Fast and Accurate Alignment of Barcoded Reads

by

Ariya Shajii

Submitted to the Department of Electrical Engineering and Computer Science
in partial fulfillment of the requirements for the degree of

Master of Science

at the

MASSACHUSETTS INSTITUTE OF TECHNOLOGY

June 2018

Author Signature redacted ....
Department of Electrical Engineering and Computer Science
May 11, 2018

Certified by.. Signature redacted ....
Bonnie Berger
Professor of Electrical Engineering and Computer Science
Thesis Supervisor

Accepted by....... Signature redacted ....
Leslie A. Kolodziejski
Professor of Electrical Engineering and Computer Science
Chair, Department Committee on Graduate Students

Fast and Accurate Alignment of Barcoded Reads

by

Ariya Shajii

Submitted to the Department of Electrical Engineering and Computer Science
on May 11, 2018, in partial fulfillment of the
requirements for the degree of
Master of Science

Abstract

Over the last few years, we have seen the emergence of several so-called "third-generation" sequencing platforms, which improve on standard short-read sequencing that has thus far been at the center of next-generation sequencing. While technologies developed by Pacific Biosciences and Oxford Nanopore accomplish this goal by producing physically longer reads, several other technologies take an alternate route by instead producing "barcoded reads", including 10x Genomics' Chromium platform and Illumina's TruSeq Synthetic Long-Read platform. With barcoded reads, long-range information is captured by the barcodes, which identify source DNA fragments. As with all sequencing data, alignment of barcoded reads is the first step in nearly all analyses, and therefore plays a central role. Here, we design and validate improved alignment algorithms for barcoded sequencing data, which enable improved downstream analyses like phasing and genotyping, and additionally uncover variants in regions containing nearby homologous elements that go undetected by other methods.

Thesis Supervisor: Bonnie Berger
Title: Professor of Electrical Engineering and Computer Science

Acknowledgments

I would like to thank Ibrahim Numanagić and my advisor Bonnie Berger for their invaluable help and guidance throughout this project, as well as the rest of the Berger lab. The content of this thesis is largely an extension of our work in [36]. I also thank Chris Whelan, Chad Nusbaum, Eric Banks, as well as the rest of the GATK SV Group from the Broad Institute for providing me with data samples and many valuable suggestions.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

Sequencing has revolutionized the way we approach biology, genealogy and medicine, as well as a host of other domains. The development of high-throughput sequencing technologies—called "next-generation sequencing", or NGS—has spurred the development of numerous algorithms to better handle the ensuing data deluge. As sequencing is the most fundamental operation in genomics, transcriptomics, and metagenomics, efficient and accurate algorithms for handling and analyzing sequencing data are pivotal to these fields, and many others. Virtually all analysis pipelines for any kind of sequencing data begin with *alignment*, or *mapping*, of reads (the A/C/G/T DNA strings produced by sequencers) to a reference genome (a fixed consensus sequence that roughly corresponds to the average genome, initially constructed via the Human Genome Project [7]). Reads have typically been on the order of 100 base pairs (bp) in length, while the human genome is roughly 3 Gbp. The large size of the human genome, coupled with the fact that reads can have sequencing errors, single-nucleotide mutations and even insertions or deletions, makes sequence alignment an algorithmically nontrivial task, and one in which there has been extensive research [22, 20, 40, 41, 25].

Now we are experiencing a similar phenomenon again as several "third-generation" [15] sequencing technologies are being developed, giving us a unique opportunity to rethink and reinvent many of the algorithms designed for standard NGS data to handle the new data types that third-generation platforms produce. The primary motivation behind third-generation sequencing is to address the shortcomings of standard NGS, the biggest of which is read length. NGS platforms have typically produced "short-reads" that are around 100 bp in length, but this short read length has limitations in many applications:

- *Alignment:* A significant fraction of the genome is comprised of repetitive or homologous elements. Moreover, these elements are often larger than the length of a short-read, making it difficult or impossible to resolve them with short-reads. Recent segmental duplications larger than 5 kbp and with over 94% sequence identity comprise roughly 4.35% of the genome; furthermore, these regions are phenotypically important [3].

- *Structural variation detection:* Structural variations are large-scale genomic events such as duplications, insertions/deletions, inversions and translocations. These events can be impossible to detect with short-reads alone (often for the same reasons as above) [8].

- *Phasing:* Phasing is the process of separating a sequenced individual's mutations into maternal and paternal haplotypes, which entails determining which mutations are on the same haplotype, often without the aid of sequencing data from the parents. As short-reads typically intersect at most one mutation, it can be difficult or impossible to determine which mutations are on the same haplotype [35].

Worth noting is the fact that short-read sequencing has been adapted to overcome some of these obstacles by employing "paired-end sequencing", where two ends of a roughly 1 kb fragment are sequenced to produce *pairs* of short reads that are known to map relatively close to one another, thereby increasing the effective read length. Nevertheless, the aforementioned downsides all still apply to paired-end reads.

Third-generation sequencing technologies address the issue of read length in two main ways. The sequencing platforms of Pacific Biosciences and Oxford Nanopore produce physically much longer reads, on the order of tens of kilobases [15]. On

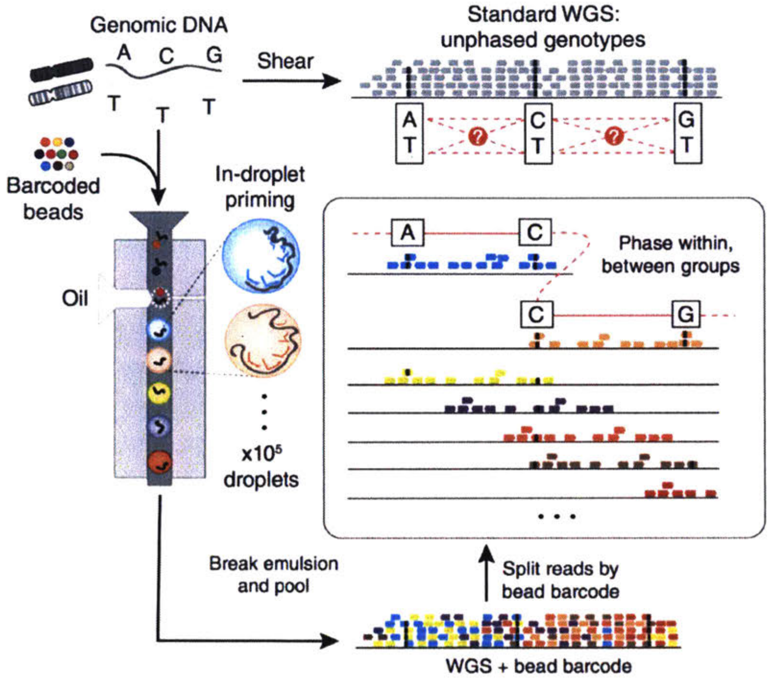


Figure 1-1: 10x Genomics' sequencing workflow; taken from Kitzman [19].

the other hand, 10x Genomics' Chromium platform produces "barcoded" short-reads (also called "linked-reads"), which capture long-range information by virtue of short barcodes ligated to the start of the reads. In particular, linked-read sequencing works as follows:

1. Long 10–200 kb DNA fragments are partitioned into droplets such that few fragments are present in each droplet.

2. The long fragments in each droplet are sheared into pieces with lengths on the order of several hundred bases.

3. A barcoded bead with a unique, known 16 bp barcode is added to each droplet, thereby ligating the barcode to the start of each sheared piece.

4. Standard paired-end short-read sequencing is applied to the newly barcoded pieces, producing barcoded short-reads.

This process is illustrated in Figure 1-1. Notice that, in barcoded read sequencing, a read's barcode provides a link to the original source fragment from which the read is derived, which has several implications. Firstly, since all reads with a given barcode came from the same source fragment (or, at the very most, few source fragments), these reads should align near one another; this property enables alignment to regions inaccessible with short-reads alone, since it is now possible to discern between multiple alignments of any single read by considering its surrounding reads. Secondly, since these reads all came from a single fragment, they are naturally on the same haplotype, and so any overlapping mutations can now be trivially phased (as shown in Figure 1-1). There are several technologies other than 10x's that employ a similar barcoded sequencing paradigm, including Illumina's TruSeq Synthetic Long-Read [27] and Complete Genomics' Long Fragment Read [28] technologies. Even standard paired-end sequencing is arguably an instance of barcoded read sequencing, since the two mates of each pair are sequenced from the same unknown longer fragment.

1.1 Alignment and *meta alignment*

Read mapping has traditionally been an algorithmically easy-to-define problem: for each read (which until recently would be on the order of at most a few hundred base pairs), locate the position on some genome to which it aligns such that some metric is minimized, be it Levenshtein distance, more general edit distance, Hamming distance or some other related metric (or perhaps not a metric at all in the mathematical sense). To achieve this aim, virtually all read mappers adhere to some form of the "seed and extend" paradigm, which generally proceeds in two steps:

- **Seed:** Short exact or nearly-exact matches ("seeds") of subsequences of the read (often called k-mers for length-k subsequences) are found in the genome (which is usually indexed in some way to facilitate this process).

- **Extend:** Wherever several colinear seeds are found, a proper end-to-end alignment of the read and the corresponding genomic region is performed.

Evidently, both of these steps can be approached in a number of ways, and a multitude of useful heuristics have been developed for each.

A major point of deviation between algorithms in the seed step, for example, is exactly how to index to the genome so as to find the seeds quickly. One class of algorithms (e.g. tools like BWA [22] and Bowtie2 [21]) makes use of an FM-index data structure [13], which is based on the Burrows-Wheeler transform, for finding seeds. FM-indices are largely appealing for their low memory footprint; for example, an FM-index of the entire human genome can be stored in around 5 GB or so, despite the human genome itself being 3 GB. A second class uses hash tables to map seeds to positions in the genome (e.g. tools like SNAP and CORA); these approaches typically use substantially more memory but are much faster [41, 40]. Beyond these, different algorithms use various heuristics for choosing seeds, spacing seeds (i.e. seeds do not necessarily need to be contiguous, and it has been shown that non-contiguous or "spaced" seeds can actually be superior [23]), when to "extend" and so on. The extend step is typically comprised of some form of dynamic programming for aligning the read to the genomic region found in the seed step, such as Needleman-Wunsch or Smith-Waterman. As it is a key kernel in most alignment algorithms, much work has been done on developing highly-optimized dynamic programming alignment code, often using SIMD instructions to boost performance [42]. CORA [40] can additionally skip the extension phase entirely in many cases by clever preprocessing of the reference genome.

Hence, we have an arsenal of tools and algorithms for efficient base-level alignment in that, given a (short) DNA sequence, we can efficiently locate its "optimal" alignment within a larger reference genome. Nevertheless, additional complications arise as sequencing technologies continue to advance, and to provide us with richer data types that go beyond just sequence. For instance, virtually all short-read sequencing technologies nowadays produce paired-end reads (i.e. two short reads that are known to have been sequenced from a single longer DNA fragment); we can align the individual reads of each pair ("mates") using our framework above, but how do we incorporate the knowledge that the reads are in fact part of a pair? Most alignment tools employ a set of heuristics for handling paired-end data, by penalizing alignments where the mates map too far or too close to one another given the distribution of insert sizes between them. This approach has worked well for paired-end data, where we only need to reconcile alignments for two reads at a time. For barcoded read sequencing technologies like 10x Genomics' or Illumina's TruSeq Synthetic Long-Read platform, however, we must account for many more reads at a time, and therefore require more principled approaches.

We can think of this process of incorporating the additional information given by our sequencing data type into the alignment process as a kind of *meta alignment*; we still need base-level alignments as a starting point, but we augment them with this additional information (e.g. barcodes) to produce more accurate results. The framework for barcoded read alignment presented here is an instance of this idea.

1.2 Third-generation sequencing

As sequencing technologies continue to advance beyond the initial introduction of next-generation sequencing (NGS), we have begun to see the emergence of so-called "third-generation" sequencing platforms, which seek to improve on the standard short-read sequencing that has thus far been at the heart of most next-generation sequencing [26]. Several organizations are at the center of this new sequencing revolution, including Pacific Biosciences [11], Oxford Nanopore [39] and 10x Genomics [43]. While the former two have developed sequencing technologies that produce much longer physical reads (e.g., 10kb–200kb) at typically higher error rates, the latter is an example of a barcoded sequencing technology, which typically produce short-reads (up to 300bp) with low error rates [15].

At a high level, barcoded sequencing is any sequencing method where long DNA fragments are sheared, and the sheared pieces have some identifier ("barcode") relating them back to the source fragment. These barcodes can be explicit (a physical barcode is ligated to each sheared piece, e.g. as in 10x sequencing) or implicit (the fragments are distributed to identifiable wells, e.g. as in Illumina's TruSeq Synthetic Long-Read sequencing, henceforth referred to as TruSeq SLR). These sheared pieces are then sequenced using standard short-read sequencing, thereby producing barcoded short-reads (Figure 2-1a). Other barcoded seqeuncing technologies include Illumina's Continuity Preserving Transposition technology (CPT-seq), Complete Genomics' Long Fragment Read technology, Drop-seq, and CEL-Seq2 [43, 27, 24, 16, 1]. Because they help identify the original source fragment, these barcodes implicitly carry long-range information, which can have a significant impact on alignment and many downstream analyses such as structural variation [43] detection and phasing.

Barcoded reads have several advantages over physically long reads. Firstly, and perhaps most importantly, barcoded read sequencing is substantially cheaper than long-read sequencing as discussed above; whereas PacBio's and Oxford Nanopore's sequencing platforms currently cost anywhere from $750–$1000 per GB of data, barcoded sequencing is a comparatively cheaper add-on to standard short-read sequencing, and therefore bears the same cost (e.g., 10x sequencing costs $30 per GB plus a $500 overhead per sample) [15]. Secondly, the error profile of barcoded reads is very similar to that of standard short-reads (roughly 0.1% substitution errors), which enables us to augment the tools and algorithms that have been developed for regular short-reads to handle their barcoded counterparts. By contrast, long-read sequencing (e.g. PacBio or Nanopore) typically produces high rates of erroneous indels (ranging from 12–13%), which presents a challenge when trying to use preexisting algorithms. These differences are summarized in Table 1.1. Beyond these advantages, barcoded reads are compatible with doing hybrid-capture exome sequencing, where introns are not sequenced (which is not possible with long-read sequencing technologies, as a contiguous long-read cannot only sequence the exons in a gene). This is a very substantial additional cost-advantage for barcoded reads (exome sequencing can mean a 10–20 fold reduction in sequencing cost over whole genome sequencing) [33]. These and other benefits have led to the recent proliferation of barcoded sequencing technologies for various use cases; for example, 10x and TruSeq SLR sequencing for whole genome sequencing, as well as Drop-seq and CEL-seq2 for single-cell RNA-seq, 10x for single-cell VDJ sequencing and so on [43, 27, 24, 16, 1]. A comprehensive review of many of these methods is available [44]. Furthermore, barcoded sequencing is also playing a greater role in downstream applications such as the generation of

Technology	Error profile	Cost per GB
pb	13% indel	$1000
○	12% indel	$750
X	0.1% subst.	$30

Table 1.1: Comparison of PacBio's (top), Oxford Nanopore's (middle) and 10x Genomics' (bottom) sequencing technologies [15]. Note that 10x also has a $500 per-sample overhead.

transcriptomic profiles [5].

As with virtually all sequencing data, the first step in the analysis pipeline for barcoded reads is typically alignment. While barcoded reads can, in theory, be aligned by a standard short-read aligner (e.g., CORA [40], BWA [22], Bowtie2 [21]), this would fail to take advantage of the information provided by the barcodes. An alternative approach [27] is to assemble the reads for each particular barcode and to treat the result as a single "synthetic long-read". While this strategy works well for technologies like TruSeq SLR, in which source fragments are generally sequenced with high coverage, it is not practical when fragments are shallowly sequenced as with 10x, which achieves high coverage not by having high per-barcode coverage but rather by having many barcodes. Also worth noting is the fact that TruSeq SLR's sequencing fragments at high coverage inflates their sequencing costs to be on par with PacBio's and Oxford Nanopore's, whereas 10x circumvents this high cost via shallow fragment sequencing [15].

Currently, the state-of-the-art in terms of barcoded read alignment employs "read clouds"—groups of reads that share the same barcode and map to the same genomic region—to choose the most likely alignment from a set of candidate alignments for each read [4]. Intuitively, read clouds represent the possible source fragments from which the barcoded reads are derived. The read cloud approach to alignment effectively begins with a standard all-mapping to a reference genome to identify these clouds, followed by an iterative update of reads' assignments to one of their possible alignments, guided by a Markov random field that is used to evaluate the probability of a given read-to-cloud configuration (taking into account the alignment scores, clouds, etc.). Notably, in this framework, clouds are inherently fixed entities to which some number of reads are assigned at any given point, which does not take into account the fact that reads can have suitable alignments in several different clouds. Since this information can be valuable in downstream analyses like genotyping, phasing, and structural variation detection, we wish to account for it.

Confounding barcoded read alignment is the fact that multiple fragments can share the same barcode; it is in general not possible to infer the source fragment of a read (and thus its correct alignment within a reference genome) merely by looking at its barcode. In order to deduce the correct placement of a read, and thus its unknown source fragment, all possible alignments of that read need to be examined. Even then, it can be difficult to determine the correct alignment, particularly in homologous regions of the genome that result in multi-mappings within a single cloud.

Here, we propose a general paradigm for barcoded read alignment that newly employs a probabilistic interpretation of clouds: EMerAld, or EMA for short (Figure 2-1). Our two-tiered statistical binning approach enables the more accurate placement of reads in and within read clouds, which is the critical step in barcoded read alignment. The two tiers consist of: (i) a novel latent variable model to probabilistically assign reads to clouds, which introduces the notion of clouds as distributions over generated reads rather than simply fixed groups of reads; and (ii) newly exploiting expected read coverage (read density) to resolve the difficult case of multiple alignments of reads *within* clouds. The idea and subsequent observation of the fixed read density distribution within source fragments is novel to EMA and can be utilized by many barcoded read analysis tools: for example, an assembler might use our idea to model the distance between reads within the same source fragment and thus break ties, if any. Note that these ambiguous alignments account for a large fraction of the rare variants that currently cannot be resolved and are of great interest to biologists [17, 34, 12].

Chapter 2

A New Framework for Aligning Barcoded Reads

General barcoded read sequencing begins with splitting the source DNA into long fragments (10–200kb) where each such fragment is assigned some barcode (e.g. a short 16bp DNA sequence in 10x sequencing). These fragments are sheared and each sheared piece has the assigned barcode ligated to it (or, alternatively, resides in an identifiable well), whereupon standard short-read sequencing is applied to the sheared pieces. As a result, barcoded reads have the same low error rates as typical Illumina whole-genome sequencing reads. An idealization of this process is illustrated in Figure 2-1a.

2.1 Standard data preprocessing

The first stage in the alignment process is to preprocess the data and to identify the barcodes. Currently, EMA uses an in-house 10x barcode preprocessor, which extracts and corrects the barcodes from the raw data. Data from many other barcoded read technologies (e.g. TruSeq SLR) can be preprocessed in a more straightforward manner, as the barcodes are given as well identifiers for each read, meaning the preprocessing stage consists of a simple demultiplexing step.

For 10x data preprocessing we largely follow the same practices used by 10x Genomics' WGS software suite, Long Ranger [14]. The purpose of this preprocessing is to:

- extract the barcode from the read sequence,
- error-correct the barcode based on quality scores and a list of known barcode sequences,
- and group reads by barcode into "barcode buckets" to enable parallelism during alignment.

In summary, in the barcode extraction stage, we remove the 16bp barcode from the first mate of each read pair, and trim an additional 7bp to account for potential ligation artifacts resulting from the barcode ligation process during sequencing (the second mate shares the same barcode as the first mate). Subsequently, we compare each barcode to a list B of known barcodes to produce a per-barcode count, and compute a prior probability for each known barcode based on these counts. Note that this list is designed such that no two barcodes are Hamming-neighbors of one another. Now for each barcode b not appearing in B, we examine each of its Hamming-1 neighbors b' and, if b' appears in B, compute the probability that b' was the true barcode based on its prior and the quality score of the changed base. Similarly, for each b appearing in B, we consider each Hamming-2 neighbor b' and compute the probability that b' was the true barcode in an analogous way (to account for the possibility that two errors changed the actual barcode to another also in B). Lastly, we employ a probability cutoff on the barcodes, and thereby omit the barcodes of reads that do not meet this cutoff. Any read not carrying a barcode after this stage is aligned with a standard WGS mapper such as CORA [40] or BWA [22].

While in standard read alignment parallelism can be achieved at the read-level, for barcoded read alignment we can only achieve parallelism at the barcode-level. Therefore, the last preprocessing step is to group reads by barcode into some number of buckets. Each such bucket contains some range of barcodes from B, which are all grouped together within the bucket. This enables us to align the reads from each bucket in parallel, and to merge the outputs in a post-processing step.

We note that the Hamming-2 search takes a substantial fraction of the total time, but is often unnecessary: on a large 980GB 10x dataset, only 276 out of almost 1.5 billion reads are affected by the Hamming-2 correction (amounting to $< 0.0001\%$ overall effect). Thus, it is safe to skip the Hamming-2 correction step. Nevertheless, we applied Hamming-2 correction on all our datasets for the sake of consistency with Lariat. Finally, EMA offers a parallelized barcode correction implementation, which significantly speeds up the overall pipeline.

2.2 A new model

Here we employ a latent variable model for determining the optimal assignment of reads to their possible clouds. A "cloud" is defined to be a group of nearby alignments of reads with a common barcode, thereby representing a possible source fragment [4]. We consider all the reads for an individual barcode simultaneously, all-mapping and grouping them to produce a set of clouds for that barcode (Figure 2-1b). The clouds are deduced from the all-mappings by grouping any two alignments that are on the same chromosome and within 50kb of one another into the same cloud, which is the same approach employed by Lariat (for TruSeq SLR or CPT-seq data, we use 15kb as a cutoff; this is a tuneable parameter that can be adjusted depending on the underlying technology). While this heuristic works well in the majority of cases, it can evidently run into issues if, for example, a single read aligns multiple times to the same cloud. We address such cases below, but assume in the subsequent analysis that clouds consist of at most one alignment of a given read.

As notation, we will denote by c the set of alignments contained in a given cloud. We restrict our analysis to a single set of clouds $\mathcal{C} = \{c_1, c_2, \ldots, c_n\}$ that corresponds to a connected component in the disjoint-set over clouds induced by alignments, as shown in Figure 2-1b (i.e. two clouds c_i and c_j will be connected if there is a read that has an alignment to both c_i and c_j). Conceptually, the clouds in $\mathcal{C}$ can be thought of as alternate possibilities for the *same* latent source fragment. By definition, for any given read aligning to some cloud in $\mathcal{C}$, we will have to consider only the clouds in $\mathcal{C}$ when determining the best alignment for that read, so we focus on each such set of clouds separately. Note that we make the same implicit assumption made by Lariat: namely that distinct fragments sharing a common barcode (i.e. fragments in the same droplet/well) do not overlap on the genome. In reality, there is nothing preventing this from happening, but we can see that it occurs rarely since fragments are effectively sampled uniformly from the entire genome. If we partition the 3Gb genome into 100kb bins (as a reasonable upper bound on mean fragment length) and assume a droplet/well contains about 10 fragments (also a reasonable bound), we can observe that only about $1 - \prod_{i=1}^{10}\left(1 - \frac{i-1}{3\text{Gb}/100\text{kb}}\right) \approx 0.15\%$ will contain overlapping fragments, where (as an approximation) we assume fragments overlap if they are contained in the same bin. By comparison, about 5–6% of all 10x reads are usually left without a barcode after standard barcode correction, so the additional 0.15% is rather marginal.

For $\mathcal{C} = \{c_1, \ldots, c_n\}$, let C_i denote the event that cloud c_i represents the true source fragment. Since the clouds $c_1, \ldots, c_n$ are different possibilities for the same source fragment, we have $\Pr(C_i \cap C_j) = 0$ $(i \neq j)$ and $\sum_{i=1}^{n} \mathbb{1}(C_i) = 1$ (where $\mathbb{1}(\cdot) \in \{0, 1\}$ is an indicator for the specified event). We assume uniform priors on the clouds so that $\Pr(C_i) = \frac{1}{n}$ (while it is possible to devise a prior that takes into account features such as cloud length, we observed a large variance between clouds in our datasets that renders this unhelpful). Now, a cloud c_i can be conceptualized as an entity that generates some number of reads K_i, parameterized by some weight θ_{c_i}, so that we can say $K_i \sim \text{Cloud}(\theta_{c_i})$ for some unknown "cloud" distribution over generated reads. We make the key assumption that, in expectation, $\Pr(C_i \mid \theta_{c_i}) \propto K_i \propto \theta_{c_i}$ for all $c_i \in \mathcal{C}$. In other words, if a cloud is expected to have generated a large number of reads, then the probability that the cloud represents a true source fragment is high. Let $\boldsymbol{\theta} = (\theta_{c_1}, \ldots, \theta_{c_n})$ be the vector of cloud weights. We assume the cloud weights are normalized so that $\Pr(C_i \mid \theta_{c_i}) = \theta_{c_i}$, and that they are drawn from a uniform Dirichlet distribution so that $\boldsymbol{\theta} \sim \text{Dir}(\mathbf{1})$. Consider now the probability γ_{r,c_i} that a read r truly originates from cloud c_i (denoted as an event by Γ_{r,c_i}) given the cloud parameters $\boldsymbol{\theta}$ (i.e. $\Gamma_{r,c_i} \mid \boldsymbol{\theta} \sim \text{Ber}(\gamma_{r,c_i})$, where $\text{Ber}(p)$ is the Bernoulli distribution with parameter p). By Bayes' rule, we can say:

$$\gamma_{r,c_i} = \Pr(\Gamma_{r,c_i} \mid \boldsymbol{\theta}) = \frac{1}{Z_{\mathcal{C}}} \Pr(\boldsymbol{\theta} \mid \Gamma_{r,c_i}) \Pr(\Gamma_{r,c_i}),$$

where $Z_{\mathcal{C}}$s (and variants thereof) are normalization constants that are the same for each $c \in \mathcal{C}$. Since Γ_{r,c_i} occurs if and only if C_i occurs, we have

$$\gamma_{r,c_i} = \frac{1}{Z_{\mathcal{C}}} \Pr(\boldsymbol{\theta} \mid C_i) \Pr(\Gamma_{r,c_i}).$$

Applying Bayes' rule again to $\Pr(\boldsymbol{\theta} \mid C_i)$ and using the fact that both $\Pr(\boldsymbol{\theta})$ and $\Pr(C_i)$ are uniform, we obtain

$$\gamma_{r,c_i} = \frac{1}{Z_{\mathcal{C}}} \frac{\Pr(\boldsymbol{\theta}) \Pr(C_i \mid \boldsymbol{\theta})}{\Pr(C_i)} \Pr(\Gamma_{r,c_i}) = \frac{1}{Z'_{\mathcal{C}}} \Pr(C_i \mid \boldsymbol{\theta}) \Pr(\Gamma_{r,c_i}) = \frac{\theta_{c_i}}{Z'_{\mathcal{C}}} \Pr(\Gamma_{r,c_i}),$$

where $Z'_{\mathcal{C}} = [\Pr(C_i)/\Pr(\theta)] Z_{\mathcal{C}}$. Note that $\Pr(\Gamma_{r,c_i})$ is a prior on the probability that r aligns to c_i that is not dependent on the barcode, but rather only on edit distance, mate alignment, and mapping quality as in standard short-read alignment. Henceforth, we refer to $\Pr(\Gamma_{r,c_i})$ as $\gamma_{r,c_i}^{(0)}$, so that $\Gamma_{r,c_i} \sim \text{Ber}(\gamma_{r,c_i}^{(0)})$.

Now we can form a prior $\boldsymbol{\theta}^{(0)} = (\theta_{c_1}^{(0)}, \ldots, \theta_{c_n}^{(0)})$, which is intuitively the initial vector of cloud weights. If we are given a set of alignment probabilities and a "current" $\boldsymbol{\theta}$ estimate $\boldsymbol{\theta}^{(t)} = (\theta_{c_1}^{(t)}, \ldots, \theta_{c_n}^{(t)})$ (initially $t = 0$), we can iteratively compute a better estimate $\boldsymbol{\theta}^{(t+1)}$ using the fact that $\theta_{c_i} \propto K_i$ in expectation:

$$\begin{aligned}
\theta_{c_i}^{(t+1)} = \frac{1}{|\mathcal{R}|} \mathbb{E}(K_i) &= \frac{1}{|\mathcal{R}|} \mathbb{E}\left(\sum_{r \in \mathcal{R}} \mathbb{1}(\Gamma_{r,c_i}) \;\middle|\; \boldsymbol{\theta}^{(t)} \right) \\
&= \frac{1}{|\mathcal{R}|} \sum_{r \in \mathcal{R}} \Pr(\Gamma_{r,c_i} \mid \boldsymbol{\theta}^{(t)}) \\
&= \frac{1}{|\mathcal{R}|} \sum_{r \in \mathcal{R}} \gamma_{r,c_i}^{(t)},
\end{aligned}$$

where $\mathcal{R}$ is the set of reads mapping to any cloud in $\mathcal{C}$, and the $\frac{1}{|\mathcal{R}|}$ factor ensures that $\sum_{c \in \mathcal{C}} \theta_c = 1$. This latent variable model formulation naturally leads to an expectation-maximization algorithm—one of the widely used ways of maximizing likelihood in such models—for determining the cloud weights and, thereby, the final alignment probabilities $\gamma^{\star}_{r,c_i}$. An implementation of this algorithm is given in Algorithm 1.

Algorithm 1 Barcoded read alignment via expectation-maximization

```
Require: $\mathcal{R}$, $\mathcal{C}$
Ensure: $\gamma^{\star}_{r,c}$ for each $r \in \mathcal{R}$, $c \in \mathcal{C}$
  $\gamma^{(0)}_{r,c} \leftarrow \Pr(r \in c), \quad \forall\, r \in \mathcal{R},\ c \in \mathcal{C}$
  $\theta^{(0)}_{c} \leftarrow \frac{1}{|\mathcal{C}|}, \quad \forall\, c \in \mathcal{C}$
  for $t \in \{0, 1, \ldots, T-1\}$ do
    E step: $\gamma^{(t+1)}_{r,c} \leftarrow \Pr(r \in c_i \mid \theta^{(t)}_{c}) \quad \forall\, r \in \mathcal{R},\ c \in \mathcal{C}$
    M step: $\theta^{(t+1)}_{c} \leftarrow \frac{1}{|\mathcal{R}|} \sum_{r \in \mathcal{R}} \gamma^{(t)}_{r,c} \quad \forall\, r \in \mathcal{R}$
  end for
  $\gamma^{\star}_{r,c} \leftarrow \gamma^{(T)}_{r,c} \quad \forall\, r \in \mathcal{R},\ c \in \mathcal{C}$
```

Each of the described variables is summarized in Table 2.1, and their interactions with each other in Figure 2-2. Once we determine the final alignment probabilities through this method (as in Figure 2-1d), we use them to compute mapping qualities ("MAPQs"), which are a standard per-alignment metric reported by all aligners and are frequently used by downstream analysis pipelines. Specifically, we take the MAPQ to be the minimum of the alignment probability, the barcode-oblivious alignment score and the MAPQ reported by BWA-MEM's API (which is used in EMA's current implementation to find candidate alignments). Importantly, we also report the actual alignment probabilities determined by EMA via a special standard-compliant SAM tag, so that they are available to downstream applications.

2.3 An analogy to Gaussian mixture models

To make this process more intuitive, consider an analogy to Gaussian mixture models (GMMs), as shown in Figure 2-3. The EM algorithm for GMMs is relatively straightforward: given current estimates of the parameters of each Gaussian distribution,

Variable	Description
$\mathcal{C}$	set of all clouds in connected component
$\mathcal{R}$	set of all reads mapping to some cloud in $\mathcal{C}$
$\boldsymbol{\theta} = (\theta_{c_1}, \ldots, \theta_{c_n})$	vector of cloud weights
K_i	number of reads generated by cloud c_i
Γ_{r,c_i}	event that read r truly originates from cloud c_i
γ_{r,c_i}	$\Pr(\Gamma_{rc_i}\vert\theta)$
$\gamma_{r,c_i}^{(0)}$	$\Pr(\Gamma_{rc_i})$ (prior based on edit distance, mate, etc.)

Table 2.1: Description of all variables used in mathematical formulation of described latent variable model.

we compute the probability of each point belonging to each distribution (E step), then recompute the distribution parameters themselves based on these probabilities (M step). EMA's optimization algorithm is completely analogous: we first compute probabilities of each read belonging to each cloud (E step), then recompute the cloud weights based on these probabilities (M step).

2.4 Read density optimization

While the 50kb-heuristic described above is typically effective at determining the clouds, it does not take into account the fact that a single read may align multiple times to the same cloud (which can occur if a cloud spans two or more homologous regions). In such cases, rather than simply picking the alignment with lowest edit distance within the cloud, as is the current practice, we propose a novel alternative approach that takes into account not only edit distance but also read *density*. We take advantage of the insight that there is typically only a single read pair per 1kb bin in each cloud; the exact distribution of read counts per 1kb bin is shown in Figure 2-4. Now consider the case where one of our source fragments spans two highly similar (homologous) regions, and thereby produces a cloud with multi-mappings, as depicted in Figure 2-1c. If we pick alignments solely by edit distance, we may observe an improbable increase in read density (as shown in the figure). Consequently, we select alignments for the reads so as to minimize a combination of edit distance *and* abnormal density deviations.

Specifically, consider any cloud with multi-mappings consisting of a set of reads $R = \{r_1, \ldots, r_n\}$, and denote by A_r the set of alignments for read $r \in R$ in the cloud. Additionally, let $a_r \in A_r$ denote the currently "selected" alignment for r. We will initially partition the cloud, spanning the region from its leftmost to its rightmost alignment, into the set of bins $B = \{b_1, \ldots, b_n\}$ of equal width w, where each bin b_i covers the alignments whose starting positions are located in the interval $[i \cdot w, (i+1) \cdot w)$, as shown in Figure 2-1c. In practice, we set w to 1kb. Denote by C_{b_i} the random variable representing the number of reads in bin b_i, where C_{b_i} is drawn from the bin density distribution CloudBin(i). Lastly, let γ_{a_r} denote the prior probability that alignment a_r is the true alignment of read r based on edit distance and mate alignments alone. Our goal is to maximize the objective:

$$\left[\prod_{r \in R} \gamma_{a_r}\right] \cdot \left[\prod_{b_i \in B} \Pr\left(C_{b_i} = \sum_{r \in R} \mathbb{1}(a_r \in b_i)\right)^{\alpha}\right],$$

where α is a parameter that dictates the relative importance of the density probabilities compared to the alignment probabilities. We determine the distribution CloudBin(i) of each C_{b_i} beforehand by examining uniquely-mapping clouds that we are confident represent the true source fragment. Taking the logarithm, this objective becomes:

$$J(a_{r_1}, \ldots, a_{r_n}) = \sum_{r \in R} \log \gamma_{a_r} + \alpha \sum_{b_i \in B} \log \Pr\left(C_{b_i} = \sum_{r \in R} \mathbb{1}(a_r \in b_i)\right).$$

We optimize J through simulated annealing by repeatedly proposing random changes to a_r and accepting them probabilistically based on the change in our objective (the corresponding algorithm is described in Algorithm 2.

Algorithm 2 Read density optimization via simulated annealing

```
Require: $R; A_r \; \forall r \in R$
Ensure: $a_r^\star \;\; \forall \, r \in R$
  $a_r \leftarrow \text{random}(A_r) \;\; \forall \, r \in R$
  $z \leftarrow J(a_{r_1}, \ldots, a_{r_n})$
  for $k \in \{1, \ldots, K\}$ do
    $r' \leftarrow \text{random}(\{r \in R \, : \, |A_r| > 1\})$
    $a'_r \leftarrow \text{random}(A_r \setminus \{a_r\})$
    $z' \leftarrow J(a_{r_1}, \ldots, a_{r'}, \ldots, a_{r_n})$
    if $z' > z$ or $\exp\left(-\frac{z-z'}{\tau(k)}\right) > \text{random}([0,1))$ then
      $a_r \leftarrow a'_r$
      $z \leftarrow z'$
    end if
  end for
  $a_r^\star \leftarrow a_r \;\; \forall \, r \in \mathcal{R}$
```

In Algorithm 2, K is the number of simulated annealing iterations, and $\tau(\cdot)$ defines the annealing schedule (which can be taken to be an exponentially decreasing function). We apply the preceding latent variable optimization algorithm to deduce optimal alignments *between* clouds and, if necessary, use this statistical binning algorithm to find the best alignments *within* a given cloud.

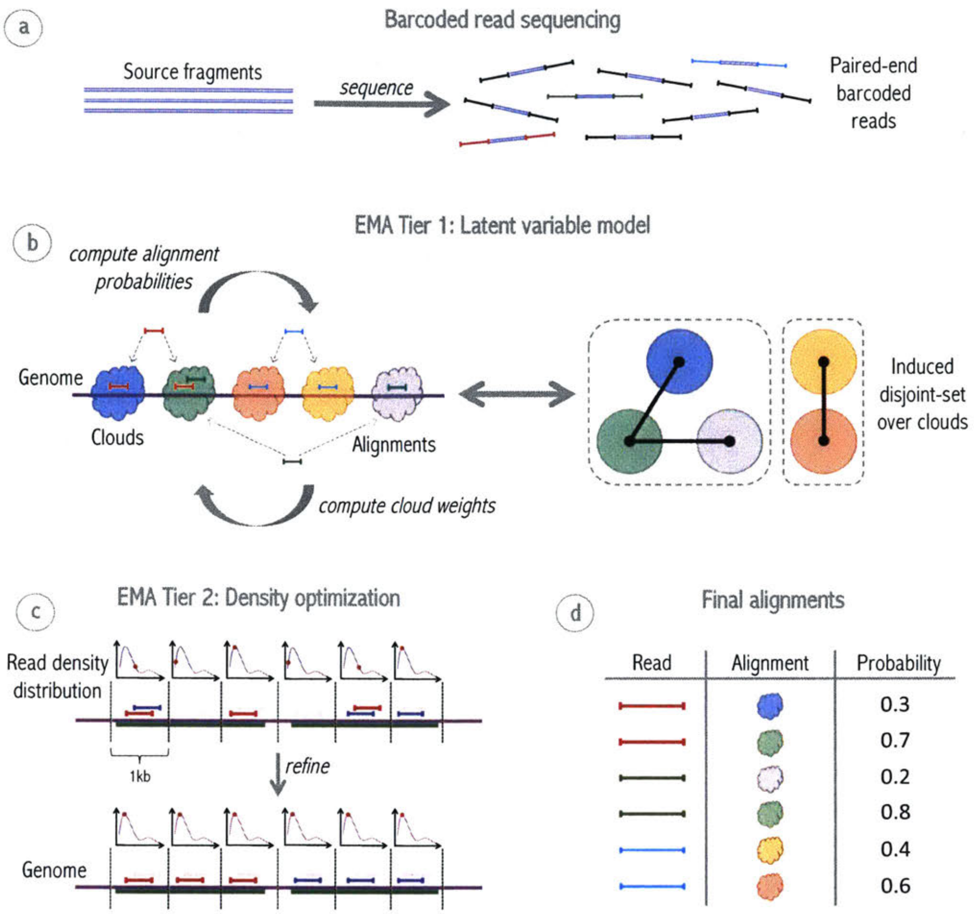
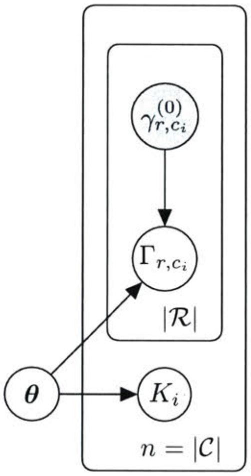

a
Barcoded read sequencing
Source fragments
sequence
Paired-end barcoded reads
b
EMA Tier 1: Latent variable model
compute alignment probabilities
Genome
Clouds
Alignments
compute cloud weights
Induced disjoint-set over clouds
c
EMA Tier 2: Density optimization
Read density distribution
1kb
refine
Genome
d
Final alignments
Read
Alignment
Probability
0.3
0.7
0.2
0.8
0.4
0.6

Figure 2-1: Overview of EMA pipeline. **(a)** Idealized model of barcoded read sequencing, wherein some number of unknown source fragments in a single droplet/well are sheared, barcoded and sequenced to produce barcoded reads. **(b)** EMA's "read clouds" are constructed by grouping nearby-mapping reads sharing the same barcode; these clouds represent possible source fragments. EMA then partitions the clouds into a disjoint-set induced by the alignments, where two clouds are connected if there is a read aligning to both; connected components in this disjoint-set (enclosed by dashed boxes) correspond to alternate possibilities for the *same* unknown source fragment. EMA's latent variable model optimization is subsequently applied to each of these connected components individually to deduce each of the potentially many fragments sharing this barcode. **(c)** EMA applies a novel read density optimization algorithm to clouds containing multiple alignments of the same read to pick out the most likely alignment, by optimizing a combination of alignment edit distances and read densities within the cloud. The green regions of the genome are homologous, thereby resulting in multi-mappings within a single cloud. **(d)** While the read density optimization operates within a single cloud, EMA's latent variable model optimization determines the best alignment of a given read between different clouds, and produces not only the final alignment for each read, but also interpretable alignment *probabilities*.



$$\begin{aligned} \boldsymbol{\theta} = (\theta_{c_1}, \ldots, \theta_{c_n}) &\sim \mathrm{Dir}(\mathbf{1}) \\ K_i \mid \boldsymbol{\theta} &\sim \mathrm{Cloud}(\theta_{c_i}) \\ \Gamma_{r,c_i} &\sim \mathrm{Ber}(\gamma^{(0)}_{r,c_i}) \\ \Gamma_{r,c_i} \mid \boldsymbol{\theta} &\sim \mathrm{Ber}(\gamma_{r,c_i}) \end{aligned}$$

Figure 2-2: Graphical representation of EMA's latent variable model involved in barcoded read alignment. $\boldsymbol{\theta}$ denotes the vector of cloud weights; K_i denotes the number of reads generated by cloud $c_i \in \mathcal{C}$; Γ_{r,c_i} denotes whether read $r \in \mathcal{R}$ maps to cloud c_i, and $\gamma^{(0)}_{r,c_i}$ is a prior on this event based on barcode-oblivious information like edit distance, mate alignment, etc.

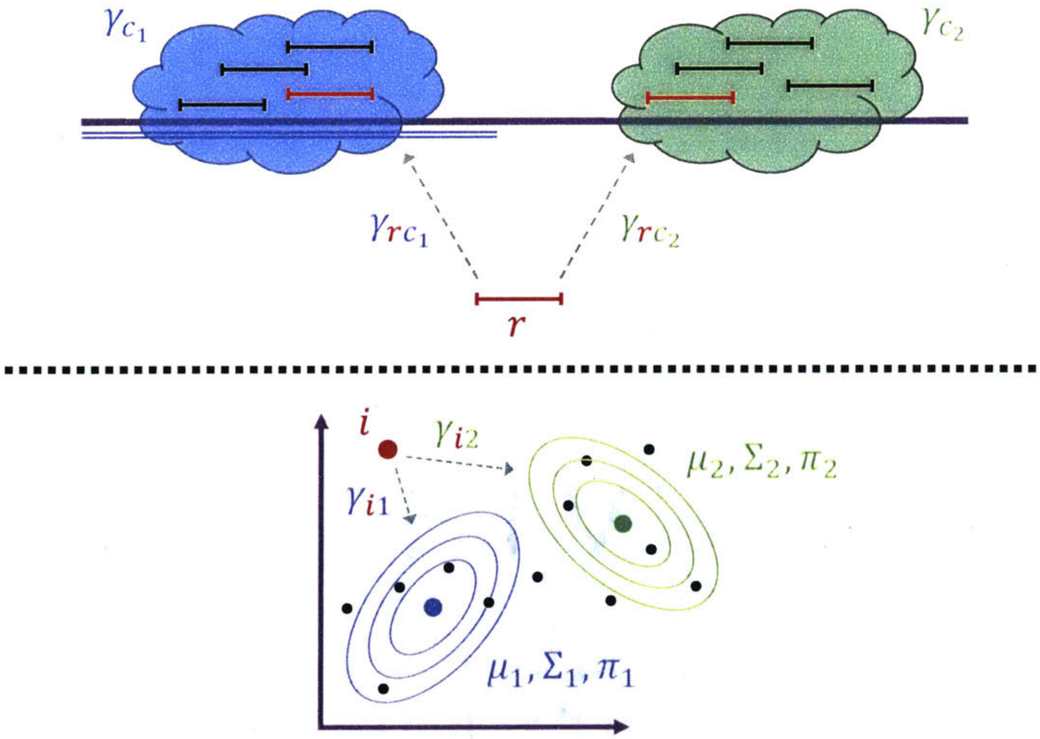


Figure 2-3: A comparison between EMA's latent variable model optimization (top) and a Gaussian mixture models' (bottom). In the former, we are trying to decide between two clouds for a particular read, while in the latter we are trying to decide between two Gaussian distributions. In both cases, the E step of the EM algorithm consists of computing membership probabilities based on distribution parameters or cloud weights, while the M step consists of updating these parameters/weights based on the just-computed probabilities.

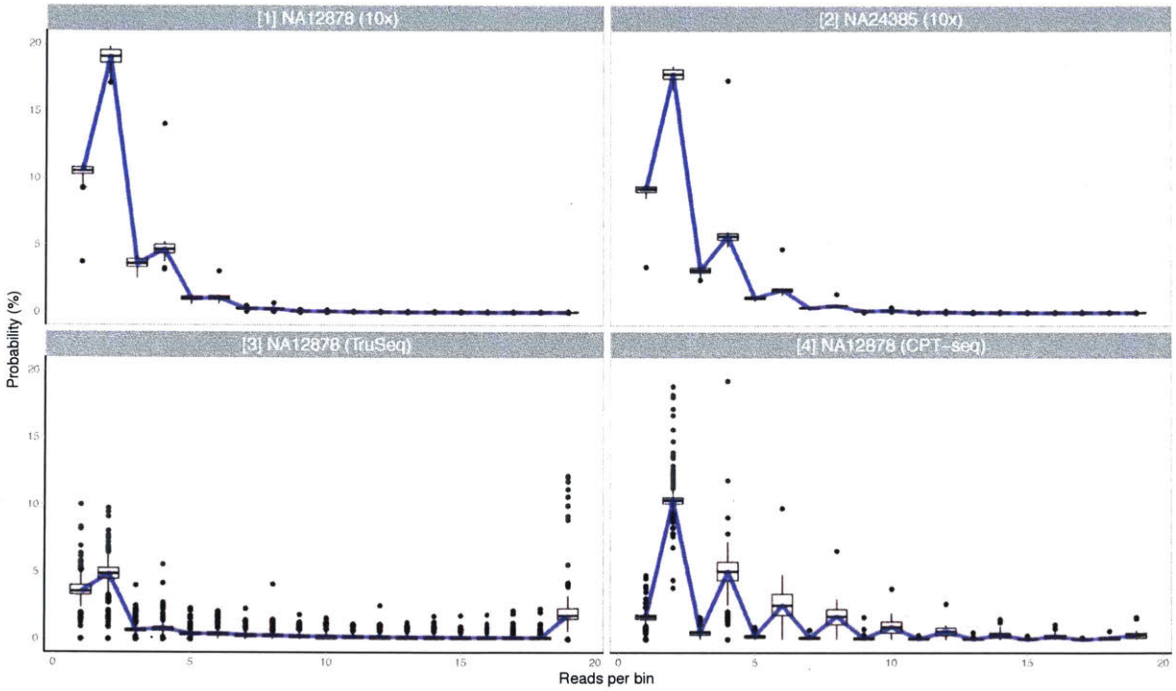


Figure 2-4: Distribution of the number of reads in a 1kb window within a cloud (first row shows the distribution of two 10x data samples, while the bottom row shows TruSeq SLR's and CPT-seq's distribution). We only consider the clouds in which no reads have multiple alignments within the cloud. The box plots correspond to different bin offsets within the cloud.

Chapter 3

Applying the EMA Framework

3.1 Experimental setting

We first compared the performance of EMA against Lariat [14] (10x's own aligner and a component of the Long Ranger software suite) and BWA-MEM [22] (which does not take advantage of barcoded data, and was therefore used as a baseline for what can be achieved with standard short-reads). In order to benchmark the quality of the aligners, we examined downstream genotyping accuracy, alignments in highly homologous regions, and downstream phasing accuracy.

We ran each tool on four 10x *H. sapiens* datasets for NA12878, NA24149, NA24143 and NA24385, and used the corresponding latest NIST GIAB [46, 45] high-confidence variant calls as a gold standard for each. For both EMA and BWA, we performed duplicate marking after alignment using Picard's MarkDuplicates tool (URL: `https://broadinstitute.github.io/picard/`), with barcode-aware mode enabled in the case of EMA; Long Ranger performs duplicate marking automatically. Genotypes were called by GATK's HaplotypeCaller [29, 9] with default settings, while phasing was done by HapCUT2 [10] in barcode-aware mode. Genotyping accuracies were computed using RTG Tools [6]. We also ran EMA and Lariat on a much higher coverage NA12878 dataset ("NA12878 v2") to test genotyping accuracy at high coverage as well as scalability.

To test EMA's improvements on other barcoded read sequencing technologies, we ran EMA and BWA on a NA12878 TruSeq SLR dataset [4] as well as a NA12878 CPT-seq dataset [2]. All analyses in this thesis were performed with respect to the GRCh37 human reference genome.

3.2 Downstream genotyping accuracy

EMA's genotyping accuracy surpassses that of other aligners (Figure 3-1). We found that for each of the four 10x *H. sapiens* datasets, EMA produced 30% fewer false positive variant calls compared to Lariat, and produced fewer false negative calls as well. Interestingly, BWA-MEM (which does not take barcodes into account) performed marginally better than Lariat here. Nevertheless, EMA also outperforms BWA-MEM, attaining the fewest false positive and false negative variant calls between the three aligners on each dataset. To verify that EMA's superior accuracy scales to higher coverage datasets, we tested it on a high-coverage NA12878 dataset (Figure 3-2). EMA attains an even more substantial improvement on the high-coverage dataset, eliminating nearly 37% of Lariat's false positives and 6% of its false negatives. Full genotyping results are given in Appendix A.

When run on TruSeq SLR and CPT-seq data, we did not observe any significant differences in genotyping accuracy between EMA and BWA. This finding is likely due to the fact that these platforms divide the source fragments into just 384 and 9128 wells ("barcodes"), respectively, limiting the utility of the barcodes in unambiguous regions of the genome, which is primarily what our NIST GIAB gold standard consists of. However, for both technologies, we did observe improvements in resolving ambiguous regions of the genome, which we detail below.

Overall, we found that typically ~20% of all reads in our various datasets had multiple suitable alignments and were therefore able to be targeted by EMA's two-tiered statistical binning optimization algorithm. These are precisely those reads that are most challenging to align, and can occur in clinically important regions of the genome, as we next demonstrate.

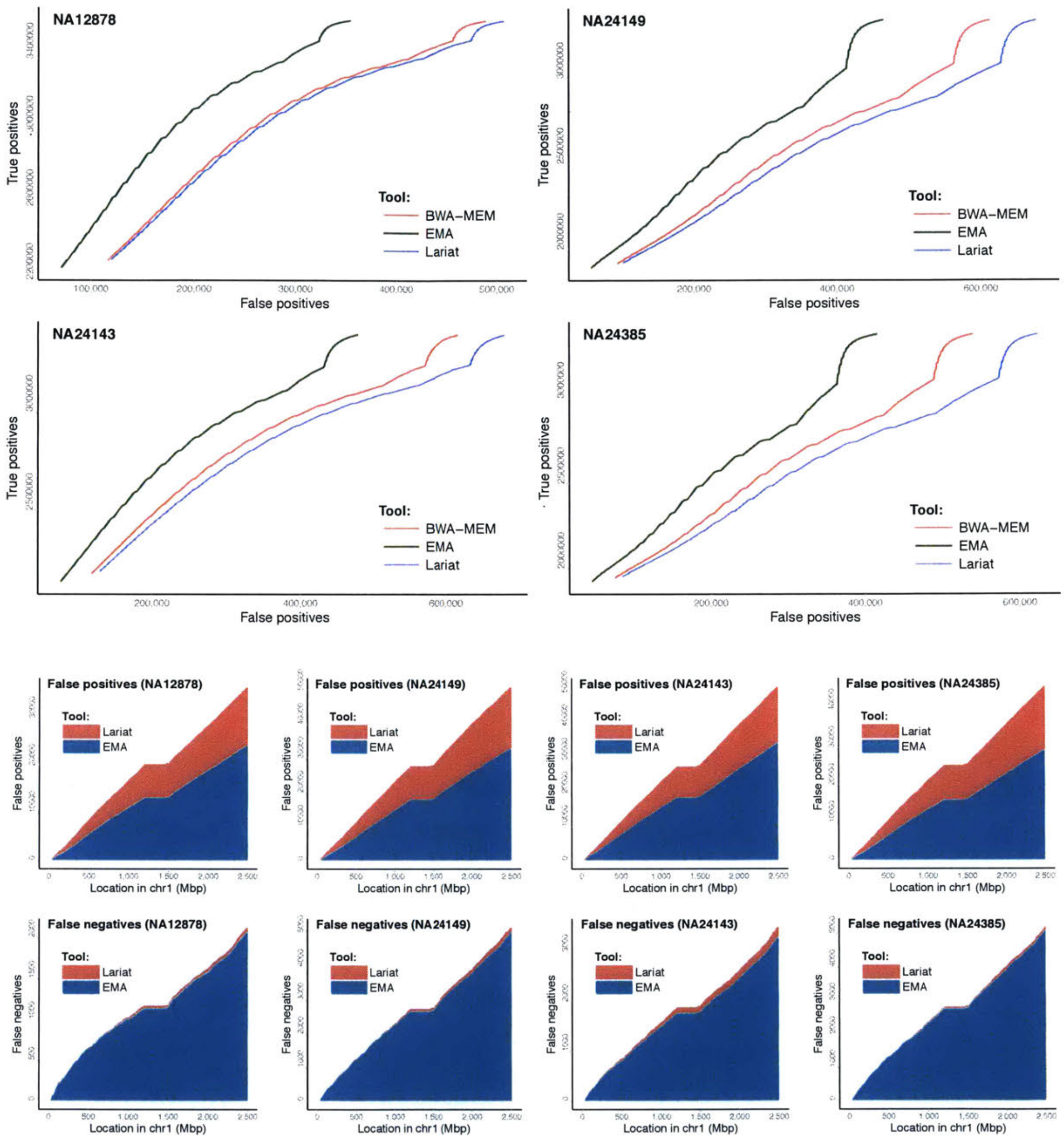


Figure 3-1: Genotyping accuracy for each aligner. The top half shows true positives as a function of false positives for alignments produced by EMA (green), Lariat (blue) and BWA-MEM (red) on the well-studied samples NA12878, NA24149, NA24143 and NA24385. Genotype confidences are determined by the GQ ("genotype quality") annotations generated by GATK's HaplotypeCaller. The bottom half contains cumulative histograms of false positives (top row) and false negatives (bottom row) throughout chromosome 1 for each dataset, for both EMA (blue) and Lariat (red). EMA achieves more than a 30% average improvement over the other methods in terms of eliminating erroneous variant calls.

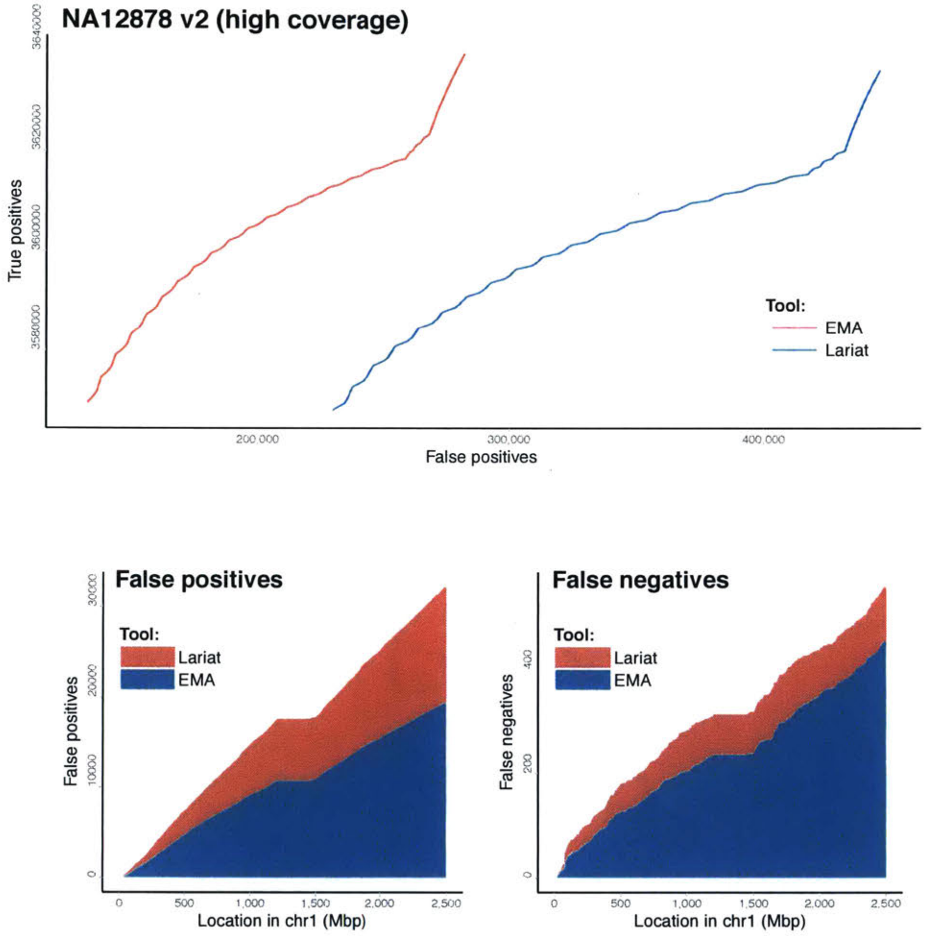


Figure 3-2: Genotyping accuracy for EMA as compared to Lariat on a high-coverage NA12878 10x dataset. The top plot shows true positives as a function of false positives, and the bottom two plots are cumulative histograms of true and false positives throughout chromosome 1. We note that EMA's improvement is even more substantial with higher coverage.

3.3 Alignments in highly homologous regions

Among the principal promises of barcoded read sequencing is better structural variation detection, which invariably requires resolving alignments in homologous regions. One of the most important such regions is the *CYP2D* region in chromosome 22, which hosts *CYP2D6*—a gene of great pharmacogenomic importance [17]—and the two related and highly homologous regions *CYP2D7* and *CYP2D8*. The high homology between *CYP2D6* and *CYP2D7* makes copy number estimation and variant calling in this region particularly challenging. Indeed, the majority of aligners misalign reads in this region. The difficulty is especially evident in NA12878 which, in addition to the two copies of both *CYP2D6* and *CYP2D7*, contains an additional copy that is a fusion between these two genes [38], as well as *CYP2D7* mutations that introduce even higher homology with the corresponding *CYP2D6* region. Especially problematic is exon/intron 8 of *CYP2D6*, where many reads originating from *CYP2D7* end up mapping erroneously (see Figure 3-3 for a visualization). Even the naïve use of barcoded reads is not sufficient: both homologous regions in *CYP2D* are typically covered by a single cloud. For example, Lariat performs no better than BWA in this region (Figure 3-3). For these reasons, we chose to evaluate EMA in *CYP2D* to benchmark its accuracy in such highly homologous regions.

As can be seen in Figure 3-3, EMA's statistical binning strategy significantly smooths out the two problematic peaks in *CYP2D6* and *CYP2D7*. This technique enabled us to detect three novel mutations in *CYP2D7* (Figure 3-3) which exhibit high homology with the corresponding region in *CYP2D6*. Thus all reads originating from these loci get misaligned to *CYP2D6*, especially if one only considers edit distance during the alignment (as Lariat and BWA do). Such misalignments are evident in the "peaks" and "holes" shown in Figure 3-3. We additionally cross-validated this region with the consensus sequence obtained from available NA12878 assemblies [18, 30, 32], and confirmed the presence of novel mutations. Notably, we found similar enhancements in other clinically important and highly homologous genes: *C4* and *AMY1A*, as depicted in the same figure.

In addition, the copy number derived from EMA's alignments in this problematic region (spanning from exon 7 up to exon 9 in *CYP2D6* and *CYP2D7*) was closer to the "expected" copy number by 20% compared to the copy number derived from Lariat's alignments (we used Aldy [31] to obtain this data). We further ran Aldy on our high-coverage NA12878 v2 dataset, where it correctly detected the *3/*68+*4 allelic combination on both EMA's and Lariat's alignments, and EMA's overall copy number error over the whole region was around 4% better than Lariat's. Finally, statistical binning did not adversely impact phasing performance in this region, as we were able to correctly phase *CYP2D6*4A* alleles in our NA12878 sample from EMA's alignments (Appendix A).

To demonstrate the generalizability of our paradigm to other similar barcoded sequencing technologies, we tested it on TruSeq SLR and CPT-seq data, where the bin distributions follow a similar pattern as 10x's. We alone were able to detect the same novel *CYP2D7*, *C4* and *AMY1A* variants in a NA12878 TruSeq SLR dataset (even with shallow coverage), and to detect the *CYP2D7* variants in an NA12878 CPT-seq dataset, as shown in Figure 3-4.

3.4 Downstream phasing

We applied the state-of-the-art phasing algorithm HapCUT2 [10], which supports 10x barcoded reads, to phase (i.e. link variants into haplotypes) the variants called by GATK for both EMA's and Lariat's alignments. We evaluated our results with the phasing metrics defined in the HapCUT2 manuscript (Table 3.1). EMA provides more accurate phasings with respect to every metric in comparison to Lariat.

3.5 Computational efficiency

Runtimes and memory usage for each aligner are provided in Table 3.2 for our small and large NA12878 datasets. These times include alignment, duplicate marking and any other data post-processing (e.g. BAM sorting/merging). The reported memory

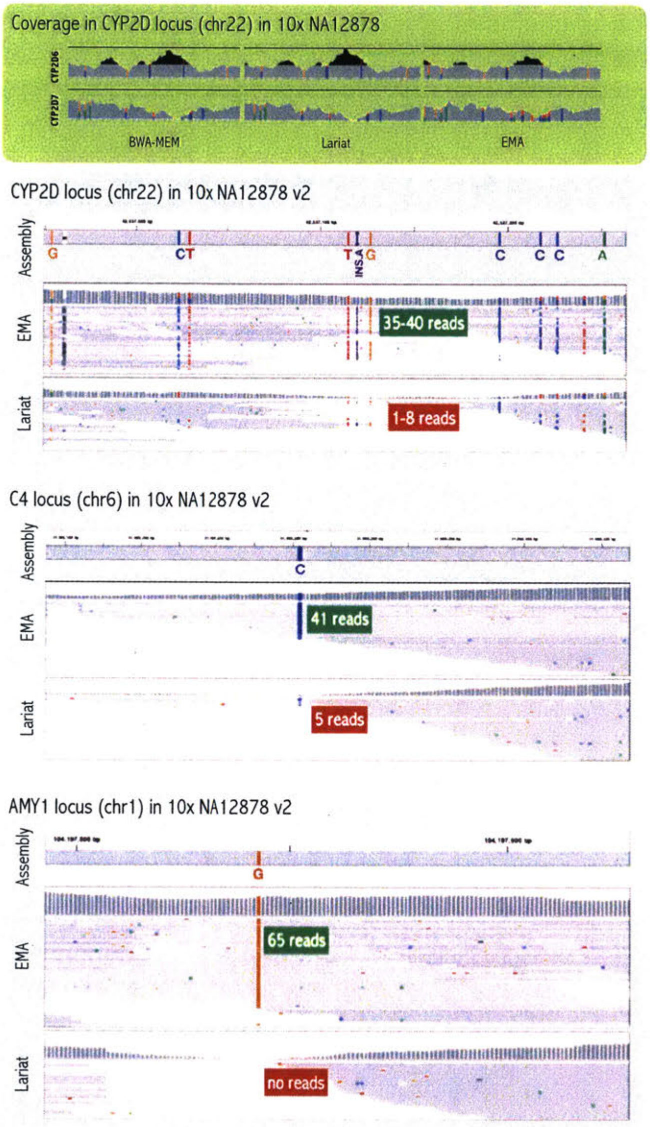

Coverage in CYP2D locus (chr22) in 10x NA12878
CYP2D6
CYP2D7
BWA-MEM
Lariat
EMA
CYP2D locus (chr22) in 10x NA12878 v2
Assembly
G
CT
T
INS
A
G
C
C C
A
EMA
35-40 reads
Lariat
1-8 reads
C4 locus (chr6) in 10x NA12878 v2
Assembly
C
EMA
41 reads
Lariat
5 reads
AMY1 locus (chr1) in 10x NA12878 v2
Assembly
G
EMA
65 reads
Lariat
no reads

Figure 3-3: Positive effect of EMA's statistical binning in the clinically important genes *CYP2D6*, *CYP2D7*, *C4* and *AMY1A*. The top image (light green) shows the read coverage for the region around exon/intron 8 of *CYP2D6* (top row) and *CYP2D7* (bottom row). Spurious coverage peaks (i.e. increases in observed coverage likely to be false) in *CYP2D6* are shaded black. EMA is clearly able to remove the problematic peaks and correctly assign them to *CYP2D7*. The bottom portion of the image (gray) shows the newly assigned mappings to *CYP2D7*: EMA's alignments agree with the assembly consensus sequence (observe the insertion and two neighboring SNPs detected by EMA). By contrast, both Lariat and BWA-MEM aligned virtually no reads to this region, and were thus unable to call these mutations. Analogous images are given below for *C4* and *AMY1A*. We observed the same effects in both the normal and high-coverage NA12878 samples.

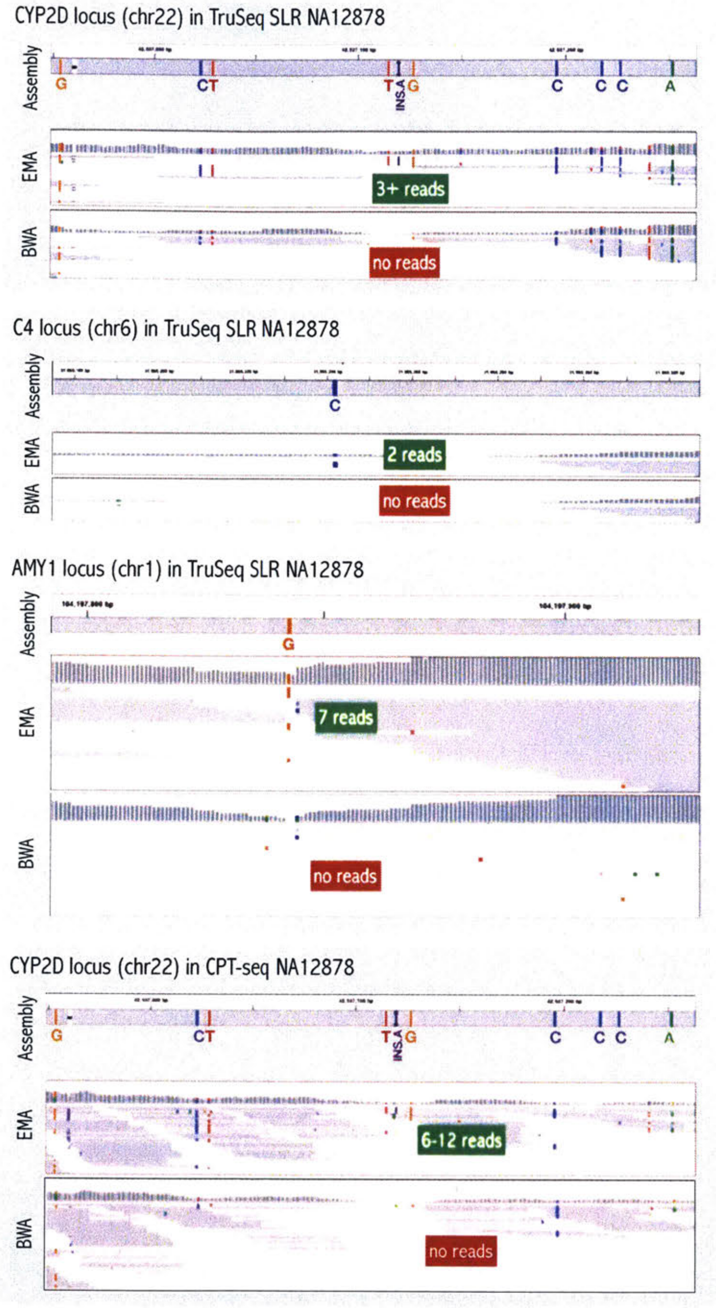

CYP2D locus (chr22) in TruSeq SLR NA12878
Assembly
G
CT
T
INS:A
G
C
C C
A
EMA
3+ reads
BWA
no reads
C4 locus (chr6) in TruSeq SLR NA12878
Assembly
C
EMA
2 reads
BWA
no reads
AMY1 locus (chr1) in TruSeq SLR NA12878
Assembly
G
EMA
7 reads
BWA
no reads
CYP2D locus (chr22) in CPT-seq NA12878
Assembly
G
CT
T
INS:A
G
C
C C
A
EMA
6-12 reads
BWA
no reads

Figure 3-4: Positive effect of EMA's statistical binning on TruSeq SLR data (top three slides) in the clinically important genes *CYP2D6*, *CYP2D7*, *C4* and *AMY1A*, showing that EMA's alignments agree with the assembly consensus sequence. By contrast, BWA-MEM aligned virtually no reads to these regions, and was thus unable to call these mutations. The last slide shows CPT-seq results in the *CYP2D* region, which are similar to those of the 10x and TruSeq SLR datasets.

Sample	Tool	Switch errors	Mismatch errors	Flat errors	N50
NA12878	EMA	**12,796**	**14,163**	**538,169**	**111,392,359**
	Lariat	13,001	14,705	609,858	92,447,569
NA24385	EMA	**10,240**	**14,110**	**377,957**	**115,423,711**
	Lariat	10,472	14,655	429,896	**115,423,711**

Table 3.1: Phasing results for EMA and Lariat on NA12878 and NA24385. Bold type indicates best results. Error metrics indicate the number of "incorrect" phasings compared to the GIAB gold standard; N50 metrics are based on the length of the phase blocks (bp). Switch errors refer to incorrect phase switches between the actual and predicted haplotypes; mismatch errors refer to incorrectly phased heterozygous variants; flat errors refer to the minimum Hamming distance between the actual and predicted haplotypes [10].

	NA12878		*NA12878 v2*	
Tool	**Time (hh:mm)**	**Mem./core (GB)**	**Time (hh:mm)**	**Mem./core (GB)**
EMA	14:58 (10:40)	5.4	28:30 (17:45)	8.7
Lariat	21:49 (12:45)	7.0	54:53 (26:01)	8.2
BWA-MEM	14:49 (9:52)	5.5		

Table 3.2: Runtime and memory usages on two NA12878 datasets (first is about 287GB of raw data, while v2 is about 823GB). Numbers in parenthesis indicate the performance of the aligner alone (i.e. without sorting, merging or duplicate marking). For the small dataset, each mapper was allocated 40 Intel Xeon E5-2650 CPUs @ 2.30GHz. For the large dataset, each was allocated 48 Intel Xeon E5-2695 CPUs @ 2.40GHz. Memory measurements include only the actual aligner's memory usage and do not include the memory requirements of pre- and post-processing steps, as they are virtually the same for all methods. Note that BWA-MEM was used only as a baseline on the smaller dataset.

usages are per each instance of the given mapper. We found that EMA scales better than Lariat: specifically, we observe a 1.5× speedup on our smaller dataset and a nearly 2× speedup on our larger one, over Lariat's runtimes. We ran EMA on a total of four high-coverage datasets and have observed that EMA scales linearly in the size of the dataset (Table A.1, Appendix A).

Chapter 4

Discussion and Future Directions

EMA's unique ability to assign interpretable probabilities to alignments has several benefits, the most immediate of which is that it enables us to set a meaningful confidence threshold on alignments. Additionally, these alignment probabilities can be incorporated into downstream applications such as genotyping, phasing and structural variation detection. We demonstrate this feature here by computing mapping qualities based on these probabilities, which consequently enhance genotyping and phasing. Nevertheless, specialized algorithms centered around these probabilities are also conceivable.

Moreover, EMA is able to effectively discern between multiple alignments of a read in a single cloud through its read density optimization algorithm. This capability addresses one of the weaknesses of barcoded read sequencing as compared to long-read sequencing; namely, that only a relatively small subset of the original source fragment is observed—and more specifically, that the order of reads within the fragment is not known—making it difficult to produce accurate alignments if the fragment spans homologous elements. By exploiting the insight that read densities within a fragment follow a particular distribution, EMA more effectively aligns the reads produced by such fragments, which can overlap regions of phenotypic or pharmacogenomic importance, such as *CYP2D*, *C4* or *AMY1*, as we demonstrated. In summary, EMA's first tier (latent variable model) helps resolve the case of distant homologs, and its second tier the case of proximal homologs, both of which have confounded existing methods.

There are several promising future directions to explore. For example, while here we presented and validated a particular model for barcoded read alignment, we can still conceive of different more general models. We could learn the patterns of read alignments within clouds with, say, a support vector machine or neural network trained on uniquely-mapping clouds, and use that to discern between clouds for multi-mapping reads. This idea may prove advantageous because it has the potential to learn subtle properties of clouds that are difficult to capture explicitly, like the fragment length distribution or read density distributions. One could conceivably train such a model for each of the different barcoded read sequencing platforms, thereby learning their subtle characteristics, and even integrate the data from several platforms to further enhance downstream analyses. As for the read density optimization component of EMA, it would be interesting to incorporate copy number information into the optimization problem, since copy number should actually affect the density distributions (e.g. high copy number would shift the distribution to the right), although our analysis did show that CNV detection already got slightly better as a result of employing this algorithm. Beyond these enhancements, integrating the density optimization into the latent variable model EM algorithm would also be a step forward, and could lead to more meaningful probabilities for reads mapping to nearby homologs.

As we usher in the next wave of next-generation sequencing technologies, barcoded read sequencing will undoubtedly play a central role, and fast and accurate methods for aligning barcoded reads, such as EMA, will ultimately prove invaluable in downstream analyses.

Appendix A

Full Results

Genotyping accuracies were determined using RTG Tools' "vcfeval" utility after genotyping with GATK's HaplotypeCaller. The full results are shown in Table A.2. We used the latest NIST GIAB high-confidence variant calls as a gold standard.

EMA runtimes on four high-coverage 10x samples are given in Table A.1. Read density distributions for 10x, TruSeq SLR and CPT-seq data are given in Figure 2-4.

A.1 *CYP2D* analysis

The copy number of each intron and exon in the *CYP2D* region was obtained by running Aldy [31] on both Lariat's and EMA's alignments. We calculated the absolute difference from the estimated copy number for exon 7, intron 7, exon 8 and intron 8 (in both *CYP2D6* and *CYP2D7*), and the expected coverage (obtained from [38]: 2 for *CYP2D6* and 3 for *CYP2D7* regions). This difference is 5.51 for EMA's alignments, and 8.22 for Lariat's, implying an improvement of 20% if one uses EMA. Similarly, on our NA12878 v2 dataset, the overall difference (on all exons and introns) is 7.16 if one uses EMA and 8.25 for Lariat, implying 4% overall improvement.

Furthermore, phased data from both Lariat's and EMA's alignments correctly linked *CYP2D6*4A* mutations together (i.e. chr22:42,524,947 C>T, chr22:42,525,811 T>C, chr22:42,525,821 G>T and chr22:42,526,694 G>A).

Sample	Size (GB)	Time (hh:mm)
NA12878	823	17:45
NA19238	483	11:57
NA19240	677	14:25
NA24385	658	14:55

Table A.1: EMA runtimes on four high-coverage 10x datasets. EMA was allocated 48 Intel Xeon E5-2695 CPUs @ 2.40GHz. These timings do not include the pre- and pose-processing steps.

A.2 Other barcoded sequencing technologies

EMA needed around 6 hours on a 48-core machine to complete on three shallowly-sequenced TruSeq SLR NA12878 lanes (accession: BioProject PRJNA287848). We also ran BWA-MEM on the same dataset. We were not able to successfully run the RFA aligner on this dataset.

On a CPT-seq NA12878 dataset (accession: BioProject PRJNA241346), EMA needed approximately 42 hours to complete on a 48-core machine. The reason for the increased runtime is CPT-seq's short 50bp read length, which results in significantly more multi-mappings than 10x's and TruSeq SLR's >100bp reads. The short length of the reads makes it much harder to properly utilize our binning technique; despite this, EMA managed to properly align reads in the problematic *CYP2D* region.

NA12878

Tool	True pos. baseline	True pos. call	False pos.	False neg.	Prec.	Sens.	F_1
EMA	**3,614,882**	**3,614,969**	**354,829**	**76,274**	**0.911**	**0.979**	**0.944**
Lariat	3,613,361	3,613,447	507,666	77,795	0.877	0.979	0.925
BWA-MEM	3,613,352	3,613,443	489,605	77,804	0.881	0.979	0.927

NA24149

Tool	True pos. baseline	True pos. call	False pos.	False neg.	Prec.	Sens.	F_1
EMA	**3,336,661**	**3,336,864**	**465,629**	**135,047**	**0.878**	**0.961**	**0.917**
Lariat	3,335,495	3,335,714	679,025	136,213	0.831	0.961	0.891
BWA-MEM	3,335,801	3,336,008	613,494	135,907	0.845	0.961	0.899

NA24143

Tool	True pos. baseline	True pos. call	False pos.	False neg.	Prec.	Sens.	F_1
EMA	**3,394,171**	**3,394,389**	**478,930**	**112,730**	**0.876**	**0.968**	**0.920**
Lariat	3,390,938	3,391,148	679,881	115,963	0.833	0.967	0.895
BWA-MEM	3,391,744	3,391,964	617,525	115,157	0.846	0.967	0.903

NA24385

Tool	True pos. baseline	True pos. call	False pos.	False neg.	Prec.	Sens.	F_1
EMA	**3,375,423**	**3,375,593**	**416,442**	**137,178**	**0.890**	**0.961**	**0.924**
Lariat	3,374,059	3,374,236	624,103	138,542	0.844	0.961	0.899
BWA-MEM	3,374,670	3,374,845	539,915	137,931	0.862	0.961	0.909

NA12878 v2 – high coverage

Tool	True pos. baseline	True pos. call	False pos.	False neg.	Prec.	Sens.	F_1
EMA	**3,639,349**	**3,639,452**	**282,534**	**51,512**	**0.928**	**0.986**	**0.956**
Lariat	3,636,020	3,636,121	446,111	54,841	0.891	0.985	0.936

Table A.2: Full genotyping results. Best results are shown in bold.

Appendix B

Implementation

A visualization of the EMA pipeline is given in Figure B-1. The following parameters for EMA were used in the various experiments:

Parameter	Description	Value
T	number of EM iterations	5
α	density probability weight in statistical binning	0.05

EMA uses BWA-MEM's C API to find candidate alignments just as Lariat does. EMA's full code is available at `http://ema.csail.mit.edu`.

B.1 Versions and parameters

Versions and parameters for other tools used in the various experiments are given in Table B.1.

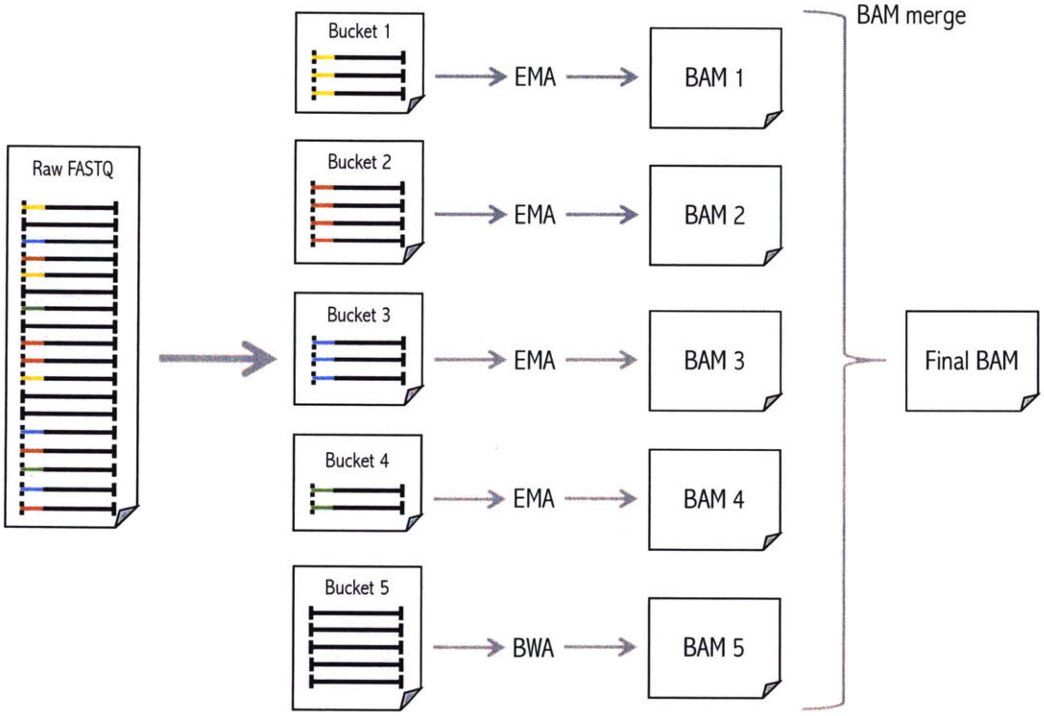


Figure B-1: EMA pipeline. Raw FASTQs are split into buckets by barcode during preprocessing, then each bucket is processed by a separate instance of EMA in parallel (e.g. using GNU Parallel [37]). A special bucket containing non-barcoded reads is processed with BWA-MEM. The resulting BAM files are subsequently marked for duplicates and merged to produce a single, final BAM file as output. EMA is also multithreaded, so multiple processors can be used to work on a single barcode bucket.

Tool/Dataset	**Version**	**Parameters**
Long Ranger	2.1.6	default
BWA	0.7.15	default
GATK HaplotypeCaller	3.8.0	default
HapCUT2	eb3b64b	linked-read mode

Picard MarkDuplicates	2.9.2	`READ_ONE_BARCODE_TAG=BX`
		`READ_TWO_BARCODE_TAG=BX`

Samtools	1.3.1	n/a
RTG Tools	3.8.4	n/a
NIST GIAB	3.3.2	n/a

Bibliography

[1] 10x Genomics. V(d)j sequencing. `https://www.10xgenomics.com/vdj-sequencing`, 2018.

[2] Sasan Amini, Dmitry Pushkarev, Lena Christiansen, Emrah Kostem, Tom Royce, Casey Turk, Natasha Pignatelli, Andrew Adey, Jacob O. Kitzman, Kandaswamy Vijayan, Mostafa Ronaghi, Jay Shendure, Kevin L. Gunderson, and Frank J. Steemers. Haplotype-resolved whole-genome sequencing by contiguity-preserving transposition and combinatorial indexing. *Nature Genetics*, 46:1343 EP –, Oct 2014.

[3] Jeffrey A. Bailey, Zhiping Gu, Royden A. Clark, Knut Reinert, Rhea V. Samonte, Stuart Schwartz, Mark D. Adams, Eugene W. Myers, Peter W. Li, and Evan E. Eichler. Recent segmental duplications in the human genome. *Science*, 297(5583):1003–1007, 2002.

[4] Alex Bishara, Yuling Liu, Ziming Weng, Dorna Kashef-Haghighi, Daniel E. Newburger, Robert West, Arend Sidow, and Serafim Batzoglou. Read clouds uncover variation in complex regions of the human genome. *Genome Res*, 25(10):1570–1580, Oct 2015. 26286554[pmid].

[5] Brian Cleary, Le Cong, Anthea Cheung, Eric S Lander, and Aviv Regev. Efficient generation of transcriptomic profiles by random composite measurements. *Cell*, 171(6):1424–1436, 2017.

[6] John G. Cleary, Ross Braithwaite, Kurt Gaastra, Brian S. Hilbush, Stuart Inglis, Sean A. Irvine, Alan Jackson, Richard Littin, Sahar Nohzadeh-Malakshah, Mehul Rathod, and et al. Joint variant and de novo mutation identification on pedigrees from high-throughput sequencing data. *Journal of Computational Biology*, 21(6):405–419, Jun 2014.

[7] International Human Genome Sequencing Consortium. Initial sequencing and analysis of the human genome. *Nature*, 409:860 EP –, Feb 2001.

[8] Mircea Cretu Stancu, Markus J. van Roosmalen, Ivo Renkens, Marleen M. Nieboer, Sjors Middelkamp, Joep de Ligt, Giulia Pregno, Daniela Giachino, Giorgia Mandrile, Jose Espejo Valle-Inclan, Jerome Korzelius, Ewart de Bruijn, Edwin Cuppen, Michael E. Talkowski, Tobias Marschall, Jeroen de Ridder, and

Wigard P. Kloosterman. Mapping and phasing of structural variation in patient genomes using nanopore sequencing. *Nature Communications*, 8(1):1326, 2017.

[9] Mark A. DePristo, Eric Banks, Ryan Poplin, Kiran V. Garimella, Jared R. Maguire, Christopher Hartl, Anthony A. Philippakis, Guillermo del Angel, Manuel A. Rivas, Matt Hanna, Aaron McKenna, Tim J. Fennell, Andrew M. Kernytsky, Andrey Y. Sivachenko, Kristian Cibulskis, Stacey B. Gabriel, David Altshuler, and Mark J. Daly. A framework for variation discovery and genotyping using next-generation DNA sequencing data. *Nature Genetics*, 43(5):491–498, May 2011.

[10] Peter Edge, Vineet Bafna, and Vikas Bansal. Hapcut2: robust and accurate haplotype assembly for diverse sequencing technologies. *Genome Research*, 27(5):801–812, Dec 2016.

[11] John Eid, Adrian Fehr, Jeremy Gray, Khai Luong, John Lyle, Geoff Otto, Paul Peluso, David Rank, Primo Baybayan, Brad Bettman, Arkadiusz Bibillo, Keith Bjornson, Bidhan Chaudhuri, Frederick Christians, Ronald Cicero, Sonya Clark, Ravindra Dalal, Alex Dewinter, John Dixon, Mathieu Foquet, Alfred Gaertner, Paul Hardenbol, Cheryl Heiner, Kevin Hester, David Holden, Gregory Kearns, Xiangxu Kong, Ronald Kuse, Yves Lacroix, Steven Lin, Paul Lundquist, Congcong Ma, Patrick Marks, Mark Maxham, Devon Murphy, Insil Park, Thang Pham, Michael Phillips, Joy Roy, Robert Sebra, Gene Shen, Jon Sorenson, Austin Tomaney, Kevin Travers, Mark Trulson, John Vieceli, Jeffrey Wegener, Dawn Wu, Alicia Yang, Denis Zaccarin, Peter Zhao, Frank Zhong, Jonas Korlach, and Stephen Turner. Real-time DNA sequencing from single polymerase molecules. *Science*, 323(5910):133–138, January 2009.

[12] Mario Falchi, Julia Sarah El-Sayed Moustafa, Petros Takousis, Francesco Pesce, Amélie Bonnefond, Johanna C. Andersson-Assarsson, Peter H. Sudmant, Rajkumar Dorajoo, Mashael Nedham Al-Shafai, Leonardo Bottolo, Erdal Ozdemir, Hon-Cheong So, Robert W. Davies, Alexandre Patrice, Robert Dent, Massimo Mangino, Pirro G. Hysi, Aurélie Dechaume, Marlène Huyvaert, Jane Skinner, Marie Pigeyre, Robert Caiazzo, Violeta Raverdy, Emmanuel Vaillant, Sarah Field, Beverley Balkau, Michel Marre, Sophie Visvikis-Siest, Jacques Weill, Odile Poulain-Godefroy, Peter Jacobson, Lars Sjostrom, Christopher J. Hammond, Panos Deloukas, Pak Chung Sham, Ruth McPherson, Jeannette Lee, E. Shyong Tai, Robert Sladek, Lena M. S. Carlsson, Andrew Walley, Evan E. Eichler, Francois Pattou, Timothy D. Spector, and Philippe Froguel. Low copy number of the salivary amylase gene predisposes to obesity. *Nature Genetics*, 46:492 EP –, Mar 2014.

[13] Paolo Ferragina and Giovanni Manzini. Opportunistic data structures with applications. In *Foundations of Computer Science, 2000. Proceedings. 41st Annual Symposium on*, pages 390–398. IEEE, 2000.

[14] 10X Genomics. Long ranger pipeline. `https://support.10xgenomics.com/genome-exome/software/pipelines/latest/what-is-long-ranger/`, 2017.

[15] Sara Goodwin, John D. McPherson, and W. Richard McCombie. Coming of age: ten years of next-generation sequencing technologies. *Nat Rev Genet*, 17(6):333–351, Jun 2016. Review.

[16] Tamar Hashimshony, Naftalie Senderovich, Gal Avital, Agnes Klochendler, Yaron de Leeuw, Leon Anavy, Dave Gennert, Shuqiang Li, Kenneth J Livak, Orit Rozenblatt-Rosen, et al. Cel-seq2: sensitive highly-multiplexed single-cell rna-seq. *Genome biology*, 17(1):77, 2016.

[17] Magnus Ingelman-Sundberg. Genetic polymorphisms of cytochrome P450 2D6 (CYP2D6): Clinical consequences, evolutionary aspects and functional diversity. *The pharmacogenomics journal*, 5(1):6–13, 2004.

[18] Miten Jain, Sergey Koren, Josh Quick, Arthur C Rand, Thomas A Sasani, John R Tyson, Andrew D Beggs, Alexander T Dilthey, Ian T Fiddes, Sunir Malla, Hannah Marriott, Karen H Miga, Tom Nieto, Justin O'Grady, Hugh E Olsen, Brent S Pedersen, Arang Rhie, Hollian Richardson, Aaron Quinlan, Terrance P Snutch, Louise Tee, Benedict Paten, Adam M. Phillippy, Jared T Simpson, Nicholas James Loman, and Matthew Loose. Nanopore sequencing and assembly of a human genome with ultra-long reads. *bioRxiv*, 2017.

[19] Jacob O. Kitzman. Haplotypes drop by drop. *Nature Biotechnology*, 34:296 EP –, Mar 2016.

[20] Ben Langmead and Steven L. Salzberg. Fast gapped-read alignment with bowtie 2. *Nat Methods*, 9(4):357–359, Mar 2012. 22388286[pmid].

[21] Ben Langmead and Steven L. Salzberg. Fast gapped-read alignment with Bowtie 2. *Nat Meth*, 9(4):357–359, Apr 2012. Brief Communication.

[22] Heng Li and Richard Durbin. Fast and accurate short read alignment with Burrows-Wheeler transform. *Bioinformatics*, 25(14):1754–1760, 2009.

[23] Heng Li and Nils Homer. A survey of sequence alignment algorithms for next-generation sequencing. *Briefings in Bioinformatics*, 11(5):473–483, 2010.

[24] Evan Z Macosko, Anindita Basu, Rahul Satija, James Nemesh, Karthik Shekhar, Melissa Goldman, Itay Tirosh, Allison R Bialas, Nolan Kamitaki, Emily M Martersteck, et al. Highly parallel genome-wide expression profiling of individual cells using nanoliter droplets. *Cell*, 161(5):1202–1214, 2015.

[25] Santiago Marco-Sola, Michael Sammeth, Roderic Guigó, and Paolo Ribeca. The gem mapper: fast, accurate and versatile alignment by filtration. *Nature Methods*, 9:1185 EP –, Oct 2012.

[26] Elaine R. Mardis. DNA sequencing technologies: 2006-2016. *Nat. Protocols*, 12(2):213–218, Feb 2017. Perspective.

[27] Rajiv C McCoy, Ryan W Taylor, Timothy A Blauwkamp, Joanna L Kelley, Michael Kertesz, Dmitry Pushkarev, Dmitri A Petrov, and Anna-Sophie Fiston-Lavier. Illumina truseq synthetic long-reads empower de novo assembly and resolve complex, highly-repetitive transposable elements. *PloS one*, 9(9):e106689, 2014.

[28] Mark A. McElwain, Rebecca Yu Zhang, Radoje Drmanac, and Brock A. Peters. *Long Fragment Read (LFR) Technology: Cost-Effective, High-Quality Genome-Wide Molecular Haplotyping*, pages 191–205. Springer New York, New York, NY, 2017.

[29] Aaron McKenna, Matthew Hanna, Eric Banks, Andrey Sivachenko, Kristian Cibulskis, Andrew Kernytsky, Kiran Garimella, David Altshuler, Stacey Gabriel, Mark Daly, and Mark A. DePristo. The Genome Analysis Toolkit: A MapReduce framework for analyzing next-generation DNA sequencing data. *Genome Research*, 20(9):1297–1303, September 2010.

[30] Yulia Mostovoy, Michal Levy-Sakin, Jessica Lam, Ernest T Lam, Alex R Hastie, Patrick Marks, Joyce Lee, Catherine Chu, Chin Lin, Željko Džakula, and et al. A hybrid approach for de novo human genome sequence assembly and phasing. *Nature Methods*, 13(7):587–590, May 2016.

[31] Ibrahim Numanagić, Salem Malikić, Michael Ford, Xiang Qin, Lorraine Toji, Milan Radovich, Todd C. Skaar, Victoria M. Pratt, Bonnie Berger, Steve Scherer, and S. Cenk Sahinalp. Allelic decomposition and exact genotyping of highly polymorphic and structurally variant genes. *Nature Communications*, 9(1):828, 2018.

[32] Matthew Pendleton, Robert Sebra, Andy Wing Chun Pang, Ajay Ummat, Oscar Franzen, Tobias Rausch, Adrian M Stütz, William Stedman, Thomas Anantharaman, Alex Hastie, and et al. Assembly and diploid architecture of an individual human genome via single-molecule technologies. *Nature Methods*, 12(8):780–786, Jun 2015.

[33] Katharina Schwarze, James Buchanan, Jenny C. Taylor, and Sarah Wordsworth. Are whole-exome and whole-genome sequencing approaches cost-effective?: A systematic review of the literature. *Genetics In Medicine*, pages EP –, Feb 2018. Systematic Review.

[34] Aswin Sekar, Allison R. Bialas, Heather de Rivera, Avery Davis, Timothy R. Hammond, Nolan Kamitaki, Katherine Tooley, Jessy Presumey, Matthew Baum, Vanessa Van Doren, Giulio Genovese, Samuel A. Rose, Robert E. Handsaker, Schizophrenia Working Group of the Psychiatric Genomics Consortium, Mark J. Daly, Michael C. Carroll, Beth Stevens, and Steven A. McCarroll. Schizophrenia

risk from complex variation of complement component 4. *Nature*, 530:177, Jan 2016. Article.

[35] Jeong-Sun Seo, Arang Rhie, Junsoo Kim, Sangjin Lee, Min-Hwan Sohn, Chang-Uk Kim, Alex Hastie, Han Cao, Ji-Young Yun, Jihye Kim, Junho Kuk, Gun Hwa Park, Juhyeok Kim, Hanna Ryu, Jongbum Kim, Mira Roh, Jeonghun Baek, Michael W. Hunkapiller, Jonas Korlach, Jong-Yeon Shin, and Changhoon Kim. De novo assembly and phasing of a korean human genome. *Nature*, 538:243 EP –, Oct 2016.

[36] Ariya Shajii, Ibrahim Numanagić, and Bonnie Berger. Latent variable model for aligning barcoded short-reads improves downstream analyses. *bioRxiv*, 2017.

[37] O. Tange. Gnu parallel - the command-line power tool. *;login: The USENIX Magazine*, 36(1):42–47, Feb 2011.

[38] Greyson P Twist, Andrea Gaedigk, Neil A Miller, Emily G Farrow, Laurel K Willig, Darrell L Dinwiddie, Josh E Petrikin, Sarah E Soden, Suzanne Herd, Margaret Gibson, Julie A Cakici, Amanda K Riffel, J Steven Leeder, Deendayal Dinakarpandian, and Stephen F Kingsmore. Constellation: A tool for rapid, automated phenotype assignment of a highly polymorphic pharmacogene, CYP2D6, from whole-genome sequences. *npj Genomic Medicine*, 1:15007, January 2016. `http://www.nature.com/articles/npjgenmed20157`.

[39] Yue Wang, Qiuping Yang, and Zhimin Wang. The Evolution of Nanopore Sequencing. *Frontiers in Genetics*, 5:449, 2014.

[40] Deniz Yorukoglu, Yun William Yu, Jian Peng, and Bonnie Berger. Compressive mapping for next-generation sequencing. *Nat Biotech*, 34(4):374–376, Apr 2016.

[41] Matei Zaharia, William J. Bolosky, Kristal Curtis, Armando Fox, David A. Patterson, Scott Shenker, Ion Stoica, Richard M. Karp, and Taylor Sittler. Faster and more accurate sequence alignment with SNAP. *CoRR*, abs/1111.5572, 2011.

[42] Mengyao Zhao, Wan-Ping Lee, Erik P. Garrison, and Gabor T. Marth. Ssw library: An simd smith-waterman c/c++ library for use in genomic applications. *PLOS ONE*, 8(12), 12 2013.

[43] Grace XY Zheng, Billy T. Lau, Michael Schnall-Levin, Mirna Jarosz, John M. Bell, Christopher M. Hindson, Sofia Kyriazopoulou-Panagiotopoulou, Donald A. Masquelier, Landon Merrill, Jessica M. Terry, Patrice A. Mudivarti, Paul W. Wyatt, Rajiv Bharadwaj, Anthony J. Makarewicz, Yuan Li, Phillip Belgrader, Andrew D. Price, Adam J. Lowe, Patrick Marks, Gerard M. Vurens, Paul Hardenbol, Luz Montesclaros, Melissa Luo, Lawrence Greenfield, Alexander Wong, David E. Birch, Steven W. Short, Keith P. Bjornson, Pranav Patel, Erik S. Hopmans, Christina Wood, Sukhvinder Kaur, Glenn K. Lockwood, David Stafford, Joshua P. Delaney, Indira Wu, Heather S. Ordonez, Susan M. Grimes, Stephanie Greer, Josephine Y. Lee, Kamila Belhocine, Kristina M. Giorda, William H.

Heaton, Geoffrey P. McDermott, Zachary W. Bent, Francesca Meschi, Nikola O. Kondov, Ryan Wilson, Jorge A. Bernate, Shawn Gauby, Alex Kindwall, Clara Bermejo, Adrian N. Fehr, Adrian Chan, Serge Saxonov, Kevin D. Ness, Benjamin J. Hindson, and Hanlee P. Ji. Haplotyping germline and cancer genomes using high-throughput linked-read sequencing. *Nat Biotechnol*, 34(3):303–311, Mar 2016. 26829319[pmid].

[44] Christoph Ziegenhain, Beate Vieth, Swati Parekh, Björn Reinius, Amy Guillaumet-Adkins, Martha Smets, Heinrich Leonhardt, Holger Heyn, Ines Hellmann, and Wolfgang Enard. Comparative analysis of single-cell rna sequencing methods. *Molecular cell*, 65(4):631–643, 2017.

[45] Justin M. Zook, David Catoe, Jennifer McDaniel, Lindsay Vang, Noah Spies, Arend Sidow, Ziming Weng, Yuling Liu, Christopher E. Mason, Noah Alexander, and et al. Extensive sequencing of seven human genomes to characterize benchmark reference materials. *Scientific Data*, 3:160025, Jun 2016.

[46] Justin M Zook, Brad Chapman, Jason Wang, David Mittelman, Oliver Hofmann, Winston Hide, and Marc Salit. Integrating human sequence data sets provides a resource of benchmark SNP and indel genotype calls. *Nat Biotech*, 32:246–251, 2014.