cture(0,0)(-30,0)10 (-30,-5)(0,1)10 (-35,0)(1,0)30 (0,30)10 (-5,30)(1,0)10 (0,35)(0,-1)30 picturepicture(0,0)(30,0)10 (30,-5)(0,1)10 (35,0)(-1,0)30 (0,30)10

Published online *Nucleic Acids Research, 0000, Vol. 00, No. 00* **1–11**
doi:10.1093/nar/gkn000

SpecHap: a diploid phasing algorithm based on spectral graph theory

Yonghan Yu [1,†] **Lingxi Chen** [1,†] **Xinyao Miao** [1,†] **Shuai Cheng Li**[1] *

[1]Department of Computer Science, City University of Hong Kong. Tat Chee Avenue 83, Kowloon, Hong Kong, 999077, China.

ABSTRACT

Haplotype phasing is essential to study diploid eukaryotic organisms. High-throughput sequencing, including next-generation sequencing and third-generation sequencing from different technologies, brings possibilities for haplotype assembly. Although there exist multiple haplotype phasing algorithms, only a few are portable across sequencing technologies with the premise of efficiency and accuracy. Herein, we proposed SpecHap, a novel haplotype assembly tool that leverages spectral graph theory, transforming haplotype phasing into an algebraic problem. On both in silico and whole-genome-sequencing datasets, SpecHap consumed less memory and required less CPU time, yet achieved comparable accuracy comparing to state-of-art methods across all the test instances of next-generation sequencing, linked-reads, high-throughput chromosome conformation capture sequencing, PacBio single-molecule real-time sequencing and Oxford Nanopore long-reads sequencing data. Furthermore, SpecHap successfully phased an individual *Ambystoma mexicanumm*, **a species with gigantic diploid genomes, within 6 CPU hours and 945MB peak memory usage, while other tools failed to yield results either due to a memory overflow (40GB) or a time limit excess (5 days). Our results demonstrated that SpecHap is scalable, efficient and accurate for diploid phasing, supporting diverse sequencing platforms.**

INTRODUCTION

Human and many other animals possess diploid genomes with paternal and maternal sets of chromosomes (Zheng *et al.*, 2016). The two haplotypes, sequential presentation of heterogeneous single nucleotide variation (SNV), small insertion and deletion (INDEL), large genome rearrangement through structure variation or copy number variation, depict the most genetic variation of diploid eukaryotes (Glusman *et al.*, 2014). Phasing, reconstruction of specific allele sequence on individual chromosome, is the fundamental for understanding compound heterozygosity (Tewhey *et al.*, 2011). Many studies have addressed the importance of haplotype phasing, including but not limited to allelic differential, epigenomic regulations, and population development (Consortium *et al.*, 2012, 2010, Onuchic *et al.*, 2018, Tan *et al.*, 2018). Furthermore, several studies have proclaimed that haplotype analysis led to disease susceptibility, which is concealed in unphased single nucleotide polymorphism (SNP) signals (Begnini *et al.*, 2010, Conrad *et al.*, 2006, Musone *et al.*, 2008, Trégouët *et al.*, 2009).

With advances in high-throughput sequencing, multiple sequencing protocols enable credible identifications and linkages of genetic variants, greatly contributing towards the construction of single individual haplotype (SIH). Next-generation sequencing (NGS) technologies, including the Ion Torrent S5 system (Life Technologies), have been widely used to study the proximal haplotypes leveraging paired-end reads (Qi *et al.*, 2018). For high-throughput chromosome conformation capture (Hi-C), HaploSeq verified 95% of heterozygous variants could be phased correctly (Panconesi and Sozio, 2004, Selvaraj *et al.*, 2013). Further, transinteractions between homologous chromosomes complicate the process of phasing. Moreover, segmental duplication and simple repeats are likely related to incorrect contig orientation based on Hi-C data (Burton *et al.*, 2013, Kaplan and Dekker, 2013, Tan *et al.*, 2018). With higher NGS throughput and cost-effectiveness, 10x synthetic long reads (SLRs) protocol provides barcoded linked-reads (>100kb long-range information) and is suitable for assembling and phasing (Zhang *et al.*, 2019, Zheng *et al.*, 2016). Despite the high error rate of ∼15%, the third-generation sequencing (TGS), including single-molecule real-time (SMRT) sequencing technology from Pacific Biosciences (PacBio) and Oxford Nanopore Technologies (ONT), offer ultra-long reads with moderate coverage, significantly promoting the resolution of haplotypes (Edge *et al.*, 2017, Pollard *et al.*, 2018).

20 **********

Several algorithms exist to assemble haplotypes from sequencing reads on diverse protocols. These methods could be summarized into three categories: optimization by reads (fragments) partitioning, by minimum error correction (MEC) and by haplotype likelihood. Several routines are adopted to solve the optimization including dynamic programming, Markov chain Monte Carlo (MCMC), and heuristic graph

*To whom correspondence should be addressed. Tel: +852 3442-9412; Email: shuaicli@cityu.edu.hk
†The authors wish it to be known that, in their opinion, the first three authors should be regarded as joint First Authors.

© The Author(s)
This is an Open Access article distributed under the terms of the Creative Commons Attribution Non-Commercial License (http://creativecommons.org/licenses/by-nc/2.0/uk/) which permits unrestricted non-commercial use, distribution, and reproduction in any medium, provided the original work is properly cited.

cut. **(i)** FastHare, a fast algorithm based on fragments partitioning optimization (Panconesi and Sozio, 2004) produces significant-high error rate (Edge *et al.*, 2017). **(ii)** Bansal *et al.* proposed an MCMC-based algorithm HASH and a max-cut heuristic algorithm HapCUT (Bansal and Bafna, 2008) that both assembling haplotype by minimising the MEC. DCHap, a divide-and-conquer algorithm that combines fragment partitioning and MEC optimization, designed for TGS data, achieved a comparable phasing accuracy (Li and Lin, 2020). **(iii)** Based on graph max-cut algorithm, RefHap presented a formulation for SIH (Duitama *et al.*, 2010). As a general algorithm for human haplotype assembly, HapCUT2, adopts max-cut computations in the haplotype graph to find the haplotype with maximum likelihood (Edge *et al.*, 2017). However, computationally efficient, scalable and accurate methods are still demanded with the advance of sequencing technology.

Herein, we developed SpecHap, a novel fast and accurate scalable algorithm for diploid haplotype phasing designed for multiplex sequencing platforms, especially for TGS error-prone long-reads. Instead of iteratively approaching haplotype by optimizing MEC or haplotype likelihood (Edge *et al.*, 2017), SpecHap assembled haplotype efficiently by transforming haplotype phasing into a standard linear algebra problem with spectral graph theory. We benchmarked SpecHap with four state-of-art phasing software and demonstrated its comparable accuracy on diverse sequencing protocol. Moreover, SpecHap phased an individual of amphibian specie *Ambystoma mexicanumm* (Axolotl), which possesses one of the largest genomes (32 billion base pairs), with $\sim 30\times$ PacBio SMRT long-reads (Nowoshilow *et al.*, 2018, Weisrock *et al.*, 2018).

MATERIALS AND METHODS

SIH involves assembling of fragment which contains allele information at heterozygous variants loci inferred by sequence alignment to reference genome. The heterozygous variants may be determined by the same set of individual genome re-sequencing data. In a simplified situation where the individual genome differs from the reference only by SNVs and the fragments are error-free, haplotype might be obtained through partitioning the fragments into two groups in which fragments contain no conflicting alleles. The elementary bi-partitioning approach is unsuitable to assemble haplotype in real cases where reads might be error-prone.

Several measurements are introduced to handle high throughput error-prone reads. One of the widely adopted criteria, MEC refers to the number of alleles within fragments for each to be consistent with estimated haplotype strain. Most haplotype assembling strategies focus on optimizing MEC objective function, while several algorithms adopt a probabilistic model for haplotype assembly (Duitama *et al.*, 2012, Edge *et al.*, 2017). These methods are algorithms that iteratively infer better haplotype. In this study, we proposed SpecHap, a novel algorithm that assembles haplotype with spectral graph theory.

Spectral Graph Theory

A simple bi-partitioning elaboration of spectral graph theory could be described as follows. Assume an undirected graph G consists of N vertices $V=\{v_1,v_2,...,v_N\}$, and M edges $E=\{e_1,e_2,...,e_M\}$ where each edge $e_m=(v_i,v_j), i\neq j$. The adjacency matrix $A^{N\times N}$ is constructed for graph G to store the linkage relationship between a pair of vertices v_i,v_j, such that $A_{i,j}=w_{i,j}$ if $(v_i,v_j)\in E$ and $A_{i,j}=0$, otherwise. Assume d_i is the degree of a given vertex v_i in graph G. The degree matrix D is a diagonal matrix with i-th diagonal element assigned to d_i: $D_{i,j}=d_i$ if $i=j$, $D_{i,j}=0$ otherwise. Then, by definition, the unnormalised Laplacian matrix L of graph G is constructed as $L=D-A$. According to spectral graph theory, these N vertices can be grouped into two clusters by the element sign $(+,-)$ of the eigenvector corresponding to the second smallest eigenvalue of L (Fiedler vector) (Chung and Graham, 1997). The result obtained through spectral analysis often outperforms classic clustering algorithm and can be solved efficiently through linear algebra (Von Luxburg, 2007).

By constructing a graph with each vertex representing an allele at a heterozygous variant locus, an edge between two vertices of different variants loci demonstrates a potential pair-wise haplotype with edge weight stands for the score or level of confidence for the corresponding haplotype. With the linkage graph structure, spectral graph analysis can be applied and haplotype string is deduced through the Fiedler vector.

Linkage graph construction We first extract the fragments information from the alignment and variant file using modified extractHAIRS (Edge *et al.*, 2017). The fragments are then disassembled into pair-wise linkage information, which could be summarized into the weight of corresponding edge. The average genomic span between pair-wise linked heterozygous variants loci varies among different sequencing technologies. For whole-genome sequencing (WGS) with short insertion, reads are scanned for het-variants loci in a local region with hundreds of base pairs length; With TGS reads, the distance between two connected variant might be extended up to millions of base pair; For 10x linked-reads, barcodes are also examined with its range inferred from the alignment result and hence the linked fragment may cover variants loci separated by thousands of base pairs; For Hi-C data, linkages among het-variants loci apart from millions of base pairs might be extracted.

With the fragment information, a *linkage graph* is constructed. In the linkage graph, a pair of nodes for each heterozygous variant locus represents its two parallel alleles (**Fig1:a,b,c**). For variant locus S_i, we denote its two parallel alleles as S_{iu} and $S_{i\bar{u}}$ respectively. Similarly, for the other het-SNV locus S_j, we have S_{jv} and $S_{j\bar{v}}$. Each fragment is then disassembled into pair-wise linkage between alleles of corresponding variants (S_i and S_j). There are two types of linkages: direct and indirect. Taking alleles S_{iu} and S_{jv} as an instance, the direct linkage is provided by the fragment covering S_{iu} and S_{jv}, while the indirect linkage is provided by the fragment covering parallel alleles $S_{i\bar{u}}$ and $S_{j\bar{v}}$. We integrate both direct and indirect linkages into an edge connecting S_{iu} and S_{jv} nodes. The linkage graph can also be generalized in situations where each node representing a

cture(0,0)(-30,0)10 (-30,-5)(0,1)10 (-35,0)(1,0)30 (0,30)10 (-5,30)(1,0)10 (0,35)(0,-1)30 picturepicture(0,0)(30,0)10 (30,-5)(0,1)10 (35,0)(-1,0)30 (0,30)10

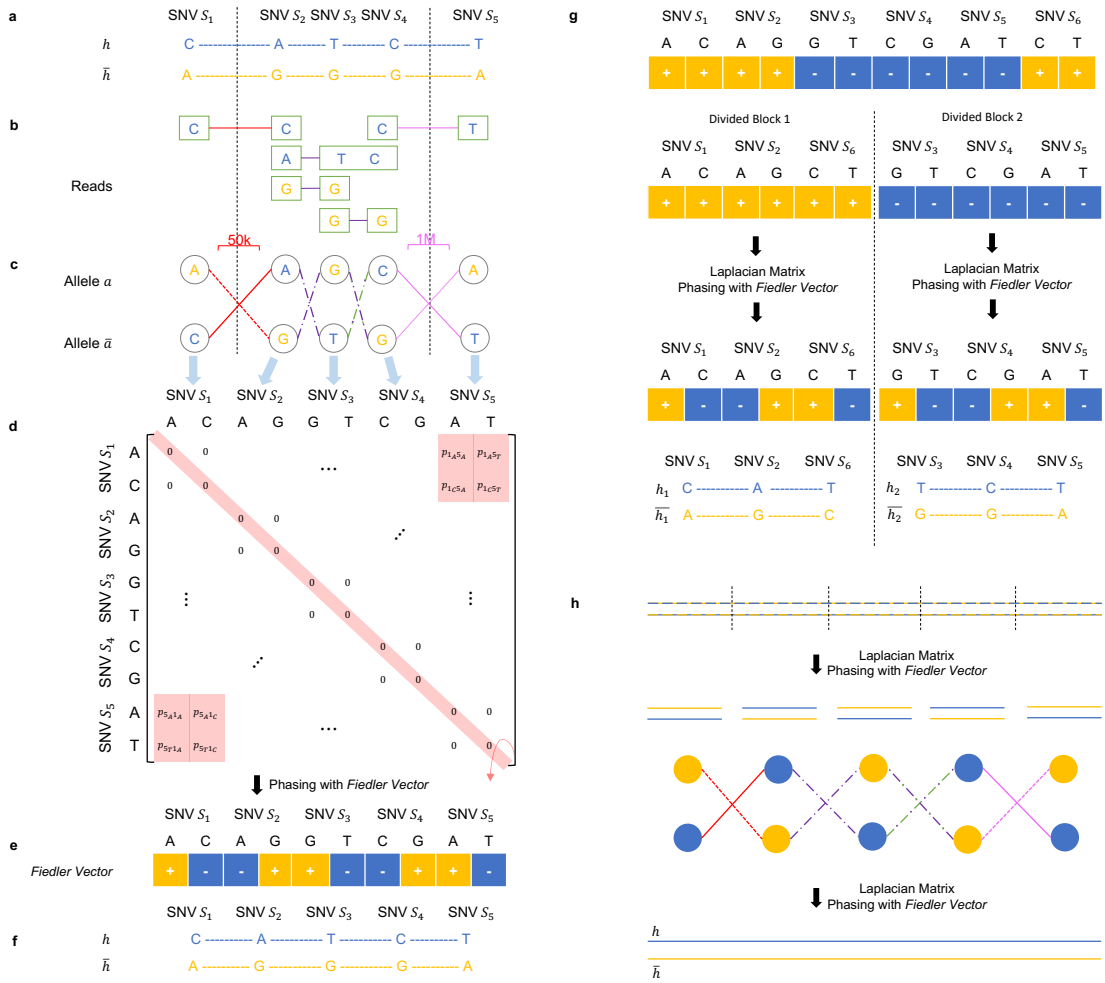

Figure 1. Illustration of SpecHap Algorithm. **(a)** Real haplotypes to be resolved. Blue and yellow alleles are allied to haplotypes h and $\overline{h}$ respectively. **(b)** Observed sequencing reads. Green frame refers to single-end read; red line implies linkage supported by 10x barcode; purple line represents the WGS paired-end linkage; pink line stands for the Hi-C or TGS linkage. **(c)** The linkage graph. The solid edge refers to the direct linkage from sequencing reads, the dashed edge is the indirect linkage from parallel alleles pair, and the "solid-dashed" alternation represents edges composed from direct and indirect linkage together. **(d)** The weighted adjacent matrix for graph. e.g. $p_{5_A 1_C}$ and $p_{1_C 5_A}$ are the same, they denote the linkage probability between SNV S_5 allele "A" and SNV S_1 allele "C". **(e)** Fiedler vector procured by spectral graph theory. **(f)** Resolved haplotypes h and $\overline{h}$ determined by Fiedler vector. **(g)** Simple example demonstrating haplotype construction from Fiedler Vector that guides variants partitioning. **(h)** Connecting haplotype blocks from phased sub-blocks by re-applying Spectral Graph Theory.

phased haplotype block, enabling SpecHap to connect/extend already phased haplotype.

Logarithmic likelihood as edge weight To calculate the edge weight of linkage graph, we introduce a likelihood heuristic method based on the Phred probability of variant in fragments. Consider $q[j]$ as the likelihood that variant j is incorrect on fragment R_i. Given any haplotype string h, the likelihood of observing fragment R_i with j variants is deduced as:

$$p(R_i|h) = \prod_{R_i[j]=h[j]} (1-q_i[j]) \quad \textbf{(1)}$$

Given a self-complemented haplotype pair H=(h, $\bar{h}$), the likelihood $p(R_i|H)$ that a fragment R_i been observed is generalized as MAX($p(R_i|h)$, $p(R_i|\bar{h})$). The likelihood that fragment set R been observed is:

$$p(R|H) = \prod_i p(R_i|H) \quad \textbf{(2)}$$

Our linkage graph incorporates edges that representing conflicting haplotype. To decrease the noise and computational load, we define the edge weight given conflicting haplotype H_1 and H_2 as:

$$E_{H1} = |log(p(R|H_1)) - log(p(R|H_2))|\delta(p(R|H_1), p(R|H_2)) \quad \textbf{(3)}$$

$$E_{H2} = |log(p(R|H_1)) - log(p(R|H_2))|\delta(p(R|H_2), p(R|H_1)) \quad (4)$$

Where as $\delta(a,b) = 1$ if $a > b$ and $\delta(a,b) = 0$ if $a \leq$ b.

Interpretation of Fiedler vector With the adjacency matrix of constructed linkage-graph, the haplotype is constructed based on the Fiedler vector. An expected Fiedler vector resembles the vector demonstrated in **Fig1:e**, in which elements corresponding to parallel alleles of given variant loci are assigned with opposite sign. The haplotype can then be constructed based on the element sign: alleles of different variant loci with the same sign belong to the same haplotype string. However, our experiments demonstrated that such Fiedler vector might not be generated occasionally. There are two exceptions where haplotype cannot be directly inferred from the Fiedler vector. The first case is trivial and it happens when the linkage graph contains pairs of variants loci that hold an equal score of two complement haplotypes. The resulted Fiedler vector possesses equal-signed elements of corresponding locus, which is pruned by our algorithm.

The second situation leads to a Fiedler vector that categorises variants into two sub-blocks, as demonstrated in **Fig1:g**. Such Fiedler vectors is commonly seen with error-prone TGS data and linked-reads where reads with the same barcode might originate from different DNA fragments. To infer haplotype with such Fiedler vector, we partition the adjacency matrix accordingly and apply spectral graph analysis respectively (see example in **Fig1:g**). This process is iteratively applied until SpecHap obtains a Fiedler vector which guides haplotype string. The sub-blocks are then merged together by treating each as a node and constructing a generalised linkage graph (**Fig1:h**)). The SpecHap algorithm might be summarised as below:

Algorithm 1: SpecHap haplotype assembling routine

```
Construct linkage graph with adjacency matrix;
if No conflicting haplotype in graph then
    Return haplotype with depth-first search;
else
    Calculate Fiedler vector;
    if Fiedler vector guides variant partitioning then
        Partition the variant graph accordingly;
        Jump to Line 2;
        Connect the phased haplotype;
    else
        Return Haplotype with Fiedler vector;
    end
end
```

Haplotype Assembling with Linked-Reads

In our experiments of phasing with 10x linked-reads, we identified major inconsistencies between the phased haplotype and the ground truth. Thus, special care was taken in order to obtain maximum phasing accuracy while preserving the completeness of assembled haplotype. First, variants are filtered by their allele depth and quality before the initiation of the phasing algorithm. We also disallow a phasing block from striding over 30 continuously filtered variants. Then, the covering range of each barcode is inferred based on the alignment results. In our implementation, a barcode can neither start nor end on an aligned read with mapping quality less than 30, and the overall barcode spanning length cannot be longer than 60kbp. This procedure reduces the rare but significant situations where reads with the same barcode are from two different DNA molecules.

Haplotype Assembling with Hi-C

For Hi-C data that introduce transinteractions between homologous chromosomes, SpecHap treats possible transinteractions as general errors and does not model them specifically. SpecHap filters read pairs with insertion larger than 40M base-pair to avoid linkage with higher transinteration error rate.

Chromosome Level Haplotype Construction

Whether all the heterozygous variants might be phased at the same time depends on the size and heterozygosity rate of genome. Since many mammalian species possess large genome, a divide and conquer strategy is applied. For simplicity in implementation, a definition of variant block and phasing batch is introduced. First, heterozygous variants that are supposed to be phased together may lie in one variant block. By default, the SpecHap assigns multiple blocks based on the input variant information. Blocks containing no sequencing noise could be phased directly by a graph depth-first search. Second, SpecHap could phase a batch, which maintains user-designated length, of the variant block at one time. Each pair of adjacent batches maintains an overlap of variant blocks, and the phased result of the previous batch is introduced when phasing its neighbor.

Algorithm Complexity

The computation conducted by SpecHap comprises three aspects: depth-first search (DFS), calculation of graph Laplacian and Fiedler vector. The time complexity can be summarized as:

$$O(N) = O(DFS) + O(GraphLaplacian) + O(FiedlerVector) \quad (5)$$

Define k as the number of eigenvector to be calculated, $n = |V|$ (#_of_nodes in graph) and $m = |E|$ (#_of_edges in graph). The time complexity of DFS is $O(n+m)$. The computation of unnormalised graph Laplacian involves matrix addition and the complexity is $O(n^2)$. LU decomposition is adopted for eigen-calculation and the complexity to calculate k smallest eigenvectors is $O(kn^2)$. Therefore:

$$O(N) = O(n+m) + O(n^2) + O(kn^2) \quad (6)$$

SpecHap calculates the first two eigenvectors to obtain the Fiedler vector, i.e. k = 2. Thus, the time complexity of the

algorithm can be summarized as:

$$O(N) = O(n^2) \quad (7)$$

Human Genome Dataset Processing

The phase3 trio-phased haplotype for NA12878 (aligned to hg19) was adopted as groud truth when benchmarking phasing accuracy.

NGS data The NGS alignment data for NA12878 were downloaded from the 1000 Genome Project.

PacBio SMRT WGS data PacBio SMRT WGS data were acquired from GIAB with NCBI accession number SRX1607993. Alignment was performed with minimap2 (Li, 2018) with preset parameters.

Nanopore data Nanopore reads were downloaded from ENA with accession PRJEB30620(Bowden *et al.*, 2019). Alignment was performed with minimap2 (Li, 2018) with preset parameters.

10x linked-reads data 10x linked-reads for NA12878 were gathered from 10x genomics officials at `https://support.10xgenomics.com/genome-exome/datasets/2.0.0/NA12878_WGS` and alignment was performed with LongRanger2.2.1.

Hi-C data Hi-C sequencing data were downloaded from NCBI PRJNA473369. Sequenced reads from seven selected cells (SRR7226668, SRR7226671, SRR7226678, SRR7226679, SRR7226681, SRR7226682 and SRR7226685) were combined for further analysis. The alignment and insertion size of each read-pair were determined with BWA (Li and Durbin, 2009).

Ambystoma Mexicanum Genome Dataset Processing

We also acquired the PacBio SMRT sequencing reads of an individual *Ambystoma mexicanum* with NCBI accession code PRJNA378970 (Nowoshilow *et al.*, 2018). Alignment was performed by minimap2 to its chromosome-scale assembly GCA_002915635.2 (Smith *et al.*, 2019). The called variant is accessible from EBI with accession number ERZ1668256.

Sequencing Data Simulation

The trio-phased sample HG00403 were taken as the haplotype for simulation on chromosome 1, 21 and 22. Reads of length 150bp and insert size 350bp were simulated for NGS and linked-reads with wgsim and LRSIM (Luo *et al.*, 2017) respectively with 30X coverage. 50X PacBio SMRT and ONT reads were also simulated based on same haplotype by PBSIM (Ono *et al.*, 2012) and DeepSimulator (Li *et al.*, 2018) with protocol-specific sequencing error rate.

RESULTS

To address the problem of haplotype phasing in the era of TGS, linked-reads, and Hi-C reads, we developed SpecHap. Instead of assembling haplotype either by fragment partitioning or likelihood optimization, SpecHap utilises spectral graph theory that transforms the problem of haplotype phasing into an algebraic problem. SpecHap also incorporates protocol-specific filtering that improves phasing accuracy with Hi-C and 10x linked-reads. To access the efficiency and accuracy of SpecHap, we benchmarked its performance with existing methods on both simulated and WGS data. Previous publication (Duitama *et al.*, 2012, Edge *et al.*, 2017)s has already demonstrated the robustness of some state-of-art methods: FastHare (Panconesi and Sozio, 2004), RefHap (Duitama *et al.*, 2010) and HapCUT2 (Edge *et al.*, 2017). Some recent publications introduce novel methods, for instance, DCHap (Li and Lin, 2020), with improved efficiency and comparable accuracy. Therefore, we compared SpecHap with four existing methods: FastHare, RefHap, HapCUT2 and DCHap. All programs were benchmarked on CentOS with Intel Xeon CPU E7-4850 v2. CPU and peak memory usage were collected with Oracle Grid Engine. **Fig2** illustrates the overview of the conducted experiment setting and status. SpecHap managed to pass all experiment settings, while the other four software either not supported a specific data type or failed to complete the task within the usage boundary (24 CPU hours for *Homo sapien*, 5 CPU days and 40GB peak memory usage for *Ambystoma mexicanumm*).

SpecHap Demonstrated Runtime and Memory Efficiency

We first used *in silico* data to benchmark SpecHap with the existing method focusing on the third-generation sequencing protocol. Since most methods are not specifically designed for Hi-C and l0X Genomics linked-reads, these methods may not function as intended. We first simulated 50X PacBio SMRT reads and 50X ONT reads with PBSIM (Ono *et al.*,

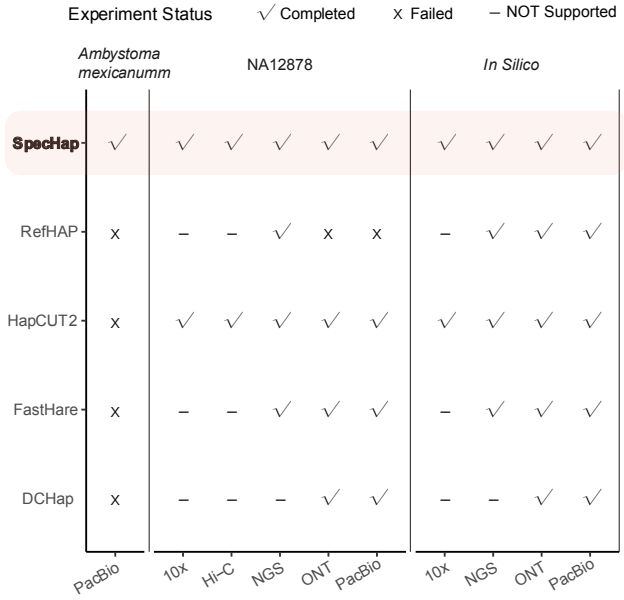

Figure 2. Overview of conducted experiment setting and status. Software that not support specific data type is marked with "-". For *Homo sapien*, methods that exceed 24 CPU hours limits are marked as "Failed". For *Ambystoma mexicanumm*, methods exceed 5 CPU days and 40GB peak memory usage are marked as "Failed".

cture(0,0)(-30,0)10 (-30,-5)(0,1)10 (-35,0)(1,0)30 (0,30)10 (-5,30)(1,0)10 (0,35)(0,-1)30 picturepicture(0,0)(30,0)10 (30,-5)(0,1)10 (35,0)(-1,0)30 (0,30)10

2012) and DeepSimulator (Li *et al.*, 2018) based on the trio-phased haplotype for HG00403 from 1000 Genomes Project on chromosome 1, 21 and 22. To ensure a fair comparison, all the methods adopted the same set of fragments as input and default parameters are used for all sequencing protocol and methods. The CPU time and peak memory usage for all the methods were summarised in **Fig3:a** and Supplementary Table S1.

When phasing with PacBio SMRT sequence, SpecHap outperformed all four existing methods considering both CPU time and peak memory usage based on the simulation. SpecHap was around 40 times faster than HapCUT2, 100 times faster than RefHAP and as fast as FastHare. DCHap did not function efficiently and accurately enough on simulations with PacBio SMRT reads. While assembling haplotype with ONT reads, SpecHap still achieved 20 times faster than HapCUT2, 30 times faster than RefHAP, 2 times faster than DCHAP and as fast as FastHare. FastHare, the second most efficient methods, demonstrated significant higher switch error rate while comparing with other methods (**Fig3:b** and Supplementary Table S2) on both PacBio SMRT and ONT sequence. The runtimes and memory comparison of NGS, Hi-C and 10x linked-reads can be found in Supplementary Table S1:a-c.

As for WGS sample NA12878, SpecHap persisted with its efficiency. It is also worth mentioning that SpecHap consumed minimum peak memory (Mega-Bytes level) compared with other methods (Giga-Bytes level) on WGS sample. With WGS 10x linked-reads, SpecHap and HapCUT2 achieved comparable speed. However, HapCUT2 requires additional computation on fragment linking (32 CPU hours on the linkage of fragment with the script provided by HapCUT2).

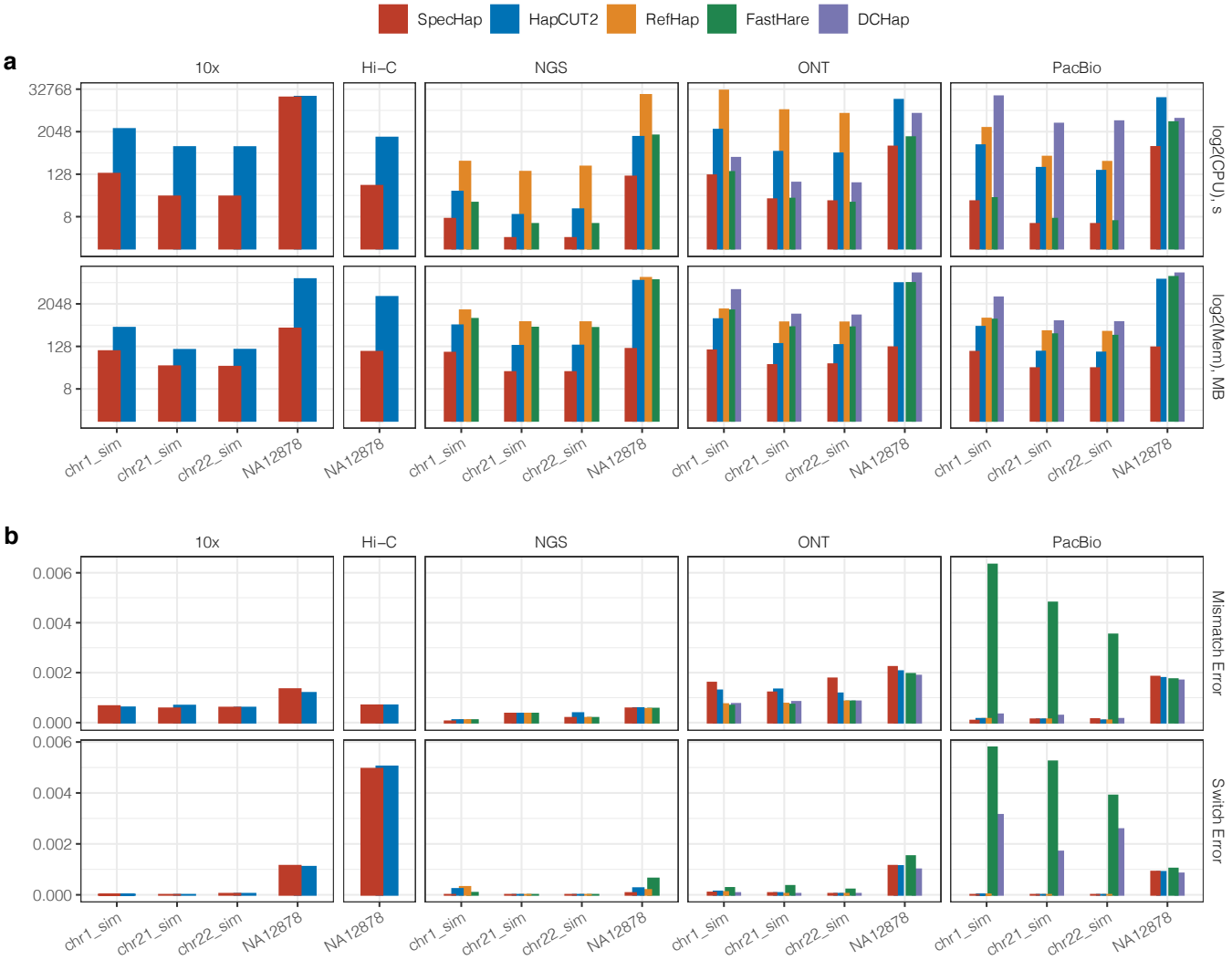

Figure 3. Overall benchmarked data on both simulation and WGS sample NA12878 for diverse sequencing protocol. **a** Log2-scaled CPU and peak memory usage, in second and MegaBytes respectively. Value exceeding 30,000 CPU seconds and 15GB peak memory usage is capped. **b** Overall switch error rate and mismatch error rate, calculated as the number of error divided by the number of possible error.

"output" — 2021/3/12 — 7:35 — page 7 — #7
cture(0,0)(-30,0)10 (-30,-5)(0,1)10 (-35,0)(1,0)30 (0,30)10 (-5,30)(1,0)10 (0,35)(0,-1)30 picturepicture(0,0)(30,0)10 (30,-5)(0,1)10 (35,0)(-1,0)30 (0,30)10

SpecHap Accurately Phased Human Individual NA12878 on Diverse Sequencing Protocols

We assessed the accuracy of SpecHap with an individual NA12878 with five sequencing protocols: NGS, 10x Genomics linked-read, Hi-C, PacBio SMRT and ONT sequencing. The trio-phased haplotype of NA12878 from the 1000 Genomes Project was taken as the ground truth and input to assemble haplotype for each data type. The measurement was conducted based on the metrics of mismatch error rate and switch error rate (**Fig3:b** and **Fig4:a-b**), which are widely adopted criteria while accessing the accuracy of the assembled haplotype (Edge *et al.*, 2017). As for the completeness of the assembled haplotype, AN50, which is the N50 of adjusted haplotype span such that half of all phased heterozygous variant loci are in the segment, and the number of phased SNVs were adopted (**Fig4:c-d**, Supplementary Table S3).

NGS data To access SpecHap on NGS data, we took a high coverage PCR-free sequenced data for individual NA12878 from the 1000 Genomes Project. As displayed in **Fig3:a**, the switch error rate for SpecHap was less than RefHap, the second most accurate algorithm among all other software. HapCUT2 shared virtually similar switch error of RefHap while keeping more variants unpruned. FastHare, however, demonstrated higher switch error rate with more pruned record comparing with SpecHap and HapCUT2. As for the mismatch error rate, all four methods shared comparable results with

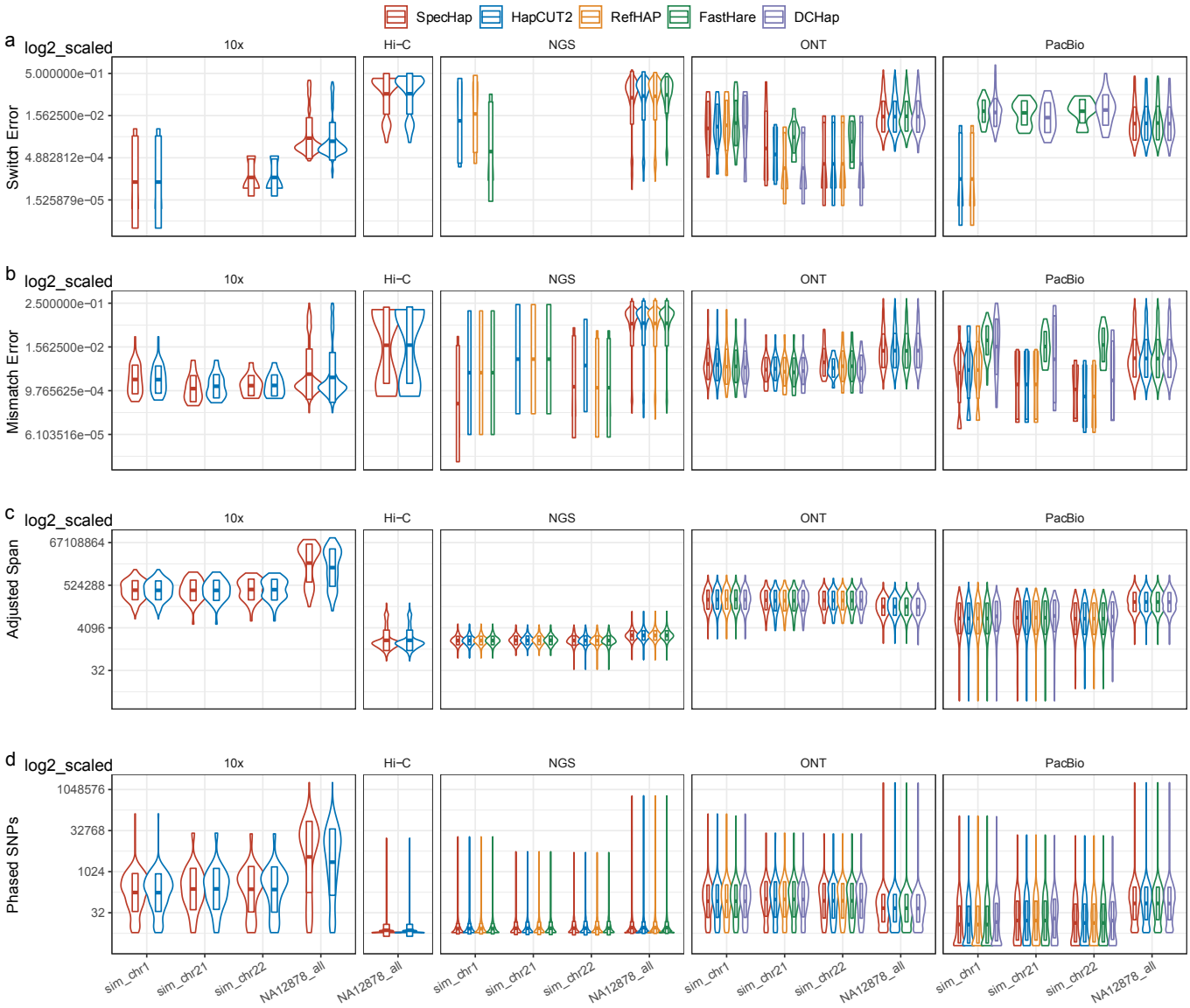

Figure 4. Log2 scaled violin plot of benchmarked data on per-phased-block scale. **a** Switch error rate. Methods with zero switch error on simulation dataset are not showed (10x chr21_sim, NGS chr21_sim, chr22_sim, PacBio chr1_sim, chr21_sim and chr22_sim). **b** Mismatch error rate. **c** Adjusted span, defined as phased block span times ratio of number of phased SNPs over number of total SNPs, in base pair. **d** Number of phased SNPs.

no statistical difference (**Fig3:b** and Supplementary Table S2-a). SpecHap completed the haplotype assembling with the least CPU (2 CPU minutes) and peak memory (109MB) while preserving the phasing accuracy (Supplementary Table S1-a).

Hi-C data Since most methods are not specialized for haplotype assembling with Hi-C, we only compared the quality of haplotype assembly generated by SpecHap and HapCUT2. HapCUT2 models the translocation error by iteratively estimating the probability of reads originating from different homologous chromosomes(Edge *et al.*, 2017). However, SpecHap demonstrated lower comparable switch error rate on the specific data set without modeling translocation error (**Fig3:b** and Supplementary Table S2-c). SpecHap also outperformed HapCUT2 on computational load by completing the haplotype assembly within 1 CPU minute (Supplementary Table S1-c).

10x linked-read data We benchmarked SpecHap on 10x genomics linked-reads. The 10x linked-reads label short reads that originated from a single long DNA fragment with the same barcode. Although DNA fragment generally spans around 60k base-pair range, reads might be sparsely distributed across the original fragment. It is also possible that two reads who shared the same barcode originated from different DNA molecules. The data set we acquired has around 50X coverage and reads were filtered that fragments with white-listed barcode were kept. The same set of variants was used to extract fragment information. SpecHap achieved a comparable accuracy by completing the assembly with 5 CPU hours while HapCUT2 spent 5.6 CPU hours (Supplementary Table S1-b). Moreover, SpecHap takes raw fragment as input while other methods requires extra computation on fragment linking. The linkage of fragment was conducted with the script provided by HapCUT2 and took 32 CPU hours to complete.

PacBio SMRT data and ONT data PacBio SMRT sequencing and ONT sequencing are known for their error-prone ($\sim$10%) long reads. We compared the performance of SpecHap with other methods on 44X coverage PacBio reads and 40X coverage ONT reads. Fragment information was extracted with the same set of variants call. On PacBio SMRT data, SpecHap achieved the fastest speed (21 CPU minutes) while achieving virtually identical switch error rate comparing with HapCUT2 (**Fig3:a** and Supplementary Table S1-d). FastHare, with second-best efficiency, demonstrated higher switch error rate with most variant pruned (**Fig3:b**). DCHap, with slightly more accurate result, assembled the haplotype with the lowest AN50 and pruned the second most variants (Supplementary Table S3-d). All methods demonstrated no significant statistical differences considering the mismatch error rate. On ONT data (**Fig3:a** and Supplementary Table S1-e), SpecHap persisted its efficiency by completing the assembly within 17 CPU minutes with comparable switch error rate. DCHap, with the least switch errors, pruned more than 2000 SNVs than SpecHap and HAPCUT2 (Supplementary Table S3-e). RefHap failed to finish the assembly process with both PacBio SMRT reads and ONT reads due to excessive time consumption and accuracy estimation was not available.

SpecHap Demonstrated Scalability by Phasing with Large Diploid Genome

SpecHap also demonstrated the scalability by assembling the 32X PacBio SMRT sequenced reads with N50 read length around 14.2kbp of an *Ambystoma mexicanum* individual which posses 32 billion base-pair-long genome (Nowoshilow *et al.*, 2018, Smith *et al.*, 2019). The variant set of the same individual that containing more than 20 million heterozygous SNV loci. As illustrated in Supplementary Figure 1, SpecHap were able to finish the assembly within 6 CPU hours with only 945MB of peak memory consumption, while all other methods failed to finish within time (5 CPU days) and memory limit (40GB).

DISCUSSION

SpecHap, a novel diploid phasing algorithm, supports sequencing data with different coverage from diverse platforms. SpecHap transforms haplotype phasing into a linear algebra problem by applying spectral graph theory. The Fiedler Vector guided allele partitioning might be interpreted as min-cut on our linkage graph (Von Luxburg, 2007). To our knowledge, this is the first time to employ spectral graph theory in haplotype phasing. Although there is no guarantee that SpecHap provides optimal solutions, we demonstrated that our model succeeded in efficiently and accurately assembling haplotypes with diverse sequencing reads from individual NA12878. SpecHap is also scalable to phase one of the largest diploid genomes of *Ambystoma mexicanumm*.

Long-reads sequencing introduces significant advances considering the completeness of assembled haplotype. However, SMRT and ONT long-reads maintain higher error rate per base and may fail to accurately detect SNVs, particularly heterozygous ones (Edge and Bansal, 2019). In our experiments, we adopted the trio-phased high-quality variants set for NA12878 from the 1000 Genome Project as input for all methods. Since most SIH phasing methods including SpecHap requires a high-quality set of variants, 30X Illumina short reads sequencing might be performed to obtain reliable calls for SNVs and INDELs. Some recent approaches based on deep learning were also able to identify variants accurately from long-reads sequencing (Luo *et al.*, 2019).

In addition to the wet-lab based phasing (Redin *et al.*, 2019), computational approaches including population-based phasing, Mendel's Law based phasing and SIH are also widely adopted (Majidian and Sedlazeck, 2020). **(i)** Although performed at a lower coverage, the population based phasing requires a haplotype population structure model and can only phase common variations (Lutgen *et al.*, 2020). Genomes of multiple non-human individuals in one population under natural condition are often difficult to obtain. **(ii)** A chromosome-scale haplotype might be constructed by applying the Mendelian laws of inheritance (also known as trio-based phasing) (Blackburn *et al.*, 2020). However, it lacks the ability to phase *de novo* mutations, especially in Mendelian diseases. Trio-based phasing usually requires high coverage sequencing and sequences from both parents (Delaneau *et al.*, 2012, Garg *et al.*, 2016). **(iii)** Compared with previously mentioned phasing methods, SIH is the most comprehensive approach as it incorporates de novo mutations and rare

variations. Although NGS usually brings the possibility of hindering due to fragmentation, the development of linked-reads and TGS long-reads introduces significant advance on phasing quality of SIH (Browning and Browning, 2011, Sedlazeck *et al.*, 2018). Thus, we developed SpecHap here, a novel method for haplotype assembly compatible with diverse sequencing protocol.

Although WGS phasing has grown as an easy-to-handle approach for studying genetic variations within or between species with "normal" genome size, this technology is immature for non-model large genomes yet (Weisrock *et al.*, 2018). From the perspective of conservation, larger genomes are assumed to become particularly vulnerable during periods of increased extinction (Vinogradov, 2003). Also, lineages with giant genomes seem to be eliminated from the most extreme habitats (KNIGHT *et al.*, 2005). The largest genomes are presently unearthed in amphibian salamanders, with sizes ranging from 14-120 gigabytes (Newman and Austin, 2016). Salamanders, the non-model species, represent the most massive regenerative repertoire, capable of rebuilding and restoring multiple organs and systems such as brain structures, and has the potential to become model organisms (Elewa *et al.*, 2017). The genome of Mexican salamander, axolotl, with a size ($\sim$ 32 gigabytes) approximately ten times bigger than homo sapiens', is one of the largest genomes (Nowoshilow *et al.*, 2018). Due to the high raising and sequencing cost for a cohort of axolotls, scientists virtually rely on an individual axolotl genome assembly other than population-based phasing to conduct biological investigations (Sotero-Caio *et al.*, 2017). Nevertheless, our experiments illustrated that some state-of-art haplotype assembly tools failed to handle the oversized genomes. We managed to phase the haplotype of axolotl with PacBio SMRT sequencing data, implying SpecHap might be a compatible tool for future research on the evolutionary history of amphibians and other organisms with immense genome scale.

In this study, we introduced a novel diploid SIH phasing algorithm SpecHap and demonstrated its robustness and efficiency. By transforming haplotype phasing into a linear algebra problem, SpecHap assembled haplotype with ultra-fast speed while preserving comparable accuracy. Moreover, a comprehensive analysis on the influence of technological specific error over phasing quality may be conducted for Hi-C and 10x linked-reads. Although our algorithm works on diploid genome, generalization towards high ploidy is expected. Besides, determination of haplotype-resolved structural variation might also be an important feature to be introduced in the future.

ACKNOWLEDGEMENTS

We would like to express sincere gratitude to Dr. Yen Kaow Ng from Kotai Biotechnologies, Japan for manuscript revision and to Dr. Zijun Xiong of the Chinese Academy of Sciences for suggestions on data collection. We appreciate Miss Santu for her creation of an image for *Ambystoma mexicanumm*. We would also like to thank Dr. Wenlong Jia and Mr. Bowen Tan for their valuable assiatance and advice.

Conflict of interest statement.

None declared.

Data Availability

All the data used in this paper can be retrieved from public database. All the experiments are reproducible with dedicated version of software with default arguments.

Software Availability

SpecHap source code is deployed at `https://github.com/deepomicslab/SpecHap` and a copy is attached in the supplemental material. The fragment information was extracted with a modified version of extractHAIRS (Edge *et al.*, 2017) which is packed with SpecHap. We adopted the implementation of FastHare from Duitama *et al.* (2012) for benchmarking.

REFERENCES

Bansal, V. and Bafna, V. (2008). Hapcut: an efficient and accurate algorithm for the haplotype assembly problem. *Bioinformatics*, **24**(16), i153–i159.

Begnini, A., Tessari, G., Turco, A., Malerba, G., Naldi, L., Gotti, E., Boschiero, L., Forni, A., Rugiu, C., Piaserico, S., *et al.* (2010). Ptch1 gene haplotype association with basal cell carcinoma after transplantation. *British Journal of Dermatology*, **163**(2), 364–370.

Blackburn, A. N., Blondell, L., Kos, M. Z., Blackburn, N. B., Peralta, J. M., Stevens, P. T., Lehman, D. M., Blangero, J., and Göring, H. H. (2020). Genotype phasing in pedigrees using whole-genome sequence data. *European Journal of Human Genetics*, pages 1–14.

Bowden, R., Davies, R. W., Heger, A., Pagnamenta, A. T., de Cesare, M., Oikkonen, L. E., Parkes, D., Freeman, C., Dhalla, F., Patel, S. Y., Popitsch, N., Ip, C. L. C., Roberts, H. E., Salatino, S., Lockstone, H., Lunter, G., Taylor, J. C., Buck, D., Simpson, M. A., and Donnelly, P. (2019). Sequencing of human genomes with nanopore technology. *Nature Communications*, **10**(1), 1869.

Browning, S. R. and Browning, B. L. (2011). Haplotype phasing: existing methods and new developments. *Nature Reviews Genetics*, **12**(10), 703–714.

Burton, J. N., Adey, A., Patwardhan, R. P., Qiu, R., Kitzman, J. O., and Shendure, J. (2013). Chromosome-scale scaffolding of de novo genome assemblies based on chromatin interactions. *Nature biotechnology*, **31**(12), 1119–1125.

Chung, F. R. and Graham, F. C. (1997). *Spectral graph theory*. Number 92. American Mathematical Soc.

Conrad, D. F., Jakobsson, M., Coop, G., Wen, X., Wall, J. D., Rosenberg, N. A., and Pritchard, J. K. (2006). A worldwide survey of haplotype variation and linkage disequilibrium in the human genome. *Nature genetics*, **38**(11), 1251.

Consortium, . G. P. *et al.* (2012). An integrated map of genetic variation from 1,092 human genomes. *Nature*, **491**(7422), 56.

Consortium, I. H. . *et al.* (2010). Integrating common and rare genetic variation in diverse human populations. *Nature*, **467**(7311), 52.

Delaneau, O., Marchini, J., and Zagury, J.-F. (2012). A linear complexity phasing method for thousands of genomes. *Nature methods*, **9**(2), 179–181.

Duitama, J., Huebsch, T., McEwen, G., Suk, E.-K., and Hoehe, M. R. (2010). Refhap: a reliable and fast algorithm for single individual haplotyping. In *Proceedings of the First ACM International Conference on Bioinformatics and Computational Biology*, pages 160–169.

Duitama, J., McEwen, G. K., Huebsch, T., Palczewski, S., Schulz, S., Verstrepen, K., Suk, E.-K., and Hoehe, M. R. (2012). Fosmid-based whole genome haplotyping of a hapmap trio child: evaluation of single individual haplotyping techniques. *Nucleic acids research*, **40**(5), 2041–2053. 22102577[pmid].

Edge, P. and Bansal, V. (2019). Longshot enables accurate variant calling in diploid genomes from single-molecule long read sequencing. *Nature communications*, **10**(1), 1–10.

Edge, P., Bafna, V., and Bansal, V. (2017). Hapcut2: robust and accurate haplotype assembly for diverse sequencing technologies. *Genome research*, **27**(5), 801–812.

Elewa, A., Wang, H., Talavera-López, C., Joven, A., Brito, G., Kumar, A., Hameed, L. S., Penrad-Mobayed, M., Yao, Z., Zamani, N., *et al.* (2017). Reading and editing the pleurodeles waltl genome reveals novel features of tetrapod regeneration. *Nature communications*, **8**(1), 1–9.

Garg, S., Martin, M., and Marschall, T. (2016). Read-based phasing of related individuals. *Bioinformatics*, **32**(12), i234–i242.

Glusman, G., Cox, H. C., and Roach, J. C. (2014). Whole-genome haplotyping approaches and genomic medicine. *Genome medicine*, **6**(9), 73.

Kaplan, N. and Dekker, J. (2013). High-throughput genome scaffolding from in vivo dna interaction frequency. *Nature biotechnology*, **31**(12), 1143–1147.

KNIGHT, C. A., MOLINARI, N. A., and PETROV, D. A. (2005). The Large Genome Constraint Hypothesis: Evolution, Ecology and Phenotype. *Annals of Botany*, **95**(1), 177–190.

Li, H. (2018). Minimap2: pairwise alignment for nucleotide sequences. *Bioinformatics*, **34**(18), 3094–3100.

Li, H. and Durbin, R. (2009). Fast and accurate short read alignment with Burrows–Wheeler transform. *Bioinformatics*, **25**(14), 1754–1760.

Li, Y. and Lin, Y. (2020). Dchap: A divide-and-conquer haplotype phasing algorithm for third-generation sequences. *IEEE/ACM Transactions on Computational Biology and Bioinformatics*.

Li, Y., Han, R., Bi, C., Li, M., Wang, S., and Gao, X. (2018). DeepSimulator: a deep simulator for Nanopore sequencing. *Bioinformatics*, **34**(17), 2899–2908.

Luo, R., Sedlazeck, F. J., Darby, C. A., Kelly, S. M., and Schatz, M. C. (2017). Lrsim: a linked-reads simulator generating insights for better genome partitioning. *Computational and structural biotechnology journal*, **15**, 478–484.

Luo, R., Sedlazeck, F. J., Lam, T.-W., and Schatz, M. C. (2019). A multi-task convolutional deep neural network for variant calling in single molecule sequencing. *Nature Communications*, **10**(1), 998.

Lutgen, D., Ritter, R., Olsen, R.-A., Schielzeth, H., Gruselius, J., Ewels, P., García, J. T., Shirihai, H., Schweizer, M., Suh, A., *et al.* (2020). Linked-read sequencing enables haplotype-resolved resequencing at population scale. *Molecular Ecology Resources*.

Majidian, S. and Sedlazeck, F. J. (2020). PhaseME: Automatic rapid assessment of phasing quality and phasing improvement. *GigaScience*, **9**(7). giaa078.

Musone, S. L., Taylor, K. E., Lu, T. T., Nititham, J., Ferreira, R. C., Ortmann, W., Shifrin, N., Petri, M. A., Kamboh, M. I., Manzi, S., *et al.* (2008). Multiple polymorphisms in the tnfaip3 region are independently associated with systemic lupus erythematosus. *Nature genetics*, **40**(9), 1062.

Newman, C. E. and Austin, C. C. (2016). Sequence capture and next-generation sequencing of ultraconserved elements in a large-genome salamander. *Molecular ecology*, **25**(24), 6162–6174.

Nowoshilow, S., Schloissnig, S., Fei, J.-F., Dahl, A., Pang, A. W. C., Pippel, M., Winkler, S., Hastie, A. R., Young, G., Roscito, J. G., Falcon, F., Knapp, D., Powell, S., Cruz, A., Cao, H., Habermann, B., Hiller, M., Tanaka, E. M., and Myers, E. W. (2018). The axolotl genome and the evolution of key tissue formation regulators. *Nature*, **554**(7690), 50–55.

Ono, Y., Asai, K., and Hamada, M. (2012). PBSIM: PacBio reads simulator—toward accurate genome assembly. *Bioinformatics*, **29**(1), 119–121.

Onuchic, V., Lurie, E., Carrero, I., Pawliczek, P., Patel, R. Y., Rozowsky, J., Galeev, T., Huang, Z., Altshuler, R. C., Zhang, Z., *et al.* (2018). Allele-specific epigenome maps reveal sequence-dependent stochastic switching at regulatory loci. *Science*, **361**(6409), eaar3146.

Panconesi, A. and Sozio, M. (2004). Fast hare: A fast heuristic for single individual snp haplotype reconstruction. In I. Jonassen and J. Kim, editors, *Algorithms in Bioinformatics*, pages 266–277, Berlin, Heidelberg. Springer Berlin Heidelberg.

Pollard, M. O., Gurdasani, D., Mentzer, A. J., Porter, T., and Sandhu, M. S. (2018). Long reads: their purpose and place. *Human molecular genetics*, **27**(R2), R234–R241.

Qi, J., Wang, T.-J., Chen, L.-P., Wang, X.-F., Wang, M.-N., and Wu, J.-H. (2018). Utility of next-generation sequencing methods to identify the novel hla alleles in potential stem cell donors from chinese marrow donor program. *International journal of immunogenetics*, **45**(4), 225–229.

Redin, D., Frick, T., Aghelpasand, H., Käller, M., Borgström, E., Olsen, R.-A., and Ahmadian, A. (2019). High throughput barcoding method for genome-scale phasing. *Scientific reports*, **9**(1), 1–8.

Sedlazeck, F. J., Lee, H., Darby, C. A., and Schatz, M. C. (2018). Piercing the dark matter: bioinformatics of long-range sequencing and mapping. *Nature Reviews Genetics*, **19**(6), 329–346.

Selvaraj, S., Dixon, J. R., Bansal, V., and Ren, B. (2013). Whole-genome haplotype reconstruction using proximity-ligation and shotgun sequencing. *Nature biotechnology*, **31**(12), 1111–1118.

Smith, J. J., Timoshevskaya, N., Timoshevskiy, V. A., Keinath, M. C., Hardy, D., and Voss, S. R. (2019). A chromosome-scale assembly of the axolotl genome. *Genome Res*, **29**(2), 317–324.

Sotero-Caio, C. G., Platt, R. N., Suh, A., and Ray, D. A. (2017). Evolution and diversity of transposable elements in vertebrate genomes. *Genome biology and evolution*, **9**(1), 161–177.

Tan, L., Xing, D., Chang, C.-H., Li, H., and Xie, X. S. (2018). Three-dimensional genome structures of single diploid human cells. *Science*, **361**(6405), 924–928.

Tewhey, R., Bansal, V., Torkamani, A., Topol, E. J., and Schork, N. J. (2011). The importance of phase information for human genomics. *Nature Reviews Genetics*, **12**(3), 215.

Trégouët, D.-A., König, I. R., Erdmann, J., Munteanu, A., Braund, P. S., Hall, A. S., Großhennig, A., Linsel-Nitschke, P., Perret, C., DeSuremain, M., *et al.* (2009). Genome-wide haplotype association study identifies the slc22a3-lpal2-lpa gene cluster as a risk locus for coronary artery disease. *Nature genetics*, **41**(3), 283.

Vinogradov, A. E. (2003). Selfish dna is maladaptive: evidence from the plant red list. *Trends in Genetics*, **19**(11), 609 – 614.

cture(0,0)(-30,0)10 (-30,-5)(0,1)10 (-35,0)(1,0)30 (0,30)10 (-5,30)(1,0)10 (0,35)(0,-1)30 picturepicture(0,0)(30,0)10 (30,-5)(0,1)10 (35,0)(-1,0)30 (0,30)10

Von Luxburg, U. (2007). A tutorial on spectral clustering. *Statistics and Computing*, **17**(4), 395–416.

Weisrock, D. W., Hime, P. M., Nunziata, S. O., Jones, K. S., Murphy, M. O., Hotaling, S., and Kratovil, J. D. (2018). *Surmounting the Large-Genome "Problem" for Genomic Data Generation in Salamanders*, pages 1–28. Springer International Publishing, Cham.

Zhang, L., Zhou, X., Weng, Z., and Sidow, A. (2019). Assessment of human diploid genome assembly with 10x linked-reads data. *GigaScience*, **8**(11), giz141.

Zheng, G. X., Lau, B. T., Schnall-Levin, M., Jarosz, M., Bell, J. M., Hindson, C. M., Kyriazopoulou-Panagiotopoulou, S., Masquelier, D. A., Merrill, L., Terry, J. M., *et al.* (2016). Haplotyping germline and cancer genomes with high-throughput linked-read sequencing. *Nature biotechnology*, **34**(3), 303.