Monte Carlo estimation of identity by descent in populations

Christopher Glazner

A dissertation
submitted in partial fulfillment of the
requirements for the degree of

Doctor of Philosophy

University of Washington

2014

Reading Committee:

Elizabeth Thompson, Chair

Volodymyr Minin

Sharon Browning

Program Authorized to Offer Degree:
Department of Statistics

University of Washington

Abstract

Monte Carlo estimation of identity by descent in populations

Christopher Glazner

Chair of the Supervisory Committee:
Professor Elizabeth Thompson
Statistics

Genetic similarity between organisms arises from segments of shared genome, which are said to be identical by descent (IBD). Modeling IBD in pedigrees forms the basis of classical linkage analysis and has been a fruitful method for inferring trait locations. We examine methods for modeling IBD in more general settings where relationships among subjects are not known completely. A natural approach is to use a hidden Markov model (HMM) based on a transition model for IBD along the chromosome, but the number of possible IBD states for more than a few individuals makes makes standard HMM calculations infeasible. We describe two broad approaches to sampling from this model. First, we decompose the group IBD model into a series of pairwise approximations which can be sampled efficiently. This decomposition permits other modifications to the model so that it can be used with unphased genotypes or incomplete pedigree information. Second, we implement a particle Gibbs sampling algorithm for the HMM, which is computationally intensive but targets the correct model. Both methods are compared against exact HMM sampling. The particle Gibbs method more accurately captures the true model distribution at the expense of increased computation time.

TABLE OF CONTENTS

LIST OF FIGURES

ACKNOWLEDGMENTS

This dissertation was made possible by my advisor, Elizabeth Thompson, who has supported me financially since my second quarter at UW and provided invaluable guidance and mentoring. I would also like to thank my committee members, Vladimir Minin, Sharon Browning, and Ellen Wijsman, for their support throughout this process. In particular, Vladimir made possible my stay at the UCLA Institute for Pure and Applied Math, which was a pivotal time in my graduate career.

I have benefitted greatly from interactions with the changing membership of the Thompson group, including Steven Lewis, Fiona Grimson, Serge Sverdlov, Charles Cheung, Marshall Brown, Lucas Koepke, Jesse Raffa, John Ranola, and others who joined us over the years. Additional thanks are due to the members of UW's statistical genetics and population genetics seminars. I have been taught by many excellent teachers, both in the university and the community.

Thank you to Jane Lange for being my thesis buddy, and to Norah Andrews, Brian Ma, and Gaby Cardos for friendship and support. Similar sentiments go to my family, especially my parents and grandmother.

I have been very fortunate to receive funding from the NIH during my studies.

Thank you to Matthew Cleveland of Genus PIC for providing the pig genotype data used in this work.

DEDICATION

To my parents, Drs. Allen Glazner and Mary Olney.

Chapter 1

INTRODUCTION

1.1 Identity by descent

Segments of genome inherited from a common ancestor by multiple individuals are said to be identical by descent (IBD). We say that two individuals share a gene copy IBD if they inherited the gene from a common ancestor who lived more recently than a specific time point. This time point is understood to be recent relative to the mutation process, so that IBD segments have nearly the same DNA sequence. By modeling this sequence similarity, we can estimate IBD segments in families or in groups of individuals not known to be related.

IBD estimation is the basis for classical linkage analysis. These methods use widely spaced marker data and a highly informative model for Mendelian segregation to infer IBD in pedigrees (Abecasis et al., 2001; Thompson, 2011). The IBD is then used to calculate LOD scores, which give the likelihood ratio for the hypotheses that a locus is linked to a trait of interest (Morton, 1955). Family-based models stand in contrast to genome-wide association studies (GWAS) (Burton et al., 2007), which eschew modeling of family inheritance in favor of independent statistical analysis of separate markers. Family studies are useful in some situations where association studies fall short; for example, a rare allele may give a strong linkage signal in a family but have insufficient power to be detected in a population.

Modern genetic datasets contain sufficient information to permit inference of IBD under less structured models than Mendelian segregation (Browning and Browning, 2010; Gusev et al., 2009; Kong et al., 2008; Brown et al., 2012). In particular, we show in this work that IBD can be estimated with high resolution without knowledge

of the true pedigree.

In the remainder of this chapter, we describe the concept of IBD and a model-based approach to inferring it between pairs of individuals, introduced by Thompson (2008). In Chapter 2, we introduce an extension of this model to larger groups of individuals by building a joint IBD state using conditional sampling of pairwise relationships. Chapter 3 explains how to express haplotype phase in the model, so that unphased or partially phased data can be used as input. In Chapter 4, we discuss an approach to incorporating IBD estimated in subpedigrees of the sample into the IBD state for the entire sample. Chapter 5 uses particle Gibbs sampling to fit the IBD model simultaneously for an entire group of individuals.

1.2 IBD in families and populations

We use the term IBD state to refer to a partition on a set of gametes at a particular locus. Two gametes are in the same block of the partition if they descend from the same ancestral gamete; thus, the partition is determined by the choice of a set of founder gametes. Two gametes in the same block are said to be shared IBD, or simply IBD. In an observed pedigree, the pedigree founders form the relevant set of ancestors. Two gametes are considered IBD if their descent can be traced to a common ancestor within the pedigree. In a population, where no relationships among individuals are known, we take the founder set to be the gametes of all individuals alive at a fixed point in time. This definition of IBD can be expressed in terms of the pedigree one if we imagine an extended pedigree formed by all individuals alive after the chosen time point. In either setting, IBD is defined relative to the ancestors of the sampled individuals.

Kingman (1982) introduced the coalescent model, which described the history of a set of gametes as a stochastic process on equivalence relations. In diploid organisms, the IBD state varies over the length of the genome because of recombination. The coalescent description of descent at a single locus was extended to model recombina-

tion along the chromosome in the ancestral recombination graph (ARG) of Hudson (1990) and Griffiths and Marjoram (1996). The ARG model gives a complete description of the history of the gametes but is computationally difficult to fit (Kuhner and Smith, 2007). The model we present approximates a slice of the ARG at a fixed but unknown time coordinate, in the same manner that fitting a standard coalescent model (Drummond et al., 2012) slices the ARG at unlinked loci along the genome.

Unfortunately for our purposes, the coalescent model with recombination is not Markov along the chromosome (McVean and Cardin, 2005), and thus neither is the induced IBD process at a fixed time point. As an alternative, we work with a class of models which have the Markov property but approximate the marginal and transition distributions of a coalescent-generated ARG.

Sobel and Lange (1996) introduced a way to express IBD states in graphical form, by letting nodes denote founder genome labels (FGLs) assigned to founder gametes, and placing labeled edges between the nodes connecting an individual's two FGLs. Extending this graphical representation along the chromosome, we will use the term IBD graph to mean a series of IBD states indexed by genome location.

Suppose at a particular locus individuals A and B share their maternal gametes IBD; we denote this IBD state by $\{\{A_m, B_m\}, \{A_p\}, \{B_p\}\}$.

1.3 Model-based IBD detection in pairs of individuals

In a randomly mating population, the probability that two gametes are IBD at a point declines exponentially with the number of meioses since their common ancestor; however, given that they are IBD, the expected length of the IBD segment declines as the inverse of the number of meioses (Donnelly, 1983). This means that given some recent IBD between two individuals, the IBD segment is likely to be long enough to detect using statistical methods. High-resolution detection of IBD segments has become possible with the availability of sufficiently dense marker data (Browning and Browning, 2010; Kong et al., 2008; Gusev et al., 2009). A number of applications

for the detection of IBD segments have been proposed, including disease mapping (Browning and Browning, 2010), haplotype phasing (Kong et al., 2008), and detection of copy number variation (Gusev et al., 2009).

Many methods of detecting IBD, including the ones presented here, are based on a hidden Markov model (HMM). An HMM consists of a Markov chain on a hidden space and a series of data points assumed to be conditionally independent given the hidden states. This structure permits efficient computation (Rabiner, 1989) and flexible model specification. The essential fact of an HMM is that adjacent data points are similar in a manner described by the hidden process. In a pedigree, adjacent marker loci are expected to be similar in a sense rigidly specified by the transition model of the recombination process. Without a pedigree, there are no explicit constraints on the possible changes in IBD state between two chromosomes, but the transition model expresses the fact that adjacent states are likely to be similar.

A model for autozygosity (IBD between the two gametes of a single individual) was proposed by Leutenegger et al. (2003). The transition process in this HMM was determined by two parameters: the marginal probability of autozygosity and the rate of switching between the autozygous and non-autozygous states. The application in that work was maximum likelihood estimation of the former parameter.

Thompson (2008) presented an expansion of the autozygosity model from two to four gametes. This larger model was again specified by two parameters: the population kinship level (which reduces to the probability of autozygosity in the case of a single individual) and a change rate parameter. The model is specified as a continuous time Markov chain on the set of fifteen identity coefficients between two individuals given by Harris (1964). The Ewens sampling formula (ESF) (Ewens, 1972), a distribution on set partitions with a single parameter θ, gives the stationary measure of the transition process.[1] Thompson (2013) discusses differences between

[1]More precisely, the stationary distribution agrees with Ewens' formula after collapsing states which differ only in haplotype ordering.

the ESF and the distribution on partitions induced by the coalescent at a fixed time point.

Two IBD states have non-zero infinitesimal transition rates if one can be transformed to the other by changing the ancestry of a single gamete. The rate matrix is parametrized by the population kinship, β, or probability of two gametes being IBD at a point. Population kinship is related to θ in the ESF as follows: $\beta = \frac{1}{1+\theta}$.

Brown et al. (2012) introduced an HMM based on a slightly modified version of the transition model suggested by Thompson (2008), implemented as the *ibd_haplo* program. The two models share a stationary distribution and their parameters are interpreted in the same way, but in the newer model a slightly different set of transitions among IBD states is permitted. Crucially, the newer model can be specified on an arbitrary number of gametes. The new model is based on the Chinese Restaurant Process (CRP) (Aldous, 1985), a one-parameter stochastic process on set partitions which induces the ESF distribution for a fixed number of individuals.

The CRP with parameter θ is specified in terms of the transition process from a partition on n items to one on $n+1$ items. Given n people sitting at k "tables" in the restaurant, a new person sits down at a new table with probability $\frac{\theta}{n+\theta}$. She has probability $\frac{1}{n+\theta}$ of sitting to the right of any of the existing individuals, so her probability of sitting at an existing table is proportional to the number of individuals already sitting there.

This process can be modified to a Markov process on partitions of n items via a straightforward modification. (We now drop the restaurant metaphor to avoid confusion between individuals and gametes). At a constant rate (for a given value of n), the following event occurs: a new item is inserted into the partition with group label chosen according to the CRP, and a uniformly chosen item is deleted from the new partition. If the new item is not deleted, it assumes the identity of the deleted item. Some of these events do not change the IBD state; the ones which do determine the transition rates of the process on IBD states. Intuitively, an event is like a Gibbs

sampling step on the partition of $n+1$ items holding constant the partition on n items; the stationary distribution of the process is thus the ESF. More detailed discussion appears in Brown et al. (2012).

In the HMM of Brown et al. (2012), observed alleles at a locus are assumed independent given the underlying IBD state: alleles shared IBD are identical, and non-IBD alleles are independent draws from the population. Population allele frequencies must be provided as input. Genotyping errors are modeled as a small probability that the reported allele differs from the true allele. Differences due to mutation between IBD alleles are indistinguishable from errors. All markers are diallelic single nucleotide polymorphisms (SNPs).

Model inference is performed using the standard forward-backward algorithm for HMMs (Rabiner, 1989). Specification of an HMM requires choices for the emission probability $e(x|i)$ of observing x given underlying state, the transition distribution probability $d_t(j|i)$ of being in state j at time t after being in state i at time $t-1$, and an initial distribution. A discretization of the hidden process is used to avoid computing matrix exponentials when calculating transition probabilities; a tuning parameter mixes a small fraction of the stationary distribution with the discrete jump chain, so that all transitions have nonzero probability (see the parameter `transition matrix null fraction` in Appendix B.1). The markers are tightly spaced relative to the rate of state transitions, so the discrete approximation is reasonable. The initial distribution used in the forward-backward algorithm is the stationary distribution of the transition process. A scaling parameter for distance along the chromosome scales the dwelling time in each IBD state. The output of the algorithm is the marginal distribution over identity states at each locus, conditional on the marker data.

The *ibd_haplo* program was applied to simulated datasets to examine performance under various combinations of parameters and input. In particular, the model was tested with different levels of linkage disequilibrium (LD) in the sample haplotypes. LD is the statistical correlation of allelic markers at different loci, and its presence

violates the conditional independence assumptions made by the HMM. The HMM assumes that the alleles observed at different loci are independent given the underlying IBD states, but LD creates dependence between alleles beyond that created by the IBD. In other words, haplotypes from unrelated individuals will be more similar than expected under independence, creating false evidence of IBD. The study showed that increased the average level of simulated LD decreased the model's sensitivity and specificity in detecting IBD.

1.4 Earlier work estimating joint IBD states

The HMM presented in Brown et al. (2012) can be naturally extended to an arbitrary number of individuals: the transition process and data likelihood constructed for any n. However, the forward-backward calculations are intractable for the model on more than three or four diploid individuals. This algorithm has time complexity quadratic in the number of hidden states (Rabiner, 1989), which in our case grows dramatically with the number of gametes: there are more than 10^5 IBD states for five individuals, and more than 10^{13} for ten. (Details on the asymptotic growth rate are given by Berend and Tassa (2010)).

It is possible to estimate IBD probabilities between all pairs of individuals, incurring a cost that grows only quadratically in the group size, but in many situations a model for the entire group is necessary. For example, if IBD estimates are used to calculate covariances among individuals, pairwise IBD may not produce a valid covariance matrix. To see an exaggerated example of this, suppose our pairwise estimates indicate that one gamete is IBD to all other gametes in the sample, but no other pairwise inferences suggest any IBD; the relatedness matrix constructed from IBD indicators is not positive definite and consequently does not specify a valid multivariate normal likelihood.

Glazner and Thompson (2012) presented a first attempt at estimating group IBD states using *ibd_haplo*, in the context of individuals in separate pedigrees with rela-

tionships between the pedigrees not known. The method used two sources of input data: IBD graphs estimated in the pedigrees using MORGAN, and pairwise marginal probabilities of IBD estimated among all pairs of individuals using *ibd_haplo*. At each locus, the pedigree IBD graphs were combined to create an initial joint IBD state, and the pairwise states were arranged in descending order of probability. Then, each pairwise state was added to the joint state if it did not conflict with the existing joint state.

The aim of this approach was to include as much pairwise information as possible in constructing the joint state, without creating an invalid joint state. The method was shown to be capable of accurately reconstructing LOD scores in a large pedigree using only information about relationships in small subpedigrees. Among the method's shortcomings were its inability to express uncertainty in the pairwise inferences and the lack of smoothing across marker loci.

We now present a way to apply the basic machinery of the *ibd_haplo* model in an algorithm for sampling group IBD states.

Chapter 2

INFERRING JOINT IBD

2.1 Joint and pairwise IBD

Here we present a further development of the method in Glazner and Thompson (2012). It follows the same approach of combining pairwise HMM inferences into a joint IBD state for the group. The joint state is built up incrementally from pairwise passes of the HMM. An important difference from the older method is that during the backwards pass of the HMM we generate a random trajectory rather than calculating marginal probabilities. Both uses of the HMM smooth information across loci, but using realized paths of IBD states, we can also transfer information across different pairs of individuals. By properly conditioning the HMM on previously generated trajectories, we can produce consistent joint IBD states.

The hidden states of the HMM take values in $\mathcal{P}_4$, the space of partitions of a set with 4 elements. The IBD states for a group of n individuals are values in $\mathcal{P}_{2n}$. At each locus, our goal is to build an element of $\mathcal{P}_{2n}$ incrementally out of elements of $\mathcal{P}_4$. This is possible in the first place because there exists an injective mapping of $\mathcal{P}_{2n}$ into the set of configurations of $\binom{2n}{2}$ elements of $\mathcal{P}_4$, one for each pair of individuals, denoted $\mathcal{P}_4^{2n}$. The target of our inference is the subset of $\mathcal{P}_4^{2n}$ corresponding to elements of $\mathcal{P}_{2n}$, which we will call valid configurations. Similarly, a collection of elements of $\mathcal{P}_4$ covering subsets of k individuals is valid if it corresponds to an element of $\mathcal{P}_{2k}$.

It is straightforward to check if an element p of $\mathcal{P}_4^{2n}$ is a valid configuration. Any two of the pairwise states composing p with an individual in common must agree on the IBD status of the two chromosomes of the common individual. Assuming all the pairs do not conflict, we only need to check triples of individuals to ensure that

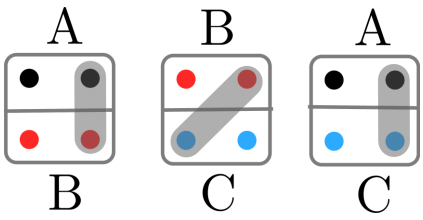


Figure 2.1: An example of an invalid configuration in $\mathcal{P}_4^6$, the space of pairwise IBD states covering a trio of individuals. The grey shading represents IBD gametes. A's right gamete and C's left gamete are both shared IBD with B's right gamete, but not to one another.

the configuration is valid for all individuals. A partition of a set of chromosomes is isomorphic to an equivalence relation on the same set. Consider the relation R formed by taking the union of all the pairwise states in p, treated as equivalence relations. If R is an equivalence relation, then p is valid and corresponds to an element of $\mathcal{P}_{2n}$. The condition is satisfied if R is transitive, meaning that for any three chromosomes a, b, and c, aRb and bRc implies aRc (reflexivity and symmetry hold trivially). In other words, if p is not valid, there is some nontransitive triple $\{a, b, c\}$. Any such triple consists of chromosomes from at most three individuals, so if the pairwise states in p for all triples of individuals do not create a nontransitive relation, p is a valid configuration. Figure 2.1 shows an example of an invalid configuration.

This procedure illustrates the useful fact that we can determine valid states by considering at most three individuals at a time. This property is used to build a joint state one pair of individuals at a time; the relevant constraints can be quickly calculated for each pair. We can then eliminate from the hidden state space any IBD states which would create an invalid joint configuration, and perform HMM calculations on the reduced state space.

2.2 IBD states among pairs and trios

To illustrate the method in a simple context, we examine the dependence among the pairwise IBD states at locus t among the trio of individuals A, B, and C. Separately, each of the states $p_4(AB)$, $p_4(BC)$, and $p_4(CA)$ can take any value from the fifteen states in $\mathcal{P}_4$. However, there are only $|\mathcal{P}_6| = 203$ IBD states the trio can be in, so the set of consistent pairwise states is a small subset of the $15^3 = 3375$ elements in $\mathcal{P}_4 \times \mathcal{P}_4 \times \mathcal{P}_4$. Fixing $p_4(AB)$, the value of $p_4(BC)$ can be any state in which the IBD status of B's two chromosomes agrees with $p_4(AB)$. Once $p_4(AB)$ and $p_4(BC)$ are fixed, $p_4(CA)$ can only be one of a small number of states: at most seven, usually three or less, and sometimes only one.

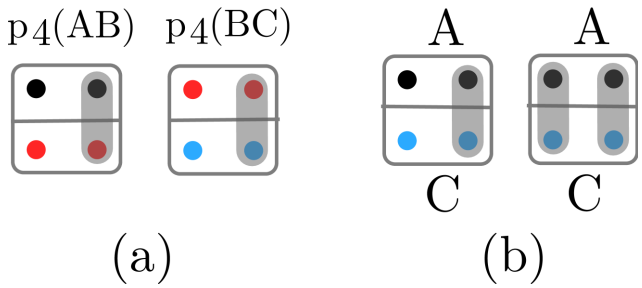


Figure 2.2: (a) Pairwise IBD states between A and B, and B and C. (b) Consequent allowed states between A and C.

An example is illustrated in Figure 2.2; suppose individuals A and B share only maternal chromosomes IBD ($p_4(AB) = \{\{A_m, B_m\}, \{A_p\}, \{B_p\}\}$), as do individuals B and C. There are only two possiblilities for the IBD state between individuals C and A: they must share maternal chromosomes IBD, and they may or may not share paternal chromosomes IBD.

The vector $\mathbf{p}_4(ij)$ of IBD states between two individuals takes values in $\mathcal{P}_4$. Suppose that over several consecutive loci $\mathbf{p}_4(AB)$ and $\mathbf{p}_4(BC)$ are constant with the values as in the example above. The transition process for $\mathbf{p}_4(CA)$ between the two permitted states is exactly that of the full process on $\mathcal{P}_2$ restricted to the two states.

The transition matrix for $\mathbf{p}_4(CA)$ is the full matrix after removing the rows and columns of all but the permitted states. We can do HMM calculations and sample $\mathbf{p}_4(CA)$ over these loci, conditional not only on the marker data of A and C but also on $\mathbf{p}_4(AB)$ and $\mathbf{p}_4(BC)$. This is possible for any fixed values of $\mathbf{p}_4(AB)$ and $\mathbf{p}_4(BC)$, although the permitted state space will change accordingly.

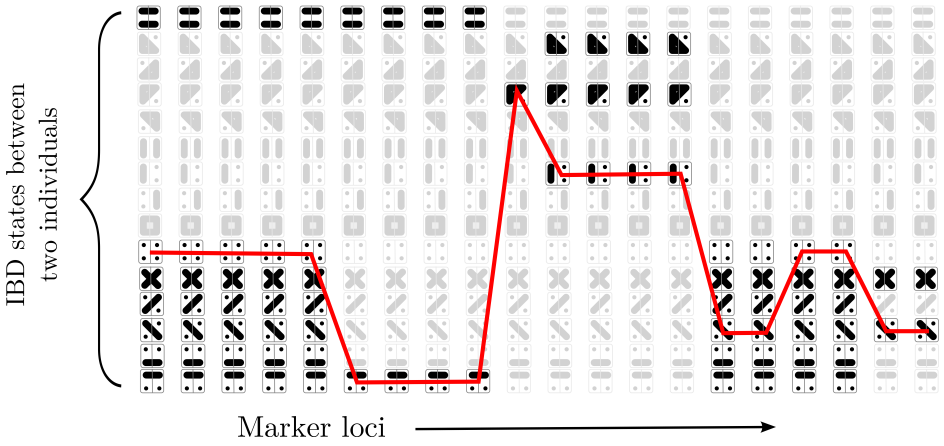


Figure 2.3: The HMM state space is restricted to the IBD states which are compatible with existing IBD states between other pairs. The red line indicates a possible trajectory through states along the chromosome.

This conditional model for $\mathbf{p}_4(CA)$ must allow for changes in $\mathbf{p}_4(AB)$ and $\mathbf{p}_4(BC)$ along the chromosome. Suppose C stops sharing maternally with B between loci t and $t+1$. The set of permitted states for $\mathbf{p}_4(CA)$ at locus $t+1$ changes to the following: C can share neither chromosome IBD with A, C can share paternally with A, or C's maternal chromosome can be shared with A's paternal. The transition matrix between these two loci is now the full transition matrix restricted to transitions beginning in the two states allowed at locus t and ending in the three states allowed at locus $t+1$.

If these sets of states are disjoint, $\mathbf{p}_4(CA)$ is forced to change states. Such a change is always possible because there is non-zero probability of going from any state to any other under both the continuous and discretized transition models (as long as some fraction of the stationary distribution is mixed into the discrete jump

chain; see Section 1.3). Further, $\mathbf{p}_4(CA)$ will be able to reach at least one of the new permitted states in a single transition as long as the change in $\mathbf{p}_4(AB)$ and $\mathbf{p}_4(BC)$ is due to a change in the ancestral origin of a single chromosome. Any single transition in the hidden model can be described (not always uniquely) as a change in the ancestral origin of a single chromosome which causes the chromosome to jump IBD groups. If the transition in one or both of $\mathbf{p}_4(AB)$ and $\mathbf{p}_4(BC)$ is due to a change in one of B's chromosomes, the IBD state between C and A is unchanged and no transition is forced. If it is due to a change in A or C, then it can be reached in a single transition in the state between A and C. While this fact is not necessary for specifying a valid conditional model, it does ensure that single transitions in $\mathbf{p}_4(AB)$ and $\mathbf{p}_4(BC)$ will not force transitions in $\mathbf{p}_4(CA)$ which violate the Chinese Retaurant Process model.

A conditional HMM can also be constructed for sampling $\mathbf{p}_4(BC)$ conditional on $\mathbf{p}_4(AB)$. In this case, the only restriction on the state space is that IBD status of B's two chromosomes in $\mathbf{p}_4(BC)$ must match $\mathbf{p}_4(AB)$. Subject to this restriction, $\mathbf{p}_4(BC)$ can be in any state, since for any values of $\mathbf{p}_4(AB)$ and $\mathbf{p}_4(BC)$ there is at least one consistent value of $\mathbf{p}_4(CA)$. We reduce the transition matrix accordingly and sample a trajectory for $\mathbf{p}_4(BC)$.

We can now specify a procedure for generating a sample of $\mathbf{p}_6(ABC)$, the vector of states in $\mathcal{P}_6$ for the trio: simulate $\mathbf{p}_4(AB)$ unconditionally, simulate $\mathbf{p}_4(BC)$ conditional on $\mathbf{p}_4(BC)$, and simulate $\mathbf{p}_4(CA)$ conditional on $\mathbf{p}_4(AB)$ and $\mathbf{p}_4(BC)$ (all conditional on the marker data of the two individuals being sampled). At each locus, the consistent configuration of pairwise states specifies a state in $\mathcal{P}_6$.

We have demonstrated above that this procedure will produce a vector of states in $\mathcal{P}_6$. We now justify its use as an approximation to sampling from the joint distribution $\Pr(\mathbf{p}_6(ABC) \mid \mathbf{x}_A, \mathbf{x}_B, \mathbf{x}_C)$, with $\mathbf{x}_i$ the vector of marker data for individual i. The

distribution decomposes as:

$$\begin{aligned}
\Pr[\mathbf{p}_6(ABC) \mid \mathbf{x}_A, \mathbf{x}_B, \mathbf{x}_C] &= \Pr[\mathbf{p}_4(AB), \mathbf{p}_4(BC), \mathbf{p}_4(CA) \mid \mathbf{x}_A, \mathbf{x}_B, \mathbf{x}_C] \\
&= \Pr[\mathbf{p}_4(CA) \mid \mathbf{p}_4(AB), \mathbf{p}_4(BC), \mathbf{x}_A, \mathbf{x}_B, \mathbf{x}_C] \times \\
&\quad \Pr[\mathbf{p}_4(AB), \mathbf{p}_4(BC) \mid \mathbf{x}_A, \mathbf{x}_B, \mathbf{x}_C] \\
&= \Pr[\mathbf{p}_4(CA) \mid \mathbf{p}_4(AB), \mathbf{p}_4(BC), \mathbf{x}_A, \mathbf{x}_B, \mathbf{x}_C] \times \\
&\quad \Pr[\mathbf{p}_4(BC) \mid \mathbf{p}_4(AB), \mathbf{x}_A, \mathbf{x}_B, \mathbf{x}_C] \times \\
&\quad \Pr[\mathbf{p}_4(AB) \mid \mathbf{x}_A, \mathbf{x}_B, \mathbf{x}_C].
\end{aligned}$$

The sampled states come from the distribution:

$$\begin{aligned}
\Pr[\mathbf{p}_6(ABC) \mid \mathbf{x}_A, \mathbf{x}_B, \mathbf{x}_C] = & Q[\mathbf{p}_4(CA) \mid \mathbf{p}_4(AB), \mathbf{p}_4(BC), \mathbf{x}_A, \mathbf{x}_C] \times \\
& Q[\mathbf{p}_4(BC) \mid \mathbf{p}_4(AB), \mathbf{x}_B, \mathbf{x}_C] \times \\
& Q[\mathbf{p}_4(AB) \mid \mathbf{x}_A, \mathbf{x}_B],
\end{aligned}$$

where Q indicates the sampling distribution of the HMM with the state space constrained according to the previously sampled IBD, described in Section 2.3. The difference between the two is that each pair of individuals is sampled conditional only on the data for those two individuals. We are ignoring the dependence of IBD states on other individuals' data, and the sampling distribution depends on the order in which the individuals are considered. We therefore randomize over orderings of pairs of individuals to obtain a procedure which is exchangeable in the input data. For each independent iteration of the sampling algorithm, we first randomly permute the individuals. Beginning with an empty group state, each individual is added to the group in the permuted order; the individuals already in the group are permuted for each new individual to obtain the order in which to sample the pairwise trajectories.

Li and Stephens (2003) used a sequential procedure based on products of ap-

proximate conditional likelihoods to calculate likelihoods in a coalescent model with recombination. In similar fashion we decompose the sampling distribution of the trajectory through $\mathcal{P}_{2n}$:

$$\begin{aligned}\Pr[\mathbf{p}_{2n}(\cdot)|\mathbf{x}] = &\Pr[\mathbf{p}_4(AB)|\mathbf{x}] \times \Pr[\mathbf{p}_4(BC)|\mathbf{p}_4(AB), \mathbf{x}]\times \\ &\Pr[\mathbf{p}_4(AC)|\mathbf{p}_4(AB), \mathbf{p}_4(BC), \mathbf{x}] \ldots,\end{aligned}$$

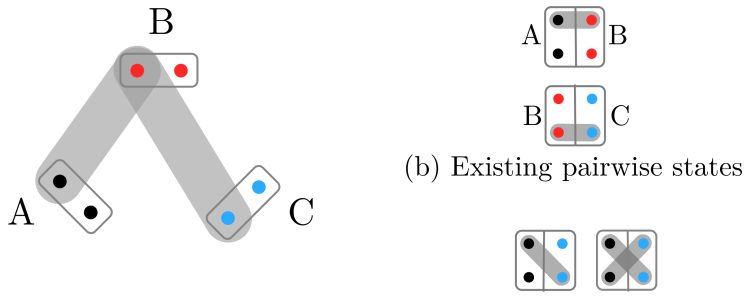
where $\mathbf{x}$ is the data of all individuals. We then replace the terms on the the right side with their approximations: letting $\mathbf{x}(ij)$ be the allele data of individuals i and j,

$$\begin{aligned}\Pr[\mathbf{p}_{2n}(\cdot)|\mathbf{y}] \approx &Q[\mathbf{p}_4(AB)|\mathbf{x}(AB)] \times Q[\mathbf{p}_4(BC)|\mathbf{p}_4(AB), \mathbf{x}(BC)]\times \\ &Q[\mathbf{p}_4(AC)|\mathbf{p}_4(AB), \mathbf{p}_4(BC), \mathbf{x}(AC)] \ldots.\end{aligned}$$

The approximation step is to sample each pairwise configuration conditionally only on the allele data for that pair. The approximation is localized, whereas in the full model all the SNP data are available at each sampling step. The sampling distribution of a pairwise trajectory for a pair of individuals given their allele data and a set of already sampled pairwise trajectories is defined as follows: the hidden trajectory distribution of the basic HMM, with the state space modified to contain only states which do not create (or force the later creation of) an invalid configuration.

2.3 Building a joint state

To build a joint state, we first sample a trajectory of IBD states for a pair of individuals using the basic HMM. Then a trajectory for one individual and a third individual is sampled, subject to the constraints imposed by the first pairwise trajectory. We sample a trajectory for the third side of the triangle, conditional on the other two. We now have three compatible pairwise states which form a joint IBD state for the three individuals.



(a) Incomplete joint IBD state

(b) Existing pairwise states

(c) State space for $p_4(AC)$

Figure 2.4: The previously sampled pairwise IBD states $p_4(AB)$ and $p_4(BC)$ restrict the possible values of $p_4(AC)$

An example is shown in Figure 2.4, which depicts a slice at a locus of sample pairwise trajectories for pairs AB and BC. On the left, the shading depicts the incomplete joint state being constructed. On the right appear the fixed states in $\mathcal{P}_4$ for pairs AB and BC. The state space for pair BC is constrained to only two possible states; any others will produce an invalid configuration.

Constraining the state space in this way is minimal in the sense that the only states ruled out are those which inevitably lead to an invalid configuration. In other words, the support of the sampling distribution matches that of the true conditional distribution; no possible valid joint configurations are eliminated.

A description of the algorithm for any number of individuals is given as Algorithm 1. The appearance of $\mathbf{p}_{2n}^{pop}$ in the conditioning of the HMM sampling distribution Q is meant to indicate that the space of possible states for the pair of individuals under consideration is reduced to only those states which do not conflict with the group IBD state in the process of being constructed.

Appendix A.1 demonstrates that the constrained state space in line 7 is never empty at any locus. There is always at least one allowed state for a pair to be in, given an existing configuration of all pairwise states for a group and some of the pairs of formed by individual k and the group.

Algorithm 1 Sample an IBD graph on n individuals from SNP data

Require: SNP haplotypes for n individuals
  Initialize $\mathbf{p}_{2n}^{pop}$ to the empty IBD graph.
  Randomly permute individuals and label them $\{A_1, \ldots, A_n\}$
  Sample from $Q[\mathbf{p}_4(A_1A_2)|\mathbf{x}(A_1A_2)]$ and add the trajectory to $\mathbf{p}_{2n}^{pop}$.
  for $2 < k \leq n$ **do**
    Randomly permute individuals $\{A_1, \ldots, A_{k-1}\}$ and label them $\{B_1, \ldots, B_{k-1}\}$.
    for $1 \leq i < k$ **do**
      Sample from $Q[\mathbf{p}_4(B_iA_k)|\mathbf{p}_{2n}^{pop}, \mathbf{x}(B_iA_k)]$ and add the trajectory to $\mathbf{p}_{2n}^{pop}$.
    end for
  end for
  return An IBD graph on n individuals, $\mathbf{p}_{2n}^{pop}$.

The approximate sampling distribution is not exchangeable in the haplotypes. Like Li and Stephens (2003), we perform the sampling procedure many times, using a random ordering of individuals at each iteration. Since many samples must be generated in any case, averaging over orderings in this fashion does not create an additional computational burden.

A software package implementing this algorithm and those presented in the next two chapters is available. The program used in the following analysis is called *ibd_stitch*; see Appendix B.1 for details on using the program.

2.4 Inference using IBD graphs

(a) Classical LOD scores

In order to make inferences about the location of genes affecting a trait, we need a way to connect IBD graphs to a likelihood model parametrized by hypothesized trait locus. The traditional mechanism for doing so is the LOD score (Morton, 1955), defined for a particular locus as the the likelihood ratio between the hypothesis of a trait driven by a gene at the locus and the hypothesis of an unlinked trait.

Suppose we observe marker data $\mathbf{x}$ and trait data $\mathbf{y}$ on a group of individuals in a

pedigree. We compare the models Γ and Γ_0, where $\Gamma(t)$ hypothesizes a trait location t on the chromosome of the markers, and Γ_0 assumes that the trait and marker data are independently distributed on the pedigree. Because it assumes this independence, model Γ_0 can be factored into a trait model Γ_T and a marker model Γ_M. We calculate the LOD scores as follows:

$$\begin{aligned}\log_{10}\frac{\Pr(\mathbf{x},\mathbf{y};\Gamma(t))}{\Pr(\mathbf{x},\mathbf{y};\Gamma_0)} &= \log_{10}\frac{\Pr(\mathbf{x},\mathbf{y};\Gamma(t))}{\Pr(\mathbf{y};\Gamma_T)\Pr(\mathbf{x};\Gamma_M)} \\ &= \log_{10}\frac{\Pr(\mathbf{y}\mid\mathbf{x};\Gamma(t))}{\Pr(\mathbf{y};\Gamma_T)},\end{aligned} \tag{2.1}$$

where the likelihood in the denominator is factored in the second step. On large pedigrees, the numerator cannot be calculated exactly. To approximate the likelihood under Γ, we express it as an expectation over all possible IBD patterns $\mathcal{P}_{2n}$ on the pedigree:

$$\begin{aligned}L(t) =& \Pr(\mathbf{y}\mid\mathbf{x};\Gamma(t)) \\ =& \sum_{\mathbf{p}\in\mathcal{P}_{2n}}\Pr(\mathbf{y}\mid\mathbf{p};\Gamma_{T(t)})\Pr(\mathbf{p}\mid\mathbf{x};\Gamma_M) \\ =& \mathrm{E}_{\mathbf{p}|\mathbf{x};\Gamma_M}\left[\Pr(\mathbf{y}\mid\mathbf{p};\Gamma_{T(t)})\right].\end{aligned}$$

In this equation, $\Gamma_{T(t)}$ is the marginal trait model, Γ_T, augmented by the trait location hypothesized in $\Gamma(t)$. Simulating B realizations $\mathbf{p}^i$ from $\Pr(\mathbf{p}\mid\mathbf{x})$ and computing the likelihood of the trait data as a function of $\mathbf{p}$, we obtain a Monte Carlo estimate of $L(t)$:

$$\hat{L}(t) = \frac{1}{B}\sum_{i=1}^{B}\Pr(\mathbf{y}\mid\mathbf{p}^i;\Gamma_{T(t)}). \tag{2.2}$$

Tong and Thompson (2008) developed methods for efficient MCMC sampling of

p on large pedigrees. The MORGAN software package implements these methods in order to estimate LOD scores using Equation 2.2. Newer MORGAN programs separate the sampling of $\mathbf{p}$ conditional on marker data and the computation of LOD scores: *gl_auto* samples IBD graphs given marker data, and *gl_lods* takes these graphs as input and produces LOD scores.

(b) Inference without pedigrees

We wish to adapt the LOD score methodology to situations where the pedigree is not available. One approach would be to sample IBD graphs using *ibd_stitch*, then use them to calculate trait likelihoods as in Equation 2.2. However, using IBD graphs sampled in a population in place of pedigree IBD graphs is inappropriate: to calculate a LOD score, we need the pedigree to calculate the trait likelihood in the denominator of Equation 2.1. More generally, normalizing $\hat{L}(t)$ by the constant factor $\Pr(\mathbf{y};\Gamma_T)$ works in a pedigree because the variation across loci is due to tightly constrained changes in descent patterns in the pedigree. When IBD is estimated without pedigrees, the average level of kinship tends to vary across the chromosome, so the variation in $\hat{L}(t)$ is driven more by average relatedness than by concordance between the trait and the estimated IBD pattern at a locus.

We want a way to normalize $\hat{L}(t)$ so that we can compare the degree to which different loci have IBD patterns which accord with the trait. To correct for the differences in likelihood due to varying levels of estimated relatedness, we can permute the trait values assigned to the edges of our sampled IBD graphs. We generate D random permutations σ^j of the trait data. For a particular IBD graph $\mathbf{p}^i$, the term

$$\frac{1}{D}\sum_{j=1}^{D}\Pr(\sigma^j(\mathbf{y}) \mid \mathbf{p}^i;\Gamma_{T(t)})$$

gives a measure of trait likelihood at t holding fixed the number and size of the IBD groups (and hence the estimated relatedness), but omitting information about

assignments of trait values to IBD groups. Summing this quantity over sampled IBD graphs, we obtain the estimator

$$\tilde{L}(t) = \frac{\frac{1}{B}\sum_{i=1}^{B} \Pr(\mathbf{y} \mid \mathbf{p}^i; \Gamma_{T(t)})}{\frac{1}{D}\sum_{j=1}^{D} \frac{1}{B}\sum_{i=1}^{B} \Pr(\sigma^j(\mathbf{y}) \mid \mathbf{p}^i; \Gamma_{T(t)})}$$

which measures the relative likelihood of the trait given the sampled IBD graphs to the trait given the sampled level of average kinship at the locus. We can treat $\log(\tilde{L}(t))$ as a pseudo LOD score because we expect it to be near zero where the the trait and IBD are uncorrelated and high when the IBD reflects the allelic similarity driving the trait.

2.5 Simulations

(a) Iceland pedigree

The *ibd_stitch* program was applied to simulated data to assess its ability to detect linkage signals. The first set of test data was one for which the true descent pattern was known. The data were generated using a subset of a large Icelandic pedigree provided by J.H. Edwards in 1995 (Thompson, 2000). The pedigree spans twelve generations and contains 107 individuals. We use it as a stand-in for the unobserved relationships that might connect several pedigrees collected in a small population. Three families from recent generations were designated as "observed" individuals.

The descent pattern and haplotypes for this data were both simulated. One founder was chosen to be the origin of a trait-associated gene in the pedigree, and a descent pattern was chosen to propagate the gene to each of the three observed families. Descent across the rest of the chromosome was simulated according to Mendelian laws, conditional on the pattern at the trait locus.

Once the descent within the pedigree was simulated, assigning haplotypes to founders determined haplotypes for the entire pedigree. In order to produce real-

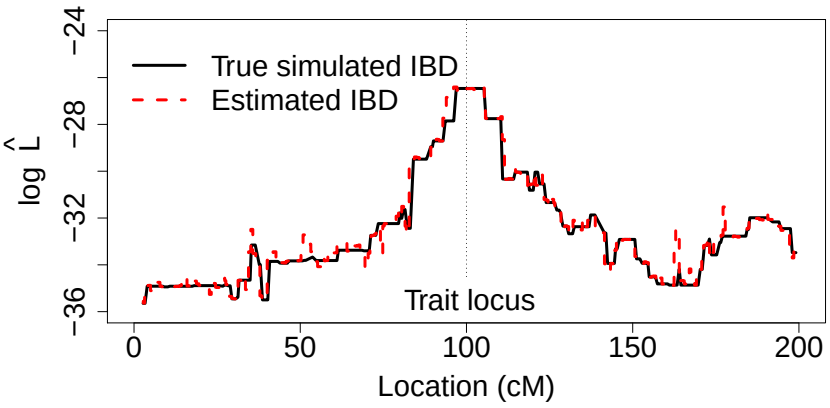


Figure 2.5: Unnormalized likelihoods for a simulated quantitative trait, calculated using the true IBD graph used to simulate the data and IBD graphs estimated using *ibd_stitch*.

istic patterns of linkage disequilibrium (LD), founder haplotypes were simulated from a model fit to 1917 X chromosomes of males in the Framingham Heart Study (Cupples et al., 2009); male X chromosomes were used because they come as naturally phased haplotypes and do not require statistical phasing.

Direct use of the Framingham haplotypes as founder chromosomes was inconvenient for two reasons: first, we desired the ability to generate arbitrary numbers of founder chromosomes in order to be able to populate larger, fixed census size populations (see Brown et al., 2012). Second, permission to use the Framingham data was time-limited (Kraja et al., 2009). To avoid these complications, the BEAGLE (Browning, 2006) and *beaglesim* programs were used to simulate haplotypes with the desired level of LD.

First, LD in the Framingham chromosomes was modeled using BEAGLE, which fits a variable length Markov chain (VLMC) to marker data in order to capture the cluster structure of the haplotypes. The fitted VLMC was then used as input to *beaglesim*, which simulates haplotypes from the model. The generated haplotypes

have the LD structure of the original haplotypes but do not contain any personal data. The LD level in the simulated haplotypes was further controlled by including some probability of jumping to a random haplotype cluster at each locus. This step can be seen as simulating additional descent and recombination in a large population of haplotypes, lowering the observed levels of LD, as determined by pairwise R^2 values (Brown et al., 2012). A jump probability of 0.2 per marker was used, limiting LD to a range of five markers on average and producing a population with lower LD than the original Framingham population.

A total of 10,188 markers was simulated over roughly 200 cM of chromosome. Two hundred independent samples were generated using *ibd_stitch*, with the average prior kinship set to 0.05 and the δ in the discretization step set to 0.1. The genotyping error rate was set 0.01 (although no errors were simulated in the data). These parameters apply to the analysis of the Iceland data in this chapter and in Chapters 3 and 4.

The quantitative trait was generated by designated one founder genome label as the "trait allele" and assigning mean trait values according to the segregation of this label. Genotype means were 0, 4, and 5 for individuals carrying zero, one, or two copies of the trait allele, respectively, and the trait was simulated as a standard normal distribution with variance 4, independent across individuals given the trait genotype.

Because the descent pattern was simulated, the true IBD among the individuals was known. This IBD is the target of the algorithm, so the sampled IBD graphs are compared against the true IBD. Figure 2.5 shows estimates of $\hat{L}(t)$ scores based on IBD graphs from the Iceland data, including the curve calculated from the true IBD and the curve obtained by sampling using *ibd_stitch*. As discussed in Section 2.4, without the pedigrees of the families composing the dataset we cannot calculate the trait likelihoods necessary for a LOD score.

The $\tilde{L}(t)$ function described above is calculated for the sampled IBD graphs and presented in Figure 2.6. The curve behaves as expected, achieving a maximum at the

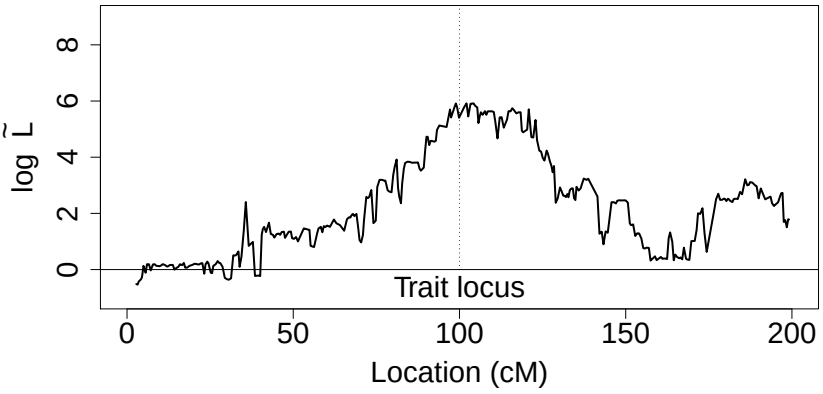


Figure 2.6: Permutation-normalized log-likelihoods ($\log\tilde{L}(t)$) calculated at points along the simulated Iceland chromosomes.

simulated trait locus and dropping off with distance from the trait locus.

Chapter 3

UNPHASED AND PARTIALLY PHASED GENOTYPES

3.1 Genotype phasing

The input to the *ibd_stitch* program discussed in the last chapter includes fully phased SNP haplotypes. In practice, most genetic data come as unphased haplotypes. In this chapter we discuss the issue of haplotype phase, then discuss how it can be incorporated into *ibd_stitch*.

Humans carry two copies of each chromosome. Genetic data generated typically come in the form of an unordered pair of alleles at each locus; one cannot tell which copy a particular allele lies on. The process of assigning alleles to particular haplotypes is called phasing.

In some situations, when long segments of genome are observed, haplotypes can be obtained directly from samples (Fan et al., 2011). However, the microarray and next-generation sequencing technologies in current use detect variants on short segments of genome, so the data produced are unphased. In this case it is possible to phase using statistical models which make use of the haplotype structure in the population (Browning and Browning, 2007b; Kong et al., 2008; Li et al., 2010; Howie et al., 2012).

The IBD transition process introduced by Thompson (2008) included two models, one for phased haplotypes and one for unphased genotypes. The latter is a projection of the model for 15 IBD states into 9 genotypic IBD states. The model for phased data detects IBD more accurately, reflecting the additional information contained in phased data (Brown et al., 2012). However, the model assumes that phase is completely known with certainty at all loci. This is a strong assumption which does not hold for statistically phased data. We desire a more flexible description of phase

information so that modeling assumptions reflect the available data.

Another reason a more flexible model is needed is that the sampling algorithm requires a choice of a specific haplotype ordering when it conditions pairwise state inference on previously sampled pairwise states. The unphased data model of (Thompson, 2008) conflates genotypically equivalent states, so an inferred state at a locus may indicate, for example "this autozygous individual's two gametes are shared IBD with either the first or second gamete of that individual." A collection of pairwise genotypic states will create a complex network of such "or" statements, which may not be mutually compatible. The method of Glazner and Thompson (2012) took IBD sampled within pedigrees and marginal probabilites of IBD calculated using *ibd_haplo* and used a logical satisfiability solver to find joint IBD states compatible with as many of the estimated IBD states as possible. In that algorithm, ambiguity in haplotypic ordering was incorporated into the search for compatible joint states, so it was possible to use the unphased model. When we are connecting pairwise inferences by conditioning, we require an explicit specification of haplotype ordering, even if any two orderings at a locus are indistinguishable using the available data.

To obtain a flexible model with explicit haplotype ordering, we incorporate allelic ordering into the model as a random variable. Assume that each individual's two chromosomes are specified in some fixed, arbitrary order (although we may not be able to determine which is which from the data): say, maternal then paternal. Then the true biological alleles on each chromosome are latent variables, observed only through a random variable which specifies whether the alleles are reported in the correct order or flipped.

To describe this random variable, we refer to the group of permutations generated by the operations which switch an individual's two alleles, as described by (Thompson, 1974). Elements of this group will be denoted g_k: using permutation notation, we let $g_0 = ()$ (the identity permutation), $g_1 = (12)$, $g_2 = (34)$ and $g_3 = (12)(34)$. The group g transforms the allelic data at a locus; for example, $g_1(TGGT) = GTGT$. It also acts

on the set of IBD states: $g_1(\{\{A_m, B_m\}, \{A_p\}, \{B_p\}\}) = \{\{A_p, B_m\}, \{A_m\}, \{B_p\}\}$. Let the random variable R_t be the ordering of the data at locus t, taking values in g which specify an ordering relative to the fixed ordering. For example, if the pair's alleles at t in maternal-paternal order are $ACAC$ and are reported in the data set as $CAAC$, then $R_t = g_1$.

We can perform calculations with R_t using this notation and the identity 3.1 below. Applying the same transformation to the allelic data and the IBD state same way preserves the emission probability. If we observe the ordered alleles y_t at locus t, then given the IBD state i at t and the ordering of the alleles R_t, the emission probability is $e(x_t|i, R_t)$. For any k,

$$e(x_t|i, R_t) = e(x_t|g_k(i), g_k(R_t)). \tag{3.1}$$

3.2 Phased data

With a means of describing haplotype phase and performing emissions calculations with it, we can examine how the HMM changes when augmented with the ordering variable R_t. We encode phase information as a prior distribution $\pi(\mathbf{R})$ on the vector $\mathbf{R}$ of ordering at all loci. The model presented in Chapter 2 uses phased data and implicitly conditions on R_t having the same value at all loci. When expressed in terms of $\mathbf{R}$, emission probabilities not conditioned on R_t are calculated using the fact that $\pi(R_t = g_k)$ is one for some k' and zero for all others:

$$e(x_t|i) = \sum_k e(x_t|i, R_t = g_k)\pi(R_t = g_k) = e(x_t|i, R_t = k'). \tag{3.2}$$

In other words, the model described in terms of haplotype phase is identical to the original model when the data are completely phased.

3.3 Unphased data

When data are unphased, we have no prior information about $\mathbf{R}$; we therefore place an independent, uniform prior on R_t for each t. We obtain emission probabilities by marginalizing over R_t. Naively, this would involve averaging the emission probabilities of a state over the four possible ordered genotypes corresponding to the observed genotypes. Equation 3.1 means we can calculate the emission probabilities once for a particular ordering, then average probabilities over groups of states.

$$\begin{aligned} e(x_t|i) &= \sum_k e(x_t, R_t = g_k|i) \\ &= \sum_k e(x_t|i, g_k) \Pr(g_k) \\ &= \frac{1}{4} \sum_k e(x_t|i, g_k) \\ &= \frac{1}{4} \sum_k e(x_t|g_k^{-1}(i), g_1) \end{aligned}$$

The emission probabilities are averaged over equivalence classes of states which map to one another under g, known as "orbits". Orbits in this case are sets of genotypically equivalent states. This averaging is the only modification required for the HMM to model unphased data.

Collapsing the results of this model down to nine genotypic states produces the same output as the genotypic model of Thompson (2008). The advantage of the haplotypic model is that the resulting trajectories specify haplotypes, even though any particular haplotype has the same probability as its image after transformation by an element of g. Working with haplotypic states is necessary for generating joint states via conditioning.

3.4 Partially phased data

The model described above can be used with phased or unphased data, but the formulation in terms of R_t can be extended to intermediate situations as well. As noted above, there are many ways to phase SNP data, and incomplete or uncertain phasing is a possibility with all of them. Incorporating the incompleteness and uncertainty into our model produces a closer fit between modeling assumptions and reality, and allows us to use partially phased data without simplification.

Genotypes can be partially phased in two ways which are modeled differently. The first, which we will call absolute phasing, is when allelic data at some loci are oriented with respect to some ordering that applies to the entire chromosome, across unphased gaps. A situation where this might arise is when IBD has been estimated within subpedigrees in the sample. Given a particular IBD graph and genotypes, phase can be resolved at some fraction of loci, depending on levels of homozygosity in the sample. All phased loci are ordered in terms of the designation of maternal and paternal haplotypes in the IBD graph. A similar situation arises in the long-range phasing method of Kong et al. (2008), which phases by identifying long haplotypes using a deterministic algorithm and resolves some, but not all, loci. Browning and Browning (2009) used long-range phase information provided by parent-offspring trios to improve genotype imputation and phasing in populations. Lin et al. (2004) also used parent-offspring relationships for phasing. In these approaches, the phasing is specified with respect to some common ordering across the entire chromosome (although the maternal/paternal ordering may be unknown).

Partial absolute phasing can be modeled by altering the emission probabilities, as we did for unphased data. If loci are simply designated as either unphased or phased, we perform the averaging over orbits described in Section 3.3 at the unphased loci and leave the phased loci unchanged. If instead we have estimated probabilities of various orderings (for example, phasings averaged over an ensemble of IBD graphs)

we use them as prior distributions for R_t and marginalize accordingly:

$$\begin{aligned} e(x_t|i) &= \sum_k e(x_t, R_t = g_k|i) \\ &= \sum_k e(x_t|i, g_k) \Pr(g_k) \\ &= \sum_k e(x_t|i, g_k) \Pr(g_k) \\ &= \sum_k e(x_t|g_k^{-1}(i), g_1) \Pr(g_k) \end{aligned}$$

Again, we need calculate the phased emission probabilities only once, and then take an expectation over allelic orderings using probabilities calculated for other states.

The other type of partial phase information will be called relative phasing. This situation arises when it is possible to phase continguous segments of loci, but phase is not known across gaps between segments. For example, BEAGLE (Browning, 2006) models local linkage disequilibrium to phase haplotypes, but accumulating switch errors mean that widely separated loci are likely to be out of phase (Browning and Browning, 2007b). The phase information in this case is not in the orientation of genotypes to some common ordering, but in the orientation of adjacent loci to one another. As a result, instead of modifying the emission probabilites, we modify the transition process between loci.

Recall from Section 1.3 that we define our HMM in terms of emission probabilities $e(x|i)$ and transition probabilities $d_t(j|i)$ (as well as an initial distribution). In the standard HMM algorithm (Rabiner, 1989), we calculate forward probabilities, defined as:

$$\alpha_t(i) = \Pr(S_{t-1} = i, X_{1:t-1}) = e(x_t|i) \left[\sum_j \alpha_{t-1}(j) d_t(i|j) \right].$$

(Analagous backward probabilities are calculated when computing exact marginal

probabilities.) Suppose loci have been partitioned into phased segments. We can represent this by giving $\mathbf{R}$ a prior distribution which gives each segment a constant value of R_t, distributed uniformly and independently of other segments. The changes to the model are implemented by modifying the calculation of the forward probabilities.

Assume we have calculated the forward probabilities at locus $t-1$, and a new phased segment begins at locus t. We begin by calculating modified forward probabilities assuming some particular ordering for the emission probabilities:

$$\alpha_{t,g_0}(i) = \Pr(S_t = i, X_{1:t} | R_t = g_0) = e(x_t|i, g_0) \left[\sum_j \alpha_{t-1}(j) d_t(i|j) \right]$$

This continues in this manner until the last locus of the phased segment, t', at which point we have $\alpha_{t',g_0}(i) = \Pr(i, X_{1:t'} | \mathbf{R}_{[t,t']} = g_0)$. We now wish to marginalize over $\mathbf{R}_{[t,t']}$.

Note that the Markov transition process is exchangeable in the order of the chromosomes, so we have for all k

$$d_t(i|j) = d_t(g_k(i)|g_k(j)). \tag{3.3}$$

Because $\alpha_{t',g_0}(i)$ is an expression made up of emission probabilities, emission probabilities, and forward probabilities $\alpha_{t-1}(i)$ which do not depend on ordering, we have that

$$\alpha_{t',g_k}(i) = \alpha_{t',g_0}(g_k^{-1}(i)).$$

As in absolute phasing, we have already calculated the desired quantities for different assumed orderings by calculating them for different states. This means that

integrating out the uniform prior on orderings is a matter of averaging among orbits:

$$\begin{aligned}\alpha_{t'}(i) =& \Pr(i, X_{1:t'}) \\ =& \sum_k \Pr(i, X_{1:t'} | \mathbf{R}_{[t,t']} = g_k) \Pr(R_{t,t'} = g_k) \\ =& \frac{1}{4} \sum_k \Pr(g_k^{-1}(i), X_{1:t'} | \mathbf{R}_{[t,t']} = g_0) \\ =& \frac{1}{4} \sum_k \alpha_{t',g_0}(g_k^{-1}(i)).\end{aligned}$$

Informally, we imagine the forward algorithm progressing through a phased segment under some assumed ordering, then forgetting the ordering used when passing forward probabilities to the next segment. If the forward-backward algorithm is used to produce marginal probabilities, a similar averaging step must be taken during the backwards pass.

Finally, we note that in the limiting case of all the phased segments being single loci, the relative partial phasing model is the same as the unphased model, which is also the limiting case of absolute partial phasing. Absolute and relative phase information are the same in the case of complete ignorance or complete knowledge of phase, but differ in which parts of the phased model they elide.

3.5 Iceland pedigree with unphased data

Figure 3.1 shows $\tilde{L}(t)$ calculated from the same dataset as in Figure 2.6, this time modeled as both phased and unphased data. The new curve has roughly the same shape as the one for phased data, although it is attenuated towards zero. This occurs because discarding the phase information creates more uncertainty in the inferred IBD, and consequently a weaker association between the trait and the IBD.

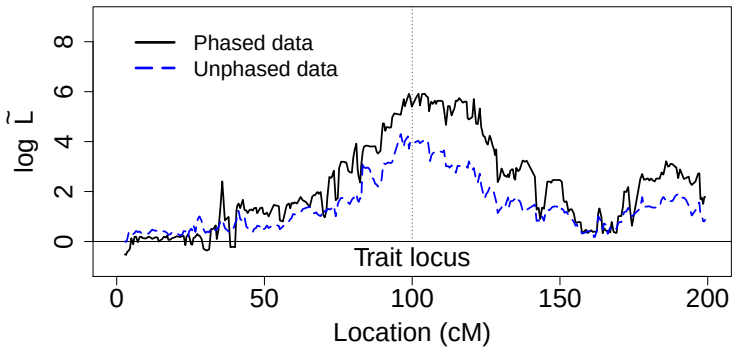


Figure 3.1: Permutation-normalized log-likelihoods ($\log\tilde{L}(t)$) for the Iceland dataset, applied to phased and unphased chromosomes.

3.6 Pig example

We now apply *ibd_stitch* to a real genetic dataset; this was not possible previously because the data in question are unphased genotypes. Genus PIC, a pig genetics firm, provided a data set containing 5772 animals genotyped at 6973 markers. These data overlap with the genotypes made publicly available by Genus PIC and described in Cleveland et al. (2012). The data provided by Genus PIC also included a pedigree containing 11,544 members and a genetic map for the markers. The genetic map in base pairs was converted to centiMorgans assuming 10^6 base pairs per cM.

SNPs were filtered as follows: markers were discarded if the genotypes were missing in more than 5% of animals, as were animals missing more than 5% of their marker genotypes. The markers were further thinned to speed computations and reduce the level of LD in the dataset, as Brown et al. (2012) demonstrated that LD interferes with inference using the pairwise model. Markers with minor allele frequency less than 0.3 were discarded on the reasoning that rare alleles will have high coefficients of variation in the estimated population frequencies. The final SNP dataset contained 1034 markers typed on 5742 animals. Population allele frequencies were estimated

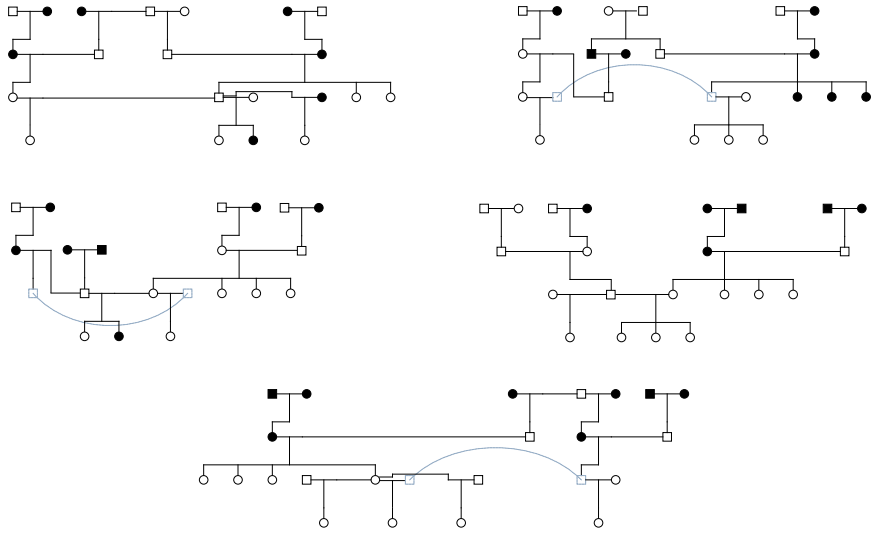


Figure 3.2: The five pedigrees analyzed in the Genus PIC pig data example. The families were selected for having a high density of genotyped animals, depicted with white icons.

from these data.

Five subpedigrees of pigs were chosen for analysis. To obtain closely related genotyped individuals, a pool of subpedigrees was created by choosing random proband animals and selecting all animals within three parent/child relationships of the proband. The candidate subpedigrees were then rendered as graphics with typed animals indicated by color and manually inspected to remove subpedigrees with few genotyped animals. Five subpedigrees with a high proportion of typed animals were selected, comprising a total of 69 animals.

The trait used in the analysis was simulated based on IBD observed in the selected animal. Preliminary samples of IBD graphs on the animals were generated and the IBD was inspected graphically. A locus which appeared to show a small number of well-resolved IBD groups was chosen as a good candidate for simulating a trait based on estimated IBD, and the cleaned marker data for the selected individuals were analyzed using version 3.3.2 of BEAGLE (Browning and Browning, 2007a) as a inde-

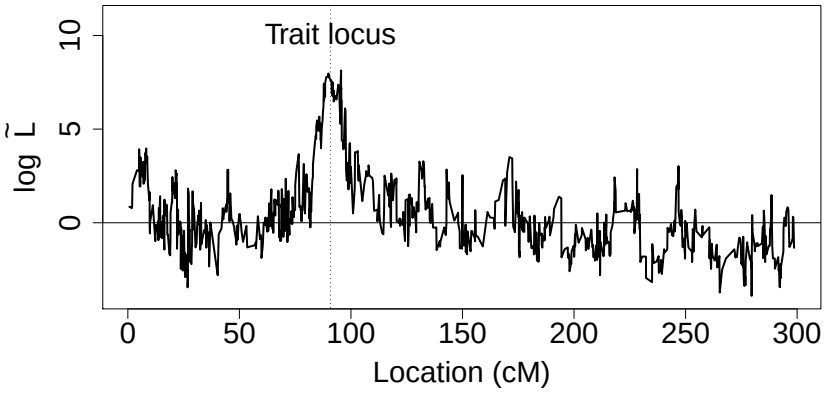


Figure 3.3: Permutation-normalized log-likelihoods ($\log\tilde{L}(t)$) calculated at points along the Genus PIC chromosomes. The vertical line indicates the trait locus.

pendent method of IBD detection. With the default settings, the *beagle* executable was used to fit a model of haplotype clusters in the sample. The *pseudomarker* executable was then used to label haplotype clusters; only two clusters were detected at the locus of interest, so the resulting marker was biallelic. This marker was used as the basis for generation of a quantitative trait using the same parameters as for the Iceland pedigree.

The simulated trait was analyzed using the permutation normalization approach described in Section 2.4. To perform the normalization, 200 permutations were used. As in the Iceland example, the curve spikes at the location of the simulated trait and remains near zero elsewhere. However, the curve is erratic compared to a traditional pedigree-based LOD score, and we caution against interpreting scores above the traditional LOD threshold of three as suggesting trait linkage.

Chapter 4

INCORPORATING PEDIGREE IBD

4.1 Pedigree vs. population IBD

In some situations we have incomplete pedigree information on the sampled individuals; for example, the sample may comprise two families, or a number of small families and single individuals. Pedigrees can be used to infer IBD using models for recombination and segregation. (In this case, IBD is defined with respect to the pedigree founders.) We may want to combine the IBD found in pedigrees with that inferred by the population HMM model.

When combining different methods of IBD inference, we must decide whether two individuals should be considered IBD at a locus if they are IBD according to both methods or merely one method or the other. Since most of the genomes of two individuals will be in the no-IBD state, it is natural to treat IBD as a more interesting property than non-IBD and adopt the broader criterion.

In the case of combining pedigree and population inferences, this choice is consistent with the definition of IBD: if two chromosomes coalesce within a pedigree, their coalesence time is limited by the depth of the pedigree. For pedigrees on which it is feasible to perform MCMC analysis, this time will typically not exceed a few generations. In contrast, the time point defining the population IBD is on the order of dozens of generations; Browning and Browning (2010) suggest that the method they present be used to detect IBD up to 25 generations ago. It is therefore reasonable to treat population IBD as a coarsening of the partition defined by the pedigree IBD; in other words, population IBD is added to a pedigree IBD.

We use the pedigree IBD as a scaffolding upon which to build a joint IBD state

covering several pedigrees. The output IBD state will be a coarsening of the pedigree IBD. We must therefore modify the algorithm to ensure that it is restricted to this subset of IBD graphs. This requires an additional filtering step on the set of allowed HMM states. In the population model, only IBD states sampled during earlier steps in the algorithm must be considered. When we incorporate pedigree IBD graphs, we must also eliminate states that create (or force the later creation of) incompatibilities with the pedigree IBD. As before, the filtering can be performed by querying a data structure representing the IBD graph. Algorithm 2 presents the method with modifications to accomodate pedigree IBD graphs.

Algorithm 2 Sample an IBD graph on n individuals given SNP data and within-pedigree IBD graphs

Require: SNP haplotypes for n individuals and pedigree IBD graphs covering disjoint subsets of the individuals
Combine pedigree IBD graphs into $\mathbf{p}_{2n}^{ped}$
Initialize $\mathbf{p}_{2n}^{pop}$ to the empty IBD graph.
Randomly permute individuals and label them $\{A_1, \ldots, A_n\}$
Sample from $Q[\mathbf{p}_4(A_1A_2)|\mathbf{p}_{2n}^{ped}, \mathbf{x}(A_1A_2)]$ and add the trajectory to $\mathbf{p}_{2n}^{pop}$.
for $2 < k \leq n$ **do**
  Randomly permute individuals $\{A_1, \ldots, A_{k-1}\}$ and label them $\{B_1, \ldots, B_{k-1}\}$.
  for $1 \leq i < k$ **do**
    Sample from $Q[\mathbf{p}_4(B_iA_k)|\mathbf{p}_{2n}^{pop}, \mathbf{p}_{2n}^{ped}, \mathbf{x}(B_iA_k)]$ and add the trajectory to $\mathbf{p}_{2n}^{pop}$.
  end for
end for

return An IBD graph on n individuals, $\mathbf{p}_{2n}^{pop}$.

After applying the constraints imposed by pedigree IBD, the HMM state space will be at most as large as in Algorithm 1, and it may be smaller. Consequently, we must augment the proof that the state space is never empty. Noting that an eliminated state necessarily creates an incompatibility, it is sufficient to show that there is some IBD graph compatible with all three of: the pedigree IBD graph on

all individuals; the sampled population IBD on a subset of individuals F; and the sampled pairwise states between another individual and some of F. Since this IBD graph does not conflict with any of the listed constraints, its projection onto the state space of the pair being sampled cannot be eliminated by the filtering criteria. More details appear as Appendix A.2.

4.2 Iceland example

The Iceland dataset introduced in Chapter 2 is appropriate for illustrating the use of IBD in small pedigrees for estimating IBD in larger groups. In the analysis in Glazner and Thompson (2012), this dataset was divided into three separate pedigrees of size eleven, eight, and twelve. The MORGAN program *gl_auto* was used to generate realizations of IBD within these subpedigrees using the pedigree information. The subpedigree IBD graphs were then combined using the *ibd_haplo* HMM to produce complete IBD graphs for the entire sample; the LOD scores from the combined graphs were higher at the trait locus than the scores from the subpedigrees and comparable elsewhere on the chromosome, giving sharper resolution of the trait.

Unlike the method discussed by Glazner and Thompson (2012), the *ibd_stitch* program can sample IBD graphs without using any pedigree information. Nevertheless, the results in that paper suggest that pedigrees contain information that can be used to detect IBD, and the algorithm outlined above shows how we can incorporate the subpedigree IBD into *ibd_stitch*. The subpedigree IBD graphs generated by *gl_auto* for Glazner and Thompson (2012) were reused as input to *ibd_stitch*; a distinct IBD graph from each subpedigree was used for each separate iteration of the *ibd_stitch* algorithm.

The *gl_auto* program uses widely spaced markers relative to *ibd_stitch*, and the markers used to generate the subpedigree IBD graphs form a sparse framework amonst the input to *ibd_stitch*. To accurately represent the information contained in them, the IBD in the subpedigree graphs was incorporated into the joint state only at the sparse framework markers; between these markers, the method for determining allowed IBD

states was the same as in Algorithm 1, described in Section 2.3.

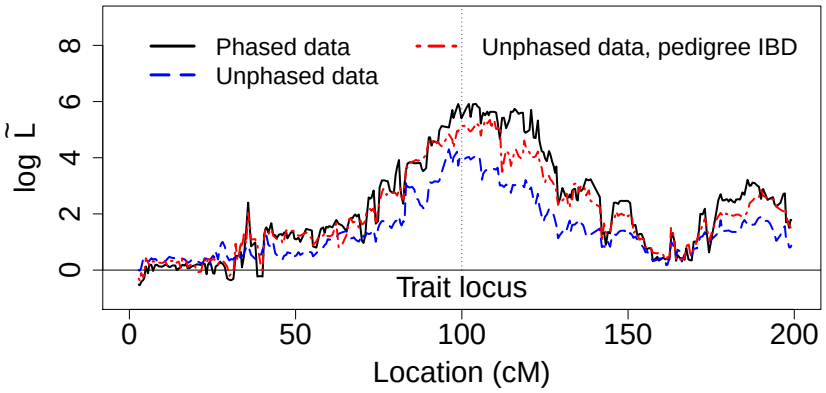


Figure 4.1: Permutation-normalized log-likelihoods from graphs sampled with and without subpedigree IBD.

Figure 4.1 shows analysis of the Iceland data incorporating IBD estimated in the sample separately using pedigrees and *gl_auto*. In this example, incorporating pedigree IBD into the sampling method produces a curve for $\log\tilde{L}(t)$very similar to the one generating using no pedigree information but phased data. It appears that long tracts of pedigree IBD function as a proxy for explicit phase information. As mentioned in Section 3.4, sampled IBD graphs can be used to partially phase a sample, although in this case we have not explicitly applied the methods for partially phased data. This result suggests that such a phasing step maybe be unnecessary if the pedigree IBD is incorporated directly into the sampling process.

The results in Figure 4.2 show a less promising outcome of incorporating sub-pedigree IBD; the purple curve was calculated from graphs sampled using both sub-pedigree IBD and phased haplotype data. The peak of the $\log\tilde{L}(t)$ curve is much less pronounced than the analyses using one or the other, and closer to the analysis using no pedigrees and unphased data. The curve is erratic, and examination of the sampled IBD graphs reveals that this is due to jumps among IBD states occurring at

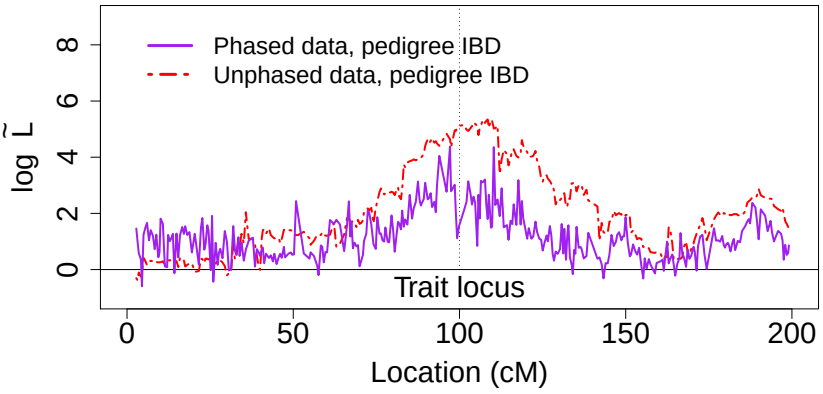


Figure 4.2: Permutation-normalized log-likelihoods from graphs sampled with subpedigree IBD.

a much higher rate than in the other scenarios.

This unexpected behavior is due to an incompatibility between the subpedigree IBD graphs and the haplotype phase given as input to the model. As demonstrated in Figure 4.1, the subpedigree IBD serves as a substitute for phase information. However, depending on pedigree structure, *gl_auto* can output graphs which reflect many different phasings which are equivalent up to permutations of the group discussed in Chapter 3, Section 3.1. The absolute phase orientation of the graphs does not necessarily accord with that of the haplotypes. Consequently, the pedigree IBD incorporated into the joint graphs at the framework markers sometimes differs from the state suggested by the marker data, forcing jumps between IBD states and producing erratic trait likelihoods.

This explanation was confirmed by examining a pair of individuals in the data whom the phased haplotypes strongly suggested shared a gamete IBD over hundreds of loci. Those same loci were examined in the graphs from Figure 4.2 and evidence a distinct disagreement between the sampled IBD states at framework and non-framework markers.

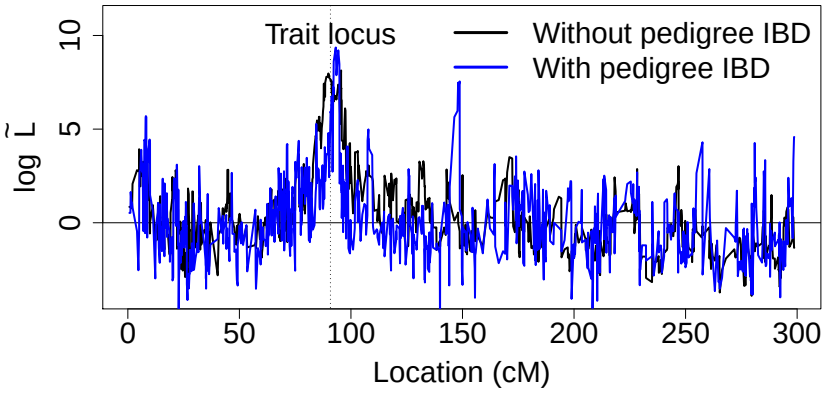


Figure 4.3: Permutation-normalized log-likelihoods for the Genus PIC pig genotypes analyzed with and without subpedigree IBD information.

4.3 Pig example with pedigrees

The Genus PIC pig data were selected for testing *ibd_stitch* because of the availability of large pedigrees for the sample animals. We are able to sample IBD graphs in the subpedigrees depicted in Figure 3.2 using *gl_auto* and incorporate them into the sampling of *ibd_stitch*. Of the 1034 markers used in the analysis, every tenth marker was selected for use in subpedigree IBD sampling, reflecting a density typical for pedigree linkage analysis.

The results of these steps appear in Figure 4.3. As in the Iceland example, the peak of the permutation-corrected likelihood curve is slightly higher when the subpedigree IBD is incorporated. However, the curve is even more erratic than in the Iceland example, jumping rapidly from locus to locus, with a large false signal around 150 cM. These problems may arise due to the relatively higher density of subpedigree-sampled markers among the markers used for sampling between all individuals; it may be preferable to enforce subpedigree IBD at sparser intervals in order to capture phase information without excessively constraining the population IBD sampling pro-

cess. There may also be differences in the constraints imposed by the subpedigrees compared to the Iceland example, due to the larger families involved.

Chapter 5

FITTING THE FULL MODEL USING PARTICLE GIBBS SAMPLING

5.1 Sequential Monte Carlo methods

The HMM forward algorithm for our model is infeasible for groups larger than three or four individuals because of the need to enumerate the state space, which grows rapidly in the number of chromosomes (see Section 1.4). Also, the discretization step in *ibd_stitch* is unsatisfying; we would prefer a method that samples from the full continuous model. An alternative class of methods, described generally as sequential Monte Carlo (SMC), deals with the problem of large or continuous hidden state spaces by exploring the space probabilistically. Whereas the forward algorithm scans the state space, considering each possible source and destination state for a given transition (as in Rabiner, 1989, Eq. (20)), an SMC algorithm simulates random transitions from a randomly chosen set of starting states, or particles. The result is a Monte Carlo approximation to the filtering distribution of the HMM, so this step is known as particle filtering. Additional steps are required to replace the backwards sampling step and produce samples from the full hidden trajectory.

In this chapter, it will be understood that all IBD states comprise the entire sample of individuals being analyzed; p is implicitly an element of $\mathcal{P}_{2n}$. The subscript is repurposed to indicate the marker position of the IBD state, numbered from 1 to T. Similarly, x_t indicates the marker data of all individuals at position t, and $x_{1:t}$ the data of all individuals at positions 1 through t.

5.2 Importance sampling

SMC methods are a form of importance sampling (Clark, 1961) for models which can be described as a sequence of distributions. The linear structure of the data in our HMM produces the sequence of filtering distributions

$$\Pr(p_1|x_1), \Pr(p_2|x_{1:2}), \ldots, \Pr(p_T|x_{1:T}) \tag{5.1}$$

which can be calculated using the forward algorithm (see Section 3.4) in the case of a small state space. In SMC, we adapt importance sampling to replace the function of the forward algorithm.

In a general setting, given a target distribution P which is hard to sample from, we can estimate the expectation of a function h on the sample space of P using samples from an importance distribution R. We require that P is absolutely continuous with respect to R. From R we generate a set of B samples, or particles, z^i. Expectations can then be calculated as a weighted average of samples from R:

$$\begin{aligned} w^i =& \frac{P(z^i)}{R(z^i)} \\ E[h(z)] \approx& \left(\frac{1}{\sum_{i=1}^{B} w^i} \right) \sum_{i=1}^{B} h(z^i) w^i. \end{aligned}$$

We now illustrate importance sampling for our HMM with only a single locus; we wish to approximate $\Pr(p_1|x_1)$. A set of B particles is generated from a proposal distribution R. Each particle's weight is calculated as the ratio of the target distribution to the sampling distribution:

$$w_1^i = \frac{\Pr(p_1^i|x_1)}{R(p_1^i)}.$$

This allows us to estimate expections of arbitrary functions of p_1 given x_1; as in the

general case, we have

$$E_P[h(p_1)|x_1] \approx \left(\frac{1}{\sum_{i=1}^{B} w^i} \right) \sum_{i=1}^{B} h(p_1^i) w^i.$$

This is standard importance sampling (Robert and Casella, 1999, Section 3.3). Since we are interested in sampling IBD states from the distribution $\Pr(p_1|x_1)$, we choose $h(p_1)$ to be a point probability mass $\delta(p_1)$ at the given IBD state, so that the resulting expectation is a multinomial distribution with probabilities equal to the normalized weights. This estimate converges in distribution to $\Pr(p_1|x_1)$ as B goes to infinity.

In this and subsequent examples of importance sampling for an HMM, choosing a particular proposal distribution simplifies the weight calculation. If we choose R to be the initial distribution $\Pr(p_1)$ (or, more generally, we propose from the hidden process), the weights simplify to

$$w_1^i = \frac{\Pr(p_1^i|x_1)}{R(p_1^i)} \propto \frac{\Pr(x_1|p_1^i)\Pr(p_1^i)}{\Pr(p_1^i)} = \Pr(x_1|p_1^i). \tag{5.2}$$

The weights enter into the estimate only up to a constant of proportionality, so it is sufficient to weight by the data likelihood $\Pr(x_1|p_1^i)$.

5.3 Sequential importance sampling and resampling

We now extend the method to target the complete trajectory of the hidden process, $\Pr(p_{1:T}|x_{1:T})$. A naive approach, known as sequential importance sampling (Doucet, 2001), is to simulate B independent hidden process trajectories $p_{1:T}^i$, then calculate the weight for each trajectory as the observed data likelihood, by the same reasoning as in expression (5.2):

$$w^i \propto \Pr(p_{1:T}^i|x_{1:T}) \propto \prod_{t=1}^{T} \Pr(x_t|p_t^i). \tag{5.3}$$

The flaw in this approach is that the magnitude of the likelihoods will be small in comparison with the variation among trajectories. As a result, the normalized weights will put most of the probability on a small number of paths, producing a highly degenerate approximation to the target distribution.

To avoid this degeneracy, we perform a resampling step on the pool of particles at each locus. Imagine the sequential importance sampling method as a series of steps on a pool of particles, rather than a set of independent trajectories. The input to the calculation at a locus is a set of particles with their accumulated weights, representing an approximation $\hat{P}_{t-1}$ to the filtering distribution $\Pr(p_{t-1}|x_{1:t-1})$. To obtain $\hat{P}_t$, we "multiply" $\hat{P}_{t-1}$ by $\Pr(p_t|p_{t-1})$ and $\Pr(x_t|p_t)$ via forward simulation of particles and weighting, respectively. By sampling from $\hat{P}_{t-1}$ before propagating the particles forward to locus t, we obtain a new, unweighted approximation of the same distribution. This step refreshes the diversity of the pool of particles and mitigates the accumulation of degenerate particle weights.

After propagating, weighting, and resampling from loci 1 through t, we have the following approximation to the filtering distribution at T:

$$w_i = \frac{\Pr(p_t^i|x_{1:t})}{\Pr(p_{t-1}|x_{1:t-1})R(p_t^i|p_{t-1}^i)}$$

$$E[h(p_T)|x_{1:T}] \approx \left(\frac{1}{\sum_{i=1}^{T} w^i}\right) \sum_{i=1}^{T} h(p^i)w^i.$$

This approximation includes only the one-step weights rather than weights accumulated over the entire chromosome, and as a result is much less degenerate than 5.3. The disadvantage is that our independent trajectories have become intertwined, placing constraints on how we might split up the calculation.

As mentioned above, choosing the proposal distribution to match the hidden process simplifies the weight calculation considerably. In the case of the models to be used here, we are forced to use the hidden Markov process as the proposal distribution, R,

because we cannot calculate the probability of a simulated transition, which becomes part of the weight calculation if a different proposal is used. With this choice, the expression for the weights simplifies to

$$w_t^i = \Pr(x_t|p_t^i),$$

the probability of the allele data at a locus given the underlying IBD state.

5.4 Particle Gibbs sampling

The sequential importance resampling algorithm produces a Monte Carlo approximation to $\Pr(p_T|x_{1:T})$. This product is directly useful only if we have a particular interest in the hidden state at locus T. In our case, we require trajectories of IBD states along a chromosome, samples from $\Pr(p_{1:T}|x_{1:T})$.

To perform this sampling, we must embed this basic particle filtering step in a Markov chain Monte Carlo (MCMC) algorithm, using an approach described by (Andrieu et al., 2010). Those authors describe two methods for wrapping a particle filter in MCMC machinery; one is based on Metropolis-Hastings sampling, and the other on Gibbs sampling. We will focus on implementation of the latter. The software components needed for the two methods largely overlap, and we implemented particle independent Metropolis-Hastings for small examples during the development process. We observed that the two methods failed to mix on the same datasets, suggesting they are roughly comparable in computational power.

The particle Gibbs (PG) algorithm is initialized by running the basic particle filter and, at each step, saving both the particles p_t^i and the indices of the ancestral particles chosen in the resampling step at time $t-1$, denoted a_{t-1}^i. A single specially designated particle at time T is sampled according to the weights at that time; this particle and its lineage of ancestral particles, $p_{1:T}^*$, are treated as a sampled trajectory.

The Markov chain is advanced by performing a particle filtering step conditionally

on $p^*_{1:T}$. This trajectory is fixed, and the particle pool at time t is guaranteed to include at least one copy of p^*_t. The other $B-1$ particles are sampled as before. The particle p^*_t always has at least one descendant, p^*_{t+1}.

After this conditional step, we have another set of sampled particles and ancestors. A particle is again sampled from the weights at time T, and its trajectory is the next MCMC-sampled trajectory, as well as the seed for the next step of the algorithm.

The PG algorithm can be understood as a standard Gibbs sampler on the extended state space $\{\mathbf{p}^i_t, \mathbf{a}^i_t, k\}$ of BT particles, $B(T-1)$ ancestors, and the final index of the specially designated trajectory. Let $r(a^i_t|\mathbf{w}_t)$ be the resampling distribution, and $\psi(\mathbf{p}^i_t, \mathbf{a}^i_t)$ be the distribution generated by the basic particle filter. The superscript k designates a particle or ancestor belonging to the specially designated trajectory. The target distribution of the PG sampler can be written as

$$\tilde{\pi}\{\mathbf{p}^i_t, \mathbf{a}^i_t, k\} = \frac{\Pr(\mathbf{p}^k_{1:T}|x_{1:t})}{N^T} \frac{\psi(\mathbf{p}^i_t, \mathbf{a}^i_t)}{\Pr(p^k_1)\prod_{t=2}^T r(a^k_{t-1})\Pr(p^k_t|p^k_{t-1})} \tag{5.4}$$

The marginal distribution of $p^k_{1:T}$ under $\tilde{\pi}$ is our desired sampling distribution $\Pr(p_{1:T}|x_{1:T})$. A derivation of this result appears in the supporting material of Andrieu et al. (2010).

The PG algorithm consists in updating the non-designated particles and ancestors alternately with updating the designation index k. The distribution of the non-designated particles and ancestors conditional on $\{k, \mathbf{p}^k, \mathbf{a}^k\}$ is the conditional particle filter. This can be seen by noting that the second ratio in Equation 5.4 is the distribution generated by the conditional particle filter. The conditional distribution of k given $\mathbf{p}^k, \mathbf{a}^k$ is simply the normalized weights at time T. Omitting constant terms from Equation 5.4, the distribution is proportional to:

$$\begin{aligned}\frac{\Pr(\mathbf{p}^k_{1:T}|x_{1:t})}{\Pr(p^k_1)\prod_{t=2}^T r(a^k_{t-1})\Pr(p^k_t|p^k_{t-1})} &= \frac{\Pr(p^k_{1:T}|x_{1:t})}{\Pr(p^k_1)\prod_{t=2}^T \Pr(x_{t-1}|p^k_{t-1})\Pr(p^k_t|p^k_{t-1})} \\ &= \Pr(x_T|p_k)\end{aligned}$$

which is simply the unnormalized weight of particle k at time T.

5.5 Convergence and mixing

In the next section, we will apply PG sampling to a simulated data set. Unlike the algorithm introduced in Chapter 2, PG sampling targets the full model distribution, provided the chain converges. The tradeoff for this model correctness is a reduction in the size of datasets for which we can perform sampling. The datasets used in Chapters 2 through 4 are too large to achieve convergence in a reasonable amount of time using our software, so we will use a subset of the data to examine the performance of PG sampling.

To discuss issues of PG convergence, we apply some population genetics vocabulary to the population of particles generated by the sampler. A new trajectory is created by sampling a particle p'_T from the pool at locus T. Each particle in the pool has an associated ancestral lineage, and the new trajectory is the lineage of the sampled particle. One particle at locus T is the representative p^*_T of the previously sampled trajectory. If the two lineages $p'_{1:T}$ and p^*_T coalesce, they will be identical at every point earlier than their coalescence.

The probability that a new trajectory is sampled for the entire chromosome is the probability that these two ancestries do not coalesce, so we can borrow intuition from the rich literature on the coalescent (Kingman, 1982). Briefly, the two parameters determining coalescence probabilities are the length of the time interval in question and the effective population size. The implication for PG sampling is that to achieve mixing, we can analyze fewer markers or increase the number of particles.

5.6 Application to simulated Iceland haplotypes

A small data set was constructed from the Iceland data introduced in Section 2.5 by reducing the number of individuals and thinning the markers. Eight individuals, including at least two from each subpedigree, were chosen by looking for high levels

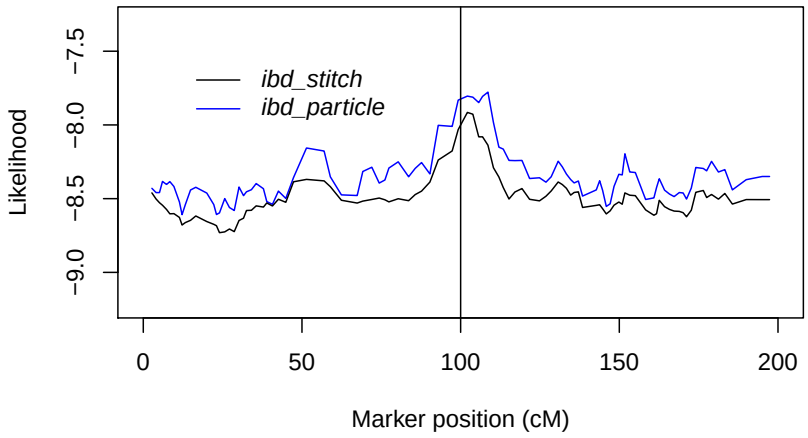


Figure 5.1: Pointwise median trait likelihoods for reduced Iceland data set chromosomes, calculated from IBD graphs sampled using *ibd_stitch* and *ibd_particle*. Vertical line indicates trait locus.

of IBD sharing in the analysis of Section 2.5. It should be noted that this sample is not intended to represent a real study sample, but merely to capture some of the quantitative trait likelihood signal analyzed in Section 2.5. An analysis of all 31 Iceland individuals was attempted, but mixing was limited to a small fraction of markers at the end of the chromosome, even when using 100,000 particles.

Markers were thinned considerably to a density of roughly one per 2 cM, for a final count of 102 markers. Several choices of particle population size B were tested, and a reasonable tradeoff between computation time and mixing rate was found at a value of $25,000$ particles. The haplotypes were analyzed as phased; the modifications to the emission function for unphased data described in Chapter 3 require enumeration of the state space, and are not computationally feasible.[1] No pedigree information was used in this example.

[1]There are 10,480,142,147 hidden states in this example.

Figure 5.1 shows the unnormalized trait likelihoods calculated from the PG-sampled IBD graphs using the trait analyzed in previous chapters, compared to those sampled using *ibd_stitch* with the same model parameters. Pointwise medians were calculated instead of estimated LOD scores to reduce Monte Carlo variance; we only interested in comparing the two methods in their detection of the likelihood signal, and in this case the true variation along the chromosome is small relative to the Monte Carlo error. Also, LOD scores are averaged on the natural scale (see Equation 2.2), so occasional large likelihood values have a disproportionate impact on the result. Since the PG sampler has larger effective sample size at the end than the beginning of the chromosome, LOD scores would tend to be larger at the end than the beginning.

The mixing behavior of the model is assessed in Figure 5.3. Recall that each sampled ancestry may coalesce with the conditioned trajectory from the previous iteration, and when it does the two trajectories agree between that point and the begining of the chromosome. At a particular marker, the figure shows how many iterations had not coalesced with the conditioned trajectory to the right of that marker. In essence, it shows how often a new IBD state was sampled at a locus, although there is a small chance that a IBD state in a newly sampled trajectory will happen to the conditioned one. If we interpret the state at each locus as a component of a multivariate Markov chain, the mixing rate is determined by the slowest-mixing locus at the left end of the chromosome. Thus, we can treat the fraction of iterations sampling a new state at marker one as analagous to a Metropolis-Hastings acceptance ratio. In the case of Figure 5.3, we see that 50,000 particles achieves the traditional rule of thumb of an acceptance ratio of greater than 0.4.

To interpret Figure 5.3 we again use coalescent terminology. The mixing fraction declines significantly around the location of the trait locus, suggesting that particle ancestries tend to coalesce at this point. In the coalescent, a large number of coalescences in a short time window suggests a bottleneck where the effective population size is small. Since in our case the census population size is fixed, the small effective

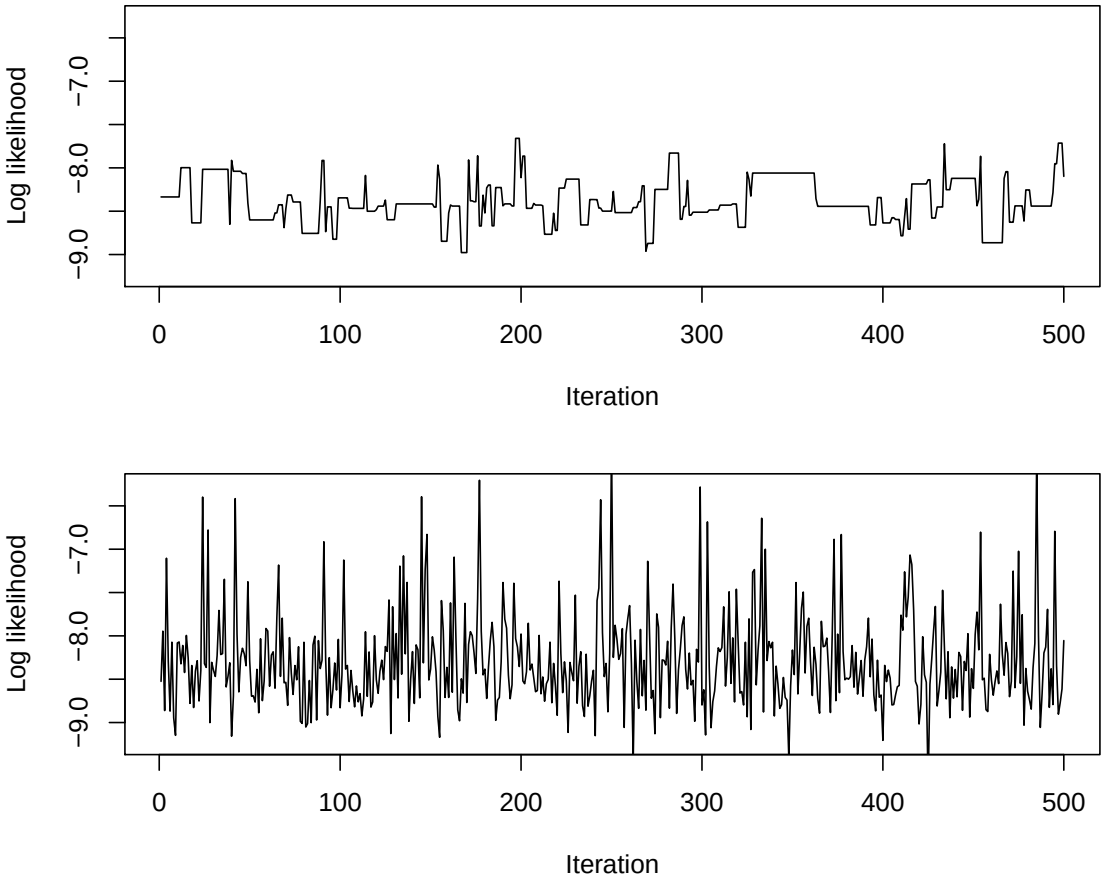


Figure 5.2: Trace plots of the trait likelihood at the first (top) and last (bottom) loci on the chromosome, from the model run with 25,000 particles. Due to properties of the particle Gibbs algorithm used, the Markov chain mixes more rapidly at the end. Figure 5.3 shows mixing rates at all loci and for different numbers of particles.

population size is explained by higher variance in the number of offspring particles (Crow and Kimura, 1970, Section 7.6). This means that we expect particle bottlenecks where there is high variance in the weights at a locus. Variance among the weights will be higher when the data are more strongly informative, or where the IBD changes along the chromosome.

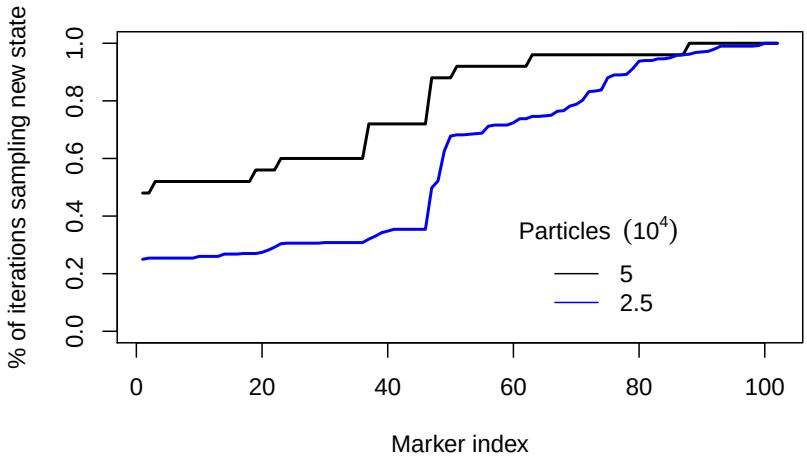


Figure 5.3: Fraction of MCMC iterations sampling a new IBD state at the given marker index.

Chapter 6

COMPARISON OF THREE SAMPLING TECHNIQUES ON A TRIO

6.1 Comparing sampling techniques

We now present a direct comparison of three programs for sampling IBD graphs: *ibd_haplo*, *ibd_stitch*, and *ibd_particle*. As noted in Section 1.4, the number of IBD states among n individuals grows too rapidly for *ibd_haplo* to be run on large groups of individuals. In light of this, we analyze a trio of individuals forming a dataset small enough to be analyzed by all three programs in a reasonable amount of time. The number of IBD states for three individuals is a manageable 203; adding a fourth would expand the state space to 4140 elements.

The marker data for the three individuals were taken from the dataset analyzed by (Brown et al., 2012), which was created using a simulated 200-generation pedigree and founder chromosomes based on LD patterns found in the Framingham Heart Study (Cupples et al., 2009). Details on the simulation study can be found in the article by (Brown et al., 2012). The simulated haplotypes used here can also be found in the examples bundled with *ibd_haplo* in version 3.2 of MORGAN.

A common set of parameters was chosen for each analysis: the population kinship and change rate parameter were both set to 0.1, and the rate of genotyping error was set to 0.05. Each analysis was run for 1000 iterations on a single core of an Intel Core i7 CPU. Running times appear in Table 6.1.

Program	*ibd_haplo*	*ibd_stitch*	*ibd_particle*				
Particles	-	-	250	1000	2000	5000	10000
Time (sec)	49	9	319	1159	2230	5616	11106

Table 6.1: Computation time for 1000 iterations of the trio example on a single core.

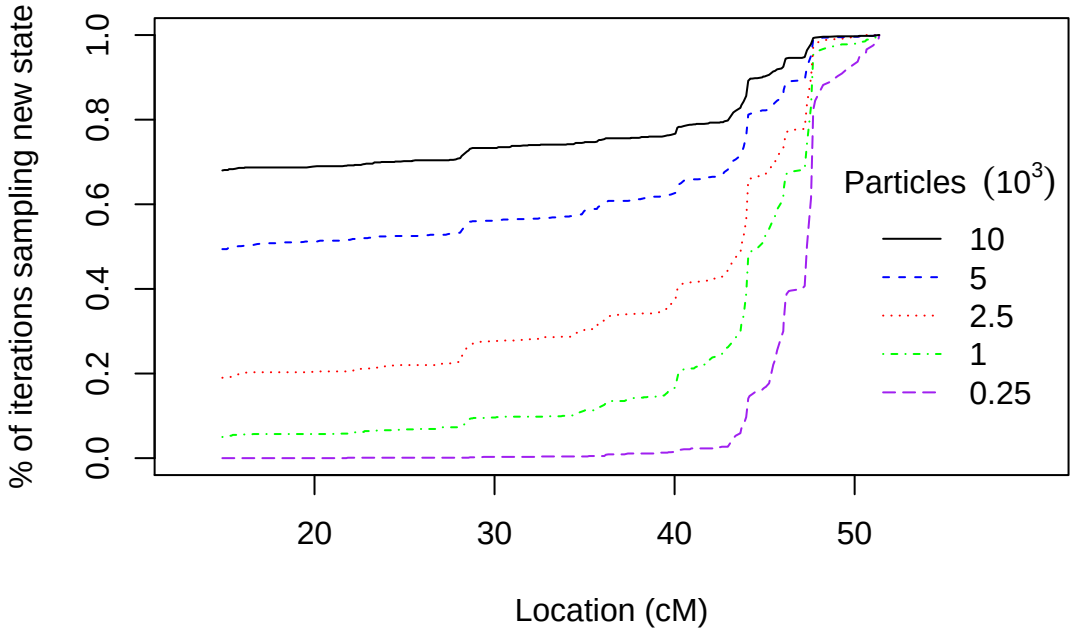


Figure 6.1: Fraction of MCMC iterations sampling a new IBD state at the given marker index. By comparison with Figure 6.2, we can observe that particle diversity drops where there are changes in the underlying IBD state.

6.2 Particle filter mixing

The analysis performed in Chapter 5 revealed that the the mixing performance of the PG sampler varies dramatically depending on the number of particles. With no *a priori* means of determining the right choice, we ran *ibd_particle* with increasing numbers of particles until a good rate of mixing was achieved. Figure 6.1 shows curves of mixing fraction at each locus for an array of choices. Using 2500 particles gives a mixing rate at the leftmost locus roughly on par with that achieved in the analysis of

the Iceland example in Chapter 5. Mixing was rapid with 10,000 particles, and those results are used in the analyses below.

6.3 Sampled IBD realizations

Figure 6.2 depicts ensembles of IBD graphs generated using the three programs. The y-axis is simply the position of the IBD state in lexicographic ordering, which begins with the state of all gametes IBD and ends with the state of no IBD. The topmost graph shows the true IBD used to simulate the sample haplotypes. Below, we can see that the three programs all capture the long segments of IBD, missing some very short segments on the left side of the chromosome. All three put some mass on an IBD segment at around 36 cM which does not appear in the truth. All three show increased uncertainty at both ends of the chromosome. The most evident different between the three is the uncertainty between about 42 and 47 cM in the *ibd_stitch* graphs. We examine this region more closely below.

Figure 6.3 provides some intuition about the poor mixing behavior when insufficient particles are used. With only 250 particles, each sample differs from the previous one mostly near the right end of the chromosome. For the first 42 markers, the trajectory established on the first run of the particle filter is never resampled. With 1000 particles, the population is large enough that some sampled trajectories do not coalesce with the previous one. However, the lack of diversity among sampled IBD states is evident near the left end of the chromosome.

6.4 Total variation distance comparison of programs

One way to quantify the comparative performance of the three programs is to examine the marginal distribution of IBD states at each locus. The marginals of the *ibd_haplo* model can be calculated using the forward-backward algorithm, so we treat these distributions as a base point for measuring distances; recall that these are not the exact marginals of the IBD model, due to the discretization in *ibd_haplo*.

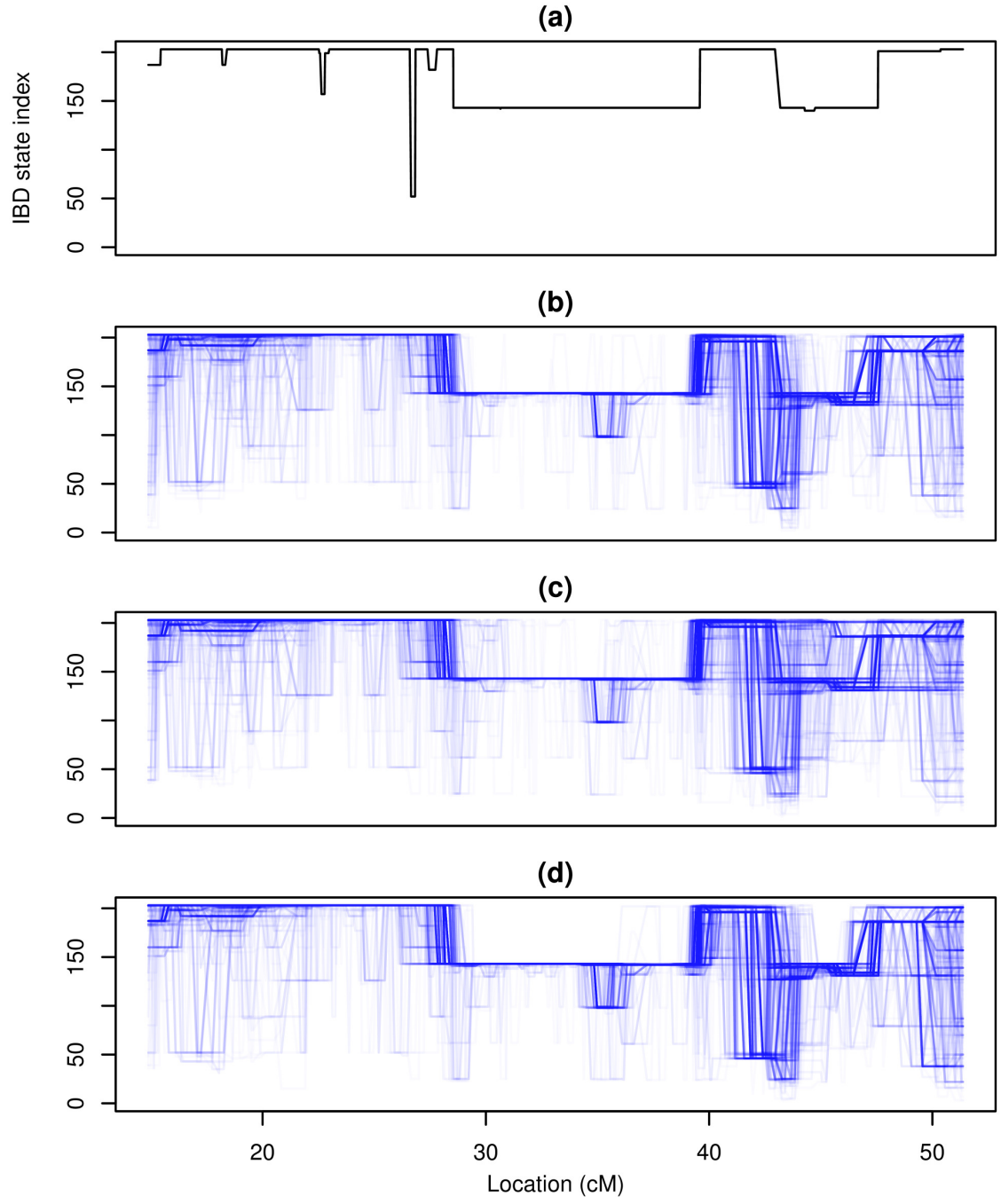


Figure 6.2: (a) Simulated true IBD, and trajectories sampled for a trio using (b) *ibd_haplo*, (c) *ibd_stitch*, and (d)*ibd_particle*. The y-axis indexes the 203 IBD states for 6 gametes in lexicographic order.

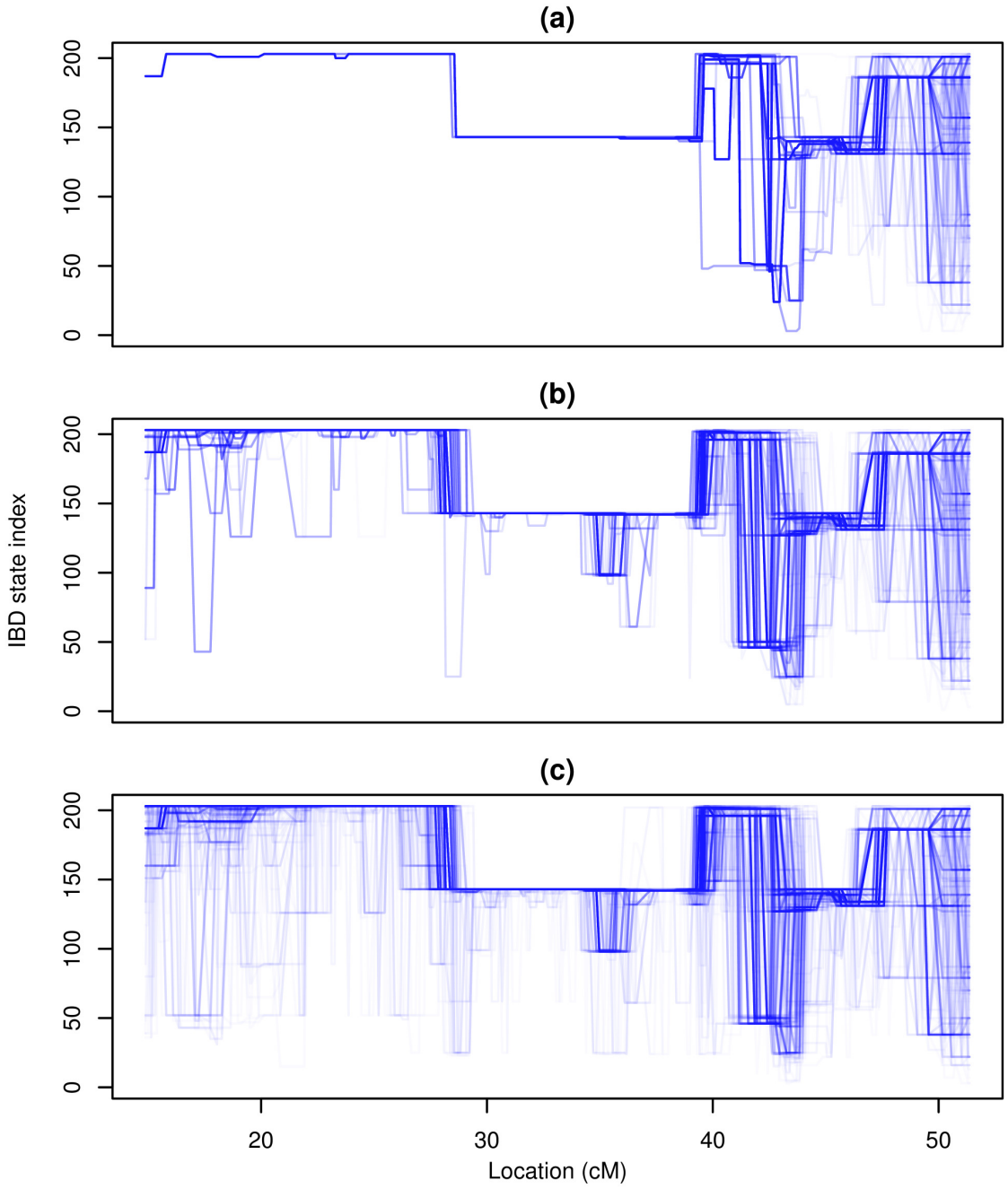


Figure 6.3: Trajectories sampled for a trio using *ibd_particle* for (a) 250 particles, (b) 1000 particles, and (c)10,000 particles. In (a), very few novel IBD states are sampled in the left half of the chromosome.

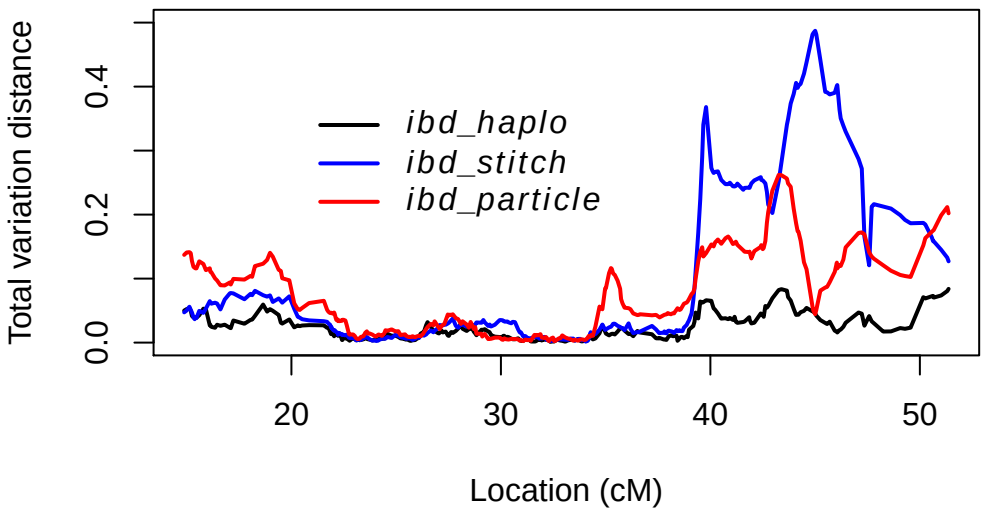


Figure 6.4: Total variation distance at each locus between the exact marginal distribution calculated using *ibd_haplo* and the distribution produced by sampling realizations using the three approaches

Given the marginal distribution $q(p)$ at a locus and the distribution $r(p)$ on IBD states at the locus induced by sampling IBD graphs, we can calculate the total variation distance (Ewens and Grant, 2005, Section 1.3) between the two as follows:

$$\text{TVD} = \frac{1}{2} \sum_{p \in \mathcal{P}_6} |q(p) - r(p)|.$$

Figure 6.4 shows the TVD between the exact marginals and the results of the three sampling methods. The black line represents samples from *ibd_haplo*, which are from the same model whose marginals are the measuring point, so it gives a sense of the scope of the Monte Carlo error involved in this comparison.

The two novel sampling methods stay close to the exact calculation on the left half of the chromosome. To the right, there is more error, as expected given the increased

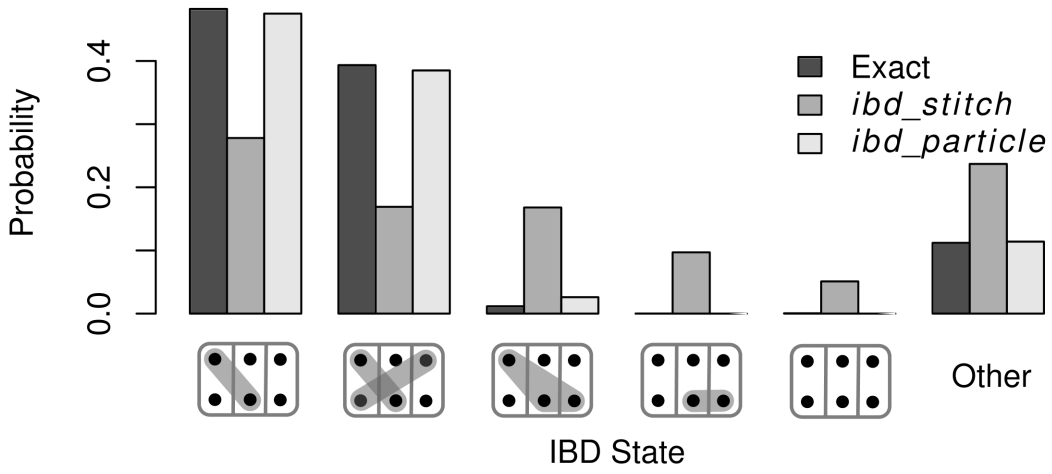


Figure 6.5: Marginal IBD state probabilities at the locus where the TVD for *ibd_stitch* spikes in Figure 6.4. The enumerated states are the five most probable under *ibd_stitch*.

uncertainty in that region evident for all three methods in Figure 6.1. Where the TVD is large, we see in general that it is smaller for the particle Gibbs method than for *ibd_stitch*.

We now briefly examine the point in Figure 6.4 where the TVD between the *ibd_stitch* samples and the marginal is greatest; this happens to be a point where *ibd_particle* is unusually close to the marginal. In Figure 6.5, we see the marginal distributions at this locus calculated exactly using *ibd_haplo* and estimated from samples from *ibd_stitch* and *ibd_particle*. The five most probable states according to *ibd_stitch* are shown, with the remainder of the probablity placed in a sixth category. All three programs place the most probability in the same two states, in which the first gamete of the first individual and the second gamete of the second are shared IBD. However, *ibd_stitch* puts significant probability in two additional states which stipulate that the second gametes (the lower ones in the figure) of the second and

third individuals are shared; one state accords with the IBD in the top two states by created a block of three gametes, and the other omits this IBD. One possible explanation is that the pairwise approach of *ibd_stitch* is sensitive to the random ordering of the pairs chosen for a given iteration of the algorithm. In a third of the iterations, the pairwise state between the second and third individuals is sampled first. IBD detected between by the pairwise model will be included in the final joint state even if it is not well-supported by the group model.

Chapter 7

DISCUSSION

7.1 Original contributions

In this work we have presented two novel methods for estimation of IBD from allele data. Building on the methods introduced by Thompson (2008), Glazner and Thompson (2012), and Brown et al. (2012), we took a model for analyzing IBD in pairs of invididuals and expanded it for use in groups of dozens of individuals. The problem posed by rapid growth in the size of the state space was overcome by breaking the model into a series of pairwise approximations of manageable size, implemented in the *ibd_stitch* program. This solution permitted two enhancements of the model's capabilities. First, the type of allele data accepted by the model was expanded to include unphased genotypes as well as phased haplotypes, because the state space of the pairwise model is small enough to integrate out the phase information at each locus. Second, sequentially sampling pairwise states requires constraining the state space to include only allowable IBD states; the algorithm can be modified to include other constraints, in particular those imposed by IBD sampled in subpedigrees of the sample. Thus, a model which in its simplest form assumes no relationships between individuals was transformed to make use of strong prior information on some components of the sample group.

The second approach to IBD estimation we have discussed is based on the same underlying model, with observed alleles related to one another via a hidden Markov process on IBD states. The distinction between the two approaches is a tradeoff of computation time and flexibility of input for model correctness. Using an algorithm called particle Gibbs sampling, we can construct a Markov chain that exactly targets

the hidden trajectory distribution of the model; our program *ibd_particle* implements this algorithm. The number of markers that can be analyzed by *ibd_particle* is in the hundreds, rather than thousands for *ibd_haplo*, but the number of individuals for whom it is possible to target the full model is increased from three or four to at least eight.

Finally, we have applied the three programs *ibd_haplo*, *ibd_stitch*, and *ibd_particle* to a small dataset in order to make a direct comparison between them. While the *ibd_particle* program took much longer to generate the same number of realizations as *ibd_stitch*, the results were closer to the output of *ibd_haplo*, which samples exactly from the (discretized) target model.

7.2 Future directions

An immediate next step for the models presented in this work is improving the performance of *ibd_particle*. We have explored but not probed the limits of particle Gibbs sampling; Chapter 5 presents results on eight individuals using 25,000 particles, so analysis of dozens of individuals in a reasonable amount of time is possible on a single computer. As the number of particles required for mixing climbs into the hundreds of thousands or millions, practical computations are only possible using parallelization over hundreds of computing cores on separate servers. A preliminary version of such an implementation has been programmed using Cloud Haskell (Epstein et al., 2011), a framework for writing distributed programs to run on separate computers in a local cluster or a computing "cloud". The development challenges and network latency of coordinating particle sampling across a cluster were found to outweigh the benefit of larger particle pools for the examples presented here. Nevertheless, very large numbers of particles should be achievable using even a naive parallelization of *ibd_particle*; more sophisticated techniques from the literature on particle filtering (e.g., Bolic et al., 2005) may expand the range of feasible datasets even more.

In addition to increasing the number of particles used by *ibd_particle*, we can

refine the PG sampling algorithm to achieve better mixing with a given number of particles. Because we cannot calculate transition probabilities between markers, we are limited in the modifications we can make to the algorithm; for instance, Lindsten et al. (2012) present a method of achieving greater diversity in sampled ancestries, but it relies on calculating quantities which are unavailable for our model. However, there may be ways to exploit the structure of the space of IBD states to create more efficient sampling methods; for example, Hajiaghayi et al. (2013) work with a user-specified potential in sampling, which could be defined in our case using distances between set partitions (Denœud and Guénoche, 2006).

As discussed by (Andrieu et al., 2010), there are numerous ways that a sequential Monte Carlo algorithm can be embedded in an MCMC algorithm. Particle Gibbs sampling can be extended to include additional Gibbs steps for other parameters or hidden data; for instance, haplotype phase could be sampled alternately with IBD so that unphased genotypes could be used as input. PG sampling is closely related to particle Metropolis-Hastings algorithms described by (Andrieu et al., 2010); relatively minor modifications to the software would permit the use of an SMC step to sample IBD graphs with a larger Metropolis-Hastings algorithm.

More generally, the possible uses of IBD modeling in genetics have yet to be fully explored. In this work we have been motivated by the use of IBD in pedigrees to calculate LOD scores, but it is not known how informative IBD is for trait detection in real datasets. Further applied research is needed to assess the role of IBD in the broader world of statistical genetics.

Appendix A

PROOFS THAT THE *IBD_STITCH* ALGORITHMS ARE WELL-DEFINED

Algorithm 1 in Chapter 2 and Algorithm 2 in Chapter 4 both sample pairwise IBD trajectories from the distribution Q, which is the HMM of Brown et al. (2012) with certain states removed from the state space to ensure that a valid joint IBD state will result once all pairs have been sampled. The algorithms assume that the reduced state space is never empty, so we must demonstrate that this is always the case.

A.1 Algorithm 1

Our proof proceeds by induction. For the first pair of individuals sampled, there are no constraints on the state space. When attaching the kth individual to the joint state, suppose we have already obtained a valid joint IBD trajectory for the first $k-1$ individuals. Elaborating on the notation of Algorithm 1, we will refer to these individuals as the set $\beta = \{B_1, \ldots, B_{k-1}\}$; let $p_{2(k-1)}(\beta)$ be the joint IBD state at a locus for these individuals. We are sampling a pairwise trajectory between individuals A_k and B_i, supposing also that we have sampled states between individual A_k and individuals B_1 through B_{i-1}. The joint IBD state composed of these pairwise states at a particular locus will be denoted $p_{2i}(\alpha)$, with $\alpha = \{A_k, B_1, \ldots, B_{i-1}\}$.

We are determining the set of valid choices for the state $p_4(B_iA_k)$. To do so, we examine each possible trio $\{A_k, B_i, B_j\}$ with $j < i$. We have sampled pairwise states for two of the three pairs in the trio, and we rule out any states which would create an invalid trio as illustrated in Figure 2.1. To show that there is at least one allowed pairwise state for $p_4(B_iA_k)$, it suffices to show that there is a valid state $p_{2k}(\gamma)$

covering the individuals $\gamma = \{A_k, B_1, \ldots, B_i\}$ which is in agreement with $p_{2(k-1)}(\beta)$ and $p_{2i}(\alpha)$. The pairwise state $p_4(B_iA_k)$ imposed by $p_{2k}(\gamma)$ cannot have been ruled out in our conditioning step.

To discuss the partitions in question more precisely, we will frame them in terms of the lattice of partitions on n individuals, $\mathcal{P}_{2n}$ (Kung et al., 2009). A lattice is a set closed under two operations, $\wedge$ and $\vee$, called meet and join and satisfying certain algebraic properties. Partitions of a set form a lattice with a partial ordering defined by refinement: $a \leq b$ if and only if a can be derived from b by splitting some blocks. Two elements are in the same block in $a \wedge b$ if and only if they are in the same block in both a and b. The partition consisting only of singletons is the least element in the lattice, and the partition of all elements into a single block is the greatest. A singular partition 1_X is the partition placing all elements of the set X in a single block and all other elements in singleton blocks. We implicitly embed a partition on a subset of the individuals in the lattice by assuming that individuals outside the subset have all gametes in singleton blocks. Conversely, we say a partition a_X of the set X induces a partition a_S of $S \subset X$ if $1_S \wedge a_X = a_S$.

Using this terminology, we can state what we wish to show as the existence of some partition $p_{2k}(\gamma)$ which induces $p_{2(k-1)}(\beta)$ on β and induces $p_{2i}(\alpha)$ on α. We have by assumption that $1_\beta \wedge p_{2i}(\alpha) = 1_\alpha \wedge p_{2(k-1)}(\beta) = p_{2(i-1)}(\delta)$; in other words, the two existing joint states agree where the overlap, on the set $\delta = \{B_1, \ldots, B_{i-1}\}$. We use a lemma from Ore (1942, Theorem 11):

Lemma A.1.1 *If a is singular and $a \geq b$, then for any c,*

$$a \wedge (b \vee c) = b \vee (a \wedge c). \tag{A.1}$$

Applying the lemma, we have $1_\beta \geq p_{2(k-1)}(\beta)$ by definition and

$$\begin{aligned} 1_\beta \wedge [p_{2(k-1)}(\beta) \vee p_{2i}(\alpha)] =& p_{2(k-1)}(\beta) \vee [1_\beta \wedge p_{2i}(\alpha)] \\ =& p_{2(k-1)}(\beta) \vee p_{2(i-1)}(\delta) \\ =& p_{2(k-1)}(\beta). \end{aligned}$$

So the partition $p_{2(k-1)}(\beta) \vee p_{2i}(\alpha)$ induces $p_{2(k-1)}(\beta)$; by the same reasoning, $p_{2(k-1)}(\beta) \vee p_{2i}(\alpha)$ induces $p_{2i}(\alpha)$. We conclude that $p_{2(k-1)}(\beta) \vee p_{2i}(\alpha)$ meets our requirements for $p_{2k}(\gamma)$, and therefore at every locus there is some pairwise state which has not been ruled out by the conditioning step.

A.2 Algorithm 2

Algorithm 2 is a version of Algorithm 1 with modifications so that the output IBD graph $\mathbf{p}_{2n}^{pop}$ incorporates an input IBD graph, $\mathbf{p}_{2n}^{ped}$, previously sampled using known pedigrees on subsets of the individuals. At every locus, we have $p_{2n}^{pop} \geq p_{2n}^{ped}$, meaning that the population IBD states are obtained by combining distinct pedigree IBD blocks. This constraint is achieved by additional restrictions on the state space of the pairwise HMM, so we must show that the algorithm with these restrictions always leaves at least one possible state at every locus.

Once again, we consider the state space at a locus for the pairwise IBD state between individuals A_k and B_i. We have a joint state $p_{2(k-1)}(\beta)$ covering the individuals in $\beta = \{B_1, \ldots, B_{k-1}\}$ and a state $p_{2i}(\alpha)$ covering $\alpha = \{A_k, B_1, \ldots, B_{i-1}\}$. We also have the pedigree IBD state $p_{2k}^{ped}(\gamma)$ (with $\gamma = \{B_1, \ldots, B_{k-1}, A_k\}$), which has been incorporated to our incomplete joint states: $1_\beta \wedge p_{2k}^{ped}(\gamma) \leq p_{2(k-1)}(\beta)$ and $1_\alpha \wedge p_{2k}^{ped}(\gamma) \leq p_{2i}(\alpha)$.

An essential difference from Algorithm 1 is that the pairwise states in $p_{2i}(\alpha)$ involving A_k have been sampled with the pedigree IBD in mind. Specifically, the following property holds: given $p_4(A_k B_j)$ with $j < i$ and $p_4(B_j B_i)$, there is at least

one state $p_4(A_kB_i)$ which forms a valid trio and satisfies $p_4(A_kB_i) \geq p_4^{ped}(A_kB_i)$.

We apply this same criterion to the state space for $p_4(A_kB_i)$. With $i < j < k$, we look ahead at $p_4^{ped}(A_kB_j)$ and rule out any states for $p_4(A_kB_i)$ which do not form a valid trio with any state statisfying $p_4(A_kB_j) \geq p_4^{ped}(A_kB_j)$. We need to show that this does not rule out all possible states for $p_4(A_kB_i)$. To do so, we construct a state $p_{2k}(\gamma)$ whose restriction to the pair $\{A_k, B_i\}$ is not ruled out by the constraints.

The state $p_{2k}(\gamma) = p_{2(k-1)}(\beta) \vee p_{2i}(\alpha) \vee p_{2k}^{ped}(\gamma)$ satisfies all our requirements. In particular, its restriction to $\{A_k, B_i\}$ cannot be ruled out by our new constraint, because its restriction to $\{A_k, B_j\}$ for any $j > i$ gives a state forming a valid trio and respecting $p_4(A_kB_j) \geq p_4^{ped}(A_kB_j)$. It remains to be shown that $p_{2k}(\gamma)$ induces both $p_{2(k-1)}(\beta)$ and $p_{2i}(\alpha)$, or in other words that it respects the IBD states which we have already sampled.

Our proof proceeds on a more granular level than in Section A.1. We will show that two gametes are in the same block of $p_{2(k-1)}(\beta)$ if and only if they are in the same block of $p_{2k}(\gamma)$. We make use of the fact that if two elements s and t are in the same block of $a \vee b$, then there is a sequence of elements beginning with s and ending with t such that any two consecutive elements are in the same block of either a or b (Ore, 1942).

Clearly, two gametes in the same block of $p_{2(k-1)}(\beta)$ are in the same block of $p_{2k}(\gamma)$. Suppose two gametes s and t of individuals in β are in the same block of $p_{2k}(\gamma)$ but not of $p_{2(k-1)}(\beta)$. We showed in Section A.1 that $p_{2(k-1)}(\beta) \vee p_{2i}(\alpha)$ preserves $p_{2(k-1)}(\beta)$, so the two gametes are in distinct blocks of $p_{2(k-1)}(\beta) \vee p_{2i}(\alpha)$. Since they are in the same block of $(p_{2(k-1)}(\beta) \vee p_{2i}(\alpha)) \vee p_{2k}^{ped}(\gamma)$, there is a sequence beginning with s and ending with t such that any two consecutive gametes are in the same block of either $p_{2(k-1)}(\beta) \vee p_{2i}(\alpha)$ or $p_{2k}^{ped}(\gamma)$.

We can assume that this sequence lies entirely within $\beta \cup \alpha$: All gametes outside this set are singletons in $p_{2(k-1)}(\beta) \vee p_{2i}(\alpha)$, so any excursion from $\beta \cup \alpha$ can only consist of consecutive pairs each in the same block of $p_{2k}^{ped}(\gamma)$, and hence can be

eliminated. Finally, we note that because $p_{2(k-1)}(\beta) \geq 1_\beta \wedge p_{2k}^{ped}(\gamma)$ and $p_{2i}(\alpha) \geq 1_\alpha \wedge p_{2k}^{ped}(\gamma)$, pairs of gametes in the sequence which are not in the same block of $p_{2(k-1)}(\beta) \vee p_{2i}(\alpha)$ (hence are in the same block of $p_{2k}^{ped}(\gamma)$) cannot both be in β or in α. As a result, any such pair in the sequence from s to t must consist of one gamete from individual A_k and one from individual B_i. Traversing the sequence starting at s, we encounter gametes in the same block of $p_{2(k-1)}(\beta) \vee p_{2i}(\alpha)$, then the first pair of gametes, u from A_k and u from B_i, which are in the same block of $p_{2k}^{ped}(\gamma)$ but not in the same block of $p_{2(k-1)}(\beta) \vee p_{2i}(\alpha)$. Say s is in the same block of $p_{2(k-1)}(\beta) \vee p_{2i}(\alpha)$ as u and hence not in the same block as v. Now considering the individual B_j containing gamete s, we note that $p_4(A_kB_j)$ was chosen such that there is some choice of $p_4(A_kB_i)$ satisfying $p_4(A_kB_i) > p_4^{ped}(A_kB_i)$ and forming a valid trio with $p_4(A_kB_j)$ and $p_4(B_jB_i)$. However, since s and v are in different blocks of $p_{2(k-1)}(\beta) \vee p_{2i}(\alpha)$ and thus are not shared IBD in of $p_4(B_iB_j)$ contains them, no valid trio can be formed by a choice of $p_4(A_kB_i)$ satisfying $p_4(A_kB_i) > p_4^{ped}(A_kB_i)$. We have reached a contradiction. Thus, if two gametes both in β are in the same block of $p_{2k}(\gamma)$, they are in the same block of $p_{2(k-1)}(\beta)$. The same reasoning holds for α and $p_{2i}(\alpha)$. We are finally able to conclude the $p_{2k}(\gamma)$ is a joint IBD state in agreement with the previously sampled states $p_{2(k-1)}(\beta)$ and $p_{2i}(\alpha)$ and whose induced pairwise state $p_4(A_kB_i)$ is guaranteed to remain in the HMM state space after the reduction of the state space.

Appendix B

SOFTWARE DEVELOPED FOR THIS WORK

The sampling algorithms discussed in this thesis have been implemented in two standalone programs, *ibd_stitch* and *ibd_particle*. Both were written in the Haskell programming language and compiled on Ubuntu Linux. The documentation for *ibd_stitch* appears below; the documentation for *ibd_particle* is largely the same, so an abbreviated version is presented.

B.1 ibd_stitch

ibd_stitch is a program for sampling identity by descent (IBD) among individuals in a population from genetic marker data. While it is not a part of the MORGAN (`http://www.stat.washington.edu/thompson/Genepi/MORGAN/Morgan.shtml`) software suite, it was developed in the same research group and as a result uses similar configuration, input, and output file formats. In many cases this documentation will refer to MORGAN formats and documentation. The software is available at `https://github.com/cglazner/ibd_stitch`.

(a) Running

The program is invoked as

```
ibd_stitch [parameter file]
```

All configuration for an analysis is specified in in the parameter file.

It is recommended to enable multicore processing with

```
ibd_stitch [parameter file] +RTS -N
```

(b) Output

A single MORGAN format IBD graph file containing realizations of IBD on the input chromosomes.

(c) Configuration

Parameters for a run of ibd_stitch are given as statements in a text file. There is no formal syntax for the various options.

- `input marker data file input.markers`
  Specifies the file of SNP data to use in the analysis. This file comes in MORGAN marker format; see that documentation at the URL above for more information.

- `output extra file ibdgraphs.out`
  The output file to be created or overwritten with the IBD graphs produced by ibd_stitch.

- `set iterations 1000`
  (Optional) The number of IBD graphs to generate if not building on pedigree IBD graphs. Either this option or `set priors` must be set.

- `set priors ["filename1","filename2","filename3"]`
  (Optional) The files containing the input pedigree IBD graphs for sampling. The files should span disjoint subsets of the sample. The total number of output IBD graphs will match the number in the shortest input pedigree file. Either this option or `set iterations` must be set.

- `set markers 5 10 15 20`
  (Required if using `set priors`) The marker indices at which to include pedigree IBD in the output graphs. Pedigree IBD is likely to have been sampled at a

lower density than the available markers, so it is enforced only at markers used in the pedigree sampling.

- `select [unphased|phased] data`
  Determines whether the input data will be modeled as unordered genotypes or phased haplotypes.

- `select population kinship 0.05`
  Sets the model's level of population kinship, or the probability of IBD between two alleles.

- `set kinship change rate 0.1`
  A scaling parameter for the change rate of the hidden Markov process.

- `set transition matrix null fraction 0.1`
  The percentage of the free transition matrix to be mixed with the the model transition matrix. This is a tuning parameter controlling the scale of the uniformization approximation used to simplify computations.

- `set genotyping error rate 0.01`
  The model probability that a given allele is observed with error, in which case it is assumed to be drawn from the population allele distribution.

- `set seed 12345`
  The seed for the pseudorandom number generator.

(d) Installation

Building requires the Haskell Platform. Building has only been tested on Ubuntu 12.04 LTS, but the following steps should be cross-platform. There are other ways to

install a cabal package, but this has worked best in the author's experience. Testing performed with GHC 7.4.1.

1. Ensure that you have the libraries `gsl` and `lapack` installed. On Ubuntu this can be done with

   ```
   apt-get install libgsl0-dev
   apt-get install liblapack-dev
   ```

2. If it is not already installed, install `cabal-dev`

   ```
   cabal install cabal-dev
   ```

3. Clone the github repository, and move into the new directory

   ```
   git clone https://github.com/cglazner/ibd_stitch
   ```

4. Build the cabal package, installing dependencies locally

   ```
   ~/.cabal/bin/cabal-dev install
   ```

5. The executable can be found at `./cabal-dev/bin/ibd_stitch`

B.2 ibd_particle

ibd_particle is a program impementing particle Gibbs sampling for IBD graphs using allelic data. Installation and invocation of the program are same as for *ibd_stitch*; see Section B.1. There is one additional output file: a table of the sampled ancestor indices for each particle Gibbs step, used to assess how well the chain mixes. The set of configuration options for *ibd_particle* is different and appears below. The software is available at `https://github.com/cglazner/ibd_particle`.

- `input marker data file input.markers`

  Specifies the file of SNP data to use in the analysis. This file comes in MORGAN marker format; see the URL provided in Section B.1 for more information.

- `select population kinship 0.05`

  Sets the model's level of population kinship, or the probability of IBD between two alleles.

- `set kinship change rate 0.1`

  A scaling parameter for the change rate of the hidden Markov process.

- `set genotyping error rate 0.01`

  The model probability that a given allele is observed with error, in which case it is assumed to be drawn from the population allele distribution.

- `set particles 1000` The number of particles used by the particle Gibbs sampler.

- `set iterations 1000` The length of the particle Gibbs Markov chain to be sampled.

- `set seed 12345`

  The seed for the pseudorandom number generator.

BIBLIOGRAPHY

Abecasis, G., Cherny, S., Cookson, W., and Cardon, L. (2001). Merlin: rapid analysis of dense genetic maps using sparse gene flow trees. *Nature Genetics*, 30(1):97–101.

Aldous, D. (1985). Exchangeability and related topics. *École d'Été de Probabilités de Saint-Flour XIII—1983*, pages 1–198.

Andrieu, C., Doucet, A., and Holenstein, R. (2010). Particle Markov chain Monte Carlo methods. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 72(3):269–342.

Berend, D. and Tassa, T. (2010). Improved bounds on bell numbers and on moments of sums of random variables. *Probability and Mathematical Statistics*, 30(2):185–205.

Bolic, M., Djuric, P. M., and Hong, S. (2005). Resampling algorithms and architectures for distributed particle filters. *Signal Processing, IEEE Transactions on*, 53(7):2442–2450.

Brown, M. D., Glazner, C. G., Zheng, C., and Thompson, E. a. (2012). Inferring coancestry in population samples in the presence of linkage disequilibrium. *Genetics*, 190(April):1447–1460.

Browning, B. L. and Browning, S. R. (2007a). Efficient multilocus association testing for whole genome association studies using localized haplotype clustering. *Genetic epidemiology*, 31(5):365–375.

Browning, B. L. and Browning, S. R. (2009). A unified approach to genotype im-

putation and haplotype-phase inference for large data sets of trios and unrelated individuals. *The American Journal of Human Genetics*, 84(2):210–223.

Browning, S. R. (2006). Multilocus association mapping using variable-length Markov chains. *The American Journal of Human Genetics*, 78(6):903–13.

Browning, S. R. and Browning, B. L. (2007b). Rapid and accurate haplotype phasing and missing-data inference for whole-genome association studies by use of localized haplotype clustering. *American Journal of Human Genetics*, 81(5):1084.

Browning, S. R. and Browning, B. L. (2010). High-resolution detection of identity by descent in unrelated individuals. *The American Journal of Human Genetics*, 86(4):526–539.

Burton, P. R., Clayton, D. G., Cardon, L. R., Craddock, N., Deloukas, P., Duncanson, A., Kwiatkowski, D. P., McCarthy, M. I., Ouwehand, W. H., Samani, N. J., et al. (2007). Genome-wide association study of 14,000 cases of seven common diseases and 3,000 shared controls. *Nature*, 447(7145):661–678.

Clark, C. E. (1961). Importance sampling in Monte Carlo analyses. *Operations Research*, 9(5):603–620.

Cleveland, M. A., Hickey, J. M., and Forni, S. (2012). A common dataset for genomic analysis of livestock populations. *G3: Genes— Genomes— Genetics*, 2(4):429–435.

Crow, J. F. and Kimura, M. (1970). *An introduction to population genetics theory.* New York, Evanston and London: Harper & Row, Publishers.

Cupples, L., Heard-Costa, N., and Lee, M. (2009). Genetics analysis workshop 16 problem 2: the framingham heart study data. *BMC*, 3(Suppl 7):S3.

Denœud, L. and Guénoche, A. (2006). Comparison of distance indices between partitions. In Batagelj, V., Bock, H.-H., Ferligoj, A., and iberna, A., editors, *Data*

Science and Classification, Studies in Classification, Data Analysis, and Knowledge Organization, pages 21–28. Springer Berlin Heidelberg.

Donnelly, K. (1983). The probability that related individuals share some section of genome identical by descent. *Theoretical Population Biology*, 23:34–63.

Doucet, A. (2001). *Sequential Monte Carlo Methods*. Wiley Online Library.

Drummond, A. J., Suchard, M. A., Xie, D., and Rambaut, A. (2012). Bayesian phylogenetics with BEAUti and the BEAST 1.7. *Molecular Biology and Evolution*, 29(8):1969–1973.

Epstein, J., Black, A. P., and Peyton-Jones, S. (2011). Towards Haskell in the cloud. In *ACM SIGPLAN Notices*, volume 46, pages 118–129. ACM.

Ewens, W. (1972). The sampling theory of selectively neutral alleles. *Theoretical Population Biology*, 3:87–112.

Ewens, W. J. and Grant, G. R. (2005). *Statistical Methods in Bioinformatics: an Introduction*, volume 746867830. Springer.

Fan, H. C., Wang, J., Potanina, A., and Quake, S. R. (2011). Whole-genome molecular haplotyping of single cells. *Nature Biotechnology*, 29(1):51–57.

Glazner, C. G. and Thompson, E. A. (2012). Improving pedigree-based linkage analysis by estimating coancestry among families. *Statistical Applications in Genetics and Molecular Biology*, 11(2): 11.

Griffiths, R. and Marjoram, P. (1996). Ancestral inference from samples of DNA sequences with recombination. *Journal of Computational Biology*, 3(4):479–502–.

Gusev, A., Lowe, J. K., Stoffel, M., Daly, M. J., Altshuler, D., Breslow, J. L., Friedman, J. M., and Pe'er, I. (2009). Whole population, genome-wide mapping of hidden relatedness. *Genome Research*, 19(2):318–326.

Hajiaghayi, M., Kirkpatrick, B., Wang, L., and Bouchard-Côté, A. (2013). Efficient continuous-time Markov chain estimation. *arXiv preprint arXiv:1309.3250*.

Harris, D. (1964). Genotypic covariances between inbred relatives. *Genetics*, 50(6):1319.

Howie, B., Fuchsberger, C., Stephens, M., Marchini, J., and Abecasis, G. R. (2012). Fast and accurate genotype imputation in genome-wide association studies through pre-phasing. *Nat Genet*, 44(8):955–959.

Hudson, R. (1990). Gene genealogies and the coalescent process. *Oxford Surveys in Evolutionary Biology*, 7(1):44.

Kingman, J. F. C. (1982). On the genealogy of large populations. *Journal of Applied Probability*, 19:27–43.

Kong, A., Masson, G., Frigge, M. L., Gylfason, A., Zusmanovich, P., Thorleifsson, G., Olason, P. I., Ingason, A., Steinberg, S., Rafnar, T., Sulem, P., Mouy, M., Jonsson, F., Thorsteinsdottir, U., Gudbjartsson, D. F., Stefansson, H., and Stefansson, K. (2008). Detection of sharing by descent, long-range phasing and haplotype imputation. *Nature Genetics*, 40(9):1068 – 1075.

Kraja, A. T., Culverhouse, R., Daw, E. W., Wu, J., Van Brunt, A., Province, M. A., and Borecki, I. B. (2009). The genetic analysis workshop 16 problem 3: simulation of heritable longitudinal cardiovascular phenotypes based on actual genome-wide single-nucleotide polymorphisms in the Framingham Heart Study. In *BMC proceedings*, volume 3, page S4. BioMed Central Ltd.

Kuhner, M. K. and Smith, L. P. (2007). Comparing likelihood and Bayesian coalescent estimation of population parameters. *Genetics*, 175(1):155–165.

Kung, J. P., Rota, G.-C., and Yan, C. H. (2009). *Combinatorics: the Rota way*. Cambridge University Press.

Leutenegger, A., Prum, B., and Verny, C. (2003). Estimation of the inbreeding coefficient through use of genomic data. *The American Journal of Human Genetics*, 73:516–523.

Li, N. and Stephens, M. (2003). Modeling linkage disequilibrium and identifying recombination hotspots using single-nucleotide polymorphism data. *Genetics*, 165(4):2213–2233.

Li, Y., Willer, C. J., Ding, J., Scheet, P., and Abecasis, G. R. (2010). MaCH: using sequence and genotype data to estimate haplotypes and unobserved genotypes. *Genetic Epidemiology*, 34(8):816–834.

Lin, S., Chakravarti, A., and Cutler, D. J. (2004). Haplotype and missing data inference in nuclear families. *Genome Research*, 14(8):1624–1632.

Lindsten, F., Jordan, M. I., and Schön, T. B. (2012). Ancestor sampling for particle Gibbs. In *NIPS*, pages 2600–2608.

McVean, G. A. and Cardin, N. J. (2005). Approximating the coalescent with recombination. *Philosophical Transactions of the Royal Society B: Biological Sciences*, 360(1459):1387–1393.

Morton, N. (1955). Sequential tests for the detection of linkage. *American Journal of Human Genetics*, 7(3):277.

Ore, O. (1942). Theory of equivalence relations. *Duke Mathematical Journal*, pages 573–627.

Rabiner, L. (1989). A tutorial on hidden Markov models and selected applications in speech recognition. *Proceedings of the IEEE*, 77(2):257–286.

Robert, C. P. and Casella, G. (1999). *Monte Carlo Statistical Methods*. Springer.

Sobel, E. and Lange, K. (1996). Descent graphs in pedigree analysis: applications to haplotyping, location scores, and marker-sharing statistics. *American Journal of Human Genetics*, 58:1323–1337.

Thompson, E. A. (1974). Gene identities and multiple relationships. *Biometrics*, 30:667–680.

Thompson, E. A. (2000). *Statistical Inferences from Genetic Data on Pedigrees*, volume 6 of *NSF-CBMS Regional Conference Series in Probability and Statistics*. Institute of Mathematical Statistics, Beachwood, OH.

Thompson, E. A. (2008). The IBD process along four chromosomes. *Theoretical Population Biology*, 73(3):369–73.

Thompson, E. A. (2011). The structure of genetic linkage data: from LIPED to 1M SNPs. *Human Heredity*, 71:86–96.

Thompson, E. A. (2013). Identity by descent: Variation in meiosis, across genomes, and in populations. *Genetics*, 194(2):301–326.

Tong, L. and Thompson, E. A. (2008). Multilocus lod scores in large pedigrees: Combination of exact and approximate calculations. *Human Heredity*, 65:142–153.

VITA

Christopher Glazner was born in North Carolina and earned a B.A. in mathematics and economics at the University of North Carolina at Chapel Hill in 2007. He worked for two years in San Francisco as a research analyst at the Brattle Group and began graduate school at the University of Washington in 2009.