

Estimating genotype probabilities in complex pedigrees

Soledad A. Fernández¹
Rohan L. Fernando²
Alicia L. Carriquiry³
Bernt Gulbrandtsen⁴

1 Introduction

Determining genotype probabilities is important in genetic counseling, linkage analysis and in genetic evaluation programs. In genetic counseling, for example, in the case of recessive disease traits it is important to know which individuals in a population are probable carriers of a bad allele. The first methods for determining genotype probabilities were developed in human genetics by Elston and Stewart (1971) and were reviewed by Elston and Rao (1978) and by Elston (1987). Also, several human genetics computer packages are available to compute genotype probabilities. In livestock, pedigrees are usually much larger because animals, specially males, have multiple mates. Thus, the application of computer intensive methods developed for humans will often be difficult or inappropriate in livestock data.

In this paper, we consider the problem of estimating genotype probabilities in a dog pedigree. The pedigree consists of 3,052 dogs (Labrador Retrievers) from “The Seeing Eye, Inc”. The trait of interest is a disease called progressive retinal atrophy (PRA). PRA is a genetic disorder of the eye and it is inherited as a simple autosomal recessive. Thus, the dog is affected when it has the homozygous recessive genotype for the disease locus. If the dog has the heterozygous genotype (Aa or aA) then it is a carrier.

¹Doctoral candidate, Depts. of Statistics and Animal Science, Iowa State University, Ames, IA 50011. She is corresponding author, soledadiastate.edu

²Professor, Dept. of Animal Science and Lawrence Baker Center for Bioinformatics and Biological Statistics, Iowa State University.

³Associate Professor, Dept. of Statistics and Lawrence Baker Center for Bioinformatics and Biological Statistics, Iowa State University.

⁴Research Scientist, Dept. of Breeding and Genetics, Danish Institute of Animal Science, Denmark

This disease can only be diagnosed by either an ophthalmic exam after the dog is 5 years old or by a very expensive electro-retinal-gram (ERG) after the dog is 18 months of age. Therefore, it is important to know which dogs are at highest risk of transmitting the PRA gene to their offspring and which dogs are at lower risk both of transmitting the gene and of being PRA affected. There are 33 affected dogs in this pedigree. That is, we only know the genotype and phenotype for 33 dogs. The rest of the dogs could be aa but yet not detected, could be Aa or could be AA .

To obtain estimates of the genotype probabilities, the likelihood of the pedigree is needed. The likelihood $L(\mathbf{g}; \mathbf{y})$ is obtained as

$$L(\mathbf{g}; \mathbf{y}) \propto \sum_{\mathbf{g}} f(\mathbf{y}|\mathbf{g})P(\mathbf{g}), \quad (1.1)$$

where \mathbf{y} is the vector of phenotypes and \mathbf{g} is the vector of genotypes, $f(\mathbf{y}|\mathbf{g})$ are the conditional probabilities of \mathbf{y} given \mathbf{g} , $P(\mathbf{g})$ are the genotype probabilities. The summation is over all possible genotypes for all the individuals in the pedigree. The computations involved in the likelihood are not feasible except in trivial examples. For example, assume that there are two possible alleles for a locus, resulting in 3 possible genotypes for every individual in the pedigree (AA , Aa or aA and aa). If the pedigree consists of 100 individuals then 3^{100} summations need to be performed in order to compute the likelihood as in (1.1).

For pedigrees without loops, genotype probabilities can be calculated by Elston-Stewart algorithm (Elston and Stewart, 1971), which is also called “peeling”. Fernando et al. (1994) presented an efficient algorithm to calculate genotype probabilities of all members in an animal pedigree without loops. This algorithm is feasible for pedigrees with few and simple loops but would become impractical as the loops increase in number and complexity. For small pedigrees (about 100 members) with loops, extensions of the Elston-Stewart algorithm have been developed for evaluating the likelihood (Lange and Elston, 1975; Cannings et al., 1978; Lange and Boehnke, 1983; Thomas, 1986a,b). For large pedigrees with complex loops exact calculations are not possible and approximations are used (Van Arendonk et al., 1989; Janss et al., 1992; Stricker et al., 1995; Wang et al., 1996).

Van Arendonk et al. (1989) presented an iterative algorithm to calculate genotype probabilities of all members in an animal pedigree. Some limitations in their algorithm were removed by Janss et al. (1992). Their method can be used to approximate the likelihood for large and complex pedigrees with loops. Stricker et al. (1995) also proposed a method to approximate the likelihood in pedigrees with loops. Their method is based on an algorithm that cuts the loops. This method gives the exact likelihood when the pedigree does not have loops. In 1996, Wang et al. proposed a new approximation to the likelihood of a pedigree with loops by cutting all loops and extending the pedigree at the cuts. This method makes use of iterative

peeling. They showed that the likelihood computed by iterative peeling is equivalent to the likelihood computed from a cut and extended pedigree.

Also, Markov chain Monte Carlo (MCMC) methods have been proposed to estimate genotype probabilities. These MCMC methods give exact estimates to any desired level of accuracy. As Jensen and Sheehan (1998) observed, the genotypes in a pedigree can be sampled according to a Markovian process because a neighborhood system can be defined on a pedigree such that the genotype of an individual, conditional on the neighbors or close relatives, is independent of the remaining pedigree members. This local dependency makes MCMC methods, such as the Gibbs sampler, very easy to implement and provides a strategy to estimate posterior genotype probabilities that are difficult to calculate otherwise.

When using the Gibbs sampler, however, mixing can be very slow due to the dependence between genotypes of parents and progeny (Janss et al., 1995). The larger the progeny groups, the stronger the dependence and thus the Gibbs chains do not move. When this happens it is said that the chains are reducible “in practice”. Mixing can be improved by applying Gibbs sampling to blocks (Jensen et al., 1995; Janss et al., 1995). This procedure is called “Blocking Gibbs Sampling” and consists of sampling a block of genotypes jointly. In this approach, the blocks are typically formed by sub-families in the pedigree. Blocking Gibbs can be applied to very large pedigrees provided there are no complex loops. The efficiency of blocking depends on the pedigree structure and the way that those blocks are built.

In this paper we propose a method to sample genotypes from large and complex pedigrees, and apply the proposed algorithm to estimate the genotype probabilities in the Labrador Retriever pedigree under study. We use the Metropolis-Hastings algorithm and sample genotypes (candidates) jointly from a proposal distribution that is close to the true posterior distribution of interest. When there are no loops, our proposal is the “true” posterior distribution and the Metropolis-Hastings algorithm reduces to direct sampling. When there are loops in the pedigree, we construct our proposal distribution by first sampling genotypes using iterative peeling. A variation of this approach consists in peeling the pedigree exactly up to the point where the complexity of loops makes it difficult to continue, and then using Metropolis-Hastings to sample from a “partially” peeled pedigree.

The paper is organized as follows. In section 2, we review the method of peeling and show how it is used to sample genotypes. The approach we propose for sampling genotypes is discussed in section 3.

In section 4, we return to the case study, and apply our methods to the Labrador Retriever pedigree provided by “The Seeing Eye, Inc”. We present the results of the analysis and briefly discuss the problem of assessing the performance of our method.

Finally, a brief conclusion and summary remarks are given in section 5.

2 The peeling approach for sampling genotypes

Before describing iterative peeling we discuss the Elston-Stewart algorithm known as *peeling*.

Consider the simple pedigree shown in Figure 2.1. To introduce the principles involved in peeling we show how to sample genotypes from $f(\mathbf{g}|\mathbf{y})$ for a monogenic trait in pedigrees without loops.

To obtain a random sample from $f(\mathbf{g}|\mathbf{y})$, we can use a rejection sampler based on $f(\mathbf{g}|\mathbf{y})$, but this may be very inefficient. Instead, we sample indi-

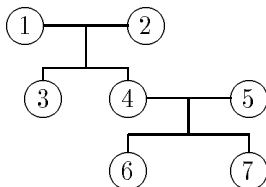


FIGURE 2.1. Simple two-generational pedigree.

viduals sequentially by conditioning on the individuals that were already sampled. Thus, to obtain a sample from $f(g_1, g_2, g_3, g_4, g_5, g_6, g_7|\mathbf{y})$ in Figure 2.1, we first sample the genotype for individual 1 from $f(g_1|\mathbf{y})$. Next we sample g_2 from $f(g_2|g_1, \mathbf{y})$, g_3 from $f(g_3|g_1, g_2, \mathbf{y})$, and so on. To compute $f(g_1|\mathbf{y})$ we use **peeling** (Elston and Stewart, 1971; Cannings et al., 1978).

The likelihood for the pedigree in Figure 2.1 can be written as

$$L(\mathbf{g}; \mathbf{y}) \propto \sum_{g_1} \sum_{g_2} \cdots \sum_{g_7} h(g_1)h(g_2)h(g_1, g_2, g_3)h(g_1, g_2, g_4)h(g_5) \times h(g_4, g_5, g_6)h(g_4, g_5, g_7), \quad (2.2)$$

where

$$h(g_j) = P(g_j)f(y_j|g_j) \quad (2.3)$$

if j is a founder, that is if the parents of j are not in the pedigree. The penetrance function, $f(y_j|g_j)$, represents the probability that an individual with genotype g_j has phenotype y_j . The founder probability denoted $P(g_j)$, represents the prior probability that an individual has genotype g_j . If individual j is not a founder then

$$h(g_m, g_f, g_j) = P(g_j|g_m, g_f)f(y_j|g_j), \quad (2.4)$$

where g_m and g_f are the genotypes for the mother and father of individual j . The transmission probability $P(g_j|g_m, g_f)$, is the probability that an individual has genotype g_j given parental genotypes g_m and g_f .

Suppose each g_j can take one of three values (AA , Aa and aa). Then $L(\mathbf{g}; \mathbf{y})$ as given by (2.2) is the sum of 3^7 terms. Thus, the total number

of computations is 2,187 even in this simple pedigree computations are overwhelming. Therefore, the number of computations is exponential in the number of genotypes in the expression.

Computing the likelihood as given by (2.2) is feasible only for small pedigrees. The Elston-Stewart algorithm (Elston and Stewart, 1971), however, provides an efficient method to compute (2.2) for simple pedigrees, and generalizations of this algorithm (Cannings et al., 1978; Fernando et al., 1993; Lange and Elston, 1975; Lange and Boehnke, 1983) provide strategies to compute the likelihood efficiently for general pedigrees with simple loops.

The Elston-Stewart algorithm can be thought of as providing an efficient reordering of the additions and multiplications in computing the likelihood. Thus, $L(\mathbf{g}; \mathbf{y})$ in (2.2) is rearranged as

$$\sum_{g_1} \sum_{g_2} \left[h(g_1) h(g_2) \sum_{g_3} h(g_1, g_2, g_3) \sum_{g_4} \left\{ h(g_1, g_2, g_4) \times \sum_{g_5} \left[h(g_5) \sum_{g_6} h(g_4, g_5, g_6) \sum_{g_7} h(g_4, g_5, g_7) \right] \right\} \right]. \quad (2.5)$$

Note that (2.5) is identical in value to (2.2) but is computationally more efficient. For example, consider the summation over g_7 . In (2.2) this summation is done over all combinations of values of g_1, g_2, g_3, g_4, g_5 and g_6 . However, the only function involving g_7 , is $h(g_4, g_5, g_7)$, which depends only on two other individual genotypes (g_4 and g_5). In (2.5), the summation over g_7 is done only for all combinations of values of g_4 and g_5 . An expression involving g_5 and g_4 is obtained after summing out g_7 . After summing out g_7 , we sum out g_6 and so on.

Computing $L(\mathbf{g}; \mathbf{y})$ sequentially as described above is referred to as *peeling*. In the first step, g_7 was *peeled* and a simpler expression was obtained which did not involve g_7 . Similarly, after peeling g_6 , $L(\mathbf{g}; \mathbf{y})$ became free of g_6 . The result, from peeling an individual is called a *cutset*. For example, after peeling g_7 we obtain a cutset of size 2,

$$c_7(g_4, g_5) = \sum_{g_7} h(g_4, g_5, g_7).$$

To compute $L(\mathbf{g}; \mathbf{y})$ efficiently, the order of peeling is critical. For example, consider peeling g_1 as the first step, so the likelihood can be written as

$$L(\mathbf{g}; \mathbf{y}) \propto \sum_{g_2} \sum_{g_3} \cdots \sum_{g_7} h(g_2) h(g_5) h(g_4, g_5, g_6) h(g_4, g_5, g_7) c_1(g_2, g_3, g_4),$$

where

$$c_1(g_2, g_3, g_4) = \sum_{g_1} h(g_1) h(g_1, g_2, g_3) h(g_1, g_2, g_4).$$

The result, $c_1(g_2, g_3, g_4)$, from peeling g_1 is a cutset of size 3, and its computation involves summing over g_1 for all genotype combinations of g_2, g_3 and g_4 . Computing $c_7(g_4, g_5)$ has lower storage and computational requirements than computing $c_1(g_2, g_3, g_4)$. The storage and computational requirements would be similar for peeling g_3 and g_6 in the first step. Peeling g_4 in the first step would be even more costly than peeling g_1, g_2 or g_5 .

Thus, to evaluate the likelihood for this pedigree we first need to define the peeling order. The peeling order is determined by the following algorithm.

1. List all the individuals in the pedigree that need to be peeled.
2. For each individual determine the size of the resulting cutset after peeling the individual.
3. Peel the individual with the smallest cutset.
4. Repeat steps 2 and 3 until all the individuals are peeled.

In this case, the peeling order can be: 3, 1, 2, 7, 6, 4 and 5. Once all individuals have been peeled we sample individual's genotypes in the reverse order to which they were peeled (reverse peeling, Heath (1998)). Then, after peeling individual 5 we compute the marginal probability for 5 as

$$f(g_5|\mathbf{y}) = \frac{P(g_5)f(y_5|g_5)c_5(g_5)}{L(\mathbf{g}; \mathbf{y})}.$$

Once $f(g_5|\mathbf{y})$ has been obtained, we sample g_5 using the inverse cumulative function. Next, we compute

$$f(g_4|g_5, \mathbf{y}) = \frac{f(y_4|g_4)c_2(g_4)c_7(g_4, g_5)c_6(g_4, g_5)}{\sum_{g_4} f(y_4|g_4)c_2(g_4)c_7(g_4, g_5)c_6(g_4, g_5)},$$

and then we sample g_4 from $f(g_4|g_5, \mathbf{y})$. Repeatedly applying this procedure, we eventually generate a sample from the joint distribution of all genotypes for the entire pedigree. The sampling sequence in this case is

- sample g_5 from $f(g_5|\mathbf{y})$,
- sample g_4 from $f(g_4|\mathbf{y}, g_5)$,
- sample g_6 from $f(g_6|\mathbf{y}, g_5, g_4)$,
- sample g_7 from $f(g_7|\mathbf{y}, g_5, g_4, g_6)$,
- sample g_2 from $f(g_2|\mathbf{y}, g_5, g_4, g_6, g_7)$,
- sample g_1 from $f(g_1|\mathbf{y}, g_5, g_4, g_6, g_7, g_2)$,
- sample g_3 from $f(g_3|\mathbf{y}, g_5, g_4, g_6, g_7, g_2, g_3)$.

In pedigrees with complex loops, peeling methods as described above are not feasible. The reason is that the cutsets generated after peeling some individuals are larger when there are loops in the pedigree.

3 Iterative Peeling and Metropolis-Hastings algorithm to sample genotypes

Peeling methods cannot be applied when pedigrees are large and have complex loops. Iterative peeling (Van Arendonk et al., 1989; Janss et al., 1992; Wang et al., 1996), however can be used to get approximate results. To describe iterative peeling, it is convenient to present the pedigree as a directed graph (Figure 3.2 (a)).

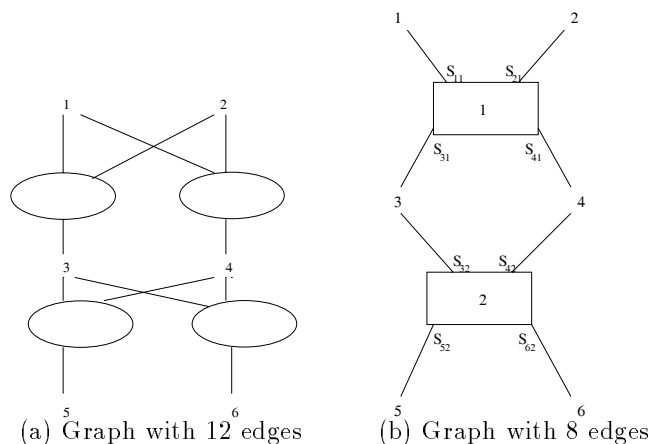


FIGURE 3.2. Graph representation of a two-generational pedigree with loops.

Before peeling, the graph contains individual nodes and mating nodes. Each individual node is indicated by the individual identification number; they correspond to the penetrance functions, and in the case of founders, also include the founder probability function. Each mating node is indicated by an oval, which corresponds to the transmission probability function. The edges in the graph connect the mating nodes with the parents and with the offspring.

Before proceeding with iterative peeling we modify the graph by merging mating nodes into nuclear family nodes. The resulting graph with the merged mating nodes is shown in Figure 3.2 (b). Here, the nuclear-family nodes are represented by rectangles. There are 8 edges: S_{11} , S_{21} , S_{31} , S_{32} , S_{41} , S_{42} , S_{52} , S_{62} in this graph. The first subindex of S indicates the individual number, and the second subindex indicates the nuclear-family node number; for example S_{31} is the edge that connects individual 3 with nuclear-family node 1. The edges S_{ij} can be interpreted as either conditional or joint probabilities, depending on their location in the graph. We use this small example to explain iterative peeling and present the general expressions for the algorithm later.

Suppose we want to sample the genotype for individual 1 from $f(g_1|\mathbf{y})$.

We first obtain an estimate for the edge probability S_{11} , connecting individual 1 to the rest of the pedigree through nuclear family 1. Once S_{11} is computed, the genotype probabilities can be obtained from the normalized values of $f(y_1|g_1)P(g_1)S_{11}$. Below we describe how to iteratively compute S_{11} .

We first initialize all the edge probabilities. In general, all edge probabilities are initialized to 1. For this example, however it is convenient to set S_{41} to be equal to the founder genotype probabilities. Once the edges are initialized we iteratively update edge probabilities using the phenotypes and the current values of the appropriate edges (explained below) of all the individuals in the corresponding nuclear family. Thus, we update S_{11} as

$$S_{11} = \sum_{g_2} \sum_{g_3} \sum_{g_4} f(y_2|g_2)P(g_2)f(y_3|g_3)P(g_3|g_1, g_2) \times \\ f(y_4|g_4)P(g_4|g_1, g_2)S_{32}S_{42}.$$

At this stage, S_{11} is the conditional probability $f(y_2, y_3, y_4|g_1)$. Note that the edges that contributed to updating S_{11} are those that connect the members of nuclear family 1 to other nuclear families.

Similarly S_{21} is updated as

$$S_{21} = \sum_{g_1} \sum_{g_3} \sum_{g_4} f(y_1|g_1)P(g_1)f(y_3|g_3)P(g_3|g_1, g_2) \times \\ f(y_4|g_4)P(g_4|g_1, g_2)S_{32}S_{42},$$

and is the conditional probability $f(y_1, y_3, y_4|g_2)$. Next, we update S_{31} as

$$S_{31} = \sum_{g_1} \sum_{g_2} \sum_{g_4} f(y_1|g_1)P(g_1)f(y_2|g_2)P(g_2)P(g_3|g_1, g_2) \times \\ f(y_4|g_4)P(g_4|g_1, g_2)S_{42},$$

which is the joint probability $f(y_1, y_2, y_4, g_3)$. Next, we update S_{32} as,

$$S_{32} = \sum_{g_4} \sum_{g_5} \sum_{g_6} f(y_5|g_5)P(g_5|g_3, g_4)f(y_6|g_6)P(g_6|g_3, g_4) \times \\ f(y_4|g_4) \underbrace{S_{41}}_{P(g_4)},$$

which is the conditional probability $f(y_4, y_5, y_6|g_3)$. Note that in these three cases, when we multiplied by an edge probability we used the initial values.

Next, we update S_{41} as

$$S_{41} = \sum_{g_1} \sum_{g_2} \sum_{g_3} f(y_1|g_1)P(g_1)f(y_2|g_2)P(g_2)f(y_3|g_3) \times \\ P(g_3|g_1, g_2)P(g_4|g_1, g_2) \underbrace{S_{32}}_{f(y_4, y_5, y_6|g_3)}.$$

In this case, the edge probability S_{32} was already updated once. Thus, the value of $S_{41} = f(y_1, y_2, y_3, y_4, y_5, y_6, g_4)$ is the joint probability of the genotype of individual 4 and of all the phenotypic values connected to 4 through nuclear family 1 in the cut-extended pedigree shown in Figure 3.3 (a). Next, we update S_{42} as

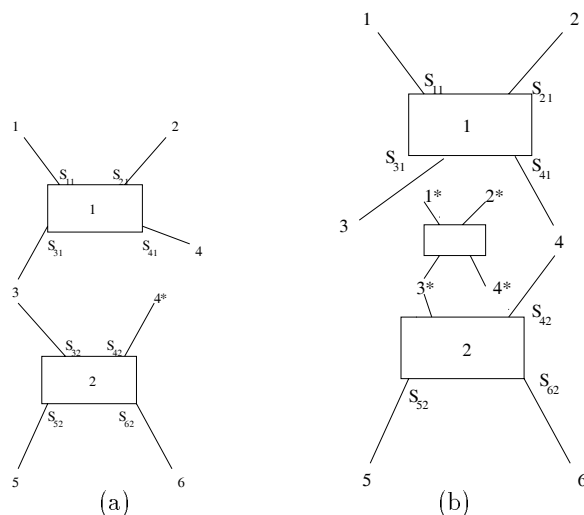


FIGURE 3.3. Cut and extended graphs

$$S_{42} = \sum_{g_1} \sum_{g_2} \sum_{g_5} \sum_{g_6} f(y_5|g_5)P(g_5|g_3, g_4)f(y_6|g_6)P(g_6|g_3, g_4)f(y_3|g_3) \times \\ P(g_3|g_1, g_2) \underbrace{S_{31}}_{f(y_1, y_2, y_4, g_3)}.$$

Again, in this case we use an edge probability that was already updated, and thus $S_{42} = f(y_1, y_2, y_3, y_4, y_5, y_6|g_3)$, which is the conditional probability of all the phenotypic values connected to 4 through nuclear family 2 in the cut-extended pedigree shown in Figure 3.3 (b), given the genotype of individual 4.

Each subsequent iteration results in further extensions to a cut pedigree. After a sufficient number of iterations we sample genotypes as follows from the iteratively peeled pedigree. First we sample the genotype of individual 1 from $f(g_1|\mathbf{y})$, which is computed using S_{11} as described above. Next, to sample the genotype of 2 we update S_{21} to reflect the sampled value for the genotype of 1 as

$$S_{21} = \sum_{g_3} \sum_{g_4} f(y_1|g_1)P(g_1)f(y_3|g_3)P(g_3|g_1, g_2) \times \\ f(y_4|g_4)P(g_4|g_1, g_2)S_{32}S_{42},$$

where g_1 is the sampled value for the genotype of 1. Using this updated value for S_{21} , $f(g_2|\mathbf{y}, g_1)$ is computed as

$$f(g_2|\mathbf{y}, g_1) = \frac{f(y_2|g_2)P(g_2)S_{21}}{\sum_{g_2} f(y_2|g_2)P(g_2)S_{21}}.$$

This process is continued until all individuals are sampled. We propose to use these sampled genotypes as the proposal distribution in Metropolis-Hastings algorithm to accept or reject the candidate draws.

We now provide the general expressions for updating edge probabilities in iterative peeling. Let S_{js} be an edge between individual j and nuclear-family node s . If j is a parent in the nuclear family, S_{js} is computed iteratively as,

$$S_{js} = \sum_{g_p} R_{sp} \prod_{k \in C_s} [\sum_{g_k} \Pr(g_k|g_j, g_p) R_{sk}], \quad (3.6)$$

where p is the other parent in the nuclear family, C_s is the set of children in nuclear family s ,

$$R_{sl} = \prod_{\substack{e \in E_l \\ e \neq s}} S_{le} f(y_l|g_l) P(g_l) \quad (3.7)$$

for $l = k, p$, E_l is the set of edges for individual l , $f(y_l|g_l)$ is the penetrance function and $P(g_l)$ is the founder probability, if individual l is not a founder then $P(g_l) = 1$. If j is a child in the nuclear family, S_{js} is computed iteratively as

$$S_{js} = \sum_{g_m, g_f} R_{sm} R_{sf} \prod_{k \in C_s} [\sum_{g_k} \Pr(g_k|g_m, g_f) R_{sk}], \quad (3.8)$$

where m and f are the parents in the nuclear-family node.

If j is an individual in the cutset node s , S_{js} is computed iteratively as

$$S_{js} = \sum c_s(g_{s_1}, \dots, g_{s_n}) \prod_{l \in c_s} R_{sl}, \quad (3.9)$$

where the summation is over the genotypes of the individuals included in cutset s , except for the genotype of individual j and s_1, \dots, s_n are the individuals in cutset s .

3.1 Metropolis-Hastings algorithm

The Metropolis-Hastings acceptance probability is

$$\eta = \min \left(1, \frac{\pi(g_c) q(g_{prev}|g_c)}{\pi(g_{prev}) q(g_c|g_{prev})} \right), \quad (3.10)$$

where π is the target distribution and q is the proposal distribution, g_{prev} is the accepted draw from the previous round and g_c is the sampled candidate from the present round. The chain moves from g_{prev} to g_c with probability η , and it stays at g_{prev} with probability $1 - \eta$, in this case the draw is rejected. The key step is to chose a good proposal distribution so that the rejection rate is minimized. We consider the special case of *independence sampling*: instead of $q(g_c|g_{prev})$ and $q(g_{prev}|g_c)$, we use $q(g_{prev})$ and $q(g_c)$, i.e. we assume that $q(g_c|g_{prev}) = q(g_c)$ and $q(g_{prev}|g_c) = q(g_{prev})$. Thus

$$\eta = \min \left(1, \frac{\pi(g_c)q(g_{prev})}{\pi(g_{prev})q(g_c)} \right). \quad (3.11)$$

We sample genotypes from the iteratively peeled pedigree and use them in the Metropolis-Hastings step to be rejected or accepted. We use the following expression to obtain $\pi(\cdot)$ on the true pedigree,

$$\pi(\mathbf{g}) = \prod_{j=1}^{n_1} P(g_j) \prod_{j=n_1+1}^n P(g_j|g_{f_j}, g_{m_j}), \quad (3.12)$$

where g_{f_j}, g_{m_j} are the genotypes of the parents of individual j and n_1 is the number of founders. In this example, $\pi(\mathbf{g})$ is

$$\pi(\mathbf{g}) = P(g_1)P(g_2)P(g_3|g_1, g_2)P(g_4|g_1, g_2)P(g_5|g_3, g_4)P(g_6|g_3, g_4).$$

To compute $q(\cdot)$ we multiply the probabilities that were used in the sampling process described above. For example, for this pedigree $q(\mathbf{g})$ is

$$q(\mathbf{g}) = f(g_1|\mathbf{y})f(g_2|\mathbf{y}, g_1) \cdots f(g_6|g_1, g_2, g_3, g_4, g_5, \mathbf{y}).$$

3.2 Improving efficiency of sampler

The efficiency of the sampler can be improved by combining exact peeling with iterative peeling. Exact peeling (Section 2) is used until the size of cutsets gets large enough to make computations infeasible. Then, iterative peeling is used, which as discussed above is equivalent to cutting and extending the remaining loops.

After iteratively updating all the edge probabilities, we sample genotypes for all individuals in the pedigree. First we sample genotypes for the individuals that were not peeled out. We start from an arbitrary “un-peeled” individual, and we sample its genotype using the marginal probability function $f(g_j|\mathbf{y})$. Then we sample all its unsampled neighbors using conditional probabilities (as we did in the small example for individual 2). Before obtaining a new sample, edges are updated to reflect the already sampled genotypes. A neighbor is defined as any individual who is also a member of those nuclear-family or cutset nodes, to which the sampled individual belongs to. Once all remaining individuals are sampled, we sample genotypes of the “peeled” individuals in the inverse order of peeling (see Section 2).

4 Estimation of genotype probabilities in the Labrador Retriever pedigree

One possible approach to estimate genotype probabilities in large pedigrees with complex loops is to approximate the calculations by using iterative peeling without the Metropolis-Hastings step.

This approach will be compared to the proposed method. The proposed sampling method can be used to estimate the genotype probabilities by sampling from a proposal distribution generated by iterative peeling for the entire pedigree or by peeling exactly up to a certain point and then perform iterative peeling.

We compare the results obtained from the three different approaches.

4.1 Application of the methods in a real pedigree

A real pedigree was used to test the performance of the sampling algorithm that we just described. The trait of interest for this pedigree is a disease called progressive retinal atrophy (PRA). This disease is transmitted by a recessive allele and the dog is affected when it has the recessive homozygous genotype (aa). The pedigree consists of 3,052 dogs (Labrador Retrievers) from “The Seeing Eye, Inc”, and 33 of them are known to have the disease. That is for these 33 dogs we know the genotype and phenotype. For the rest of the dogs we are interested in obtaining estimates of the genotype probabilities to determine which dogs are at highest risk of transmitting the PRA gene to their offspring and which dogs are at lower risk of both of transmitting the gene and of being PRA affected.

Exact peeling methods cannot be used in this pedigree because there are 679 loops that need to be cut. Thus, we used two different variations of the proposed method to compute genotype probabilities. In the first variation we peeled exactly the pedigree up to cutsets of size 5 (C_5). We then iteratively peeled the remainder core pedigree and sampled genotypes according to Metropolis-Hastings algorithm as described above. The second variation was similar to the first with the only difference that we exactly peeled the pedigree up to cutsets of size 7 (C_7). The chain length in both cases (C_5 and C_7) was 10,000. To compare the results we computed the absolute difference between the genotype probabilities for each animal under the two variations of the proposed method. Those results are shown in the left-hand side of Figure 4.4. Also, we computed the absolute difference between the genotype probabilities between one variation of the proposed method (C_7) and the approximate method (iterative peeling without sampling). These results are shown in Figure 4.4, right-hand side. In Figure 4.4 the plots in the right hand side show that the differences between the two variations of the proposed method (C_7 and C_5) are not large. For most of the animals, the differences between the genotype probabilities estimated using the C_5

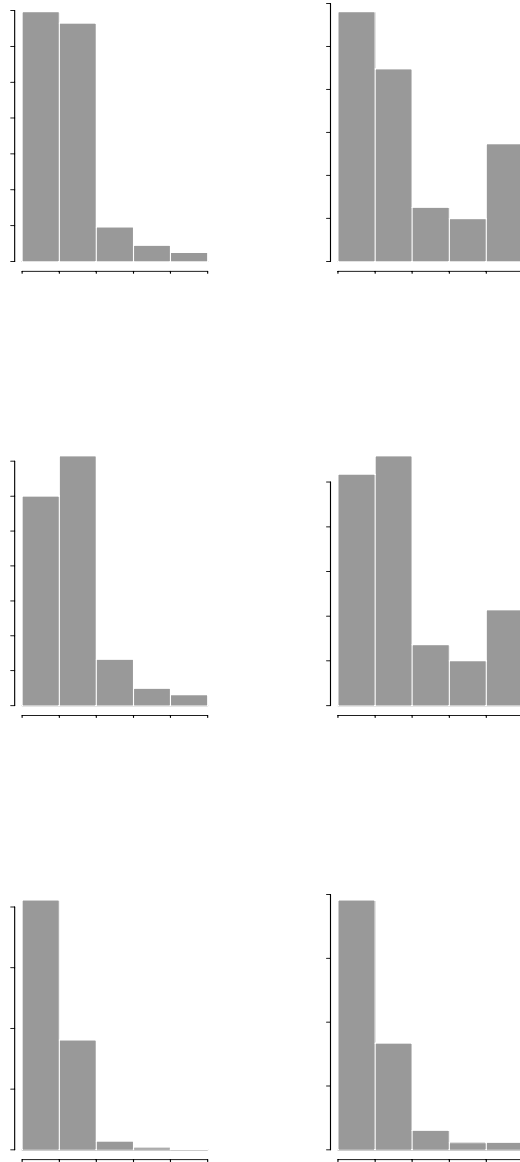


FIGURE 4.4. Histograms for the absolute differences for the three possible genotypes for the dog pedigree

or C_7 cutset sizes were smaller than 0.010. When one of the variations of the proposed method (C_7) is compared to the approximate method, we observe that there is a considerably larger number of individuals (more than 800 for the genotypes AA and Aa) with absolute differences larger than 0.010. The chain consisted of only 10,000 samples, we expect that these estimates differ more from the approximation when we run the program for longer time.

The rejection rate for the C_5 and C_7 approaches were 86% and 83%, respectively. The computational time was reduced by more than 3 hours in the C_7 approach. Thus, it seems that the C_7 approach is more efficient.

We also tried to run the program for cutset size=0, that is using the proposal generated by performing iterative peeling for the entire pedigree (with no exact peeling). After three days the program was at iteration 568 and we decided to stop the program.

In general, it seems that it is more efficient to peel exactly the pedigree as much as possible and then perform iterative peeling in the remainder core pedigree to obtain the proposal distribution to be used in Metropolis-Hastings algorithm. But, we cannot peel too much because then the cutsets become larger and that is very expensive in computation time and memory.

Note that the rejection rates are large, more than 80%. The reason is that in this pedigree only 33 out of the 3,052 dogs have known genotype, thus there are many possible configurations.

4.2 Results: proportion of dogs carrying the PRA gene

The estimated number of affected and carrier animals are presented in Table 4.1, we observe that the estimated number of affected dogs is 40 (including the 33 dogs known to have the disease).

TABLE 4.1. Estimated number of affected and carrier animals.

Genotype probability	No. of animals
$0.5 \leq P(aa) < 0.6$	546
$0.6 \leq P(aa) < 0.7$	455
$0.7 \leq P(aa) < 0.8$	257
$0.8 \leq P(aa) < 0.9$	107
$0.9 \leq P(aa) < 1$	157
$P(aa) = 1$	40
$0.5 \leq P(Aa) < 0.6$	440
$0.6 \leq P(Aa) < 0.7$	236
$0.7 \leq P(Aa) < 0.8$	58

4.3 Assessing the performance of the algorithm

To assess the performance of the algorithm we used a small pedigree with loops. We considered the inheritance at a single biallelic disease locus. This small pedigree consists of 77 individuals, 2 of them are affected. We sampled genotypes for all the individuals and computed the genotype probabilities. The length of the chain was 50,000.

In this small pedigree we can perform exact calculations by exact peeling. We compare the results of exact peeling (truth) with our approach (proposed method), and also with the approximate iterative peeling method without sampling (approximate method). We computed the absolute difference between the true probabilities and the approximate method for the three possible genotypes for the 77 individuals. The results are shown in the histograms in Figure 4.5 left-hand side. We observe that the differences were not larger than 0.030 in all cases. Also, we computed the absolute differences between the true probabilities and the proposed method. Those results are shown in Figure 4.5 right-hand side. There, we observe that there are less number of individuals in the categories of larger differences. For example, for genotype AA , more than 70 individuals have an absolute difference between 0 and 0.01, and there are only a few individuals in the class 0.010 to 0.015. Thus, the proposed method yields more accurate estimates (closer to the truth). If we increase the number of samples, the estimates improve even more. Even though this is a small pedigree and not a thorough evaluation our method gives good results.

5 Summary and Conclusions

Estimating the frequency of genotypes at biallelic loci is non-trivial when pedigrees are large and contain loops. In this case, scalar Gibbs sampling approach cannot be used, as the chains are reducible in practice. We propose a more general Metropolis-Hastings approach to sampling genotypes jointly from complex pedigrees, in which candidate draws are obtained from modified pedigrees. These modified pedigrees are obtained by applying extensions of traditional peeling methods, and are used as candidate draws in the Metropolis-Hastings step. The resulting Markov chains satisfy the assumptions that are required for good performance of MCMC methods.

The method for sampling genotypes was developed to address the problem of estimating genotype probabilities of the alleles responsible for causing progressive retinal atrophy (PRA) in Labrador Retrievers. The pedigree data of interest, collected by veterinarians at “The Seeing Eye, Inc.” included over 3,000 animals, and had over 600 closed loops created by inbreeding and multiple matings. A summary of the results of this pedigree analysis is given in Table (4.1).

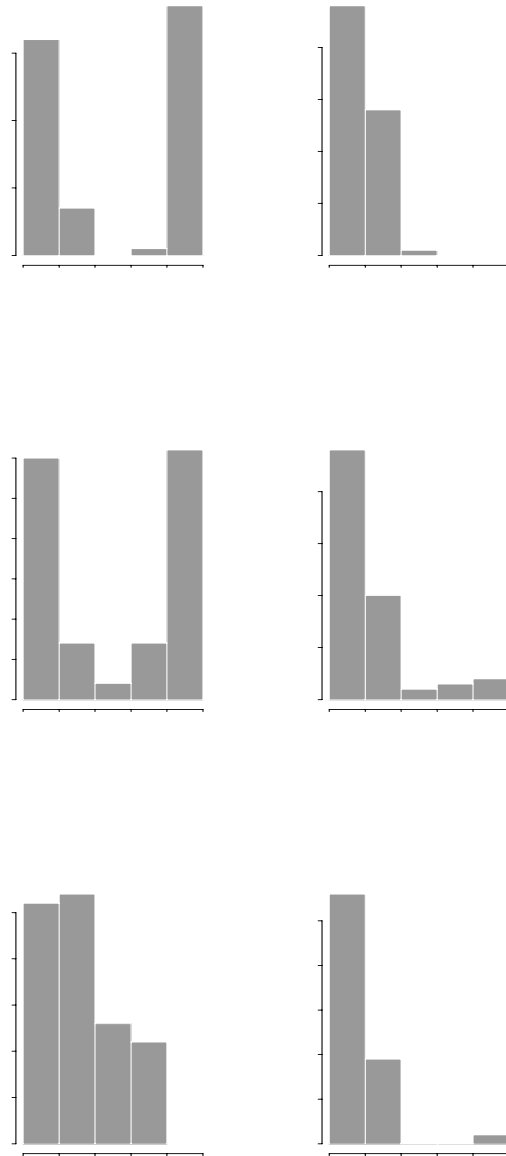


FIGURE 4.5. Histograms for the absolute differences for the three possible genotypes in a small pedigree.

Acknowledgments

The authors are grateful to Dr. Eldin A. Leighton, Director of Canine Genetics of “The Seeing Eye, Inc.”, who provided the pedigree data used in this analysis.

Bibliography

- Cannings, C., Thompson, E., and Skolnick, E. (1978). Probability functions on complex pedigrees. *Adv. Appl. Prod.*, 10:26–61.
- Elston, R. (1987). Human quantitative genetics. In G., W. B. E. E. G. M. N., editor, *Proc. 2nd. Int. Conf. Quant. Genet.*, pages 281–282, Sinauer, Sunderland.
- Elston, R. and Rao, D. C. (1978). Statistical modeling and analysis in human genetics. *Annu Rev Biophys Bioeng*, (7):253–286.
- Elston, R. and Stewart, J. (1971). A general model for the genetic analysis of pedigree data. *Hum Hered*, 21:523–542.
- Fernando, R., Stricker, C., and Elston, R. (1993). An efficient algorithm to compute the posterior genotypic distribution for every member of a pedigree without loops. *Theor. Appl. Genet.*, 87:89–93.
- Fernando, R., Stricker, C., and Elston, R. (1994). The finite polygenic mixed model: an alternative formulation for the mixed model of inheritance. *Theor. Appl. Genet.*, 88:573–580.
- Heath, S. C. (1998). Generating consistent genotypic configurations for multi-allelic loci and large complex pedigrees. *Human Heredity*, 48:1–11.
- Janss, L., der Werf J.H.J, V., and van Arendonk J.A.M. (1992). Detection of a major gene using segregation analysis in data from generations. *Proc. Eur. Assoc. Anim. Prod.*
- Janss, L., Thompson, R., and Van Arendonk, J. (1995). Application of Gibbs sampling for inference in a mixed major gene-polygenic inheritance model in animal populations. *Theor. Appl. Genet.*, 91:1137–1147.
- Jensen, C., Kong, A., and U., K. (1995). Blocking Gibbs Sampling in very large probabilistic expert systems. *International-Journal of Human Computer Studies*, 42:647–66.
- Jensen, C. and Sheehan, N. (1998). Problems with determination of non-communicating classes for Monte Carlo Markov chain applications in pedigree analysis. *Biometrics*, 54:416–425.

- Lange, K. and Boehnke, M. (1983). Extensions to pedigree analysis. v. Optimal calculation of Mendelian likelihoods. *Hum. Hered.*, 33:291–301.
- Lange, K. and Elston, R. (1975). Extensions to pedigree analysis. i. Likelihood calculations for simple and complex pedigrees. *Hum. Hered.*, 25:95–105.
- Sheehan, N. (1990). *Genetic restoration on complex pedigrees*. PhD thesis, University of Washington, Seattle.
- Stricker, C., Fernando, R., and Elston, R. (1995). An algorithm to approximate the likelihood for pedigree data with loops by cutting. *Theor. Appl. Genet.*, 91:1054–1063.
- Thomas, A. (1986a). Approximate computations of probability functions for pedigree analysis. *IMA J Math Appl Med Biol*, 3:157–166.
- Thomas, A. (1986b). Optimal computations of probability functions for pedigree analysis. *IMJ J Math Appl Med Biol*, 3:167–178.
- Van Arendonk, J., Smith, C., and Kennedy, B. (1989). Method to estimate genotype probabilities at individual loci in farm livestock. *Theor. Appl. Genet.*, 78:735–740.
- Wang, T., Fernando, R., Stricker, C., and Elston, R. (1996). An approximation to the likelihood for a pedigree with loops. *Theor. Appl. Genet.*, 93:1299–1309.