Haplotype Inference in Complex Pedigrees

Bonnie Kirkpatrick[1], Javier Rosa[2], and Eran Halperin[3], Richard M. Karp[4]

[1] Computer Science Dept, University of California, Berkeley. Email: bbkirk@eecs.berkeley.edu

[2] Computer Science Dept, Rutgers, The State University of New Jersey, New Brunswick

[3] School of Computer Science and the Dept. of Biotechnology, Tel-Aviv University, and the International Computer Science Institute, Berkeley. Email: heran@icsi.berkeley.edu

[4] Computer Science Dept, University of California, Berkeley and the International Computer Science Institute. Email: karp@icsi.berkeley.edu

Abstract. Despite the desirable information contained in complex pedigree datasets, analysis methods struggle to efficiently process these datasets. The attractiveness of pedigree data sets is their power for detecting rare variants, particularly in comparison with studies of unrelated individuals. In addition, rather than assuming individuals in a study are unrelated, knowledge of their relationships can avoid spurious results due to confounding population structure effects. However, a major challenge for the applicability of pedigree methods is the ability handle complex pedigrees, having multiple founding lineages, inbreeding, and half-sibling relationships.

A key ingredient in association studies is imputation and inference of haplotypes from genotype data. Existing haplotype inference methods either do not efficiently scales to complex pedigrees or their accuracy is limited. In this paper, we present algorithms for efficient haplotype inference and imputation in complex pedigrees. Our method, PhyloPed, leverages the perfect phylogeny model, resulting in an efficient method with high accuracy. In addition, PhyloPed effectively combines the founder haplotype information from different lineages and is immune to inaccuracies in prior information about the founders.

1 Introduction

Both genetic and environmental factors affect the etiology of complex conditions such as cancer or Alzheimer's disease. In an attempt to reveal the genetic factors of these conditions, many researchers use pedigree studies, in which the genomes of a set of related cases and controls are compared. Regions in which the allelic distribution of the cases differs from the expected distribution given the pedigree relationships are suspect for direct or indirect involvement in the disease mechanism.

Current studies focus on analyzing *single nucleotide polymorphisms* (SNPs), which are mutations that occurred once in history and propagated through the population. Common SNPs are usually bi-allelic with both of the alleles appearing in at least 5% of the population. Current genotyping technologies allow us to genotype a set of a million SNPs spread across the whole genome for less than a thousand dollars per person. Thus, large scale studies, involving hundreds of thousands of SNPs and thousands of individuals, are feasible. Indeed, if the genealogical data exists, it is possible to obtain a pedigree for thousands of individuals (e.g. the Hudderite data with more than 1600 individuals [1], and animal breeding data [17]).

Although many current study designs employ population case-control designs, with unrelated individuals, there are substantial advantages to using pedigree study designs. Designing population-based studies may be problematic due to confounding effects such as population substructure and heterogeneity within the case population, (i.e., the set of cases consists of a few subsets, where each subset has a different disease that is manifested in the same way, but is genetically and probably biologically different). These phenomena may reduce power or lead to false discoveries. However, incorporating related individuals into association studies bypasses these problems by correcting for sources of heterogeneity and population substructure. Knowing relationships between family members, as in a pedigree, can assist in obtaining a more accurate estimate of each individual's haplotypes, which are sequences of alleles on a chromosome that were inherited from the same ancestor. Haplotypes can be used for imputation of unobserved genotypes or alleles, which have been useful in finding new associations [2]. Furthermore, imputation in family-based association studies has been shown to increase power [5, 3].

The theoretical usefulness of large pedigree datasets is diminished in practice by computational issues. Pedigree analysis is known to be NP-hard [16], and all known algorithms have exponential running time. The classical trade-off for running time is between being exponential in the number of loci (and linear in the number of individuals) or exponential in the number of individual (and linear in the number of loci), for example, the Elston-Stewart and Lander-Green algorithms respectively [8, 14]. More recent work, in the form of Superlink [10], heuristically optimizes this trade-off. Among the MCMC approaches to pedigree analysis, blocked Gibbs sampling has been successfully applied to large pedigrees [13, 18]. Blocked Gibbs samplers are a generalization of Gibbs samplers where a set of variables is updated at each step rather than a single variable. These samplers elegantly deal with inbreeding by conditioning and rely on the random steps of the Markov Chain to propagate the effect of inbreeding through the graphical model. Convergence occurs quickly in practice. However correctness is typically proved via irreducibility of the state space for a particular problem instance, and these proofs themselves may correspond to difficult computational problems [4, 13].

In this paper, we consider a special case of the pedigree haplotyping problem for complex pedigrees, having multiple lineages. Specifically, we are interested in regions of the genome that are sufficiently linked that there is little evidence of recombination during pedigree meiosis. Further, there are two cases for these regions. First, if there is little evidence of ancestral recombinations or recurrent mutations in the founding haplotypes, the perfect phylogeny model [11] would apply to the pedigree haplotypes. The perfect phylogeny model has been shown to be realistic as long as

the studied region is physically short [6, 7, 9]. Second, if there is evidence of ancestral recombinations in the region, then ancestral recombinations must be allowed, and the founding haplotypes are not restricted to a perfect phylogeny. We make no other assumptions about recombination rates or founder allele frequencies. These two cases allow us to make simplifying assumptions and allow efficient computation over large and complex pedigrees without compromising accuracy.
To solve the first case of the problem, we propose a blocked Gibbs sampler with running time polynomial in the number of SNPs and linear in the number of individuals. Roughly, PhyloPed, our method, chooses overlapping blocks of individuals that correspond to lineages in the pedigree. A single sampling step updates the haplotype assignments for all the individuals in the lineage of interest. The algorithm considers each lineage in turn, updates that lineage, and continues until convergence. PhyloPed begins the blocked Gibbs sampler at an initial state that is a feasible haplotype configuration that is compatible with the perfect phylogeny. In practice, the initial haplotype state can often be obtained quickly, though in the worst case, due to disallowing recombination, the running time may be exponential in the number of individuals. In the case that the founder haplotypes could not have come from a perfect phylogeny, PhyloPed reverts to the second case, without the perfect phylogeny, and runs the same blocked Gibbs sampler from an initial haplotype configuration with unrestricted founder haplotypes (with running time exponential in the number of SNPs). Furthermore, PhyloPed does not require knowledge of recombination rates or founder-allele frequencies. The perfect phylogeny allows more accurate haplotype inference, for a small number of SNPs.

2 Methods

We represent a pedigree on a set of individuals I as a directed graph having individuals as nodes (either circles or squares) and relationships indicated by edges and marriage nodes (solid diamonds, see Figure 1). The pedigree edges are usually *implicitly* directed, with the edges being directed downwards. Parent-child relationships are drawn with a vertical arrangement of nodes and edges, with edges from the parent down to a marriage node and from the marriage node down to the child (Fig. 1). The *founders* of the pedigree are individuals $F \subset I$ whose parents are not represented in the graph. According to convention, assume that every non-founder has both their parents represented in the pedigree, so that every marriage node has two parents above and adjacent to it. For each of the M bi-allelic SNPs, every individual w has an unordered single-locus genotype g_w^m at SNP m. An individual with a fully observed genotype has a single set of possible alleles $g_w^m \in \{\{0,0\},\{0,1\},\{1,1\}\}$. An individual with an unobserved or partially observed genotype has several possible sets of alleles $g_w^m \in \{\{\{0,0\},\{0,1\},\{1,1\}\},\ \{\{0,0\},\{0,1\}\},\ \{\{0,1\},\{1,1\}\}\}$.
We denote a haplotype by a sequence of binary alleles, $h \in \{0,1\}^M$, where M is the number of SNPs in a region of the genome. Let the m'th allele

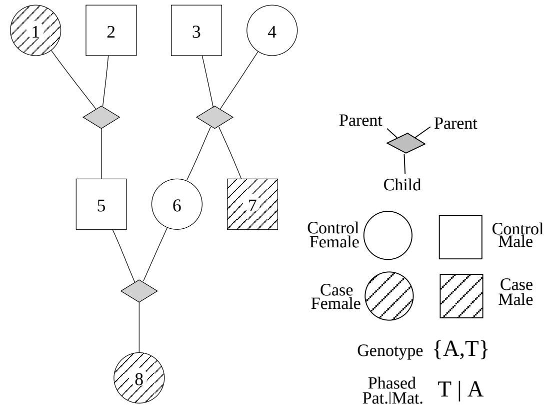

Fig. 1. Pedigree Example and Key

of haplotype h be denoted $h(m)$. A pedigree *state* associates an ordered pair of haplotypes (or multi-locus genotype), $s_w = (h_w, h'_w)$, with each individual $w \in I$. The haplotypes of an individual are *consistent* with the observed genotypes provided that at each SNP m, all known alleles are represented in the haplotypes, meaning that $\{h_w(m), h'_w(m)\} \subset g_w^m$. Let $C(w, s_w)$ be an indicator variable that is 1 when the haplotype state s_w is consistent with the observed genotypes. Let $S(w)$ be the set of all haplotype states that are consistent with w's observed genotype.

We assume that the M SNPs are tightly linked and are effectively unable to recombine when passed from one generation to the next. In other words, haplotypes are passed according to Mendelian inheritance. More precisely, we define an individual's haplotype state s_w as *non-recombinant* when the haplotype h_w inherited from the father, $f(w)$, exactly matches one of the father's haplotypes, $h_w = h_{f(w)}$ or $h_w = h'_{f(w)}$, and similarly h'_w, inherited from the mother, $m(w)$, matches one of the mother's haplotypes.

Mendelian inheritance gives the probability that each of the father's (or mother's) haplotypes are inherited by the child: $Pr[h_w = h_{f(w)} | h_{f(w)} \neq h'_{f(w)}] = 1/2$ and $Pr[h_w = h_{f(w)} | h_{f(w)} = h'_{f(w)}] = 1$. This assumption determines a family of probability distributions over states of the pedigree, given genotype data for some of the individuals. Our goal is to find the haplotypes that maximize the conditional distribution of pedigree states given genotype data for some individuals.

Lineage Decomposition. Rather than computing the joint distribution of haplotype assignments to all the founders, we decompose a complex pedigree into tree-like lineages. Roughly speaking, each lineage is a block of variable that will be updated in a single iteration of the blocked Gibbs sampler, and the lineages are not necessarily disjoint from each other.

A lineage is defined as follows. Let $H(w)$ denote the set of children of node w. Founders p and q are called a *monogamous founding pair (MFP)* if and only if $H(p) = H(q)$ (i.e. there are no half-siblings of the founders' children). Assume that the pedigree contains only monogamous married pairs of founders. The *lineage of the monogamous founding pair* (p, q) is the induced subgraph of the pedigree that contains all the descendants of p and q. Formally, the lineage $L(p, q)$ is a directed, acyclic graph that contains a source node for each p, q and a node for each descendant of p, q. This means that $L(p, q)$ is the smallest subgraph of the given pedigree such that L contains both founders p, q and, if L contains a node w, then L contains the children of w. If a parent of w is not in L then that parent is called the *non-lineage* parent.

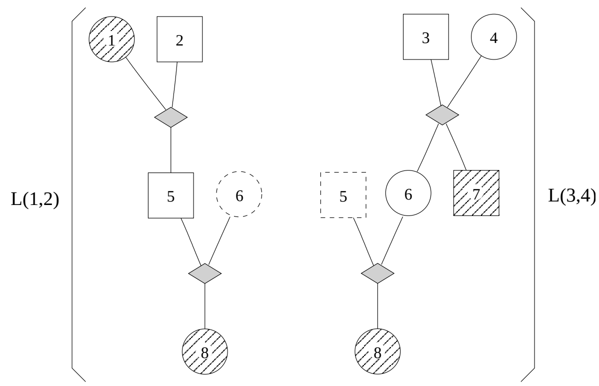

Fig. 2. These are the lineages for the pedigree in Fig. 1. The non-lineage parents are dashed, and individuals 6 and 5, respectively, are parents of individuals in the lineages $L(1, 2)$ and $L(3, 4)$

For example the lineages of the pedigree in Fig. 1 are shown in Fig. 2 as two distinct pedigree sub-graphs. Notice that parents of lineage members fall into four categories: 1) founders participating in the MFP, and 2) non-lineage founding parents, 3) non-lineage parents (non-founders who are descendants of another lineage) and 4) lineage descendants (descended from the MFP).

Our goal is to chose a haplotype state for the pedigree from the posterior distribution of haplotype states given the genotype data of the pedigree and assumptions about haplotype sharing between lineages. A side effect of this is that PhyloPed infers all missing alleles, including haplotypes (and genotypes) for ungenotyped individuals. To find a haplotype state, we consider each lineage separately and calculate the distribution of haplotype states for the monogamous pairs of founders, given the genotype data of their descendants and probabilistic assumptions about the states of the non-lineage founders and non-lineage parents. The calculation proceeds in three phases:

1. Find a consistent, non-recombinant state for the pedigree. If there is such a state that is also compatible with some perfect phylogeny on all of the observed genotypes [11, 9], choose that state (see Supplement). Otherwise, when the founder haplotypes require ances-

tral recombinations or back mutations, choose any consistent, non-recombinant state for the pedigree haplotypes.
2. Decompose the pedigree into lineages, as described above.
3. Iterate over the collection of lineages: first, compute the distribution for the MFP haplotypes conditioning on the genotypes in the lineage and conditioning on the current states of the non-lineage parents, and second, sample new haplotype states for the lineage descendants from the computed distribution.

Inference for a Single Lineage. To describe step 3 in detail, we need to establish a few assumptions. First, assume that we have a consistent, non-recombinant state for the pedigree (for details, see Supplement). Also, assume for the moment that there is a known prior probability for founder haplotypes, $\alpha(h)$ for the 2^m possible haplotypes. Each time inference is performed on a lineage, the algorithm removes inbreeding loops by randomly choosing an individual to condition on. This is done by successively finding the oldest inbred descendant (whose parents are not inbred) and flipping a coin to choose which parent will be designated the non-lineage parent for the duration of the iteration.

For a single, non-inbred lineage, we can compute the probability of the MFP haplotypes by conditioning on 1) the haplotype assignments of the non-lineage parents, 2) the genotypes, and 3) the prior probability α. The child of a non-lineage parent inherits either one of the two equally-likely non-lineage haplotypes, if the non-lineage parent has a haplotype assignment, or one of the 2^m possible founder haplotypes drawn from α, if the non-lineage parent is an ungenotyped founder. The prior and transmission probabilities yield a tree-like graphical model from which to learn the MFP haplotype distribution. For the lineage $L(p, q)$, let $\phi_{p,q}(i, j, k, l)$ be the marginal probability of the haplotype assignment (i, j) to p and (k, l) to q conditioned on the genotypes in the lineage and the haplotype assignments of the non-lineage parents. This is a marginal probability, because it is computed by summing over possible haplotype assignments for lineage descendants.

Some fairly standard bottom-up dynamic-programming equations yield the MFP marginal $\phi_{p,q}(i, j, k, l)$ and incomplete marginals, or messages, for the lineage descendants (see the peeling algorithm in [15]). The descendant marginals are incomplete, because they are computed by summing only over possible haplotype assignments to their descendants (rather than summing also over possible assignments to their ancestors), and are conditioned only on the genotypes of their descendants (rather than all of the lineage genotypes). For an MFP child, r, with two lineage haplotypes (i, j), define $\phi_r(i, j)$ as the incomplete marginal probability of r having haplotypes (i, j) conditioned on the genotypes of r's descendants. Similarly, for all other lineage descendants, with one lineage haplotype, define $\phi_w(i)$ as the incomplete marginal probability of w having lineage haplotype i conditioned on the genotypes of w's descendants (see the Supplementfor equations).

When considering all possible haplotype assignments, computation of these conditional probabilities takes time $O(N^4 \cdot L)$ where $N = 2^m$ is the number of possible haplotypes and L is the number of individuals in the lineage. Recall that m is small and the computation is feasible, because all the SNPs are in a short region of the genome and are in linkage disequilibrium. In cases where only perfect phylogeny haplotypes are considered, the running time is reduced to $O((m+1)^4 L)$. This follows from the fact that a perfect phylogeny contains at most $m+1$ haplotypes. In order to update the haplotype state of a lineage, we use a top-down random propagation algorithm that chooses a new pair of haplotypes for each individual in the lineage (similar to the random propagation algorithm described in [15]). Random propagation allows us to choose haplotype assignments for each person from the correct, or complete, marginal haplotype distribution for that individual. For the pair of founders, p, q, haplotypes (i, j, k, l) are chosen proportional to $\alpha(i)\alpha(j)\alpha(k)\alpha(l)\phi_{p,q}(i, j, k, l)$. Children, r, of the MFP are randomly assigned haplotypes $(h_r, h'_r) \in \{(i, k), (i, l), (j, k), (j, l)\}$, conditional on the MFP haplotype assignment (i, j, k, l), with probability proportional to $\phi_r(h_r, h'_r)$. All other lineage descendants, w, are given a lineage haplotype h_w conditional on the haplotypes $(h_{l(w)}, h'_{l(w)})$ of their lineage parent $l(w)$. So, $Pr[h_w = h_{l(w)}]$ is proportional to $\phi_w(h_{l(w)})$. And the non-lineage haplotypes h'_w is chosen from the set $\{h_{n(w)}, h'_{n(w)}\}$ of haplotypes for the non-lineage parent $n(w)$ and probability proportional to $\alpha(h'_w)C(w, h_w, h'_w)$. The random propagation scheme is also accomplished in time $O(N^4 L)$, except when perfect phylogeny haplotypes are known, making the running time $O((m + 1)^4 L)$.

Inference for Multiple Lineages. We now extend our algorithm to consider several monogamous founding pairs simultaneously. We no longer assume that there is a fixed α distribution or that the haplotype states never change. Instead, we use an iterative process that computes a new haplotype distribution α^t at each iteration t and maintains a consistent, non-recombinant haplotype state for all the individuals in the pedigree. For each iteration, t, consider each MFP (p, q) and its lineage $L(p, q)$:

1. Given the previous estimate of α^{t-1}, perform the bottom-up dynamic programming calculation to compute $\phi^t_{p,q}(i, j, k, l)$, $\phi^t_r(i, j)$ and $\phi^t_w(i)$ for the MFP (p, q).
2. Use α^{t-1} together with the various ϕ^t probabilities in the random propagation scheme to sample a new haplotype state for the individuals in the lineage.

After obtaining an updated $\phi^t_{p,q}(i, j, k, l)$ for each MFP (p, q), compute the updated prior distribution as the marginal average
$\alpha^t(h) \propto \sum_{(p,q)} m^t_{p,q}(i) + m^t_{p,q}(j) + m^t_{p,q}(k) + m^t_{p,q}(l)$
where the marginal $m^t_{p,q}(i) = \sum_j \sum_k \sum_l \phi^t_{p,q}(i, j, k, l)$ and similar definitions apply for $m^t_{p,q}(j)$, $m^t_{p,q}(k)$, and $m^t_{p,q}(l)$. The iterations continue until the l_1 deviation between α^t and α^{t-1} falls below a pre-determined threshold. Clearly the running time of our method depends on the number of iterations until convergence. In practice, the l_1 deviation of the α^t estimates drop rapidly and most of the blocks in Fig. 3 converged in roughly 6-8 iterations (see Supplement).

Correctness. We have described a blocked Gibbs sampling scheme where in each iteration, the updated block is a non-inbred subgraph of a pedigree lineage. Each update step uses a mixture of bottom-up recursion and top-down sampling to update the haplotype assignments in each block. A Markov Chain employing this update algorithm will converge to the correct posterior probability distribution when the haplotype states of the pedigree form an irreducible state space. In each update iteration, the haplotypes for the lineage individuals are updated conditional on the haplotypes assigned to the non-lineage parents, while the haplotypes of the non-lineage parents and all other pedigree individuals are unchanged. If the unchanged haplotypes are drawn from the stationary distribution, then after an update, all the haplotypes together represent a sample from the stationary distribution. This would by true for any blocking scheme, but we have chosen the lineage blocking scheme for ease of computation.

3 Results

Pedigree Simulations. In order to test the accuracy of our method, we simulated a set of pedigrees with their corresponding haplotypes. Given a pedigree, founder haplotypes were generated uniformly at random from the phased HapMap CEU haplotypes for Chromosome 1. We considered only common SNPs (with minor allele frequencies at least 0.05). We performed multiple trials, where each trial consisted of a distinct sample of SNPs chosen to have a specific density along the genome. This allowed us to vary the mean physical distance between neighboring SNPs. Each sample of SNPs was arbitrarily partitioned into non-overlapping blocks of a fixed length for haplotype inference.

The non-founders were generated in successive generations using Poisson-distributed recombinations (without interference), where the recombination rate was a function of the physical distance, such that there is an average of two recombinations on the length of Chromosome 1. Considering each non-founder in turn, we obtained one haplotype from each parent by uniformly choosing one of the parental haplotypes to provide the allele for the first SNP. Alleles for successive SNPs were chosen either to be non-recombinant or recombinant according to the recombination rate. We refer to the complete simulation output (of phased haplotypes) as the *gold-standard data*.

We chose pedigrees with fixed structure. For each pedigree we fixed the number and set of individuals to be genotyped in the data input to each of the phasing algorithms (and removed the phase information for all of the ungenotyped individuals).

L1 10 copies of a 20-individual family with 1 lineage and exactly 13 genotyped individuals (1000 blocks of 3 SNPs with 11kbp between SNPs)

S1 single family with 10 lineages and 59 individuals, exactly 24 of them being genotyped on 1000 blocks of 3 SNPs.

M1 5 copies of the family from S1, with exactly 24 individuals genotyped in each family (1000 blocks with 3 snps).

M2 10 copies of a 10-individual family with 2 lineages and exactly 5 genotyped individuals (10,000 blocks of 3 SNPs).

H1 single 16-individual, 2-lineage pedigree with half-siblings and exactly 9 genotyped individuals on 300 blocks of 5 SNPs.

Comparison. We compared our approach to two others, Merlin [5] and Superlink [10]. Both Merlin and Superlink perform a maximum-likelihood calculation on a similar graphical model of inheritance in a pedigree, where recombination rates and founder allele frequencies are given as fixed parameters of the model. However, Merlin employs a different elimination order for the EM algorithm than does Superlink and has an option for non-recombinant haplotype inference (this option was not used here, because it seemed to make little difference to inference accuracy). PhyloPed uses a graphical model of inheritance that is similar to that used by Merlin and Superlink but does not require the founder allele frequencies or recombination rates.
The input data consisted of the pedigree relationships and the genotype data for only the typed pedigree members. Merlin and Superlink were additionally provided with the correct recombination rates and with either an uninformative prior for the founder alleles or the perfect prior (i.e., the correct allele frequencies). Every phasing program was run on consecutive, non-overlapping blocks of k SNPs, and all programs ran on the same blocks. The output of each of the phasing programs was compared to the gold-standard data, and again the comparison used the same k-sized blocks. In cases where phasing programs provided a list of possible phasings, the first phasing was tested for accuracy. Accuracy was measured as the percentage of haplotype assignments in the phase estimate that matched the haplotypes in the gold-standard haplotype data. Notice that in this definition of accuracy the parental origin of the haplotype is irrelevant. Notice also that if the assumptions of PhyloPed are not satisfied, meaning that a particular family required recombinant haplotypes, then PhyloPed produced no estimate, and we conservatively chose to penalize our method by scoring the lack of a prediction as zero accuracy.

Simple vs Complex Pedigrees. For the single-lineage pedigree L1, we simulated blocks of size $k = 3$ with the average physical distance between SNPs being 11kbp. All methods estimated haplotypes with similar accuracy (Table 1, row L1). This suggests that the models have few practical differences on simple pedigrees.
For multi-lineage pedigrees S1, M1, M2, and H1, we see that PhyloPed outperforms the other methods (Table 1, rows S1,M1, and H1, and Fig. 3). Most of these results were generated for blocks with $k = 3$ SNPs, because larger blocks were infeasible for Merlin. However, pedigree H1 was simulated with $k = 5$ SNPs, and still PhyloPed outperforms the others.

Violations of Assumptions. We consider two violations of the assumptions for the three methods. First, we consider the performance of the three methods for different physical distances between SNPs in the block (resulting in a range of recombination rates). PhyloPed consistently outperforms Superlink and Merlin even as the recombination rate increases (Fig. 3). Second, it is possible for the founder allele frequencies to be unknown, even while the recombination rates may be known. We provide both Superlink and Merlin with uninformative founder allele frequencies (i.e. frequency 0.5 for all alleles). In this scenario, Merlin performs comparable to when it is given a perfect prior, but Superlink's accuracy decreases dramatically.

		Perfect Prior		Uninformative Prior	
Pedigree	**Method**	**Avg**	**Std-Dev**	**Avg**	**Std-Dev**
L1	PhyloPed	**0.867**	0.030	**0.867**	0.030
	Merlin	0.855	0.018	0.857	0.018
	Superlink	0.836	0.034	0.819	0.023
S1	PhyloPed	**0.809**	0.065	**0.809**	0.065
	Superlink	0.796	0.064	0.642	0.066
M1	PhyloPed	**0.808**	0.060	**0.808**	0.060
	Superlink	0.795	0.058	0.636	0.058
H1	PhyloPed	**0.816**	0.161	**0.816**	0.161
	Merlin	0.750	0.138	0.761	0.124
	Superlink	0.799	0.116	0.717	0.148

Table 1. Average accuracy and standard deviation. In all cases, PhyloPed dramatically outperforms Merlin. PhyloPed is substantially better than Superlink, when given a non-informative perfect prior. When given the perfect prior, Superlink performs no better than PhyloPed. In cases where Superlink and PhyloPed have comparable performance, we see that the uninformative prior particularly hurts Superlink's accuracy. Merlin was unable to execute S1 and M1 due to running time.

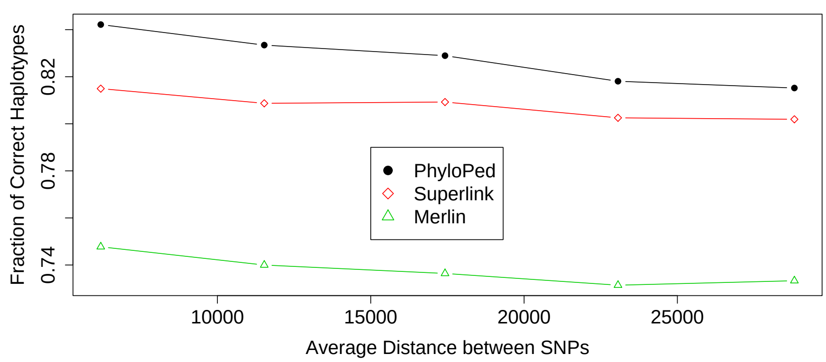

Fig. 3. Accuracy Against Recombination Rate. This plot shows results of 10,000 blocks for M2 the 2-lineage, 10-individual family. The accuracy of each method was computed for different physical distance between neighboring SNPs.

4 Discussion

We have introduced PhyloPed which leverages the population genetics of the founders to produce superior haplotype estimates for multi-lineage pedigrees. Specifically, PhyloPed assumes that the founder haplotypes are drawn from a perfect phylogeny and that haplotypes are inherited without recombination in the pedigree. As we have shown, this approach works very well for short regions with dense SNPs.

In addition to the perfect phylogeny model, there are several other reasons that PhyloPed outperforms other methods. Intuitively, Occam's razor suggests that our method would be preferable on blocks having little recombination. Assuming no recombination provides not only fewer phasing options to consider but also fewer parameters and less over-fitting. PhyloPed requires no prior information, either for the recombination rates or for the founder allele frequencies, which avoids the possibility that an inaccurate prior might mislead our algorithm.

Many factors influence the accuracy of haplotype estimation, including the complexity of the pedigree, the number and relationships of genotyped individuals, and the number of linked SNPs. The number of genotyped individuals in the pedigree and their relationships with the other pedigree members influences the number of constraints available for haplotype estimation. Typically, having genotypes for more individuals yields better haplotype estimates. Similarly, simultaneous phasing of larger numbers of linked SNPs can reveal more haplotype information, provided that the pedigree is not so large that the computational burden is infeasible. This paper has focused on inference in deep and complex pedigrees and partitioned the genome into blocks before phasing. In order to properly treat the whole genome, future research should consider partitioning schemes and methods for producing whole genome haplotype estimates from the estimates for each partition. One possible approach is using an HMM, similar to some of the tag SNP research for unrelated individuals [12].

Pedigrees should not be made unnecessarily complex. Multiple-lineage pedigrees are only useful in the case where each founding lineage provides information about either the phenotype or the relatedness of genotyped individuals. For example, estimation of haplotypes in a nuclear family whose members are genotyped and phenotyped would not benefit from the introduction of grandparents whose additional degrees of freedom provide no additional constraints on the haplotypes or phenotypes. However, if a pair of grandparents are the common ancestors of this nuclear family and another genotyped family, then the grandparents' presence in the pedigree (along with the additional family) would provide useful constraints.

Within 10 years, it is plausible that cost-effective sequencing methods will provide haplotypes for samples. However, the availability of haplotypes for some individuals in the pedigree does not obviate the need for phasing the unsampled individuals. Algorithms such as the one presented here provide consistent resolutions for the ancestry of each haplotype and yield haplotype assignments for pedigree members whose DNA is unavailable. Notice that phasing the unsampled individuals is (nearly) equivalent to the problem of finding the recombinations that produced the observed haplotypes.

There are a number of open problems, and further work is needed to take full advantage of the information provided by both genotype and sequence data. For instance, other population genetic models could be applied to the founder haplotypes. These models have the added benefit of inferring which recombinations occurred in the pedigree versus in the ancestral haplotypes. Another important question for sequencing data is how best to take advantage of known information about identity by descent.

PhyloPed Implementation and Supplementary Materials. Available at: `http://phyloped.icsi.berkeley.edu/phyloped/`

Acknowledgments. We thank the reviewers for their insightful comments. B.K. was supported under a National Science Foundation Graduate Research Fellowship. E.H. and R.M.K. were supported by NSF grant IIS-0513599. Eran Halperin is a faculty fellow of the Edmond J. Safra Bioinformatics program at Tel Aviv University.

References

1. Mark Abney, Carole Ober, and Mary Sara McPeek. Quantitative-trait homozygosity and association mapping and empirical genomewide significance in large, complex pedigrees: Fasting serum-insulin level in the hutterites. *The American Journal of Human Genetics*, 70(4):920 – 934, 2002.
2. J.C. Barrett, S. Hansoul, D.L. Nicolae, J.H. Cho, R.H. Duerr, J.D. Rioux, S.R. Brant, M.S. Silverberg, K.D. Taylor, M.M. Barmada, and et al. Genome-wide association defines more than 30 distinct susceptibility loci for crohn's disease. *Nature Genetics*, 40:955–962, 2008.
3. J.T. Burdick, W. Chen, G.R. Abecasis, and V.G. Cheung. In silico method for inferring genotyeps in pedigrees. *Nature Genetics*, 38:1002–1004, 2006.
4. C. Cannings and N. A. Sheehan. On a Misconception About Irreducibility of the Single-Site Gibbs Sampler in a Pedigree Application. *Genetics*, 162(2):993–996, 2002.
5. Wei-Min Chen and Gonçalo R. Abecasis. Family-based association tests for genomewide association scans. *American Journal of Human Genetics*, 81:913 – 926, 2007.
6. MJ Daly, JD Rioux, SF Schaffner, TJ Hudson, and ES Lander. High-resolution haplotype structure in the human genome. *Nature Genetics*, 29(2):229–32, Oct 2001.
7. Z. Ding, V. Filkov, and D. Gusfield. A linear-time algorithm for perfect phylogeny haplotyping. *Journal of Computational Biology*, (2):522–553, 2006.

8. R.C. Elston and J. Stewart. A general model for the analysis of pedigree data. *Human Heredity*, 21:523–542, 1971.
9. E. Eskin, E. Halperin, and R. Karp. Efficient reconstruction of haplotype structure via perfect phylogeny. *Journal of Bioinformatics and Computational Biology*, 1(1):1–20, 2003.
10. M. Fishelson, N. Dovgolevsky, and D. Geiger. Maximum likelihood haplotyping for general pedigrees. *Human Heredity*, 59:41–60, 2005.
11. Gusfield. Haplotyping as perfect phylogeny: Conceptual framework and efficient solutions. In *Proceedings of the 6th Annual International Conference on (Research in) Computational (Molecular) Biology*, 2002.
12. E. Halperin, G. Kimmel, and R. Shamir. Tag SNP selection in genotype data for maximizing SNP prediction accuracy. *Bioinformatics*, 21(Suppl. 1):i195–i203, 2005.
13. C.S. Jensen and A. Kong. Blocking gibbs sampling for linkage analysis in large pedigrees with many loops. *American Journal of Human Genetics*, 65, 1999.
14. E.S. Lander and P. Green. Construction of multilocus genetic linkage maps in humans. *Proceedings of the National Academy of Science*, 84(5):2363–2367, 1987.
15. S. L. Lauritzen and N. A. Sheehan. Graphical models for genetic analysis. *Statistical Science*, 18(4):489–514, 2003.
16. A. Piccolboni and D. Gusfield. On the complexity of fundamental computational problems in pedigree analysis. *Journal of Computational Biology*, 10(5):763–773, 2003.
17. Nathan B. Sutter, Carlos D. Bustamante, Kevin Chase, Melissa M. Gray, Keyan Zhao, Lan Zhu, Badri Padhukasahasram, Eric Karlins, Sean Davis, Paul G. Jones, Pascale Quignon, Gary S. Johnson, Heidi G. Parker, Neale Fretwell, Dana S. Mosher, Dennis F. Lawler, Ebenezer Satyaraj, Magnus Nordborg, K. Gordon Lark, Robert K. Wayne, and Elaine A. Ostrander. A Single IGF1 Allele Is a Major Determinant of Small Size in Dogs. *Science*, 316(5821):112–115, 2007.
18. A. Thomas, V. Abkevich A. Gutin, and A. Bansal. Multilocus linkage analysis by blocked gibbs sampling. *Statistics and Computing*, 10(3):259–269, 2000.