BMC Genomics

SOFTWARE

Open Access

# nTChap: an accurate method for polyploid haplotype reconstruction

Yun Gao[1,2,3], Junhai Qi[1,2], Ting Yu[1,2*] and Guojun Li[1,2*]

**Abstract**

Haplotypes of polyploid organisms provide important insights into polyploid evolution and advanced breeding strategies. Significant challenges remain in polyploid haplotype phasing, including the great number of haplotype copies and the presence of haplotypes with high local sequence similarity. We present nTChap, a new reference-based polyploid phasing algorithm for long reads, which applies an iterative procedure involving a two-round clustering and consensus construction to generate accurate haplotypes. Using simulated datasets, we show that nTChap outperforms current tools in terms of accuracy and error rates, while maintaining robust performance across high-ploidy genomes and low-coverage data. We then show that nTChap assembles reasonable haplotype structures using real datasets.

**Keywords** Polyploid, Phasing, Haplotypes, Clustering, Third-generation sequencing

## Introduction

Polyploid organisms, which possess more than two homologous sets of chromosomes, are widespread in nature, including many important agricultural crops (e.g., wheat, potato, and sugarcane), as well as significant evolutionary plant lineages. Polyploid organisms frequently demonstrate enhanced genetic and phenotypic plasticity, with polyploidy standing as a major driver of plant evolution, empowering rapid adaptation to drastically shifting environments [1–3]. Homologous chromosomes can be heterozygous. The sequence of variants on a single copy of a chromosome is called a haplotype [4]. Determining which alleles are located on the same chromosomal copy is known as haplotype phasing. Haplotypes play an important role in understanding the interplay of evolutionary processes and plant breeding strategies in agricultural improvements [5, 6].

Diploid phasing involves distinguishing two parental chromosomes and their haplotypes are homologous in genomic level and complementary at single-nucleotide polymorphism (SNP) level: determining one haplotype could directly get the another one. In contrast, polyploid phasing requires disentangling multiple homologous chromosomes, each potentially harboring distinct allelic combinations. This complexity arises from genome duplications and allele-specific structural variations, which underpin phenotypic diversity and adaptation in polyploids [7]. We focus on phasing polyploid by using third-generation sequencing data. Although short reads are more accurate, they are often too short to span multiple heterozygous sites consistently across longer distances, posing limitations for complex genomes. However, the long sequencing reads can span more variant positions, directly enabling the long phasing of variants within individual reads.

*Correspondence:
Ting Yu
yutingsdu@163.com
Guojun Li
guojunsdu@gmail.com
[1]Research Center for Mathematics and Interdisciplinary Sciences, Shandong University, Qingdao 266237, China
[2]Frontiers Science Center for Nonlinear Expectations, Ministry of Education, Shandong University, Qingdao 266237, China
[3]School of Mathematics, Shandong University, Jinan 250100, China

The first method for phasing polyploid genomes is HapCompass [8, 9], which constructs a compass graph and minimizes the conflicts between reads by finding a spanning tree of the graph. HapTree [10] uses the relative likelihood algorithm to find most likely haplotypes. This approach lays the first theoretical foundation of polyploidy phasing problems [6]. SDhaP [11] solves an approximate Minimum Error Correction (MEC) model by a semi-definite programming approach. H-PoP [12] is a read-clustering based method by solving Polyploid Balanced Optimal Partition (PBOP) model, which can be seen as a generalization of the MEC model, as k = 2, it equals the diploid MEC model. WhatsHap polyphase [13], designed specifically for long-read sequencing, applies heuristics to solve the cluster editing problem. nPhase [14] partitions the long reads into haplotypes by iteratively clustering similar reads together until unique haplotypes remain. Flopp [15] partitions the long reads by solving the uniform probabilistic error minimization (UPEM) function. The locally identical haplotypes complicate the phasing of haplotypes. Computational limitations also persist, as existing algorithms often struggle with scalability for high-ploidy genomes (e.g., hexaploids or octoploids or higher). Additionally, improving phasing accuracy remains an important area of research.

Here, we present nTChap, an algorithm for haplotype reconstruction of the polyploid genome from long reads sequencing data. It applies an iterative procedure involving a **t**wo-round **c**lustering and consensus construction to obtain accurate **hap**lotypes, where the first round of clustering using the label propagation algorithm [16] is based on the similarity between reads and the second round further subdivides the clusters obtained from the first round to account for local similarities between haplotypes. It does not require prior knowledge of the ploidy of the reference. We validated the performance of nTChap on two simulated and two real datasets. The evaluation results demonstrated that nTChap showed significant improvement over the other methods, including WhatsHap polyphase, nPhase, and flopp, in terms of both accuracy and error rates.

### Overview of the nTChap

nTChap requires a reference genome, a set of aligned reads and SNPs as input and then outputs the phased SNPs and the clustered reads for each haplotype (Fig. 1). The long reads aligned to the reference are reduced to a set of heterozygous SNPs, which we call reduced reads. On the first iteration, nTChap builds an overlap graph, which encodes the pairwise distances between the reduced reads. It then clusters reduced reads iteratively in three phases: rough clustering, sub-clustering and consensus. nTChap firstly uses the label propagation algorithm [16] to partition the graph into densely connected clusters. Due to the local similarities between different haplotypes and the sequencing errors, reads originating from two distinct haplotypes may contain partially identical SNPs, which cannot distinguish between these two haplotypes but can differentiate them from other haplotypes. Consequently, reads derived from two different haplotypes might be clustered together after roughly clustering. Therefore, we perform finer subdivisions for such clusters to achieve more precise haplotype separation (see Methods section for details). After conducting two rounds of clustering, we compute consensus sequences for every cluster. These consensus sequences subsequently are served as novel "reduced reads" to build a new overlap graph and participate in the next iteration. This process is carried out iteratively until the graph becomes indivisible during either of the two clustering steps.

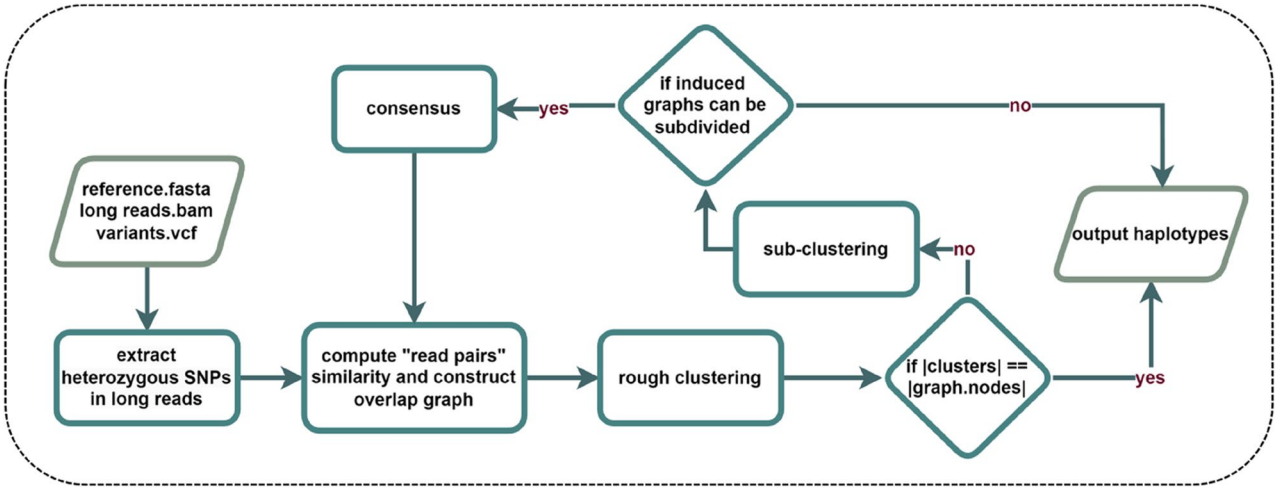


**Fig. 1** The flowchart of nTChap algorithm

## Results

### Phasing metrics

We benchmarked nTChap against WhatsHap polyphase [13] v2.3, nPhase [14] v1.2.0, flopp [15] v0.2.0. We ran all methods for their default settings with 16 threads, with inputs starting from VCF and BAM files. We compared the SNPs of the predicted results against the ground truth SNPs of haplotypes and got the best-matchings between predicted haplotigs (i.e., sequences of phased SNPs) and ground truth haplotypes for evaluation. We used the following metrics for benchmarking:

Accuracy: the percentage of all SNPs that were correctly attributed to their haplotypes. We calculated accuracy as: accuracy = TP / (TP + FP + FN), where TP = true positive; the number of SNPs is attributed to the correct haplotypes. FP = false positive; the number of SNPs does not belong to this haplotype. FN = false negative; the number of SNPs is not represented in the results.

Error rate: the percentage of all SNPs which erroneously attributed to the wrong haplotypes.

Missing rate: the percentage of all SNPs that present in the ground truth haplotypes but not present in the predicted results.

Reads clustering accuracy: the percentage of reads which were correctly attributed to their haplotypes.

nHaplotigs per haplotype: the predicted number of haplotigs divided by the true number of haplotypes.

### Evaluation on simulated potato chr1 dataset

We took the first 3.5 Mb of chromosome 1 of the v4.04 assembly of *Solanum tuberosum* produced by the Potato Genome Sequencing Consortium [17] and removed all "N"s, leaving about 3.02 Mb as operations in flopp [15]. We used the 3.02 Mb sequence of chromosome 1 as a reference and Haplogenerator [18] to simulate haplotypes and bi-allelic SNPs from the reference. We simulated haplotypes with five heterozygosity levels (3%, 1%, 0.3%, 0.1%, 0.05%). To assess the impact of ploidy, we simulated varying ploidy ranging from 3n to 12n. The ground truth is the simulated haplotypes. Next, we simulated reads with 10x, 15x and 20x coverages per haplotype. We used Badread [19] version 0.4.1 to simulate Oxford Nanopore Technology (ONT) reads and mapped the long reads to the reference using minimap2 [20]. To reduce running time, we used LoFreq [21], a fast and sensitive variant-caller for inferring SNVs and indels from next-generation sequencing data, for variant calling on long reads with different coverages and mutation rates. Although LoFreq was originally developed for high-throughput sequencing reads, LoFreq has subsequently been applied successfully to Oxford Nanopore Technology reads for variant calling [22–24]. Since LoFreq can only capture variant positions, we generate VCF files by integrating these variant positions with the simulated bi-allelic SNP files, which equates to sampling on the simulated bi-allelic SNP files with different heterozygosity levels based on read coverage. The distribution histogram of SNPs for simulated potato chr1 datasets (Supplementary Fig. 3) reveals that the simulated SNPs along the chromosome are non-uniform. Overall, it resulted in 5 (heterozygosity) * 3 (coverages) * 10 (ploidy) = 150 datasets. We compared the nTChap to WhatsHap polyphase, nPhase and flopp using various phasing metrics.

#### *Comparing four methods across different ploidy levels*

To evaluate the performance of four methods across different ploidy levels, as an example, we chose the outputs of the four methods under simulated data with 20x coverages and heterozygosity levels of 1%, 0.1% and 0.05% (Fig. 2). Additional results for different ploidy levels in other coverage depths and heterozygosity levels are available in the Supplementary Fig. 1.

On the 1% heterozygosity, 20x dataset (Fig. 2a), across ploidy levels 3n-12n, nTChap and flopp had the highest mean accuracy (99.96%), the lowest error rates (0.04%) and missing rates (0%). nTChap had the highest mean reads clustering accuracy (99.11% versus 97.34% for flopp, the second highest). While WhatsHap polyphase had 0% error rates in certain low-ploidy scenarios, its error rates increased, and its accuracy and reads clustering accuracy declined significantly after 8n. nTChap and flopp output 1 haplotigs per haplotype.

On the 0.1% heterozygosity, 20x dataset (Fig. 2b), nTChap had the highest mean accuracy (97.98%) versus the second highest WhatsHap polyphase (95.33%). nTChap had the lowest mean error rates (1.12%) versus the second lowest WhatsHap polyphase (3.26%), and had the lowest mean missing rates (0.9%) versus the second lowest WhatsHap polyphase (1.42%). WhatsHap polyphase had the highest mean reads clustering accuracy (91.58%) versus the second highest nTChap (89.8%). nTChap output mean of 23.22 haplotigs per haplotype versus the second least WhatsHap polyphase mean of 20.2, which is comparable. The first least output haplotigs per haplotype was flopp, as the number of haplotigs output by flopp is always the same as the number of haplotypes.

On the 0.05% heterozygosity, 20x dataset (Fig. 2c), nTChap had the highest mean accuracy (93.8%) versus the second highest WhatsHap polyphase (88.27%). nTChap had the lowest mean error rates (2.17%) versus the second lowest WhatsHap polyphase (3.73%), and had the lowest mean missing rates (4.45%) versus the second lowest WhatsHap polyphase (8%). nPhase output mean of 16.07 haplotigs per haplotype versus nTChap 27.23 versus WhatsHap polyphase 69.2. We observed that after the increasing point (ploidy = 7) where the reads clustering accuracy of WhatsHap polyphase exceeded other tools, the number of haplotigs of WhatsHap polyphase

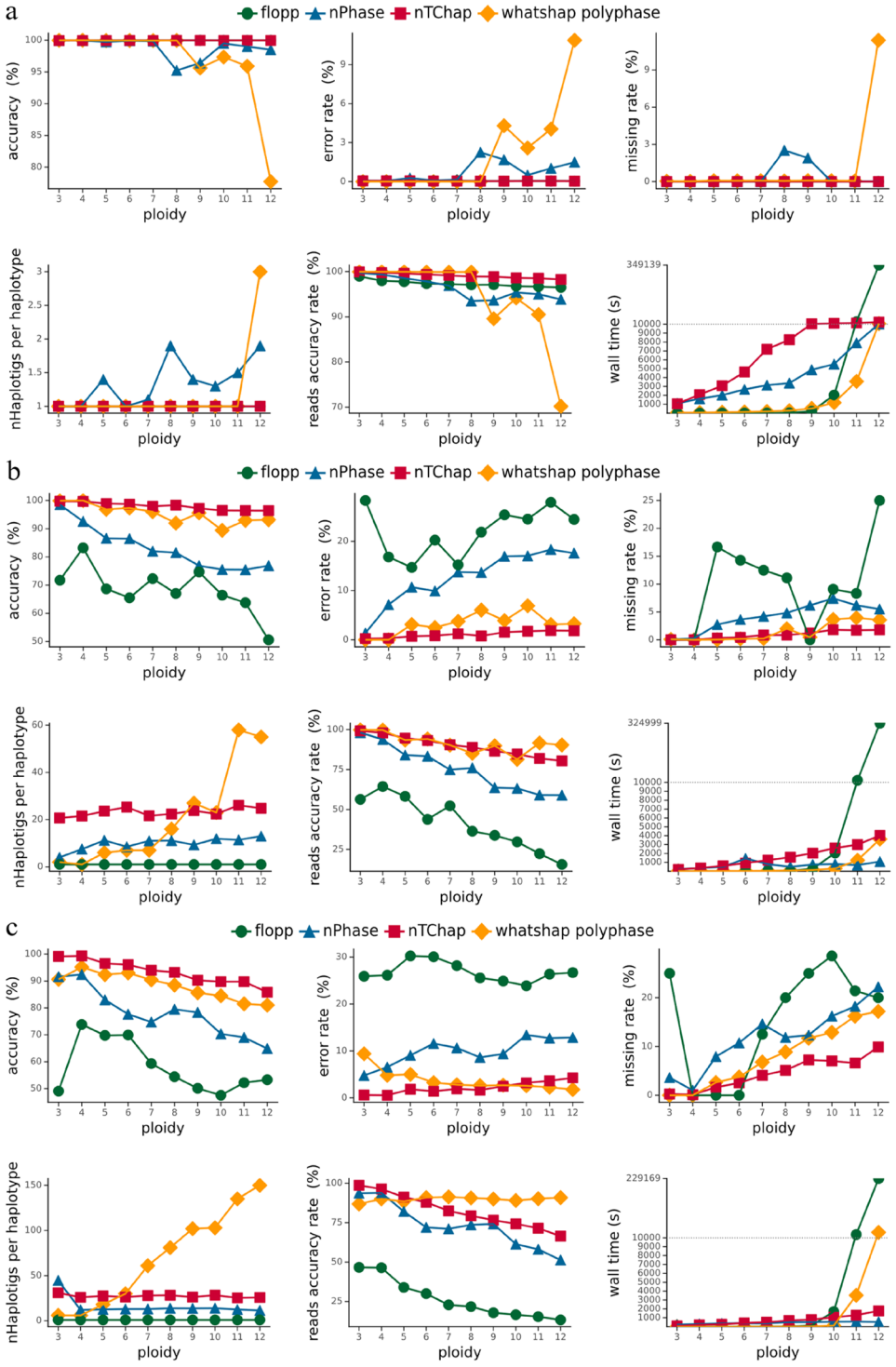


**Fig. 2** Effects of ploidy for various metrics on simulated potato chr1 datasets. **a** Various statistics for four methods on 1% heterozygosity and 20x coverage datasets of ploidy ranging from 3 to 12. **b** Various statistics for four methods on 0.1% heterozygosity and 20x coverage datasets of ploidy ranging from 3 to 12. **c** Various statistics for four methods on 0.05% heterozygosity and 20x coverage datasets of ploidy ranging from 3 to 12

also increased significantly. This might be due to the smaller heterozygosity, and as the ploidy increased, the output of WhatsHap polyphase was more fragmented, which means there are fewer reads in haplotigs, resulting in an increase in reads clustering accuracy.

With the increase of ploidy, the effect of each method decreases. The sensitivity of WhatsHap polyphase to ploidy becomes more pronounced with higher ploidy levels. WhatsHap polyphase shows relatively poorer performance in high-ploidy data compared to low-ploidy cases, suggesting it may be more suitable for moderately low-ploidy data. Compared with the other three methods, our method maintains stable and better performance in different ploidies. Notably, flopp demonstrates a significant increase in runtime when ploidy reaches 11.

### *Comparing four methods in different heterozygosity*

To assess the influence of heterozygosity, we divided the 150 datasets into 5 groups based on heterozygosity levels, with 30 datasets per heterozygosity and tested all four methods (Fig. 3). As shown in Fig. 3, with decreasing heterozygosity, the performance of flopp reduced faster, followed by nPhase. Flopp and nPhase were relatively more sensitive to heterozygosity, especially flopp, and it may be more suitable for data with high heterozygosity. At higher heterozygosity levels (3% and 1%), all four methods achieved good performance, with relatively good accuracy and reads clustering accuracy, low error rate and missing rate, and a mean number of haplotigs per haplotype of no more than 2.5.

At 0.3% heterozygosity, nTChap had higher accuracy (mean of 99.58%) than WhatsHap polyphase (mean of 93.69%). At 0.1% heterozygosity, nTChap had higher accuracy (mean of 97.07%) than WhatsHap polyphase (mean of 92.84%). At 0.05% heterozygosity, nTChap had higher accuracy (mean of 91.41%) than WhatsHap polyphase (mean of 86.21%). As heterozygosity decreases, the accuracy of all methods shows declining trends. When heterozygosity dropped below 0.3%, the accuracy of flopp decreased fastest, followed by nPhase. Regarding the error rates, missing rates, and reads clustering accuracy, their changing trends are almost consistent with accuracy (Fig. 3).

With the decrease of heterozygosity, the haplotigs output by each method except flopp (the number of haplotigs output by flopp is always consistent with that of ploidy) were increasing. While nTChap had more haplotigs at some heterozygosity, it produced substantially more accuracy and reads clustering accuracy, and fewer errors and missing errors, which is similar to WhatsHap polyphase. This indicates that flopp and nPhase may be too aggressive in separating haplotypes, which results in chimeric clusters. Additional detailed analyses can be found in the "Details of comparing four methods in

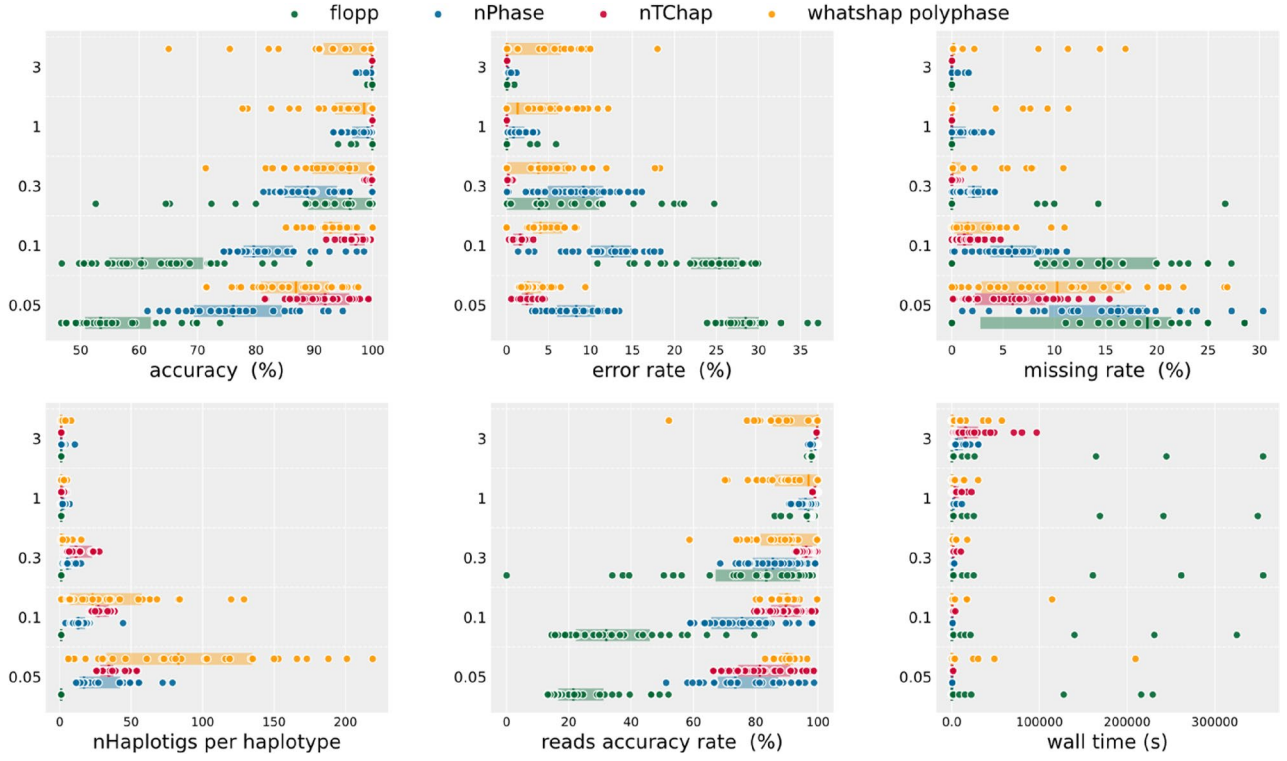


**Fig. 3** Effects of heterozygosity levels for various metrics on simulated potato chr1 datasets. Various statistics for heterozygosity levels of 0.05%, 0.1%, 0.3%, 1.0% and 3.0% datasets. Boxes are showing median, top and bottom quartiles; all individual data points shown with dots

different heterozygosity" section of the Supplementary Materials.

### *Comparing four methods in different coverage*

To assess the performance of four methods in different coverage, we chose 10x, 15x and 20x data, and every coverage with 50 datasets (Fig. 4). With the decrease of coverage, the performance of all four methods showed a decline, though not a significant one, which may be due to the limited range of coverage we chose. Considering the dataset size and computational efficiency, we did not choose larger coverage. Even at relatively low coverage, nTChap maintained higher accuracy and lower errors. When coverage reduced to 10x, nTChap still achieved stable and accurate.

At coverage 20x, 15x, and 10x, nTChap had better mean accuracy (98.23% vs. 94.87% for WhatsHap polyphase, the second best; 97.87% vs. 93.73% for WhatsHap polyphase; 96.63% vs. 89.05% for WhatsHap polyphase, respectively). Regarding the error rates, and missing rates, the changing trends are almost consistent with accuracy (Fig. 4). For reads clustering accuracy, nTChap had better mean reads clustering accuracy at coverage 20x and 15x (vs. WhatsHap polyphase, the second best). At coverage 10x, nTChap had better reads clustering accuracy (93.11%vs 87.77% for nPhase, the second best). The number of haplotigs of nTChap might be a little more, which means it was more careful to cluster to ensure the contiguity while guaranteeing the accuracy and completeness of the results. Additional detailed analyses can be found in the "Details of comparing four methods in different coverage" section of the Supplementary Materials.

### Evaluation on simulated *Saccharomyces cerevisiae* dataset

We simulated *Saccharomyces cerevisiae* S288C genomes following the workflow of nPhase. We selected four natural S. cerevisiae isolates: ACA, a haploid strain, and three homozygous diploid strains: BMB, CCN and CRL. We simulated triploid (3n; ACA, BMB, CCN) and tetraploid (4n; ACA, BMB, CCN, CRL) strains and, then performed subsampling on the Oxford Nanopore Technology (ONT) long reads and Illumina short reads to obtain reads with varying heterozygosity rates (0.01%, 0.05%, 0.1%, and 0.5%) and 10x and 20x coverages, and in total, we got 2(ploidy) * 4(heterozygosity) * 2(coverage) = 16 datasets. We used ngmlr [25] and BWA mem [26, 27] for ONT reads alignment and illumina reads alignment, and we ran GATK [28], a widely used toolkit for variant discovery and genotyping from next-generation sequencing (NGS) data [29], to generate a vcf file. The distribution histogram of SNPs for simulated Saccharomyces cerevisiae datasets (Supplementary Fig. 4) reveals that the simulated SNPs exhibit a non-uniform distribution.

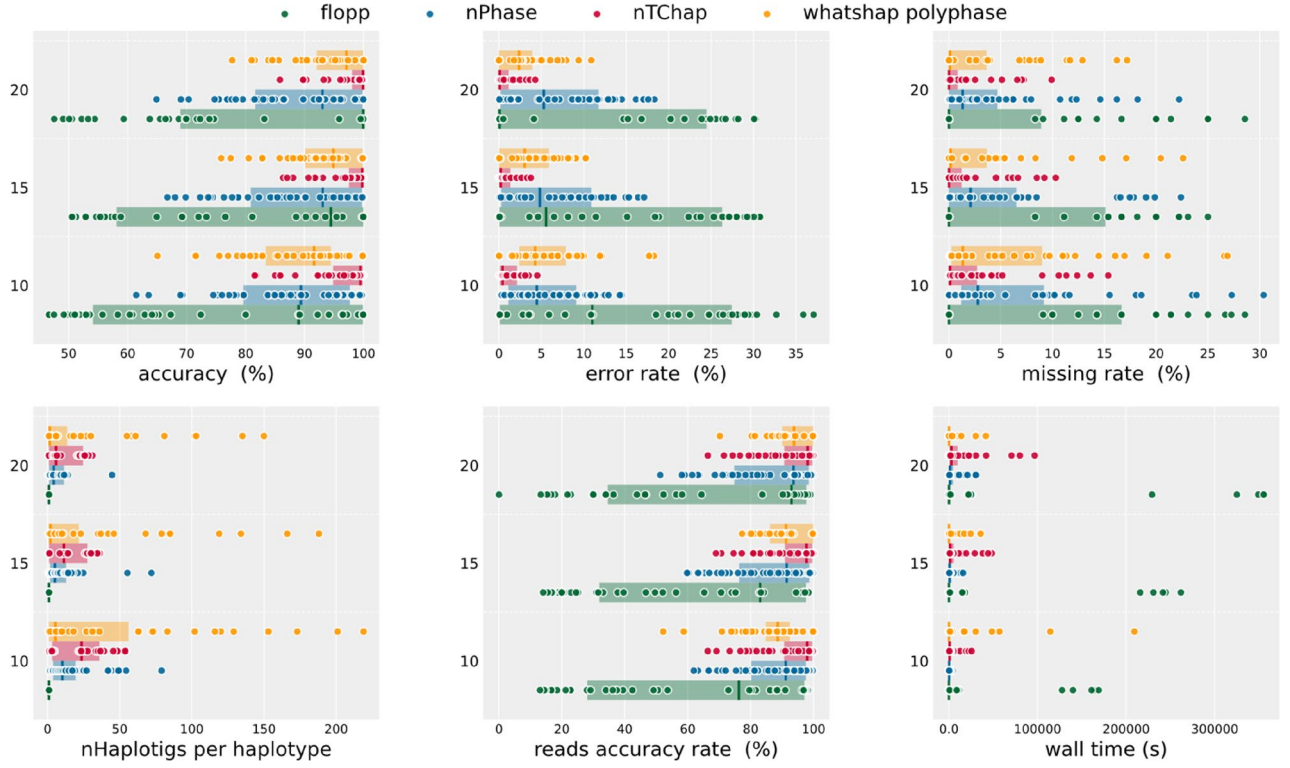


**Fig. 4** Effects of coverage for various metrics on simulated potato chr1 datasets. Various statistics for 10x, 15x and 20x coverage datasets. Boxes are showing median, top and bottom quartiles; all individual data points shown with dots

The ground truth, consistent with that in nPhase, was the SNPs individually called and subsampled from single-strain illumina sequencing data.

We compared the phasing performance of nTChap, WhatsHap polyphase, nPhase, and flopp on these simulated datasets (Fig. 5). nTChap had higher accuracy (mean of 87.5%) than WhatsHap polyphase (mean of 86.72%), nPhase (mean of 83.95%) and flopp (mean of 67.34%). nTChap had a lower error (mean of 3.34%) than WhatsHap polyphase (mean of 4.18%), nPhase (mean of 5.92%) and flopp (mean of 23.17%). WhatsHap polyphase had lower missing errors (mean of 9.1%) than nTChap (mean of 9.16%), flopp (mean of 9.49%) and nPhase (mean of 10.13%). We observed that four methods showed closely overlapping trends with no significant differences in missing rates plot. For reads clustering accuracy, nTChap was better (mean of 94.09%) than WhatsHap polyphase (mean of 92.44%), nPhase (mean of 89.26%) and flopp (mean of 31.22%). Flopp had the least haplotigs (mean of 1 haplotigs per haplotype) than nPhase (mean of 4.34), nTChap (mean of 8.51) and WhatsHap polyphase (mean of 64.25). Although flopp had better contiguity compared to other methods, its accuracy and error performance is comparatively worser. nTChap generally sustains superiority over other

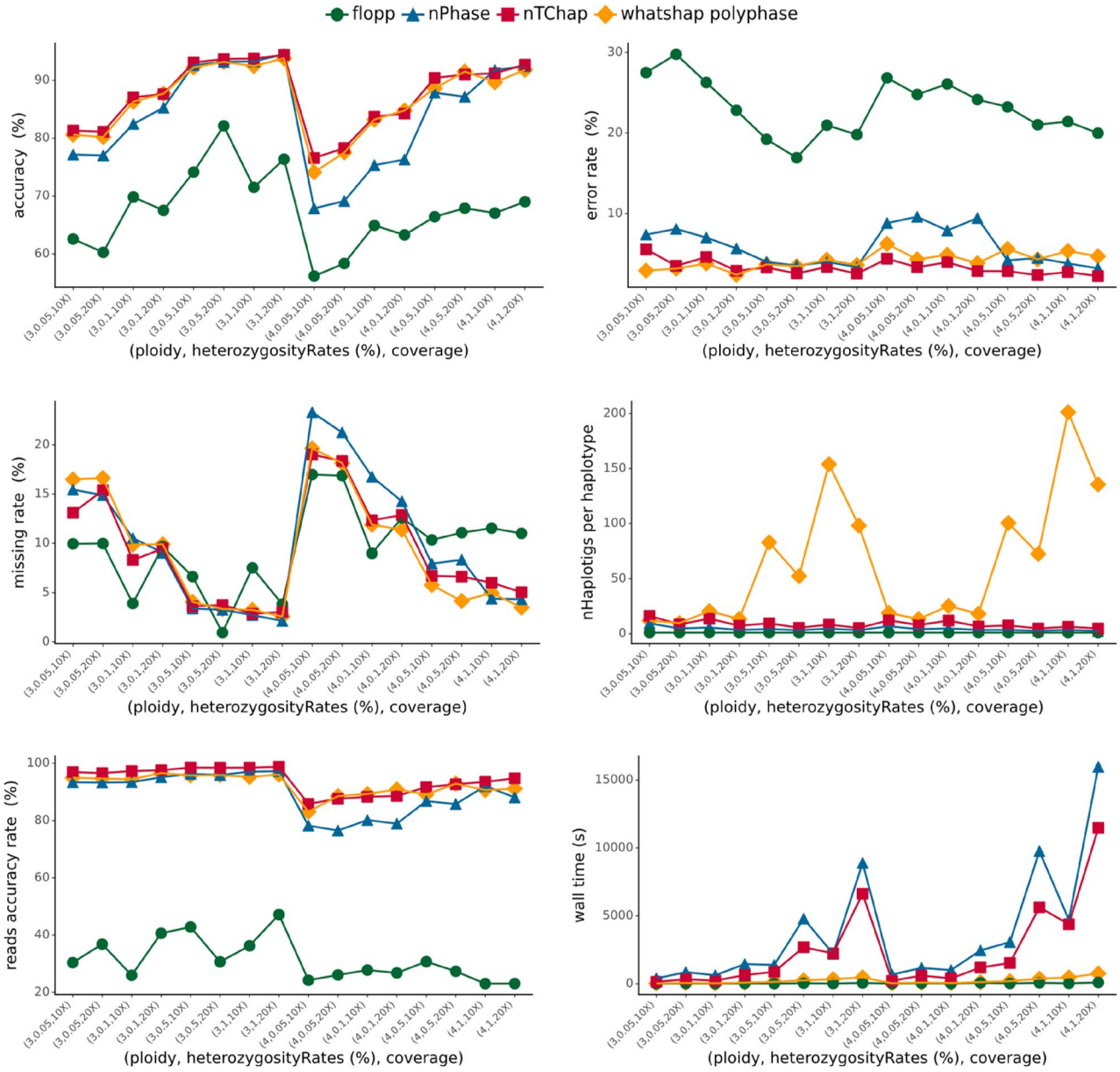


**Fig. 5** Benchmarking long-read haplotype phasing methods on simulated *Saccharomyces cerevisiae* datasets. The x-axis indicates an individual dataset categorized by (ploidy, heterozygosity level (%), coverage)

methods in accuracy, error rates and reads clustering accuracy and it has comparable contiguity as shown in Fig. 5.

#### *Comparisons of running time and memory usage*

We show the running time of nTChap, WhatsHap polyphase, nPhase and Flopp with 16 threads on a 2.0 GHz Intel(R) Xeon(R) Platinum server. For the simulated potato chromosome 1 datasets, on average, nTChap took about 109 min, while nPhase took about 43 min, WhatsHap polyphase took about 85 min and flopp took about 428 min. We observed a dramatic increase in runtimes of floop and WhatsHap polyphase (Supplementary Fig. 1) when the simulated ploidy reached 12. Therefore, we excluded the simulated datasets with ploidy 12 and recalculated the runtime for each method. nTChap took about 92 min, on average, while nPhase took about 38 min, WhatsHap polyphase took about 18 min and flopp took about 34 min. For the simulated potato chromosome 1 datasets, on average, flopp took about 7.00GB of memory, WhatsHap polyphase took about 11.59GB of memory, nTChap took about 24.61GB of memory, and nPhase took about 85.92GB of memory (Supplementary Fig. 1). For the simulated *Saccharomyces cerevisiae* datasets, on average, nTChap took about 40.9 min with a peak memory of about 22.79GB, while nPhase took about 61.9 min with a peak memory of about 153.63GB, WhatsHap polyphase took about 3.7 min with a peak memory of about 1.85GB and flopp took about 0.7 min with a peak memory of about 1.24GB (Supplementary Fig. 2). These statistics implied that nTChap might not be the most efficient, but it could be quite acceptable for practical use, and we are currently working on an optimized implementation.

### Validation on real *Brettanomyces bruxellensis* sequencing dataset

We tested nTChap on GB54 dataset [14], a triploid *Brettanomyces bruxellensis* strain of the yeast species. ASM1107488v2 [30] was chosen as the reference. We used minimap2 and BWA mem for real Oxford Nanopore Technology reads alignment and illumina reads alignment, and we ran GATK to generate a VCF file from illumina reads bam file. We filtered out heterozygous SNP sites from the VCF file and got a heterozygous SNP VCF file with a heterozygosity rate of 1.2%. Chromosome NC_054684.1 had the lowest heterozygous SNP rate (0.74%), with a length of 0.59 MB (Fig. 6a). Chromosome NC_054690.1 was the longest chromosome (3.89 MB), with 1.29% heterozygous SNP rate.

Since GB54 is a triploid, we expect a phasing result that three haplotypes fully cover each genomic region. Among 162,373 heterozygous SNP sites, 99.91% were spanned by haplotigs, with 98,557 sites exactly covered by three clusters (Fig. 6b). In our results (Fig. 6b, c), we observed that most regions were covered by three haplotypes and nTChap output 7 haplotigs per haplotype, which appeared to be somewhat fragmented. Therefore, we conducted a simple post-processing process similar to the cleaning process in nPhase (Supplementary Note 1). After testing post-processing on nTChap results with GB54, the number of haplotigs per haplotype decreased to 3.5 (a reduction of nearly half), with improved contiguity (Supplementary Fig. 8 (a), (b)). 99.24% of heterozygous SNP sites were spanned by haplotigs, a 0.67% reduction compared to pre-processing. 111,251 heterozygous SNP sites are exactly covered by three clusters, a 12.88% increase.

### Validation on chromosome 2 of real potato (*Solanum tuberosum*) sequencing dataset

We tested flopp, nPhase, nTChap and WhatsHap polyphase on real and larger tetraploid potato (*Solanum tuberosum*) genomes. We chose the chromosome-scale, haplotype-resolved genome assembly of the cultivated autotetraploid potato, Cooperation-88 (C88) [31], as the ground truth. The corresponding PacBio high-fidelity (HiFi) and Oxford Nanopore Technology (ONT) sequencing data for C88 were subsequently downloaded. We chose to call variants using the HiFi data, as the phase accuracy for C88 was assessed through the phased blocks, generated using WhatsHap polyphase on ONT alignments with variants called from HiFi data [31]. First, the HiFi reads were mapped to the C88 assembly by minimap2, retaining exclusively primary alignments mapped to chromosomes, which allowed us to determine which chromosome and haplotype each HiFi read comes from. Similarly, the ONT reads were mapped to the C88 assembly by minimap2, and we can determine which chromosome and haplotype each ONT read comes from. Subsequently, the HiFi reads for each haplotype were aligned to the reference genome, and variant calling was performed using FreeBayes [32] with the --ploidy 2 for each haplotype to obtain haplotype-specific variants, which served as the ground truth.

We used the DM 1–3 516 R44 v6.1 assembly [33] as the reference genome. Subsequently, the HiFi reads were mapped to the reference using minimap2, and variants were called from HiFi alignments using FreeBayes. To reduce computation time, we followed the same process in nPhase, specifically, we randomly sampled the raw ONT reads to 20x coverage and phased at a 0.5% heterozygosity level. The raw ONT reads were mapped to the reference by minimap2. As with flopp, nPhase, and WhatsHap polyphase, where they were all tested on chromosome 2 of the potato, we also test on chromosome 2. Tests were conducted on two datasets: HiFi (variant calling) + ONT (20x) (Table 1 (a)) and HiFi (variant calling) + HiFi (32x) (Table 1 (b)).

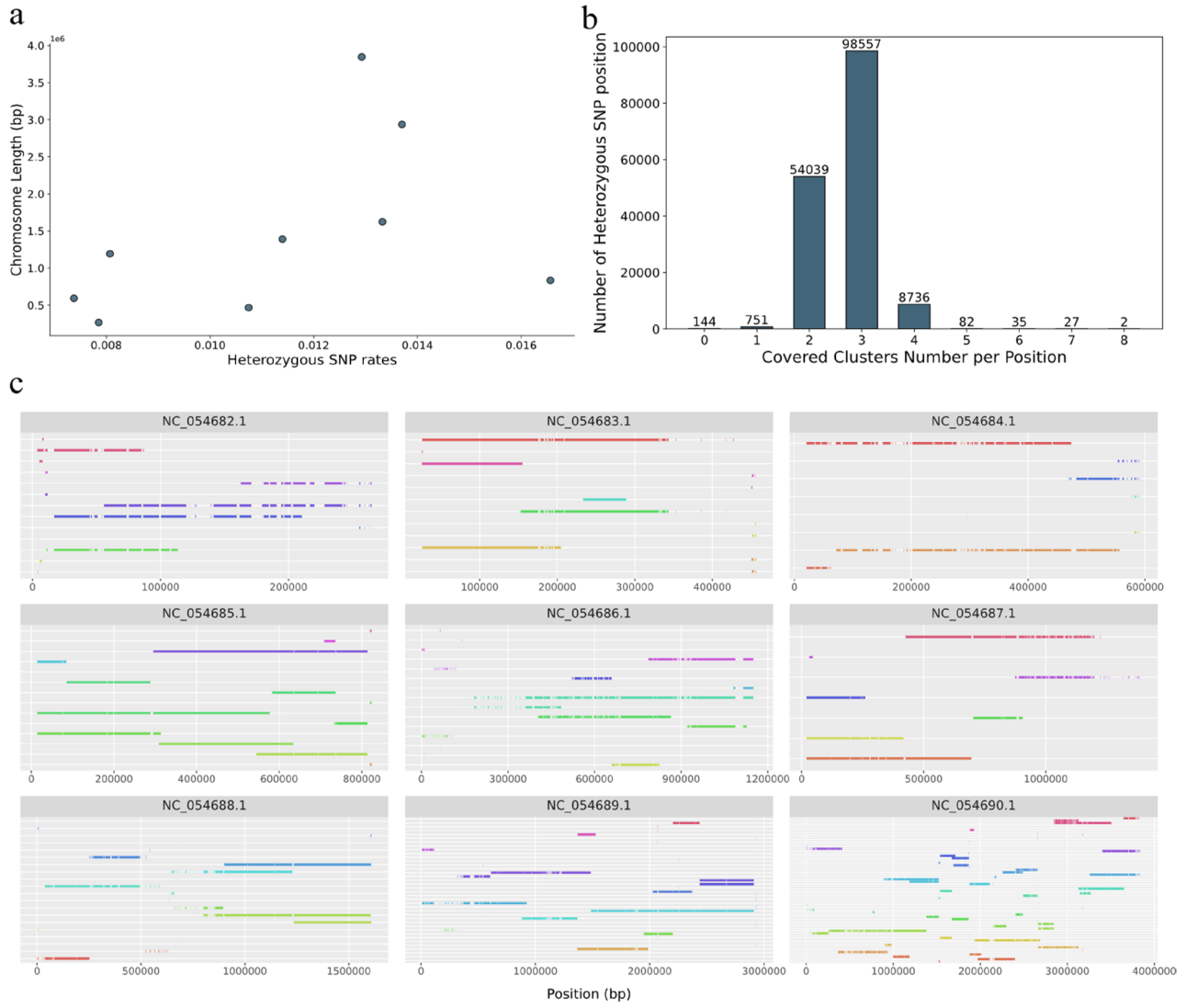


**Fig. 6** Phasing of GB54 genome. **a** The chromosome length and heterozygous SNP rate per chromosome of GB54 genome. **b** Distribution of phased heterozygous SNP number covered by clusters. Each bar displays the number of phased heterozygous SNP positions covered by *x* clusters (i.e., *x* haplotypes) and the *x*-axis displays cluster number. **c** Predicted haplotypes for the GB54 genome. Each subgraph displays the predicted haplotypes for a different chromosome. The *x*-axis displays the position along the chromosome and the lines with different colors on the *y*-axis are predicted haplotypes. All predicted haplotypes are color coded randomly as the ground truth is not known

**Table 1** Comparison of flopp, nPhase, nTChap and WhatsHap polyphase on chromosome 2 of the potato with datasets HiFi (variant calling) + ONT (20x) (a) and HiFi (variant calling) + HiFi (32x) (b). ER: error rate (%); MR: missing rate (%); RAR: reads accuracy rate (%); #Contigs / Ploidy: number of predicted haplotigs / number of truth haplotypes; WH-PP: WhatsHap polyphase

| Methods | Accuracy | ER | MR | RAR | #Contigs/Ploidy | Time (h) | Memory (GB) |
|---|---|---|---|---|---|---|---|
| (a) ONT | | | | | | | |
| nTChap | 58.35 | 24.35 | 17.30 | 90.51 | 2407/4 | 35.17 | 27.11 |
| flopp | 25.93 | 48.10 | 25.97 | 30.91 | 4/4 | 0.04 | 2.45 |
| nPhase | 41.02 | 41.14 | 17.84 | 47.94 | 1477/4 | 19.84 | 57.67 |
| WH-PP | 63.38 | 25.67 | 10.95 | 81.92 | 112,176/4 | 11.50 | 2.33 |
| (b) HiFi | | | | | | | |
| nTChap | 63.83 | 18.46 | 17.71 | 94.13 | 3728/4 | 30.83 | 36.04 |
| flopp | 26.76 | 48.11 | 25.13 | 28.24 | 4/4 | 0.05 | 2.42 |
| nPhase | 63.01 | 24.07 | 12.92 | 80.44 | 3679/4 | 18.98 | 48.95 |
| WH-PP | 69.04 | 21.89 | 9.07 | 81.67 | 111,436/4 | 8.62 | 3.31 |

Table 1 shows all used evaluation metrics on flopp, nPhase, nTChap and WhatsHap polyphase for their default settings. nTChap had the lowest error rate on both datasets, and nTChap had the highest reads clustering accuracy, reaching a 90.51% reads clustering accuracy on the ONT dataset, exceeding the second-best by 10.5%, and reaching a 94.13% on the HiFi dataset, exceeding the second-best by 15.3%. While the accuracy of nTChap (58.35%) on the ONT dataset was 5.03% lower than WhatsHap polyphase (63.38%), nTChap (2407) produced approximately 46.6 times fewer contigs than WhatsHap polyphase (112176). Similarly, on the HiFi dataset, nTChap's accuracy was 5.21% lower than WhatsHap polyphase, however, it generated approximately 30 times fewer contigs than WhatsHap polyphase. Compared to nPhase (41.02%, 1477), nTChap produced 1.6 times as many contigs but significantly improved accuracy by 17.33% on the ONT dataset. Flopp had the fastest speed and the lowest memory usage on both datasets. Our approach can be applied to larger genomes in practice, though further optimization is needed.

A good haplotype prediction must possess both high accuracy and high contiguity [34]. High accuracy can be achieved by highly fragmented predictions because such strategies avoid correctly linking distant SNPs. Conversely, aggressively clustering reads or SNPs to maximize contiguity risks creating chimeric haplotypes and false joins, which fail to resolve true haplotypes. Maintaining both high accuracy and high contiguity simultaneously remains a significant challenge, one not adequately addressed by the current popular phasing algorithms.

## Discussion

In this work, we present a method called nTChap for generating polyploid haplotypes from long reads sequencing data. nTChap builds an overlap graph from the reduced reads composed of variants and then resolves haplotypes through iterative two-stage clustering, without the need for explicit ploidy information. The existing clustering algorithms tend to discover compact structures within local, small-scale regions in a vertical manner. Converting clusters into consensus sequences as new "reduced reads" increases the length of the "reduced reads", enabling them to connect to more distant "reduced reads" and thereby allowing the clusters to extend horizontally. We could vertically find read clusters belonging to each haplotype and horizontally extend these clusters to generate longer and more broadly covered haplotigs. The two rounds of clustering algorithms are designed to ensure that each cluster originates as much as possible from the same haplotype, thereby avoiding the accumulation of errors in subsequent consensus steps.

Recent advances in long-read sequencing technologies have significantly advanced the development of strain-level metagenomic phasing. The metagenomic phasing and polyploid phasing share conceptual parallels -- both seek to resolve mixtures of haplotypes (or strains) from mixed sequencing reads by leveraging long-read information to connect variants across large genomic regions. To some extent, both problems can be formulated as clustering tasks. However, Polyploid genome phasing differs from metagenomic phasing in several fundamental aspects. For instance, polyploid genomes are typically larger than bacterial genomes and polyploid genomes have a lower variant density compared to the frequent differences between bacterial strains. In metagenomic phasing, the number of strains is often unknown, and coverage usually varies between strains. Strainy [35], a tool for phasing bacterial strains from long-read sequencing data, also first applies a label propagation algorithm for clustering, then iteratively refines the clusters into sub-clusters by recursively removing non-informative SNPs in each cluster per iteration and then re-running the label propagation algorithm in each cluster. These sub-clusters are subsequently assembled using Flye [36], after which a graph is constructed and traversed to resolve the final strain haplotypes. However, we iteratively partition the clusters into sub-clusters by merging the most similar sequences within a cluster and constraining this merging process: the merged sequence must be highly similar to the two sequences prior to merging. This constraint specifically prevents merging sequences originating from distinct haplotypes. Then these sub-clusters are treated as new "reduced reads" and subjected to a new round of clustering and subdivision. This process continues until final clusters become indivisible during either of the two steps. In each iteration, the "reduced reads" progressively increase the length, enabling to connect to more distant "reads". Our method is more suitable for polyploid phasing.

We test nTChap on 150 simulated datasets based on the 3.02 Mb sequence of chromosome 1 of *Solanum tuberosum* and 16 simulated datasets based on *Saccharomyces cerevisiae* S288C. We observe that nTChap presents higher accuracy, lower error rates, lower missing rates, and higher read clustering accuracy. All methods have a better performance in higher coverage or higher heterozygosity level genomes, while nTChap performs remarkably well in low coverage and low heterozygosity genomes. Moreover, nTChap works well and is stable in high-ploidy genomes. However, haplotypes output by nTChap are fragmented since it clusters nodes more cautiously, avoiding error merging without enough information. Besides, in genomic regions with uneven variant distribution, the overlap graph can be sparse, which may lead to cluster fragmentation or partial misclustering. In genomic regions with highly repetitive sequences, the computational distinction between variations arising

from repeats and true haplotype differences can become less clear, which may lead to misclustering. We believe that addressing these challenges would likely benefit from future advancements in sequencing technology, particularly the availability of longer and more accurate reads. Additionally, we are actively working on refining the method to better handle these complex scenarios. Although we ensured the same input for all tools, the use of only bi-allelic SNPs in our simulated *Solanum tuberosum* experiments may lead to a profound underestimation of real-world data complexity. To mitigate this limitation, we additionally conducted simulations of *Saccharomyces cerevisiae* S288C and applied our method to real potato datasets. Since nTChap employs parallelism only during the initial construction of the read similarity overlap graph, but parallelization was not utilized in the main clustering process that is most time-consuming, it exhibits slightly longer runtimes. We will later explore ways to refine.

The main limitation of nTChap is that it is a reference-based method, which greatly depends on reference genomes, and the quality of alignment and the subsequent variant calling. The reference-based approaches can lead to biases, for example, reference genomes may not be available for certain polyploid organisms. De novo haplotype assembly generates more comprehensive polyploid genomes and can produce novel sequences. Recent breakthroughs in sequencing technologies, particularly PacBio HiFi [37] and ultra-long nanopore [38], are accelerating advances in polyploid de novo haplotype assembly. Currently, the de novo haplotype assembly strategy that is commonly used for phasing plant genomes firstly assembles and phases initial allelic contigs by allele-aware algorithms (such as Hifiasm [39, 40] and HiCanu [41]), with subsequent Hi-C scaffolding (such as ALLHiC [42], 3D-DNA [43] and HapHiC [44]), achieving chromosome-level haplotype-resolved assemblies [45]. For example, the haplotype-resolved genome of the tea plant (*Camellia sinensis*) was successfully assembled using the above strategy [46]. Reference-based haplotype phasing can be used to complement de novo assembly. Combining reference-based and de novo approaches will harness complementary strengths and gain a better understanding of genetic diversity.

## Methods

Here, we describe nTChap algorithm, which takes a reference genome, a long-reads bam file aligned to the reference using minimap2 and a VCF file containing information on genetic variations generated using high-accuracy short reads or the long-reads bam file as input. nTChap utilizes an iterative process (Fig. 1): building an overlap graph, applying two-round clustering, and generating a consensus sequence for each cluster, and it then outputs the phased SNP sequences and the clustered long reads for each haplotype, without prior knowledge of ploidies.

### Generating reduced reads and building overlap graphs

Given a set of all reads aligned to the reference, nTChap represents each aligned read as a set of heterozygous SNPs (Fig. 7a). It records the SNPs positions of the reference that the reads are aligned to and the allele bases that the reads contain. It skips SNPs in a read if either the SNP site has a deletion in the read or the read's base at the SNP site is neither an alternate nor reference allele in the VCF.

After reducing all long reads to sets of SNPs, nTChap calculates pairwise similarity scores between all these reduced reads. The reduced read pair similarity is defined as similarity = (number of same bases in overlap variants positions) / (number of overlap variants positions). nTChap builds an overlap graph. Each node corresponds to a reduced read, and two nodes are connected with an undirected edge if the similarity of the two corresponding reduced reads is at least min_sim (default 0.3). The edge weights are the similarity between the corresponding reads.

### Two rounds of clustering

nTChap is performed iteratively. It performs two rounds of clustering in each iteration on the overlap graph (Fig. 7b). First, the graph is partitioned using the label propagation algorithm [16], a community detection algorithm. Due to the local identity between polyploid haplotypes and the sequencing errors, reads from distinct haplotypes might be connected in the overlap graph, which can lead to reads from distinct haplotypes being erroneously grouped into the same cluster. We performed raw clustering (i.e., the label propagation algorithm) on the simulated potato chr1 datasets (with various ploidy and heterozygosity data under 10x and 20x coverage) (Supplementary Fig. 5) and the simulated Saccharomyces cerevisiae datasets (Supplementary Fig. 6) with the same parameters as nTChap. The results indicate that using only raw clustering causes reads from different haplotypes to be clustered together, failing to effectively differentiate between distinct haplotypes. Therefore, we perform intra-cluster refinement individually on each cluster generated in the community detection algorithm.

For each cluster, nTChap extracts its corresponding induced subgraph from the overlap graph (Fig. 8). An induced subgraph is defined as a subgraph of a graph, formed by a subset of the vertices of the graph together with any edges whose endpoints are both in this subset. First, the edges of the induced graph are sorted in decreasing order of their weights. The node pair with the maximum edge weight is chosen to compute their

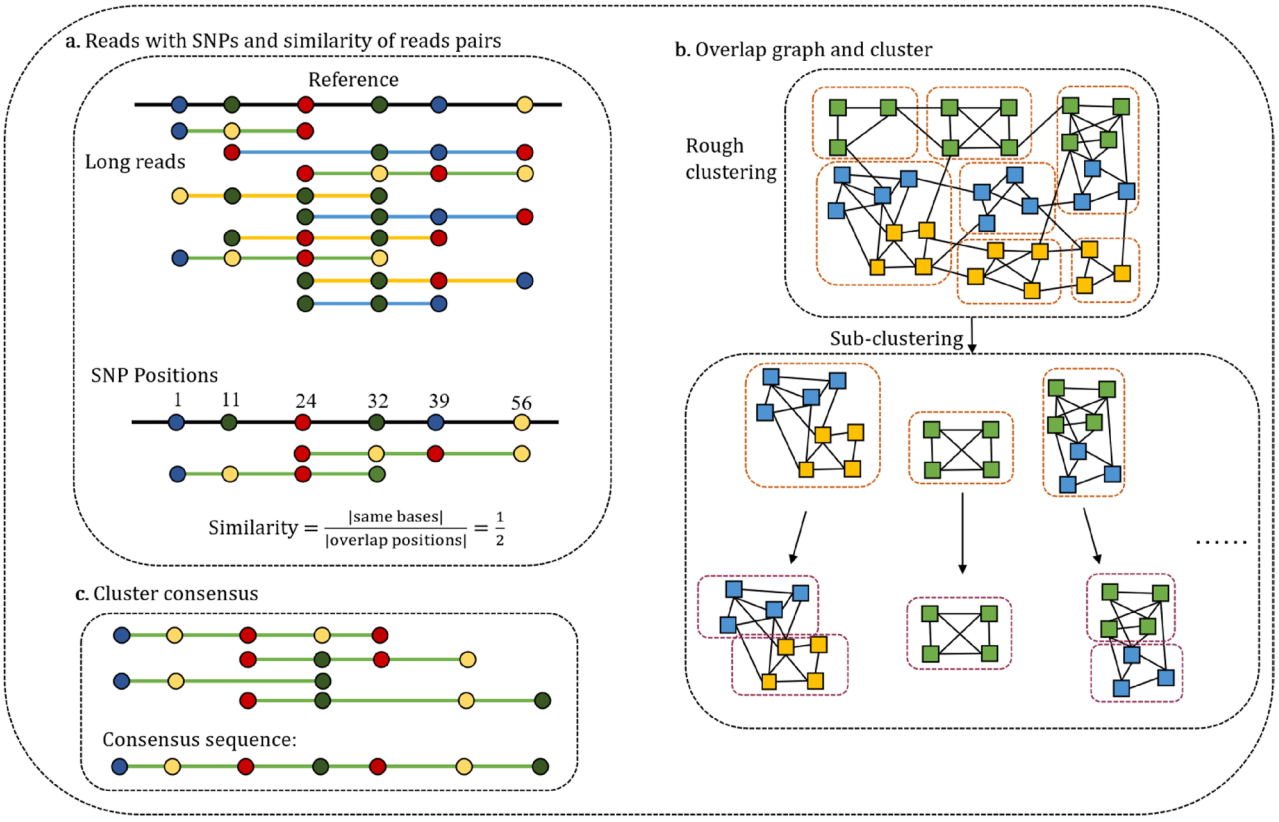


**Fig. 7** **a** Reads that are aligned to a reference are converted to a set of SNPs named as a reduced read. The colored dots are SNP bases. The similarity between reduced reads is defined as the number of same SNPs divided by the number of overlapped SNPs positions. **b** The overlap graph is built by reduced reads and iterative two rounds of clustering are used to cluster the graph nodes. The colored boxes are nodes (i.e., reduced reads). **c** Each cluster is converted to a consensus, generated by allowing every read in the cluster to vote for a specific base for a given position

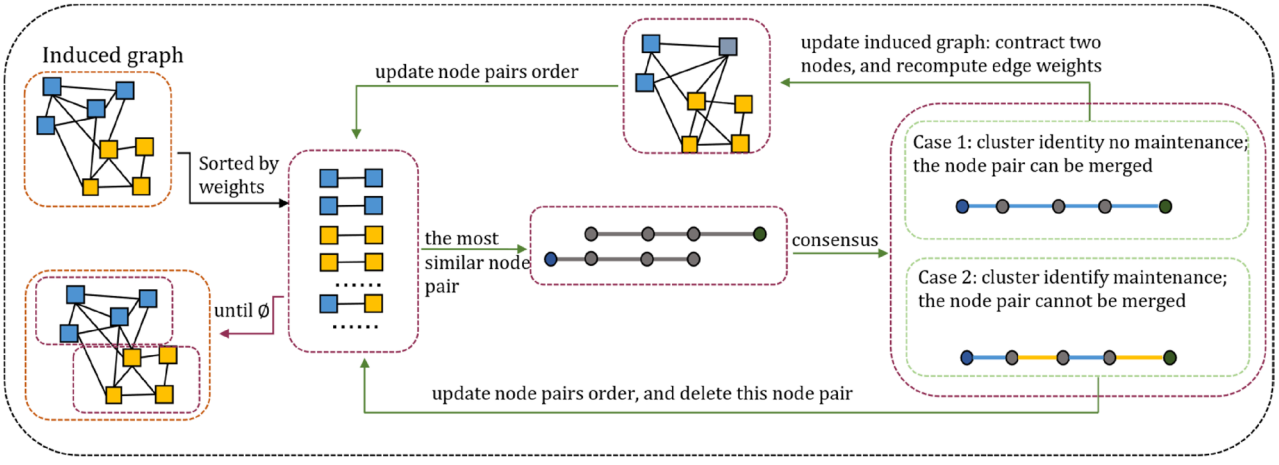


**Fig. 8** The flowchart of sub-clustering. The colored boxes are nodes of an induced graph and the dots are SNP bases

consensus sequence (Fig. 7c). Consensus sequence is defined as the most support base by the reduced reads in two nodes at each variant position.

nTChap uses the "cluster identity" inspired by nPhase to check whether two nodes can be merged. It considers that each node has its own "identity" defined by the population of reduced reads that comprise it. If merging two nodes results in increased support for their consensus sequence bases, then that change is considered positive, otherwise, that change is considered negative. It then calculates how many votes every base lost. The number of votes lost is divided by the total number of votes in the region that both nodes have in common to obtain the cluster identity change percentage. If the change value

does not exceed max_id (default 0.05), "cluster identity" is deemed unpreserved, and the two nodes can be merged. nTChap contracts the two nodes into a new node in the induced graph, recalculates similarity scores between the new node and all adjacent nodes and updates the ordering of node pairs (Fig. 8 case 1). If the change value exceeds the max_id threshold, "cluster identity" is preserved, and the two nodes cannot be merged (Fig. 8 case 2). nTChap removes this node pair from the ordering set. The sub-clustering algorithm iterates the node merging and graph updating procedures until all remaining node pairs exhibit sufficient difference. The remaining nodes are the subclusters of the previous cluster, generated in the first round of clustering.

### Cluster consensus

Upon completing two rounds of clustering on each iteration, each remaining node corresponds to a cluster, containing several reduced reads. nTChap then computes a consensus sequence for each node of reduced reads (Fig. 7c). These consensus sequences are then served as new reduced reads to reconstruct an overlap graph, participating in the next iteration.

## Supplementary Information

The online version contains supplementary material available at https://doi.org/10.1186/s12864-025-12440-w.

Supplementary Material 1

**Acknowledgements**
Not applicable.

**Authors' contributions**
Y.G., T.Y., J.Q. conceived and designed the experiments. Y.G. performed the experiments. Y.G. contributed reagents/materials/analysis tools. Y.G. wrote the paper. Y.G. and T.Y. designed the software used in the analysis. G.L. and T.Y. oversaw the project.

**Funding**
This work was supported by the National Key R&D Program of China with code 2020YFA0712400; the Fundamental Research Funds for the Central Universities; the National Natural Science Foundation of China with codes 12471461 and 11931008. The funders had no role in the study design, data collection and analysis, decision to publish, or preparation of the manuscript.

**Data availability**
All scripts needed to handle the input data and to reproduce the analyses, and the partial genomic sequence of potato chromosome 1 for read simulation can be found at https://github.com/gaoyun-25/nTChap-test. Illumina data for the *Saccharomyces cerevisiae* strains is available with the SRA accessions [47]: ERR1308732, ERR1309429, ERR1308952, and ERR1308675. Oxford Nanopore data for the *Saccharomyces cerevisiae* strains is available with the SRA accessions [14]: ERR4352153, ERR4352154, ERR4352155, ERR4352156. Illumina and Oxford Nanopore data for the *Brettanomyces bruxellensis* GB54 strain is available under the study accession number PRJEB40511 [14]. Oxford Nanopore data for the *Solanum tuberosum* is available at NGDC Genome Read Archive with accession number SAMC586267 [31] and PacBio HiFi data for the *Solanum tuberosum* is available at NGDC Genome Read Archive with accession number SAMC586266 [31].

**Code Availability**
The source code for the latest version of the nTChap package is available at https://github.com/gaoyun-25/nTChap.

## Declarations

**Ethics approval and consent to participate**
Not applicable.

**Consent for publication**
Not applicable.

**Competing interests**
The authors declare no competing interests.

Received: 3 August 2025 / Accepted: 11 December 2025
Published online: 13 January 2026

**References**

1. Sattler MC, Carvalho CR, Clarindo WR. The polyploidy and its key role in plant breeding. Planta. 2015;243(2):281–96.
2. Soltis PS, et al. Polyploidy and genome evolution in plants. Curr Opin Genet Dev. 2015;35:119–25.
3. Wang Y, et al. *Sequencing and assembly of polyploid Genomes*, in *Polyploidy: methods and protocols*. New York, NY: Springer US; 2023. pp. 429–58. Y. Van de Peer, Editor.
4. Browning SR, Browning BL. Haplotype phasing: existing methods and new developments. Nat Rev Genet. 2011;12(10):703–14.
5. Brinton J et al. A haplotype-led approach to increase the precision of wheat breeding. Commun Biology, 2020. 3(1):712.
6. Garg S. Computational methods for chromosome-scale haplotype reconstruction. Genome Biol, 2021. 22(1):101.
7. Van de Peer Y, Mizrachi E, Marchal K. The evolutionary significance of polyploidy. Nat Rev Genet. 2017;18(7):411–24.
8. Aguiar D, Istrail S. Haplotype assembly in polyploid genomes and identical by descent shared tracts. Bioinformatics. 2013;29(13):i352–60.
9. Aguiar D, Istrail S. HapCompass: A fast cycle basis algorithm for accurate haplotype assembly of sequence data. J Comput Biol. 2012;19(6):577–90.
10. Berger E, et al. HapTree: A novel bayesian framework for single individual polyplotyping using NGS data. PLoS Comput Biol. 2014;10(3):e1003502.
11. Das S, Vikalo H. SDhaP: haplotype assembly for diploids and polyploids via semi-definite programming. BMC Genomics. 2015;16(1):260.
12. Xie M, et al. H-PoP and H-PoPG: heuristic partitioning algorithms for single individual haplotyping of polyploids. Bioinformatics. 2016;32(24):3735–44.
13. Schrinner SD, et al. Haplotype threading: accurate polyploid phasing from long reads. Genome Biol. 2020;21(1):252.
14. Abou Saada O, et al. nPhase: an accurate and contiguous phasing method for polyploids. Genome Biol. 2021;22(1):126.
15. Shaw J, Yu YW. Flopp: extremely fast Long-Read polyploid haplotype phasing by uniform tree partitioning. J Comput Biol. 2022;29(2):195–211.
16. Raghavan UN, Albert R, Kumara S. Near linear time algorithm to detect community structures in large-scale networks. Phys Rev E. 2007;76(3):036106.
17. Hardigan MA, et al. Genome reduction uncovers a large dispensable genome and adaptive role for copy number variation in asexually propagated solanum tuberosum. Plant Cell. 2016;28(2):388–405.
18. Motazedi E, et al. Exploiting next-generation sequencing to solve the haplotyping puzzle in polyploids: a simulation study. Brief Bioinform. 2018;19(3):387–403.
19. Wick RR. Badread: simulation of error-prone long reads. J Open Source Softw, 2019;4(36) :1316.
20. Li H. Minimap2: pairwise alignment for nucleotide sequences. Bioinformatics. 2018;34(18):3094–100.
21. Wilm A, et al. LoFreq: a sequence-quality aware, ultra-sensitive variant caller for Uncovering cell-population heterogeneity from high-throughput sequencing datasets. Nucleic Acids Res. 2012;40(22):11189–201.
22. González-Recio O, et al. Sequencing of SARS-CoV-2 genome using different nanopore chemistries. Appl Microbiol Biotechnol. 2021;105(8):3225–34.

23. Liu Y, et al. Rescuing low frequency variants within intra-host viral populations directly from Oxford nanopore sequencing data. Nat Commun. 2022;13(1):1321.
24. Barbé L et al. SARS-CoV-2 Whole-Genome Sequencing Using Oxford Nanopore Technology for Variant Monitoring in Wastewaters. 2022;13–2022.
25. Sedlazeck FJ, et al. Accurate detection of complex structural variations using single-molecule sequencing. Nat Methods. 2018;15(6):461–8.
26. Li H, Durbin R. Fast and accurate short read alignment with Burrows–Wheeler transform. Bioinformatics. 2009;25(14):1754–60.
27. Li H. Aligning sequence reads, clone sequences and assembly contigs with BWA-MEM. arXiv, 2013. https://doi.org/10.48550/arXiv.1303.3997.
28. Van der Auwera GA, et al. From FastQ data to High-Confidence variant calls: the genome analysis toolkit best practices pipeline. Curr Protocols Bioinf. 2013;43(1):11101–111033.
29. Brouard J-S, et al. The GATK joint genotyping workflow is appropriate for calling variants in RNA-seq experiments. J Anim Sci Biotechnol. 2019;10(1):44.
30. Roach MJ, Borneman AR. New genome assemblies reveal patterns of domestication and adaptation across brettanomyces (Dekkera) species. BMC Genomics. 2020;21(1):194.
31. Bao Z, et al. Genome architecture and tetrasomic inheritance of autotetraploid potato. Mol Plant. 2022;15(7):1211–26.
32. Garrison, E. and G. Marth, Haplotype-based variant detection from short-read sequencing. Preprint at arXiv, 2012. https://doi.org/10.48550/arXiv.1207.3907.
33. Pham GM, et al. Construction of a chromosome-scale long-read reference genome assembly for potato. GigaScience. 2020;9(9):giaa100.
34. Saada OA, Friedrich A, Schacherer J. Towards accurate, contiguous and complete alignment-based polyploid phasing algorithms. Genomics. 2022;114(3):110369.
35. Kazantseva E, et al. Strainy: phasing and assembly of strain haplotypes from long-read metagenome sequencing. Nat Methods. 2024;21(11):2034–43.
36. Kolmogorov M, et al. Assembly of long, error-prone reads using repeat graphs. Nat Biotechnol. 2019;37(5):540–6.
37. Wenger AM, et al. Accurate circular consensus long-read sequencing improves variant detection and assembly of a human genome. Nat Biotechnol. 2019;37(10):1155–62.
38. Jain M, et al. Nanopore sequencing and assembly of a human genome with ultra-long reads. Nat Biotechnol. 2018;36(4):338–45.
39. Cheng H, et al. Haplotype-resolved de Novo assembly using phased assembly graphs with hifiasm. Nat Methods. 2021;18(2):170–5.
40. Cheng H, et al. Haplotype-resolved assembly of diploid genomes without parental data. Nat Biotechnol. 2022;40(9):1332–5.
41. Nurk S, et al. HiCanu: accurate assembly of segmental duplications, satellites, and allelic variants from high-fidelity long reads. Genome Res. 2020;30(9):1291–305.
42. Zhang X, et al. Assembly of allele-aware, chromosomal-scale autopolyploid genomes based on Hi-C data. Nat Plants. 2019;5(8):833–45.
43. Dudchenko O, et al. De Novo assembly of the Aedes aegypti genome using Hi-C yields chromosome-length scaffolds. Science. 2017;356(6333):92–5.
44. Zeng X, et al. Chromosome-level scaffolding of haplotype-resolved assemblies using Hi-C data without reference genomes. Nat Plants. 2024;10(8):1184–200.
45. Kong W, et al. Recent advances in assembly of complex plant genomes. Proteom Bioinf. 2023;21(3):427–39. Genomics.
46. Garg S, et al. A haplotype-aware de Novo assembly of related individuals using pedigree sequence graph. Bioinformatics. 2020;36(8):2385–92.
47. Peter J, et al. Genome evolution across 1,011 Saccharomyces cerevisiae isolates. Nature. 2018;556(7701):339–44.


## Publisher's note