Harnessing methylation signals inherent in long-read sequencing data for improved variant phasing

Aaron Pfennig[1] and Joshua M. Akey[1,*]

[1] Lewis-Sigler Institute for Integrative Genomics, Princeton University, Princeton, NJ 08544, USA

[*] Corresponding author: jakey@princeton.edu

Abstract

Accurate phasing of genetic and epigenetic variation is crucial for many downstream analyses, including association testing, clinical variant interpretation, and inference of population history. Although long-read sequencing significantly improves the continuity and completeness of genome sequencing, reconstructing chromosome-scale haplotypes remains challenging, often requiring the integration of multiple technologies, such as PacBio HiFi and Oxford Nanopore Technologies (ONT) sequencing. While these sequencing platforms detect the epigenetic modification 5-methylcytosine (5mC), current read-based phasing algorithms do not incorporate this information. We developed a read-based phasing method named LongHap that seamlessly integrates sequence and methylation data and shows that it significantly improves haplotype reconstruction. LongHap first creates phase blocks based on overlapping heterozygous sequence variants, accurately phasing complex variants by embedding them into the broader haplotype context through belief propagation. LongHap then dynamically identifies differentially methylated sites that are informative for phasing to refine and extend initial phase blocks. Through extensive analyses, we demonstrate that LongHap outperforms existing tools, including WhatsHap, HapCUT2, LongPhase, and MethPhaser, by achieving lower switch error rates and greater phase block contiguity. Crucially, we show that LongHap also improves variant phasing in challenging, medically relevant genes. In summary, by leveraging native methylation signals from long-read sequencing data, LongHap enhances long-range haplotype reconstruction, enabling more accurate haplotype-based genome analysis. LongHap is available from: https://github.com/AkeyLab/LongHap.

Introduction

Accurate variant phasing is a prerequisite for powerful haplotype-based analyses in population and statistical genetics, such as inferring recent relatedness and fine-mapping causal variants[1]. Commonly used approaches for phasing include statistical, pedigree (trio), and read-based phasing[2]. Statistical and trio phasing can infer haplotypes on a chromosome-scale, using large reference panels and parental sequencing data, respectively[3,4]. Read-based phasing, in contrast, directly analyzes the raw sequencing data to identify alleles that co-occur across heterozygous sites in individual reads[5–9]. Compared to statistical phasing, read-based phasing avoids reference panel biases and is better suited to phase rare variants without requiring parental sequencing data[10]. However, the ability of read-based approaches to phase variants over long distances has historically been limited by short read lengths.

The improving accuracy and declining costs have made long-read sequencing with PacBio HiFi and Oxford Nanopore Technologies (ONT) viable alternatives to short-read sequencing. The longer read lengths of these technologies dramatically improve the ability of read-based phasing approaches, such as WhatsHap, HapCUT2, and HaploMaker, to infer accurate haplotypes over longer genomic segments[5–8]. A recent method, called LongPhase, was designed specifically for longer read lengths and can achieve chromosome-scale phasing with ultra-long ONT (UL-ONT) data while providing improvements in phasing contiguity and computational efficiency compared with WhatsHap and HapCUT2[9]. However, LongPhase relies on heuristic graph traversal and efficiency-driven simplifications and does not explicitly model alignment uncertainty or complex variant structure, which can increase sensitivity to sequencing, alignment, and variant calling artifacts. In addition to DNA sequence variation, variation in 5-methylcytosine (5mC) can, in principle, also be leveraged for phasing. To date, only one method, MethPhaser[11], has been

51 developed to harness the 5mC data that is contained in long reads. However, MethPhaser can only
52 be applied *post hoc* to the output of another read-based phasing tool to refine haplotypes. Thus, no
53 read-based variant phasing method exists that integrates sequence and epigenetic information in a
54 unified framework.

55 Here, we developed LongHap, a read-based variant phasing method that seamlessly integrates
56 sequence and 5mC methylation signals from PacBio HiFi and ONT sequencing data. LongHap
57 applies a multi-stage approach, where phase blocks are initially constructed from sequence
58 variation, complex variants are embedded into the larger haplotype context by belief propagation,
59 and then initial phase blocks are refined based on methylation information. By integrating
60 methylation data in a principled way, LongHap outperforms all other tools on genome-wide
61 benchmarks, achieving up to 5% lower switch error rates and 30% larger phase block N50s with
62 PacBio HiFi data. These improvements extend to all variant classes, including complex structural
63 variants and functional genomic elements. In addition to these genome-wide performance gains,
64 LongHap provides locus-specific improvements by leveraging methylation information, phasing
65 of challenging, medically relevant genes that cannot be phased based on sequence information
66 alone, such as *LIX1*, which is a highly conserved protein that regulates cell fate decisions[12–15].
67 These results demonstrate the power of exploiting more of the information present in modern
68 sequencing data rather than simply extending short-read paradigms to long-read data. The
69 substantial improvements in phasing accuracy and contiguity afforded by LongHap will facilitate
70 inferences into the biology and evolution of complex eukaryotic genomes.

71

72 # Materials & Methods

73 ## LongHap

74 *Input for LongHap*

75 As a minimal input, LongHap requires aligned sequencing reads in BAM format and variant calls
76 in VCF format. To incorporate epigenetic information from long-read sequencing data, LongHap
77 can optionally use site-specific methylation calls, generated, for example, with pb-CpG-tools and
78 provided as unphased methylation annotations.

79 *Determining allelic support in individual reads*

80 Using pysam v0.22.1 (https://github.com/pysam-developers/pysam) and cyvcf2 v0.31.1[16],
81 LongHap streams through the BAM file and infers the allele each read supports at individual
82 overlapping heterozygous sites in the VCF. Alignments with a mapping quality less than a
83 specified threshold (by default, 20) are ignored along with secondary alignments, duplicate reads,
84 and reads that are flagged as quality control failures. Unlike other read-based phasing tools,
85 LongHap can include "multiallelic" heterozygous sites, where both haplotypes are different from
86 the reference. On an individual genome level, these sites are still biallelic and simply require a
87 dynamic indexing for accurate phasing. However, for comparability purposes, all of the results
88 described below are based on biallelic sites only.

89 To optimize runtime, LongHap first attempts to infer allelic states based on CIGAR strings and
90 matching subsequences, resorting to local realignments only if the allele supported by a read
91 cannot be confidently identified. For single-nucleotide variants (SNVs), LongHap requires an
92 exact match of bases when using highly accurate PacBio Revio reads, and for noisier ONT reads,

93 a matching base upstream and downstream of the heterozygous SNV is required in addition to a
94 supporting CIGAR string. For insertions of any size, LongHap requires that the CIGAR string
95 supports either no insertion/deletion (INDEL) or structural variant (SV) at all, or an insertion of
96 the exact length as expected based on the VCF record, along with a matching inserted sequence.
97 Similarly, for deletions, the CIGAR string must not support an INDEL, an SV of any size, or a
98 deletion with the exact length specified in the VCF record. If the allelic state of a read at a given
99 heterozygous site cannot be unambiguously identified according to the above criteria, the read-site
100 pair is left for local realignment.

101 If a read-site pair is left for realignment, the read is locally re-aligned to synthetic references
102 representing allele A and B, respectively. Synthetic references are created by embedding both
103 alleles in the surrounding reference sequence. For PacBio, SNVs, INDELs, and SVs are embedded
104 in 33 bps, 100 bps, and 100 bps, respectively. For noisier ONT data, SNVs, INDELs, and SVs are
105 embedded in 66 bps, 200 bps, and 200 bps, respectively. However, the embedding size can also be
106 defined from the command line for specific use cases. Gap open and extension penalties are
107 dynamically set based on sequencing technology, variant type, and sequence context. Using the
108 CIGAR string, we calculate the INDEL rate for each read. That is, we count the number of bases
109 implicated in an insertion or deletion operation and divide it by the total number of CIGAR
110 operations. For PacBio HiFi sequencing, the indel rate is typically less than 0.005, and we set the
111 gap open and extension penalty to seven and two, respectively. For ONT sequencing, the indel rate
112 can be higher depending on the kit used. For reads with indel rates, r, satisfying $0.005 \leq r < 0.02$,
113 we set the gap open and extension penalties to 5 and 1, respectively. For reads with $0.02 < r \leq 0.05$,
114 we set the gap open penalty to 4, and for reads with $r > 0.05$, we set the gap open penalty to 3,
115 while keeping the gap extension penalty set to 1. Independent of the sequencing technology, we

then double the gap open penalty for SNVs. Similarly, because both PacBio and ONT sequencing have higher error rates in homopolymeric sequences, we reduce the gap open penalty by 2 if a heterozygous site falls within a homopolymer, which we define as at least two identical bases surrounding the heterozygous site in both directions. Finally, we realign the read against both synthetic reference sequences, using a semiglobal alignment approach implemented in parasail v1.3.4 (*sg_stats_striped_sat*)[17]. We then assume that a read supports the allele yielding the greater alignment score.

Construction of directed-acyclic graph

After determining the allelic states of reads at overlapping heterozygous sites, we construct a directed acyclic graph (DAG) $\mathcal{G} = (V, E, \omega)$. V, the set of vertices, is defined by the heterozygous sites in the VCF file, including INDELs and SVs. For each heterozygous site, we create a layer with two vertices, representing two alleles (A and B), such that:

$$V = \{(v_A^0, v_B^0), (v_A^1, v_B^1), \dots, (v_A^{L-2}, v_B^{L-2}), (v_A^{L-1}, v_B^{L-1})\},$$

where L is the number of heterozygous sites in the VCF (Figure 1A). E, the set of edges, comprises all possible edges between vertices of adjacent layers in $\mathcal{G}$, that is:

$$E = \{(v^i\ ,\ v^{i+1}) \mid v^i\ \in\ \{v_A^i, v_B^i\},\ v^{i+1} \in \{v_A^{i+1}, v_B^{i+1}\}\ \forall\ i = 0, \dots, L-2\}$$

where (v_A^i, v_B^{i+1}) represents the edge connecting allele A and B at site i and $i+1$, respectively.

Edge weights, ω, are calculated by tallying the co-occurrence of alleles at two adjacent heterozygous sites i and $i+1$, calculating the frequency with which allele pairs (A_i, A_{i+1}), (A_i, B_{i+1}) etc., co-occur at sites i and $i+1$. The weights of edges connecting layers i and $i+1$ $(\omega(v^i\ , v^{i+1}))$ are thus defined by:

$$\omega(v^i, v^{i+1}) = \begin{bmatrix} \omega(v_A^i, v_A^{i+1}) & \omega(v_A^i, v_B^{i+1}) \\ \omega(v_B^i, v_A^{i+1}) & \omega(v_B^i, v_B^{i+1}) \end{bmatrix}$$

$$= \begin{bmatrix} \frac{\sum A_i \rightarrow A_{i+1}}{\sum A_i} & \frac{\sum A_i \rightarrow B_{i+1}}{\sum A_i} \\ \frac{\sum B_i \rightarrow A_{i+1}}{\sum B_i} & \frac{\sum B_i \rightarrow B_{i+1}}{\sum B_i} \end{bmatrix}$$

$$= \begin{bmatrix} P(A_j|A_i) & P(B_j|A_i) \\ P(A_j|B_i) & P(B_j|B_i) \end{bmatrix}$$

Finally, LongHap mirrors the more confident transition, creating a symmetrical $2x2$ matrix for each pair of adjacent heterozygous sites.

When analyzing ONT data, we require that each allele is observed on both DNA strands since it was observed that variants supported by reads from only one DNA strand are more frequently false-positive calls[9]. Variants that are only observed on reads from one strand are marked as not phaseable and are excluded from subsequent analyses.

Re-estimating transitions for difficult-to-phase variants using loopy belief propagation

To improve haplotype reconstruction of difficult-to-phase variants, that is, INDELs, SVs, and variants for which at least one allele was observed less than AN times ($AN < 1$ by default), LongHap embeds them into the broader haplotype context by considering long-range edges between non-adjacent heterozygous sites. Note that alleles observed zero times ($AN = 0$) can infrequently occur as the result of alignment and variant calling errors or difficulty in determining the allelic state of individual reads by LongHap. To re-phase m consecutive difficult-to-phase variants $(i, \dots, i+m)$, a subgraph $\mathcal{G}_S = (V_S, E_S, \omega_S)$ is constructed with the vertices $V_S = \{i-n, \dots, i, \dots, i+m, \dots, i+m+n\}$, where n is the number of surrounding variants that can be

confidently phased, and $E = \{(v^j, v^{j+1}) | j \in i - n, \dots i + m + n\}$. In practice, we choose $n = 2$. Each vertex can take one of two states $x = \{A, B\}$. For each pair of adjacent vertices, we already have a $2x2$ transition matrix $\omega(v^i, v^{i+1})$, indicating probabilities with that vertex i transitions from its current state into either state at vertex $i + 1$. In addition to these edges between adjacent pairs, long-range edges are considered, connecting the preceding and succeeding confidently phased vertices with all other vertices in the graph that are more than one step away, that is, $\omega(v^j, v^k) \mid j \in \{i - n, \dots, i - 1\} \forall k \in \{j + 2, \dots, j + m + n + 1\}$ and $\omega(v^j, v^k) \mid j \in \{i, \dots, i + m - 1\} \forall k \in \{j + 2, \dots, j + m + n - 1\}$. For example, if $m = 1$ and $n=2$, long-range transitions are inferred between vertex $i - 2$ and vertices $i \dots i + 2$, vertex $i - 1$ and vertices $i + 1$ and $i + 2$, and vertex i and vertex $i + 2$ (Figure 1B).

Belief Propagation (BP) provides an efficient means to infer marginal distributions of states at individual vertices by factorizing the graph and propagating previously calculated messages between vertices[18,19]. In acyclic graphs, BP yields an exact solution, while in graphs with cycles, only an approximation can be obtained. Although the above-described directed subgraph has no cycles *a priori*, it does have cycles in its moralized (undirected) form. We thus must approximate marginal distributions at individual vertices, using loopy BP[19,20].

Given the above-described moralized subgraph (Figure 1B), LongHap determines the neighbors for each vertex based on whether they are connected by an edge. Furthermore, states for each vertex j (ϕ_j) are initialized uniformly, except for the first vertex ($i - n$), which is initialized to be in state A. The message a node j sends to a neighbor j ($m_{j \to k}(x_k)$) depends on its state ϕ_j, the messages it receives from its other neighbors, and the local factor connecting j and k ($\psi_{j,k}(x_j, x_k) = \omega(v^j, v^k)$):

$$m_{j\to k}(x_k) = \sum_{x_j} \phi_j(x_j)\psi_{j,k}(x_j, x_k) \prod_{l\in N(j)\setminus\{k\}} m_{l\to j}(x_j)$$

The loopy BP is started at an arbitrary pair of vertices j and k, and an updated $\widehat{m_{j\to k}(x_k)}$ is calculated. Based on $\widehat{m_{j\to k}(x_k)}$, all other messages in the graph are updated, and the algorithm is run iteratively until it converges or reaches a maximum number of iterations (by default, 500). Although convergence is not guaranteed, we found that it virtually always converges within a few iterations. Given the messages received by a node j, its normalized marginal belief ($b_j(x_j)$) is given by:

$$b_j(x_j) = \frac{\phi_j(x_j)\prod_{k\in N(j)} m_{k\to j}(x_j)}{\sum_{x_j}\phi_j(x_j)\prod_{k\in N(j)} m_{k\to j}(x_j)}$$

Similarly, unnormalized pairwise marginal beliefs of vertices j and k are given by:

$$b_{jk}(x_j, x_k) = \phi_j(x_j)\phi_k(x_k)\psi_{j,k}(x_j, x_k)\left(\prod_{l\in N(j)\setminus\{k\}} m_{l\to j}(x_j)\right)\left(\prod_{m\in N(k)\setminus\{j\}} m_{m\to k}(k)\right)$$

$b_{jk}(x_j, x_k)$ can be interpreted as the transition probability between j and k. However, we are only interested in $\omega(\widehat{v^j, v^{j+1}})$ of adjacent heterozygous sites in the graph that is stripped of the long-range connections but implicitly still contains their information, that is, the information of long-range connections is projected in pairwise adjacent connections. $\omega(\widehat{v^j, v^{j+1}})$ is thus obtained by solving:

$$b_{j+1}(x_{j+1}) = \sum_{x_j} b_j(x_j)\,\omega(\widehat{v^j, v^{j+1}})$$

yielding:

$$\omega\left(\widehat{v^{j}, v^{j+1}}\right) = \frac{b_j(x_j)+b_{j+1}(x_{j+1})-1}{2b_j(x_j)-1}$$

$\omega\left(\widehat{v^{j}, v^{j+1}}\right)$ is consistent with the full DAG described above. In the example graph shown in Figure 1B, LongHap aims to calculate $\omega\left(\widehat{v^{i-1}, v^{i}}\right)$ and $\omega\left(\widehat{v^{i}, v^{i+1}}\right)$ as described above.

Initial attempt to resolve ambiguous transitions

Although long-read sequencing allows phasing of most pairs of adjacent heterozygous sites, not all transitions can be unambiguously resolved. For some pairs i and $i+1$, all transitions between allelic states may have equal probability, causing a break in the phase block (see below). In such cases, LongHap attempts to extend the phase block by testing whether the transitions from i to $i+2$ can be confidently inferred. If this is the case, LongHap tests if it can use this long-range edge to update the posterior beliefs for the transitions from i to $i+1$ and i to $i+2$, or marks the heterozygous variant at site $i+1$ as not phaseable and connects i to $i+2$.

Leveraging methylation information to resolve remaining ambiguous transitions

After constructing the initial DAG based on sequence evidence, LongHap leverages methylation information inherent to PacBio and ONT long-read sequencing data to resolve remaining ambiguous transitions. These include pairs of adjacent heterozygous sites i and $i+1$ that are not overlapped by any read, making it impossible to phase them solely based on sequence evidence.

To leverage methylation calls, LongHap first identifies putatively differentially methylated sites, that is, sites at which 20-80% of the reads are methylated. A haplotype is considered methylated if the product of methylation probabilities across all reads assigned to the haplotype is greater than 0.5, and its log10 odds ratio of being methylated versus unmethylated is greater than three, and

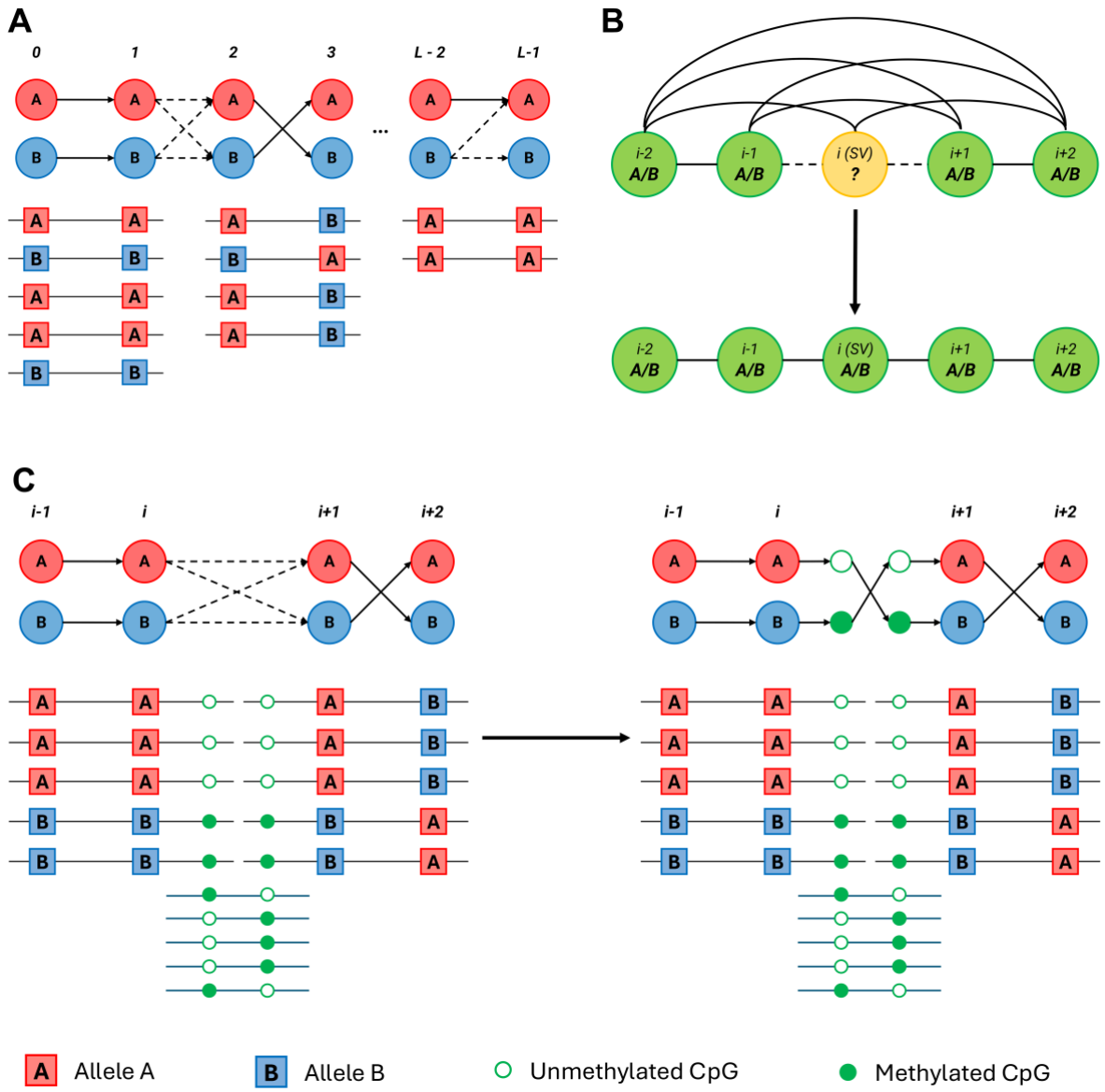

Figure 1. Overview of LongHap's core steps. A) Schematic for constructing the directed acyclic graph based on co-occurrences of heterozygous variants on individual sequencing reads. **B)** Illustration of the moralized embedding of a difficult-to-phase variant i, such as an SV, into the broader haplotype context. Transitions into and out of i are re-inferred using long-range connections between non-adjacent variants. Layers in the original DAG (**A**) are collapsed into single nodes with two allele states (A and B). **C)** Differentially methylated sites are identified on-the-fly and used to assign additional reads to the maternal and paternal haplotypes. This procedure is repeated until no more reads can be assigned to either haplotype, and the transition between two previously ambiguous layers has been resolved.

unmethylated otherwise. Finally, a site is considered differentially methylated and informative for phasing if the inferred methylation states differ between the two haplotypes. LongHap uses these differentially methylated sites as additional markers to connect to phase blocks.

LongHap anchors reads based on their allelic states at sites $i-1$ and i and then tests if parental haplotypes are differentially methylated at any candidate sites overlapped by these reads as described above. It then uses the differentially methylated sites to iteratively assign reads to either haplotype and identify additional differentially methylated sites based on the newly assigned reads. This procedure is repeated until either the gap between sites i and $i+1$ is bridged or no more reads can be assigned to either haplotype (Figure 1C). If the gap was successfully bridged, LongHap uses all reads assigned to a haplotype to infer the transition probabilities between sites i and $i+1$ ($\omega\left(v^{i}, v^{i+1}\right)$).

Viterbi-like inference of most likely haplotype

After completing the DAG, LongHap employs a Viterbi-like decoding scheme to identify the most likely haplotypes. That is, there are just transition probabilities and no emission probabilities in our model. At the beginning of the forward pass, we assign equal probabilities of being in state A and B at the first layer. LongHap then calculates the probability of being in either state at each subsequent layer of the DAG, keeping track of the more likely state. Whenever it encounters an ambiguous transition between two layers, that is, all transitions are equally likely, it stops the forward pass, randomly initializes a new phase block, and backtraces the previous phase block.

Output

LongHap writes a phased VCF, providing a PS tag for each phased variant. Optionally, LongHap also provides phase block coordinates, haplotagged read alignments in BAM format (HP tag), and information on the methylation sites used for phasing.

Sequencing data

We retrieved multiple PacBio and ONT sequencing datasets for HG002 for benchmarking purposes from the T2T consortium[21]. Specifically, we downloaded PacBio Revio HiFi reads, ONT Simplex R10.4.1 reads base-called with Dorado (PAW70337), and UL-ONT Duplex R10.4.1 reads base-called with Dorado. Table 1 summarizes the characteristics of the sequencing datasets used and provides links to each dataset.

Table 1: Sequencing datasets used in this study and their characteristics.

Sequencing dataset	Coverage	Read length N50 (kb)	Base quality (Q)	URL
PacBio Revio HiFi	~38x	18	>30	https://human-pangenomics.s3.amazonaws.com/submissions/80d00e88-7a92-46d8-88c7-48f1486e11ed--HG002_PACBIO_REVIO/m84039_230117_233243_s1.hifi_reads.default.bam
ONT Simplex R10.4.1, Dorado	~45x	29	~26	s3://ont-open-data/giab_2025.01/basecalling/sup/HG002/PAW70337/calls.sorted.bam
UL-ONT Duplex R10.4.1, Dorado	~44x	111	N/A	https://s3-us-west-2.amazonaws.com/human-pangenomics/index.html?prefix=T2T/scratch/HG002/sequencing/ont/11_16_22_R1041_UL_HG002_dorado0.4.0/

246 **Sequence alignment, variant calling, and methylation calling**

247 Sequences were aligned to CHM13v2.0 using minimap2 v2.28-r1221-dirty with *--map-hifi* and --
248 *map-ont* presets for PacBio Revio HiFi and ONT sequencing data, respectively[22,23]. We then used
249 PacBio Revio HiFi sequence alignments to call SNVs and small INDELs with DeepVariant v1.9.0
250 and larger structural variants (SVs) with Sniffles2 v2.7.1[24,25]. We removed sites with a genotype
251 quality less than 20 (GQ < 20) and merged DeepVariant and Sniffles2 variant calls. Finally, we
252 left-aligned and merged multiallelic variants using *bcftools norm -m+* [26]. We determined
253 methylation states using *aligned_bam_to_cpg_scores* v3.0.0 from pb-CpG-tools
254 (https://github.com/PacificBiosciences/pb-CpG-tools).

255 **Benchmarking of read-based variant phasing methods**

256 We compared the performance of LongHap to WhatsHap v2.3[5,7], HapCUT2 v1.3.4[6], LongPhase[9],
257 and WhatsHap + MethPhaser v0.0.3[11]. For each tool, we calculated phasing statistics using
258 *whatshap stats* and switch error rates relative to the T2T HG002 Q100 variant call set using
259 *whatshap compare*[5,7,21]. Prior to evaluation, variant calls in the T2T HG002 Q100 ground truth call
260 set and phased variant calls were left-aligned, and multiallelic variants were split using *bcftools*
261 *norm -m -any*[26]. As these tools phased variable percentages of heterozygous sites, we calculated
262 switch error rates based on sites phased by all tools to allow a fair comparison.

263 We also evaluated the ability of each tool to contiguously phase a recently curated list of 273
264 challenging, medically relevant genes (CMRGs) made available by the Genome in a Bottle
265 consortium by intersecting their gene bodies with phase blocks using bedtools[27,28]. Although some
266 tools support parallelization, we used single CPU threads for all cores to enable fair comparisons
267 of computational requirements.

Results

Overview of LongHap

LongHap is a read-based variant phasing method that integrates methylation signals native to long-read sequencing data, such as PacBio Revio HiFi and ONT sequencing, into a unified framework. As input, LongHap requires variant calls in VCF format, aligned sequencing reads in BAM format, and, optionally, unphased methylation calls. LongHap then uses a stepwise approach to co-phase SNVs, INDELs, and SVs. First, LongHap phases pairs of variants based on sequence information, embedding complex and low-support variants into the broader haplotype context using graph theory, that is, loopy belief propagation. In a second step, LongHap identifies differentially methylated sites on the fly and leverages them as additional, phase-informative markers to resolve variant pairs that could not be confidently phased based on sequence information alone, extending initially inferred phase blocks. Finally, LongHap outputs a phased VCF and, optionally, haplotagged read alignments, phase block coordinates, and the set of differentially methylated sites used for phasing.

LongHap outperforms all other tools by leveraging methylation information

We evaluated LongHap's performance using PacBio Revio HiFi and various ONT long-read sequencing datasets for HG002 that were made available by the T2T consortium (Table 1)[21]. For all autosomes, we calculated the switch error rates, phase block length N50s, and the percentage of phased heterozygous sites for LongHap, WhatsHap, HapCUT2, LongPhase, and WhatsHap + MethPhaser using WhatsHap compare and stats[5–7,9,11]. To increase comparability and highlight the benefits of integrating methylation information, we compare LongHap's performance to the other

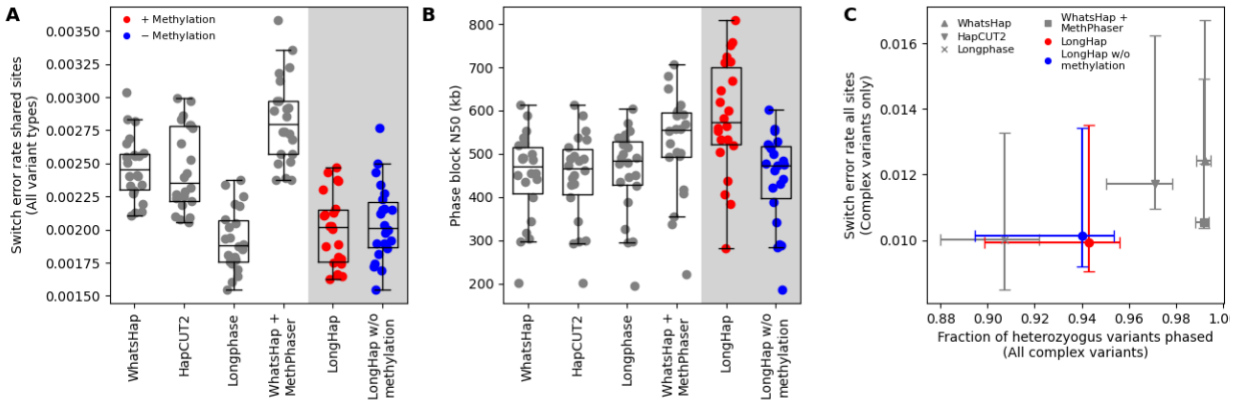

Figure 2. Leveraging methylation information reduces switch error rates, including for INDELs and SVs, and increases phasing contiguity. **A)** Switch error rates for different read-based variant phasing tools using 38x PacBio Revio HiFi data for individual chromosomes of the well-characterized genome of HG002. LongHap makes 164 fewer errors when considering methylation information (4,860 vs. 5,024), and 2,117 fewer errors than WhatsHap + MethPhaser (6,977), another tool that integrates methylation information. LongPhase makes the fewest errors (4,754) but phases the fewest sites of all tools. Switch error rates were calculated based on the intersection of sites phased by all tools (96.13% of all heterozygous sites). **B)** Simultaneously, LongHap's integration of methylation information significantly increases the phase block N50 (mean N50: 584 kb compared to 443 kb), a greater gain than achieved by WhatsHap + MethPhaser (mean N50: 523 kb compared to 450 kb for WhatsHap alone). **C)** LongHap phases more INDELs and SVs than LongPhase, while maintaining a lower error rate than all other tools, through the integration of methylation information and its rigorous embedding of INDELs and SVs into the broader haplotype context. Switch error rates were calculated based on the intersection of INDEL and SV sites phased by all tools (84.80% of all heterozygous sites with complex variants).

tools without and with methylation information. No tool phases all heterozygous sites, and we thus limit the comparisons to the intersection of biallelic sites that were phased by all tools.

LongHap and LongPhase outperform all other tools in terms of switch error rate. Using PacBio Revio HiFi data, LongHap achieves a switch error rate of 0.203% and 0.196% without and with methylation information, respectively. In comparison, LongPhase, HapCUT2, WhatsHap, and WhatsHap + MethPhaser achieve switch error rates of 0.192%, 0.243%, 0.246%, and 0.281%, respectively. In absolute terms, these improvements translate to LongHap making 1,081 and 1,245 fewer switch errors than WhatsHap without and with methylation information, respectively. Unlike MethPhaser, LongHap's integration of methylation information leads to a 3.3% decrease in switch errors (4,860 vs. 5,024 errors) (Figure 2A). While LongHap and LongPhase have similar switch

error rates, LongHap phases a greater fraction of heterozygous sites than LongPhase. LongHap phases, on average, 99.08% and 99.13% of the heterozygous sites without and with methylation information compared to 98.42%, 99.89%, 99.55%, and 99.83% for LongPhase, WhatsHap, HapCUT2, and WhatsHap + MethPhaser (Figure S1). As discussed below, this trend is exacerbated for noisier ONT sequencing data.

All tools achieve similar phase block N50s without methylation data. The average phase block N50 across all autosomes is 443 kb for LongHap when not considering methylation information. When integrating methylation data, LongHap's average phase block N50 increases by 31.9% to 584 kb, outperforming all other tools, including WhatsHap + MethPhaser, which only achieves an increase of 16.2% from 450 kb for WhatsHap alone to 523 kb (Figure 2B).

LongHap accurately co-phases complex structural variants

LongHap co-phases INDELs and complex SVs along with SNVs, using local realignment to account for sequencing errors and inaccurate alignments around SVs breakpoints and loopy belief propagation to efficiently embed these variants into the broader haplotype context. While WhatsHap and HapCUT2 are primarily designed for SNV phasing and are based on formulations of the minimum error correction problem that embeds all variants into the broader haplotype context[5–7], LongPhase co-phases indels and structural variants by treating them similarly to SNVs[9]. LongPhase's approach is highly efficient but does not explicitly model alignment uncertainty or complex variant structure[9].

To assess the benefits of LongHap's comprehensive approach to INDEL and SV phasing, we separately evaluated the accuracy of each tool, excluding SNVs. All tools have higher switch error rates for complex variants (on average, 0.99% - 1.24%), with LongHap and LongPhase having

321 lower error rates than WhatsHap and HapCUT2. With PacBio Revio HiFi data, LongHap's error
322 rate for complex variants without integrating methylation information is slightly higher than
323 LongPhase's error rate (on average, 1.01% compared to 1.00%) but lower than WhatsHap +
324 MethPhaser's error rate (on average, 1.05%). When integrating methylation information, LongHap
325 achieves the lowest switch error rate of all tools (on average, 0.99%), while also phasing more
326 heterozygous variants than LongPhase (on average, 94.29% compared to 90.73%) (Figure 2C).
327 Again, as discussed in the following section, the tradeoff between phasing as many variants as
328 possible while maintaining a low switch error rate becomes starker for ONT sequencing data.

329 **LongHap capitalizes on longer read lengths of ONT sequencing data**

330 The performances of the different phasing algorithms are more heterogeneous when using ONT
331 sequencing data. When using ONT sequencing data, LongHap achieves a lower switch error rate
332 than WhatsHap, HapCUT2, and WhatsHap + MethPhaser (0.110% without and with methylation
333 information compared to 0.129%, 0.135%, and 0.141%, respectively) but has a slightly higher
334 error rate than LongPhase (0.081%) (Figure S2). However, LongPhase achieves a superior switch
335 error rate at the expense of phasing significantly fewer sites, phasing only 93.81% - 96.42%
336 compared to 98.35%-99.41% and 98.37% - 99.43% for LongHap without and with methylation
337 information (Figure S3). Thus, there is a tradeoff between maximizing the fraction of phased
338 heterozygous sites and minimizing the switch error. LongHap optimizes this tradeoff, phasing
339 more variants than LongPhase but fewer than WhatsHap or HapCUT2, while maintaining an
340 intermediate switch error rate. The importance of this tradeoff seems to be increasing with
341 increasing read length, as the contrasts are becoming more pronounced for ultra-long ONT (UL-
342 ONT) sequencing (Figures S4 & S5) and for INDELs and SVs. LongPhase phases, on average,
343 only 74.93% and 70.74% of all heterozygous INDELs and SVs, while LongHap phases 94.85%

344 and 95.04% without and with methylation information when using ONT and UL-ONT sequencing
345 data, respectively (Figures S2C & S4C). Importantly, integrating methylation information from
346 ONT sequencing yields similar improvements as observed when using PacBio Revio HiFi data,
347 showing that LongHap can leverage various modern long-read sequencing technologies to improve
348 variant phasing.

349 We also note that LongHap successfully capitalizes on the greater read length in ONT and UL-
350 ONT sequencing data, achieving chromosome-scale phasing (average phase block N50s: 1.97 Mb
351 and 80.7 Mb, respectively). In contrast, WhatsHap fails to translate the longer read lengths into
352 meaningfully longer phase blocks (average phase block N50s: 1.47 Mb and 5.82 Mb), and
353 MethPhaser cannot compensate for WhatsHap's poor phase block N50 (average phase block N50s:
354 1.69 Mb and 7.22 Mb) (Figures S2B & S4B).

355 **Gap-bridging properties of phase-informative methylation signals**

356 Leveraging methylation information, LongHap connects 1,636 out of 9,005 phase blocks inferred
357 based on sequence information (18.2%) using PacBio Revio HiFi data. The average gap between
358 two connected blocks is 32 kb (median: 29 kb; maximum: 233 kb; standard deviation: 12 kb), and
359 LongHap leverages, on average, 18.8 differentially methylated sites (median: 12; SD: 20.2) to
360 connect two phase blocks. With the longer read length of ONT sequencing data, LongHap can
361 bridge larger gaps between two phase blocks. For ONT and UL-ONT sequencing data. The average
362 distances between two phase blocks connected with differentially methylated sites are 56 kb
363 (median: 51 kb; maximum: 325 kb; standard deviation: 24 kb) and 415 kb (median: 239 kb;
364 maximum: 2,579 kb; standard deviation: 626 kb), respectively.

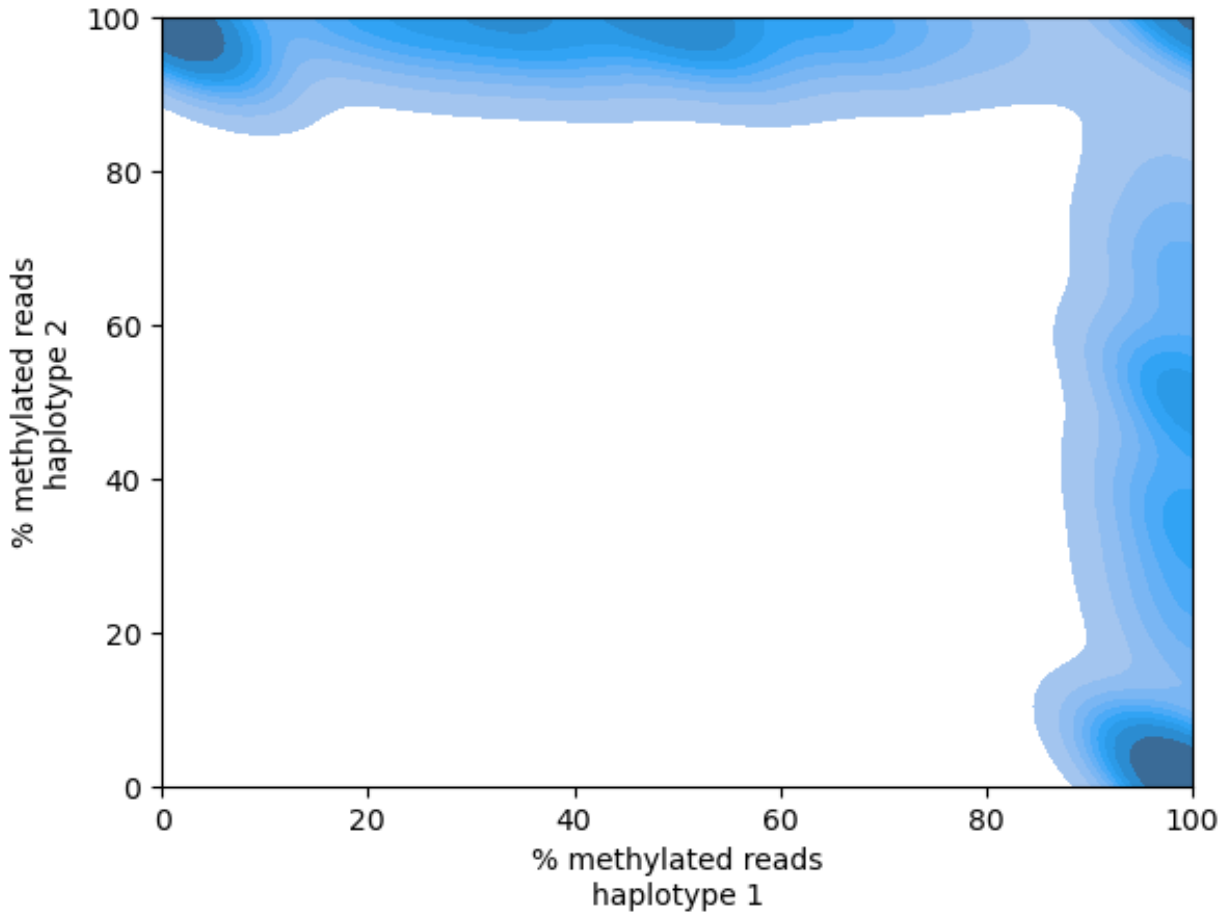

Figure 3. Contour plot the fraction of reads with methylation marks assigned to either haplotype at sites used for phasing by LongHap using 38x PacBio Revio HiFi data. Most sites show clear signals of differential methylation, that is, they are truly phase-informative, as most reads are methylated on one haplotype while most reads are not methylated on the other haplotype (top left and bottom right corners). However, LongHap also identifies some false-positive differentially methylated sites (top right corner), injecting noise into the methylation-based phasing. A read is methylated at a given site if $P(meth) > 0.5$.

Phase-informative methylation sites are characterized by strongly asymmetric methylation patterns between haplotypes. To confirm that the differentially methylated sites leveraged by LongHap are informative for phasing, we compared the fraction of reads with evidence of methylation ($P(meth) > 0.5$) that support each haplotype. Most sites used by LongHap show differential methylation patterns with no or only an intermediate fraction of reads assigned to one haplotype being methylated, while most reads assigned to the other haplotype are unmethylated (Figure 3).

Table 2. Number of phase blocks inferred by each tool using 38x PacBio Revio HiFi data that overlap challenging, medically relevant genes (CMRGs). 247 CMRGs are covered by all tools. LongHap infers the most contiguous phase blocks overlapping CMRGs, as indicated by the lowest number of phase blocks and the second most overlapping base pairs.

	CMRGs covered	Phase blocks overlapping CMRGs	CMRGs fully phased	Phase blocks overlapping CMRGs covered by all tools	Overlapping Base pairs of CMRGs covered by all tools (Mb)
LongHap	250	635	211	633	128.9
LongHap w/o methylation information	248	714	206	714	125.2
LongPhase	254	741	209	764	127.2
WhatsHap	254	774	210	781	127.3
HapCUT2	253	759	209	756	126.6
WhatsHap + MethPhaser	254	873	211	883	126.7

Methylation information enables the phasing of biomedically important loci

To evaluate LongHap's ability to phase a recently curated list of 273 challenging, medically relevant genes (CMRGs)[28], we intersected this gene list with phase block coordinates for each assessed tool. While LongHap has phase blocks covering the fewest CMRGs using PacBio Revio HiFi data (250 versus 253 and 254 for other tools), it fully phased the most CMRGs when considering methylation information (211), along with WhatsHap and WhatsHap + MethPhaser. When limiting the comparison to 247 CMRGs overlapping with at least one phase block of each tool, LongHap achieves the most continuous phasing as indicated by the lowest number phase blocks overlapping those genes (714 and 633 without and with methylation information, respectively) compared to 756 blocks achieved HapCUT2 and covering the most base pairs (128.9 Mb) when integrating methylation information compared to 127.3 Mb achieved by WhatsHap (Table 2). These data highlight the value of integrating methylation information to phase medically

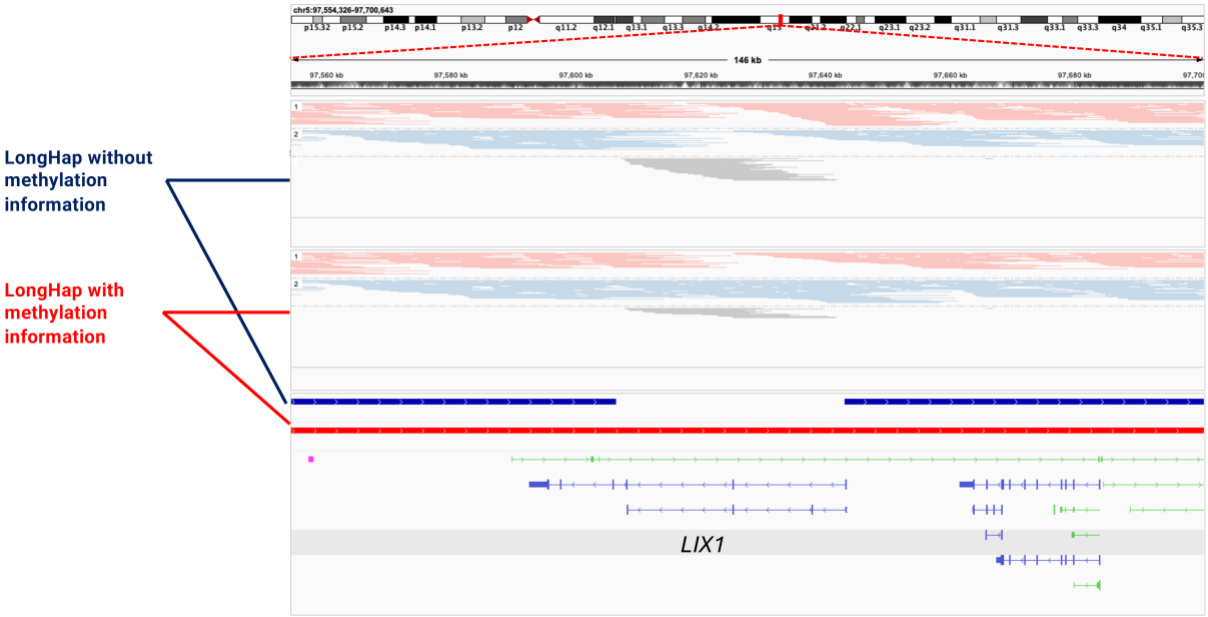

Figure 4. LongHap improves phasing in challenging, medically relevant genes. An example of the improved phasing is shown for the LIX1 locus. By utilizing methylation information, LongHap assigns additional reads to distinct parental haplotypes. In this way, LongHap fills in gaps that cause a phase block to be broken into two if only the sequence data is considered. The top two tracks show haplotagged read alignments when only considering sequence information and when also considering methylation information. Reads assigned to parent *A* and *B* are shown in red and blue, respectively, and unassigned reads are shown in grey. The bottom three tracks show the inferred phase blocks when running LongHap without (blue) and with (red) methylation information and annotated protein-coding genes.

relevant genes, increasing phasing contiguity of such genes. We observe similar patterns when using ONT and UL-ONT sequencing data (Tables S1 & S2).

To illustrate how adding methylation information improves phasing of CMRGs, we analyzed the *LIX1* gene in more detail (CHM13 coordinates: chr5:97592630-97643516; Figure 4). *LIX1* is expressed in the central nervous system and thyroid in humans and is involved in cell regulation and differentiation, as well as mitochondrial metabolism[12–15]. *LIX1* has recently been associated with worse prognoses for individuals with gastrointestinal stromal tumors[29]. When methylation information is not considered, LongHap cannot phase the regions spanning the *LIX1* gene.

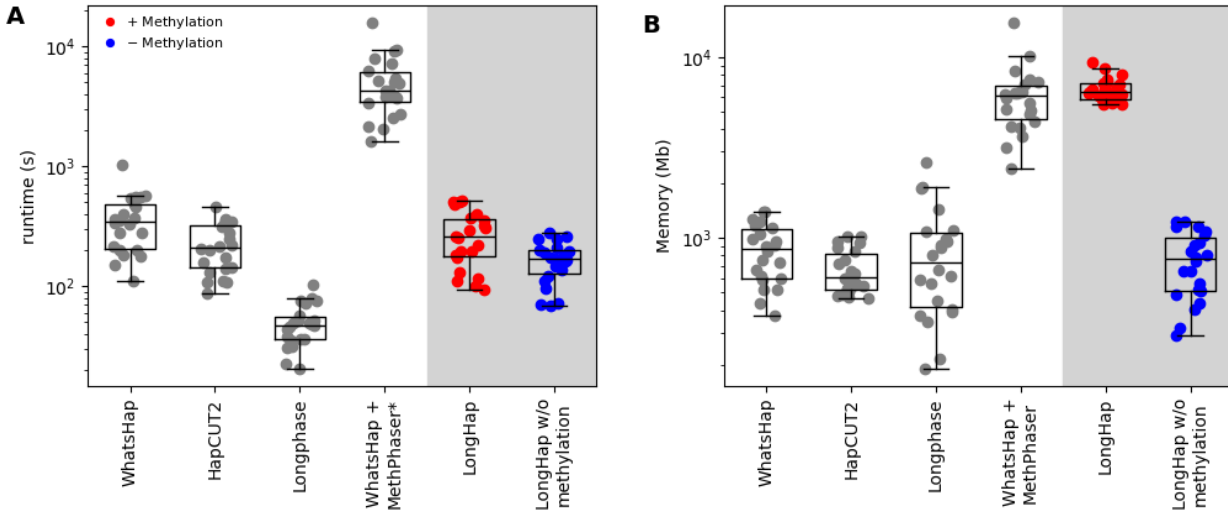

Figure 5. LongHap integrates methylation information with little computational overhead using 38x PacBio Revio HiFi data. **A)** Runtime in seconds and **B)** memory requirements in Mb for each tool for individual chromosomes of the well-characterized genome of HG002. LongHap requires between 70 and 282 seconds per chromosome when not integrating methylation information and between 94 and 523 seconds with methylation information. This is faster than WhatsHap (106 – 912 seconds), HapCUT2 (82 – 522 seconds), and MethPhaser (1,624 – 15,810 seconds without counting the runtime for WhatsHap and additional preparation of input files), but slower than LongPhase (22 – 84 seconds). LongHap requires modest amounts of memory, ranging between 289 Mb and 1,228 Mb without methylation information and 5,475 Mb and 9,315 Mb when integrating methylation information. Note that all y-axes are shown in log scale.

Integrating methylation information enables it to connect two adjacent phase blocks by leveraging phase-informative differentially methylated sites to assign additional reads to distinct haplotypes, avoiding a dropout in haplotype-specific coverage that caused the phase block to be broken into when only sequence information is considered (Figure 4).

Computational performance and scalability

We also evaluated the computational performance of LongHap and additional read-based phasing tools. LongHap runs faster than WhatsHap, HapCUT2, and WhatsHap + MethPhaser, but slower than LongPhase. For example, to phase chromosome 1 of HG002 using 38x PacBio Revio HiFi data with and without methylation information requires 372 seconds (and 6,265 Mb of memory) and 259 seconds (and 1,226 Mb of memory), respectively. In comparison, LongPhase, HapCUT2,

402 WhatsHap, and MethPhaser require 84, 522, 912, and 15,810 seconds, respectively (Figure 5). The
403 runtime estimate of MethPhaser does not include running WhatsHap and haplotagging the
404 sequencing alignments based on WhatsHap's initial phasing, underscoring that LongHap's
405 efficient integration of methylation information causes minimal computational overhead. We
406 observe similar relative computational requirements for ONT and UL-ONT sequencing data,
407 although the runtimes are generally longer for longer read lengths (Figures S6 & S7).

408 In general, LongHap's runtime scales approximately linearly with sequencing coverage and
409 sublinearly with the number of variants. Its runtime, however, also depends on the density of
410 complex variants and low-confidence variants, as they are locally realigned, as well as the read
411 length. Memory requirements are independent of these factors.

412 Discussion

413 Here, we introduce LongHap, a read-based variant phasing algorithm that directly integrates
414 methylation signals native to modern long-read sequencing technology. With PacBio Revio HiFi
415 data, LongHap achieves lower switch error rates and substantially longer phase blocks than
416 existing methods (Figure 2). ONT and UL-ONT sequencing generally yield lower switch error
417 rates due to increased read length and phasing contiguity. With ONT and UL-ONT sequencing,
418 LongHap consistently outperforms WhatsHap and HapCUT2 and produces more contiguous
419 haplotypes than LongPhase, which attains lower switch error rates primarily by avoiding difficult-
420 to-phase variants, including structural variants (Figures S2–S5).

421 To our knowledge, LongHap is the first phasing method to selectively apply belief propagation to
422 embed INDELs, SVs, and low-support variants into a broader haplotype context. This targeted
423 embedding – combined with local realignment around complex breakpoints –improves phasing

accuracy at challenging loci without incurring the computational overhead associated with globally embedding all variants, as reflected by LongHap's ability to phase a large fraction of INDELs and SVs while maintaining a low switch error rate and fast runtimes (Figures 2C, S2C, S4C).

By integrating methylation signals directly into the phasing process, LongHap extracts substantially more information than MethPhaser, a recently developed tool that leverages methylation signals to refine haplotypes inferred by an external phasing tool. While MethPhaser slightly increases switch error rates in our benchmarks, LongHap consistently reduces switch errors and extends phase blocks, demonstrating that joint modeling of genetic and epigenetic signals is more effective than *post hoc* refinement (Figures 2, S2, and S4).

Importantly, the benefits of methylation-aware phasing extend to challenging, medically relevant genes, such as *LIX1* (Table 2 & Figure 4). Harnessing methylation information enables the phasing of these CMRGs that could not be phased based on sequence information alone, underscoring LongHap's ability to support disease-centered projects that require accurate haplotype information.

By jointly leveraging information from read sequences and methylation patterns, LongHap provides a flexible framework that can naturally be extended to additional epigenetic or read-level signals. While orthogonal experimental assays such as chromatin conformation capture can provide powerful long-range phasing information, they are not always practical in population-scale studies due to cost, sample availability, or experimental complexity. As long-read sequencing becomes increasingly prevalent, methods that maximize the information content of these data by integrating complementary signals will be essential.

Our analyses are based on HG002 sequencing data derived from an Epstein–Barr virus–transformed lymphoblastoid cell line of a European individual, which may exhibit atypical

446 methylation patterns. As interindividual variability of methylation has been estimated to be low
447 and cell-type-specific differentially methylated regions are typically demethylated in one cell-
448 type[30], we expect the benefits to transfer to other individuals and samples derived from different
449 tissues, but the benefits may vary across loci. Thus, future studies using primary tissues and diverse
450 genetic ancestries will be important to further characterize the generalizability of methylation-
451 aware phasing.

452 In summary, by jointly leveraging sequence and methylation information in a unified, probabilistic,
453 and computationally efficient framework, LongHap substantially improves the phasing of
454 INDELs, SVs, and medically relevant genomic regions. These advances position LongHap as a
455 powerful tool for studies of genetic variation, disease mechanisms, haplotype-specific epigenetic
456 regulation, and human history.

457 Data availability

458 LongHap and all code for supplementary analyses are available from:
459 https://github.com/AkeyLab/LongHap.

460 Acknowledgements

461 We thank Rob Biermann for providing computational support, helping to improve the
462 computational footprint of LongHap. We are also pleased to acknowledge that the work reported
463 on in this paper was substantially performed using high-performance computing resources at the
464 Lewis-Sigler Institute for Integrative Genomics and the Princeton Research Computing
465 resources at Princeton University. Princeton Research Computing is a consortium of groups,

including the Princeton Institute for Computational Science and Engineering (PICSciE) and Research Computing at Princeton University.

Declaration of Interest

The authors declare no competing interests.

Supplemental information

Figures S1–S7 and Tables S1 and S2

References

1. Tewhey, R., Bansal, V., Torkamani, A., Topol, E. J. & Schork, N. J. The importance of phase information for human genomics. *Nat. Rev. Genet.* **12**, 215–223 (2011).
2. Browning, S. R. & Browning, B. L. Haplotype phasing: existing methods and new developments. *Nat. Rev. Genet.* **12**, 703–714 (2011).
3. Hofmeister, R. J., Ribeiro, D. M., Rubinacci, S. & Delaneau, O. Accurate rare variant phasing of whole-genome and whole-exome sequencing data in the UK Biobank. *Nat. Genet.* **55**, 1243–1249 (2023).
4. Browning, B. L., Tian, X., Zhou, Y. & Browning, S. R. Fast two-stage phasing of large-scale sequence data. *Am. J. Hum. Genet.* **108**, 1880–1890 (2021).
5. Patterson, M. *et al.* WhatsHap: Weighted Haplotype Assembly for Future-Generation Sequencing Reads. *J. Comput. Biol.* **22**, 498–509 (2015).
6. Edge, P., Bafna, V. & Bansal, V. HapCUT2: robust and accurate haplotype assembly for diverse sequencing technologies. *Genome Res.* **27**, 801–812 (2017).
7. Martin, M. *et al.* WhatsHap: fast and accurate read-based phasing. 085050 Preprint at https://doi.org/10.1101/085050 (2016).
8. Fruzangohar, M., Timmins, W. A., Kravchuk, O. & Taylor, J. HaploMaker: An improved algorithm for rapid haplotype assembly of genomic sequences. *GigaScience* **11**, giac038 (2022).
9. Lin, J.-H., Chen, L.-C., Yu, S.-C. & Huang, Y.-T. LongPhase: an ultra-fast chromosome-scale phasing algorithm for small and large variants. *Bioinformatics* **38**, 1816–1822 (2022).
10. Choi, Y., Chan, A. P., Kirkness, E., Telenti, A. & Schork, N. J. Comparison of phasing strategies for whole human genomes. *PLOS Genet.* **14**, e1007308 (2018).

496 11. Fu, Y. *et al.* MethPhaser: methylation-based long-read haplotype phasing of human
497 genomes. *Nat. Commun.* **15**, 5327 (2024).
498 12. Le Guen, L., Notarnicola, C. & de Santa Barbara, P. Intermuscular tendons are essential
499 for the development of vertebrate stomach. *Development* **136**, 791–801 (2009).
500 13. Lonsdale, J. *et al.* The Genotype-Tissue Expression (GTEx) project. *Nat. Genet.* **45**, 580–
501 585 (2013).
502 14. McKey, J., Martire, D., de Santa Barbara, P. & Faure, S. LIX1 regulates YAP1 activity
503 and controls the proliferation and differentiation of stomach mesenchymal progenitors. *BMC*
504 *Biol.* **14**, 34 (2016).
505 15. Guérin, A. *et al.* LIX1-mediated changes in mitochondrial metabolism control the fate of
506 digestive mesenchyme-derived cells. *Redox Biol.* **56**, 102431 (2022).
507 16. Pedersen, B. S. & Quinlan, A. R. cyvcf2: fast, flexible variant analysis with Python.
508 *Bioinformatics* **33**, 1867–1869 (2017).
509 17. Daily, J. Parasail: SIMD C library for global, semi-global, and local pairwise sequence
510 alignments. *BMC Bioinformatics* **17**, 81 (2016).
511 18. Pearl, J. *Probabilistic Reasoning in Intelligent Systems*. (Morgan Kaufmann, 1988).
512 doi:10.1016/B978-0-08-051489-5.50010-2.
513 19. Kschischang, F. R., Frey, B. J. & Loeliger, H.-A. Factor graphs and the sum-product
514 algorithm. *IEEE Trans. Inf. Theory* **47**, 498–519 (2001).
515 20. Yedidia, J. S., Freeman, W. T. & Weiss, Y. Constructing free-energy approximations and
516 generalized belief propagation algorithms. *IEEE Trans. Inf. Theory* **51**, 2282–2312 (2005).

517 21. Hansen, N. F. *et al.* A complete diploid human genome benchmark for personalized
518 genomics. *bioRxiv* https://doi.org/10.1101/2025.09.21.677443 (2025)
519 doi:10.1101/2025.09.21.677443.
520 22. Li, H. Minimap2: pairwise alignment for nucleotide sequences. *Bioinformatics* **34**, 3094–
521 3100 (2018).
522 23. Rhie, A. *et al.* The complete sequence of a human Y chromosome. *Nature* **621**, 344–354
523 (2023).
524 24. Poplin, R. *et al.* A universal SNP and small-indel variant caller using deep neural
525 networks. *Nat. Biotechnol.* **36**, 983–987 (2018).
526 25. Smolka, M. *et al.* Detection of mosaic and population-level structural variants with
527 Sniffles2. *Nat. Biotechnol.* **42**, 1571–1580 (2024).
528 26. Danecek, P. *et al.* Twelve years of SAMtools and BCFtools. *GigaScience* **10**, (2021).
529 27. Quinlan, A. R. & Hall, I. M. BEDTools: a flexible suite of utilities for comparing
530 genomic features. *Bioinformatics* **26**, 841–842 (2010).
531 28. Wagner, J. *et al.* Curated variation benchmarks for challenging medically relevant
532 autosomal genes. *Nat. Biotechnol.* **40**, 672–680 (2022).
533 29. Guérin, A. *et al.* LIX1 regulates YAP activity and controls gastrointestinal cancer cell
534 plasticity. *J. Cell. Mol. Med.* **24**, 9244–9254 (2020).
535 30. Loyfer, N. *et al.* A DNA methylation atlas of normal human cell types. *Nature* **613**, 355–
536 364 (2023).
537