r s_r} \lambda_r^{(n_r+1)} s_r^{n_r}}{n_r!}$$

Similarly, for the probability of observing a matching region of size s_q, composed of $n_q + 1$ exponential random variables is given by

$$P\left(s_q\middle|n_q+1,\lambda_q\right)=\frac{e^{-\lambda_q s_q}\lambda_q^{(n_q+1)}s_q^{\,n_q}}{n_q!}$$

Thus

$$f_{\theta_0}=P(s_r|n_r+1,\lambda_r)P\left(s_q\middle|n_q+1,\lambda_q\right)$$

Derivation of f_{θ_1}

Under the alternative-hypothesis, θ_1, s_r and s_q are no longer independent as they belong to corresponding genomic locations. n_q & n_r represent the false-negative & false-positive labels observed on the contig matching region with respect to the reference. These quantities are still independent as they are created by independent processes.

$$f_{\theta_1}\ = P_{\theta_1}\left(s_r,s_q,n_r,n_q\middle|f_n,f_q,\lambda_r,\lambda_q,\sigma\right)$$

$$= P_{\theta_1}\left(s_q\middle|f_n,f_q,\lambda_r,\lambda_q,\sigma\right)P_{\theta_1}\left(s_r\middle|s_q,f_n,f_q,\lambda_r,\lambda_q,\sigma\right)P_{\theta_1}\left(n_r\middle|s_r,s_q,f_n,f_q,\lambda_r,\lambda_q,\sigma\right)\ldots$$

$$P_{\theta_1}\left(n_q\middle|n_r,s_r,s_q,f_n,f_q,\lambda_r,\lambda_q,\sigma\right)\ \text{(Product rule)}$$

$$= P_{\theta_1}\left(s_q\middle|\lambda_q\right)P_{\theta_1}\left(s_r\middle|s_q,\sigma\right)P_{\theta_1}\left(n_r\middle|s_r,f_p\right)P_{\theta_1}\left(n_q\middle|s_r,f_n\ \right)\ \text{(Conditional independence)}$$

Thus

$$f_{\theta_1}=P_{\theta_1}\left(s_q\middle|\lambda_q\right)P_{\theta_1}\left(s_r\middle|s_q,\sigma\right)P_{\theta_1}\left(n_r\middle|s_r,f_p\right)P_{\theta_1}\left(n_q\middle|s_r,f_n\ \right)$$

Calculation of terms in f_{θ_1}

Next we derive how to calculate each of the four probability terms in the likelihood ratio under each hypothesis.

- $P_{\theta_1}\left(s_q\middle|\lambda_q\right)$.

  The distribution of s_q given the label density in Q follows an exponential distribution.

$$P_{\theta_1}(s_q|\lambda_q) = \lambda_q e^{-\lambda_q s_q}$$

- $P_{\theta_1}(s_r|s_q,\sigma)$

  We calculate $P_{\theta_1}(s_r|s_q,\sigma)$ by assuming that the error model for a given s_r that has reference matching region of size s_q follows $\mathcal{N}(s_q, \sigma^2)$. This implies

$$P_{\theta_1}(s_r|s_q,\sigma) = \frac{1}{\sqrt{2\pi\sigma^2}} exp\left(\frac{-(s_r - s_q)^2}{2\sigma^2}\right)$$

- $P_{\theta_1}(n_r|s_r,f_p)$

  This term is modeled as a Poisson process with shape parameter $\lambda = s_r f_p$. Thus

$$P_{\theta_1}(n_r|s_r,f_p) = \frac{\lambda^{n_r}}{n_r!} e^{-\lambda}$$

- $P_{\theta_1}(n_q|s_r,f_n)$

  This term is modeled as a Poisson process with shape parameter $\lambda = s_r f_n$. Thus

$$P_{\theta_1}(n_q|s_r,f_n) = \frac{\lambda^{n_q}}{n_q!} e^{-\lambda}$$

Appendix

Table 1. Properties of the BioNano technology for two comparable sequencing experiments. The columns indicate the property described for each instrument type and the sample name is listed next to the instrument type. Saphyr data tends to produce much larger contigs, spanning chromosome arms, though there are still many small contigs.

Feature	BioNano Irys (NA12878)	BioNano Saphyr (HCC827)
Molecule coverage	89x	142x
Measured Label Density	10.1/100kb	17.3/100kb
Reference Label Density	12.1/100kb	18.5/100kb
Molecule N50	274.4 kb	263.9 kb
Molecule Average Length	278.0 kb	257.3 kb
Contig N50	4.2 Mb	46.6 Mb
Contig Average Length	2.4 Mb	5.0 Mb

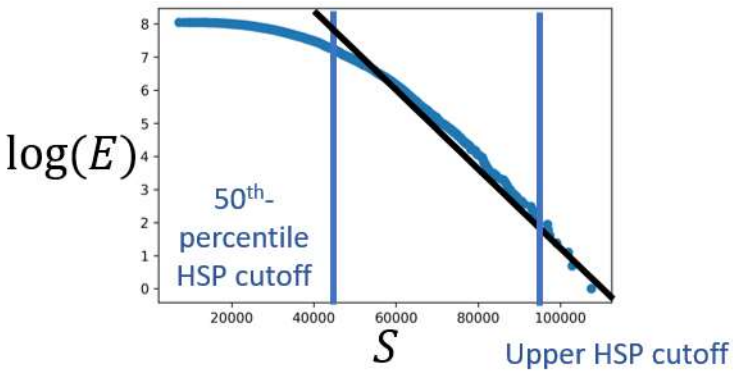


Figure 1. Parametrizing the E-value model using linear regression. Random best-scoring alignments for use in the model are included if they are above the 50th-percentile of the distribution. An upper HSP cutoff is also set to remove potential true alignments from skewing the random model.

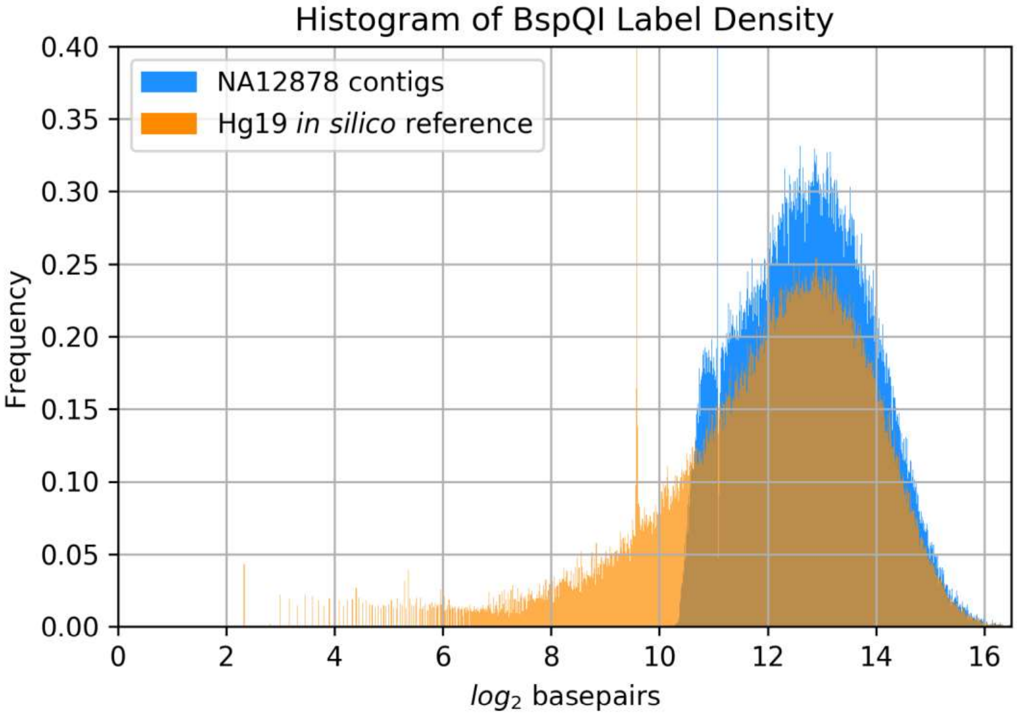


Figure 2. BspQI nickase labeling densities for measured data (NA12878) and reference in-silico data (Hg19).

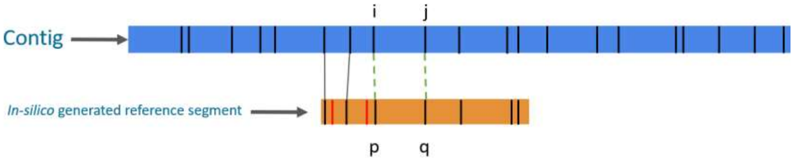
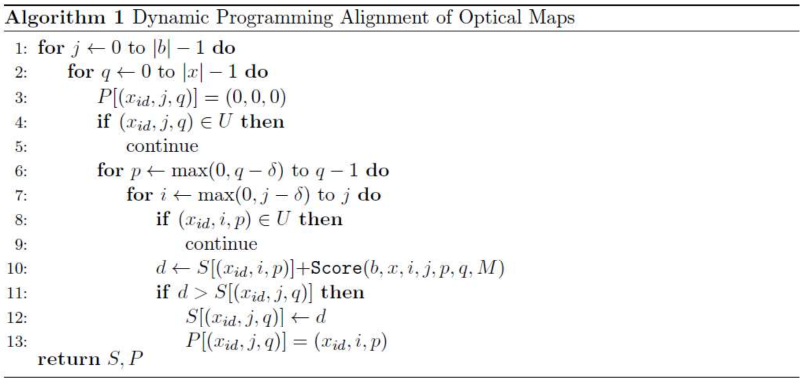


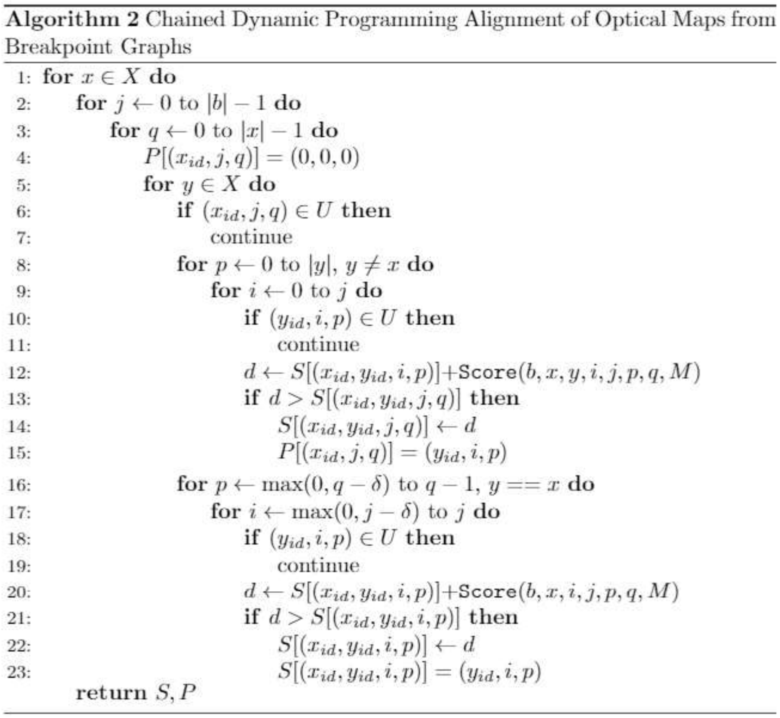
Figure 3. Representation of a partially aligned pair of optical maps. The matching region is indicated with dotted green.



Algorithm 1 Dynamic Programming Alignment of Optical Maps

```
for $j \leftarrow 0$ to $|b| - 1$ do
    for $q \leftarrow 0$ to $|x| - 1$ do
        $P[(x_{id}, j, q)] = (0, 0, 0)$
        if $(x_{id}, j, q) \in U$ then
            continue
        for $p \leftarrow \max(0, q - \delta)$ to $q - 1$ do
            for $i \leftarrow \max(0, j - \delta)$ to $j$ do
                if $(x_{id}, i, p) \in U$ then
                    continue
                $d \leftarrow S[(x_{id}, i, p)] + \texttt{Score}(b, x, i, j, p, q, M)$
                if $d > S[(x_{id}, j, q)]$ then
                    $S[(x_{id}, j, q)] \leftarrow d$
                    $P[(x_{id}, j, q)] = (x_{id}, i, p)$
return $S, P$
```

Figure 4. Dynamic programming alignment of optical maps. *b* refers to the list of label positions on an assembled contig. *P* stores references for backtracking. *U* stores previously used pairings in prior alignments so that this method can be used to identify multiple different alignment. Lowercase delta refers to a lookback threshold, so that the alignment may be banded and sped up. *S* is the dynamic programming scoring matrix.



Algorithm 2 Chained Dynamic Programming Alignment of Optical Maps from Breakpoint Graphs

```
for $x \in X$ do
    for $j \leftarrow 0$ to $|b| - 1$ do
        for $q \leftarrow 0$ to $|x| - 1$ do
            $P[(x_{id}, j, q)] = (0, 0, 0)$
            for $y \in X$ do
                if $(x_{id}, j, q) \in U$ then
                    continue
                for $p \leftarrow 0$ to $|y|$, $y \neq x$ do
                    for $i \leftarrow 0$ to $j$ do
                        if $(y_{id}, i, p) \in U$ then
                            continue
                        $d \leftarrow S[(x_{id}, y_{id}, i, p)] + \texttt{Score}(b, x, y, i, j, p, q, M)$
                        if $d > S[(x_{id}, y_{id}, j, q)]$ then
                            $S[(x_{id}, y_{id}, j, q)] \leftarrow d$
                            $P[(x_{id}, j, q)] = (y_{id}, i, p)$
                for $p \leftarrow \max(0, q - \delta)$ to $q - 1$, $y == x$ do
                    for $i \leftarrow \max(0, j - \delta)$ to $j$ do
                        if $(y_{id}, i, p) \in U$ then
                            continue
                        $d \leftarrow S[(x_{id}, y_{id}, i, p)] + \texttt{Score}(b, x, i, j, p, q, M)$
                        if $d > S[(x_{id}, y_{id}, i, p)]$ then
                            $S[(x_{id}, y_{id}, i, p)] \leftarrow d$
                            $S[(x_{id}, y_{id}, i, p)] = (y_{id}, i, p)$
    return $S, P$
```

Figure 5. Chained alignment for multiple reference segments in a dynamic programming strategy. Extending the same variables from Algorithm 1, *X* is a collection of reference segments, which are linked.

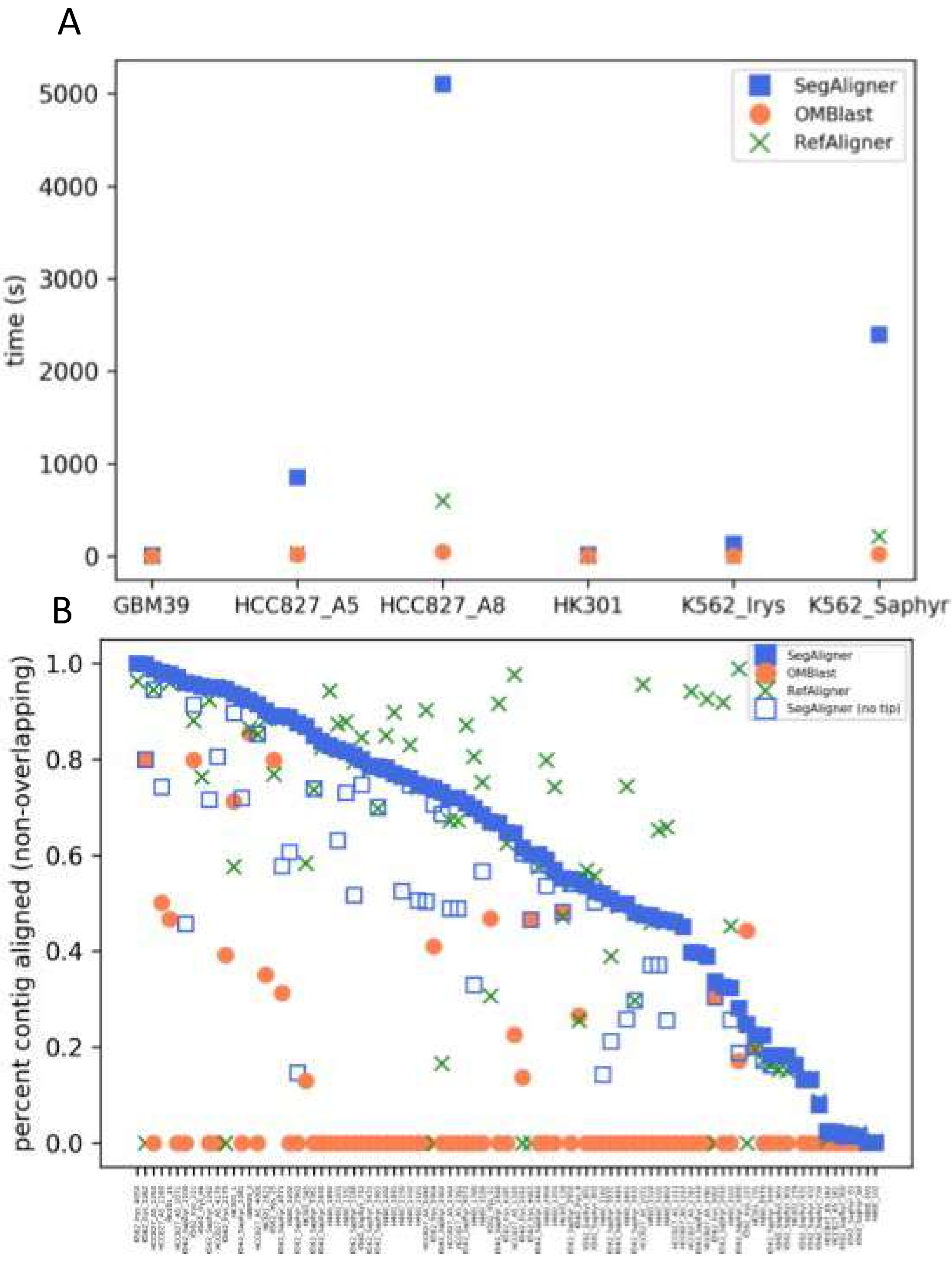


Figure 6. A. Runtimes for commonly used optical map alignment tools and SegAligner on AmpliconArchitect identified amplicons in four cell lines with cytogenetically validated amplicons. B. Percentage of contig aligned for assembled contigs in the cell lines from A. SegAligner finds generally more alignments for segments with contigs.

References

Altschul, S. F., Gish, W., Miller, W., Myers, E. W., & Lipman, D. J. (1990). Basic local alignment search tool. *Journal of Molecular Biology*, *215*(3), 403–410. https://doi.org/10.1016/S0022-2836(05)80360-2

Anantharaman, T. S., Mishra, B., & Schwartz, D. C. (1997). Genomics via Optical Mapping II: Ordered Restriction Maps. *JOURNAL OF COMPUTATIONAL BIOLOGY*, *4*(2), 91–118. Retrieved from http://online.liebertpub.com/doi/pdfplus/10.1089/cmb.1997.4.91

Deshpande, V., Luebeck, J., Nguyen, N.-P. D., Bakhtiari, M., Turner, K. M., Schwab, R., … Bafna, V. (2019). Exploring the landscape of focal amplifications in cancer using AmpliconArchitect. *Nature Communications*, *10*(1), 392. https://doi.org/10.1038/s41467-018-08200-y

Hastie, A. R., Dong, L., Smith, A., Finklestein, J., Lam, E. T., Huo, N., … Xiao, M. (2013). Rapid Genome Mapping in Nanochannel Arrays for Highly Complete and Accurate De Novo Sequence Assembly of the Complex Aegilops tauschii Genome. *PLoS ONE*, *8*(2), e55864. https://doi.org/10.1371/journal.pone.0055864

Karlin, S., & Altschul, S. F. (1990). Methods for assessing the statistical significance of molecular sequence features by using general scoring schemes. *Proceedings of the National Academy of Sciences of the United States of America*, *87*(6), 2264–2268. Retrieved from http://www.ncbi.nlm.nih.gov/pubmed/2315319

Leung, A. K.-Y., Kwok, T.-P., Wan, R., Xiao, M., Kwok, P.-Y., Yip, K. Y., & Chan, T.-F. (2016). OMBlast: alignment tool for optical mapping using a seed-and-extend approach. *Bioinformatics*, btw620. https://doi.org/10.1093/bioinformatics/btw620

Muggli, M. D., Puglisi, S. J., & Boucher, C. (2014). Efficient Indexed Alignment of Contigs to Optical Maps (pp. 68–81). Springer, Berlin, Heidelberg. https://doi.org/10.1007/978-3-662-44753-6_6

Nagarajan, N., Read, T. D., & Pop, M. (2008). Scaffolding and validation of bacterial genome assemblies using optical restriction maps. *Bioinformatics (Oxford, England)*, *24*(10), 1229–1235. https://doi.org/10.1093/bioinformatics/btn102

Pendleton, M., Sebra, R., Pang, A. W. C., Ummat, A., Franzen, O., Rausch, T., … Bashir, A. (2015). Assembly and diploid architecture of an individual human genome via single-molecule technologies. *Nature Methods*, *12*(8), 780–786. https://doi.org/10.1038/nmeth.3454

Valouev, A., Li, L., Liu, Y.-C., Schwartz, D. C., Yang, Y., Zhang, Y., & Waterman, M. S. (2006). Alignment of Optical Maps. *JOURNAL OF COMPUTATIONAL BIOLOGY*, *13*(2), 442–462. Retrieved from http://online.liebertpub.com/doi/pdfplus/10.1089/cmb.2006.13.442

Waterman, M. S., Smith, T. F., & Katcher, H. L. (1984). Algorithms for restriction map comparisons. *Nucleic Acids Research*, *12*(1Part1), 237–242. https://doi.org/10.1093/nar/12.1Part1.237