Pedigree Reconstruction Using Identity by Descent

Bonnie Kirkpatrick[1], Shuai Cheng Li[2], Richard M. Karp[3], and Eran Halperin[4]

[1] Electrical Engineering and Computer Sciences, University of California, Berkeley, and International Computer Science Institute, Berkeley, bbkirk@eecs.berkeley.edu.
[2] International Computer Science Institute, Berkeley, scli@icsi.berkeley.edu.
[3] Electrical Engineering and Computer Sciences, University of California, Berkeley, and International Computer Science Institute, Berkeley, karp@cs.berkeley.edu.
[4] Tel Aviv University, Tel Aviv, Israel, and International Computer Science Institute, Berkeley, heran@icsi.berkeley.edu.

Abstract. Can we find the family trees, or pedigrees, that relate the haplotypes of a group of individuals? Collecting the genealogical information for how individuals are related is a very time-consuming and expensive process. Methods for automating the construction of pedigrees could stream-line this process. While constructing single-generation families is relatively easy given whole genome data, reconstructing multi-generational, possibly inbred, pedigrees is much more challenging.
This paper addresses the important question of reconstructing monogamous, regular pedigrees, where pedigrees are regular when individuals mate only with other individuals at the same generation. This paper introduces two multi-generational pedigree reconstruction methods: one for inbreeding relationships and one for outbreeding relationships. In contrast to previous methods that focused on the independent estimation of relationship distances between every pair of typed individuals, here we present methods that aim at the reconstruction of the entire pedigree. We show that both our methods out-perform the state-of-the-art and that the outbreeding method is capable of reconstructing pedigrees at least six generations back in time with high accuracy.
The two programs are available at http://cop.icsi.berkeley.edu/cop/.

1 Introduction

Pedigrees, or family trees, are important in computer science and in genetics. The pedigree graph encodes all the possible Mendelian inheritance options, and provides a model for computing inheritance probabilities for haplotype or genotype data. Even thirty years after the development of some of the first pedigree algorithms [20, 11], pedigree graphical models continue to be a challenging graphical model to work with. Known algorithms for inheritance calculations are either exponential in the number of individuals or exponential in the number of loci [21]. There have been numerous and notable attempts to increase the speed of these calculations [32, 1, 12, 7, 14, 23, 9]. Recent work from statistics has focused on fast and efficient calculations of linkage that avoid the full inheritance calculations [5, 38]. Recent contributions to genetics from pedigree calculations include fine-scale recombination maps for humans [8], discovery of regions linked to Schizophrenia [26], discovery of regions linked to rare Mendelian diseases [27], and insights into the relationship between cystic fibrosis and fertility [13].

Manual methods for constructing human pedigree graphs are very tedious. It requires careful examination of genealogical records, including marriage records, birth dates, death dates, and parental information found in birth certificates. Medical researchers then must carefully check records for consistency, for instance making sure that two married individuals were alive at the same time and

making sure that children were conceived while the parents were alive. This process is very time consuming. Despite the care taken, there are sometimes mistakes [4, 25, 34].

For constructing non-human pedigrees, of diploid organisms, it is often impossible to know the pedigree graph since there are no genealogical records [2, 6]. In this case it is particularly important to develop methods of automatically generating pedigrees from genomic data.

The problem of reconstructing pedigrees from haplotype or genotype data is not new. The oldest such method that the authors know of is due to Thompson [37]. Her approach is essentially a structured machine learning approach where the aim is to find the pedigree graph that maximizes the probability of observing the data under the pedigree model, also called the likelihood of the pedigree. (This approach is directly analogous to maximum likelihood methods for phylogenetic reconstruction which also try to find the phylogenetic tree that maximize the likelihood.) Notice that this method reconstructs both the pedigree graph and the ancestral haplotypes which is a very time-consuming step. Thus, this approach is limited to extremely small families, perhaps 4-8 people, since the algorithms for computing the likelihood of a fixed pedigree graph are exponential [21] and there are an exponential number of pedigree graphs to consider [35].

The current state-of-the-art method is an HMM-based approximation of the number of meioses separating a pair of individuals [33]. This approach dispenses with any attempt to infer haplotypes of ancestral individuals, and instead focuses on the number of generations that separate a pair of individuals. In this approach the hidden states of the HMM represent the identity-by-descent (IBD) of a pair of individuals. Two individuals are identical-by-descent for a particular allele if they each have a copy of the same ancestral allele. The probability of the haplotype data is tested against a particular type of relationship. The main draw-back of this approach is that it may estimate a set of pair-wise relationships that are inconsistent with a single pedigree relating all the individuals.

Thatte and Steel [36] examined the problem of reconstructing arbitrary pedigree graphs from a synthetic model of the data. Their method used an HMM model for the ancestry of each individual to show that the pedigree can be reconstructed only if the sequences are sufficiently long and infinitely dense. Notice that this paper uses an unrealistic model of recombination where every individual passes on a trace of their haplotypes to all of their descendants. Kirkpatrick [18] introduced a more simple, more general version of the reconstruction algorithm introduced by Thatte and Steel.

Attempts to construct sibling relationships are known to be NP-hard, and attempts to infer pedigrees by reconstructing ancestral haplotypes are be NP-hard. Two combinatorial versions of the sibling relationship problem were proven to be NP-hard, both whole- and half-sibling problem formulations [2, 31]. If ancestral haplotypes are reconstructed in the process of inferring a pedigree, as in Thompson's structured machine learning approach, then the inheritance probabilities of data must be computed on the pedigree graph. For instance, we might want to compute the likelihood, or the probability of observing the data given inheritance in the pedigree. This calculation is NP-hard for both genotype [29, 22] and haplotype [17] data. This means that any efficient pedigree reconstruction method will need to find ways to avoid both these hardness problems.

Our contribution to pedigree reconstruction is two algorithms that avoid the exponential likelihood calculations. We do this by specifically *not* reconstructing ancestral haplotypes and by *not* trying to optimize sibling groups. We use estimates of the length of genomic regions that are shared identical-by-descent. In two related individuals, a region of the genome is identical-by-descent (IBD) if and

only if a single ancestral haplotype sequence was the source of the sequence inherited in the two individuals. The length of IBD regions gives a statistic that accurately detects sibling relationships at multiple generations. We have two algorithms: one for constructing inbred pedigrees (CIP) and one for constructing outbred pedigrees (COP). For our outbreeding algorithm the statistic is testable in polynomial time. For our inbreeding algorithm, the statistic is computable in time dependent on the number of meioses in the predicted pedigree. Our outbreeding method works to reconstruct at least six generations back in time. Both methods are more accurate than the state-of-the-art method by Stankovich et al. [33].

The remainder of the paper is organized into sections on pair-wise IBD, practical reconstruction algorithms, and results. The section on pair-wise IBD considers the expected length of a genomic region shared between a pair of individuals. This establishes the limits of reconstruction methods that are based only on pair-wise relationships. The section on practical algorithms introduces our CIP and COP algorithms, which go beyond pair-wise relationships and actually use transitive relationship information to infer a pedigree graph. The results section considers simulation results and results running the algorithm on several HapMap Phase III populations.

2 Background

A pedigree graph has diploid individuals as nodes and edges from parents to children. The edges are typically implicitly directed down from parent to child, without drawing the actual direction arrow on the edge. Circle nodes are females, boxes are females. Let the generations be numbered backwards in time, with larger numbers being older generations. Let g be the number of generations of individuals in the graph. For example, if $g = 1$, then we are discussing only the extant individuals, whereas if $g = 2$ the graph contains the extant individuals and their parents.

In this paper, we will only consider monogamous, regular pedigrees, where a pedigree is *regular* when individuals only mate with other individuals at the same generation. Of course, a pedigree is *monogamous* if and only if every individual mates with at most one other individual, so that there are no half-siblings.

Recombination along the genome is typically modeled as a Poisson process, where the distance between recombination breakpoints is drawn from an exponential distribution. The mean of the exponential is a function of the recombination rate [10, 3]. This is a model for recombination without interference, where interference means that the presence of one recombination breakpoint suppresses the occurrence of breakpoints in neighboring regions of the sequence [24]. The simulation and experimental results seem to support the use of the simplifying assumption made by using the Poisson model for recombination, however relaxing this assumption might be one way to improve on the model.

3 A Lower Bound for Pair-Wise Relationships with Out-breeding

In order to shed light on the problem we first provide a lower bound on the best that one could do in pedigree reconstruction. Stankovich et al. [33] have been able to detect up to 3rd cousins (or relationships of 8 total meioses). We claim that this should be near optimal in the case of an

infinite population size. Notice that in the infinite population size, there is no inbreeding. Therefore, the graph relating people has a path-like subgraph connecting every pair of individuals (i.e. the subgraph is a path having exactly two founders whose adjacent edges can be contracted to form a simple path). This implies that in order to estimate pedigree graphs that are more accurate than the conglomerate of a set of pair-wise relationship estimates, we need to exploit features of the relationships that are not simply outbred paths between pairs of individuals. Specifically, we need to consider sets of individuals and the graphs that connect them, and we need to consider graphs, not paths, that connect pairs of individuals. This means that we need to be considering inbreeding and transitive relationships (i.e. person a is related to person c through person b).

Now, we derive a lower bound on the pair-wise outbred relationships. In an infinite population, consider two individuals i, and j, where their most recent common ancestor is g generations ago. For instance, if $g = 2$ they are siblings. Note that they have two common ancestors in this case. For general g, each individual has 2^g ancestors, where exactly two of them are shared across i and j; this is where we use the fact that the population is infinite and monogamous, since the probability of having more than two shared ancestors is zero and monogamy ensures that there are at least two shared ancestors.

Each of the ancestors of i and j has two haploids. Each of the haploids arrived from a different pedigree. Consider only the haploids that arrived from the shared pedigree (the case $g = 2$, i.e. siblings, is different since there there is IBD sharing on both haploids of i and j). These haploids of i and j are generated by a random walk over the ancestors of i and j in the gth generation. The total number of *haploid* ancestors in that generation is 2^g for each of i and j. Out of those, four are shared across i and j (two shared ancestors, each has two haploids). Let k be the number of meioses separating individuals i and j, where $k = 2(g-1)$. For this reason, the expected number of bases shared between i and j is $\frac{4L}{2^k} = \frac{L}{2^{k-2}}$, where L is the length of the genome.

On the other hand, we can calculate the average length of a shared region between the two haploids. The number of recombinations across all generations is Poisson distributed with parameter krL, where r is the recombination rate, L is the length of the genome. Now, the length, X, of a shared region that originated from one of the four shared haploids is $X_1 + X_2$ where $X_i \sim exp(kr)$. Notice that X_i is the length of the IBD region conditioned on starting at an IBD position. Therefore from an arbitrary IBD position, we need to consider the length of the IBD region before arriving at that position, X_1, and the length after that position, X_2. So the expected length, $E[X]$, is $\frac{2}{kr}$. Since the probability to move from one shared haploid to another is negligible, we get that this is the expected length of a shared region.

Now, if t_k is the expected number of regions shared between two individuals separated by k meioses, we know that $t_k \frac{2}{kr} = \frac{L}{2^{k-2}}$, and therefore, $t_k = \frac{krL}{2^{k-1}}$, where rL is the expected number of recombinations after one generation. Therefore, $t_{10} < 1$ since $rL = 30$, and it is impossible to detect a pair-wise relationship with high probability between 4th cousins.

This is not to say that it is impossible to accurately construct a 6-generation pedigree, only that it is impossible to accurately construct a 6-generation pedigree from pair-wise relationship estimates. As noted earlier, to get accuracy on deep pedigrees, we need to consider relationships on sets of individuals, inbreeding and transitive relationships.

4 Algorithms for Constructing Pedigrees

The principle innovation of this method is to reconstruct pedigree graphs *without* reconstructing the ancestral haplotypes. This is the innovation that allows this algorithm to avoid the exponential calculation associated with inferring ancestral haplotypes, and allows the algorithm to be efficient.

The approach we employ is a *generation-by-generation* approach. We reconstruct the pedigree backwards in time, one generation at a time. Of course if we make the correct decisions at each generation, then we will construct the correct pedigree. However, since we use the predictions at previous generations to help us make decisions about how to reconstruct subsequent generations, we can accumulate errors as the algorithm proceeds backwards in time.

Given a set of extant individuals with haplotype information available, we want to reconstruct their pedigree. We construct the pedigree recursively, one generation at a time. For example, the first iteration consists of deciding which of the extant individuals are siblings. The next iteration would determine which of the parents are siblings (yielding cousin relationships on the extant individuals).

At each generation, we consider a *compatibility* graph on the individuals at generation g, where the nodes are individuals and the edges are between pairs of individuals that could be siblings. The presence or absence of edges will be determined by a statistical test, discussed later. For the moment, assume that we have such a graph.

Now, we will find sibling sets in the compatibility graph. We do this by partitioning the graph into disjoint sets of vertices with the property that each set in the partition has many edges connecting its vertices while there are few edges connecting vertices from separate sets in the partition. Of course any partitioning method can be used, and later we will introduce a partitioning heuristic. For rhetorical purposes, we will now discuss how to use a Max-Clique algorithm to partition the graph. The graph is partitioned by the following iterative procedure. Iteratively, find the Max-Clique, for all the individuals in the Max-Clique, make them siblings, by creating monogamous parents in generation $g+1$. Remove those Max-Clique individuals from the graph. Now, we can iterate, by finding the next Max-Clique and again creating a sibling group, etc.

Next, we consider how to create the edges in the compatibility graph. Let individuals k and l be in generation g. Recall that we have an edge in the compatibility graph if k and l could be siblings. To determine this, we look at pairs i and j of descendants of k and l, respectively. Let $\hat{s}_{ij}$ be the observed average length of shared segments between haplotyped individuals i and j. This can be computed directly from the given haplotype data and need only be computed once as a preprocessing step for our algorithm. Now, for a pair of individuals k and l in the oldest reconstructed generation, $X_{i,j}$ is the random variable for the length of a shared region for individuals i, j under the pedigree model that we have constructed so far. Later, we will discuss two models for $X_{i,j}$. For now, consider the test for the edge (k, l)

$$v_{k,l} = \frac{1}{|D(k)||D(l)|} \sum_{i \in D(k)} \sum_{j \in D(l)} \frac{(\hat{s}_{ij} - \mathbb{E}[X_{ij}])^2}{var(X_{ij})} \tag{1}$$

where $D(k)$ is the set of extant individuals descended from ancestor k, and $D(k)$ is known based on the pedigree we have constructed up to this point. We compute $v_{k,l}$, making edges when $v_{k,l} < c$ for all k, l in the oldest generation, g, for some threshold c. Notice that this edge test is similar to

a χ^2 test but does not have the χ^2 null distribution, because the term in the sum will not actually be normally distributed. We choose the the threshold, c, empirically by simulating many pedigrees and choosing the threshold which provides the best reconstruction accuracy.

Now, we need to calculate $\mathbb{E}[X_{i,j}]$ and $Var(X_{i,j})$. We propose two models for the random variable X_{ij}, the outbred model (COP) and the inbred model (CIP). The outbred, COP, model only allows prediction of relationships between two individuals that are unrelated at all previous generations. The inbred model, CIP, allows prediction of a relationship that relates two individuals already related in a previous generation.

4.1 IBD Model for Constructing Outbred Pedigrees (COP)

To obtain the edges in the compatibility graph, we do a test for relationship-pairs of the form shown in Figure 1. If a pair of extant individuals i and j are related at generation g via a single ancestor at that generation, then the length of the regions they share IBD will be distributed according to the sum of two exponential variables, specifically, $exp(2(g-1)\lambda)$. This is the waiting time, where time corresponds to genome length, for a random walk to leave the state of IBD sharing. So, we have $X_{ij} = X_1 + X_2$ where $X_i \sim exp(2(g-1)\lambda)$. Once again, we must consider the sum of the two exponential random variables, just as we did in Section 3. Due to these random variables being exponentially distributed, we can quickly analytically compute $\mathbb{E}[X_{ij}]$ and $Var(X_{ij})$. Of course, the edges created respect the outbreeding constraint, such that a pair of individuals, k and l at the gth generation can only have an edge between them in the compatibility graph if none of the extant individuals in $D(k)$ and $D(l)$ are related to each other at a previous generation.

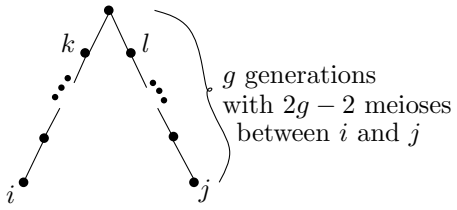

Fig. 1. Pair of Individual Related at Generation g. To test whether individuals k and l are siblings at generation g, we look at the distribution on the length of genetic regions shared IBD between all pairs of i and j descended from k and l, respectively.

4.2 IBD Model for Constructing Inbred Pedigrees (CIP)

We will do a random-walk simulation to allow for inbreeding, resulting in an algorithm with exponential running-time. The number of states in the IBD process is exponential in the number of meioses in the graph relating individuals i and j. So, the random-walk simulation is exponential in the size of the inferred pedigree.

For individuals k and l in generation g, and their respective descendants i and j, we consider the case given in Figure 2. The triangles represent the inferred sub-pedigree containing all the descendants

of the individual at the point of the triangle, and individuals at the base of the triangle are extant individuals. Note that the triangles may overlap, indicating shared ancestry at an earlier generation (i.e. inbreeding).

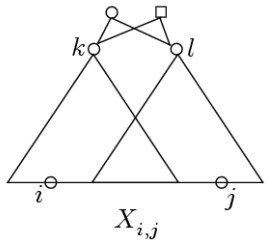

Fig. 2. Test Case. Specific individuals in the pedigree are indicated with either circles or squares. The triangle represents all the descendants of a particular individual. This represents the case where individuals i and j are cousins via the oldest generation.

Brief Description of the IBD Simulation. Let $X_{i,j}$ be the length of a shared region based on the pedigree structure of the model. In order to estimate this quantity, we can sample random walks in the space of inheritance possibilities. Specifically, consider the inheritance of alleles at a single position in the genome. When there are n non-founder individuals, define an inheritance vector as a vector containing $2n$ bits, where each pair of bits, $2i$ and $2i+1$, represents the grand-parental origin of individual i's two alleles. Specifically, bit $2i$ represents the maternal allele and is zero if the grand-paternal allele was inherited and is one otherwise. Similarly, bit $2i+1$ represents the paternal allele of individual i. The set of possible inheritance vectors comprise the 2^{2n} vertices of a $2n$-dimensional hypercube, where n is the number of non-founders in the pedigree. A random walk on the hypercube represents the recombination process by choosing the inheritance vectors of neighboring regions of the genome.

Given an inheritance vector, we can model the length, in number of positions, of the genomic region that is inherited according to that inheritance vector. The end of that genomic region is marked by a recombination in some individual, and constitutes a change in the inheritance vector. The random walk on the hypercube models the random recombinations, while the length of genomic regions are modeled using an exponential distribution. This model is the standard Poisson model for recombinations. Details can be found below.

Poisson Process. Given a pedigree and individuals of interest i and j, we will compute the distribution on the length of shared regions. Here we mean sharing to be a contiguous region of the genome for which i and j have at least one IBD allele at each site.

We can model the creation of a single zygote (i.e. haplotype) as a Poisson process along the genome where the waiting time to the next recombination event is exponentially distributed with intensity $\lambda = -ln(1-\theta)$ where θ is the probability of recombination per meiosis (i.e. per generation, per chromosome) between a pair of neighboring loci. For example, if we think of the genome as being composed of 3000 blocks with each block being 1Mb in length and the recombination rate $\theta = 0.01$

between each pair of neighboring blocks, then we would expect 30 recombinations per meiosis, and the corresponding intensity for the Poisson process is $\lambda = 0.01$.

Now, we have $2n$ meioses in the pedigree, with each meiosis creating a zygote, where n is the number of non-founder individuals. Notice that at a single position in the genome, each child has two haplotypes, and each haplotype chooses one of the two parental alleles to copy. These choices are represented in an inheritance vector, a binary vector with $2n$ entries. The 2^{2n} possible inheritance vectors are the vertices of a $2n$-dimensional hypercube. We can model the recombination process as a random walk on the hypercube with a step occurring each time there is a recombination event. The waiting time to the next step is drawn from $exp(2n\lambda)$, the meiosis is drawn uniformly from the $2n$ possible meioses, and a step taken in the dimension that represents the chosen meiosis. The equilibrium distribution of this random walk is uniform over all the 2^{2n} vertices of the hypercube.

Detailed IBD Simulation. Recall that we are interested in the distribution of the length of a region that is IBD. Recall that IBD is defined as the event that a pair of alleles are inherited from the same founder allele. For individuals i and j, let D be the set of hypercube vertices that result in i and j sharing at least one allele IBD. Given x_0 a hypercube vertex drawn uniformly at random from D, we can compute the hitting time to the first non-IBD vertex by considering the random walk restricted to $D \cup \{d\}$ where d is an aggregate state of all the non-IBD vertices. The hitting time to d is the quantity of interest. In addition, we also need to consider the length of the shared region before reaching x_0, which is the time reversed version of the same process, for the same reason that we summed two exponential random variables while computing the lower bound in Section 3.

The transition matrix for this IBD process is easily obtained as $Pr[x_{i+1} = u | x_i = v] = \frac{1}{2n}$ when vertices u and v differ by exactly one coordinate, and $Pr[x_{i+1} = u | x_i = v] = 0$ otherwise. Transitions to state d are computed as $Pr[x_{i+1} = d | x_i = u] = 1 - \sum_{v \in D} Pr[x_{i+1} = v | x_i = u]$.

Now we can either analytically compute the hitting time distribution or estimate the distribution by simulating paths of this random walk. Since the number of IBD states may be exponential, it may be computationally infeasible to find eigenvectors and eigenvalues of the transition matrix [10]. We choose to simulate this random walk and estimate the distribution. This simulation is at worst exponential in the number of individuals.

4.3 Heuristic Graph Partitioning Method

The Max-Clique algorithm was only used to illustrate the graph partitioning method. For both the COP and CIP algorithms we use an efficient heuristic for partitioning the vertices of the *compatibility* graph. This method is beneficial, because it looks for densely connected sets of vertices, rather than cliques, which allows for missing edges.

The algorithm is used to partition the vertices, $V(G^g)$, of graph G^g, into a partition $P = \{P_1, P_2, ..., P_C\}$, where $P_i \cap P_j = \emptyset$ for all i, j, and $V(G^g) = \cup_{i=1}^{C} P_i$. For a given partition set, let E_i be the edges of the subgraph induced by vertices P_i. We wish to find a partition such that each set in the partition is a clique or quasi-clique of vertices. The objective function is to find a partition that maximizes $\sum_{i=1}^{C} (a+1)|E_i| - \binom{|P_i|}{2}$ where $a = 0.1$ is a parameter of the algorithm. This objective function is chosen, because it is equivalent to $\sum_{i=1}^{C} a|E_i| - \left(\binom{|P_i|}{2} - |E_i|\right)$, where the term in parentheses is the

number of missing edges in the clique. Details of the partitioning method can be found in Karp and Li [16].

The running-time of this graph-partitioning heuristic largely determines the running-time of the pedigree reconstruction algorithm. The partitioning algorithm runs in polynomial time in the size of the graph, if the size of each set in the partition is constant. The step of creating the graph is polynomial in the size of the previous generation graph. Clearly it is possible, if no relationships are found, for the size of the graph at each generation to double. So, in the worst case, this algorithm is exponential. However, in practice this method performs quite quickly for constructing eight-generation pedigrees on large inputs.

5 Results

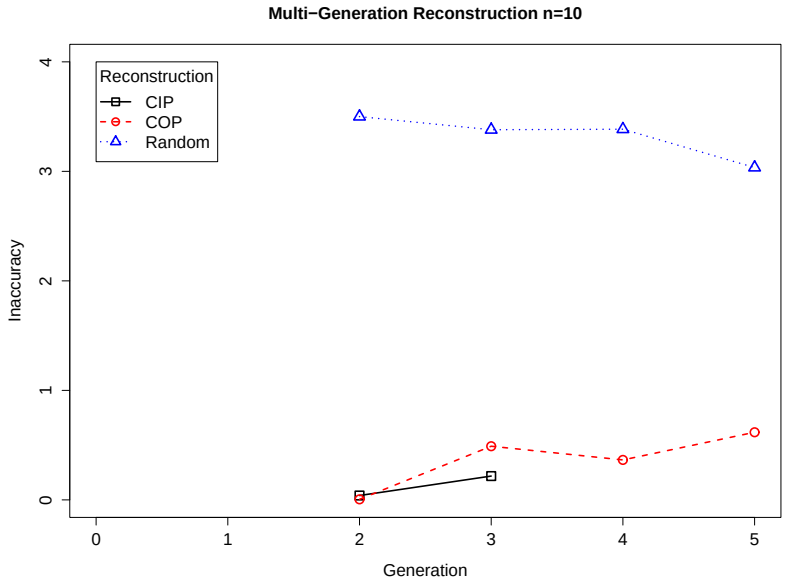

Fig. 3. Reconstruction under High Inbreeding. Here the pedigrees were simulated with a fixed population size of $n = 10$ individuals per generation. Over multiple generations, this results in a high level of inbreeding. The inaccuracy on the y-axis is measured by computing the kinship distance. (Reconstruction accuracy of 50 simulated pedigrees were averaged.)

Pedigrees were simulated using a variant of the Wright-Fisher model with monogamy. The model has parameters for a fixed population size, n, a Poisson number of offspring λ, and a number of generations g. In each generation g, the set of n_g individuals is partitioned into $n_g/2$ pairs, and for each pair we randomly decide on a number of offspring using the Poisson distribution with expectation $\lambda = 3$.

The human genome was simulated as 3,000 regions, each of length 1Mb, with recombination rate 0.01 between each region and where each founder haplotype had a unique allele for each region.

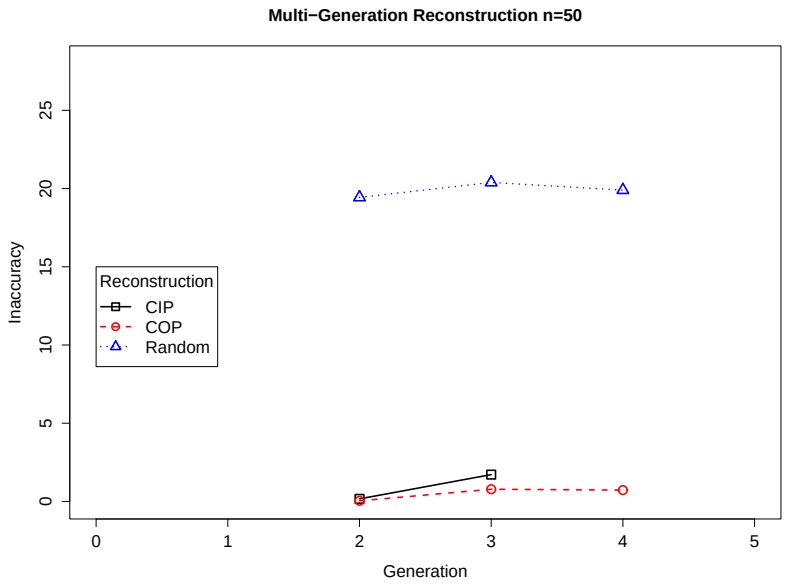

Fig. 4. Reconstruction under Less Inbreeding. Pedigrees here were simulated with a population size of $n = 50$. The y-axis show inaccuracy measured by kinship distance. (Reconstruction accuracy of 50 simulated pedigrees were averaged.)

The assumption here is that IBD information can be given as input to our method. This is not highly restrictive, since if two individuals have some phasing of their genotypes for which there is a common haplotype for a 1Mb region (typically 500 SNPs), they are likely IBD. Notice that Stankovich et al. require haplotypes as input to their method [33], which can be thought of as a form of IBD input.

In each experiment we end up having the true pedigree generated by the simulation, as well as an estimated pedigree. We evaluate the accuracy of the estimated pedigree by comparing the kinship matrices of the two pedigrees. Kinship is a model-based quantity defined as the frequency of IBD allele-sharing between a pair of individuals in a pedigree (averaged over the alleles of each individual). Since both pedigrees have the same set of haplotyped individuals, the comparison we consider is an L_1 distance between the kinship estimates of those individuals. Let K^P and K^Q be the kinship matrices of the actual pedigree P and the estimated pedigree Q, respectively. Then the evaluation method is

$$\sum_{i<j} |K^P_{i,j} - K^Q_{i,j}|$$

for haplotyped individuals i and j.

Selecting Parameters. Notice that there is some interaction between setting threshold c for creating edges in the compatibility graph and the parameter a for how much the quasi-cliques can differ from actual cliques. For a fixed choice of parameter a, we simulated pedigrees and reconstructed them in order to choose the threshold c that gave the best performance. There is competition between how much the quasi-cliques differ from cliques, i.e. how large a is, and the permissiveness of the

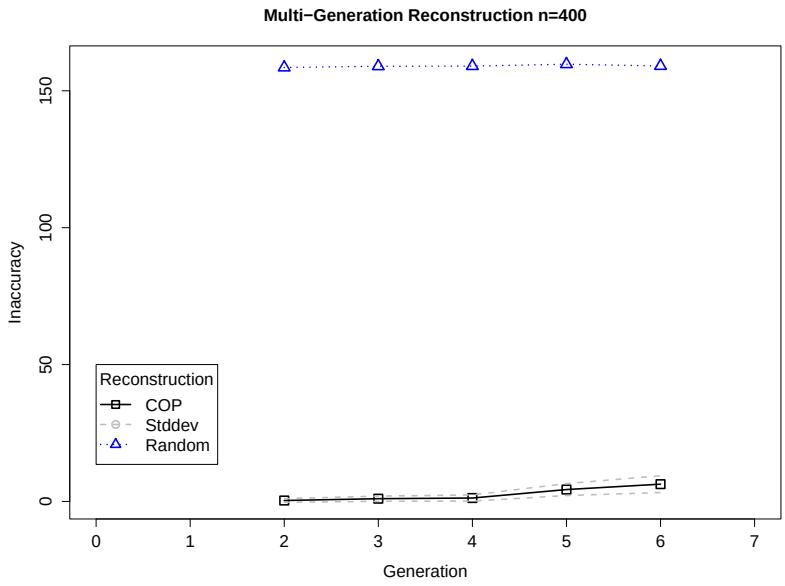

Fig. 5. Reconstruction for Deep Pedigrees. Pedigrees here were simulated with a population size of $n = 400$. (Reconstruction accuracy of 50 simulated pedigrees were averaged.)

edge-creation threshold. The larger a is the fewer edges must be created and the smaller c must be in order to maintain accuracy. (Data not shown.) However, for both algorithms we find that $a = 0.01$ and $c = 0.7$ yield the best performance.

Accuracy of COP versus CIP. We compare the COP and CIP methods on inbred pedigree simulations with high and moderate inbreeding, respectively $n = 10$ and $n = 50$, in Figures 3 and 4. These figures show the kinship-based inaccuracy on the y-axis and the number of generations in the reconstructed pedigree on the x-axis. As the depth of the estimated pedigree increases the error in the kinship of the estimated pedigree increases. However the accuracy is still much better than the accuracy of a randomly constructed pedigree, which is the highest, i.e. worst, line in each figure. CIP performs better on more inbred populations, which we would expect from the modeling assumptions. The running time on the 50 replicates of the $n = 50$ pedigree was 455.32s for COP and 1818.56s for CIP as a total running-time for all the simulated generation sizes.

Size of Reconstructed Pedigrees. Both the COP and CIP methods can reconstruct pedigree with four generations. The COP method for outbred pedigrees can reconstruct pedigrees going back to the most-recent common ancestor of the extant individuals. Provided with enough individuals, the method can construct pedigrees many generation deep. For example, given 400 individuals the method can construct 6 generations. As Figure 5 shows, the performance relative to a random reconstruction method is very good and so is the variance of the COP reconstruction method.

Comparison with GBIRP. We compare our two methods with the state-of-the-art method, called GBIRP, by Stankovich et al. [33]. Since GBIRP is limited to small pedigrees, we compare the methods on three-generation simulated pedigrees with population size $n = 10$. The simulated pedigrees

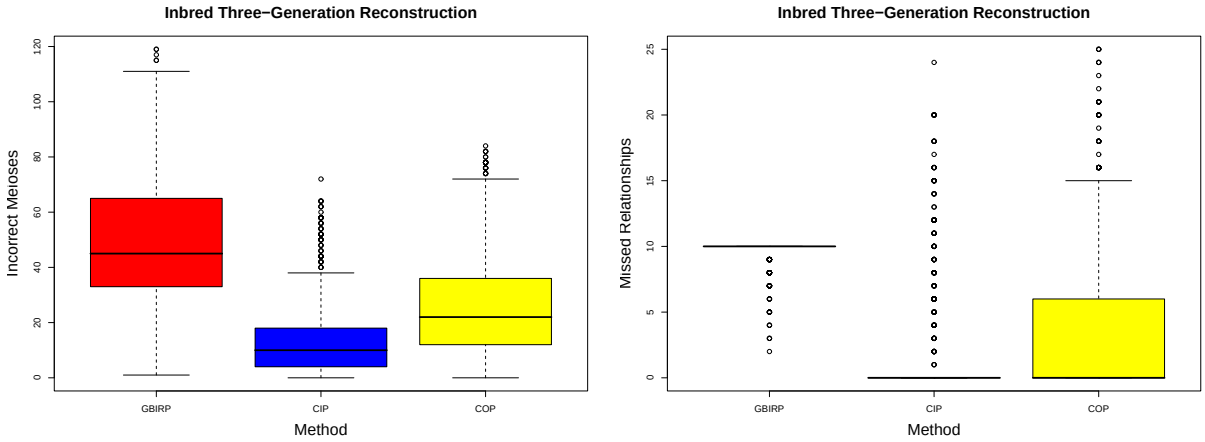

Fig. 6. Comparison with GBIRP on Inbred Simulations. The three-generation pedigrees here were simulated with $n = 10$ extant individuals, since GBIRP could not process larger pedigrees. The accuracy of 1000 simulated pedigrees were computed and plotted. Here the CIP method performs the best, i.e. closest to zero on both plots.

are connected graphs, so we can look at two accuracy measures, relationships that are mis-specified and relationships that should have been predicted but where not. GBIRP predicts meiosis distance, g_{ij}, between pairs of individuals, i, j, without inferring pedigree relationships. In order to compare GBIRP with the actual pedigree, we extract the minimum number of meiosis, a_{ij}, separating every pair of individuals i and j in the simulated pedigree. From our predicted pedigrees, we again extract a minimum meiosis distance $p_{i,j}$. Now can compute L_1 distances between the actual and predicted meiosis distances. These quantities are $\sum_{i<j:g_{i,j}\neq\infty} |a_{i,j} - g_{i,j}|$, and $\sum_{i<j:p_{i,j}\neq\infty} |a_{i,j} - p_{i,j}|$. This is the number of meioses, or edges in the pedigree graph, which are wrong on paths connecting all pairs of extant individuals. This is plotted in the left panels of Figures 6 and 7. Now, for a pair of extant individuals, there is always some relationship in the simulated pedigree, since it is a connected graph. But it is possible that one of the inference algorithms did not predict a relationship. Specifically this quantity is $\sum_{i<j:g_{i,j}=\infty} 1$, and $\sum_{i<j:p_{i,j}=\infty} 1$, and it is plotted in the right panel of both figures.

Figure 6 was done with the simulation method described above. However, in Figure 7, to obtain pedigrees with even more outbreeding, a large population size was simulated and a connected sub-pedigree with the desired number of extant individuals was extracted from the large simulation. Notice that with more inbred pedigrees, under this measure of accuracy, the CIP algorithm performs superior to both the COP and the GBIRP methods. The accuracy of COP and CIP increase on the inbred data as compared to the outbred data, perhaps because inbreeding increases the apparent IBD making relationships easier to detect.

Relationships in the HapMap and Wellcome Trust Data. A recent paper by Pemberton et al. [28] reported many familial relationships among MKK individuals in HapMap and few relationships among the CEU and YRI individuals. The method they used did not reconstruct pedigrees, but estimated pair-wise relationships. As a follow-up to their study, we ran our method on the parents of the CEU and YRI trios (for which Pemberton et al. found no relationships) and on the unrelated MKK individuals (for which Pemberton, et al. found 9 first degree relationships). Our results

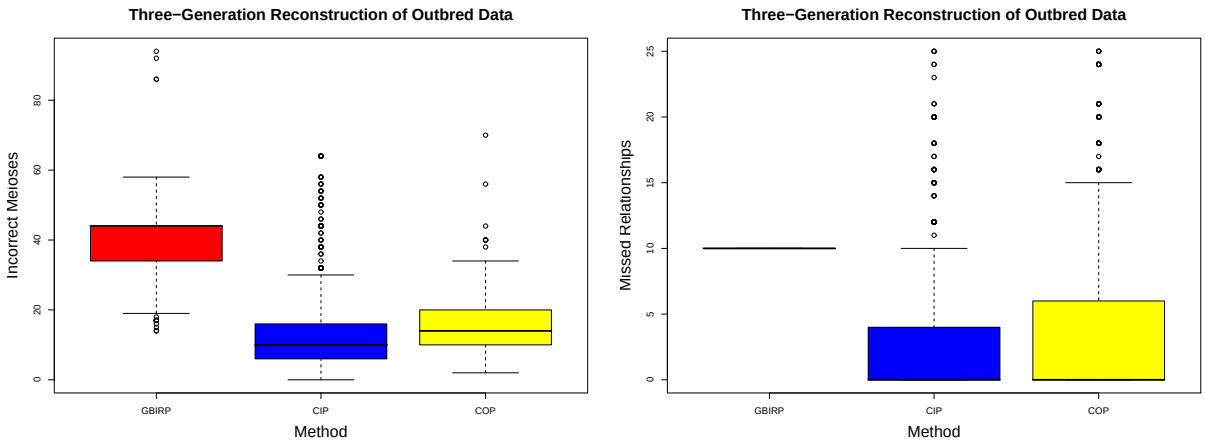

Fig. 7. Comparison with GBIRP on Outbred Simulations. The three-generation pedigrees here were simulated with $n = 10$ extant individuals, since GBIRP could not process larger pedigrees. Here, the simulated pedigree relating the extant individuals was outbred. The accuracy of 1000 simulated pedigrees were computed and plotted. All methods perform better than they did on the inbred data set. Over all, the COP method performs best on the outbred data.

contradicted theirs in that we found no evidence of first degree relationships among the MKK individuals and evidence of 2nd and 4rd cousin relationships in the YRI and CEU, respectively. We also ran our method on the Wellcome Trust individuals having at least 85% identity by state (IBS) and found that some individuals look like 2nd cousins.

Taking the data from the individuals of interest, between every pair of people we inferred IBD states (0,1, or 2 alleles shared IBD) along the genome and gave those predictions as input to our reconstruction method. To infer IBD, we used a method applied to consecutive, non-overlapping 1Mb windows of the genome: if the two individuals are homozygous for the same alleles across the window, then the IBD state is two shared alleles; if the two individuals have some phasing of the window such that one haplotype can be shared in the region, then the IBD state is one shared allele. Note, that since our reconstruction method takes the IBD predictions as input, a more sophisticated method may be used, such as the HMM used by Plink [30] or the hashing method used by GERMLINE [15]. However, we believe that this simple method is sufficient, because it is unlikely for a pair of non-IBD individuals to share a haplotype for a whole 1Mb region.

Our reconstruction method infers the average length of shared regions between every pair of individuals from the input IBD states. For fixed IBD states, there are multiple sets of shared segments that can explain the IBD states. However, if we assume that segments can only begin and end at transitions from one IBD state to another, then the number of shared segments is fixed. Since the sum of the lengths of the shared segments is also fixed, the expected length of the shared segments is the same regardless of the particular explanation chosen. The variance is not the same, but the edge test only depends on the expectation. Therefore, the estimation of average length of shared regions from the IBD states is straightforward.

For the MKK, CEU, and YRI HapMap individuals, we ran our COP reconstruction method. For the MKK unrelated individuals, we found some individuals related who are 3rd cousins and related by a 5th generation ancestor. For the CEU individuals, we found some individuals related by a 6th

generation ancestor, meaning they are 4th cousins. For the YRI individuals, we found 2nd cousins. These results are not consistent with the results found by Pemberton et al [28]. This can be explained by possible errors in the inferences made by the method of Pemberton et al., by our method, or both. We found that some of the first-degree relatives predicted by Pemberton et al. in the MKK individuals did not pass close inspection of the data. For example, true parent-child pairs must share a whole chromosome by Mendelian inheritance, since the child inherits a chromosome from the parent. This sharing happens regardless of the transmitted chromosome being recombinant. Several parent-child pairs predicted by Pemberton et al. had many 1Mb regions in disagreement, and had 30 disagreeing SNPs out of 500 SNPs in a typical window. Furthermore there is a set of three individuals, two pairs of which they predicted to be full siblings, yet the third pair of individuals was not predicted to be siblings. Since full sibling relationships must be transitive, there is clearly an error in their prediction.

Taking the individuals from the Wellcome Trust data that have at least 85% identity-by-state (IBS) with some other individual, we ran our IBD inference method on the genotypes and ran the COP reconstruction method on the IBD inferences. We found some 2nd cousins, meaning individuals related via some 4th generation ancestor.

For all these results, it should be noted that every relationship prediction method has difficulty making reliable predictions. Our method is heavily dependent on accurate IBD predictions and can be misled by genotyping errors. Such errors lead our method to under-predict rather than over-predict relationships, since our simple determination of IBD is disrupted by a single dis-agreeing SNP. Indeed, it is important not to phase the genotypes before predicting IBD, since the phasing process can lead to incorrectly imputed missing genotypes and disrupted IBD estimates. It is quite possible that all relationship prediction methods are very sensitive to genotyping errors. Due to these difficulties, we believe that these aspects of relationship prediction should continue to be investigated.

6 Discussion

The reconstruction of pedigrees from haplotype data is undoubtedly a natural question of interest to the scientific community. Reconstructing very small families, or first generation relationships is a relatively easy task, but reconstructing a full inbred pedigree involving a few generations is inherently difficult since the traces left in our genomes by an ancestor drops exponentially with the distance to the ancestor. Here, we proposed a reconstruction method for pedigrees given haplotype data from the most recent generation. We use a generation-by-generation pedigree reconstruction approach that takes haplotype data as input and finds the pedigree(s) that relate the individuals. Notably, our methods are the first to reconstruct multi-generational pedigrees, rather than a set of pair-wise relationships which may not be consistent with each other.

We present two methods of inferring the pedigrees that relate the input haplotypes. Both our methods proceed from the bottom of the pedigree towards the top. The main difference between our methods is that in CIP we assume an inbreeding model, and in COP we assume an outbreeding model. We show that our methods perform considerably better than the state of the art.

One of the basic questions that we ask is how many generations back would it be possible to reconstruct a pedigree. By simulations, we show that one can reconstruct at least fifth cousins with

some accuracy. Furthermore, we obtain a lower bound showing that given two individuals with the most-recent-common ancestor being five generations back there is a constant probability for the two not to share any genomic region inherited from the common ancestor. This bound obviously does not apply to inbred pedigrees or to multi-way relationships (i.e. rather than pair-wise relationships, consider relationships on a set of individuals). One of the open problems naturally arising from this is whether our lower bound can be extended to the case of inbreeding and to multi-way relationships. More generally, a major challenge would be to understand what are the limitations of pedigree reconstruction and under which conditions.

We note that our methods and analysis are limited to a restricted scenario in which there is monogamy and the generations are synchronous. If monogamy is broken then our approach will not work since the sibling relationships in the compatibility graph at each level will not be a simple partition. It is plausible that a different graph formulation may still provide an accurate solution to more complex pedigrees, however the exact formulation that will resolve such pedigrees is currently unknown and is left as an open challenge.

There are significant open challenges with pedigree reconstruction. For example, it would be nice to obtain confidence values on the inferred pedigree edges. However this seems very difficult, even if we can draw pedigrees from the posterior distribution of pedigree structures given the data. Since edges in a pedigree are not labeled, obtaining confidence values for a pedigree P would translate to: drawing pedigree samples, Q, from the distribution, identifying the edges in P and Q that provide the same relationships, and scoring the edges of P according to the probability of pedigree Q. As discussed in Kirkpatrick et al. [19], the second step, identifying the edges in P and Q that provide the same relationships, is a hard problem.

Acknowledgments

We thank Eleazar Eskin for helpful conversations. B.K. was supported by the NSF Graduate Research Fellowship. E.H. is a faculty fellow of the Edmond J. Safra Bioinformatics program at Tel-Aviv University. E.H. was supported by the Israel Science Foundation grant no. 04514831. R.M.K. was supported by NSF grant no. CCF-1052553.

References

1. GR Abecasis, SS Cherny, WO Cookson, and LR Cardon. Merlin-rapid analysis of dense genetic maps using sparse gene flow trees. *Nature Genetics*, 30:97–101, 2002.
2. T. Y. Berger-Wolf, S. I. Sheikh, B. DasGupta, M. V. Ashley, I. C. Caballero, W. Chaovalitwongse, and S. L. Putrevu. Reconstructing sibling relationships in wild populations. *Bioinformatics*, 23(13):i49–56, 2007.
3. H. Bickeboller and E. A. Thompson. Distribution of genome shared ibd by half-sibs: Approximation by the poisson clumping heuristic. *Theoretical Population Biology*, 50(1):66 – 90, 1996.
4. M. Boehnke and N. J. Cox. Accurate inference of relationships in sib-pair linkage studies. *American Journal of Human Genetics*, 61:423–429, 1997.
5. C. Bourgain, S. Hoffjan, R. Nicolae, D. Newman, L. Steiner, K. Walker, R. Reynolds, C. Ober, and M. S. McPeek. Novel case-control test in a founder population identifies p-selectin as an atopy-susceptibility locus. *American Journal of Human Genetics*, 73(3):612–626, 2003.
6. D. Brown and T. Berger-Wolf. Discovering kinship through small subsets. *WABI 2010: Proceedings for the 10th Workshop on Algorithms in Bioinformatics*, 2010.

7. S. R. Browning, J. D. Briley, L. P. Briley, G. Chandra, J. H. Charnecki, M. G. Ehm, K. A. Johansson, B. J. Jones, A. J. Karter, D. P. Yarnall, and M. J. Wagner. Case-control single-marker and haplotypic association analysis of pedigree data. *Genetic Epidemiology*, 28(2):110–122, 2005.
8. G. Coop, X. Wen, C. Ober, J. K. Pritchard, and M. Przeworski. High-Resolution Mapping of Crossovers Reveals Extensive Variation in Fine-Scale Recombination Patterns Among Humans. *Science*, 319(5868):1395–1398, 2008.
9. D. Doan and P. Evans. Fixed-parameter algorithm for haplotype inferences on general pedigrees with small number of sites. *WABI 2010: Proceedings for the 10th Workshop on Algorithms in Bioinformatics*, 2010.
10. K. P. Donnelly. The probability that related individuals share some section of genome identical by descent. *Theoretical Population Biology*, 23(1):34 – 63, 1983.
11. R.C. Elston and J. Stewart. A general model for the analysis of pedigree data. *Human Heredity*, 21:523–542, 1971.
12. M. Fishelson, N. Dovgolevsky, and D. Geiger. Maximum likelihood haplotyping for general pedigrees. *Human Heredity*, 59:41–60, 2005.
13. I Gallego Romero and C Ober. CFTR mutations and reproductive outcomes in a population isolate. *Human Genet*, 122:583–588, 2008.
14. D. Geiger, C. Meek, and Y. Wexler. Speeding up HMM algorithms for genetic linkage analysis via chain reductions of the state space. *Bioinformatics*, 25(12):i196, 2009.
15. A. Gusev, J. K. Lowe, M. Stoffel, M. J. Daly, D. Altshuler, J. L. Breslow, J. M. Friedman, and I. Pe'er. Whole population, genomewide mapping of hidden relatedness. *Genome Research*, 19:318–26, 2009.
16. R. M. Karp and S. C. Li. An efficient method for quasi-cliques partition. *Manuscript in preparation*, 2011.
17. B. Kirkpatrick. Haplotype versus genotypes on pedigrees. *WABI 2010: Proceedings for the 10th Workshop on Algorithms in Bioinformatics*, 2010.
18. B. Kirkpatrick. Pedigree reconstruction using identity by descent. *Class project, Prof. Yun Song, 2008. Technical Report No. UCB/EECS-2010-43*, 2010.
19. B. Kirkpatrick, Y. Reshef, H. Finucane, H. Jiang, B. Zhu, and R. M. Karp. Algorithms for comparing pedigree graphs. *CoRR*, abs/1009.0909, 2010.
20. 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.
21. S. L. Lauritzen and N. A. Sheehan. Graphical models for genetic analysis. *Statistical Science*, 18(4):489–514, 2003.
22. J. Li and T. Jiang. An exact solution for finding minimum recombinant haplotype configurations on pedigrees with missing data by integer linear programming. In *Proceedings of the 7th Annual International Conference on Research in Computational Molecular Biology*, pages 101–110, 2003.
23. X Li, X-L Yin, and J Li. Efficient identification of identical-by-descent status in pedigrees with many untyped individuals. *Bioinformatics*, 26(12):i191–i198, 2010.
24. M S McPeek and T P Speed. Modeling interference in genetic recombination. *Genetics*, 139(2):1031–44, 1995.
25. M.S. McPeek and L. Sun. Statistical tests for detection of misspecified relationships by use of genome-screen data. *Amer. J. Human Genetics*, 66:1076 – 1094, 2000.
26. Ng MY, Levinson DF, and et al. Meta-analysis of 32 genome-wide linkage studies of schizophrenia. *Mol Psychiatry*, 14:774–85, 2009.
27. S. B. Ng, K. J. Buckingham, C. Lee, A. W. Bigham, H. K. Tabor, K. M. Dent, C. D. Huff, P. T. Shannon, E. W. Jabs, D. A. Nickerson, J. Shendure, and M. J. Bamshad. Exome sequencing identifies the cause of a mendelian disorder. *Nature genetics*, 42(1):30–35, January 2010.
28. T. J. Pemberton, C. Wang, J.Z. Li, and N.A. Rosenberg. Inference of unexpected genetic relatedness among individuals in hapmap phase iii. *Am J Hum Genet*, 87(4):457–64, 2010.
29. A. Piccolboni and D. Gusfield. On the complexity of fundamental computational problems in pedigree analysis. *Journal of Computational Biology*, 10(5):763–773, 2003.
30. S. Purcell, B. Neale, K. Toddbrown, L. Thomas, M. Ferreira, D. Bender, J. Maller, P. Sklar, P. Debakker, and M. Daly. PLINK: A Tool Set for Whole-Genome Association and Population-Based Linkage Analyses. *The American Journal of Human Genetics*, 81(3):559–575, September 2007.
31. S. I. Sheikh, T.Y. Berger-wolf, A. A. Khokhar, I. C. Caballero, M. V. Ashley, W. Chaovalitwongse, C. Chou, and B. Dasgupta. Combinatorial reconstruction of half-sibling groups from microsatellite data. *8th International Conference on Computational Systems Bioinformatics (CSB)*, 2009.
32. E. Sobel and K. Lange. Descent graphs in pedigree analysis: Applications to haplotyping, location scores, and marker-sharing statistics. *American Journal of Human Genetics*, 58(6):1323–1337, 1996.
33. J. Stankovich, M. Bahlo, J.P. Rubio, C.R. Wilkinson, R. Thomson, A. Banks, M. Ring, S.J. Foote, and T.P. Speed. Identifying nineteenth century genealogical links from genotypes. *Human Genetics*, 117(2–3):188–199, 2005.

34. L. Sun, K. Wilder, and M.S. McPeek. Enhanced pedigree error detection. *Hum. Hered.*, 54(2):99–110, 2002.
35. B. D. Thatte. Combinatorics of pedigrees, 2006.
36. B. D. Thatte and M. Steel. Reconstructing pedigrees: A stochastic perspective. *Journal of Theoretical Biology*, 251(3):440 – 449, 2008.
37. E. A. Thompson. *Pedigree Analysis in Human Genetics*. Johns Hopkins University Press, Baltimore, 1985.
38. T. Thornton and M.S. McPeek. Case-control association testing with related individuals: A more powerful quasi-likelihood score test. *American Journal of Human Genetics*, 81:321–337, 2007.