HAPLOTYPE INFERENCE FROM SHORT SEQUENCE READS USING A POPULATION GENEALOGICAL HISTORY MODEL

JIN ZHANG and YUFENG WU*

Department of Computer Science and Engineering
University of Connecticut
Storrs, CT 06269, U.S.A.
E-mail: {jinzhang,ywu}@engr.uconn.edu

High-throughput sequencing is currently a major transforming technology in biology. In this paper, we study a population genomics problem motivated by the newly available short reads data from high-throughput sequencing. In this problem, we are given *short* reads collected from individuals in a population. The objective is to infer haplotypes with the given reads. We first formulate the computational problem of haplotype inference with short reads. Based on a simple probabilistic model on short reads, we present a new approach of inferring haplotypes directly from given reads (i.e. without first calling genotypes). Our method is finding the most likely haplotypes whose local genealogical history can be approximately modeled as a perfect phylogeny. We show that the optimal haplotypes under this objective can be found for many data using integer linear programming for modest sized data when there is no recombination. We then develop a related heuristic method which can work with larger data, and also allows recombination. Simulation shows that the performance of our method is competitive against alternative approaches.

Keywords: High-throughput sequencing; haplotype inference; bioinformatics algorithms; population genomics.

Biocomputing 2011 Downloaded from www.worldscientific.com
by TEXAS STATE UNIVERSITY on 03/03/18. For personal use only.

1. Introduction

High throughput DNA sequencing is increasingly recognized as a major transforming technology in biology. During the last decade, several novel high throughput sequencing (HTS) technologies have been developed and commercialized (such as the Roche 454 FLX, Illumina Genome Analyzer, and ABI SOLiD), and several more are under development. These high throughput technologies dramatically bring down the sequencing cost and are generating huge amount of data. Several individual genomes have been sequenced,[1,2] and an effort is underway to sequence one thousand individuals.[3] Sequencing may give entire *diploid* genomes of individuals in a population and potentially reveal *all* the common (and many of the *rare*) variations in the sequenced region. Thus, increasingly complete sequencing using HTS technologies will become the preferred approach to attack population genomics problems.

On the other hand, the current HTS technologies have some technical *limitations*. First, the reads generated by HTS technologies are often *short*. Although longer sequence reads may become available in the near future,[4] it is expected that short sequence reads are likely to be still useful in the coming years. Thus, we focus on short reads in this paper. Second, many HTS technologies have higher error rates than the traditional Sanger sequencing. Some technologies have error rates of 1% or even higher, which can make it difficult to distinguish between error and population-scale variation. Additional error sources include inaccurate sequence

*Corresponding author.

reads mapping (i.e. locating the reads within a reference genome). Sometimes high coverage sequencing may reduce the noise and uncertainty, but with increased cost. Therefore, robust data analysis methods are needed to process the (somewhat noisy) HTS data.

In this paper, we focus on a population genomics problem: inference of a pair of haplotypes for each individual in the population from the given HTS reads for diploid organisms (such as human). Here, a haplotype refers to the DNA sequence collected from the same chromosome, which describes the alleles at polymorphic sites on this chromosome. Collecting haplotypes from populations is an important population genomics problem, which is evident in the HapMap project.[5,6] See Section 2 for more description on haplotypes. To formulate a concrete computational problem, we make several assumptions:

Biocomputing 2011 Downloaded from www.worldscientific.com
by TEXAS STATE UNIVERSITY on 03/03/18. For personal use only.

(1) In this paper, we only consider *short* reads. That is, our problem is different from the haplotype reconstruction problem based on *long* sequence reads (e.g. Bansal and Bafna,[7] He, et al.[8]). Since the sequence reads are short and often the variations in a population are relatively sparsely located along the genome, we assume that a short (single or paired-end) read covers no more than *one* SNP site. When there is a read covering more than one SNP sites, our current implementation treats this read as multiple reads, each covering one SNP, although our implementation can be modified to use the haplotype phase information contained in such reads.
(2) We do *not* consider pooling here: we know the individual a sequence read originates.
(3) In this paper, we only concern single nucleotide polymorphisms (SNPs), which can be stated as a binary value: 0 or 1.
(4) A standard analysis step in analyzing short reads is *mapping* the short reads against a reference genome (which we assume is available). We assume that reads mapping is performed properly so that reads covering one polymorphic site are properly mapped. We only consider reads that are uniquely mapped and remove reads that are ambiguous in mapping. Once the reads are mapped, we can identify polymorphic genomic positions by comparing the mapped reads with the reference genome. Thus, we assume that the SNP sites can be determined from the mapped reads.

We are now ready to define the precise problem formulation.

Haplotype Inference with Single Short Reads. We are given a set of mapped single short reads $\mathcal{R}$, each covering a specific SNP site. That is, a sequence read reports an allele at a polymorphic site for an individual, but we do not know which homologous chromosome it comes from and also there is some chance the allele reported is incorrect. The goal is inferring two haplotypes for each individuals from the reads $\mathcal{R}$.

Note that our method also calls genotypes: once haplotypes are inferred, we can obtain genotypes from the haplotypes. This problem formulation may be useful for (1) sequencing a new population, where no previously sampled population haplotypes (such as those provided by the HapMap project) are available, and (2) whole-genome sequencing, where we want to infer haplotypes for *all* SNPs (not only common SNPs but also *rare* SNPs). We note that rare variants are becoming more important in understanding genotype-phenotype association.[9]

2. Background

2.1. *Haplotypes and Genealogical History*

An important genetic variation is the single nucleotide polymorphism (SNP). A SNP site in the genome can generally take only two states (alleles) among the individuals in a population. Thus, we use binary alleles (0 and 1) to represent the state at any SNP site. In this paper, we focus on SNPs and do not consider other variations such as copy number variation (CNV) or polymorphism (CNP). Often, we collect genetic variations data at multiple genomic sites. We call a sequence of genetic variations at these sites a *haplotype*. A haplotype based on SNPs can be represented as a *binary* vector. A *diploid* organism (such as human) has two haplotypes per chromosome, and although these are often called 'copies', they are not identical. A description of the conflated (mixed) data from the two haplotypes is called a *genotype*. When both haplotypes have state 0 (resp. 1) at a site, the genotype has state 0 (resp. 2), and is called a *homozygote*. Otherwise, the genotype has state 1 at that site and is called a *heterozygote*. We let n be the number of individuals sampled in the population, and m be the number of SNP sites. The genotypes of these individuals are represented by an n by m matrix with entries 0/1/2, while their haplotypes are represented by a $2n$ by m binary matrix. We call the two ordered alleles from the two haplotypes at a single site of a diploid individual *diploid type*. Diploid type can be 0/0, 0/1, 1/0 and 1/1. Note that there are two diploid type (0/1 and 1/0) for the same genotype 1.

Genealogical history of sequences in a population explicitly shows the origin and derivation of extant sequences, the locations of all the genomic alterations (both in the genome and in time), and how the variants are transmitted from parents to descendants. The simplest genealogical model is the tree model, when recombination is ignored. See Figure 1 for an illustration. A common assumption is that at most one mutation occurs at any site, which is supported by the *infinite sites model*[10] from population genetics. We assume infinite sites model throughout this paper. Therefore, the genealogical tree is a perfect phylogeny (see, e.g. Gusfield[11]). A perfect phylogeny implies that at any two SNP sites, the four ordered pairs of alleles 00, 01, 10 and 11 (called gametes) can *not* be all present (called four-gamete test in population genetics). Two sites satisfying this property are said to be *compatible*. If all pairs of sites are compatible, the sequences allow a perfect phylogeny (see e.g. Gusfield[11]). Note that although gamete and diploid type use similar values, conceptually they are different: gamete means the setting of the two alleles at two sites of the same haplotype, while diploid type is for the two alleles at the same site of an individual.

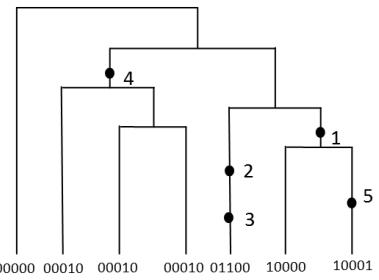


Fig. 1. *A genealogical tree.*

When meiotic recombination is considered, a more complex model is needed. Recombination takes two homologous chromosomes (haplotypes) and produces a third chromosome consisting of alternating segments (usually a small number) of the two chromosomes. With recombination, genealogical history can no longer be modeled as a single tree. Nonetheless, sometimes we can use local trees to represent local genealogical history for a short region, within which recombination does not affect the genealogy of the sampled sequences.

Biocomputing 2011 Downloaded from www.worldscientific.com
by TEXAS STATE UNIVERSITY on 03/03/18. For personal use only.

2.2. *High Throughput Sequencing (HTS)*

A main application of the HTS is on *resequencing*. In resequencing, we want to find genetic variations (e.g. SNPs) in a sample of individuals by sequencing the genomes of those individuals, when an existing, fully-sequenced, reference genome is already known. The general procedure for many resequencing applications is to first find where a new sequence read originates by comparing the sequence read with the reference genome (called reads mapping). Once the originating positions of sequence reads are found, we can then examine the mapped reads to find variations such as SNPs.

The current HTS data does not contain information on which of the two haplotypes (from a diploid organism) a read is from. This often adds complexity to data analysis. For example, suppose we have two mapped sequence reads that give the same alleles as the reference genome. We can not assert that the individual is a homozygote because the two reads may come from the *same* haplotype, and yet the sequenced individual is a heterozygote at the site. Moreover, suppose we have two mapped reads that give allele 0 and 1 at a SNP site. The individual can still be a homozygote 0 if the read with 1 allele is caused by a sequencing error. See Figure 2 for an illustration of sequencing diploid samples.

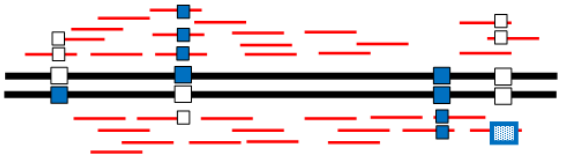


Fig. 2. *Illustration of the HTS technologies. Two thick lines are the two haplotypes of a diploid individual. Boxes are the genetic variations (e.g. SNPs), where colors indicate different allele. The short, red lines are the short sequence reads from this diploid individual, which are mapped to the proper location. The read with a dotted box (on lower right) has a sequencing error.*

3. Haplotyping with Short Reads

Haplotype inference from given *genotypes* has been actively studied recently.[12–15] Thus, a straightforward approach of inferring haplotypes with short reads is a two-stage one: first call the genotypes from the given reads (say taking the genotypes with the highest posterior probability as described in Section 3.1) and then run a population haplotype inference program (e.g. fastPHASE[15]) on the called genotypes. The main problem with this two-stage approach is that inaccurately called genotypes may lead to haplotypes of low quality. This is especially a concern when the sequencing coverage is low, which may lead to more noise in the called genotypes. In this paper, we present a new method based the *one-stage* approach, which infers haplotypes directly from the reads (i.e. without calling genotypes first). We note that few published haplotype inference approaches work directly on sequence reads, with the exception of program Beagle.[16] In Section 4, we compare out method with program Beagle.

3.1. *Posterior Probability of Genotypes at a Single Site*

Given the reads at a SNP site, it is easy to compute the posterior probability of a genotype. For ease of exposition, we assume each read has probability ϵ of reporting an incorrect allele at the site. Note that it is straightforward to allow reads having reads-specific error probability. Consider an individual i with genotype g at site s_j. We let $\mathcal{R}_{i,j}$ be the reads covering s_j for individual i, which report $r_{i,j,0}$ 0-allele and $r_{i,j,1}$ 1-allele for s_j. The single SNP genotypic

Biocomputing 2011 Downloaded from www.worldscientific.com
by TEXAS STATE UNIVERSITY on 03/03/18. For personal use only.

posterior probability is the probability of observing a particular genotype $g \in \{0, 1, 2\}$ at a site s_j given all the reads for all individuals at this site (denoted as $\mathcal{R}_{-,j}$). We define $f_j(g)$ as the genotype frequency for genotype g at site s_j. We assume that the read of interest was obtained with equal prior probability from either haplotype. Now the posterior probability of genotype g can be calculated [b]:

$$P(g = 0|\mathcal{R}_{-,j}) \propto P(\mathcal{R}_{-,j}|g = 0)P(g = 0) = (1 - \epsilon)^{r_{i,j,0}} \epsilon^{r_{i,j,1}} f_j(0) \quad (1)$$

$$P(g = 1|\mathcal{R}_{-,j}) \propto P(\mathcal{R}_{-,j}|g = 1)P(g = 1) = 0.5^{r_{i,j,0}+r_{i,j,1}} f_j(1) \quad (2)$$

$$P(g = 2|\mathcal{R}_{-,j}) \propto P(\mathcal{R}_{-,j}|g = 2)P(g = 2) = (1 - \epsilon)^{r_{i,j,1}} \epsilon^{r_{i,j,0}} f_j(2) \quad (3)$$

We use the Hardy-Weinberg equilibrium to estimate genotype frequency $f_j(g)$ at site s_j, from the frequency of alleles 0 and 1 in the population. Allele frequency can be estimated from the reads $\mathcal{R}_{-,j}$ from the observed alleles at site j. Once posterior probability is computed, a simple two-stage approach calls genotypes at each locus by picking the genotypes with maximum posterior probability, and then infer haplotypes for the called genotypes using some population haplotype inference method. As shown in Section 4, this approach is generally not as accurate as the one-stage approach we now present.

3.2. *The Special Case: No Recombination with Small Number of SNPs*

We now present an *one-stage* approach, which infers haplotypes from short reads directly. Our method rely on the shared *genealogical history* of the sampled sequences to infer haplotypes. To get started, we first consider the case when there is *no* recombination. Later, we will extend our method to allow recombination.

When there is no recombination, the underlying genealogy is a perfect phylogeny. Gusfield[13] first exploited the approach of inferring haplotypes with the perfect phylogeny model. Here, we develop a perfect phylogeny based method for inferring haplotypes with short reads. That is, we want to infer haplotypes that allow a perfect phylogeny. Note that perfect phylogeny alone can not determine the haplotypes since there are many possible haplotypes allowing perfect phylogeny. Since some haplotypes fit the given short reads better than others, a natural objective is to find the haplotypes that allow a perfect phylogeny *and* the probability of short reads given these haplotypes is maximized.

We now give the technical details. The short reads based perfect phylogeny haplotyping is, given short reads $\mathcal{R}$, finding a set of haplotypes H s.t. $P(\mathcal{R}|H)$ is maximized *and* H allows a perfect phylogeny. We let H_i denote the i-th haplotype, where $1 \leq i \leq 2n$. We let $H_{i,j}$ denote the allele (0 or 1) at the j-th site of H_i. As before, we let $\mathcal{R}_{i,j}$ be the set of reads that are taken from individual i, and cover site s_j. Consider a read $R_{i,j,k} \in \mathcal{R}_{i,j}$, which reports allele $k \in \{0, 1\}$ for site s_j. Now, $P(R_{i,j,k}|H)$ depends on $H_{2i-1,j}$ and $H_{2i,j}$. The following is related to equations 1 to 3 in Section 3.1.

[b]Similar equations have been used in Duitama, et al.,[17] and also in other statistical genetics papers

Biocomputing 2011 Downloaded from www.worldscientific.com
by TEXAS STATE UNIVERSITY on 03/03/18. For personal use only.

$$P(R_{i,j,k}|H) = 0.5 \times (P(R_{i,j,k}|H_{2i-1,j}) + P(R_{i,j,k}|H_{2i,j}))$$

Here, $P(R_{i,j,k}|h) = \epsilon$ if $k \neq h$, and $1-\epsilon$ otherwise. We consider all the reads covering site s_j in individual i, where there are $r_{i,j,0}$ 0-reads and $r_{i,j,1}$ 1-reads. Then, based on the assumption that all reads are independent, we have:

$$logP(\mathcal{R}_{i,j}|H) = r_{i,j,0} log(P(R_{i,j,0}|H)) + r_{i,j,1} log(P(R_{i,j,1}|H))$$

When $H_{2i-1,j}$ and $H_{2i,j}$ are known, these two alleles determine the diploid type $d(H) \in \{0/0, 0/1, 1/0, 1/1\}$. To simplify notations, we simply use d for diploid type. When d is given, $H_{2i-1,j}$ and $H_{2i,j}$ are also known. We let $w_{i,j,d} = logP(\mathcal{R}_{i,j}|H)$, where d is the diploid type at site s_j of individual i. We assume the reads are independent, since reads are short and thus can be treated as independent given the haplotypes. Note, however, that in practice there may exist other factors such as mapping bias that can make this assumption less accurate. Then,

$$logP(\mathcal{R}|H) = \sum_{i=1}^{n} \sum_{j=1}^{m} w_{i,j,d}$$

Our goal is finding haplotypes H, s.t. H allows a perfect phylogeny and $logP(\mathcal{R}|H)$ is maximized. Since $logP(\mathcal{R}|H)$ can be computed easily for fixed H, naively we can enumerate all possible haplotypes H to find the ones that allow a perfect phylogeny and maximize $logP(\mathcal{R}|H)$. But this is infeasible even for data of moderate size. We do not currently know an efficient algorithm for finding the optimal solution. To develop a practical method, we use integer linear programming (ILP) to solve the optimization problem *exactly*.

In our ILP formulation, we have a binary variable $D_{i,j,d}$ for individual i, site s_j and diploid type $d \in \{0/0, 0/1, 1/0, 1/1\}$, where $D_{i,j,d} = 1$ if the diploid type formed by $H_{2i-1,j}$ and $H_{2i,j}$ is d. That is, $D_{i,j,d}$ specifies which diploid type individual i carries at site s_j. For any two sites s_{j_1} and s_{j_2}, we define a binary variable $G_{j_1,j_2,g}$. $G_{j_1,j_2,g} = 1$ if sites s_{j_1} and s_{j_2} have gamete $g \in \{00, 01, 10, 11\}$. Now we give the sketch of the ILP formulation.

Objective: maximize $\sum_{i=1}^{n} \sum_{j=1}^{m} \sum_{d \in \{0/0,0/1,1/0,1/1\}} w_{i,j,d} \times D_{i,j,d}$.

Subject to

1 $D_{i,j,0/0} + D_{i,j,0/1} + D_{i,j,1/0} + D_{i,j,1/1} = 1$, for each $1 \leq i \leq n$ and $1 \leq j \leq m$.

[We now impose constraints on $G_{j_1,j_2,d}$. We only give the constraints for $G_{j_1,j_2,00}$. The rest are similar and thus omitted.]

2 $G_{j_1,j_2,00} + D_{i,j_1,1/1} \geq D_{i,j_2,0/0}$, for all $1 \leq j_1 < j_2 \leq m$ and $1 \leq i \leq n$.

3 $G_{j_1,j_2,00} + D_{i,j_2,1/1} \geq D_{i,j_1,0/0}$, for all $1 \leq j_1 < j_2 \leq m$ and $1 \leq i \leq n$.

4 $G_{j_1,j_2,00} + 1 \geq D_{i,j_1,0/1} + D_{i,j_2,0/1}$, for all $1 \leq j_1 < j_2 \leq m$ and $1 \leq i \leq n$.

5 $G_{j_1,j_2,00} + 1 \geq D_{i,j_1,0/1} + D_{i,j_2,0/0}$, for all $1 \leq j_1 < j_2 \leq m$ and $1 \leq i \leq n$.

6 $G_{j_1,j_2,00} + 1 \geq D_{i,j_1,1/0} + D_{i,j_2,1/0}$, for all $1 \leq j_1 < j_2 \leq m$ and $1 \leq i \leq n$.

7 $G_{j_1,j_2,00} + 1 \geq D_{i,j_1,1/0} + D_{i,j_2,0/0}$, for all $1 \leq j_1 < j_2 \leq m$ and $1 \leq i \leq n$.

[We now ensure no four gametes exists at any pair of sites]

8 $G_{j_1,j_2,00} + G_{j_1,j_2,01} + G_{j_1,j_2,10} + G_{j_1,j_2,11} \leq 3$ for all $1 \leq j_1 < j_2 \leq m$.

For each $1 \leq i \leq n$, $1 \leq j \leq m$ and $d \in \{0/0, 0/1, 1/0, 1/1\}$, there is a binary variable $D_{i,j,d}$.

Biocomputing 2011 Downloaded from www.worldscientific.com
by TEXAS STATE UNIVERSITY on 03/03/18. For personal use only.

For each $1 \leq j_1 < j_2 \leq m$ and $g \in \{00, 01, 10, 11\}$, there is a binary variable $G_{j_1,j_2,g}$.

Briefly, constraint (1) states that each individual must take exactly one of the four diploid type at a site. Constraints (2) to (7) relate diploid variables $D_{i,j,d}$ with the gamete variables $G_{j_1,j_2,00}$. For example, constraint (2) states that if the diploid type at site s_{j_1} is not 1/1 (i.e. $D_{i,j_1,1/1} = 0$) and the diploid type at site s_{j_2} is 0/0 (i.e. $D_{i,j_2,0/0} = 1$), then there exists gamete 00 at sites s_{j_1} and s_{j_2}. Constraint (8) states that there are at most three gametes for any two sites, which is required by the perfect phylogeny model. The constraints for the other diploid type variables are similar. Finally, the objective function uses the diploid type variables times the weights, which means that only the selected diploid types (i.e. $D_{i,j,d} = 1$) contribute to the objective. Once the ILP formulation is solved, the haplotypes are readily retrieved from the values of the $D_{i,j,d}$ variables.

Simulation in Section 4 shows that this ILP formulation can be practically solved for many data, especially when the number of sites (i.e. m) is relatively small (say less than 20).

Biocomputing 2011 Downloaded from www.worldscientific.com
by TEXAS STATE UNIVERSITY on 03/03/18. For personal use only.

3.3. *The General Case: with Recombination and Larger Number of SNPs*

When data size grows or recombination occurs, we can no longer directly use the ILP-based approach in Section 3.2. We now extend our approach to handle data with recombination and/or larger number of sites. Our strategy is similar in high-level to the approach in Halperin and Eskin:[14] we first infer haplotypes using the ILP based approach in Section 3.2 on small number of consecutive (and overlapping) SNPs (called *windows*); then we *concatenate* these overlapping haplotypes to create complete haplotypes for the entire data. This approach may work well when recombination rate is relatively low: in this case, there are relatively long genomic regions with no recombination. Also, even when there is a small number of recombinations within a region, perfect phylogeny may still be a good approximation of the genealogical history of the region.

Specifically, we let the size of the sliding window (i.e. number of sites) be W, which starts from the first site. Each time, we move the window to the right by $\frac{W}{2}$ sites to obtain haplotypes in a list of overlapping windows by the ILP approach. Then, we concatenate the haplotypes of the overlapped windows from the left to the right. Let h_{2i-1} and h_{2i} be the haplotypes of individual i in a window, and h'_{2i-1} and h'_{2i} be the haplotypes in an overlapping window. Note that the haplotypes of an individuals within two overlapped windows are obtained from different ILP solutions, and thus the two pairs of haplotypes need to be paired up properly. Moreover, sometimes concatenation may require changes to these haplotypes for consistency.

Here are the main steps of haplotype concatenation.

(1) First concatenate obvious haplotypes. Sometimes only one pairing between the two pairs of haplotypes is perfect (e.g. the overlapped portions of h_{2i-1} and h'_{2i-1} match perfectly, so do those of h_{2i} and h'_{2i}, and the other pairing of the haplotypes is not perfect). In this case, we simply greedily choose the obvious pairing to obtain two concatenated haplotypes.
(2) The previous step often generates a set of inferred haplotypes. Now we use these already inferred haplotypes to help to resolve the other undecided haplotype pairs. If two haplotypes (say h_{2i-1} and h'_{2i-1}) can be merged perfectly (i.e. with no mismatches within the

overlapped region) to generate one of the existing haplotypes, we just take this particular pairing if the other haplotype pair is approximately consistent.

(3) Since haplotypes within a window are usually closely related through mutation and recombination, this provides more hints on how to concatenate the haplotypes. Suppose we are evaluating two choices of pairing, which generate two sets of candidate haplotypes. We compare the two sets of haplotypes and choose the ones that are closely related to the already inferred haplotypes. A haplotype h is closely related to a set of haplotypes H if (a) the Hamming distance between h and a haplotype $h' \in H$ is small, or (b) h can be broken into a small number of segments, s.t. each segment appears in H. The later can be easily evaluated by either a dynamic programming algorithm or a greedy algorithm.[18]

(4) Here is one more rule in deciding how to concatenate the haplotypes, which is applied if the previous step leads to multiple equally good choices. When recombination occurs, some pairs of sites become incompatible. However, a site is still likely to be compatible with its neighboring sites.[19] For a site s, the compatible region of s is a continuous set of sites, each of which is compatible with s (but there may exist two incompatible sites among these sites other than s). Based on this observation, we select the haplotype pairings that give longer compatible regions.

Biocomputing 2011 Downloaded from www.worldscientific.com
by TEXAS STATE UNIVERSITY on 03/03/18. For personal use only.

4. Results

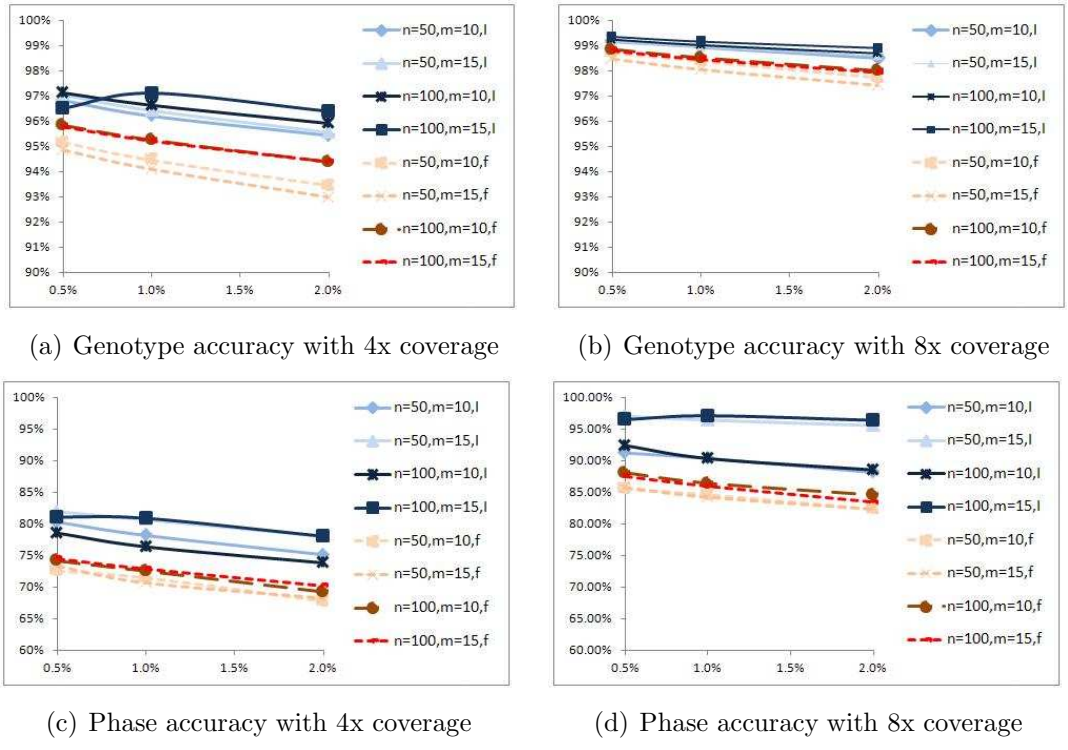
We have implemented our method in a program (called HapReads) written with C++, which uses either CPLEX (a commercial and faster ILP solver) or GNU GLPK ILP solver. HapReads can be downloaded from: http://www.engr.uconn.edu/~jiz08001. Our simulation results are from the CPLEX version. We test our method on simulated data on a 3192 MHz Intel Xeon workstation. We use Hudson's program ms[20] to generate haplotypes for different settings on the number of diploid individuals, the number of sites and recombination rate. Then, for each set of haplotypes, we simulate the sequence reads by (1) deciding the number of reads to generate based on the sequencing coverage, and (2) randomly picking the sites for the reads and one of the two haplotypes when reporting the alleles in the reads. To simulate the sequencing errors and other noise, we generate sequence reads with some error probability ϵ (the probability of reporting a wrong allele). We generate 100 datasets for each setting.

To evaluate the accuracy of our method (and the two-stage approach using fastPHASE), we compare the inferred haplotypes with the true simulated haplotypes. We run program fastPHASE by letting the program to choose the number of clusters itself. We assume error probability ϵ is known to *both* our method and fastPHASE. Different from haplotyping from given genotypes (where there is only phasing errors), there are two types of errors: (a) genotype errors, and (b) haplotype phase errors. Genotype errors refer to the genotypes (implied by the inferred diploid types) that are different from the true genotypes. We define genotype accuracy A_g as the percentage of correctly called genotypes. We define phase accuracy A_p as the switching accuracy[21] that is related to the incorrectly phased neighboring heterozygotes. Note that calculating phase accuracy needs first *correcting* the genotype errors (i.e. changing the diploid types in some ways so that the corresponding genotypes match the true genotypes). There is a subtle issue in computing phase accuracy A_p when there are genotype errors.

Suppose the true diploid type is 0/1 (i.e. heterozygote), while the inferred diploid type is 0/0 (i.e. homozygote). We can use either 0/1 or 1/0 to correct the genotype that may lead to different phase accuracy. To get over this issue, we use the *average* phase accuracy over all possible choices for these corrected diploid types.

4.1. *Accuracy of the ILP formulation*

We first evaluate the accuracy of the ILP-based approach in Section 3.2. Recall that the ILP approach is practical when there is no recombination and the number of sites is relatively small. In Figure 3, we show the average (genotype and phase) accuracy for various number of individuals and sites, sequence read error rates and coverage.



(a) Genotype accuracy with 4x coverage

(b) Genotype accuracy with 8x coverage

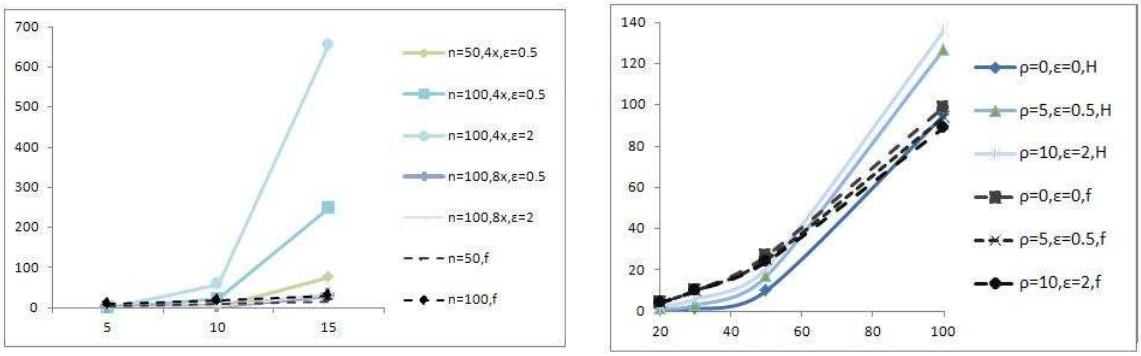
(c) Phase accuracy with 4x coverage

(d) Phase accuracy with 8x coverage

Fig. 3. Accuracy of ILP-based method and fastPHASE under different reads error rates and coverage. n: the number of individuals, m: the number of sites. I: ILP-based method (solid line). f: fastPHASE (dashed line).

Figure 3 shows that our ILP approach outperforms the two-stage approach using fastPHASE (or simply fastPHASE) in both genotype accuracy and phase accuracy in most datasets of the simulations. For example, for 50 individuals, 15 sites, error rate 1% and coverage 4x, the phase accuracy of our method is roughly 10% more than that of the two-stage approach, even when the difference between genotype accuracy is about 2.5%. This suggests that our method works well in inferring haplotypes when there is no recombination. As expected, when read error rate is higher and coverage is lower, phase accuracy tends to be lower. One downside is that the ILP solving gets slower when the number of sites increases, which is shown in Figure 4(a).

Biocomputing 2011 Downloaded from www.worldscientific.com
by TEXAS STATE UNIVERSITY on 03/03/18. For personal use only.



(a) Our ILP-based method and fastPHASE (b) Our heuristic approach and fastPHASE

Fig. 4. Running time of ILP-based method, heuristic approach and fastPHASE. n: the number of individuals. x: reads coverage. ϵ: reds error rates. ρ: recombination rate. H: our heuristic approach. f: fastPHASE.

Biocomputing 2011 Downloaded from www.worldscientific.com
by TEXAS STATE UNIVERSITY on 03/03/18. For personal use only.

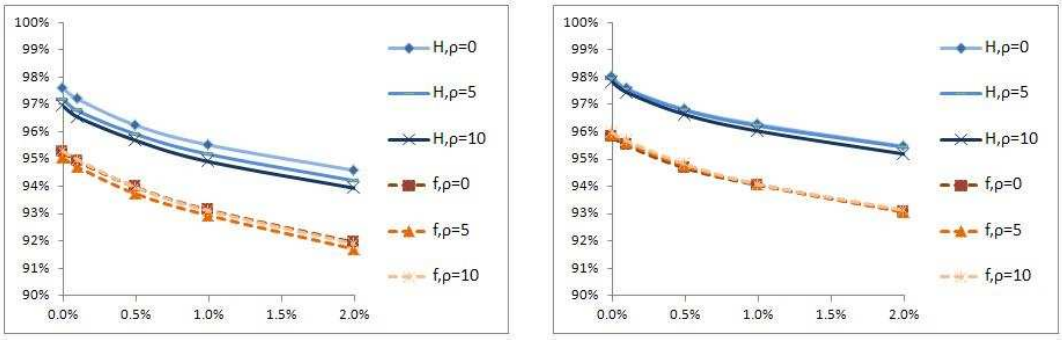
4.2. *Accuracy of the case with larger data*

We now evaluate the performance of the heuristic approach in Section 3.3, which allows us to handle problem instances that are larger or with recombination. We use 4x coverage in this simulation. The results are obtained by inferring haplotypes from a sliding window of 10 sites, and then concatenating the overlapped haplotypes.

Figure 5 shows that in terms of genotype accuracy and phase accuracy, our one-stage approach is consistently more accurate. Thus, the simulation results show that our one-stage approach outperforms the two-stage approach in general. Also, our method remains reasonably accurate with higher sequence reads error (up to 2%) or when recombination rate increases (up to 10). We note that genotype accuracy in our simulation is often fairly accurate. Phase accuracy, on the other hand, is in general not very high for both methods. One reason may be the low sequencing coverage: we use 4x coverage here and increasingly coverage may improve the phase accuracy. Moreover, as shown in Figure 4(b), the running time of our method is similar to the two-stage approach for the data we simulate.

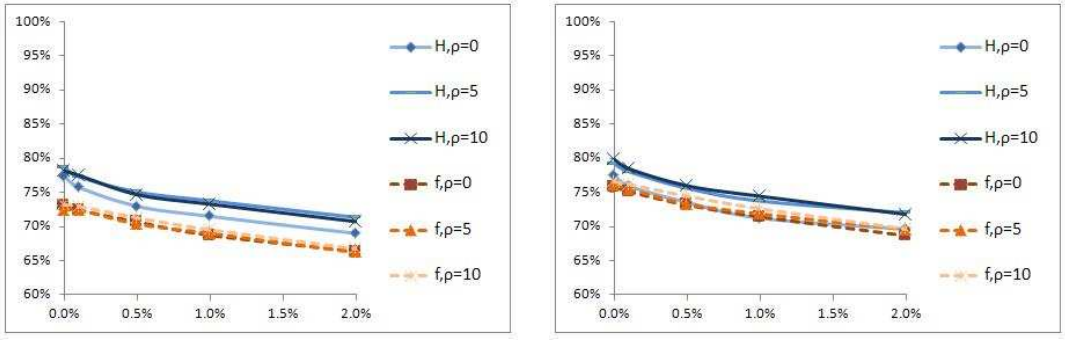
4.3. *Comparing with program Beagle with simulated and biological data*

Program Beagle[16] allows uncertain genotypes which are specified by genotype probabilities. Thus, Beagle can be used as a one-stage approach so we compare program Beagle and our approach. We run program Beagle with the same data sets generated by program ms in Section 4.2. The result is given in Figure 6. For data sets with 25 individuals and 50 sites, our method and Beagle have similar genotype accuracy and our method has slightly higher phase accuracy, but from data sets with 50 individuals and 100 sites, our method is less accurate than Beagle. We also test the two approaches on simulated reads for HapMap haplotypes. We generated 100 data sets of 25 individuals by 50 sites from 100 regions on chromosome 1 of CEU population. The results are similar (results omitted, with Beagle being slightly more accurate). One possible reason is that HapMap haplotypes are only for common SNPs, where haplotypes within a window are less likely to allow a perfect phylogeny. More simulations are needed to further compare the two methods. Overall, one-stage approaches appear to perform



(a) A_g: 25 individuals and 50 sites.

(b) A_g: 50 individuals and 100 sites.



(c) A_p: 25 individuals and 50 sites.

(d) A_p: 50 individuals and 100 sites.

Fig. 5. Accuracy of heuristic approach and fastPHASE with different reads error rates. H refers to heuristic approach (solid lines) and f refers to fastPHASE (dashed lines). ρ is recombination rates.

Biocomputing 2011 Downloaded from www.worldscientific.com
by TEXAS STATE UNIVERSITY on 03/03/18. For personal use only.

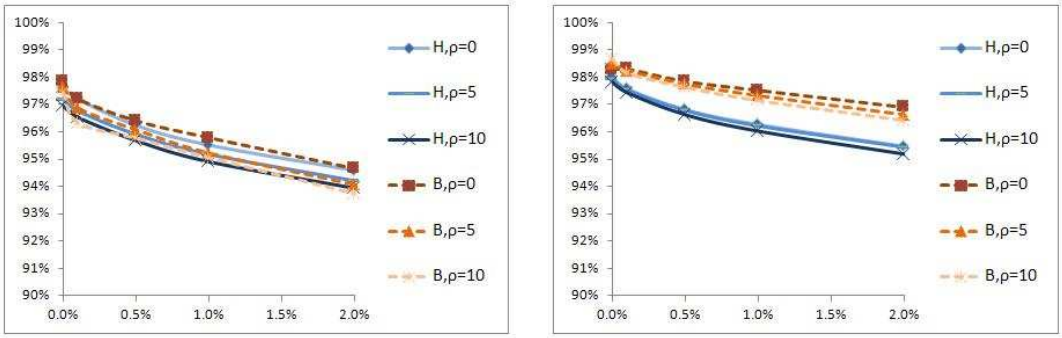
better than two-stage approaches.

Funding and Acknowledgment

The research is supported by grants from U.S. National Science Foundation (IIS-0803440, IIS-0916948 and IIS-0953563).

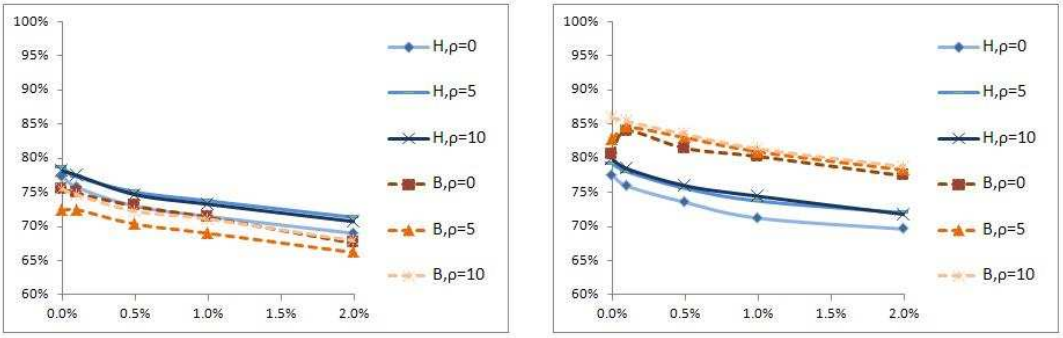
References

1. S. Levy *et al.*, *PLoS Biology* **5**, e254+ (2007).
2. D. Wheeler *et al.*, *Nature* **452**, 872 (2008).
3. The 1000 genomes project consortium http://www.1000genomes.org/.
4. Pacific Biosciences http://www.pacificbiosciences.com/index.php?q=home.
5. International HapMap Consortium, *Nature* **426**, 789 (2003).
6. International HapMap Consortium, *Nature* **449**, p. 851861 (2007).
7. V. Bansal and V. Bafna, *Bioinformatics* **24**, 153 (2008).
8. D. He, A. Choi, K. Pipatsrisawat, A. Darwiche and E. Eskin, *Bioinformatics* **26**, i183 (2010).
9. T. A. Manolio, F. S. Collins, N. J. Cox, D. B. Goldstein, L. A. Hindorff, D. J. Hunter, M. I. McCarthy, E. M. Ramos, L. R. Cardon, A. Chakravarti, J. H. Cho, A. E. Guttmacher, A. Kong, L. Kruglyak, E. Mardis, C. N. Rotimi, M. Slatkin, D. Valle, A. S. Whittemore, M. Boehnke, A. G. Clark, E. E. Eichler, G. Gibson, J. L. Haines, T. F. C. Mackay, S. A. McCarroll and P. M. Visscher, *Nature* **461**, 747 (2009).



(a) A_g: 25 individuals and 50 sites.

(b) A_g: 50 individuals and 100 sites



(c) A_p: 25 individuals and 50 sites.

(d) A_p: 50 individuals and 100 sites

Fig. 6. Accuracy of heuristic approach (solid lines) and Beagle (dashed lines) under different reads error rates. H refers to Heuristic approach and B refers to Beagle. ρ is recombination rates.

Biocomputing 2011 Downloaded from www.worldscientific.com
by TEXAS STATE UNIVERSITY on 03/03/18. For personal use only.

10. G. A. Watterson, *Theoretical Population Biology* **7**, 256 (1975).
11. D. Gusfield, *Algorithms on Strings, Trees and Sequences: Computer Science and Computational Biology* (Cambridge University Press, Cambridge, UK, 1997).
12. M. Stephens, N. Smith and P. Donnelly, *Am. J. Human Genetics* **68**, 978 (2001).
13. D. Gusfield, Haplotyping as Perfect Phylogeny: Conceptual Framework and Efficient Solutions (Extended Abstract), in *Proceedings of RECOMB 2002: The Sixth Annual International Conference on Computational Biology*, 2002.
14. E. Halperin and E. Eskin, *Bioinformatics* **20**, 1842 (2004).
15. P. Scheet and M. Stephens, *Am. J. Human Genetics* **78**, 629 (2006).
16. B. L. Browning and Z. Yu, *American Journal of Human Genetics* **85**, 847 (2009).
17. J. Duitama, J. Kennedy, S. Dinakar, Y. Hernández, Y. Wu, and I. Măndoiu, Linkage Disequilibrium Based Genotype Calling from Low-Coverage Shotgun Sequencing Reads, manuscript.
18. Y. S. Song, Y. Wu and D. Gusfield, *Bioinformatics* **21**, i413 (2005), Bioinformatics Suppl. 1, Proceedings of ISMB 2005.
19. Y. Wu, New methods for inference of local tree topologies with recombinant snp sequences in populations (2010), IEEE/ACM Trans. of Comput. Biol. and Bioinfo., in press.
20. R. Hudson, *Bioinformatics* **18**, 337 (2002).
21. S. Lin, D. Cutler, M. Zwick and A. Chakravarti, *Am. J. of Hum. Genet.* **71**, 1129 (2002).