bioRxiv preprint doi: https://doi.org/10.1101/293944; this version posted April 12, 2018. The copyright holder for this preprint (which was not certified by peer review) is the author/funder, who has granted bioRxiv a license to display the preprint in perpetuity. It is made available under aCC-BY-ND 4.0 International license.

Haplotype-aware genotyping from noisy long reads

Jana Ebler[1,2,*], **Marina Haukness**[3,*], **Trevor Pesout**[3,*], **Tobias Marschall**[1,2,†], **and Benedict Paten**[3,†]

[1]Center for Bioinformatics, Saarland University, Saarbrücken, Germany
[2]Max Planck Institute for Informatics, Saarbrücken, Germany
[3]UC Santa Cruz Genomics Institute, University of California Santa Cruz, Santa Cruz, CA 95064, USA
[*]These authors contributed equally to this work
[†]Joint last/corresponding authors.

Motivation: **Current genotyping approaches for single nucleotide variations (SNVs) rely on short, relatively accurate reads from second generation sequencing devices. Presently, third generation sequencing platforms able to generate much longer reads are becoming more widespread. These platforms come with the significant drawback of higher sequencing error rates, which makes them ill-suited to current genotyping algorithms. However, the longer reads make more of the genome unambiguously mappable and typically provide linkage information between neighboring variants.**

Results: **In this paper we introduce a novel approach for haplotype-aware genotyping from noisy long reads. We do this by considering bipartitions of the sequencing reads, corresponding to the two haplotypes. We formalize the computational problem in terms of a Hidden Markov Model and compute posterior genotype probabilities using the forward-backward algorithm. Genotype predictions can then be made by picking the most likely genotype at each site. Our experiments indicate that longer reads allow significantly more of the genome to potentially be accurately genotyped. Further, we are able to use both Oxford Nanopore and Pacific Biosciences sequencing data to independently validate millions of variants previously identified by short-read technologies in the reference NA12878 sample, including hundreds of thousands of variants that were not previously included in the high-confidence reference set.**

Correspondence: ***t.marschall@mpi-inf.mpg.de, bpaten@ucsc.edu***

1 Introduction

Reference-based genetic variant identification comprises two related processes: genotyping and phasing. Genotyping is the process of determining which genetic variants are present in an individual's genome. A genotype at a given site describes whether both chromosomal copies carry a variant allele, only one of them carries it, or whether the variant allele is not present at all. Phasing refers to determining an individual's haplotypes, which consist of variants that lie near each other on the same chromosome and are inherited together. To completely describe all of the genetic variation in an organism, both genotyping and phasing are needed. Together, the two processes are called *diplotyping*.

Many existing variant analysis pipelines are designed for short DNA sequencing reads (1, 2). Though short reads are very accurate at a per-base level, they can suffer from being difficult to unambiguously align to the genome, especially in repetitive or duplicated regions (3). The result is that millions of bases of the reference human genome are not currently reliably genotyped by short reads, primarily in multi-megabase gaps near the centromeres and short arms of chromosomes (4). While short reads are unable to uniquely map to these regions, long reads can potentially span into or even across them. This makes it so long reads are advantageous over short reads for tasks such as haplotyping, large structural variant detection, and *de novo* assembly (5–8). Here, we attempt to demonstrate the utility of long reads for more comprehensive genotyping.

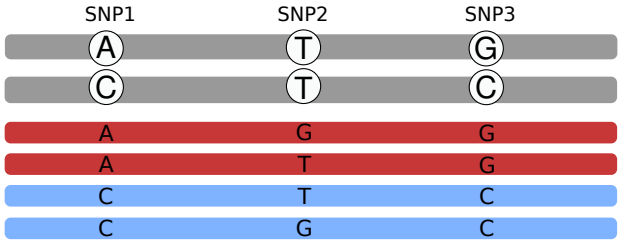

Fig. 1. ***Motivation.*** *Gray sequences illustrate the haplotypes; the reads are shown in red and blue. The red reads originate from the upper haplotype, the blue ones from the lower. Genotyping each SNV individually would lead to the conclusion that all of them are heterozygous. Using the haplotype context reveals uncertainty about the genotype of the second SNV.*

Long read DNA sequencing technologies are rapidly falling in price and increasing in general availability. Such technologies include Single Molecule Real Time (SMRT) Sequencing by Pacific Biosciences (PacBio) and nanopore sequencing by Oxford Nanopore Technologies (ONT). However, due to the historically greater relative cost and higher sequencing error rates of these technologies, little attention has been given thus far to the problem of genotyping single nucleotide variants (SNVs) with long reads. Recently, (9) have taken first steps in this direction, but their approach does not scale to process whole human genomes in reasonable time.

For an illustration of the benefit of using long reads to diplotype, consider Figure 1. Shown are three SNV positions covered by long reads. The gray sequences represent the true haplotype sequences and reads are colored in blue and red. The colors correspond to the haplotype which the respective read stems from: the red ones from the upper sequence, and the blue ones from the lower one. Since sequencing errors can occur, the alleles supported by the reads are not always equal to the true ones in the haplotypes shown in gray. Considering the SNVs individually, we would probably genotype the first one as A/C, the second one as T/G and the third one as G/C, since the number of reads supporting each al-

bioRxiv preprint doi: https://doi.org/10.1101/293944; this version posted April 12, 2018. The copyright holder for this preprint (which was not certified by peer review) is the author/funder, who has granted bioRxiv a license to display the preprint in perpetuity. It is made available under aCC-BY-ND 4.0 International license.

lele are the same. This leads to a wrong genotype prediction for the second SNV. However, if we knew which haplotype each read stems from, that is, if we knew their colors, then we would be unsure about the genotype of the second SNV. It could also be G/G or T/T, since the reads stemming from the same haplotypes must support the same alleles. Therefore, using haplotype information during genotyping makes it possible to compute more reliable genotype predictions and to detect uncertainties.

Contributions. In this paper, we show that for contemporary long read technologies, read-based phase inference can be simultaneously combined with the genotyping process for SNVs to produce accurate diplotypes and to detect variants in regions not mappable by short reads. We show that key to this inference is the detection of linkage relationships between heterozygous sites within the reads. To do this, we describe a novel algorithm to accurately predict diplotypes from noisy long reads that scales to deeply sequenced human genomes. We achieve this by considering bipartitions of all given sequencing reads, corresponding to the two haplotypes of an individual. The problem is formalized using a Hidden Markov Model (HMM) from which we compute genotype likelihoods using the forward-backward algorithm and make genotype predictions by determining the likeliest genotype at each position.

We then apply this algorithm to diplotype one individual from the 1000 Genomes Project, NA12878, using long reads from both PacBio and ONT. NA12878 has been extensively sequenced and studied, and the Genome in a Bottle consortium has published sets of highly confident variant calls (10). We demonstrate that our method is accurate, that it can be used to confirm variants in regions of uncertainty, and that it allows for the discovery of variants in regions which are unmappable using short DNA read sequencing technologies.

2 Methods

We describe a probabilistic model for diplotype and genotype inference, and in this paper use it to find maximum posterior probability genotypes. The approach builds upon the WhatsHap approach (11), but incorporates a full probabilistic allele inference model into the problem. It has similarities to that proposed by Kuleshov et al. (12), but we here frame the problem using Hidden Markov Models (HMMs).

2.1 Alignment Matrix

Let $\mathbf{M}$ be an alignment matrix whose rows represent sequencing *reads* and whose columns represent genetic *sites*. Let m be the number of rows, let n be the number of columns, and let $\mathbf{M}_{i,j}$ be the jth element in the ith row. In each column let $\Sigma_j \subset \Sigma$ represent the set of possible *alleles* such that $\mathbf{M}_{i,j} \in \Sigma_j \cup \{-\}$, the "$-$" gap symbol representing a site at which the read provides no information. We assume no row or column is composed only of gap symbols, an uninteresting edge case. An example alignment matrix is shown in Figure 2. Throughout the following we will be informal and refer to a row i or column j, being clear from the context whether we are referring to the row or column itself or the coordinate.

```
  1 2 3 4 5
1 A G T - -
2 A G T - -
3 - C - G -
4 - C T G -
5 - - T C T
6 - - T C T
```

Fig. 2. ***Alignment Matrix.*** *Here, the alphabet of possible alleles is the set of DNA nucleotides, i.e.* $\Sigma = \{A, C, G, T\}$

2.2 Genotype Inference Problem Overview

A diplotype $H = (H^1, H^2)$ is a pair of haplotype (segments); a *haplotype (segment)* $H^k = H^k_1, H^k_2, \ldots, H^k_n$ is a sequence of length n whose elements represents alleles such that $H^k_j \in \Sigma_j$. Let $B = (B^1, B^2)$ be a bipartition of the rows of $\mathbf{M}$ into two parts (sets): B^1, the first part, and B^2, the second part. We use bipartitions to represent which haplotypes the reads came from, of the two in a genome. By convention we assume that the first part of B are the reads arising from H^1 and the second part of B are the reads arising from H^2.

The problem we analyze is based upon a probabilistic model that essentially represents the (Weighted) Minimum Error Correction (MEC) problem (13, 14), while modeling the evolutionary relationship between the two haplotypes and so imposing a cost on bipartitions that create differences between the inferred haplotypes.

For a bipartition B, and making an i.i.d. assumption between sites in the reads:

$$P(H|B,\mathbf{M}) = \prod_{j=1}^{n} \sum_{Z_j \in \Sigma_j} P(H^1_j|B^1, Z_j) P(H^2_j|B^2, Z_j) P(Z_j)$$

Here $P(Z_j)$ is the prior probability of the ancestral allele Z_j of the two haplotypes at column j, by default we can use a simple flat distribution over ancestral alleles (but see below). The posterior probability $P(H^k_j|B^k, Z_j) =$

$$\frac{P(H^k_j|Z_j) \prod_{\{i \in B^k : \mathbf{M}_{i,j} \neq -\}} P(\mathbf{M}_{i,j}|H^k_j)}{\sum_{Y_j \in \Sigma_j} P(Y_j|Z_j) \prod_{\{i \in B^k : \mathbf{M}_{i,j} \neq -\}} P(\mathbf{M}_{i,j}|Y_j)}$$

for $k \in \{1, 2\}$, where the probability $P(H^k_j|Z_j)$ is the probability of the haplotype allele H^k_j given the ancestral allele Z_j. For this we can use a continuous time Markov model for allele substitutions, such as Jukes-Cantor (15), or some more sophisticated model that factors the similarities between alleles (see below). Similarly, $P(\mathbf{M}_{i,j}|H^k_j)$ is the probability of observing allele $\mathbf{M}_{i,j}$ in a read given the haplotype allele H^k_j.

The genotype inference problem we consider is finding for each site:

$$\underset{(H^1_j, H^2_j)}{\arg\max}\, P(H^1_j, H^2_j|\mathbf{M}) = \underset{(H^1_j, H^2_j)}{\arg\max} \sum_{B} P(H^1_j, H^2_j|B, \mathbf{M})$$

bioRxiv preprint doi: https://doi.org/10.1101/293944; this version posted April 12, 2018. The copyright holder for this preprint (which was not certified by peer review) is the author/funder, who has granted bioRxiv a license to display the preprint in perpetuity. It is made available under aCC-BY-ND 4.0 International license.

i.e. finding the genotype (H_j^1, H_j^2) with maximum posterior probability for a generative model of the reads embedded in $\mathbf{M}$.

2.3 A Graphical Representation Of Read Partitions

For a column j in $\mathbf{M}$, a row i is *active* if the first non-gap symbol in row i occurs at or before column j and the last non-gap symbol in row i occurs at or after column j. Let A_j be the set of active rows of column j. For a column j a row i is *terminal* if its last non-gap symbol occurs at column j or $j = n$. Let A'_j be the set of active, non-terminal rows of column j.

Let $B_j = (B_j^1, B_j^2)$ be a bipartition of A_j into a first part B_j^1 and a second part B_j^2. Let $\mathbf{B_j}$ be the set of all possible such bipartitions of the active rows of j. Similarly, let $C_j = (C_j^1, C_j^2)$ be a bipartition of A'_j, and $\mathbf{C_j}$ be the set of all possible such bipartitions of the active, non-terminal rows of j.

For two bipartitions $B = (B^1, B^2)$ and $C = (C^1, C^2)$, B is *compatible* with C if the subset of B^1 in $C^1 \cup C^2$ is a subset of C^1, and, similarly, the subset of B^2 in $C^1 \cup C^2$ is a subset of C^2. Note this definition is symmetric and reflexive, although not transitive.

Let $G = (V_G, E_G)$ be a directed graph. The vertices V_G are the set of bipartitions of both the active rows and the active, non-terminal rows for all columns of $\mathbf{M}$ and a special *start* and *end* vertex, i.e. $V_G = \{start, end\} \cup (\bigcup_j \mathbf{B_j} \cup \mathbf{C_j})$. The edges E_G are a subset of compatibility relationships, such that (1) for all j there is an edge $(B_j \in \mathbf{B_j}, C_i \in \mathbf{C_j})$ if B_j is compatible with C_j, (2) for all $0 < j < n$ there is an edge $(C_j \in \mathbf{C_j}, B_{j+1} \in \mathbf{B_{j+1}})$ if C_j is compatible with B_{j+1}, (3) there is an edge from the start vertex to each member of $\mathbf{B_1}$, and (4) there is an edge from each member of $\mathbf{B_n}$ to the end vertex (Note that $\mathbf{C_n}$ is empty and so contributes no vertices to G). Figure 3 shows an example graph.

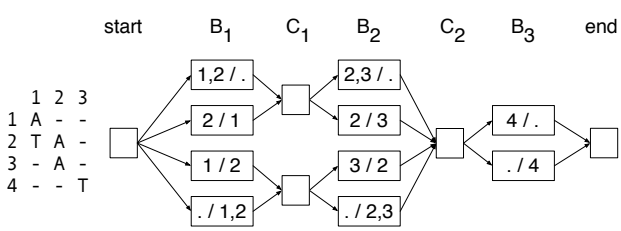

Fig. 3. ***Example Graph.*** *Left: An alignment matrix. Right: The corresponding directed graph representing the bipartitions of active rows and active non-terminal rows, where the labels of the nodes indicate the partitions, e.g. '1,2 / .' is shorthand for* $A = (\{1,2\},\{\})$.

The graph G has a large degree of symmetry and the following properties are easily verified:

- For all j and all $B_j \in \mathbf{B_j}$, the indegree and outdegree of B_j is 1.
- For all j the indegree of all members of $\mathbf{C_j}$ is equal.
- Similarly, for all j the outdegree of all members of $\mathbf{C_j}$ is equal.

Let the *maximum coverage*, denoted $maxCov$, be the maximum cardinality of a set A_j over all j. By definition, $maxCov \leq m$. Using the above properties it is easily verified that: (1) the cardinality of G (number of vertices) is bounded by this maximum coverage, being less than or equal to $2 + (2n-1)2^{maxCov}$, and (2) the size of G (number of edges) is at most $2n2^{maxCov}$.

Let a directed path from the start vertex to the end vertex be called a *diploid path*, $D = (D_1 = start, D_2, \ldots, D_{2n+1} = end)$. The graph is naturally organized by the columns of $\mathbf{M}$, so that $D_{2j} = (B_j^1, B_j^2) \in \mathbf{B_j}$ and $D_{2j+1} = (C_{j+1}^1, C_{j+1}^2) \in \mathbf{C_j}$ for all $0 < j \leq n$. Let $B_D = (B_D^1, B_D^2)$ denote a pair of sets, where B_D^1 is the union of the first parts of the vertices of $D_2, \ldots, D_{2n+1}$ and, similarly, B_D^2 is the union of second parts of the vertices of $D_2, \ldots, D_{2n+1}$.

B_D^1 and B_D^2 are disjoint because otherwise there must exist a pair of vertices within D that are incompatible, which is easily verified to be impossible. Further, because D visits a vertex for every column of $\mathbf{M}$, it follows that the sum of the cardinalities of these two sets is m. B_D is therefore a bipartition of the rows of $\mathbf{M}$ which we call a *diploid path bipartition*.

Lemma 1: The set of diploid path bipartitions is the set of bipartitions of the rows of $\mathbf{M}$ and each diploid path defines a unique diploid path bipartition.

Proof: We first prove that each diploid path defines a unique bipartition of the rows of $\mathbf{M}$. For each column j of $\mathbf{M}$, each vertex $B_j \in \mathbf{B_j}$ is a different bipartition of the same set of active rows. B_j is by definition compatible with a diploid path bipartition of a diploid path that contains it, and incompatible with every other member of $\mathbf{B_j}$. It follows that for each column j two diploid paths with the same diploid path bipartition must visit the same node in $\mathbf{B_j}$, and, by identical logic, the same node in $\mathbf{C_j}$, but then two such diploid paths are therefore equal.

There are 2^m partitions of the rows of $\mathbf{M}$. It remains to prove that there are 2^m diploid paths. By the structure of the graph, the set of diploid paths can be enumerated backwards by traversing right-to-left from the end vertex by depth-first search and exploring each incoming edge for all encountered nodes. As stated previously, the only vertices with indegree greater than one are for all j the members of $\mathbf{C_j}$, and each member of $\mathbf{C_j}$ has the same indegree. For all j the indegree of C_j is clearly $2^{|C_j|-|B_j|}$: two to the power of the number of number of active, terminal rows at column j. The number of possible paths must therefore be $\prod_{j=1}^{n} 2^{|C_j|-|B_j|}$. As each row is active and terminal in exactly one column, we obtain $m = \sum_j |C_j| - |B_j|$ and therefore:

$$2^m = \prod_{j=1}^{n} 2^{|C_j|-|B_j|}$$

. ■

bioRxiv preprint doi: https://doi.org/10.1101/293944; this version posted April 12, 2018. The copyright holder for this preprint (which was not certified by peer review) is the author/funder, who has granted bioRxiv a license to display the preprint in perpetuity. It is made available under aCC-BY-ND 4.0 International license.

2.4 A Hidden Markov Model For Genotype and Diplotype Inference

In order to infer diplotypes, we define a Hidden Markov Model which is based on G, but additionally represents all possible genotypes at each genomic site (i.e. in each B column). To this end, we define the set of states $\mathbf{B_j} \times \Sigma_j \times \Sigma_j$, which contains a state for each bipartition of the active rows at position j and all possible assignments of alleles in Σ_j to the two partitions. Additionally, the HMM contains a hidden state for each bipartition in $\mathbf{C_j}$, exactly as defined for G above. Transitions between states are defined by the compatibility relationships of the corresponding bipartitions as before. This HMM construction is illustrated in Figure 4.

For all j and all $C_j \in \mathbf{C_j}$ each outgoing edge has transition probability $P(a_1, a_2) = \sum_{Z_j} P(a_1|Z_j)P(a_2|Z_j)P(Z_j)$, where $(B_j, a_1, a_2) \in \mathbf{B_j} \times \Sigma_j \times \Sigma_j$ is the state being transitioned to. Similarly, each outgoing edge of the start node has transition probability $P(a_1, a_2)$. The outdegree of all remaining nodes is 1, so these edges have transition probability 1.

The start node, the end node, and members of $\mathbf{C_j}$ for all j are silent states, and hence do not emit symbols. For all j, members of $\mathbf{B_j} \times \Sigma_j \times \Sigma_j$ output the entries in the j-th column of $\mathbf{M}$ that are different from "–". We assume every matrix entry to be associated with an error probability, which we can compute from $P(\mathbf{M}_{ij}|H_j^k)$ defined previously. Based on this, the probability of observing a specific output column of $\mathbf{M}$ can be easily calculated.

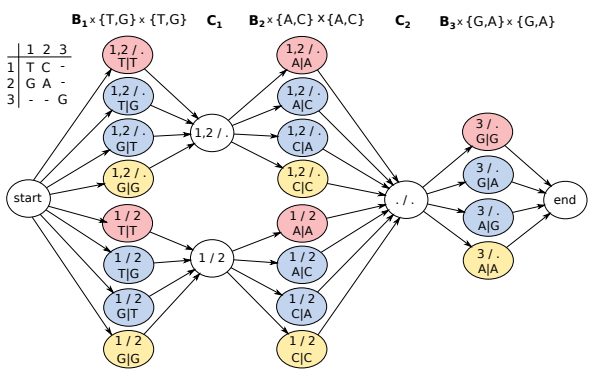

Fig. 4. ***Genotyping HMM.*** *Colored states correspond to bipartitions of reads and allele assignments at that position. States in C_1 and C_2 correspond to bipartitions of reads covering positions 1 and 2 or 2 and 3, respectively. In order to compute genotype likelihoods after running the forward-backward algorithm, states of the same color have to be summed up in each column.*

2.4.1 Computing Genotype Likelihoods

The goal is to compute genotype likelihoods for the possible genotypes for each variant position using the HMM defined above. Performing the forward-backward algorithm returns forward and backward probabilities of all hidden states. Using those, the posterior distribution of a state $(B, a_1, a_2) \in \mathbf{B_j} \times \Sigma_j \times \Sigma_j$, corresponding to bipartition B and assigned alleles a_1 and a_2, can be computed as

$$P((B, a_1, a_2)|\mathbf{M}) = \frac{\alpha_j(B, a_1, a_2) \cdot \beta_j(B, a_1, a_2)}{\sum_{B' \in \mathcal{B}(A_j)} \sum_{a'_1, a'_2 \in \Sigma_j} \alpha_j(B', a'_1, a'_2) \cdot \beta_j(B', a'_1, a'_2)} \quad \textbf{(1)}$$

where $\alpha_j(B, a_1, a_2)$ and $\beta_j(B, a_1, a_2)$ denote forward and backward probabilities of the state (B, a_1, a_2) and $\mathcal{B}(A_j)$, the set of all bipartitions of A_j. The above term represents the probability for a bipartition $B = (B^1, B^2)$ of the reads in A_j and alleles a_1 and a_2 assigned to these partitions. In order to finally compute the likelihood for a certain genotype, one can marginalize over all bipartitions of a column, and all allele assignments corresponding to that genotype.

Example 2.1: In order to compute genotype likelihoods for each column of the alignment matrix, posterior state probabilities corresponding to states of the same color in Figure 4 need to be summed up. For the first column, adding up the red probabilities gives the genotype likelihood of genotype T/T, blue of genotype G/T and yellow of G/G.

2.5 Implementations

We created two independent software implementations of this model, one based upon WhatsHap and one from scratch, which we call MarginPhase. Each uses different optimizations and heuristics that we briefly describe.

2.5.1 WhatsHap Implementation

We extended the implementation of WhatsHap (11, `bitbucket.org/whatshap/whatshap`) to enable haplotype aware genotyping of bi-allelic variants based on the above model. WhatsHap focuses on re-genotyping variants, i.e. it assumes SNV positions to be given. In order to detect variants, a simple SNV calling pipeline was developed. It is based on `samtools mpileup` (16) which provides information about the bases supported by each read covering a genomic position. A set of SNV candidates is generated by selecting genomic positions at which the frequency of a non-reference allele is above a fixed threshold (0.25 for PacBio data, 0.4 for Nanopore data) and the absolute number of reads supporting the non-reference allele is at least 3.

Allele Detection. In order to construct the alignment matrix, a crucial step is to determine whether each read supports the reference or the alternative allele at each of n given genomic positions. In WhatsHap, this is done based on realigning sections of the reads (17). Given an existing read alignment from the provided BAM file, its sequence in a window around the variant is extracted. It is aligned to the corresponding region of the reference sequence and additionally, to the alternative sequence, which is artificially produced by inserting the alternative allele into the reference. The alignment cost is computed by using affine gap costs. Phred scores representing the probabilities for opening and extending a gap and for a mismatch in the alignment can be estimated from the given BAM file. The allele leading to a lower alignment cost is assumed to be supported by the read and is reported in the alignment matrix. If both alleles lead to the same cost, the corresponding matrix entry is "–". The absolute difference of both alignment scores is assigned as a weight to the corresponding entry in the alignment matrix. It can be interpreted as a phred scaled probability for the allele being wrong and is utilized for the computation of output

bioRxiv preprint doi: https://doi.org/10.1101/293944; this version posted April 12, 2018. The copyright holder for this preprint (which was not certified by peer review) is the author/funder, who has granted bioRxiv a license to display the preprint in perpetuity. It is made available under aCC-BY-ND 4.0 International license.

probabilities.

Read Selection. Our algorithm enumerates all bipartitions of reads covering a variant position and thus has a runtime exponential in the maximum coverage of the data. To ensure that this quantity is bounded, the same read selection step implemented previously in the WhatsHap software is run before constructing the HMM and computing genotype likelihoods. Briefly, a heuristic approach described in (18) is applied, which selects phase informative reads iteratively taking into account the number of heterozygous variants covered by the read and its quality.

Transitions. Defining separate states for each allele assignment in $\mathbf{B_j}$ enables easy incorporation of prior genotype likelihoods by weighting transitions between states in $\mathbf{C_{j-1}}$ and $\mathbf{B_j} \times \Sigma_j \times \Sigma_j$. Since there are two states corresponding to a heterozygous genotype in the bi-allelic case ($0|1$ and $1|0$), the prior probability for the heterozygous genotype is equally spread between these states.
In order to compute such genotype priors, the same likelihood function underlying the approaches described in (19) and (20) was utilized. For each SNV position, the model computes a likelihood for each SNV to be absent, heterozygous, or homozygous based on all reads that cover a particular site. Each read contributes a probability term to the likelihood function, which is computed based on whether it supports the reference or the alternative allele (19). Furthermore, the approach accounts for statistical uncertainties arising from read mapping and has a runtime linear in the number of variants to be genotyped (20). Prior genotype likelihoods are computed before read selection. In this way, information of all input reads covering a position can be incorporated.

2.5.2 MarginPhase Implementation

MarginPhase (`github.com/benedictpaten/marginPhase`) is an experimental, open source implementation of the described HMM written in C. It differs from the WhatsHap implementation in the method it uses to explore bipartitions and the method to generate allele support probabilities from the reads.

Read Bipartitions. The described HMM scales exponentially in terms of increasing read coverage. For typical 20-60x sequencing coverage (i.e. average number of active rows per column) it is impractical to store all possible bipartitions of the rows of the matrix. MarginPhase implements a simple, greedy pruning and merging heuristic outlined in recursive pseudocode in Algorithm 1.
The procedure computePrunedHMM takes an alignment matrix and returns a connected subgraph of the HMM for $\mathbf{M}$ that can be used for inference, choosing to divide the input alignment matrix into two if the number of rows exceeds a threshold t, recursively.
The sub-procedure mergeHMMs takes two pruned HMMs for two disjoint alignment matrices with the same number of columns and joins them together in the natural way such that if at each site i there are $|\mathbf{B_i^1}|$ states in $\mathbf{HMM_1}$ and $|\mathbf{B_i^2}|$ in $\mathbf{HMM_2}$ then the resulting HMM will have $|\mathbf{B_i^1}| \times |\mathbf{B_i^2}|$

Algorithm 1

procedure COMPUTEPRUNEDHMM($\mathbf{M}$)
  if maxCov $\geq t$ **then**
    Divide $\mathbf{M}$ in half to create two matrices, $\mathbf{M_1}$ and $\mathbf{M_2}$, such that $\mathbf{M_1}$ is the first $\frac{n}{2}$ rows of $\mathbf{M}$ and $\mathbf{M_2}$ is the remaining rows of $\mathbf{M}$.
    $\mathbf{HMM_1} \leftarrow$ computePrunedHMM($\mathbf{M_1}$)
    $\mathbf{HMM_2} \leftarrow$ computePrunedHMM($\mathbf{M_2}$)
    $\mathbf{HMM} \leftarrow$ mergeHMMs($\mathbf{HMM_1}, \mathbf{HMM_2}$)
  else
    Let **HMM** be the read partitioning HMM for $\mathbf{M}$.
  return subgraph of **HMM** including visited states and transitions each with posterior probability of being visited $\geq v$, and which are on a path from the start to end nodes.

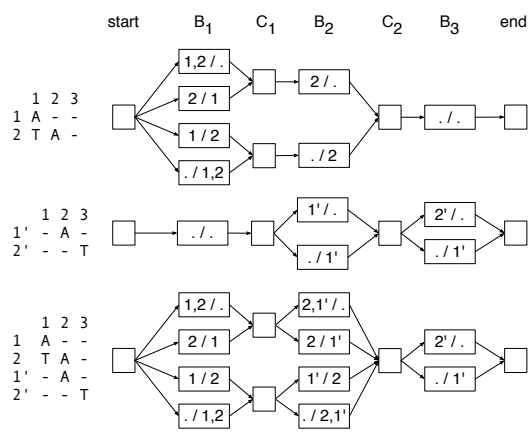

Fig. 5. *The merger of two read partitioning HMMs with the same number of columns. Top and middle: Two HMMs to be merged; bottom: the merged HMM. Transition and emission probabilities not shown.*

states. This is illustrated in Figure 5. In the experiments used here $t = 8$ and $v = 0.01$.

Allele Supports. In MarginPhase, the alignment matrix has a site for each base in the reference genome. To generate the allele support from the reads, for each read we calculate the posterior probability of each allele using the implementation of the banded forward-backward pairwise alignment described in (21). The result is that for each reference base, for each read that overlaps (according to an initial guide alignment extracted from the SAM/BAM file) the reference base we calculate the probability of each possible nucleotide (i.e. { 'A', 'C', 'G', 'T' }). Gaps are ignored and treated as missing data. This approach allows summation over all alignments within the band.

3 Results

3.1 Data Preparation and Evaluation

To test our methods, we used sequencing data for NA12878 from two different long read sequencing technologies. NA12878 is a participant from the 1000 Genomes Project (2) who has been extensively sequenced and analyzed. We used Oxford Nanopore reads from (7) and PacBio reads from (26).

bioRxiv preprint doi: https://doi.org/10.1101/293944; this version posted April 12, 2018. The copyright holder for this preprint (which was not certified by peer review) is the author/funder, who has granted bioRxiv a license to display the preprint in perpetuity. It is made available under aCC-BY-ND 4.0 International license.

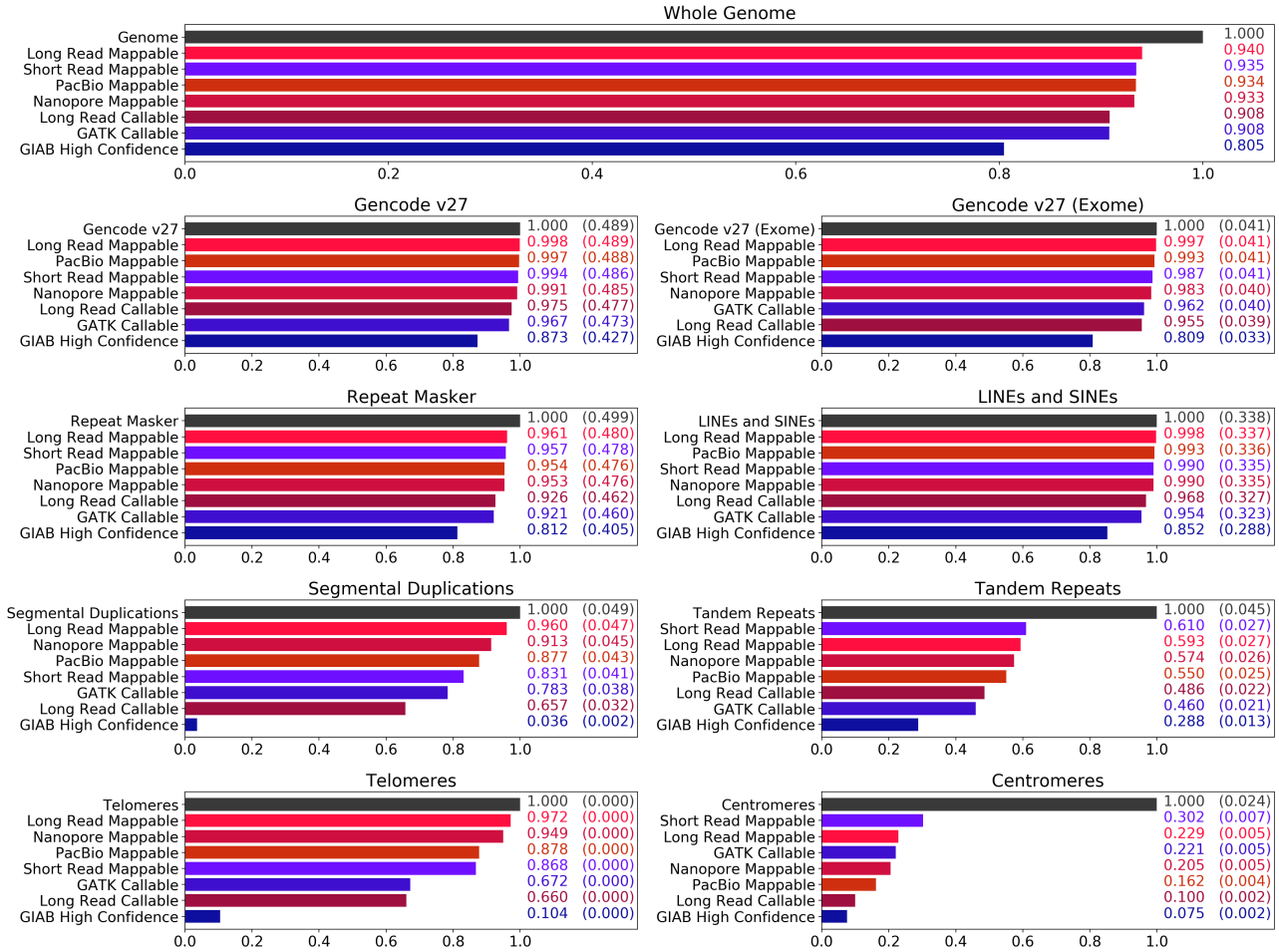

Fig. 6. ***Reach of short read and long read technologies.*** *The callable and mappable regions for NA12878 spanning various repetitive or duplicated sequences on GRCh38 are shown. Feature locations are determined based on BED tracks downloaded from the UCSC Genome Browser (22). Other than the Gencode regions (23, 24), all features are subsets of the Repeat Masker (25) track. Four coverage statistics for long reads (shades of red) and three for short reads (shades of blue) are shown. The labels 'PacBio Mappable' and 'Nanopore Mappable' describe areas where at least one primary read with GQ $\geq$ 30 has mapped, and 'Long Read Mappable' describes where this is true for at least one of the long read technologies. 'Long Read Callable' describes areas where both read technologies have coverage of at least 10 and less than twice the median coverage. 'GIAB High Confidence', 'GATK Callable' and 'Short Read Mappable' are the regions associated with the evaluation callsets. For the feature-specific plots, the numbers on the right detail coverage over the feature and coverage over the whole genome (parenthesized).*

Both sets of reads were aligned to GRCh38 with `minimap2`, a mapper designed to align error-prone long reads (27).

To ensure that any variants we found were not artifacts of misalignment, we filtered out reads flagged as secondary or supplementary, as well as reads with a mapping quality score less than 30. Genome-wide, this left approximately 12 million Nanopore reads and 34 million PacBio reads. The Nanopore reads had a median depth of 37× and length of 5950, including a set of ultra-long reads with lengths up to 900 kilobases. The PacBio reads had a median depth of 46× and length of 2650.

To validate the performance of our methods, we used callsets from Genome in a Bottle's (GIAB) benchmark small variant calls v3.3.2 (10). First, we compared against GIAB's set of high confidence calls, generated by a consensus algorithm spanning multiple sequencing technologies and variant calling programs. The high confidence regions associated with this callset exclude structural variants, centromeres, and heterochromatin. We used this to show our method's accuracy in well-understood and easy-to-map regions of the genome.

We also analyzed our results compared to two callsets which were used in the construction of GIAB's high confidence variants, one made by GATK HaplotypeCaller v3.5 (GATK/HC, 1) and the other by Freebayes 0.9.20 (28), both generated from a 300× PCR-free Illumina sequencing run (10).

All of our evaluation statistics were generated with the tool `vcfeval` from Real Time Genomics (29). We restrict the analysis to SNVs due to the error distribution of both PacBio and Nanopore long reads which leads to insertions and deletions being the most common type of sequencing error by far (30, 31).

Short read variant callers. We explored the suitability of current state-of-the-art callers for short reads to process long read data (using default settings), but were unsuccessful. The absence of base qualities in the PacBio data prevented any calling; for Nanopore data, FreeBayes was prohibitively slow and neither Platypus nor GATK/HC produced calls.

bioRxiv preprint doi: https://doi.org/10.1101/293944; this version posted April 12, 2018. The copyright holder for this preprint (which was not certified by peer review) is the author/funder, who has granted bioRxiv a license to display the preprint in perpetuity. It is made available under aCC-BY-ND 4.0 International license.

3.2 Long Read Coverage

We determined the regions where long and short reads can be mapped to the human genome. In Figure 6, various coverage metrics for short and long reads are plotted against different genomic features, which were mostly selected for being repetitive or duplicated.

The callsets on the Illumina data made by GATK/HC and FreeBayes come with two BED files describing where calls were made with some confidence. The first, described in Figure 6 as *Short Read Mappable*, was generated using GATK CallableLoci v3.5 and includes regions where there is a) at least a read depth of 20, and b) at most a depth of twice the median depth, only including reads with mapping quality of at least 20. This definition of callable only considers read mappings. The second, described as *GATK Callable*, was generated from the GVCF output from GATK/HC by excluding areas with genotype quality less than 60. This is a more sophisticated definition of callable as it reflects the effects of homopolymers and tandem repeats. We use these two BED files in our analysis of how short and long reads map differently in various areas of the genome.

For long reads, we show four coverage statistics. The records marked as "Mappable" describe areas where there is at least one high quality long read mapping (PacBio, Nanopore, and *Long Read Mappable* for areas where at least one of the technologies mapped). The *Long Read Callable* entries cover a conservative region which has a sufficient read depth to illustrate the efficacy of our method; it covers regions where both sequencing technologies had a minimum depth of ten and maximum of $2\times$ the median depth (similar to the CallableLoci metric).

Figure 6 shows that in almost all cases, long reads map to more area than is mappable by short reads. For example, nearly half a percent of the genome is mappable by long reads but not short reads. Long reads also map to one percent more of the exome, and thirteen percent more of segmental duplications. Centromeres and Tandem Repeats are outliers to this generalization, where neither PacBio nor Nanopore cover appreciably more than Illumina.

3.3 Comparison Against High Confidence Truthset

To validate our method, we first analyzed the SNV detection and genotyping performance of our algorithm using the GIAB high confidence callset as a benchmark. All variants reported in these statistics fall within the GIAB high confidence regions.

Figure 7 (top) shows precision and recall of our algorithms on both the PacBio and Oxford Nanopore data sets. MarginPhase and WhatsHap perform similarly overall. MarginPhase achieved higher precision and recall on Nanopore reads, with precision of 0.7686 and recall of 0.8089, compared to WhatsHap's precision of 0.7131 and recall of 0.7248 on the same set of Nanopore reads. WhatsHap obtained better results on PacBio data, with a precision of 0.9738 and recall of 0.9593, compared to MarginPhase's precision of 0.9497 and recall of 0.9147.

In addition to considering the two methods individually, we examine a combined set of variants which occur in both the calls made by WhatsHap on the PacBio reads and MarginPhase on the Nanopore data and where both tools report the same genotype. This improves the precision to 0.9969 at a recall of 0.7859. In further analysis, we refer to this combined variant set as *Long Read Variants*. It reflects a high precision subset of long read variants, validated independently by both sequencing technologies.

In order to further analyze the quality of the genotype predictions of our methods, we computed the genotype concordance of our callsets with respect to the GIAB ground truth inside of the high confidence regions. This was done by considering all variant positions correctly identified by MarginPhase and WhatsHap, and finding what fraction of these were also correctly genotyped (homozygous or heterozygous) with respect to the truth set. Figure 7 (bottom) shows the results. On the PacBio data, WhatsHap genotypes 99.78% of the variants contained in the truth set correctly, and MarginPhase genotypes 96.59% correctly. On the Nanopore data, MarginPhase performs slightly better by genotyping 98.02% of the SNVs contained in the GIAB callset correctly, while WhatsHap computed correct genotypes for 97.42% of the variants overlapping the GIAB truth set. Considering the intersection of the WhatsHap calls on PacBio, and MarginPhase calls on Nanopore data (i.e. our *Long Read Variants* set), we obtain a genotype concordance of 99.98%.

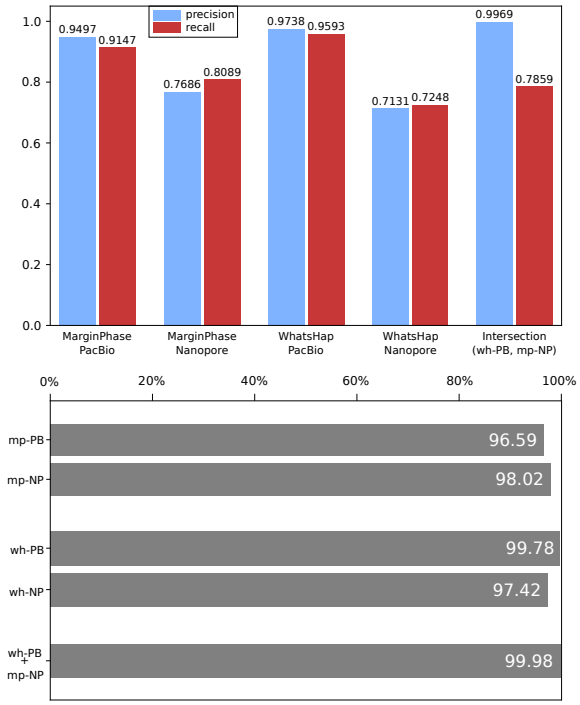

Fig. 7. ***Precision and Recall (Top)*** *of MarginPhase and WhatsHap on PacBio and Nanopore data sets in GIAB high confidence regions.* ***Genotype Concordance (Bottom)*** *(wrt. GIAB high confidence calls) of MarginPhase (mp, top) and WhatsHap (wh, middle) callsets on PacBio (PB) and Nanopore (NP) data. Furthermore, genotype concordance for the intersection of the calls made by WhatsHap on the PacBio and MarginPhase on the Nanopore reads is shown (bottom).*

bioRxiv preprint doi: https://doi.org/10.1101/293944; this version posted April 12, 2018. The copyright holder for this preprint (which was not certified by peer review) is the author/funder, who has granted bioRxiv a license to display the preprint in perpetuity. It is made available under aCC-BY-ND 4.0 International license.

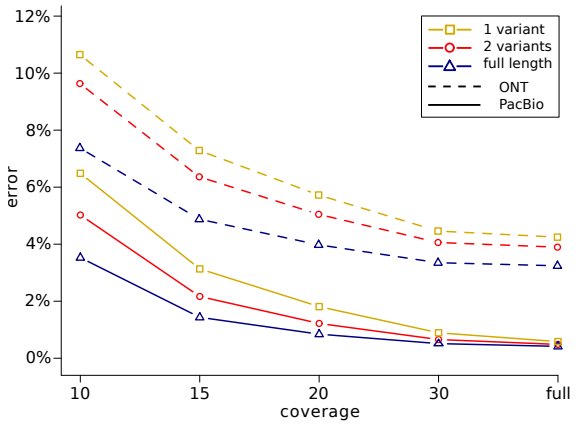

Fig. 8. ***Genotyping Errors*** *(wrt. to GIAB calls) as a function of coverage. The full length reads were used for genotyping (blue) and additionally, reads were cut such as to cover at most two variants (red) and one variant (yellow). Solid lines correspond to PacBio, dashed lines to Nanopore data.*

3.4 Cutting and Downsampling Reads

Our genotyping model incorporates haplotype information into the genotyping process by using the property that long sequencing reads can cover multiple variant positions. Therefore, one would expect the genotyping results to improve as the length of the provided sequencing reads increases. Furthermore, the coverage of the data would also affect the genotyping results.

In order to examine how the genotyping performance depends on the length of the sequencing reads and the coverage of the data, the following experiment was performed using the WhatsHap implementation. Both data sets (PacBio, Nanopore) were downsampled to average coverages $10\times, 20\times, 25\times$ and $30\times$. All SNVs inside of the high confidence regions in the GIAB truth set were re-genotyped from each of the resulting downsampled read sets, as well as from the full coverage data sets. Two versions of the genotyping algorithm were considered. First, the full length reads as given in the BAM files were provided to WhatsHap. Second, in an additional step prior to genotyping, the aligned sequencing reads were cut into shorter pieces such that each resulting fragment covered at most two variants. Additionally, we cut reads into fragments covering only one variant position. The genotyping performances of these genotyping procedures were finally compared by determining the amount of incorrectly genotyped variants.

Figure 8 shows the results of this experiment. On both data sets, the genotyping error increases as the length of reads decreases. Especially at lower coverages, the genotyping algorithm benefits from using the full length reads, which leads to much lower genotyping errors compared to using the shorter reads. In general, the experiment demonstrates that incorporating haplotype information gained from long reads does indeed improve the genotyping performance. Computing genotypes based on bipartitions of reads that represent possible haplotypes of the individual helps to reduce the number of genotyping errors, because it makes it easier to detect sequencing errors in the given reads.

3.5 Callset Consensus Analysis

In Figure 9, we further dissect the relation of our intersection call set (*Long Read Variants*, which refers to variants called by both WhatsHap on PacBio reads and MarginPhase on nanopore reads) to the GIAB truth set, as well as to the callsets from GATK/HC and FreeBayes, which both contributed to the GIAB truth set.

Figure 9a reveals that 399 156 variants in our *Long Read Variants* callset were called by both the GATK Haplotype Caller and FreeBayes, but are not in the GIAB truth set. To gather additional support for the quality of these calls, we consider two established quality metrics: the transition/transversion ratio (Ti/Tv), and the heterozygous/non-ref homozygous ratio (het/hom) (32). The Ti/Tv ratio of these variants is 2.10 and the het/hom ratio is 1.29. These ratios are comparable to those of the GIAB truth set, which are 2.10 and 1.55, respectively. An examination of the Platinum Genomes benchmark set (33), an alternative to GIAB, reveals 71371 such long-read validated variants outside of their existing truth set.

We hypothesized that a callset based on long reads is particularly valuable in regions that were previously difficult to characterize. To investigate this, we separately examined the intersections of our *Long Read Variants* callset with the two short-read callsets both inside the GIAB high confidence regions and outside of them, see Figure 9b and Figure 9c, respectively. These Venn diagrams clearly indicate that the concordance of GATK and FreeBayes was indeed substantially higher in high confidence regions than outside. An elevated false positive rate of the short-read callers outside the high confidence regions is a plausible explanation for this observation. Interestingly, the fraction of calls concordant between FreeBayes and GATK for which we gather additional support is considerably lower outside the high confidence regions. This is again compatible with an increased number of false positives in the short read callsets, but we emphasize that these statistics should be interpreted with care in the absence of a reliable truth set for these regions.

3.6 Candidate Novel Variants

To demonstrate that our method allows for variant calling on more regions of the genome than short read variant calling pipelines, we have identified 15 498 variants which lie outside of the *Short Read Mappable* area, but inside the *Long Read Callable* regions, i.e. regions in which there is sequencing depth of at least 10 and not more than $2\times$ the median depth for both sequencing technologies. We determined that 4.43 megabases of the genome (0.146%) is only mappable by long reads in this way.

Table 1 provides the counts of all variants found in each of the regions from Figure 6, as well as the counts for candidate variants, among the different types of genomic features described in Section 3.2. Over two thirds of the candidate variants occurred in the repetitive or duplicated regions described in the UCSC Genome Browser's repeatMasker track. The transition/transversion ratio of NA12878's 15 498 candidate variants is 1.64, and the heterozygous/homozygous ratio of these variants is 0.31. Given that we observe one candi-

bioRxiv preprint doi: https://doi.org/10.1101/293944; this version posted April 12, 2018. The copyright holder for this preprint (which was not certified by peer review) is the author/funder, who has granted bioRxiv a license to display the preprint in perpetuity. It is made available under aCC-BY-ND 4.0 International license.

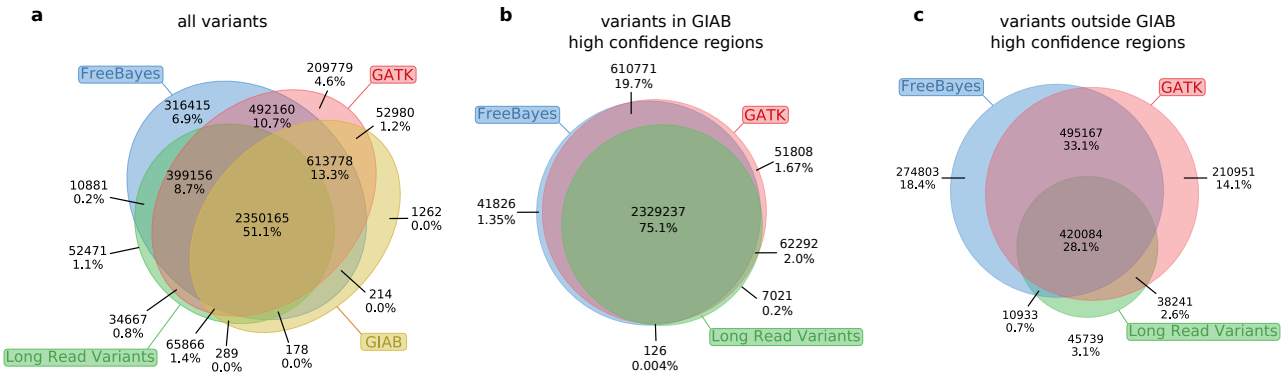

Fig. 9. ***Confirming Short Read Variants.*** *We examine all distinct variants found by our method, GIAB High Confidence, GATK/HC, and FreeBayes. Raw variant counts appear on top of each section, and the percentage of total variants is shown on bottom.*

date variant in every 325 haplotype bases, compared to one variant in every 1151 haplotype bases in the GIAB truth set, these candidate variants exhibit a $3.6\times$ increase in the haplotype variation rate.

3.7 Runtimes

Whole genome variant detection using WhatsHap took 147 CPU hours on PacBio reads and 79.5 hours on Nanopore, of which genotyping took 42.2 and 32.8 hours respectively. The MarginPhase implementation took 583 CPU hours on PacBio and 330 on Nanopore, with an additional 1730 and 1220 hours for realignment.

4 Discussion

We present a method that uses a Hidden Markov Model to partition long reads into haplotypes, which we found to improve the quality of variant calling. This is evidenced by our experiment in cutting and downsampling reads, where reducing the number of variants spanned by any given read leads to decreased performance at all levels of read coverage.

Our analysis of the method against a high confidence truth set in high confidence regions shows false discovery rates (corresponding to one minus precision) between 3 and 6 percent for PacBio, and between 24 and 29 percent for Nanopore. However, when considering a conservative set of variants confirmed by both long read technologies, the false discovery rate drops to around 0.3%, comparable with contemporary short read methods in these regions.

Table 1. *Distribution of candidate novel variants across different regions of interest. All variants refers to the variants in the* Long Read Variants *set, and Novel Variant Candidates are those described in Section 3.6.*

	All Variants	Novel Variant Candidates
Total	2,913,942	15,498
Gencode v27 (ALL)	1,363,064	5,594
Gencode v27 exome	86,357	538
Repeat Masker	1,583,684	10,677
LINEs	690,859	5,161
SINEs	421,340	1,432
Segmental Duplications	157,341	5,683
Tandem Repeats	96,871	5,437
Centromeres	18,644	2,031
Telomeres	295	14

In analyzing the area of the genome with high quality long read mappings, we found roughly a half a percent of the genome (approximately fifteen megabases) that is mappable by long reads but not by short reads. This includes one percent of the human exome, as well as over ten percent of segmental duplications. Even though some of these areas have low read counts in our experimental data, the fact that they have high quality mappings means that they should be accessible with sufficient sequencing. We note that this is not the case for centromeric regions, where Illumina reads were able to map over twice as much as we found in our PacBio data. This may be a result of the low quality in long reads preventing them from uniquely mapping to these areas with an appreciable level of certainty.

Over our entire set of called variants, the Ti/Tv and het/hom ratios were similar to those reported by the truth set. The Ti/Tv ratio of 2.18 is slightly above the 2.10 reported in the GIAB callset, and the Het/Hom ratio of 1.36 is lower than the 1.55 found in the GIAB variants. In the 15 498 novel variant candidates produced by our method in regions unmappable by short reads, the Ti/Tv ratio of 1.64 is slightly lower than that of the truth set. This is not unexpected as gene-poor regions such as these tend to have more transversions away from C:G pairs (34). We also observe that the Het/Hom ratio dropped to 0.31, which could be due to systematic biases in our callset or in the reference genome. The rate of variation in these regions was also notably different than in the high confidence regions, where we find three variants per thousand haplotype bases ($3.6\times$ the rate in high confidence regions). A previous study analyzing NA12878 (35) also found an elevated variation rate in regions where it is challenging to call variants, such as low complexity regions and segmental duplications. The study furthermore found clusters of variants in these regions, which we also observe.

The high precision of our intersected Nanopore/PacBio long read variants set makes it useful as strong evidence for confirming existing variant calls. As shown in the read coverage analysis, in both the GIAB and Platinum Genomes efforts many regions cannot be called with high confidence. In

bioRxiv preprint doi: https://doi.org/10.1101/293944; this version posted April 12, 2018. The copyright holder for this preprint (which was not certified by peer review) is the author/funder, who has granted bioRxiv a license to display the preprint in perpetuity. It is made available under aCC-BY-ND 4.0 International license.

the excluded regions of GIAB we found just under 400 thousand variants using both Nanopore and PacBio reads with our methods, which were additionally confirmed with Illumina reads by two other variant callers, FreeBayes and GATK/HC. Given the extensive support of these variants from multiple sequencing technologies and variant callers, these variants are good candidates for addition to the GIAB truth set. Expansion of benchmark sets to harder-to-genotype regions of the human genome is generally important for the development of more comprehensive genotyping methods, and we plan to work with these efforts to use our results. Further, our method is likely to prove useful for future combined diplotyping algorithms when both genotype and phasing is required, for example as may be used when constructing phased diploid *de novo* assemblies (36) or in future hybrid long/short read diplotyping approaches.

Acknowledgements

We thank the GIAB project for providing the data sets used. In particular, we thank Justin Zook for helpful discussions on how to use GIAB data, and Miten Jain for help with the nanopore data. This work was supported, in part, by the National Human Genome Research Institute of the National Institutes of Health under Award Number 5U54HG007990 and grants from the W.M. Keck foundation and the Simons Foundation.

1. Geraldine A Van der Auwera, Mauricio O Carneiro, Christopher Hartl, et al. From fastq data to high-confidence variant calls: the genome analysis toolkit best practices pipeline. *Current protocols in bioinformatics*, pages 11–10, 2013.
2. 1000 Genomes Consortium. A global reference for human genetic variation. *Nature*, 526 (7571):68–74, sep 2015. ISSN 0028-0836. doi: 10.1038/nature15393.
3. Wentian Li and Jan Freudenberg. Mappability and read length. *Frontiers in Genetics*, 5: 381, 2014. ISSN 1664-8021. doi: 10.3389/fgene.2014.00381.
4. Nicolas Altemose, Karen H. Miga, Mauro Maggioni, and Huntington F. Willard. Genomic Characterization of Large Heterochromatic Gaps in the Human Genome Assembly. *PLoS Computational Biology*, 10(5):e1003628, May 2014. ISSN 1553-7358. doi: 10.1371/journal.pcbi.1003628.
5. David Porubsky, Shilpa Garg, Ashley D Sanders, Jan O Korbel, Victor Guryev, Peter M Lansdorp, and Tobias Marschall. Dense and accurate whole-chromosome haplotyping of individual genomes. *Nat. Commun.*, 8(1):1293, November 2017.
6. Mark J P Chaisson, Ashley D Sanders, Xuefang Zhao, et al. Multi-platform discovery of haplotype-resolved structural variation in human genomes. *bioRxiv*, page 193144, September 2017.
7. Miten Jain, Sergey Koren, Karen H Miga, et al. Nanopore sequencing and assembly of a human genome with ultra-long reads. *Nature biotechnology*, 2018.
8. Fritz J Sedlazeck, Hayan Lee, et al. Piercing the dark matter: bioinformatics of long-range sequencing and mapping. *Nature Reviews Genetics*, page 1, 2018.
9. Fei Guo, Dan Wang, and Lusheng Wang. Progressive approach for SNP calling and haplotype assembly using single molecular sequencing data. *Bioinformatics*, February 2018.
10. Justin Zook, Jennifer McDaniel, Hemang Parikh, et al. Reproducible integration of multiple sequencing datasets to form high-confidence snp, indel, and reference calls for five human genome reference materials. *bioRxiv*, 2018. doi: 10.1101/281006.
11. Murray Patterson, Tobias Marschall, Nadia Pisanti, Leo van Iersel, Leen Stougie, Gunnar W Klau, and Alexander Schönhuth. WhatsHap: Weighted haplotype assembly for Future-Generation sequencing reads. *J. Comput. Biol.*, 22(6):498–509, June 2015.
12. Volodymyr Kuleshov, Dan Xie, Rui Chen, Dmitry Pushkarev, Zhihai Ma, Tim Blauwkamp, Michael Kertesz, and Michael Snyder. Whole-genome haplotyping using long reads and statistical methods. *Nat Biotechnol*, 32(3):261–266, March 2014. doi: 10.1038/nbt.2833.
13. Rudi Cilibrasi, Leo van Iersel, Steven Kelk, and John Tromp. The Complexity of the Single Individual SNP Haplotyping Problem. *Algorithmica*, 49(1):13–36, August 2007. doi: 10.1007/s00453-007-0029-z.
14. Harvey J Greenberg, William E Hart, and Giuseppe Lancia. Opportunities for Combinatorial Optimization in Computational Biology. *INFORMS Journal on Computing*, 16(3):211–231, August 2004. doi: 10.1287/ijoc.1040.0073.
15. T H Cantor and C R Cantor. Evolution of protein molecules. *Mammalian protein metabolism*, 1:22–123, 1969.
16. Heng Li, Bob Handsaker, Alec Wysoker, et al. The sequence alignment/map format and SAMtools. *Bioinformatics*, 25(16):2078–2079, 2009.
17. Marcel Martin, Murray Patterson, Shilpa Garg, Sarah Fischer, Nadia Pisanti, Gunnar W Klau, Alexander Schoenhuth, and Tobias Marschall. Whatshap: fast and accurate read-based phasing. *bioRxiv*, page 085050, 2016.
18. Sarah O Fischer and Tobias Marschall. Selecting reads for haplotype assembly. *bioRxiv*, page 046771, 2016.
19. Jayne Y. Hehir-Kwa, Tobias Marschall, et al. A high-quality human reference panel reveals the complexity and distribution of genomic structural variants. *Nature communications*, 7: 12989, 2016.
20. Jana Ebler, Alexander Schönhuth, and Tobias Marschall. Genotyping inversions and tandem duplications. *Bioinformatics*, 33(24):4015–4023, 2017.
21. Miten Jain, Ian T Fiddes, Karen H Miga, Hugh E Olsen, Benedict Paten, and Mark Akeson. Improved data analysis for the minion nanopore sequencer. *Nature methods*, 12(4):351, 2015.
22. Donna Karolchik, Angela S Hinrichs, Terrence S Furey, Krishna M Roskin, Charles W Sugnet, David Haussler, and W James Kent. The ucsc table browser data retrieval tool. *Nucleic acids research*, 32(suppl_1):D493–D496, 2004.
23. Jennifer Harrow, Adam Frankish, Jose M Gonzalez, et al. Gencode: the reference human genome annotation for the encode project. *Genome research*, 22(9):1760–1774, 2012.
24. Kate R Rosenbloom, Cricket A Sloan, Venkat S Malladi, et al. Encode data in the ucsc genome browser: year 5 update. *Nucleic acids research*, 41(D1):D56–D63, 2012.
25. AFA Smit, R Hubley, and P Green. Repeatmasker open-4.0. 2013-2015. *URL http://repeatmasker. org*, 2017.
26. Justin M Zook, David Catoe, , et al. Extensive sequencing of seven human genomes to characterize benchmark reference materials. *Scientific data*, 3:160025, 2016.
27. Heng Li. Minimap and miniasm: fast mapping and de novo assembly for noisy long sequences. *Bioinformatics*, 32(14):2103–2110, 2016.
28. Erik Garrison and Gabor Marth. Haplotype-based variant detection from short-read sequencing. *arXiv preprint arXiv:1207.3907*, 2012.
29. John G. Cleary, Ross Braithwaite, Kurt Gaastra, et al. Comparing variant call files for performance benchmarking of next-generation sequencing variant calling pipelines. *bioRxiv*, 2015. doi: 10.1101/023754.
30. Jonas Korlach. Perspective - Understanding Accuracy in SMRT Sequencing. 2013.
31. Christopher R O'Donnell, Hongyun Wang, and William B Dunbar. Error analysis of idealized nanopore sequencing. *Electrophoresis*, 34(15):2137–44, aug 2013. ISSN 1522-2683. doi: 10.1002/elps.201300174.
32. Jing Wang, Leon Raskin, David C Samuels, Yu Shyr, and Yan Guo. Genome measures used for quality control are dependent on gene function and ancestry. *Bioinformatics (Oxford, England)*, 31(3):318–23, feb 2015. ISSN 1367-4811. doi: 10.1093/bioinformatics/btu668.
33. Michael A Eberle, Epameinondas Fritzilas, Peter Krusche, et al. A reference data set of 5.4 million phased human variants validated by genetic inheritance from sequencing a three-generation 17-member pedigree. *Genome research*, 27(1):157–164, jan 2017. ISSN 1549-5469. doi: 10.1101/gr.210500.116.
34. Peter F Arndt, Terence Hwa, and Dmitri A Petrov. Substantial regional variation in substitution rates in the human genome: importance of GC content, gene density and telomere-specific effects. 2005.
35. Neil I Weisenfeld, Shuangye Yin, Ted Sharpe, Bayo Lau, et al. Comprehensive variation discovery in single human genomes. *Nature genetics*, 46(12):1350, 2014.
36. Chen-Shan Chin, Paul Peluso, Fritz J Sedlazeck, et al. Phased diploid genome assembly with single-molecule real-time sequencing. *Nature methods*, 13(12):1050, 2016.