Practical probabilistic and graphical formulations of long-read polyploid haplotype phasing

By: Jim Shaw and Yun William Yu

Motivation

- With the interest in determining the sequence of alleles on each specific chromosome and not just the presence of an allele within the genome or in other words to solve Haplotype phasing for polyploid genomes.

Keyword Definitions

Haplotype:
“A haplotype is a set of DNA variations, or polymorphisms, that tend to be inherited together. A haplotype can refer to a combination of alleles or to a set of single nucleotide polymorphisms (SNPs) found on the same chromosome. “[1]

Polyploid
“The polyploid cell or organism has three or more times the haploid chromosome number.”[2]

Keyword Definitions

Haplotype Phasing:
“Haplotype estimation (also known as "phasing") refers to the process of statistical estimation of haplotypes from genotype data.”[3]

MEC Score: “Minimum error correction(MEC) is a prominent computational problem for haplotype assembly and, given a set of fragments, aims at reconstructing the two haplotypes by applying the minimum number of base corrections.”[4]

Problem Formulation

- R is the set of all reads that align to a chromosome
- m is the number of variants in our chromosome.
- every read r_i is an element of the space $r_i \in \{-, 0, 1, 2, 3\}$
- $r_i[j]$ is the jth coordinate of r_i, $r_i[j] \in \{0, 1, 2, 3\}$ if the jth variant is contained in the read ri where 0 represents the reference allele, 1 represents the first alternative allele, and so forth. $r_i[j] = -$ if ri does not contain the jth allele.

Problem Formulation

For any two reads r1, r2 : d and s represents the number of different and same variants .

$$s(r_1, r_2) = |\{k : r_1[k] = r_2[k], (r_1[k] \neq -) \wedge (r_2[k] \neq -)\}|.$$

$$d(r_1, r_2) = |\{k : r_1[k] \neq r_2[k], (r_1[k] \neq -) \wedge (r_2[k] \neq -)\}|$$

Problem Formulation

Define the *consensus haplotype* $H(R_i) \in \{-, 0, 1, 2, 3\}^m$ associated to a subset of reads as follows. For all indices $l = 1, \ldots, m$ let $H(R_i)[l] = \arg\max_a |\{r \in R_i : r[l] = a\}|$ and break ties according to some arbitrary order. If only $-$ appear at position l over all reads, we take $H(R_i)[l] = -$. It is easy to check that $H(R_i)$ is a sequence in $\{-, 0, 1, 2, 3\}^m$ such that $H(R_i)[k] \neq -$ at indices for which some read overlaps, and $\sum_{r \in R_i} d(H(R_i), r)$ is minimized.

In our formalism, we can phrase the MEC model of haplotype phasing as the task of finding a partition $\{R_1, \ldots, R_k\}$ of R such that

$$\sum_{i=1}^{k} \sum_{r_j \in R_i} d(r_j, H(R_i))$$

Problem Formulation

Min-sum max tree partition (MSMTP) model. Let $G(R) = (R, E, w)$ be an undirected graph where the vertices are R and edges E are present between two reads r_1, r_2 if r_1, r_2 overlap, i.e. $d(r_1, r_2)+s(r_1, r_2) > 0$. Let the weight of $e = (r_1, r_2)$ be $w(e) = w(r_1, r_2)$ for some weight function w. We call $G(R)$ the *read-graph*; a similar notion is found in [17,18,19,26,27].

For a partition of R into disjoint subsets $\{R_1, \ldots, R_k\}$ we take $G(R_i)$ as defined above. We only consider partitions of vertices such that all $G(R_i)$ are connected, which we will denote as valid partitions. Let $MST(G)$ be the maximum spanning tree of a graph G. Define

$$SMTP_R^k(R_1, \ldots, R_k) = \sum_{i=1}^{k} \sum_{e \in MST(G(R_i))} w(e). \tag{1}$$

We formulate the *min-sum max tree partition (MSMTP) problem* as finding a valid partition $\{R_1, \ldots, R_k\}$ of R such that $SMTP_R^k(R_1, \ldots, R_k)$ is minimized.

The MSMTP problem falls under a class of problems called graph tree partition problems [28], most of which are NP-Hard. We give a proof that MSMTP is NP-Hard in Appendix A.

Intuitively, assuming each $G(R_i)$ is connected, a maximum spanning tree is a maximum measure of discordance along the entire haplotype. We prove below that under a specific constraint on the read-graph, the SMTP score for $w(r_1, r_2) = d(r_1, r_2)$ is an upper bound for the MEC score.

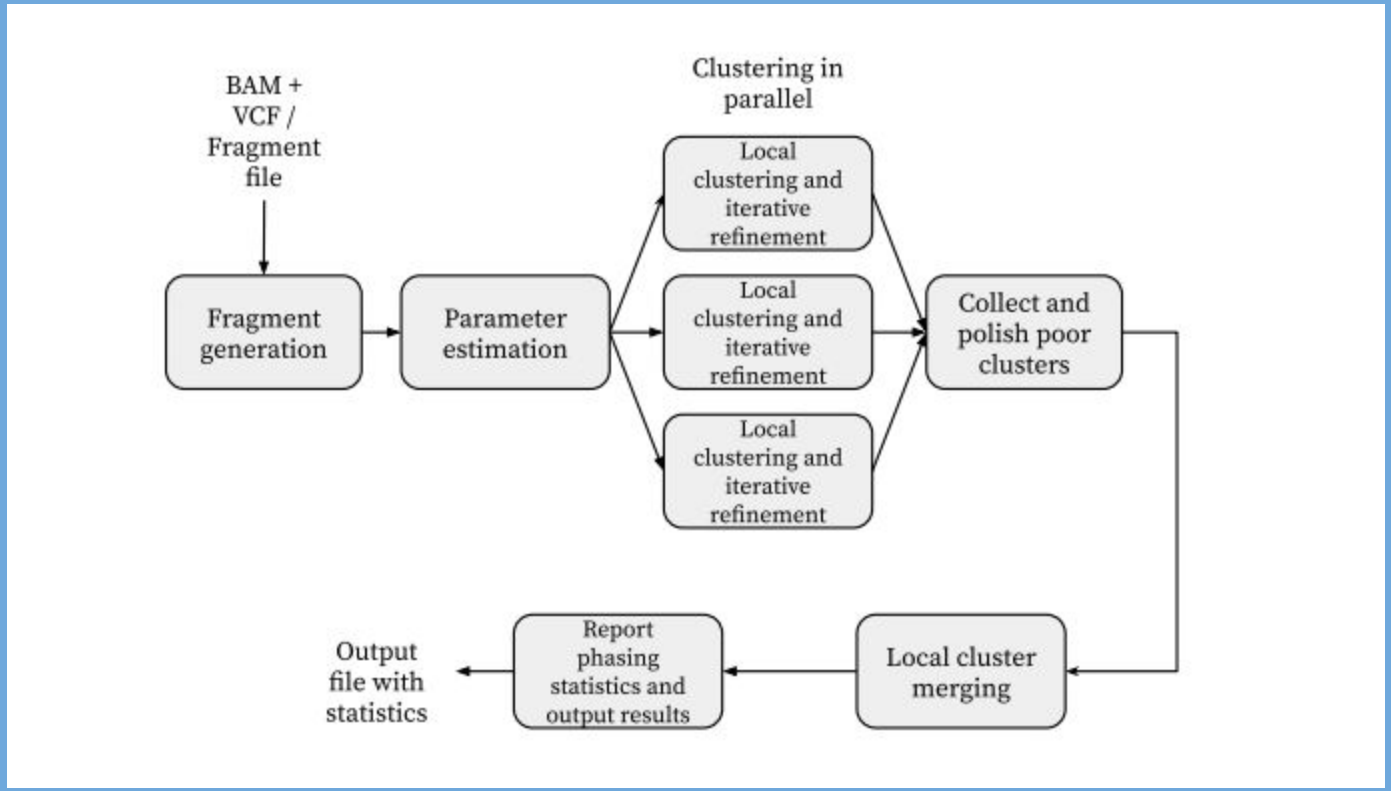
Problem Formulation

Let ϵ represent the probability that a variant is called incorrectly. Let $\sigma \in \mathbb{R}$ be a normalizing constant, and $X_i \sim Binomial(\lceil (D(R_i) + S(R_i))/\sigma \rceil, \epsilon)$ be a binomial random variable. Then

$$UPEM_R(R_1, \ldots, R_k) = \sum_{i=1}^{k} \log \left[\Pr \left(X_i > \left\lceil \frac{S(R_i)}{\sigma} \right\rceil \right) \right] + \log[\chi^2(|R_1|, \ldots, |R_k|)].$$

The $\chi^2(x_1, \ldots, x_n)$ term is the p-value for the χ^2 test while the binomial term is a sum of log one-sided binomial tests where the null hypothesis is that the error rate of a clustering is ϵ. Therefore the UPEM score is just a sum of log p-values.

WorkFlow of Flopp



MSMTP Algorithm

Algorithm 1: Greedy min-max read partitioning

Input : Read-graph $G(S)$, ploidy k, iterations n
Output: A partition $\{S_1, \ldots, S_k\}$ of S

$\{v_1, \ldots, v_k\} \leftarrow$FindMaxClique$(G(S), k)$
for $i = 1$ *to* k **do**
  $S_i \leftarrow \{v_i\}$
end
for $i = 1$ *to* n **do**
  $V \leftarrow G(S) \setminus \bigcup_{i=1}^{k} S_i$
  Reverse-sort V by assigning to $v \in V$ the value $v \rightarrow \min_{S_i \in \{S_1, \ldots, S_k\}} \max_{r \in S_i} s(r, v) + d(r, v)$
  $V \leftarrow V[: \lceil \frac{|V|}{n} \rceil]$
  for v *in* V **do**
    $S' \leftarrow \arg\min_{S_i} \max_{r \in S_i} w(v, r)$
    $S' \leftarrow S' \cup \{v\}$
  end
end
Return $\{S_1, \ldots, S_k\}$

Iterative refinement of local clusters:

- The algorithm checks how moving reads from one partition to another partition changes the UPEM score for every read.
- stores the best moves and execute a fraction of them.
- Proceed for n iterations or until the UPEM score does not improve anymore.

Local phasing procedure:

Local phasing procedure Note that Algorithms 1 and 2 work on subsets of reads or subgraphs of the underlying read-graph. Let $b \in \mathbb{N}$ be a constant representing the length of a local block. We consider subsets $B_1, \ldots, B_l \subset R$ where

$$B_i = \{r \in R : \exists j, b(i-1) \leq j \leq b(i), r[j] \neq -\}.$$

The subsets are just all reads that overlap a shifted interval of size b, similar to the work done in [31]. After choosing a suitable b, we run the read-partitioning and iterative refinement on all $B_1, \ldots, B_l$ to generate a set of partitions $P_1, \ldots, P_l$. We found that a suitable value of b is the $\frac{1}{3}-$quantile value of read lengths. By read length we mean the last non $'-'$ position minus the first non $'-'$ position of $r \in \{0, 1, 2, 3, -\}$.

Polishing, merging, and parameter estimation

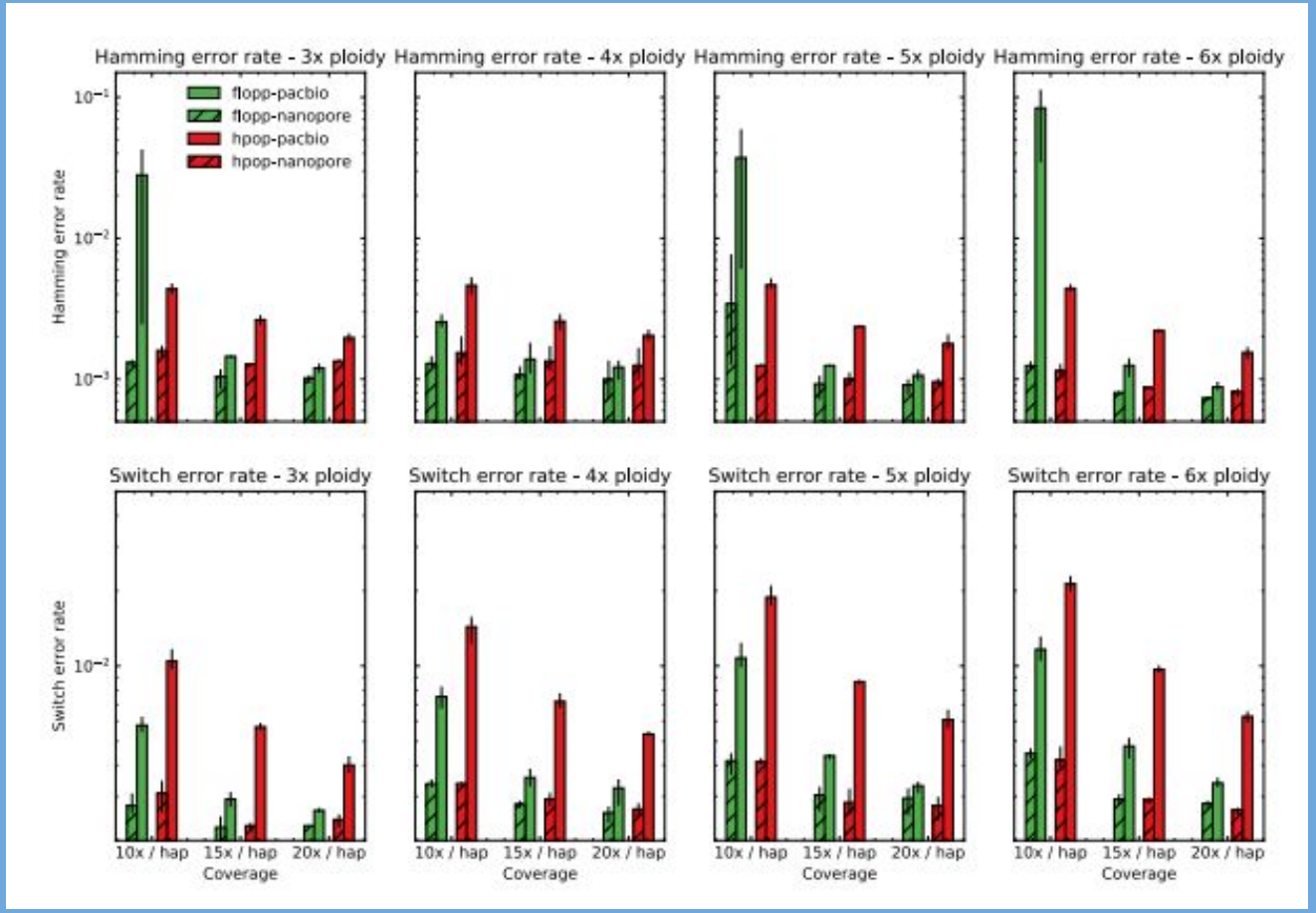
- Filling in erroneous blocks
- Local cluster merging
- Parameter estimation

Results and Discussion

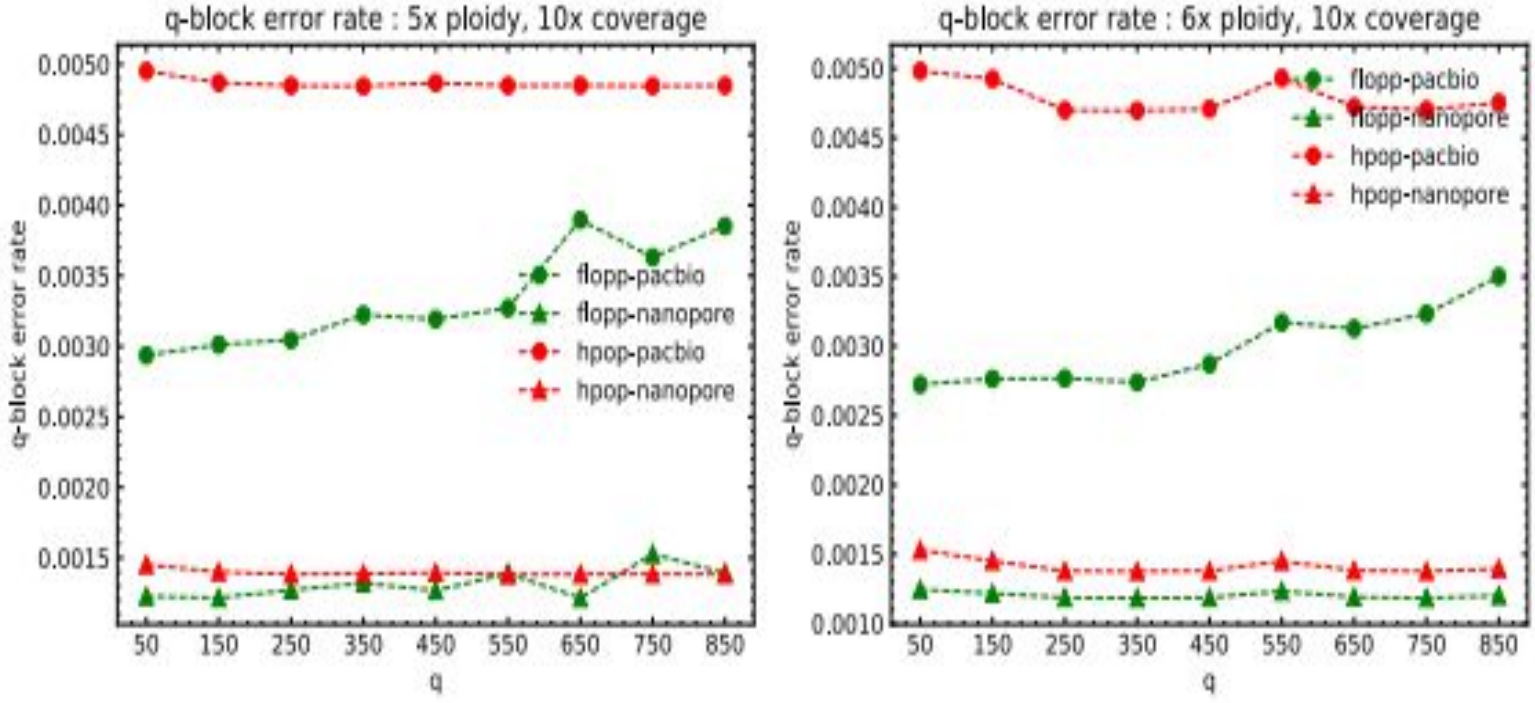
They have used three metrics to compare the results:

- Hamming error rate.
- switch error rate
- q-block error rate

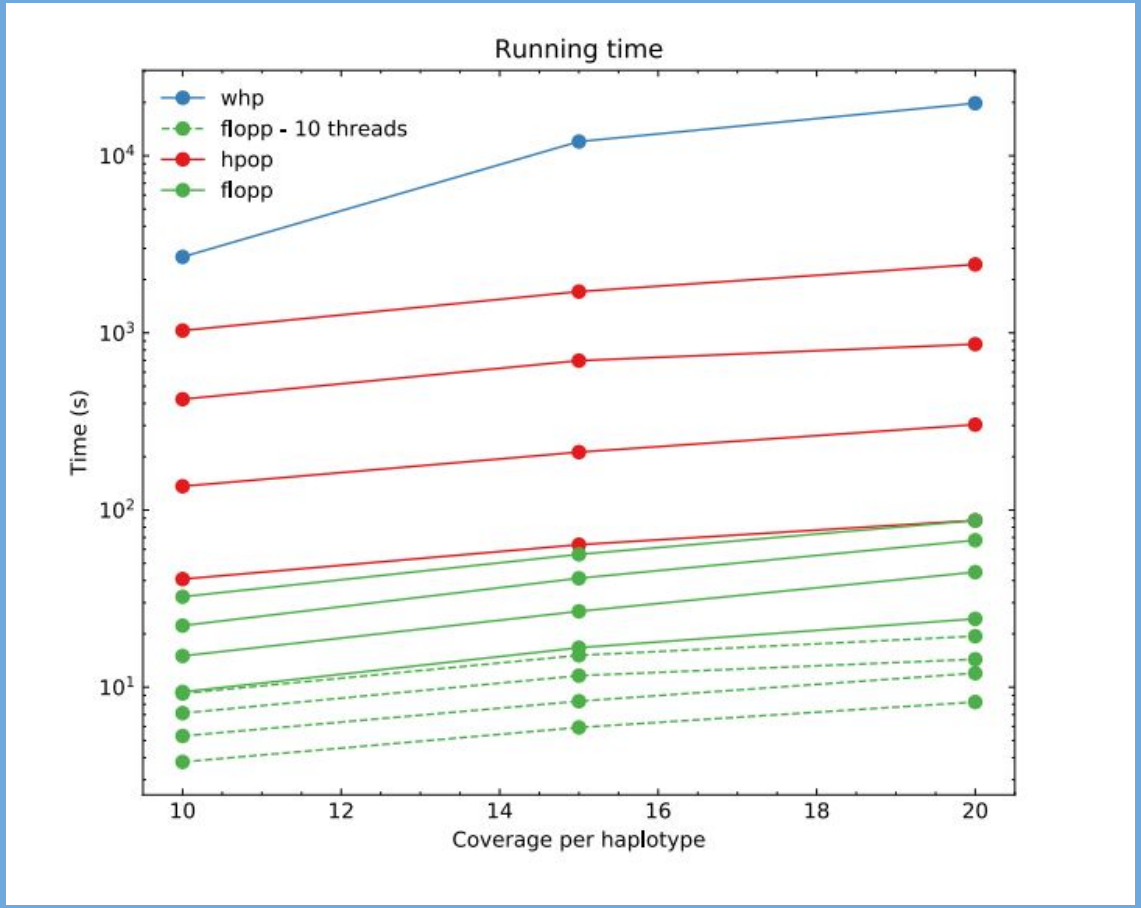
Comparison


Hamming error rate - 3x ploidy
Hamming error rate - 4x ploidy
Hamming error rate - 5x ploidy
Hamming error rate - 6x ploidy
flopp-pacbio
flopp-nanopore
hpop-pacbio
hpop-nanopore
Hamming error rate
Switch error rate - 3x ploidy
Switch error rate - 4x ploidy
Switch error rate - 5x ploidy
Switch error rate - 6x ploidy
Switch error rate
10x / hap
15x / hap
20x / hap
Coverage

Comparison


q-block error rate : 5x ploidy, 10x coverage
q-block error rate
0.0050
0.0045
0.0040
0.0035
0.0030
0.0025
0.0020
0.0015
50
150
250
350
450
550
650
750
850
q
flopp-pacbio
flopp-nanopore
hpop-pacbio
hpop-nanopore
q-block error rate : 6x ploidy, 10x coverage
q-block error rate
0.0050
0.0045
0.0040
0.0035
0.0030
0.0025
0.0020
0.0015
0.0010
50
150
250
350
450
550
650
750
850
q
flopp-pacbio
flopp-nanopore
hpop-pacbio
hpop-nanopore

Comparison


Running time
whp
flopp - 10 threads
hpop
flopp
Time (s)
10^4
10^3
10^2
10^1
10
12
14
16
18
20
Coverage per haplotype

Reference

1. https://www.genome.gov/genetics-glossary/haplotype
2. https://www.britannica.com/science/polyploidy
3. https://en.wikipedia.org/wiki/Haplotype_estimation#:~:text=In%20genetics%2C%20haplotype%20estimation%20(also,from%20a%20group%20of%20individuals.
4. Bonizzoni P, Dondi R, Klau GW, Pirola Y, Pisanti N, Zaccaria S. On the Minimum Error Correction Problem for Haplotype Assembly in Diploid and Polyploid Genomes. J Comput Biol. 2016 Sep;23(9):718-36. doi: 10.1089/cmb.2015.0220. Epub 2016 Jun 9. PMID: 27280382.