Haplotype Inference Using a Hidden Markov Model with Efficient Markov Chain Sampling

by

Shuying Sun

A thesis submitted in conformity with the requirements for the
Degree of Doctor of Philosophy
Graduate Department of Statistics
University of Toronto

Haplotype Inference Using a Hidden Markov Model with Efficient Markov Chain Sampling

Shuying Sun
Doctor of Philosophy
Department of Statistics, University of Toronto
Convocation of June 2007

Abstract

Knowledge of haplotypes is useful for understanding block structures of the genome and finding genes associated with disease. Direct measurement of haplotypes in the absence of family data is presently impractical. Hence several methods have been developed previously for reconstructing haplotypes from population data. In this thesis, a new population-based method is developed using a Hidden Markov Model (HMM) for the source of ancestral haplotype segments. A higher-order Markov model is used to account for linkage disequilibrium in the ancestral haplotypes. The HMM includes parameters for the genotyping error rate, the mutation rate, and the recombination rate. Four mutation models with varying number of parameters are developed and compared. Parameters of the model are inferred by Bayesian methods, using Markov Chain Monte Carlo (MCMC). Crucial to the efficiency of the Markov chain sampling is the use of a Forward-Backward algorithm for summing over all possible state sequences of the HMM. This model is tested by reconstructing the haplotypes of 129 children in the data set of Daly et al. (2001) and of 30 children in the CEU and YRI data of the HAPMAP project. For these data sets, family-based haplotype reconstructions found using MERLIN (Abecasis et al. 2002) are used to check the correctness of the population-based reconstructions. The results of this HMM method are quite close to the family-based reconstructions and comparable to the PHASE program (Stephens et al. 2001, Stephens and Donnelly 2003, Stephens and Scheet 2005) and the fastPHASE program (Scheet and Stephens 2006). The recombination rates inferred from this HMM method can help to predict haplotype block boundaries, and identify recombination hotspots.

Acknowledgements

Love is everywhere.

First of all, I am most grateful to my thesis advisors, Professors Radford Neal and Celia Greenwood, for their guidance and kind support. I really appreciate my external examiner, Professor Hugh Chipman, for his thoughtful questions and comments. I am grateful to Professors Radu Craiu and Shelley Bull for their questions and suggestions as well.

This work was made possible by the facilities of the Shared Hierarchical Academic Research Computing Network (SHARCNET: www.sharcnet.ca), the computer systems of Hospital for Sick Children and Department of Statistics.

This work was supported by MITACS/NCE (Mathematics of Information Technology and Complex Systems/Networks of Centres of Excellence Program) and NSERC (Natural Sciences and Engineering Research Council of Canada).

Department of statistics is like a big family. It has nurtured me for the past four years. I have had wonderful time. I am very grateful for the professors, the staff members and my colleagues who have helped me a lot. In particular, I am very grateful for Professor Evans, and my colleagues Longhai, Mark, Steve, Mahinda and Xiaobin for spending time discussing questions with me and helping me in different ways.

I would like to thank my friends: Jacinta, Yaxing, Changshun, Du Ying, Ding Ning, Vanessa, Qihui, Xiaoxiao, and Emile for their sincere friendships. In addition, I

would like to show my gratitude to all my toastmaster friends for their kind support.

One important person in my life is Professor Zongjing Wang. Her unconditional love is always with me for the past twelve years.

I am grateful to Professor Terry Speed. I really appreciate his kind help and his wisdom.

Two cats (named Barbie and Book) and Lake of Ontario are the non-human beings that have brought a lot of peace and comfort to my life in the past four years. I really appreciate them.

My mom, three sisters, two brothers and their families always love me. Without their deepest love and support, I would not have been able to finish this thesis.

Finally, I would love to dedicate this thesis to my father and Professor Yang, Xiaoyong. I would like to ask Toronto's wind to be the messenger to whisper to them that I have finished my thesis. I wish this piece of news will bring peace and comfort to their souls.

Love is everywhere.

Contents

List of Tables

List of Figures

Chapter 1

Introduction

1.1 Basic genetics background

In order to explain what haplotype inference is, I will first introduce some basic genetic terminology. Each human cell has 23 pairs of chromosomes. Each chromosome consists of two long strands of deoxyribonucleic acid (DNA) that are tightly connected. One strand of DNA is a chain of nucleotides coded as A, C, G, and T. The other strand contains complementary nucleotides in which A pairs with T and C pairs with G. A gene is a region of chromosomal DNA that encodes a specific functional product such as a protein. Each gene has a fixed location in the genome. A marker is a short identifiable sequence that has a known location, and this sequence varies between individual chromosomes. Variants of genes or markers are called alleles. The genotype of a gene or a marker is the pair of alleles occurring at that locus on the two chromosomes of an individual. The pair of alleles is usually coded as letters or numbers: for example, AC

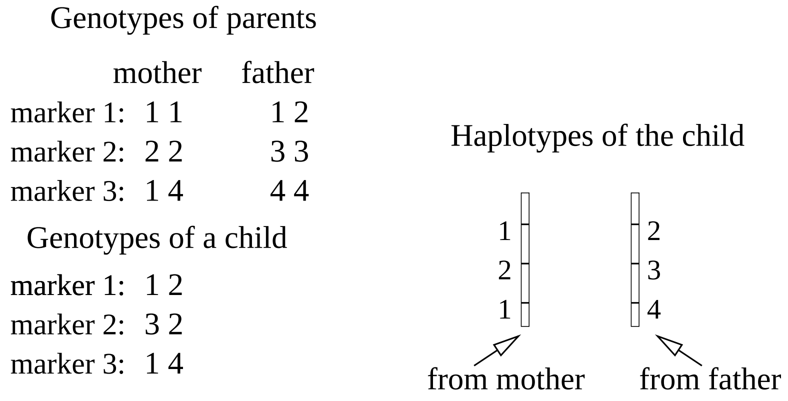


Figure 1.1: Examples of genotypes, haplotypes and genetic inheritance.

and 24 may be used to represent the genotypes of two markers. Two common types of genetic markers are single nucleotide polymorphisms (SNPs), which have two alleles, and microsatellite markers, which can have many alleles.

DNA (genotype) is inherited from the two parents of each child, with each parent contributing one of his or her chromosomes. The parental origin of any particular DNA segment is not directly observable without additional information. However, by observing the alleles (genotypes) in the parents and the child, inference can be made about the parental origins of alleles. For example, in Figure 1.1, if only the genotypes of the child are observed, how those alleles were inherited from the two parents is unknown. However, with the parents' genotyping information, one can infer that alleles 1, 2 and 1 were inherited together from the mother, and alleles 2, 3 and 4 were inherited from the father.

An ordered sequence of alleles on part or all of a chromosome is called a haplotype. For example, in Figure 1.1, 1 2 1 is one haplotype, and 2 3 4 is another haplotype.

Haplotype inference tries to reconstruct the two haplotypes of each individual from his (or her) genotypes. This reconstruction might be done with or without additional family genotype information.

A haplotype block is a sequence of markers that are closely located and usually inherited as a group (block). This 'inheriting together' concept is related to linkage disequilibrium (LD). When alleles at different loci in a haplotype are not independent, they are said to be in linkage disequilibrium. For example, if A/a and B/b are the possible alleles at two loci, under linkage disequilibrium, the haplotype frequency of haplotype AB is not equal to the product of allele frequencies of allele A and allele B (i.e. $f(AB) \neq f(A)f(B)$). If the two loci are independent, then $f(AB) = f(A)f(B)$, and this is called linkage equilibrium.

Recombination and mutation are important for haplotype inference, since these two genetic processes can lead to variability in DNA sequences between individuals after inheritance and evolution over many generations. The chromosomes transmitted to children are created during meiosis, the production of gametes. During meiosis, crossover of the strands of parental chromosomes may happen, and recombination may occur. Recombination leads to decay of linkage disequilibrium; closely related individuals will share long chromosomal segments whereas distantly related individuals share much shorter segments. The expected number of recombinations between two loci is called the genetic distance. Genetic distance increases monotonically with physical distance, but not always linearly.

Mutation is another process that is responsible for genetic variation. Mutation refers to the rare event that the genetic material is altered. For example, a C nucleotide may undergo a mutation to become a T nucleotide. If this change has no effect on cell viability, the mutation may be inherited and can eventually become common in the population. In general, mutation rates are thought to be very small for nucleotide substitutions per generation; they range from 10^{-9} to 10^{-4} per generation per nucleotide (Griffiths et al. (2005), page 627).

1.2 Why do we do haplotype inference?

Haplotypes can be more informative with respect to patterns of inheritance than genotypes at single markers because haplotypes combine the information at close markers and also capture information about common patterns that may be descended from ancestral haplotypes (Daly et al. 2001, Akey et al. 2001, Pritchard 2001, Niu et al. 2002, Eronen et al. 2004). That haplotypes are more informative has led to the increasing importance and application of haplotype analysis. For example, haplotypes can capture regional LD information (Niu et al. 2002). In particular, identifying haplotype blocks is one way of studying linkage disequilibrium patterns (Daly et al. 2001, Zhu et al. 2004, Greenspan and Geiger 2004).

Linkage disequilibrium causes associations between markers and disease even for markers that are not part of a disease-causing gene. Therefore, one important application of haplotype analysis is in disease risk association studies (Mander 2001, Zaykin et al. 2002, Butt et al. 2005, The International HAPMAP Consortium 2005). These studies examine the association between a particular set of haplotypes and disease traits. Disease risk mutations are usually more strongly associated with a haplotype than with any one marker. For example, Yuan et al. (1999) identified the mutation MSH2*1906G>C (mismatch-repair gene) which causes colorectal cancer. Foulkes et al. (2002) studied the genetic characteristic of this mutation. Their haplotype analysis of this study shows that a haplotype of nine markers (A-3-G-288-255-177-A-G-C, the last allele 'C' is the mutation allele) is shared by all 14 families with affected individuals. That is, there is a strong association between this nine marker haplotype and the colorectal cancer. Therefore, increased success at identifying disease risk mutations may be obtained by examining associations with haplotypes. In addition to association studies, haplotype analysis is also helpful for studying population history (Chapman and Thompson 2001).

Haplotypes are very important in genetic studies. However, it is still very impractical to measure them directly (Stephens et al. 2001). As seen in Figure 1.1, obtaining haplotype estimates from family-based genotypes is possible and usually reliable. However, obtaining family data can be very difficult because family members may not be alive or may refuse to participate. As a result, researchers often need to reconstruct haplotypes from data on unrelated individuals only.

1.3 How is haplotype inference for unrelated individuals possible?

The general concept of haplotype reconstruction will be motivated by a small example with three markers. Consider an individual taken from a specific population, for whom genotypes of three heterozygous (i.e. the two alleles are different) markers are known. Without parental genotyping information, the genotypes at these three markers are equally likely fall into any of the four possible haplotype combinations as shown in Figure 1.2. If an individual has L heterozygous markers, there are 2^{L-1} possible different haplotype combinations, which is a huge number even for a moderate L. However, due to historical relatedness between all humans, only a small number of common haplotypes are likely to be present among many sampled individuals. The basic idea behind reconstructing haplotypes for unrelated individuals involves finding these common haplotype patterns. This idea is used in many current haplotype inference methods (Clark 1990, Excoffier and Slatkin 1995, Hawley and Kidd 1995, Qin et al. 2002, Stephens et al. 2001, Eronen et al. 2004).

A vivid example of common haplotypes can be seen in Figure 2 of Daly et al. (2001), in which 11 haplotype blocks were identified (Appendix A of this thesis reproduces this figure). In the first haplotype block, only two common haplotypes are present, and more than 90% of the samples have one or both of these two haplotypes. How can common patterns like this be identified and used to reconstruct haplotypes? The first haplotype reconstruction method (Clark 1990) showed how it is possible. Clark

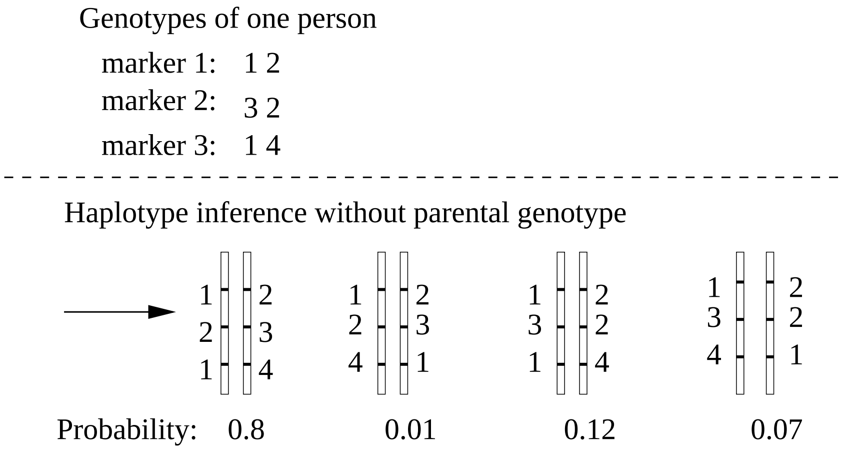


Figure 1.2: Haplotype inference without family genotype information. Note that the alleles (1 and 2) of the first marker are not "flipped" since the parental order is not irrelevant.

proposed a parsimony algorithm that starts with individuals whose genotypes for all markers are all homozygous, or for whom only one marker is heterozygous. Haplotypes are known for such individuals. Haplotypes of other individuals are estimated by assuming the known haplotypes are the only correct ones. Since Clark's work, much more sophisticated methods have been developed. When there is no unique haplotype reconstruction for an individual, haplotype inference will provide an estimate of the probability distribution of different haplotypes for this individual, as shown in the last row of the toy example in Figure 1.2.

1.4 The structure of this thesis

In the following chapters, I review some existing haplotype inference methods (chapter 2), present my new Bayesian method which uses a Hidden Markov Model (HMM) (chapter 3), and show how to implement this HMM using Markov Chain Monte Carlo

(MCMC) methods (chapter 4). During the course of this research, I learned of three other haplotype inference methods that use an HMM; these methods are discussed in chapter 5. In chapter 6, I compare the performance of my model with other methods. Finally, I discuss these results and present some ideas for future research directions in chapter 7.

Chapter 2

Review

2.1 Introduction

Since Clark's first haplotype inference method (Clark 1990), there have been many haplotype inference methods developed using different ideas and computational techniques. In this chapter, I will mainly review those methods that are closely related to the new methodology presented in this thesis. In the following sections, I will review haplotype inference methods that use the EM algorithm, Bayesian methods, and methods that directly model linkage disequilibrium. Finally I will list a few other methods. Note that this thesis focuses on developing haplotype inference for unrelated individuals; therefore, no family-based haplotype inference methods are reviewed.

2.2 Haplotype inference using the EM algorithm

The EM algorithm (Dempster et al. 1977) is an iterative method of finding the maximum likelihood estimates for unknown parameters when the model includes some latent variables, or the data set has some missing data. This algorithm consists of an Expectation (E) step and a Maximization (M) step. At the beginning, the unknown parameters are given some initial values. The distribution of the latent variables (or the missing data) is then estimated using the observed variables and current estimates for the unknown parameters. This is the E-step. In the M-step, using the estimated distribution of latent variables, an estimate for the unknown parameter is obtained by maximizing the expected log likelihood. These E and M steps are repeated alternately until there is little change in the estimates of the unknown parameters. The final estimate will be at least a local maximum of the likelihood for the model, marginalizing over latent variables.

Quite a few researchers, such as Excoffier and Slatkin (1995), Hawley and Kidd (1995), Long et al. (1995), Qin et al. (2002) and Polanska (2003), have used the EM algorithm to estimate haplotype frequencies and reconstruct haplotypes for unrelated individuals. The general idea of these methods is as follows.

Suppose there are P people in the sample. Let $G = (G_1, \cdots, G_P)$ be their genotypes, and let $H = (h_1, \cdots, h_n)$ be the haplotypes in the population. If the total number of heterozygous loci in G is Z, the maximum number of different haplotypes need to be included in the EM algorithm is 2^{Z-1}. Let $\theta = (\theta_1, \cdots, \theta_n)$ be the frequencies of those n haplotypes. Some people may have the same genotypes, even though their haplotypes may be different. Suppose there are m different genotype classes, and each genotype class is observed with count x_i $(1 \leq i \leq m)$, where $\sum_i x_i = P$. Assume the frequency of each genotype class is α_i $(1 \leq i \leq m)$, the probability of obtaining these genotypes for all P people is,

$$P(\text{genotype frequencies}|\alpha_1, \cdots, \alpha_m) = \frac{P!}{x_1!x_2!\cdots x_m!} \times \alpha_1^{x_1} \times \alpha_2^{x_2} \times \cdots \times \alpha_m^{x_m} \quad (2.1)$$

For genotype class i $(1 \leq i \leq m)$, if there are r_i different heterozygous markers, there are $w_i = 2^{r_i - 1}$ different haplotype combinations. Therefore,

$$\alpha_i = \; P(\text{genotype class i}) = \sum_{j=1}^{w_i} P(h_{u_j}, h_{v_j}) = \sum_{j=1}^{w_i} P(h_{u_j})P(h_{v_j}) = \sum_{j=1}^{w_i} \theta_{u_j}\theta_{v_j}$$

In the above formula, for each j $(1 \leq j \leq w_i)$, u_j and v_j are the haplotype indexes, $1 \leq u_j, v_j \leq n$. Substituting the above equation in equation 2.1, the likelihood of haplotype frequencies is obtained as follows,

$$L(\theta_1, \cdots, \theta_n) \propto \prod_{i=1}^{m} \alpha_i^{x_i} \propto \prod_{i=1}^{m} \left[\sum_{j=1}^{w_i} \theta_{u_j}\theta_{v_j} \right]^{x_i}$$

The EM algorithm can be used to estimate the haplotype frequencies as follows. First, assign some initial values to the haplotype frequencies, $\theta^{(0)}$, this is the initialization step. In the E step, reconstruct haplotypes for each genotype class in a probabilistic way, and estimate the genotype class frequencies $(\alpha_1^{(t)}, \cdots, \alpha_m^{(t)})$ using the genotypes and the haplotype frequencies $\theta^{(t-1)}$, where $t \geq 1$. In the M step, use these estimated genotype class frequencies to get the MLE of θ.

For convenience, I will refer EM methods by their software names, or by a name related to the key idea of a method or by its author's name, as follows: refer to Excoffier and Slatkin (1995) as EMDECODER, Hawley and Kidd (1995) as HAPLO, Long et al. (1995) as LONG-EM, Qin et al. (2002) as PLEM, and Polanska (2003) as Polanska-EM.

An important issue when using the EM algorithm is that it may find only a local maximum of the likelihood. To deal with this problem, Excoffier and Slatkin (1995) suggests several ways of setting initial parameter values in EMDECODER. The EM algorithm in HAPLO sets all haplotype frequency initial values equal, though this may not solve this problem. Long-EM (Polanska 2003) randomly sets values for haplotype frequencies as the starting values. In order to check if the global maximum has been reached, Long-EM tries 1000 random initial values. The author of Polanska-EM recommends using randomized initial values as well.

Another important issue is constraints on memory and computation time, since the number of haplotype frequency parameters grows exponentially with the number of heterozygous markers. In order to solve this problem, PLEM uses the idea of Partition-Ligation (Niu et al. 2002). In fact, PLEM is the idea of Partition-Ligation in combination with the EM algorithm. Partition-Ligation means dividing many markers into small groups of adjacent markers (partition) and doing haplotype estimation in each small unit, and then combining the estimated haplotypes from each unit (ligation). This PL idea can help solve the memory problem, but the local maximum problem may become more serious (Qin et al. 2002). In addition, PLEM uses a 'backup-buffering' strategy to deal with the local mode problem. Detailed explanations and discussions about strategies using in PLEM can be found in Qin et al. (2002).

All of the above methods can perform well to some extent. However, they have some limitations. First, starting the EM algorithm from different initial conditions may help get closer to the global optimum, but, the sensitivity of the final estimates to the initial conditions is largely unknown. Second, these methods may not perform well if the data are in low linkage disequilibrium (LD). In fact, the LD level affects the shape of the likelihood hypersurface (Polanska 2003); that is, high LD leads to a smooth shape for the likelihood, whereas low LD can cause a non-smooth shape for the likelihood. In particular, when there are recombination hotspots, where the LD level may be very low, PLEM results may not be very consistent across different partitions. Third, missing genotypes may also affect the performance of the EM algorithm (Qin et al. 2002), since all possible genotypes must be considered when a genotype is missing, this may increase the memory problem.

2.3 Bayesian methods

2.3.1 Haplotype inference using Bayesian methods

This thesis research was first presented in the annual meeting of American Society of Human Genetics in October 2004 (Sun et al. 2004). Prior to that time, only a few Bayesian methods have been developed for haplotype inference. In particular, the following four algorithms use Bayesian concepts to motivate their haplotype estimators:

the PHASE program (Stephens et al. 2001, Stephens and Donnelly 2003), HAPLOTYPER (Niu et al. 2002), the modified SSD method (Lin et al. 2002), and a method using the Dirichlet process (Xing et al. 2004). In this section, I will briefly review the main ideas behind these four models, and then investigate the prior distributions and computational techniques used. The latter two aspects of model fitting are particularly important in Bayesian methods.

The fundamental idea of Bayesian inference is that both the model parameters (θ) and the observed data are considered as random variables and are modeled using probability distributions (Gelman et al. 1995). The parameters are given a prior distribution, $P(\theta)$, then through the likelihood function, $P(Y|\theta)$, the parameter can be estimated from the posterior density, $P(\theta|Y) \propto P(\theta)P(Y|\theta)$. All of the above four methods therefore treat the unknown haplotypes of each individual as random variables. The main difference between using the EM algorithm and a Bayesian method to do haplotype inference is whether the haplotype frequencies in the population are treated as random variables or not. Another important common aspect of these above four methods is that they all used Markov Chain Monte Carlo (MCMC) methods to sample from the posterior distribution. Other common and different aspects of these methods are summarized in Table 2.1. More details are explained below.

2.3.2 Comparing the four Bayesian methods

The HAPLOTYPER software has three key elements in its model: genotypes, haplotypes and haplotype frequencies. A Dirichlet prior was assigned to the haplotype frequencies, and the Gibbs sampler was used to sample from the posterior distribution. Two computational tricks used are 'prior annealing' and 'Partition-Ligation (PL)', which make the Markov Chain converge faster (Niu et al. 2002). The prior annealing trick involves using large pseudo-counts as the initial prior for the haplotype frequencies, to allow the Markov chain to move freely in its state space. The PL idea here is the same as the one mentioned in the previous section (Qin et al. 2002). That is, long sequences of markers are first divided into small segments to do the haplotype inference, then results are combined across the small segments.

This PL idea has also been incorporated into the new version of the PHASE pro-

Method	**Probability**	**Prior**	**MCMC**	**Trick**
HAPLOTYPER	$P(G, H, \theta)$	Dirichlet (θ)	Gibbs	PL, PA
PHASE v2.0	$P(H_p \mid G, H_{-p})$	Coalescent ($\pi(\cdot \mid H)$)	Gibbs	PL
Modified SSD	$P(H_p \mid G, H_{-p})$	Dirichlet ($\pi(\cdot \mid H)$)	Gibbs	PL
Xing	$P(A, S, m, G, H, r)$	DP (A)	Gibbs-MH	-

Table 2.1: Summary of the common and different aspects of four Bayesian methods. In the 'probability' column, the probability model of HAPLOTYPER is $P(G, H, \theta) = P(G, H|\theta)P(\theta)$. The probability model of PHASE is: $P(H_p|G, H_{-p}) \propto \pi(h_{p1}|H_{-p})\pi(h_{p2}|H_{-p}, h_{p1})$. Here H_p means the two haplotypes of person p. In the 'trick' column, 'PL, PA' means 'partition-ligation' and 'prior annealing'. In the 'MCMC ' column, 'Gibbs-MH' means both the Gibbs sampler and the Metropolis Hastings algorithm are used. In the 'prior' column, 'DP' means that the Dirichlet process is the prior.

gram, PHASEv2.0 (Stephens and Donnelly 2003). In the first version of PHASE (PHASEv1.0) (Stephens et al. 2001), two approaches were introduced. One (their Algorithm 2) was to use a simple Gibbs sampler with a Dirichlet prior for the haplotype frequencies. Note, different features of the Dirichlet prior are attributed compared with HAPLOTYPER as commented in Stephens and Donnelly (2003). Whereas the second algorithm used a prior that approximates distributions under a coalescent model. The key idea in PHASE is to capture the similarities between known haplotypes and unresolved haplotypes. It is worth mentioning that PHASE is a pseudo-Bayesian method (Niu et al. 2002, Stephens et al. 2001, Stephens and Donnelly 2003). Specifically, that means that the posterior distribution of the haplotypes (given the genotypes and other quantities of interest) is not explicitly derived based on the prior and the likelihood. Instead, the posterior is defined as the stationary distribution of a Markov chain.

Note, in this section, the PHASE program is the version corresponding to Stephens et al. (2001) and Stephens and Donnelly (2003), it merely uses an approximate 'coalescent prior' without recombination. The version that approximates a 'coalescent with recombination' (Stephens and Scheet 2005) will be reviewed in section 2.4.

The modified SSD method (Lin et al. 2002) is based on a modification of Algorithm 2 of Stephens et al. (2001). Two main modifications were made. (1) While estimating the haplotype of a person, instead of considering haplotype sequences at all markers, it considers only the heterozygous loci, estimates are updated merely by accounting for the heterozygous loci of other individuals (that is, the homozygous markers are ignored). (2) The modified SSD method allowed for missing data. With reference to modification (1), ignoring the homozygous markers and only considering heterozygous loci when estimating haplotypes might not be the best strategy since the homozygous loci may contain useful information (e.g. LD between markers) (Stephens and Donnelly 2003). Similar to Algorithm 2 of Stephens et al. (2001) and HAPLOTYPER, the modified SSD method uses the Dirichlet prior.

Another Bayesian method was developed by Xing et al. (2004), who introduce the concept of a pool of ancestral templates and use a Dirichlet Process as prior for ancestral templates. Parameters for mutation rates and genotyping errors are included in the model as well. Both the Gibbs sampler and the Metropolis Hastings algorithm are used to sample from the posterior distribution of quantities of interest. As remarked in Xing et al. (2004), when sampling the parameters which indicate from which ancestral haplotype each individual inherited his (or her) genetic information, the Metropolis Hastings algorithm can produce better results than the Gibbs sampler.

2.3.3 Comments on recombination

None of the above Bayesian methods account for recombination between markers. HAPLOTYPER may be sensitive to recombination hotspots (Niu et al. 2002). PHASE works well for markers that are tightly linked and when loci span large distances but with no recombination hotspots (Stephens et al. 2001). The method of Xing et al. (2004) explicitly assumes no recombination. This assumption of no recombination is unlikely to be realistic for large sets of markers spanning several centimorgans. Hence, the existence of recombination could be one reason that the Gibbs sampler does not work as well as the Metropolis Hastings algorithm in Xing et al. (2004). This is because, at consecutive loci, the parameters that indicate from which ancestral haplotypes each individual inherited the genetic information might be highly correlated. This high correlation may make the Gibbs sampler prefer the states that are same as the previous one, so it is hard to move to other distinct states which might be more suitable due to the underlying existence of recombination.

2.4 Modeling linkage disequilibrium

Most of the current haplotype inference methods do not model the effect and level of linkage disequilibrium (LD) in an explicit way (Stephens et al. 2001, Stephens and Donnelly 2003), even though their methods are developed under the assumption of LD. As far as I am aware, there are two methods that have considered this issue. They are the variable order Markov model by Eronen et al. (2004) and the idea of 'coalescence with recombination' by Stephens and Scheet (2005). These two methods were motivated by different ideas. The former paper was mainly motivated by the basic definition of LD and disease risk association studies. The latter paper was motivated by the connection between recombination and linkage disequilibrium.

2.4.1 Using a variable order Markov model

Linkage disequilibrium can be thought of as dependence (association) among markers close to each other. Since a Markov chain can model the dependence between random variables, Eronen et al. (2004) used a Markov model to account for linkage disequilibrium. In order to model LD, a haplotype segment concept was introduced. A haplotype segment is a haplotype sequence from the i^{th} to the j^{th} marker, denoted as $H(i, j)$. Unlike the other haplotype estimation methods that estimated haplotype frequencies, Eronen et al. (2004) estimated frequencies of haplotype segments. The probabilities (frequencies) of haplotype segments were estimated from the genotypes according to the level of heterozygosity (that is, the proportion of individuals in a population that are heterozygous for a particular locus). A haplotype, H, is modeled using a high order Markov model, that is, $P(H) = P(H(1, d)) \prod_{i=d+1}^{L} P(H(i)|H(i-d, i-1))$.

Eronen et al. (2004) remarked that fixed order Markov models may not perform very well. For order $d = 1$, the results are poor, but improve with increasing d initially, then they eventually become worse, as markers span larger distances and there are few common patterns in the segments. This is related to overfitting, since as d gets larger, inference requires more data in the informative neighborhood of a marker. Therefore, Eronen et al. (2004) use a variable order Markov model to account for the level of LD at different markers and haplotypes.

In addition to the main idea of modeling LD, a couple of additional points about this method are worth mentioning. First, no explicit assumptions about the existence of haplotype blocks are made, but the method does assume all samples are from one population. Second, missing genotypes were imputed based on the allele frequencies. Third, for long haplotype segments, the idea of Partition Ligation (Qin et al. 2002, Niu et al. 2002) is used to avoid the problem of the exponentially large number of haplotypes associated with increasing numbers of heterozygous markers.

2.4.2 Using a recombination model

Stephens and Scheet (2005) modified the previous version of PHASE (Stephens et al. 2001, Stephens and Donnelly 2003) by adding a feature allowing for recombination between markers (The newer version is PHASEv2.1.1). Specifically, they used a prior that they called 'coalescent with recombination'. The recombination parameter is $\rho = (\rho_1, \cdots, \rho_{L-1})$ with each $\rho_l = 4N_e c_l/d_l$, where N_e is the effective population size, c_l is the recombination rate per generation, and d_l is the physical distance. The product $\rho_l d_l$ is a measure of LD between marker l and $l+1$. In the previous version of PHASE, a haplotype for person i was sampled from $P(H_i|G_i, H_{-i})$, whereas in this new version, sampling is based on $P(H_i|G_i, H_{-i}, \rho)$. The parameter ρ is updated using the Metropolis-Hastings algorithm.

Incorporating recombination into the model increases the computation time. Therefore, in order to speed up the algorithm, instead of modeling the recombination at all iterations, PHASEv2.1.1 provides another choice, that is, to assume no recombination at first, then incorporate the recombination at the final steps. As in the previous version of PHASE, to reduce the computational cost associated with long haplotypes, the Partition-Ligation idea was used as well.

One additional point about the newer version of PHASE (Stephens and Scheet 2005) is that imputation of missing alleles and missing genotypes are done separately. For the case of missing alleles, the most common allele at that locus is imputed; for the case of missing genotypes, the most common genotype at that locus is used. While imputing the missing data, the strength of LD is not considered.

2.5 Other methods

In addition to the above methods, there are some other methods which have used different techniques and assumptions for estimating haplotypes. Clark (1990) attempted to resolve each new haplotype by drawing from the set of known haplotypes. Wang and Xu (2003), Gusfield (2003), Brown and Harrower (2005) and Huang et al. (2005) used the maximum parsimony (or pure parsimony) method which finds a minimum set of haplotypes to resolve all genotypes.

Others have used perfect phylogeny, such as Chung and Gusfield (2003), Bafna et al. (2003) and Damaschke (2003). The basic idea of perfect phylogeny (Hudson 1990) is that under the assumption of no recombination, each genetic sequence is from one single ancestor in the previous generation. As summarized in Gusfield and Orzack (2005), the perfect phylogeny haplotype problem is: given a set of genotypes, G, find a set of haplotypes, H, which define a perfect phylogeny. However, as mentioned in Halperin and Eskin (2004), only common haplotypes (rather than the full haplotype set) fit the perfect phylogeny. The idea of considering common and rare haplotypes differently is implemented using imperfect phylogeny method. Note that all of these methods will not be discussed in detail since these methods are not directly related to the method presented in this thesis.

Chapter 3

Haplotype inference using an HMM

Today's genetic information is inherited from many generations of ancestors. Therefore, all current haplotypes and genotypes can be assumed to descend from some ancestral haplotype set. These ancestral haplotypes form the hidden building blocks of the haplotype inference model in this thesis. Through repeated recombinations and mutations, ancestral haplotypes can yield many different haplotypes over generations (http://www.hapmap.org/). In this thesis, A Hidden Markov Model (HMM) is used to incorporate the idea of ancestral haplotypes, recombinations and mutations, and estimate the haplotypes of unrelated individuals based on their genotypes. Brief knowledge about a Hidden Markov Model is reviewed in the next section, before it is applied to haplotype inference.

3.1 Hidden Markov Models

The Hidden Markov Model was originally introduced in the late 1960s and early 1970s (Baum and Petrie 1966, Baum and Eagon 1967, Baum and Sell 1968, Baum et al. 1970, Baum 1972). It was first applied in speech signal processing in Baker (1975). Later it was widely applied in many areas, such as economics (Albert and Chib 1993), signal processing (Juang and Rabiner 1991), image analysis (Romberg et al. 2001), and biology, especially genetics (Churchill 1989, Kruglyak et al. 1996, Liu et al. 1999, Daly et al. 2001, Siepel and Haussler 2004).

The basic idea of Hidden Markov models is that there are hidden sequences underlying the observed ones. These hidden sequences have a Markov structure. There are three fundamental problems using a Hidden Markov model (Baum and Petrie 1966). They are calculating the probability of the observed sequences given a hidden Markov model (evaluating, also called inference); finding the most likely hidden sequences that generated the observed sequences (decoding, also known as maximization); estimating the parameters of a hidden Markov model (learning, or called estimation). Usually the learning and evaluating problems can be solved using the Forward-Backward algorithm (Baum et al. 1970), the decoding problem can be solved using the Viterbi algorithm (Viterbi 1967). In this thesis, the Forward-Backward algorithm is used. Therefore, the definition of a hidden Markov model and the Forward-Backward algorithm will be reviewed. The following material is based on Koski (2001), Rabiner (1989), Scott (2002), the web notes of Professor Roger Boyle (www.comp.leeds.ac.uk/roger/).

In a Hidden Markov Model, there are hidden states, $X_n \in \{1, \cdots, N\}$, and the observations, $Y_n \in \{1, \cdots, M\}$. The hidden states are modeled using a Markov structure defined by the initial distribution of hidden states and the transition distributions between two successive hidden states. The relations between the hidden states and observed states are modeled using emission probabilities. The following notations are used. π is the initial distribution of hidden states, i.e. $\pi(i)$ for $1 \leq i \leq N$. The transition probabilities between hidden states are written as $a_{i,j} = P(x_{n+1} = j | x_n = i)$, for all $1 \leq i \leq N, 1 \leq j \leq N$. The emission probabilities between the hidden states and the observed states are written as $b_{j,k} = P(y_n = k | x_n = j)$, that is the probability of observing $y_n = k$ when the hidden state is $x_n = j$, $1 \leq k \leq M, 1 \leq j \leq N$.

The recursive Forward-Backward algorithm was originally developed by Baum et al. (1970). The basic idea of this algorithm is explained as follows. If a sequence of symbols, $\{y_1, \cdots, y_p\}$, is observed, then the computational cost of calculating the probability of the observed sequence $(P(y_1, \cdots, y_p))$ by summing over all possible hidden states at each time $n, 1 \leq n \leq p$, is exponential in p. Instead, using the Forward-Backward algorithm, this probability can be calculated as follows: $P(y_1, \cdots, y_p) = \sum_{i=1}^{N} \alpha_n(i)\beta_n(i)$. In this formula, $\alpha_n(i)$ is the probability of obtaining the observed sequences up to time n and ending in hidden state i. That is,

$$\alpha_n(i) = P(y_1, \cdots, y_n, x_n = i)$$

$\beta_n(i)$ is the probability of obtaining the observed sequences from time $n + 1$ to time p given that the hidden state is i at time n. That is,

$$\beta_n(i) = P(y_{n+1}, \cdots, y_p | x_n = i)$$

$\alpha_n(\cdot)$ and $\beta_n(\cdot)$ are called the forward variables and backward variables respectively. They can be calculated recursively. For the forward variables, when time $n = 1$, $\alpha_1(i) = \pi(i)b_{i,y_1}$. For $n > 1$, $\alpha_n(i) = \left[\sum_{j=1}^{N} \alpha_{n-1}(j)a_{j,i}\right] b_{i,n}$. Similarly, for the backward variables, since there is no observed data after time p, $\beta_p(i) = 1, (1 \leq i \leq N)$. For the times $n < p$, $\beta_n(i) = b_{i,n+1} \sum_{j=1}^{N} a_{i,k}\beta_{n+1}(k)$.

The Viterbi algorithm is used to find the most probable sequence that generates the observed sequences. It uses a recursive method similar to the Forward-Backward algorithm, but with a maximization step instead of summation over all possible hidden states. Since the Viterbi algorithm is not used in this thesis, no details are reviewed in this chapter. In fact, in this thesis, instead of finding the most probable hidden sequence, I sample from its distribution using the Forward-Backward algorithm as shown later in Chapter 4 (section 4.4).

3.2 The Hidden Markov Model of haplotype inference

In order to describe the hidden Markov model, key parameters and their notations will be introduced first. Observed genotypes (G) and unobserved haplotypes (H) are elements that have descended from ancestors. H is modeled using a set of ancestral haplotypes. Ancestral haplotype indexes (S) indicate from which ancestral haplotypes each individual inherited his (or her) genetic information. Present-day haplotypes are derived from the ancestral haplotypes through recombinations and mutations. Hence, recombination and mutation rates have been built into this model, with notations T and m, respectively. Since ancestral haplotypes may vary in frequencies of occurrence, another set of parameters Q has been introduced to denote the probability of randomly choosing an ancestral haplotype given that a recombination occurred between two consecutive loci. Finally, a genotyping error rate, e, allows allele differences between genotypes (G) and haplotypes (H). Details about the above parameters are given in Table 3.1.

Relationships between ancestral haplotypes and present-day haplotypes and genotypes are modeled using a Hidden Markov Model (HMM), as shown in Figure 3.1. In this Hidden Markov Model, Genotypes (G) are the only observed elements, and the goal of building this model is to infer the haplotypes H for each person. The structure in Figure 3.1 enables the joint probability of the model to be expressed as the product of several conditional probabilities. For example, genotyping error e influences only the relationship between G and H. Hence, the joint probability of all quantities (A, S, T, Q, m, H, G, e) in this Hidden Markov Model can be written as

$$\begin{aligned} Prob &= P(T)P(Q)P(m)P(A)P(S|T,Q)P(G|H,e)P(H|A,S,m) \\ &= P(T)P(Q)P(m)P(A)P(S|T,Q)\left[\prod_{p=1}^{P}\prod_{l=1}^{L}P\left(G_{p,l}|H_{l,p},e\right)\right]\times \\ &\qquad \left[\prod_{p=1}^{P}\prod_{h=1}^{2}\prod_{l=1}^{L}P\left(H_{h,l,p}|A_{(S_{h,l,p}),l},m\right)\right] \end{aligned} \tag{3.1}$$

To illustrate the hidden Markov model and the above joint probability, a toy example is given in Figure 3.2, in which the genotype data set consists of two people, each with six loci. There are three ancestral haplotypes ($ch1, ch2$ and $ch3$) in A. For person

G: $2 \times P \times L$ matrix of observed genotypes. $G_{p,l}$: the genotype pair of person p at locus l.
H: $2 \times P \times L$ array of unobserved haplotypes. $H_{h,l,p}$: the haplotype allele of person p at locus l, on chromosome h.
A: $C \times L$ matrix for C unobserved ancestral haplotypes. $A_{c,l}$: the allele of ancestral haplotype c at locus l.
S: $2 \times P \times L$ array for unobserved indexes of ancestral haplotypes. $S_{h,l,p}$: the ancestral haplotype index for haplotype h of person p at locus l.
T: a vector of $L - 1$ transition probabilities. T_l: probability of S staying with the same ancestral haplotype between locus l and $l + 1$.
Q: $C \times L$ matrix of probabilities for selecting different ancestral chromosomes when recombination occurs. $Q_{c,l}$: the probability of selecting ancestral haplotype c at locus l when recombination happens at l $(l > 1)$. $Q_{c,1}$: the probability of picking chromosome c at the first locus.
e: the genotyping error rate that explains $G - H$ inconsistencies.
m: the mutation rate, either a single number, or a vector of length L, or a matrix of size $C \times L$.

Table 3.1: Notations for key elements in the haplotype inference model. In total, there are P people, L loci and C ancestral haplotypes. The index h $(h = 1, 2)$ refers to the first and the second haplotype of each person, where parental origin order is arbitrary.

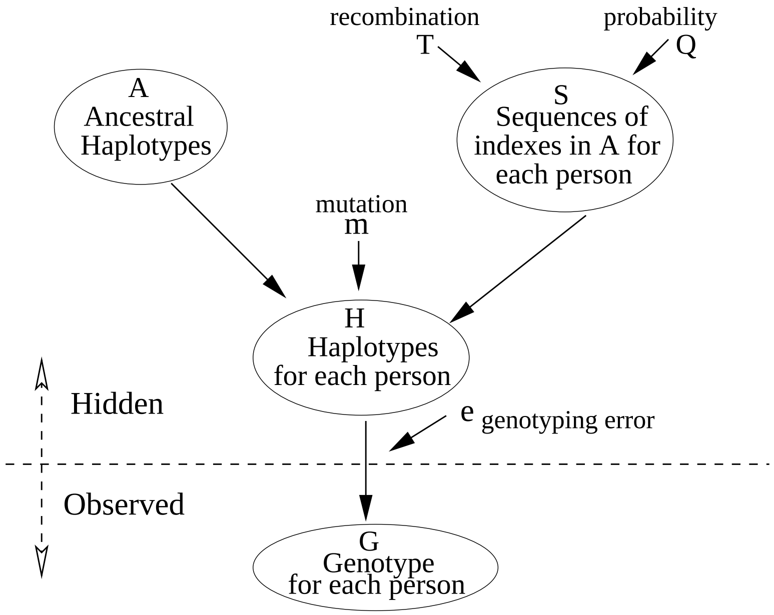


Figure 3.1: The Hidden Markov Model for haplotype inference

$p1$, the chromosome index S shows that two haplotypes are inherited entirely from ancestral chromosome $ch1$ and $ch2$, respectively, without recombination and mutation. For person $p2$, the first part of haplotype $h1$ is inherited from ancestral haplotype $ch2$, but a recombination occurs between locus 4 and 5, so the rest of this haplotype is inherited from ancestral haplotype $ch3$. The second haplotype $h2$ of person $p2$ is inherited from ancestral haplotype $ch1$, but, there is a mutation at the first locus changing the allele from 1 to 3. For the fifth locus of person p2, the haplotype pair is $(2, 2)$, but the genotype is $(4, 2)$ due to a genotyping error. $(0, 0)$ denotes a missing genotype. Note that allele differences between (A, S) and H can be explained by mutations. Differences between G and what is expected from (A, S) can be explained by either a mutation or a genotyping error.

It is worth emphasizing that in this model, the mutation and recombination parameters do not represent the commonly used definitions of these quantities. Mutation rates or recombination rates are usually defined as the rate of new mutation or recombinations in a single meiosis. However, the HMM parameters are designed to capture haplotype patterns that evolved over many generations. Similarly, the ancestral haplotypes are not necessarily the real haplotypes of ancestors of the set of individuals.

In this Hidden Markov Model shown in Figures 3.1 and 3.2, every element above the long dashed line is unobserved ("hidden"), and the genotypes (G) are the only observed elements. Haplotypes (H) are latent variables, and A, T, Q, m and e are unknown parameters. The hidden index sequences, (S), are the states of the hidden Markov chain. In order to do inference on haplotypes (H), a Bayesian approach has been used. Prior distributions are assigned to unknown parameters A, T, Q, and m. Currently the parameter e is fixed to be a constant. The posterior distributions of these unknown parameters are calculated using Bayes' theorem. The indexes S can be estimated using the Forward-Backward algorithm which will be explained in the next chapter. After obtaining the posterior estimates for all other parameters (A,S,T,Q,m), haplotypes (H) can be reconstructed; the estimation of H will also be explained in detail in the next chapter. In the following sections, I will present the models for A, T, Q, S, m, e, and explain how the model copes with missing genotypes.

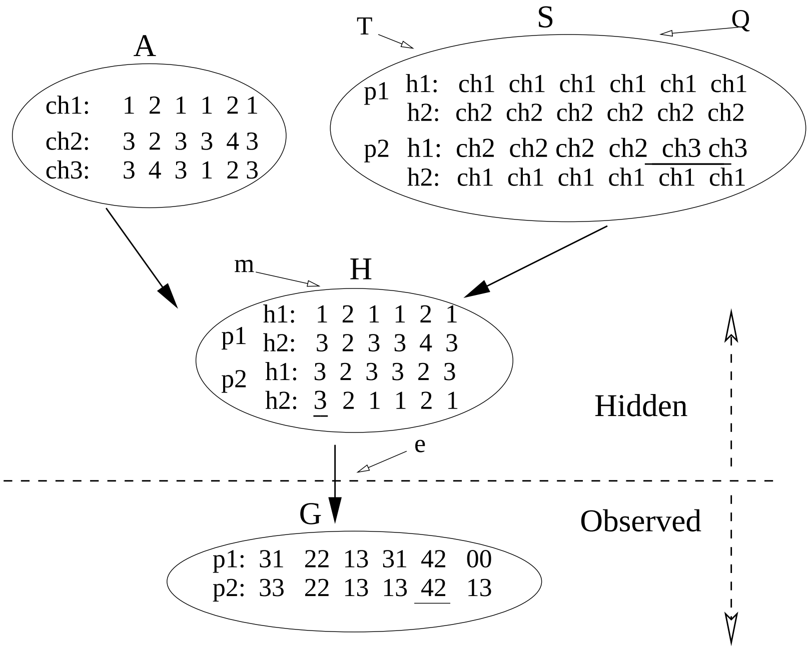


Figure 3.2: A toy example illustrating how today's haplotypes (H) and genotypes (G) were inherited from unobserved ancestors (A). See text for detailed descriptions.

3.3 The ancestral haplotypes, A

In my initial algorithm, a prior was assumed for ancestral haplotypes (A) in which each marker locus and each ancestral haplotype are modeled as independent. That is, $P(A) = \prod_{c=1}^{C} \prod_{l=1}^{L} P(A_{c,l})$, with $P(A_{c,l}) = 1/N_l$, where N_l is the number of possible alleles at locus l. This simple model works well for data sets with a small number of loci (e.g. less than 15) and low haplotype diversity. However, for a large number of markers with different levels of linkage disequilibrium (LD), this model may not always perform well. Therefore, in later work on SNP markers (with only 2 alleles), a high-order Markov model of order d was used as a prior distribution for A. This helps to account for linkage disequilibrium between loci. For an order d Markov model, the probability of A is then:

$$P(A) = P(A_1, \cdots, A_d) \prod_{l=d+1}^{L} P(A_l | A_{l-1}, \cdots, A_{l-d}) \tag{3.2}$$

There are 2^d possible allele combinations at d consecutive SNP markers. At locus l, let the probability of seeing allele 1, conditional on the i^{th} ($i = 1, \cdots, 2^d$) allele combination at the d previous loci, be p_i. The probability of allele 2 is $1 - p_i$. Each p_i is independently given a $Beta(a_1, b_1)$ prior distribution, with density denoted as $f(p_i)$. At each locus l, the observed counts of the 2^{d+1} possible ancestral haplotypes at the $d + 1$ loci ($l - d, \cdots, l$) are denoted as $x_1^1, x_1^2, x_2^1, x_2^2, \cdots, x_{2^d}^1, x_{2^d}^2$. Thus, we obtain

$$P(A_l | A_{l-1}, \cdots, A_{l-d}, p_1, \cdots, p_{2^d}) = p_1^{x_1^1}(1 - p_1)^{x_2^2}, \cdots, p_{2^d}^{x_{2^d}^1}(1 - p_{2^d})^{x_{2^d}^2} \tag{3.3}$$

After integrating out the p_i parameters, $i = 1, 2, \cdots, 2^d$, denote the second part of equation (3.2), $\prod_{l=d+1}^{L} P(A_l | A_{l-1}, \cdots, A_{l-d})$, as II. In fact,

$$\begin{aligned} II &= \prod_{l=d+1}^{L} \int_0^1 \cdots \int_0^1 f(p_1), \cdots, f(p_{2^d}) P(A_l | A_{(l-d):(l-1)}, p_1, \cdots, p_{2^d}) \\ &= \prod_{l=d+1}^{L} \left[\frac{\Gamma(x_1^1 + a_1)\Gamma(x_2^2 + b_1)}{\Gamma(x_1^1 + a_1 + x_1^2 + b_1)} \cdots \frac{\Gamma(x_{2^d}^1 + a_1)\Gamma(x_{2^d}^2 + b_1)}{\Gamma(x_{2^d}^1 + a_1 + x_{2^d}^2 + b_1)} \right] \end{aligned} \tag{3.4}$$

In the above formula, the probabilities for the first d loci have been separated out from the probabilities for loci $d + 1$ and higher. These loci can be incorporated into the same model by assuming the existence of hypothetical "previous" loci, $-(d - 1), ..., -1, 0$, where each locus carries the same allele, say all carry allele 1. For example, if d=4, when looking at loci $l = 1, 2, 3$, and 4, we assume the existence of loci $-3, -2, -1$

and 0 loci all carrying allele 1. With this assumption, the above formula (3.2) can be modified to:

$$P(A) \quad = \quad \prod_{l=1}^{L} \left[\frac{\Gamma(x_1^1 + a_1)\Gamma(x_2^2 + b_1)}{\Gamma(x_1^1 + a_1 + x_1^2 + b_1)} \cdots \frac{\Gamma(x_{2^d}^1 + a_1)\Gamma(x_{2^d}^2 + b_1)}{\Gamma(x_{2^d}^1 + a_1 + x_{2^d}^2 + b_1)} \right] \qquad (3.5)$$

The number of ancestral haplotypes, C, and the order of the Markov model d are fixed as constants. For a data set with a small number of markers and low haplotype diversity, $C = 4$ to 8 will suffice. For data sets with a large number of markers and diverse haplotypes, larger C values would likely give better performance. Values of d between 1 and 4 may be appropriate, depending on the degree of linkage disequilibrium (LD).

The prior for ancestral haplotypes A in Equation (3.5) is derived for single nucleotide polymorphism (SNP) markers only. However, it could be extended to microsatellite (multiallelic) markers as well.

3.4 The recombination parameter, T

When genetic information is passed on from one generation to another, two homologous chromosomes may recombine and exchange segments during the early stages of cell division. This is the definition of recombination over one generation. In this Hidden Markov Model, there are many generations between today's sampled individuals and their ancestors. A parameter, T, accounting for all recombinations over generations, has been used. Specifically, at locus l, let T_l be the probability that the ancestral haplotypes do not recombine. Between each locus l and $l + 1$, the total number of sampled ancestral haplotype indexes in S that retain the same ancestral haplotype as at locus l is assumed to follow a binomial distribution with probability T_l. T_l is given a conjugate prior $Beta(a, b)$, with a and b selected to make the prior of T favor values close to 1, since recombination is unlikely over short distances. Note that $1 - T_l$ can be interpreted loosely as the cumulative recombination rate over all the hypothetical generations between today and many generations ago. It should not be interpreted in the same way as the usual recombination fraction. Therefore, T_l is not necessarily constrained to be greater than 0.5, and can range between 0 and 1.

3.5 Ancestral haplotype probabilities, Q

When a recombination happens between locus $l - 1$ and l, the probabilities of selecting each ancestral haplotype at locus l is assumed to be $Q_l = (Q_{1,l}, \cdots, Q_{C,l})$. When an ancestral haplotype does recombine between $l - 1$ and l in S, the number of copies of ancestral haplotype c that are selected is assumed to be a random variable, $Y_{c,l}, (c = 1, \cdots, C)$. For each locus l, $(Y_{c,l}, \cdots, Y_{C,l})$ is assumed to follow a multinomial distribution with probabilities $Q_{1,l}, \cdots, Q_{C,l}$. These probabilities are given a Dirichlet prior which is the conjugate prior for the multinomial distribution. Specifically, Dirichlet $(1/C, \cdots, 1/C)$ is used as the prior for the C ancestral haplotypes in Q_l. At the first locus, there is no recombination involved. Let the number of each ancestral haplotype selected at the first locus also be a random variable $Y_{c,1}$ $(c = 1, \cdots, C)$. This vector of C variables is also assumed to follow a multinomial distribution with probabilities $Q_{1,1}, \cdots, Q_{C,1}$. Similar to the other loci, these probabilities are given the same Dirichlet prior. The Dirichlet distribution is a flexible choice for this prior since it permits multiple symmetric and asymmetric modes.

3.6 The hidden Markov chain, S

The hidden sequences (S) indicate from which ancestral haplotypes each individual inherits genetic information. For example, for the haplotypes of person p, suppose the two hidden sequences of S_p are as follows:

$S_{1,1:L,p}$: ch1 ch1 ch1 ch1 ch1 ch1 ch1 ch1

$S_{2,1:L,p}$: ch2 ch2 ch2 ch2 ch2 ch3 ch3 ch3

This means, for person p, the alleles on the first haplotype are inherited from ancestral haplotype $ch1$; the alleles on the second haplotype are inherited from ancestral haplotype $ch2$ for the first 5 markers, and the rest of the alleles are inherited from $ch3$. For each person, the hidden sequence $(S_{h,1:L,p}, h = 1, 2)$ forms a hidden Markov chain with the initial distribution $P(S_{h,1,p} = c) = Q_{c,1}$, $(h = 1, 2, c = 1, \cdots, C)$ and the transition probabilities given below:

$$P(S_{h,l,p} = ch_i \to S_{h,l+1,p} = ch_j \,|\, T_l, Q_{l+1}) = \begin{cases} T_l + (1 - T_l)Q_{ch_j,l+1} & \text{if } ch_i = ch_j \\ (1 - T_l)Q_{ch_j,l+1} & \text{if } ch_i \neq ch_j \end{cases} \tag{3.6}$$

Therefore, the conditional distribution of the entire set of sequences $(S|T, Q)$ is:

$$P(S|T, Q) \;=\; \prod_{p=1}^{P} \prod_{h=1}^{2} \left[P(S_{h,1,p}|Q_1) \prod_{l=1}^{L-1} P(S_{h,l,p} \to S_{h,l+1,p} \,|\, T_l, Q_{l+1}) \right]$$

3.7 The mutation rate, m

In a living cell, many changes or mutations may occur within the deoxyribonucleic acid (DNA), the genetic material of a gene. There are several different types of mutations such as single base substitutions (a base is replaced by another), insertions (a base is added to the DNA of a gene), deletions (a base is removed from the genetic material of a gene), duplications (a part of the genome is doubled) and translocations (a small piece of a chromosome has been transmitted to nonhomologous chromosome).

In this Hidden Markov Model, the possible mutation processes have been simplified to allele substitutions. In fact, the underlying model assumption is that, for any marker, there is a list of possible alleles. For SNPs, there are two possible alleles, and for microsatellite markers, there may be N_l ($N_l \geq 2$) alleles. For a given mutation rate m, a sampled haplotype allele $H_{h,l,p}$ may carry any of the possible alleles at locus l with the probabilities shown in the following model:

$$P(H_{h,l,p}|A_l, S_{h,l,p}, m) = \begin{cases} 1-m & \text{if } A_{S_{h,l,p},l} \text{ and } H_{h,l,p} \text{ are consistent} \\ \frac{m}{N_l-1} & \text{if } A_{S_{h,l,p},l} \text{ and } H_{h,l,p} \text{ are inconsistent} \end{cases} \tag{3.7}$$

The goal of introducing this parameter is simply to explain visible allele differences between $A_{S_{h,l,p},l}$ and $H_{h,l,p}$. Hence, this 'mutation' concept explains differences between A and H that may be due to a real mutation many generations ago, or may be due to other chromosomal rearrangements and recombinations. The parameter m can not be interpreted as per-meiosis mutation rate.

In the above formula, the mutation rate can be modeled in the following four different ways: (1) m is fixed as a constant, e.g. $m = 0.0005$, (2) m is a random variable for the whole data set, i.e. one mutation rate for all loci, (3) mutation rates vary by locus, that is, the mutation rate m is a vector of length L, $m = (m_1, m_2, \cdots, m_L)$, with m_l as the mutation rate for locus l. (4) mutation rates vary by ancestral haplotype and by locus, therefore, the mutation rate m is a $C \times L$ matrix $m = (m_{c,l})$ with $m_{c,l}$ being the mutation rate of ancestral haplotype c at locus l. To simplify the notation, for the options (2), (3) and (4), the following notations, 'm.one', 'm.l' and 'm.cl' are used respectively. For the 'm.one' model (option (2)), the overall mutation rate m is assigned a prior Beta(a_m, b_m). For the 'm.l' model (option (3)), this mutation rate is a vector of length L, i.e. a mutation rate is assigned to each locus. At each locus l, the mutation rate m_l is given a Beta(a_m, b_m) prior. Finally, for the 'm.cl' model (option (4)), at each locus $l, (l = 1, \cdots, L)$ of each ancestral haplotype $c, (c = 1, \cdots, C)$, the mutation rate $m_{c,l}$ is assumed to follow a Beta(a_m, b_m) prior.

In the '*m.l*' and '*m.cl*' models, the priors for the mutation rate of each locus and each ancestral haplotype are assumed to be the same. For a real data set, if information about mutation rates were known for different markers, then the Beta priors could be chosen to incorporate this information. The parameters a_m and b_m are chosen to favor values close to zero since mutations are relatively rare. Only the last three mutation models will be used, since the first one (with fixed mutation rate for all loci) is not a reasonable choice for real genetic studies.

3.8 The genotyping error rate, e

Although recombination and mutation can explain the connections between ancestral haplotypes and current haplotypes, any inconsistencies between haplotypes and genotypes can not be explained by these parameters. Such inconsistencies are deemed to be measurement errors in this HMM model. In fact, all large genotype data sets are likely to contain some genotype measurement errors (Sobel et al. 2002), with error rates depending on marker type and technology used. Several authors have mentioned that even a small genotyping error rate may have a large impact on the statistical genetic analysis. A few studies, like Buetow (1991) and Abecasis et al. (2001), have shown the impact of genotyping error rate on linkage analysis, and Kirk and Cardon (2002) and Quade et al. (2005) have studied the impact of genotyping error on haplotype frequency estimation and haplotype reconstruction. Therefore, the genotyping error is also incorporated in this hidden Markov model.

Before presenting the genotyping error model, let us briefly introduce different types of genotyping error. Generally speaking, there are two types of genotyping errors: Mendelian-consistent errors (genotypes are consistent with inheritance patterns) and Mendelian-inconsistent errors (genotypes are not consistent with inheritance patterns in families). These errors could be due to errors in allele calling, incorrect data entry, interpretation and so on. Sobel et al. (2002) summarized five types of genotyping errors: (1) missing an allele, (2) misreading an allele, (3) jointly misreading both alleles, (4) adding an allele and (5) pre-gel errors, and they produced an ‘empirical penetrance model' for the genotyping error in family studies. However, their method does not apply to data on unrelated samples (Kirk and Cardon 2002).

For unrelated individuals in the Hidden Markov Model, the inferred haplotype alleles are assumed to be the "true" genotypes. If the observed genotype is not consistent with the inferred haplotype alleles, this is considered to be a potential "genotyping error" and the following model (Sobel et al. 2002) can be used:

$$P(G_{l,p}|H_{l,p}, e) = \begin{cases} 1 - e & \text{if } G_{l,p} \text{ and } H_{l,p} \text{ are consistent} \\ \frac{e}{w-1} & \text{if } G_{l,p} \text{ and } H_{l,p} \text{ are inconsistent} \end{cases} \tag{3.8}$$

Here e is the genotyping error rate (currently fixed at $e = 0.001$), and w is the total number of all possible genotypes at locus l. If at locus l, there are N_l possible alleles, then $w = N_l + \binom{N_l}{2}$.

If more detailed information about a marker type and the technology used for genotyping at a specific locus is known, then the genotype error could be modeled in a more specific way. For example, if there is genotyping error at a SNP marker which is called as homozygous 'AA', then the true genotype is more likely to be 'AB' than 'BB'.

3.9 Dealing with missing data

Observing some missing genotypes is a common phenomenon in genetic studies, especially for large genotype collections, this is because obtaining more complete genotypes involves trade-offs between quality and quantity within certain budgets (Liu et al. 2006). Therefore, dealing with missing genotypes and estimating their effects are necessary in genetic studies, especially in haplotype estimation (Kelly et al. 2004) and haplotype analysis (Zhao et al. 2002, Becker and Knapp 2005, Liu et al. 2006). The usual way of handling missing genotypes for haplotype reconstruction is to assume explicitly or implicitly that they are missing at random (Excoffier and Slatkin 1995, Stephens et al. 2001, Kimmel and Shamir 2005, Rastas et al. 2005, Scheet and Stephens 2006).

In this Hidden Markov Model, the missing genotypes are also assumed to be missing at random. Generally, there are two types of missing genotypes. One type has only one allele missing, another type has both alleles missing. I treat these two cases as though the genotypes of both alleles are missing. For a specific $G_{p,l} = (0,0)$ (i.e. genotype missing), $P(G_{p,l}|H_{l,p}, e) = 1$, and then, $P(G_{p,l}|A_l, S_{l,p}, m) = 1$. When the genotype is missing, the alleles of this genotype are determined by the haplotype. The probability of $G_{p,l}$ given $(A_l, S_{l,p})$ is the sum over all possible haplotype allele pairs, so it is 1 as well. Therefore, the haplotype and genotype allele pair estimation will be based on the posterior samples obtained from the ancestral haplotype (A) and ancestral haplotype index (S), that is, the haplotype (genotype) allele pair is estimated based on the conditional probability $P(H_{l,p}|A_l, S_{l,p}, m)$.

Chapter 4

Markov Chain Monte Carlo methods

4.1 The overall algorithm

The joint probability of all quantities in the haplotype model of chapter 3 is:

$$\begin{aligned} Prob &= P(T)P(Q)P(m)P(A)P(S|T,Q)P(H|A,S,m)P(G|H,e) \\ &= P(T)P(Q)P(m)P(A)P(S|T,Q) \times \\ &\left[\prod_{p=1}^{P}\prod_{h=1}^{2}\prod_{l=1}^{L} P\left(H_{h,l,p}|A_{S_{h,l,p},l}, m_l\right)\right]\left[\prod_{p=1}^{P}\prod_{l=1}^{L} P\left(G_{p,l}|H_{l,p}, e\right)\right] \end{aligned}$$

It would be difficult to sample A, S, T, Q and m from the conditional distribution given G, which distribution is derived from the above joint probability directly. Instead, a Markov chain that will converge to the conditional distribution (given G) is defined. The Metropolis-Hastings sampler (Metropolis et al. 1953, Hastings 1970) and the Gibbs sampler (Geman and Geman 1984, Gelfand and Smith 1990) are two usual ways of defining such a Markov Chain. Let $Z = \{Z_1, \cdots, Z_n\}$ with a probability distribution $P(Z_1, Z_2, \cdots, Z_n)$. The general idea of the Metropolis-Hastings algorithm is to propose a change to the components of Z, acception or rejection is based on how will this affect the probability of the Markov chain state. The basic idea of the Gibbs sampler is to sample each element by conditioning on all the other elements. That is, sample Z_i^t from $P(Z_i^t | z_1^t, \cdots, z_{i-1}^t, z_{i+1}^{t-1}, \cdots, z_n^{t-1})$. In this thesis, A, T, Q and m are updated using the Gibbs sampler. To make the Markov chain converge faster, S is updated using the Forward-Backward algorithm.

At each step of the Markov chain, the simplest version of the overall algorithm is:

Step 1. Update ancestral haplotypes A using the Gibbs sampler to sample from the posterior distribution of each $A_{c,l}$ with all the other quantities fixed, where $c = 1, \cdots, C$ and $l = 1, \cdots, L$.

Step 2. Update S using the Forward-Backward algorithm.

Step 3. Update T and Q by sampling from $P(T_l | S_{l:(l+1)}, Q_{l+1})$, where $l = 1, \cdots, L-1$, and $P(Q_l | S_{l:(l+1)}, T_{l-1})$, where $l = 2, \cdots, L$, respectively. For Q_1, it is sampled from $P(Q_1 | S_1)$.

Step 4. Update m by sampling from $P(m | A, S, H)$.

In the following sections, I will explain how to use the Gibbs sampler to update each key parameter. To make the Markov chain that samples A, S, T, Q and m converge faster, haplotypes (H) are integrated out of the model, this will be discussed first in section 4.2. Modeling and updating A, S, T, Q, m will be presented in sections 4.3-

4.6. Estimating the prior parameters for recombination and mutation will be shown in section 4.7. Improvements for convergence will be discussed in 4.8. Methods for estimating haplotypes (H) for each individual will be described in section 4.8.

4.2 Summing over H

In the above Hidden Markov Model of Figure 3.1, the haplotypes of all people, H, are elements of the model that lie between (A, S) and G. Thus, on one side, (A, S) determines H; on the other side, H is closely related to G, since H determines the probability of G. In such situations, convergence of the Gibbs sampler may be very slow. Hence, haplotype parameters (H) have been integrated out by summing over all the possible N_l^2 allele combinations, where N_l is the total number of alleles at locus l. The following is the joint probability of the hidden Markov model after integrating out H:

$$\begin{aligned}
&P(A, S, G, T, Q, m, e) \\
&\quad = \sum_H P(A, S, H, G, T, Q, m, e) \\
&\quad = P(T)P(Q)P(A)P(S)P(m)\left[\sum_H P(H|A, S)P(G|H)\right] \\
&\quad = P(T)P(Q)P(A)P(S)P(m)\left[\sum_H \prod_{p,l} P(H_{l,p}|A_l, S_{l,p}, m)P(G_{l,p}|H_{l,p}, e)\right] \\
&\quad = P(T)P(Q)P(A)P(S)P(m) \times \\
&\qquad\qquad \left[\prod_{p,l} \sum_{H_{l,p}} P(H_{l,p}|A_l, S_{l,p}, m)P(G_{p,l}|H_{l,p}, e)\right] \qquad (4.1)
\end{aligned}$$

The order of the sum and product can be switched in the last line of Equation (4.1) because the P individuals are independent and the L loci are independent after conditioning on A and S.

Although estimating H is the primary goal of this thesis, the gain in speed of convergence by eliminating H from the likelihood is extremely beneficial. The parameters H are then estimated by sampling from the converged chain (sections 4.9, 4.10).

4.3 Updating A by summing over S

The set of ancestral haplotypes (A), is an essential element in this model. A is updated by integrating out all possible S, similar to the arguments in section 4.2, elimination of S from the likelihood can speed up convergence by improving the mixing of A in the Markov chain. For each element $A_{c,l}$, integrating out S implies summing over all $(C^2)^L$ possible values of S. This would be computationally infeasible, even for a data set with only a small number of markers. To overcome this computational difficulty, the Forward-Backward algorithm has been used. Let $f[v, l, p]$ be the probability, of summing over all possible ways of, reaching state $S_{l,p} = v = (ch_i, ch_j)$ at locus l of person p. For the first locus, $f[v = (ch_i, ch_j), 1, p] = Q_{ch_i,1}Q_{ch_j,1}$. This is the Forward step (Equation 4.2). Let $g[v, l, p]$ be the probability of obtaining all the states conditioning on $S_{l,p} = v = (ch_i, ch_j)$ after locus l. For the last locus, $g[v, L, p] = P(\text{no genotypes}|S_{l,p}) = 1$. This is the Backward step (Equation 4.3). (Note, a technique from Scheet and Stephens (2006) is adopted to reduce computation time by reducing the necessary summation from $O(C^4)$ to $O(C^2)$).

$$\begin{aligned}
f[v,l,p] &= \sum_{S_{1:(l-1),p}} P(G_{p,1:(l-1)}|A_{1:(l-1)}, S_{1:(l-1),p}P(S_{l,p}=v|S_{l-1,p}) \\
&= \sum_{w} f[w,l-1,p]P(S_{l,p}=v|S_{l-1,p}=w) \qquad (4.2)
\end{aligned}$$

$$\begin{aligned}
g[v,l,p] &= \sum_{S_{(l+1):L,p}} P(G_{p,(l+1):L}|A_{(l+1):L}, S_{(l+1):L,p}P(S_{l,p}=v, S_{(l+1):L,p}) \\
&= \sum_{u} g[u,l+1,p]P(S_{l+1,p}=u|S_{l,p}=v)P(G_{p,l+1}|A_{u,l+1}) \qquad (4.3)
\end{aligned}$$

In the above formulas, $\sum\limits_{u}$ (or $\sum\limits_{v}$, $\sum\limits_{w}$) means summing over all all those C^2 different ancestral haplotype combinations, that is $u = (ch_i, ch_j), 1 \leq i, j \leq C$. After summing over all possible S for each person, the joint probability of A and G given fixed T, Q, m and e is given in Equation 4.4. A toy example of using the Forward-Backward algorithm to sum over all possible S is given in Figure 4.1.

$$\begin{aligned}
P(A,G) &= P(A)\prod_{p=1}^{P} P(G_p|A) \\
&= P(A)\prod_{p=1}^{P}\left[\sum_{\text{all } S_p} P(G_p|A,S_p)P(S_p)\right] \\
&= P(A)\prod_{p=1}^{P}\left[\sum_{v} f[v,l,p]P(G_{p,l}|A_l, S_{l,p}=v)g[v,l,p]\right] \qquad (4.4)
\end{aligned}$$

The Gibbs sampler updates A based on the following conditional probabilities:

$$P(A_{c,l}|\text{other elements of A}, G) \propto P(G|A)P(A_{c,l}, \text{other elements of A}) \qquad (4.5)$$

The probability of A in th above Equation is calculated using Equation (3.5) of chapter 3; the probability of $(G|A) = \prod\limits_{p=1}^{P} P(G_p|A)$ is calculated as shown in Equation (4.4).

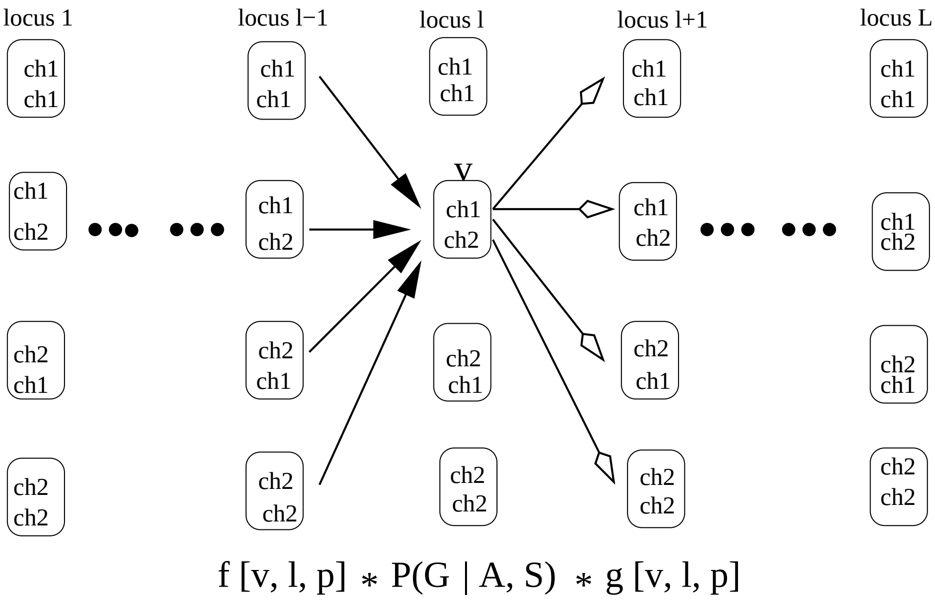


Figure 4.1: Forward-Backward algorithm for summing over all possible S for each person ($C = 2$). This plot is used to illustrate the summation term of Equation 4.4, $\left[\sum_v f[v,l,p]P(G_{p,l}|A_l, S_{l,p} = v)g[v,l,p]\right]$, with $v = (ch_i, ch_j)$ taking all C^2 cases ($1 \leq i, j \leq C$). Therefore, each column has C^2 'boxes' since both $f[,l,p]$ and $g[,l,p]$ is a function that consider all possible C^2 choices at each locus (locus $l-1$ for $f[,l,p]$ and locus $l+1$ for $g[,l,p]$).

4.4 Updating S using the Forward-Backward algorithm

Although updating A by integrating out S helps get an accurate estimate of A, S is the key element needed for estimating other parameters, such as T, Q and m. Therefore, S needs to be updated at each iteration. Originally, the Gibbs sampler was used to update each element of S, but the dependence between S_l and S_{l+1} makes convergence very slow when updating one element at a time. A variation of the Forward-Backward algorithm solves this problem by sampling a new sequence S_p for each person, independent of the previous sequence (Scott 2002).

The Forward-Backward algorithm consists the following steps: (1) The Forward step calculates the probability of producing the observed ancestral haplotype index (ch_i, ch_j) for person p at locus l by accumulating all the information before and up to locus l; (2) The Backward step samples $S_{l,p}$ for each locus, starting from the last locus L, based on the probabilities calculated in the Forward step and on the rules sampled for the later loci. For person p, at locus l, the formula for the Forward step is:

$$f[l, (ch_i, ch_j)] = P(G_{p,l} | A_{ch_i,l}, A_{ch_j,l}, S_{l,p} = (ch_i, ch_j), l) \times R$$

In the above formula, R of transitions is:

$$\sum_{x,y} P(S_{l,p} = (ch_i, ch_j) | S_{l-1,p} = (ch_x, ch_y)) f[l-1, (ch_i, ch_j)],$$

and the summation is over all C^2 possible $S_{l-1,p} = (ch_x, ch_y)$ at locus $l-1$, where $x = 1, \cdots, C$ and $y = 1, \cdots, C$. For the Backward step, once a state is chosen at locus $l+1$, for example, $S_{l+1,p} = (ch_i, ch_j)$, then the probability of reaching this state

(ch_i, ch_j) from state (ch_u, ch_v) at the previous locus l is:

$$b[l,(ch_u,ch_v)] = f[l,(ch_u,ch_v)]P(S_{l+1,p} = (ch_i,ch_j)|S_{l,p} = (ch_u,ch_v))$$

A toy example of updating S with C=2 using the Forward-Backward algorithm is given in Figures 4.2 and 4.3. In the Forward step (Figure 4.2), at each locus $l, (l = 1, 2, \cdots, L)$, there are $C^2 = 4$ possible states for $S_{l,p}$, and at each state (for example, state $(ch1, ch1)$ at locus 2), there are $C^2 = 4$ possible transitions (lines) that reach this state. The sum of the C^2 probabilities of these transitions is the probability of reaching each state at each locus. This calculation is done forward locus by locus until the last locus L. Figure 4.3 shows the backward step. Suppose the probabilities of obtaining each of those $C^2 = 4$ states at locus L are $0.1, 0.6, 0.2$ and 0.1, then in the

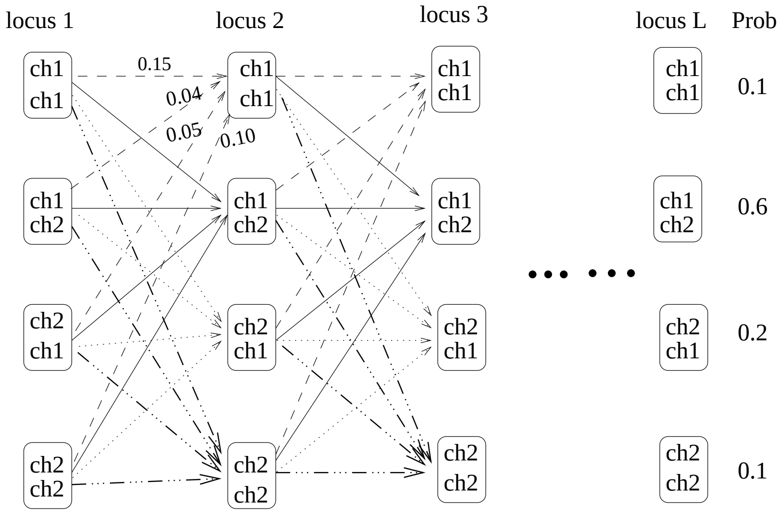


Figure 4.2: Forward-Backward sampling S_p- Forward step ($C = 2$). For example, the probability of reaching (ch_1, ch_1) at locus 2 is $0.15 + 0.04 + 0.05 + 0.10 = 0.34$. Further description is in the text.

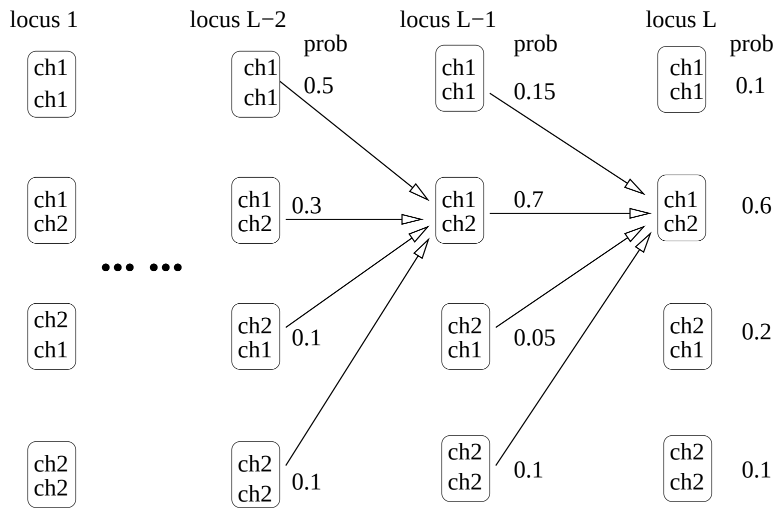


Figure 4.3: Forward-Backward sampling S_p- Backward step ($C = 2$). Suppose at the last locus (L), ancestral haplotype pair (ch_1, ch_2) are chosen with probability 0.6, then at previous locus ($L-1$), (ch_1, ch_2) is selected with probability 0.7. See text for further explanation.

Backward step (Figure 4.3), a state is sampled based on these probabilities. Suppose $(ch1, ch2)$ is selected. The probabilities of reaching this state from each of the C^2 states at the previous locus are calculated. For example, the probabilities of reaching this state from C^2 states at locus $L-1$ are 0.15, 0.7, 0.05 and 0.1, then at locus $L-1$ at state is sampled based on these probabilities. This process is repeated locus $L-2, L-3, \cdots, 1$, until a complete sequence of ancestral haplotype indexes S_p for person p is obtained.

4.5 Updating T and Q

After updating S, the parameters T and Q, which are closely related to S, can be estimated. For person p, let $S_{h,l,p} = c, S_{h,l+1,p} = c_1$, if $c = c_1$, there are two possible reasons:

(1) The ancestral chromosome "stays the same" as at locus l, i.e. there is no recombination of ancestral haplotypes.

(2) There is a recombination between ancestral chromosomes between locus l and $l+1$, and ancestral haplotype $c_1 = c$ is randomly selected with probability $Q_{c_1,l}$

Therefore, $P(S_{h,l,p} = S_{h,l+1,p} = c) = T_l + (1 - T_l)Q_{c,l+1}$, where T_l is the probability of staying with the same ancestral haplotypes as mentioned in chapter 4. To determine the number of ancestral chromosomes that do not recombine, an index I has been introduced for each h, l and p. That is,

$$I_{h,l,p} = \begin{cases} 0 \quad \text{if } S_{h,l,p} \neq S_{h,l+1,p} \\ 1 \quad \text{with probability } \frac{T_l}{T_l+(1-T_l)Q_{c,l+1}}, \text{ if } S_{h,l,p} = S_{h,l+1,p} = c \\ 0 \quad \text{with probability } \frac{(1-T_l)Q_{c,l+1}}{T_l+(1-T_l)Q_{c,l+1}}, \text{ if } S_{h,l,p} = S_{h,l+1,p} = c \end{cases} \cdot \tag{4.6}$$

For recombination parameters, the prior of T_l is $Beta(a, b)$. At a specific locus l, if the number of ancestral haplotypes that do not recombine between $l-1$ and l is X_l, where $X_l = \sum\limits_{h,p} I_{h,l,p}$, then the posterior distribution of $(T_l|S, Q)$ is $Beta(a + X_l, b + 2P - X_l)$.

At each locus, the prior of $(Q_{1,l}, \cdots, Q_{C,l})$ is assumed to be Dirichlet $(1/C, \cdots, 1/C)$. Let the number of each ancestral haplotype selected after a recombination be $Y_{c,l}, c =$

$1, \cdots, C$. Therefore, $Y_{c,l} = \sum\limits_{h,p}(1-I_{h,l,p})I(S_{h,l+1,p} = c)$, where $I(S_{h,l+1,p} = c)$ is an indicator function. Then the posterior distribution is Dirichlet $(1/C+Y_{1,l}, \cdots, 1/C+Y_{C,l})$.

4.6 Updating m

For the mutation rate, four models were presented in section 3.7. For the first model, the mutation rate is fixed as a constant, so there is no need to update it. For the second model (the 'm.one' model), a single mutation rate is used for all ancestral haplotypes and all loci. If the total number of inconsistent allele pairs between $(A_l, S_{h,l,p})$ and $H_{h,l,p}$ is Z, so that $Z = \sum\limits_{h,l,p} I(A_{c,l} \neq H_{h,l,p}, S_{h,l,p} = c)$, for $h = 1, 2, p = 1, \cdots, P$, and $l = 1, \cdots, L$, then the posterior distribution of m is Beta $(a_m + Z, b_m + 2PL - Z)$.

For the third model (the 'm.l' model), a vector $(m_1, \cdots, m_L)$ of mutation rates is used (i.e. one mutation rate for each locus). Similar to the second model, for locus l $(l = 1, \cdots, L)$, if the total number of inconsistent allele pairs between $(A_l, S_{h,l,p})$ and $H_{h,l,p}$ is Z_l, so that $Z_l = \sum\limits_{h,p} I(A_{c,l} \neq H_{h,l,p}, S_{h,l,p} = c)$, for $h = 1, 2, p = 1, \cdots, P$, then the posterior distribution of m_l is Beta $(a_m + Z_l, b_m + 2P - Z_l)$.

For the fourth model (the 'm.cl' model), there is a mutation rate for every locus of each ancestral haplotype. Let $Z_{c,l}$ be the total number of inconsistent allele pairs between $H_{h,l,p}$ and $A_{c,l}$, where $c = S_{h,l,p}$, then the posterior distribution of $m_{c,l}$ is Beta $(a_m + Z_{c,l}, b_m + U_c - Z_{c,l})$, where U_c is the number of times that ancestral haplotype c that has been used.

Note, in order to update m, H is sampled based on A, S and the previous mutation rate estimate in each iteration, even though H has been integrated out as mentioned in section 4.2.

4.7 Using hyperparameters

The recombination and mutation parameters are assigned Beta priors of different fixed parameters. Another option is to estimate the parameters of these Beta distributions by assigning them a hyperprior instead of using the fixed values. In particular, for recombination parameters, the prior of T_l is $Beta(a, b)$. The parameters (a, b) can be considered as variables and assigned a hyperprior for them. First, let $\phi = a/(a + b)$ and $\gamma = a + b$, then let $\phi \sim Beta(a_0, b_0)$ and $\gamma \sim uniform(u, v)$. The hyperprior (the joint distribution) for (ϕ, γ) is:

$$P(\phi,\gamma) = \frac{1}{v-u} I_{(u<\gamma<v)} \left[\frac{\Gamma(a_0+b_0)}{\Gamma(a_0)\Gamma(b_0)} \phi^{a_0-1}(1-\phi)^{b_0-1} \right]$$

Then the hyperprior of (a, b) is:

$$P(a,b) \quad = \quad \frac{1}{v-u} I_{(u<a+b<v)} \left[\frac{\Gamma(a_0+b_0)}{\Gamma(a_0)\Gamma(b_0)} \frac{a^{a_0-1}b^{b_0-1}}{(a+b)^{a_0+b_0-1}} \right]$$

After some simple calculations, the marginal posterior distribution of (a, b) is:

$$P(a,b|X) \propto P(a,b) \prod_{l=1}^{L-1} \frac{\Gamma(a+b)}{\Gamma(a)\Gamma(b)} \frac{\Gamma(a+X_l)\Gamma(b+2P-X_l)}{\Gamma(a+b+2P)}$$

In the above formula, $X = (X_1, \cdots, X_{L-1})$, and X_l is the number of ancestral haplotypes that do not recombine between l and $l + 1$ ($l = 1, \cdots, L - 1$). Variables (a, b) can be updated using the above posterior distributions. However, there is not a specifically analytic conditional probability function for either a or b, therefore, the

Slice sampling (Neal 2003) method can be used. The basic principle of this sampling is that in order to sample from the distribution of one variable, one can sample uniformly from the region under the plot of its density function.

For the 'm.l' model (i.e. one mutation rate for each locus), the prior distribution is $m_l \propto Beta(a_m, b_m)$. Similarly as the above, (a_m, b_m) can be treated as variables and assign a hyperprior to them by doing the similar variable transformation. That is, let $\phi = a_m/(a_m + b_m)$ and $\gamma = a_m + b_m$, then let $\phi \sim Beta(a_{m0}, b_{m0})$ and $\gamma \sim uniform(u, v)$. The hyperprior (the joint distribution) for (ϕ, γ) is:

$$f(\phi,\gamma) = \frac{1}{v-u} I_{(u<\gamma<v)} \left[\frac{\Gamma(a_{m0}+b_{m0})}{\Gamma(a_{m0})\Gamma(b_{m0})} \phi^{a_{m0}-1} (1-\phi)^{b_{m0}-1} \right]$$

Then the hyperprior of (a_m, b_m) is:

$$P(a_m, b_m) = \frac{1}{v-u} I_{(u<a_m+b_m<v)} \left[\frac{\Gamma(a_{m0}+b_{m0})}{\Gamma(a_{m0})\Gamma(b_{m0})} \frac{a^{a_{m0}-1} b^{b_{m0}-1}}{(a_m+b_m)^{a_{m0}+b_{m0}-1}} \right]$$

After some simple calculations, the marginal posterior distribution of

(a_m, b_m) is:

$$P(a_m, b_m|Z) \propto P(a_m, b_m) \prod_{l=1}^{L} \frac{\Gamma(a_m+b_m)}{\Gamma(a_m)\Gamma(b_m)} \times \frac{\Gamma(a+Z_l)\Gamma(b+2P-Z_l)}{\Gamma(a_m+b_m+2P)}$$

For the 'm.cl' model (i.e. one mutation rate for each locus of each ancestral haplotype), similar marginal posterior distribution of (a_m, b_m) can be obtained as follows,

$$P(a_m, b_m|Z, U) \propto P(a_m, b_m) \prod_{c=1}^{C} \prod_{l=1}^{L} \frac{\Gamma(a_m+b_m)}{\Gamma(a_m)\Gamma(b_m)} \frac{\Gamma(a+Z_{c,l})\Gamma(b+U_c-Z_{c,l})}{\Gamma(a_m+b_m+U_c)}$$

In the above formula, $Z = (Z_{c,l})$ and $Z_{c,l}$ is the total number of inconsistent allele pairs between $H_{h,l,p}$ and $A_{c,l}$, where $c = S_{h,l,p}$. $U = (U_c)$ with U_c denoting the number of times that ancestral haplotype c has been used.

Note that there is only one variable in the *m.one* model, no need to assign a hyperprior for this case. Therefore, no hyperparameter is used.

4.8 Estimating haplotypes H

Sampled haplotypes for each person can be obtained from the following probabilities:

$$P(H_{l,p}|A_l, S_{l,p}, G_{p,l}, m, e) = \frac{P\left(H_{l,p}|A, S, m\right) P\left(G_{p,l}|H_{l,p}, e\right)}{\sum\limits_{H_{l,p}} P\left(H_{l,p}|A, S, m\right) P\left(G_{p,l}|H_{l,p}, e\right)} \tag{4.7}$$

In the above formula, e is fixed at chosen value ($e = 0.001$). The mutation rate m is one mutation rate for all ancestral haplotypes and all loci. For different mutation models, the mutation rate term m would be specified as m_l and $m_{c,l}$ corresponding to third and the fourth mutation models. When the genotypes are missing, the above formula can be simplified to $P(H_{l,p}|A_l, S_{l,p}, G_{p,l}, m, e) = P\left(H_{l,p}|A, S, m\right)$.

Methods for reconstructing the best haplotypes are discussed in section 4.10. It is worth noting that e can be fixed to be a different value during haplotype reconstruction from the value used during MCMC estimation. For example, if e is set to zero during reconstruction, estimated haplotypes will be forced to match genotype data.

4.9 Improving and checking MC convergence

4.9.1 Improving the Markov chain convergence rate

In order to make the Markov chain converge faster and hence to get more accurate results, two additional improvements have been implemented in addition to integrating out H and S. They are (1) a swapping of parts of two randomly selected ancestral haplotypes to enable better mixing, and (2) improved sampling for the first locus of Q.

For improvement (1), the algorithm for swapping two randomly selected ancestral haplotypes is a Metropolis Hastings update. Specifically, this swap includes the following steps:

Step 1: At locus l, $l \geq 2$, two ancestral chromosomes $c1$ and $c2$ are randomly selected, then the first $l-1$ loci of A are exchanged to get $A1$. Meanwhile the corresponding part of Q is swapped to get $Q1$. Note that if the mutation model is the fourth model (one mutation rate for each locus of each ancestral haplotype), the first $l-1$ loci of the mutation matrix need to be swapped too.

Step 2: Calculate the joint probabilities before and after this change, and the ratio of these two probabilities:

$P_0 = P(A, G, Q, T, m, e) = P(T)P(Q)p(m)P(A, G|Q, T, m, e)$

$P_1 = P(A1, G, Q1, T, m, e) = P(T)P(Q1)p(m)P(A1, G|Q1, T, m, e)$

$P_1/P_0 = P(Q1)P(A1, G|Q1, T, m, e)/P(Q)P(A, G|Q, T, m, e)$

Note that the prior of Q is $Dirichlet(1/C, \cdots, 1/C)$, which is a symmetric distribution, therefore, $P_1/P_0 = P(A1, G|Q1, T, m, e)/P(A, G|Q1, T, m, e)$

Step 3: Determine whether to accept or reject this change by randomly sampling a number x from $unif(0, 1)$. The probability of acceptance is $P_{accept} = min(1, P_1/P_0)$. If $x < P_{accept}$, the change is accepted, otherwise, it is rejected.

For improvement (2), a Metropolis Hastings procedure was developed too. Two ancestral haplotypes $c1$ and $c2$ are randomly chosen, then $Q_{c1,1}$ and $Q_{c2,1}$ are resampled while keeping the sum of these two probabilities invariant. For example, suppose $Q_{c1,1} = 0.3$ and $Q_{c2,1} = 0.1$, then we might propose to change them to be: $Q_{c1,1} = 0.4 - x$ and $Q_{c2,1} = x$, where $0 < x < 0.4$. Similar to improvement (1), the joint probabilities of before and after this change and the ratio of these two probabilities are calculated:

$$P_0 = P(A, G, Q, T, m, e) = P(T)P(Q)p(m)P(A, G|Q, T, m, e)$$

$$P_1 = P(A, G, Q1, T, m, e) = P(T)P(Q1)p(m)P(A, G|Q1, T, m, e)$$

$$P_1/P_0 = P(Q1)P(A, G|Q1, T, m, e)/P(Q)P(A, G|Q, T, m, e)$$

Then the decision of acceptance or rejection is made using the strategy similar to the one used for swapping.

4.9.2 Checking the convergence of Markov chain

To check the convergence of the Markov chain, the joint likelihood and the recombination parameters T at each iteration are plotted to look for stable patterns. When using hyperparameters, the estimated hyperparameters of recombination and mutation can be plotted to check convergence too.

4.10 Summarizing the posterior distribution

Sampled haplotypes can be summarized by reporting the proportions of all different possible haplotype combinations in order to show the uncertainty of the haplotype distributions. These proportions can be obtained directly by counting the sampled haplotypes. Alternatively, one can find the best single estimate of the haplotypes for each person. Three different methods for obtaining this haplotype estimate are introduced in this section.

The first method is a straightforward one which uses the mode of the counts of all sampled haplotype pairs. For small numbers of markers, and for the sets of markers in strong LD, the diversity of haplotypes is low, this method is effective. However, for data sets with large numbers of markers or rarer haplotypes, it is likely that there will be many different haplotype pairs each with only a small probability of occurring. In this case, there may be no unique mode, or the mode may be a poor summary of the results.

The second method is based on the determination of the mother-father labels of each haplotype sample, so it is called 'label-method'. The key idea behind this method is that at convergence, each sampled haplotype H^r $(r = 1, \cdots, R)$ represents a possible true haplotype. First the algorithm is presented, and details about its interpretation follow afterwards. The label method can be implemented using the following four steps:

Step 1: By comparing to H^R (the last sampled haplotype), assign chromosome labels $h^r = 1, 2$ (or mother, father) to the two haplotypes of each H^r, where $r = 1, \cdots, R-1$. That is to choose the parental labels that best matches H^R, let us denote the parental labels as (h_1^r, h_2^r). The label choice of H^R is arbitrary at this point.

Step 2: Define the most commonly observed allele at each locus as the estimated haplotype allele at locus l of chromosome h, $\hat{H}_{h,l}$.

Step 3: Choose new labels h^r for H^r to minimize the following I_{label},

$$I_{label} = \sum_{r=1}^{R} \sum_{l=1}^{L} \delta(\hat{H}_l, H_l^r)$$

where δ is an indicator function.

Step 4: Repeat steps 2 and 3 until $\hat{H}$ is no longer changing.

Further explanation of this algorithm follows. Labels in step 1 are assigned to gain most similarity to haplotype pair H^R. For example, for the r^{th} sample, the label $h_1^r = 1$ (or h_1^r=mother) and $h_2^r = 2$ (or h_2^r=father) implies that H_1^r is more like H_1^R than H_2^R. If the label is $h_1^r = 2$ (or h_1^r=father) and $h_2^r = 1$ (or h_2^r=mother), then the similarity match is the other way around. The comparison is made simply by counting the number of matching alleles. The assignment of a parent (e.g. mother) to a haplotype is a convenient way of explaining results but has no relationship to which haplotype actually belongs to which parent. Without genotyping parents, the parent-of-origin is unknown.

The indicator $\delta(\hat{H}_l, H_l^r)$ is defined as follows. If the label is $h_1^r = 2$ (i.e. h_1^r=father) and $h_2^r = 1$ (i.e. h_2^r=mother), then $\delta(\hat{H}_l, H_l^r) = \delta(\hat{H}_{1,l}, H_{2,l}^r) + \delta(\hat{H}_{2,l}, H_{1,l}^r)$. If $\hat{H}_{1,l} = H_{2,l}$, $\delta(\hat{H}_{1,l}, H_{2,l}^r) = 1$, otherwise, it is 0. Similarly, if $\hat{H}_{2,l} = H_{1,l}$, $\delta(\hat{H}_{2,l}, H_{1,l}^r) = 1$, otherwise, it is 0. The parental labeling of H^r is switched to minimized I_{label}.

The third method, due to Scheet and Stephens (2006), is based on minimizing the switch distance. The switch distance (Lin et al. 2002) is the number of switches between consecutive heterozygous markers (in the inferred haplotypes) that are needed in order to recover correct haplotypes. In order to obtain the best haplotype estimate by minimizing the switch distance, first, all loci are classified into three different types:

heterozygous markers, homozygous markers and genotype-missing markers. For the markers with genotype missing, the haplotype estimates are only based on the sampled haplotypes obtained from A and S. For the heterozygous markers, first the sampled haplotypes (H) at each marker are checked to make sure the majority of sampled haplotype pairs are heterozygous, and if not, report a potential G-H inconsistence, i.e., a potential genotyping error. Similarly, for the homozygous type, check and report any potential genotyping error.

After finishing checking and reporting some potential genotyping errors, starting from one side of the genetic region, estimate the haplotype of two consecutive heterozygous markers based on the proportion of each haplotype combination in the sampled H. This minimizes the switch distance. For example, if l_1 and l_2 are two consecutive markers of person p with genotypes $G_{p,l_1} = 12$ and $G_{p,l_2} = 12$. Then there are two possible genotype allele combinations for these two markers: (1) hap 1: 1 2 and hap 2: 2 1; or (2) hap 3: 1 1 and hap 4: 2 2. If the count for (hap 1, hap 2) is 10, the count for (hap 3, hap 4) is 4, then the one (hap 1, hap 2) with higher counts is selected as the haplotype estimate for markers l_1 and l_2. Then consider the next two consecutive heterozygous marker l_2 and l_3, this process is repeated until the haplotypes of the last two consecutive heterozygous markers are obtained.

Heterozygous and homozygous markers that may have a genotyping error are reported. However, differently from the label-method, such G-H inconsistence loci are only reported, but they are not included in the estimated haplotype. For a specific locus (of some individual) that has G-H inconsistence, the haplotype alleles are just obtained by the random assignment of the genotypes at that locus. This will not affect the whole results significantly since only a very small proportion of markers have potential genotyping error.

In total, three methods for obtaining the best haplotype are presented in this section. Since the first counting-based method is not good for data sets with a large number of markers, only the last two methods are used in this thesis.

Chapter 5

Comparison with other HMM methods

I know of three other recently published haplotype inference methods that have used a Hidden Markov Model. They are Rastas et al. (2005) with software named HIT, Kimmel and Shamir (2005) with software named HINT, and Scheet and Stephens (2006) with software named fastPHASE. In this chapter I will compare these three methods to my approach, commenting on their common features, the main differences between my HMM method and the other three methods, and other common and different noteworthy aspects of these models.

5.1 Common features of the four HMM methods

Three main ideas are held in common among these four HMM methods. First, all methods introduce the concept of a set of ancestral haplotypes, even though this idea may be phrased differently. In HIT (Rastas et al. 2005) the term 'founder haplotypes' is used, in fastPHASE (Scheet and Stephens 2006), they are called 'clusters', and in HINT (Kimmel and Shamir 2005) the phrase 'blocky structure' is used. All methods assume that the current haplotypes are formed from a mosaic of ancestral haplotypes by allowing the founder haplotypes to 'recombine' with each other. The introduction of 'recombination' leads to the hidden Markov chain, where the 'recombination parameters' play the role of transition probabilities. The hidden Markov chain is formed by the sequence of ancestral haplotype states, which shows from which founder haplotypes the current haplotypes have 'inherited' the alleles at each locus. Between two consecutive markers, when a recombination occurs, the transition probabilities (recombination parameters) determine the exchanges between ancestral haplotypes. Note that even though all four methods have recombination parameters, there are some differences among them which will be described in section 5.3.

The second main common feature is that all these methods assume genotypes are missing at random. Missing genotypes are handled by summing over all possible genotypes in Rastas et al. (2005) and Kimmel and Shamir (2005). The method in this thesis and fastPHASE handle this by not only summing over all possible genotypes, but also summing over all possible ancestral haplotype states. In addition, since the fastPHASE algorithm focuses on imputation of missing genotypes, the imputation approach was further refined; this will be commented on in section 5.3.

The third common aspect is that all these four methods assume implicitly or explicitly haplotypes are in the Hardy-Weinberg Equilibrium (HWE), that is, two haplotypes of each individual are independent.

5.2 Main differences between my HMM method and other HMM methods

Important differences between my HMM method and the other three HMM methods are listed in Table 5.1. The most important difference is that my method is Bayesian-based. All the key parameters are treated as random variables and assigned appropriate prior distributions. For example, as mentioned in the previous chapter, a beta prior which favors values that are close to 1 was assigned to the recombination parameters, and a beta prior which favors values close to 0 was given to the mutation parameters. Inference in this thesis is based on the posterior distributions, that is, it is based on both the prior knowledge and the data. The other three methods are not Bayesian methods. They find maximum likelihood estimates using the EM algorithm. Inference using these methods is based only on the data set itself, and does not incorporate external knowledge through the priors.

A second main difference is that my HMM method uses a high order Markov model for A to account for linkage disequilibrium in the ancestral haplotypes. None of the other three methods has parameters or models to account for LD in ancestral haplotypes.

A third difference is that genotyping error was built into the hidden Markov model to explain the inconsistencies between the haplotype pairs and the genotype pairs. The other three methods were developed based on the assumption of no genotype measurement error.

Finally, the fourth main difference is that in this thesis, four different types of mutation models were developed, as described in Chapter 4, namely (1) a fixed constant, e.g. m=0.005, (2) the 'm.one' model, that is, one mutation rate for all ancestral haplotypes and all loci, (3) the 'm.l' model, that is, one mutation rate for each locus, that is, the mutation rate is a random vector of length L, (4) the 'm.cl' model, that is, one mutation rate for each ancestral haplotype and each locus, that is, a random matrix of C rows and L columns. The other three methods do not have any explicit mutation model built into their hidden Markov model. However, these three methods all used the ancestral haplotype allele frequencies $\theta_{c,l}$ as parameters. That is, they use

My HMM method	**Other three HMM methods**
Bayesian method	Non-Bayesian, EM algorithms
High order Markov model for A	No specific model for A
Genotyping error is built in	Assume no genotyping error
Four different mutation models	No explicit mutation model

Table 5.1: The main differences between my HMM method and the other HMM methods. For more details, see the text.

$\theta_{c,l} = P(A_{c,l} = \text{allele } 1)$. With $\theta_{c,l}$ included, the results of the other three methods should be comparable to my fourth model (i.e. the $m.cl$ model) mentioned here.

5.3 Other comments on the four HMM methods

In addition to the above major common and different features among the four HMM methods, there are a few other aspects, listed in Table 5.2, which are worth mentioning in this section.

Firstly, when there is a 'recombination' between two consecutive loci, in fastPHASE and my HMM method, the selection of a new ancestral haplotype state at the next locus, after a recombination, does not depend on the current state. However, in HIT and HINT, these transitions do depend on the current state. Therefore, between each two consecutive loci (l and $l+1$), HIT and HINT used $C \times (C-1)$ parameters, and in total, there are $C \times (C-1) \times (L-1)$ parameters. Whereas fastPHASE and my method used C parameters between two consecutive loci, with one parameter as the overall recombination parameter (T_l) and $C-1$ parameters for the probabilities of selecting C ancestral chromosome at locus $l+1$, in total there are $(L-1) \times C$ parameters.

Secondly, these four different methods have different focuses. The algorithm behind fastPHASE focuses on imputing missing genotypes, as well as inferring haplotypes, HINT focuses on disease risk association studies, HIT and my HMM method mainly focus on haplotype reconstruction. In addition, the recombination parameters of my HMM method can be used to identify regions with high or low recombination rates.

Thirdly, to improve convergence, HINT and HIT use some initialization tricks to make their EM algorithms converge faster, and fastPHASE uses an ‘averaging’ scheme by running their EM algorithm 20 times with different starting points. My method and fastPHASE use the Forward-Backward (FB) algorithm to sum over all possible states of ancestral haplotypes (that is to integrate out S), but HINT and HIT do not use the procedure of integrating out S. In my HMM method, several approaches are used to improve convergence. Haplotypes (H) are integrated out, swapping part of two ancestral haplotypes is implemented, and Q is updated at the first locus by summing over all possible ancestral haplotype states at locus 1.

Fourthly, HINT and HIT use the EM algorithm to estimate parameters and then use the Viterbi algorithm to reconstruct the haplotypes. In contrast, fastPHASE

	My HMM	**fastPHASE**	**HIT**	**HINT**
Transition	(L-1)C	(L-1)C	C(C-1)(L-1)	C(C-1)(L-1)
Focus of algorithm	HI	HI, IMG	HI	AS
Convergence tricks	Swap A Integrate S, H Update Q_1	Averaging integrate S, H	Initialization	Initialization
Estimation	Posterior	Monte-Carlo	Viterbi	Viterbi

Table 5.2: Additional features that differ among the four methods. The ‘transition’ row lists the number of transition parameters for each method. In the ‘focus of algorithm’ row, HI = ‘Haplotye Inference’, IMG = ‘Imputing Missing Genotypes’, and AS=‘Association Studies’. In the ‘convergence tricks’ row, ‘Initialization’ means HIT and HNT used some complex initialization tricks. For more details, see the text.

samples the haplotype pairs based on Monte Carlo methods. When fastPHASE runs the EM algorithm T times, the haplotype estimate is based on all T runs as well. For this thesis, the inference is based on sampling from the posterior distribution.

Finally, even though all methods assume genotypes are missing at random, and all four methods estimate the missing genotypes by summing over all possible genotypes, fastPHASE imputes the missing genotypes a bit differently. They run their EM algorithm T times, then the genotypes are imputed as those that maximize the average likelihood for the parameters of all the T runs. If T=1, then all these three methods impute missing genotypes in the same way.

Chapter 6

Results

In order to evaluate the performance of the HMM method, and to illustrate its different features, I analyzed publicly available data sets from Daly et al. (2001) and the HAPMAP project (The International HAPMAP Consortium (2003), (2004), (2005)) using the HMM-based method. These data sets all contain the genotypes of trio families, in which there are two parents and one child, so the estimated haplotypes of children can be compared to known haplotypes inferred from the parents' genotype information. I compare the results of the HMM method to the results obtained using the PHASE (Stephens et al. 2001, Stephens and Donnelly 2003, Stephens and Scheet 2005) and the fastPHASE (Scheet and Stephens 2006) programs.

6.1 Data sets

The first data set is from Daly et al. (2001), which I will refer to as the Daly data. It contains genotypes of 129 trio families. The parents and the child of each family were genotyped at 103 markers in a 500-kb region on human chromosome 5 in a study on Crohn disease. At the time of publication, these genotypes showed some novel patterns of genetic variation. Specifically, a small number of common haplotypes in small blocks were discovered among the haplotypes transmitted to the individuals with Crohn disease; there were 11 haplotype blocks identified in total, separated by regions showing more recombinations, see Appendix A for detailed information. Two to four different common haplotypes accounted for over 90% of all the observed chromosomes in each block, and each block spanned 100kb or less. Time has proven that this data exhibited low haplotype diversity and particularly strong block structure.

The CEU families (Utah residents with ancestry from northern and western Europe) and YRI families (Yoruba in Ibadan, Nigeria) are two of the data sets used in the HAPMAP project. For the CEU data, I selected genotypes of 100 contiguous markers in the ENCODE region 7p15.2. Most SNPs in the ENCODE region are rare and only 54% of them have minor allele frequency (MAF) ≥ 0.05. Therefore, in order to restrict analysis to markers with only common alleles, 56 markers (with lowest MAF=0.083) were selected for haplotype inference. The physical distance spanned by those 56 markers is approximately 100 kb (from 26700834 bp to 26804065 bp). For the YRI data, I selected 199 markers from 37128000 bp to 37272000 bp of chromosome region 19q13, a region highlighted for recombination hotspots (The International HAPMAP Consortium 2005). In particular, there are two recombination hotspots identified by the The International HAPMAP Consortium (hapmap.org) in the region I selected. One is from 37128001bp to 37137001bp, with hotspot center at 37134001bp, and another from 37262001bp to 37272001bp, with hotspot center at 37264001bp. The beginning, center and end of each recombination hotspot will be shown in later Figures. Similar to the CEU data, only 91 markers (with lowest MAF=0.15) have been analyzed for the haplotype inference, again, with the goal of restricting analysis to SNPs with common alleles.

6.2 Evaluation of the HMM method performance

6.2.1 Evaluation criterion

So that the performance of the HMM-based method can be evaluated and compared with other algorithms, only the children's haplotypes in the trio family data sets are estimated, without using parental genotypes. The haplotypes inferred for the children can then be compared with the haplotypes inferred by the software MERLIN (Abecasis et al. 2002), which reconstructs haplotypes based on the genotypes of the parents. Before comparing results with other programs, I first check the performance of my algorithm for different parameter settings (described in 6.2.2). Then I compare with other programs using both the best single estimate of the haplotypes (the 'mode' of the haplotype distributions) in section 6.2.3 and the distribution of posterior samples in section 6.2.4.

The best single estimates of my HMM method are obtained by using the label method and the minimizing switch distance method. (These two methods were introduced in section 4.10.) For the best single estimates of each method, two criteria have been used to do the comparisons. First, for long sequences, moving ‘windows’ are used to compare haplotypes inferred from the HMM method, PHASE and fastPHASE with MERLIN estimates. The i^{th} window is the sequence from the i^{th} locus to the $(i+s-1)^{th}$ locus, where s represents the window size; $s=5$ and $s=L$ are used below. When s=5, the 1st window will include loci 1-5, the 2nd window will include loci 2-6, the 3rd window will include loci 3-7, and so on. When s=L, there is just one window which includes loci $1-L$, that is the whole sequence. In each window, the number of incorrect haplotype pairs is counted. Second, the switch distance criterion (Lin et al. 2002) is used. Switch distance is the count (or proportion) of haplotype switches between all consecutive heterozygous markers needed to recover the correct haplotypes.

In addition to the above criteria, the number of markers with the ‘wrong’ genotype inference has been counted and denoted as mis.G. Note that even though the number in ‘mis.G’ is called ‘wrong’ genotype inferences, it might be that the recorded genotypes are incorrect. This is because my HMM method incorporates genotyping error, and in real genetic studies, it is quite possible that genotyping error exists. For missing genotypes, mis.M denotes the number of genotype pairs inferred from my HMM method, the PHASE and fastPHASE programs that are not consistent with the ones inferred from MERLIN.

Due to the incorporation of genotyping error, when using the label method to obtain the best single estimate, the s=5 and s=L column counts include the ‘G-H inconsistent’ loci. For example, in the first row of the Daly data results as shown in

Appendix C, the ‘Not using hyperparameters” results are mis.G=3 and ‘s=L count’ = 205. If those mis.G=3 ‘G-H inconsistent’ loci are not included, ‘s=L count’ is 202 instead of 205. When counting the switch distance, those loci with the ‘G-H inconsistencies’ are ignored, since it is hard to define the switch distance in this context. Then at the end, some penalized error count is added to the final switch distance count by adding one switch distance for each G-H inconsistent locus. Similar to the above example, without adding the penalty, the switch distance count would be sw=100 instead of 103. This penalized count is only for the label method. For the minimizing switch distance method, the potential G-H inconsistent loci are only reported, but are not included in the final haplotype estimate as described in section 4.10.

6.2.2 Performance of the HMM method as a function of chosen parameters and models

As described in Chapter 4, several parameters play very important roles in this Hidden Markov Model. For example, the number of ancestral chromosomes, C, and the linkage disequilibrium modeling parameter, d, influence the model’s flexibility. Furthermore, there are several models for the mutation parameter, and for the recombination and mutation parameters. Also, one can choose whether to use hyperparameters or not. In order to check which values and which models perform better, and whether the optimal choices vary for the three data sets, the program has been run with 36 different parameter combinations for each data set by choosing 3 levels for C ($C = 5, 10, 15$), 2 values for d ($d = 0, 3$), 3 mutation models ($m.one$, $m.l$ and $m.cl$ denote the three mutation models respectively), and 2 choices of whether to use hyperparameters or not. For each of these 36 parameter combinations, the HMM program was run with 3 different random seeds. For each seed, 600 MCMC iterations were done. The best single estimate was based on every 4th of the last 300 iterations. Two different ways of finding the best single estimate for a run were used, namely, the label method and the minimizing switch distance method. The results are organized into tables in Appendix C, and Appendix B gives the values of parameters that are fixed at the same values for all runs.

For results obtained from the models without using hyperparameters, an analysis of variance (ANOVA) is performed to look for the effects of C, d, mutation model (m), and best-single-estimate-method (*single*). Each quantity is treated as a main effect factor. The ANOVAs are done for each of the three data sets (Daly, CEU and YRI) by fitting the following model:

$$\text{measure} = C + d + m + single + C * d + C * m + d * m + \epsilon \qquad (*)$$

In this model, 'single' refers to the two different methods for estimating the best haplotypes ('label' means the label method, 'sw' means the minimizing switch distance method). Three error counts, the window length s=5 error (s5) count, the window length s=L error (sL) count and the switch distance (sw) count, are used as the 'measure' response, separately. For the models using hyperparameters, the same ANOVA models are separately fit, again for each of the three data sets.

Small p-values (less than or equal to 0.05) from the above ANOVAs are listed in the following tables: Table 6.1 for the Daly data; Table 6.4 for the CEU data; Table 6.7

Factors	N-s5	N-sL	N-sw	H-s5	H-sL	H-sw
C	<0.0001	<0.0001	<0.0001	0.0002	<0.0001	0.0005
d			0.014			
m	<0.0001	<0.0001	<0.0001	<0.0001	0.0003	<0.0001
C*d						0.022
C*m			0.007			0.031
m*d						
Single	<0.0001	0.0027	<0.0001	<0.0001		<0.0001

Table 6.1: Small p-values (less than or equal to 0.05) from the ANOVA of model $(*)$ for the Daly data. Columns with names 'N-s5', 'N-sL' and 'N-sw' are p-values obtained from the ANOVA done for the results without using hyperparameters; columns with names 'H-s5', 'H-sL' and 'H-sw' are p-values obtained from the ANOVA done for the results from model using hyperparameters.

for the YRI data. In these tables, there are a few small p-values corresponding to the interaction effects. In order to explore the interactions between different factors and see which parameter combination can produce better results, some interaction plots are made: Figures 6.1, 6.2 and 6.3 for the Daly, CEU and YRI data, respectively.

The cell means of different levels of each main effect (C, d, m, $single$) are calculated and the pair comparisons between any two levels of each factor are done using t tests. Results of cell mean calculation, and pairwise comparison p-values are listed in the following tables: Tables 6.2 and 6.3 for the Daly data; 6.5 and 6.6 for the CEU data; 6.8 and 6.9 for the YRI data.

The Daly data. When not using hyperparameters (Table 6.1 columns N-s5, N-sL and N-sw), all three measures (s5, sL and sw) consistently show that the choices of C and of the mutation model have significant effects. Moreover, Table 6.2 shows that the best choices are C=10 or 15 and the mutation model ‘m.cl’. In addition, $d = 3$ is a better choice for all three measures even though the effect is not significant. There are not many significant interaction effects except for some interactions between C and the mutation model (p=0.007) when the measure is sw. The interaction effect of C*m is presented in the first plot of Figure 6.1 (Daly-A). This plot shows that among those (9 choose 2 =36) pair-wise C*m level contrasts comparisons, the ones with mutation model ‘m.cl’ are much better than the ones with the other two mutation models, and the one with C=10 or C=15 are much better than the one with C5. The combination of C15 and m.cl produces the best results.

The results obtained using hyperparameters (Table 6.1, columns H-s5, H-sL and H-sw) also show that the factors C and m have significant effect. When the measure is sw (the H-sw column), there is some evidence of interaction between C and d (p=0.022), and between C and m (p=0.031). The relevant interaction effect plots are presented in the second and the third plot of Figure 6.1 (Daly-B and Daly-C). Plot Daly-B shows that $m.cl$ is the best for all three C levels, but that C=15 produces the best results. Plot Daly-C shows that d=3 produces better results with C=10 or 15. There are no obvious large differences between C=10 and C=15, but the C=15 results are a bit better.

	N-s5	p-values	N-sL	p-values	N-sw	p-values
C5	154.6	5 v10:<0.0001	217.1	5 v10:<0.0001	96.1	5 v10:<0.000
C10	134.6	5 v15:<0.0001	190.9	5 v15:<0.0001	86.8	5 v15:<0.0001
C15	136.1	(C10<C15)	187.8	(C15<C10)	84.9	(C15<C10)
d0	143.4		201.6		91.0	0.014
d3	140.1	(d3)	195.7	(d3)	87.5	(d3)
m.one	154.9	one v cl: <0.0001	206.9	one v cl: 0.0002	92.1	one v cl:<0.0001
m.l	152.9	l v cl:<0.0001	199.0		93.4	l v cl: <0.0001
m.cl	117.3	(m.cl)	189.0	(m.cl)	82.4	(m.cl)
label	152.9	<0.0001	193.5	0.0027	92.5	<0.0001
sw	130.5	(sw)	203.7	(label)	86.0	(sw)

Table 6.2: Summary of the Daly data results in Appendix C by calculating the cell means and p-values of pairwise comparisons of different levels for each factor. Columns with names 'N-s5', 'N-sL' and 'N-sw' are the cell means for each main effect when not using hyperparameters, these cell means are calculated for three measures (s5, sL and sw). Columns named 'p-values' are the pairwise comparison p-values for each main effect. Only the p-values that are less than 0.05 are presented in the table. Items listed inside ( ), for example, (m.cl), and (d), are the levels of some factors that can produce better results. In particular, if there are two terms, say, ($C10 < C15$), this means, C10 and C15 are significantly better than C5, but there are no significant difference between C10 and C15 even though C10 seems to produce better results (smaller error counts).

	H-s5	p-values	H-sL	p-values	H-sw	p-values
C5	168.0	5 v10:0.0008	209.8	5 v10: 0.0353	95.6	5 v10:0.005
C10	150.0	5 v15:0.0012	199.1	5 v15:<0.0001	88.9	5 v15:0.000
C15	151.4	(C10<C15)	189.6	(C15)	87.7	(C15<C10)
d0	154.2		200.5		91.0	
d3	159.3	(d0)	198.5	(d3)	90.5	(d3)
m.one	162.2	one v cl:<0.0001	194.0	one v l:0.0009	91.6	one v cl:0.0057
m.l	168.0	l v cl:<0.0001	209.9	cl v l:0.0015	95.5	l v cl:<0.0001
m.cl	139.9	(m.cl)	194.6	(m)	85.0	(m.cl)
label	174.0	<0.0001	200.1		94.6	<0.0001
sw	139.5	(sw)	198.9	(sw)	86.9	(sw)

Table 6.3: Summary of the Daly Data results in Appendix C by calculating the cell means and p-values of pairwise comparisons of different levels for each factor. Columns with names ‘H-s5’, ‘H-sL’ and ‘H-sw’ are the cell means for each main effect when using hyperparameters, these cell means are calculated for three measures (s5, sL and sw). Columns named ‘p-values’ are the pairwise comparison p-values for each main effect. Only the p-values that are less than 0.05 are presented in the table. Items listed inside ( ), for example, (m.cl), and (d), are the levels of some factors that can produce better results. In particular, if there are two terms, say, $(C10 < C15)$, this means, C10 and C15 are significantly better than C5, but there are no significant difference between C10 and C15 even though C10 seems to produce better results (smaller error counts).

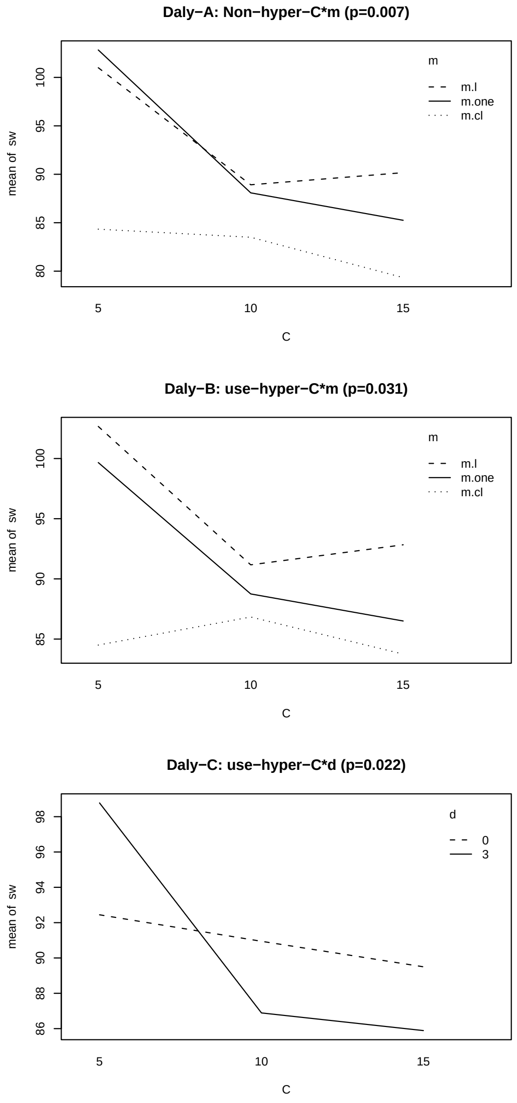


Figure 6.1: The interaction plots for the Daly data, showing mean switch distance scores for significant interaction effects in Table 6.1. The title of the first plot is “Daly-A: non hyper-C*m (p=0.007)”, this means that this plot is the interaction plot of C*m for the Daly data when not using hyperparameters, the p-value for the this interaction effect is 0.007 as listed in the 6th row and the 4th column of Table 6.1. Plot Daly-B and Daly-C have the similar meaning.

Factors	N-s5	N-sL	N-sw	H-s5	H-sL	H-sw
C		0.014		0.006	<0.0001	0.001
d				0.006	0.0003	0.0001
m	<0.0001	<0.0001	<0.0001	<0.0001	<0.0001	<0.0001
C*d				<0.0001	0.047	0.001
C*m		0.018		0.004	0.045	0.003
m*d						0.018
Single				<0.0001		

Table 6.4: Small p-values (less than or equal to 0.05) from the ANOVA of model $(*)$ for the CEU data. Columns with names 'N-s5', 'N-sL' and 'N-sw' are p-values obtained from the ANOVA done for the results without using hyperparameters; columns with names 'H-s5', 'H-sL' and 'H-sw' are p-values obtained from the ANOVA done for the results from model using hyperparameters.

The CEU data. When not using hyperparameters (Table 6.4, columns N-s5, N-sL and N-sw), the factor that matters most for all three measures is the mutation model. Similar to the Daly data, it is the *m.cl* model that produces the best results. There is an interaction effect between C and m when the measure is sL (p=0.018); in particular, the combination of C5 and *m.one* did not produce good results. When the mutation model is *m.cl*, there is no obvious difference between the results of C=5, 10 and 15. Overall, because the error count of this CEU data is very small, and the sample size is very small (only 30 individuals), this interaction effect may not be very important and no interaction plot is presented. The C factor only matters for the sL measure (p=0.014), with C=10 and C=15 producing better results than C=5 as shown in Table 6.5. There is no significant difference between C=10 and C=15. Overall, Table 6.5 suggests that the combination of C=10 and *m.cl* produces the best results.

When using hyperparameters, Table 6.4 (columns H-s5, H-sL and H-sw) shows that all three factors have significant effects, and that there are significant interactions between them. Since the interaction patterns of three measures (s5, sL, sw) are very similar, only the interaction plot of the sw measure is presented in Figure 6.2. These plots show that the *m.cl* model is significantly better, and that d=3 is better with *m.cl*. Table 6.6 and Figure 6.2 show that the combination of C=10, m.cl model and d=3 produces the best results.

	N-s5	p-values	N-sL	p-values	N-sw	p-values
C5	6.9		3.1	5 v10: 0.013	3.3	
C10	6.0		2.0		3.1	
C15	6.4	(C10)	2.3	(C10)	3.2	(C10)
d0	6.3		2.5		3.2	
d3	6.5		2.5		3.2	
m.one	5.2	all <0.0001	2.6	one v cl: 0.0023	2.3	all <0.0001
m.l	13.8		3.7	one v l : 0.011	6.0	(m.cl)
m.cl	0.3	(m.cl)	1.2	l v cl: <0.0001	1.1	
				(m.cl)		
label	6.6	<0.0001	2.6		3.3	<0.0001
sw	6.3	(sw)	2.4	(sw)	3.1	(sw)

Table 6.5: Summary of the CEU data results in Appendix C by calculating the cell means and p-values of pairwise comparisons of different levels for each factor. Columns with names 'N-s5', 'N-sL' and 'N-sw' are the cell means for each main effect when not using hyperparameters, these cell means are calculated for three measures (s5, sL and sw). Columns named 'p-values' are the pairwise comparison p-values for each main effect. Only the p-values that are less than 0.05 are presented in the table. Items listed inside ( ), for example, (m.cl), and (d), are the levels of some factors that can produce better results. In particular, if there are two terms, say, ($C10 < C15$), this means, C10 and C15 are significantly better than C5, but there are no significant difference between C10 and C15 even though C10 seems to produce better results (smaller error counts).

	H-s5	p-values	H-sL	p-values	H-sw	p-values
C5	12.8	5 v 10: 0.017	3.5	5 v 10:<0.0001	5.1	5 v 10:0.0018
C10	9.9	5 v 15: 0.013	2.1	5 v 15: 0.0048	3.8	5 v 15:0.0091
C15	9.8	(C10>C15)	2.5	(C10<C15)	3.9	(C10<C15)
d0	12.0	0.0061	3.2	0.0003	4.9	0.0001
d3	9.7	(d3)	2.2	(d3)	3.6	(d3)
m.one	11.6	all <0.0001	2.9	one v cl:<0.0001	4.5	all <0.0001
m.l	16.1		3.7	l v cl :<0.0001	6.3	
m.cl	4.9		1.6	l v one: 0.029	2.0	
		(m.cl)		(m.cl)		(m.cl)
label	13.3	<0.0001	3.0		4.5	<0.0001
sw	8.4	(sw)	2.5	(sw)	4.0	(sw)

Table 6.6: Summary of the CEU Data results in Appendix C by calculating the cell means and p-values of pairwise comparisons of different levels for each factor. Columns with names 'H-s5', 'H-sL' and 'H-sw' are the cell means for each main effect when using hyperparameters, these cell means are calculated for three measures (s5, sL and sw). Columns named 'p-values' are the pairwise comparison p-values for each main effect. Only the p-values that are less than 0.05 are presented in the table. Items listed inside ( ), for example, (m.cl), and (d), are the levels of some factors that can produce better results. In particular, if there are two terms, say, ($C10 < C15$), this means, C10 and C15 are significantly better than C5, but there are no significant difference between C10 and C15 even though C10 seems to produce better results (smaller error counts).

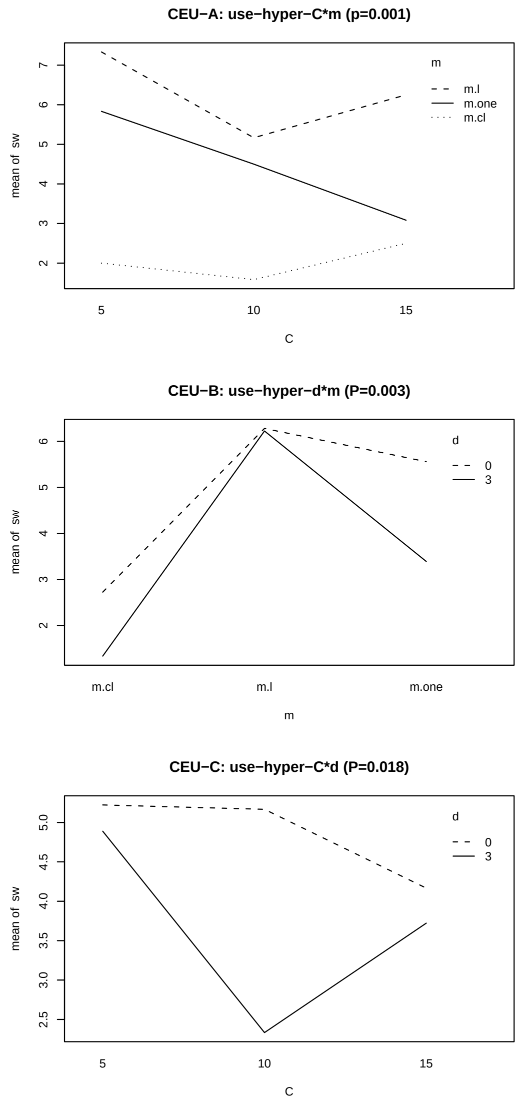


Figure 6.2: The interaction plots for the CEU data, showing mean switch distance scores for significant interaction effects in Table 6.4. The title of the first plot is "CEU-A: use hyper-C*m (p=0.001)", this means that this plot is the interaction plot of C*m for the CEU data when using hyperparameters, the p-value for the this interaction effect is 0.001 as listed in the 5th row and the 7th column of Table 6.4. Plot CEU-B and CEU-C have the similar meaning.

The YRI data. When not using the hyperparameters (Table 6.7, columns N-s5, N-sL and N-sw), C and m are significant main effects. Table 6.8 shows that C=10 or C=15 produces better results, and that the *m.one* model produces best results. Table 6.7 shows that there are three significant interaction effects. Two are for the sL measure, one is between C and m (p=0.0178), another is between m and d (p=0.0021). The third one is for the sw measure with the p value 0.035 between C and m. The interaction plot YRI-A (the first plot in Figure 6.3) shows that for all three C levels, the worst mutation model is m.cl. The $m.l$ model is good for C=5 and 10, but when C=15, m.one model is better. Plot YRI-A shows that the best combination is C10 and the model 'm.l'. Plot YRI-B in Figure 6.3 is the interaction plot for m and d, it shows that the best combination is m.l and d=3. Considering both YRI-A and YRI-B plot, the best combination for the sL measure is $C = 10, d = 3$, and the $m.l$ model. For the sw measure, there are some interaction effects (p=0.035), plot YRI-C shows

Factors	N-s5	N-sL	N-sw	H-s5	H-sL	H-sw
C	<0.0001	0.0117	0.011	0.001	0.0049	0.0007
d			0.017		0.031	
m	<0.0001	0.0003	<0.0001			
C*d						
C*m		0.0178	0.035	0.024		0.016
m*d		0.0021				
Single	<0.0001		<0.009	<0.0001		<0.0001

Table 6.7: Small p-values (less than or equal to 0.05) from the ANOVA of model (∗) for the YRI data. Columns with names 'N-s5', 'N-sL' and 'N-sw' are p-values obtained from the ANOVA done for the results without using hyperparameters; columns with names 'H-s5', 'H-sL' and 'H-sw' are p-values obtained from the ANOVA done for the results from model using hyperparameters.

	N-s5	p-values	N-sL	p-values	N-sw	p-values
C5	79.2	5 v10:<0.0001	107.1	5 v10:0.008	46.8	5 v10:0.0486
C10	67.2	5 v15: 0.008	96.3		44.3	5 v 15: 0.014
C15	70.2	(C10<C15)	101.7	(C10)	43.8	(C15<C10)
d0	72.9		103.5		46.0	
d3	71.4	(d3)	99.9	(d3)	43.9	(d3)
m.one	58.0	one v cl:< 0.0001	99.1	one v cl:0.0063	41.1	one v cl:< 0.0001
m.l	78.4	one v l:< 0.0001	95.9	l v cl:0.0003	46.8	one v l:< 0.0001
m.cl	80.2	(m.one)	110.3	(m.l<m.one)	47.0	(m.one)
label	77.6	<0.0001	100.4		46.1	0.009
sw	66.8	(sw)	103.1	(label)	43.8	(sw)

Table 6.8: Summary of the YRI data results in Appendix C by calculating the cell means and p-values of pairwise comparisons of different levels for each factor. Columns with names 'N-s5', 'N-sL' and 'N-sw' are the cell means for each main effect when not using hyperparameters, these cell means are calculated for three measures (s5, sL and sw). Columns named 'p-values' are the pairwise comparison p-values for each main effect. Only the p-values that are less than 0.05 are presented in the table. Items listed inside ( ), for example, (m.cl), and (d), are the levels of some factors that can produce better results. In particular, if there are two terms, say, ($C10 < C15$), this means, C10 and C15 are significantly better than C5, but there are no significant difference between C10 and C15 even though C10 seems to produce better results (smaller error counts).

that m.one is better than the other two models for C=10 and C=15. Overall, when not using hyperparameters, the best combination for all measures is C=10 (or C=15), d=3 and m.one.

When using hyperparameters (Table 6.7, columns H-s5, H-sL and H-sw), C has a significant main effect. There are significant interaction effects between C and m, one is for the s5 measure, (p=0.024), another is for the sw measure (p=0.016). Plot YRI-D (the interaction plot of C*m for s5 measure in Figure 6.3) shows that this interaction

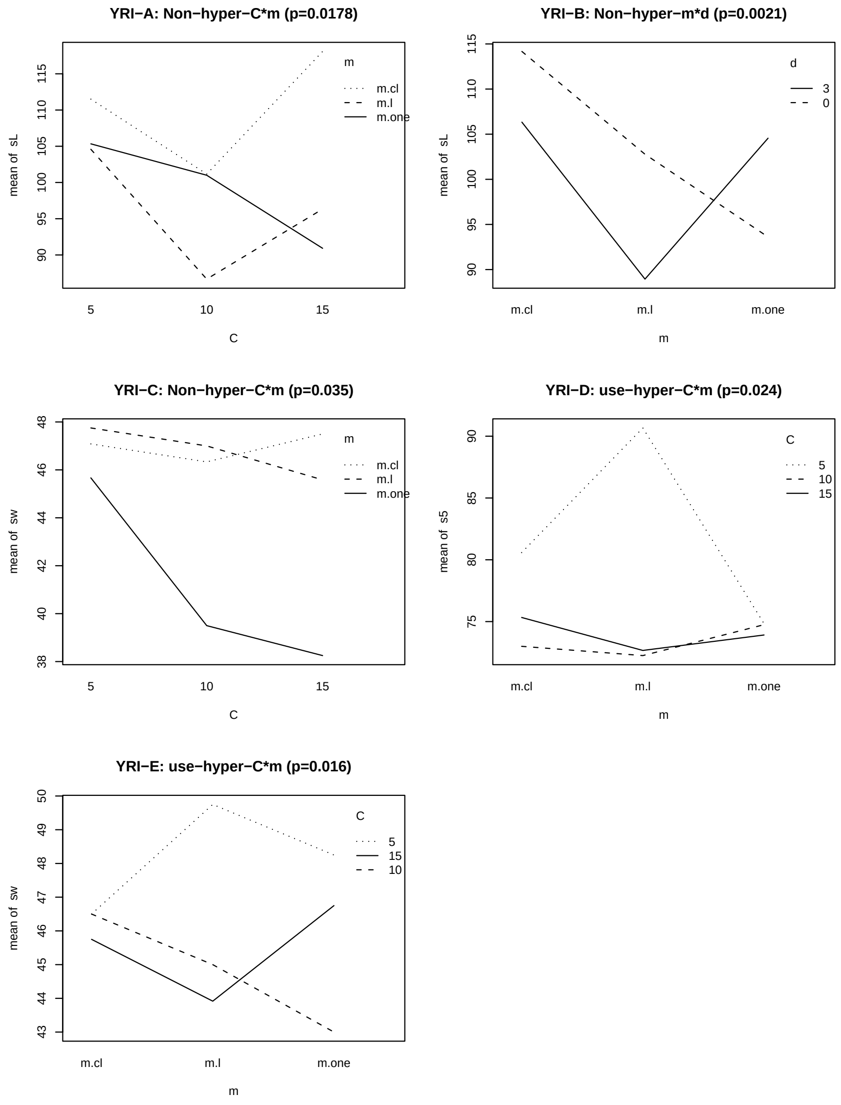


Figure 6.3: The interaction plots for the YRI data, showing mean switch distance scores for significant interaction effects in Table 6.7. The title of the first plot is "YRI-A: Non-hyper-C*m (p=0.0178)", this means that this plot is the interaction plot of C*m for the YRI data when using hyperparameters, the p-value for the this interaction effect is 0.001 as listed in the 6th row and the 3rd column of Table 6.7. Plot YRI-B, YRI-C, YRI-D, YRI-E have the similar meaning.

	H-s5	p-values	H-sL	p-values	H-sw	p-values
C5	82.0	5 v 10: 0.002	115.3	5 v10: 0.01	48.2	5 v 10: 0.001
C10	73.3	5 v 15: 0.005	102.3	5 v15: 0.016	44.8	5 v 15: 0.0095
C15	74.0	(C10<C15)	103.0	(C10<C15)	45.5	(C10<C15)
d0	75.4		103.0	0.031	45.7	
d3	77.5	(d0)	110.8	(d0)	46.6	(d0)
m.one	74.5		102.4		46.0	
m.l	78.5		111.3		46.2	
m.cl	76.3	(m.one)	107.0	(m.one)	46.3	(m.one)
label	85.3	<0.0001	108.7		48.3	<0.0001
sw	67.6	(sw)	105.1	(sw)	44.0	(sw)

Table 6.9: Summary of the YRI Data results in Appendix C by calculating the cell means and p-values of pairwise comparisons of different levels for each factor. Columns with names 'H-s5', 'H-sL' and 'H-sw' are the cell means for each main effect when using hyperparameters, these cell means are calculated for three measures (s5, sL and sw). Columns named 'p-values' are the pairwise comparison p-values for each main effect. Only the p-values that are less than 0.05 are presented in the table. Items listed inside ( ), for example, (m.cl), and (d), are the levels of some factors that can produce better results. In particular, if there are two terms, say, $(C10 < C15)$, this means, C10 and C15 are significantly better than C5, but there are no significant difference between C10 and C15 even though C10 seems to produce better results (smaller error counts).

is mainly due to the bad performance of C5, when the mutation model is $m.l$. C10 and C15 are significantly better than C5 for all three mutation models, but there is not much difference between the performance of C10 and C15. Plot YRI-E (the interaction plot between C and m for sw measure in Figure 6.3) shows that C10 and C1 are better than C=5, since there are some interactions between C10/C15 and m.l/m.one. The best parameter choice is the combination of C=10 and $m.one$. Overall, Tables 6.7, 6.9 and Figure 6.3 show that C=10, $m.one$ and $d = 0$ is the best combination.

In the last row of Tables 6.1, 6.4 and 6.7, the p-values for 'single' factor is often very small for the Daly and YRI data. This shows that how the best haplotypes are obtained has a significant effect on measures of performance. For the s5 and sw measures, the minimizing switch distance method produce better results, for the sL measure, sometimes, the label method is better.

The results of not using the hyperparameters is, in general, better than the ones using hyperparameters. This can be seen by comparing the cell means in Tables 6.2 and 6.3, 6.5 and 6.6, 6.8 and 6.9. When "hyperparameter" is treated as one main effect factor in a larger model

measure $= C + d + m + hyper + single + C * d + C * m + d * m + hyper * C + hyper * C + hyper * d + \epsilon$,

the ANOVA from the above model shows that the hyperparameter factor sometimes has a significant effect and it also shows there are some interactions between the *hyper* factor and other factors such as C and mutation model (results not shown). Therefore, the analysis for all three data sets was done separately by hyperparameter status.

6.2.3 Comparing the best single estimates with PHASE and fastPHASE

PHASE v2.1.1 (Stephens and Scheet 2005) is currently considered to be the best population haplotype inference method, as noted in Marchini et al. (2006). In addition, among the three HMM based haplotype inference methods compared in chapter 5, fastPHASE seems to be the best. Therefore, in this thesis, these two programs are compared with the performance of my HMM method. Table 6.10 displays five runs of PHASE results for the Daly, YRI and CEU data. For the CEU data, all of five PHASE runs produce results with 0 error counts for all measures (mis.G, mis.M, s=5, s=L and sw), hence, only one row is shown in Table 6.10.

Table 6.11 shows the fastPHASE results for Daly, YRI and CEU data. I present results using both methods of summarizing the best single estimate (minimizing individual error and minimizing switch distance error), as described in Scheet and Stephens (2006). For both Tables 6.10 and 6.11, the sw.prob column lists the switch proportions which can be obtained from the sw columns by dividing the total number of heterozygous markers that are needed to switch.

For the Daly data, I use the combination of C=10, d=3, and the *m.cl* mutation

Daly	Seeds	mis.G	mis.M	s=5	s=L	sw	sw.pro
	seed 1	0	11	133	210	79	0.03044
	seed 2	0	12	165	210	76	0.02929
	seed 3	0	13	142	218	79	0.03044
	seed 4	0	13	142	181	79	0.03044
	seed 5	0	12	161	192	98	0.03776
YRI	seed 1	0	1	65	61	41	0.05318
	seed 2	0	2	75	61	37	0.04799
	seed 3	0	1	53	62	39	0.05058
	seed 4	0	1	58	59	38	0.04929
	seed 5	0	1	67	63	42	0.05447
CEU	seeds 1-5	0	0	0	0	0	0

Table 6.10: PHASE results of three data sets. All parameters are set to be the default values except that the numbers of burn in and main iterations are increased to be both 500 (default is 100), and the thinning interval is increased to 3 (the default is 1). Note, the default modeling setting in PHASE v2.1.1 is that the "coalescent-with-recombination" prior is used.

Daly	single	mis.G	mis.M	s=5	s=L	sw	sw.pro
T50K10	indiv:	0	11	99	153	67	0.02582
	min.sw:	0	11	96	196	71	0.02736
T50CV15	indiv:	0	12	109	186	77	0.02967
	min.sw:	0	12	105	181	74	0.02852

YRI	single	mis.G	mis.M	s=5	s=L	sw	sw.pro
T50K10	indiv:	0	1	87	92	46	0.05966
	min.sw:	0	1	58	66	36	0.04669
T50CV10	indiv:	0	1	43	76	34	0.0441
	min.sw:	0	1	61	67	41	0.05318

CEU	single	mis.G	mis.M	s=5	s=L	sw	sw.pro
T50K10	indiv:	0	0	5	1	2	0.00417
	min.sw:	0	0	5	1	2	0.00417
T50CV15	indiv:	0	0	5	1	2	0.00417
	min.sw:	0	0	5	1	2	0.00417

Table 6.11: The fastPHASE results for three data sets. T50K10 means that fastPHASE was run with 50 different starting values in order to avoid the local mode problem of the EM algorithm, and the number of ancestral haplotypes was fixed at 10. In the fourth row and first column of each table, T50CV10 (or T50CV15) means running fastPHASE with 50 different starting values and letting the program do cross validation to pick the best number of ancestral haplotypes and 'CV10' or 'CV15' means that the best number of ancestral haplotypes selected by fastPHASE is 10 or 15. The parameters of fastPHASE were set to their default values. Rows labeled (from the second column) with "indiv" show results with the best single estimate obtained by minimizing the individual error (Scheet and Stephens, 2006). Rows labeled with "min.sw" (from the second column) show results with the best single estimate obtained by minimizing the switch distance.

Daly	s=5 range	s=L range	sw range
PHASE :	133-161	181-218	76- 98
fastPHASE:	96 -109	153-196	67-77
my HMM:	101-128	176-198	76-92

YRI	s=5 range	s=L range	sw range
PHASE :	53-75	59-63	37-42
fastPHASE :	43-87	66-92	36-46
my HMM:	43-82	93-124	33-45

CEU	s=5 range	s=L range	sw range
PHASE :	0	0	0
fastPHASE :	5	1	2
my HMM:	0	0-2	1-2

Table 6.12: Comparison of three methods using three data sets. The PHASE results of three data sets are the summary of Table 6.10. The fastPHASE results are the summary of Table 6.11. For my HMM results, the parameter combinations as follows, the Daly data: C=10, d=3 and the *m.cl* model; the YRI data: C=10, d=3 and *m.one* model; the CEU data: C=10, d=3 and the *m.cl* model.

model (without using hyperparameters) for my HMM method to compare with PHASE and fastPHASE. The range of performance scores for each method (for all three measures, s5, sL, sw) is shown in the first 4 rows of Table 6.12. This summary table shows that fastPHASE is better than my HMM and my HMM is slightly better than PHASE.

For YRI data, I use the combination of C=10, d=3, and m.one for my HMM method (without using the hyperparameters). The comparison range is summarized in the 5th to 8th row of Table 6.12. This summary shows that PHASE is only better than the other two programs for the s=L measure. For the s=5 and sw measure, my HMM is slightly better than fastPHASE.

For the CEU data, the summary is in the the last 4 rows of Table 6.12. The combination of C=10, d=3 and the $m.cl$ mutation model is used for my HMM method (without using hyperparameters). PHASE produces the best results, but my HMM is slightly better than fastPHASE for the s=5 measure.

For the $mis.M$ column in the result tables, PHASE and fastPHASE have smaller values than the HMM method on average. For example, the $mis.M$ count for these two methods are about 11 or 12, while in my HMM method, it is 12 to 15. For the $mis.G$ column, these two methods have "0" error which is due to their assumption of no genotyping error.

In conclusion, all of these three methods produce fairly comparable results, but occasionally, one method seems a bit better or worse. However, differences could be due to the variation among different runs, to the variation caused by the missing genotypes, for example, there are about 10% missing genotype in the Daly data; or to the small sample size, for example, there are only 30 individuals in the YRI data and CEU data.

6.2.4 Comparing the posterior samples with PHASE

In this subsection, two figures, for the Daly and YRI data, have been used to compare the HMM results with the PHASE results based on the distributions of the posterior samples. Note that the results of my HMM methods and PHASE are so close for the CEU data, I did not graph their posteriors. This comparison still uses MERLIN inference as the standard. For the Daly and YRI data, there is one plot for each data set to compare the posterior sample person by person. Loci are ignored when no true

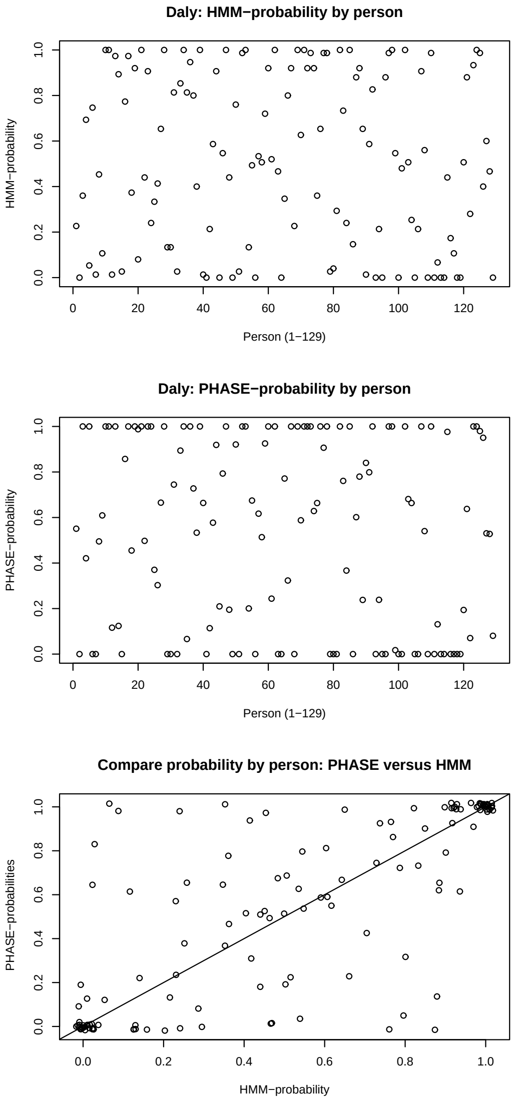


Figure 6.4: Comparison of the posterior probabilities of the correct haplotype as produced by the PHASE program and the HMM method for the Daly data (with 103 markers), person by person. For the HMM method, C=10, d=3, mutation model is "m.cl" and the above probabilities are based on every 4th iteration of last 300 out of a total of 600 iterations. For the PHASE program, the probabilities are based on every 4th of last 500 out of 1000 iterations.

parental haplotype origin can be inferred because of missing data, or because both parents and child have same heterozygous genotypes. These plots are made as follows:

(1) For both the PHASE and the HMM method, take R posterior samples for the haplotypes of each person, summarize all possible haplotype combinations, and calculate the probability of each haplotype combination. (PHASE has provided this in its output files).

(2) For one person, compare all haplotype combinations with MERLIN, and add up the probabilities of the haplotype combinations that are consistent with MERLIN's inference.

(3) Repeat (1) and (2) for all P people to get a vector of probabilities with length P, and plot this vector.

For the Daly data, the posterior plot (Figure 6.4) shows that the PHASE posterior samples generate more "0" probabilities (which is bad) than the HMM method, and it has more "1" probabilities (which is good) as well. Specifically, PHASE has 33 (out of 129) "0" probabilities, the HMM method has only 17 (out of 129) "0" probabilities; PHASE has 36 (out of 129) "1" probabilities, the HMM method has only 4 (out of 129) "1" probabilities. The HMM methods have more probabilities that are between 0 and 1 than the PHASE program.

For the YRI data, the posterior plot (Figure 6.5) shows that PHASE posterior samplers also have more "0" and "1" probabilities as well. Specifically, PHASE has 7 "0"s and the HMM has 2 "0"s; PHASE has 2 "1"s and the HMM has 0 "1" s. There are more 'points' above the straight line in the third plot, which shows that PHASE

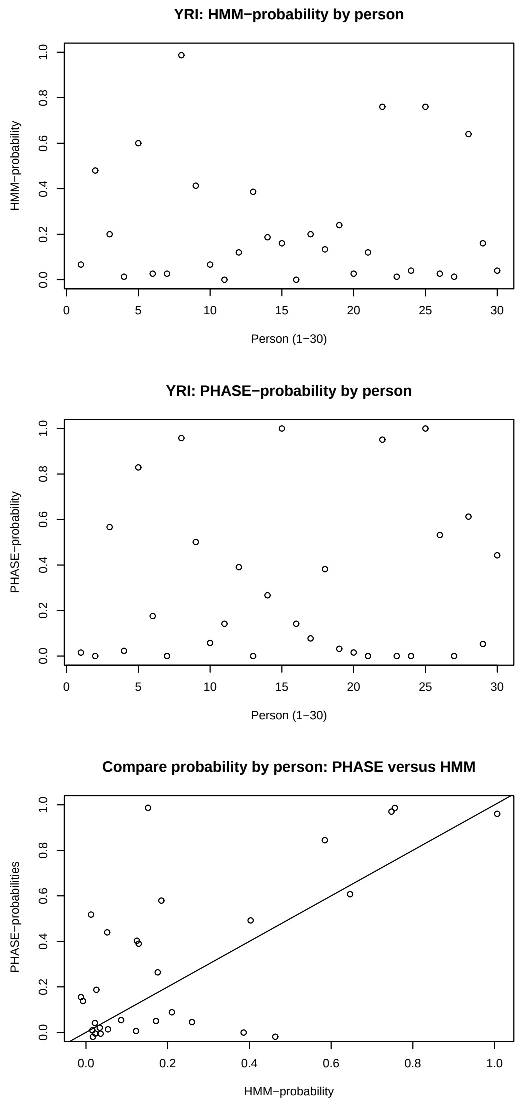


Figure 6.5: Comparison of the probabilities of the correct haplotype as produced by the PHASE program and the HMM method for the YRI data (with 91 markers), person by person. For the HMM method, C=10, d=3, mutation model is “m.one” and the above probabilities are based on every 4th iteration of last 300 out of a total of 600 iterations. For the PHASE program, the probabilities are based on every 4th of last 500 out of 1000 iterations.

	Sampled A							Sampled Q_1
ch1	C	G	T	T	T	A	G	0.49
ch2	T	A	A	T	T	G	G	0.02
ch3	T	G	T	T	T	G	A	0.30
ch4	T	G	A	T	T	A	G	0.08
ch5	C	G	T	T	G	A	G	0.05
ch6	C	G	T	C	T	A	G	0.07

Table 6.13: Sampled ancestral haplotypes (A) for haplotype block 8 of the Daly data (see Appendix A) and sampled Q_1 from iteration 90 in a run with $C = 6$ and $d = 0$.

is more certain in its haplotype assignment than the HMM method for this particular YRI data.

6.3 Ancestral haplotypes

The idea of ancestral haplotypes can be very helpful when constructing haplotypes near disease-causing mutations. Specifically, these ancestral haplotypes may be considered as the putative progenitor haplotypes (Neuhausen et al. 1996, Niell et al. 2003) of the most recent common ancestors (MRCA). On those progenitor haplotypes, all individuals may carry the same mutation. The estimated ancestral haplotypes from the HMM are not guaranteed to resemble the true ancestral haplotypes from which the population was derived, but it is still likely that more common haplotypes are older than rare haplotypes.

In order to show a simple picture of the HMM estimated ancestral haplotypes, a run of the HMM method is done only on the genotypes of haplotype block 8 of the Daly data (Daly et al. 2001), with the parameter choices that C=6, d=0, mutation model is *m.one*, and without using hyperparameters. The sampled A for this block and the occurrence frequencies of different ancestral haplotypes at the first locus, Q_1, is given in Table 6.13. For short sequences in strong LD, Q_1 shows how frequently each ancestral haplotype is used. The first four haplotypes in A are the same as the four common haplotypes reported in the 'A row' of Figure 2, Daly et al. (2001). The last two haplotypes each differ from $ch1$ at only one marker. The values of Q_1 for the first four ancestral haplotypes are $0.49, 0.02, 0.30$, and 0.08, and the sum of these values is 0.89, which is very close to 0.91, the value reported in Daly et al. (2001) as the combined frequencies of their four common haplotypes.

In order to get a more detailed picture of the frequency of usage for ancestral haplotypes, a plot of the sampled S for all 103 loci in the Daly data is given in Figure 6.6. Each of these 10 lines represents one ancestral haplotype. The space between the i^{th} and $(i-1)^{th}$ line, for $i = 1, \cdots, 9$, is the frequency of the i^{th} ancestral haplotype

Two common haplotypes in sampled A for the Daly data (103 markers)
Red haplotype : GGACAACC (G) TTACG (C) CGGAGACGA CGCGCCCGGAT CCAGC CCGAT CCCTGCTTACGGTGCAGTGGCACGTATTGCA (C) CGTTTAG (C) ACAACA GTTCTGA TATAG
Green haplotype: AATTCGTG (G) CCCAA (C) CGCAGACGA CTGCTATAACC GCGCT CTGAC TCCCATCCATCATGGTCGAATGCGTACATTA (C) TGTTTGA (G) GCGGTG TGTGCGG CGGCG

Table 6.14: Alleles grouped together are in the same haplotype block. Alleles in parentheses are between two consecutive blocks. Alleles (TGTGCGG) that are underlined mean that they are the same as the blue haplotype in block 8 of 'row A' , Figure 2, Daly et al. (2001). (Note, this figure is listed in Appendix A)

in S. It can be seen that, these 10 lines are relatively stable between block boundaries. However, there are some sharp curves or jumps near the block boundaries, showing that the individuals appear to have recombined chromosomes.

Two haplotypes, named 'Red haplotype' and 'Green haplotype' in Table 6.14, are most frequently sampled, as seen in Figure 6.6. Their frequencies in S correspond to the space between the 8^{th} and 9^{th} line in the plot, and the space between the 9^{th} and 10^{th} line in the plot, respectively. From Figure 6.6, we can see that at almost all loci, those two ancestral haplotypes are used more than 50% of the time in the sampled S. In fact, Red haplotype and Green.hap are exactly the same as the red and green haplotypes in the "A row" of Figure 2, Daly et al. 2001, except for a swap between green and blue haplotypes at block 10 in the analysis of Daly et al. (2001). Between block 9 and 10 the haplotype exchange rate has the high of 27%, as mentioned in 'D row' of Figure 2 in Daly et al. (2001), and hence a swap in the HMM is not unlikely. Note that the figure in the Appendix A (Figure 2 in Daly et al. (2001)) is attached as a black and white figure, so it is worth pointing out that the 'Red haplotype' and the 'Green haplotype' in this section corresponds to the first and the third haplotype in the A row of this attached figure. The "Blue haplotype" corresponding to the second haplotype in the A row this appendix figure.

In addition to plotting the sampled S of the Daly data, the plots of sampled S's for the CEU and YRI data are shown in Figure 6.7 and 6.8, respectively. For the CEU data, there is no obvious recombination. Three ancestral haplotypes are more commonly used than others. For the YRI data, Figure 6.8 shows an obvious pattern

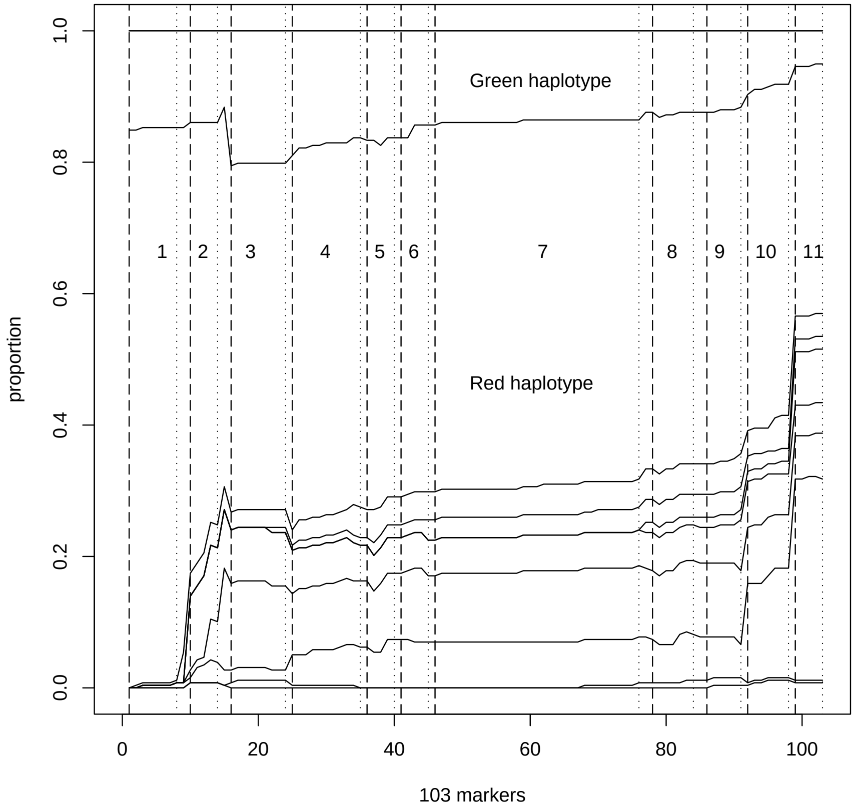


Figure 6.6: Proportion of S that uses each ancestral haplotype (A), in an analysis of all 103 loci in the Daly data. Samples are drawn at iteration 600 from a run with C=10, d=3, mutation model m.cl and without using hyperparameters. The vertical long dashed line is the first marker of each haplotype block, the vertical dotted line is the last marker of each block, and 1 to 11 indicate the haplotype block numbers. Red haplotype and Green haplotype are the two haplotypes that most commonly used among all individuals, the length between the line below and above them is the frequencies that they are used.

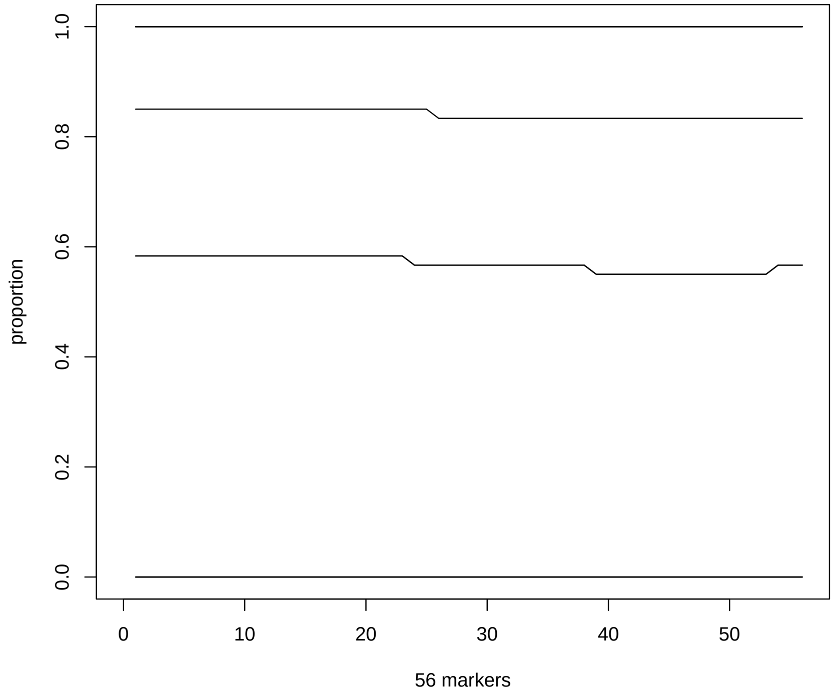


Figure 6.7: Proportion of S that uses each ancestral haplotype (A), in an analysis of all 56 loci in the CEU data. Samples are drawn at iteration 600 from a run with C=10, d=3, mutation model m.cl and without using hyperparameters.

of recombination. Three ancestral haplotypes are used most frequently between the $20th$ and the $60th$ marker, but no single ancestral haplotype is used very frequently at all loci.

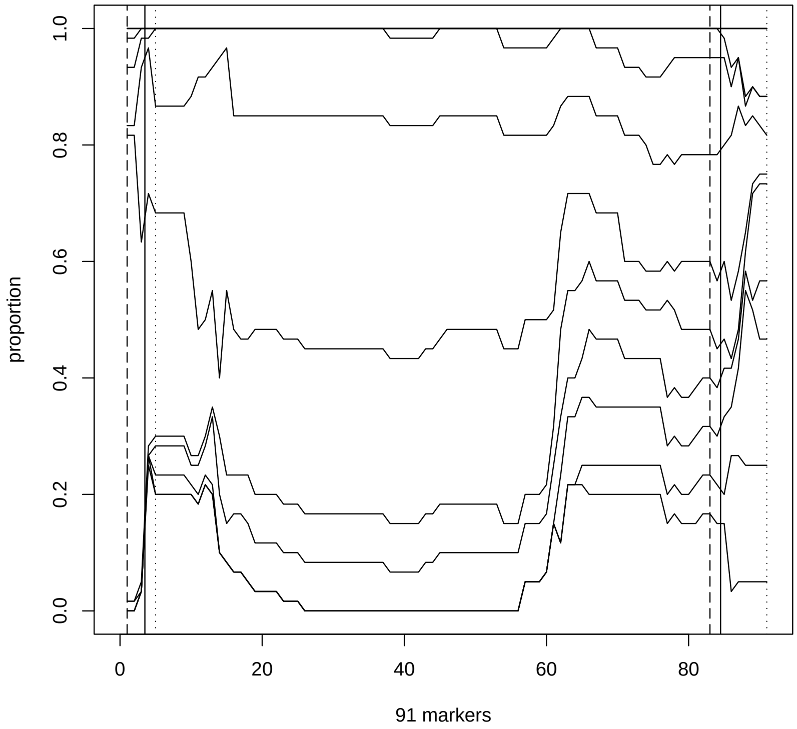


Figure 6.8: Proportion of S that uses each ancestral haplotype (A), in an analysis of 91 loci in the YRI data. Samples are drawn at iteration 600 from a run with C=10, d=3, the mutation model m.one and without using hyperparameters. Vertical long-dashed line, solid line and dotted line are the beginning, center and end of each of those two recombination hotspots mentioned in section 6.1.

6.4 Recombination and mutation models

6.4.1 The mutation models

As shown in the previous ANOVA results, the mutation model used plays a significant role. Comparisons of the three mutation models allow us to observe some interesting patterns in the three data sets.

First, for the CEU data, the result table (Table A3 of Appendix C) shows that when using the mutation model that has one mutation rate for each locus (i.e. 'm.l' model), the performance of the HMM method is obviously worse than the others. After investigating the error count and the posterior mutation estimates, I learned that at a specific locus ($l = 47$), the posterior estimate of the mutation rate is higher than the other loci. Since this mutation rate is higher, more inconsistencies between (A, S) and H are allowed at locus 47. On the other hand, the genotyping error is very small, and the recombination rate is relatively low (T_l is very large) for this CEU data. As a result, more heterozygous genotypes are placed in the wrong order in the estimated haplotypes. When the *m.cl* model is used, the apparent high mutation rate at $l = 47$ only applies to a few ancestral haplotypes and hence the estimates of the mutation rate have much less effect on the overall error.

Second, another interesting pattern occurs in the results of the YRI data. Contrary to the other two data sets (Daly and CEU), for which the 'm.cl' mutation model performs best, the YRI data show that the best results are obtained when the mutation model is *m.one* (one mutation variable for all loci). In my opinion this is because the recombination pattern is irregular and this data set contains a lot more recombinations than the other two data sets. This can be seen in the recombination plots of the YRI data Figure 6.8, recombinations occur at several loci. Therefore, for this YRI data set, recombination should play more of a role in the whole 'inheritance' process. If the mutation model 'm.cl' (or 'm.l') is used, more variation is added to the whole model through additional parameters, and the estimated variance of the mutation rate are higher. On the other hand, when using just one mutation variable for all loci, the mutation estimate is forced to become very small by the overall rarity of mutations. This allows the recombination parameters to play more of a role as it is expected, which in turn produces better results. That is, when the mutation rate is reasonably small, the small mutation rate forces the ancestral chromosomes to recombine more frequently to reduce the inconsistencies between (A, S) and H; this can then help produce better results.

6.4.2 The recombination parameter

Another important parameter of the HMM method is the recombination parameter, T. Plots of the sampled T for both the Daly data (Figure 6.9) and the YRI data (Figure 6.10) show that this parameter is helpful in estimating the location of recombination hotspots.

In the plot of the Daly data (Figure 6.9), the probability of staying with the same ancestral chromosome is lower at the block boundaries of Daly et al. (2001). In addition, there are about 4 large "dips" in the HMM recombination parameter plot between blocks 2 and 3, 3 and 4, 9 and 10, and 10 and 11, where all sampled T's are less than 90%. These dips are qualitatively quite consistent with the 'D row' of Figure 2 of Daly et al. (2001) (see Appendix A), which shows the estimated haplotype exchange rate between blocks. Hence, our recombination parameter T can identify the regions where recombination occurs in the Daly data, and produce estimate similar in magnitude.

The 91 markers in the YRI data were deliberately selected from a region containing recombination hotspots on chromosome region 19q13. As mentioned in section 6.1, this region spanned from 37128001bp to 37272001bp, includes two identified recombination hotspots, one is from 37128001bp to 37137001bp, with hotspot center at 37134001bp, and another from 37262001bp to 37272001bp, with hotspot center at 37264001bp. Figure 6.7 shows that at those two regions, the sampled T's of the HMM method have a 'dip' at or close to each hotspot center. Once again this implies our sampled T is useful for predicting the recombination hotspots in this data set. Furthermore, although the HAPMAP web resource did not mention other recombination hotspots in this region, the sampled T's suggest that there may be other additional recombination hotspots between these known hotspots.

The CEU data was not deliberately selected from a recombination hotspot region, and the recombination plot in Figure 6.11 shows that there is no obvious recombination pattern.

One interesting and important question is how the estimated amount of recombination changes when the other aspects of the model change, for example, the number of ancestral chromosome (C), or the mutation models. In order to answer this, I plotted the 600th sampled T (the last iteration) for three different numbers of ancestral chromosomes (C=5, 10, and 15) with mutation model fixed as 'm.cl' and d=3. I also plotted the 600th sampled T for the three mutation models (with C=10, d=3 fixed). These plots were done for both models without using hyperparameters and models using hyperparameter, for all three data sets (Daly, CEU, YRI), and are shown in Figures 6.12, 6.13 and 6.14. These three plots show that the overall main pattern of recombination is not affected much when C changes, or when different mutation models are used. This is true for both models with and without using hyperparameters. However, there are two noticeable differences. First, when not using hyperparameters, at low-recombination areas, there is still some variation among the sampled T, but when using hyperparameters, the sampled T is pushed to be strictly close to 1. Second, when using hyperparameters, the estimated T_l becomes smaller in regions with more recombination (small T_l values) in the Daly and YRI data,

In conclusion, the recombination parameters of the Hidden Markov Model can help predict recombination hotspots. The overall pattern of recombination does not change much when the number of ancestral chromosomes, or the mutation model changes. After running the program on a specific data set for a few times, plotting several sampled T against the physical distance or the marker number may reveal a pattern of recombination. If most or all sampled T_l's between any two consecutive loci l and $l + 1$ have values less than 0.85, this could be evidence of a recombination hotspot.

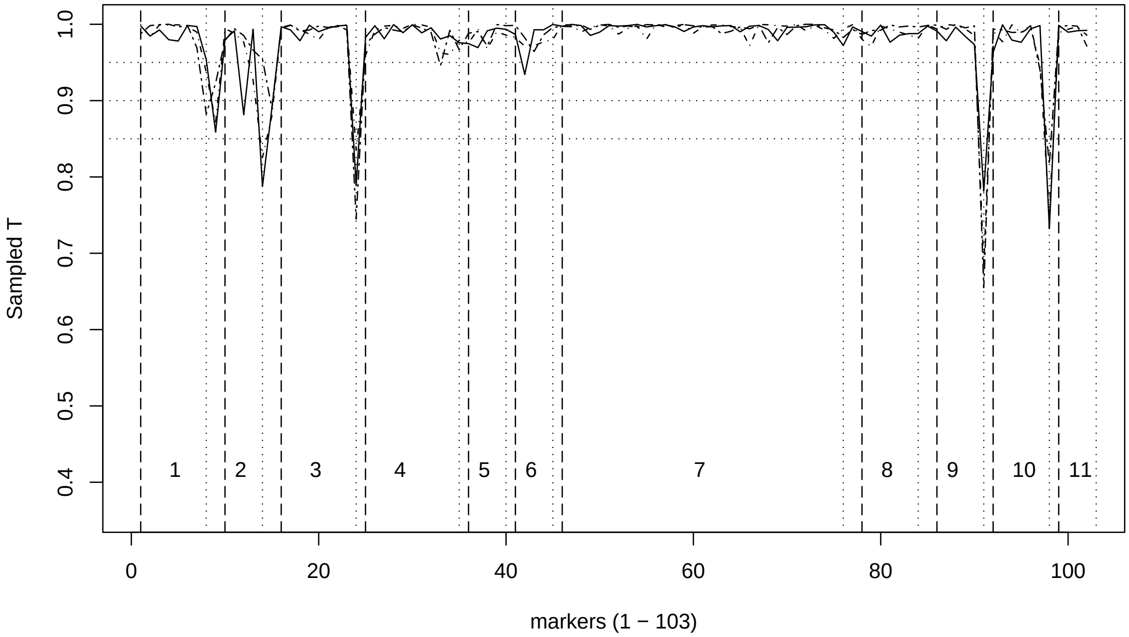


Figure 6.9: Samples of $T_l, l = 1, \cdots, 103$, from the posterior distribution for the Daly data. Samples are drawn at iteration 400 (dotted line), 500 (longdash line) and 600 (solid line) from a model with the combination of $C = 10, d = 3$ and the mutation model 'm.cl'. Vertical dashed and dotted lines represent each block's start and end, respectively, for all the 11 haplotype blocks defined in Daly et al. (2001). The number (1 - 11) in the bottom indicates those 11 haplotype blocks.

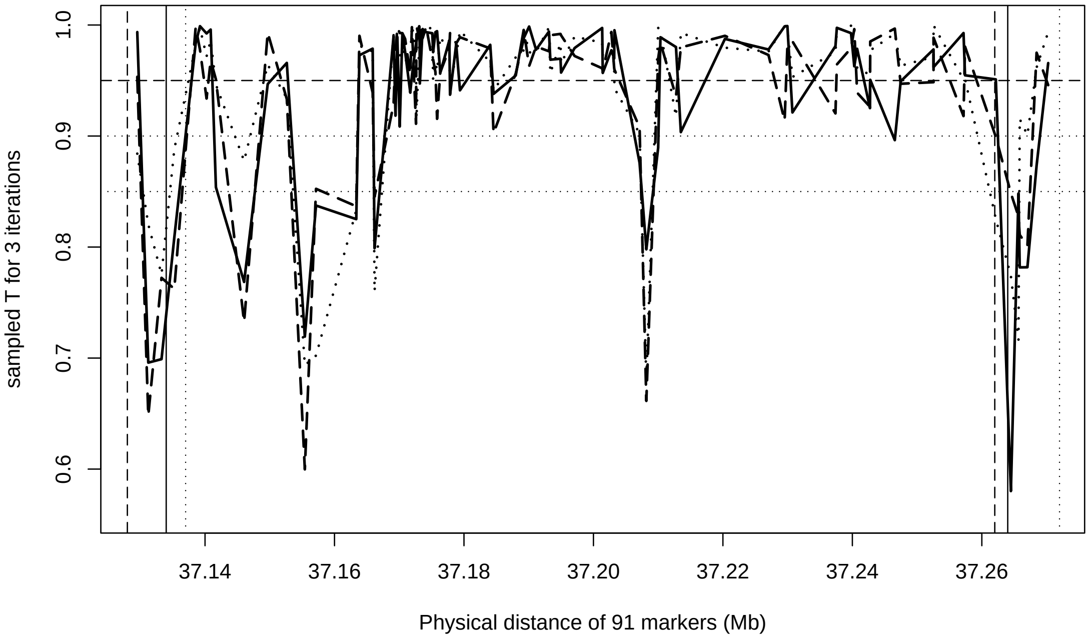


Figure 6.10: Samples of $T_l, l = 1, \cdots, 91$, for the YRI data on ch19q13. Markers are plotted by their physical location. Vertical long-dashed line, solid line and dotted line are the beginning, center and end of each recombination hotspot. Samples are taken from a run at iteration 400 (dotted line), 500 (longdash line) and 600 (solid line) with parameter choice $C = 10$ and $d = 3$. In addition, the mutation model is one mutation variable for all loci.

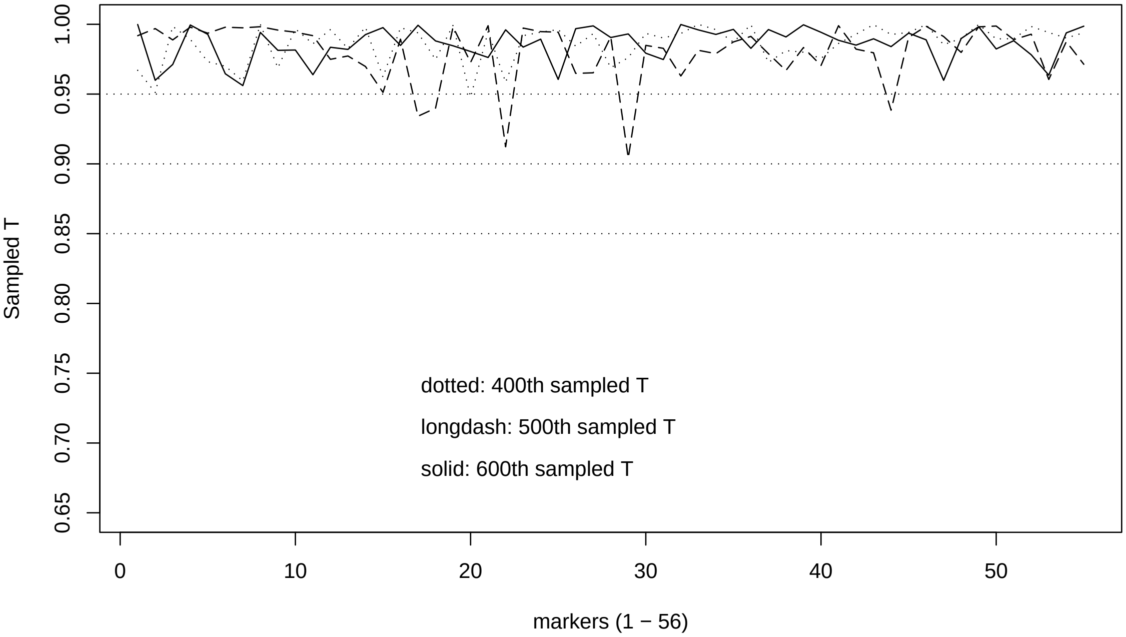


Figure 6.11: Samples of $T_l, l = 1, \cdots, 56$, for the CEU data on 7p15.2. Samples are taken from a run with $C = 10$ and $d = 3$ at iteration 400, 500 and 600. The mutation model is 'm.cl'.

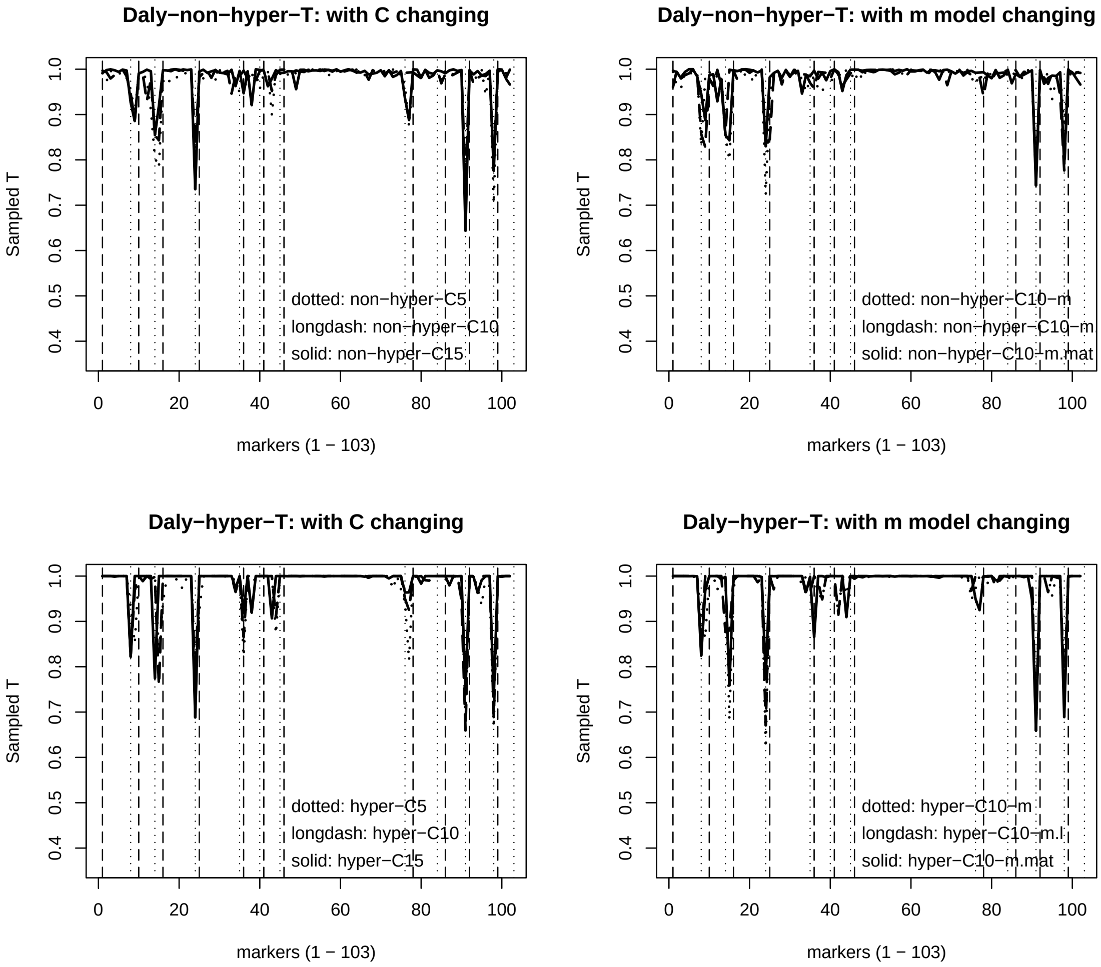


Figure 6.12: Samples of T_l ($l = 1, \cdots, 102$), for different C values (C=5, 10 and 15), different mutation models (*m.one*, *m.l* and *m.cl*) and different hyperparameter choices (using or not using hyperparameters for T and *m*).

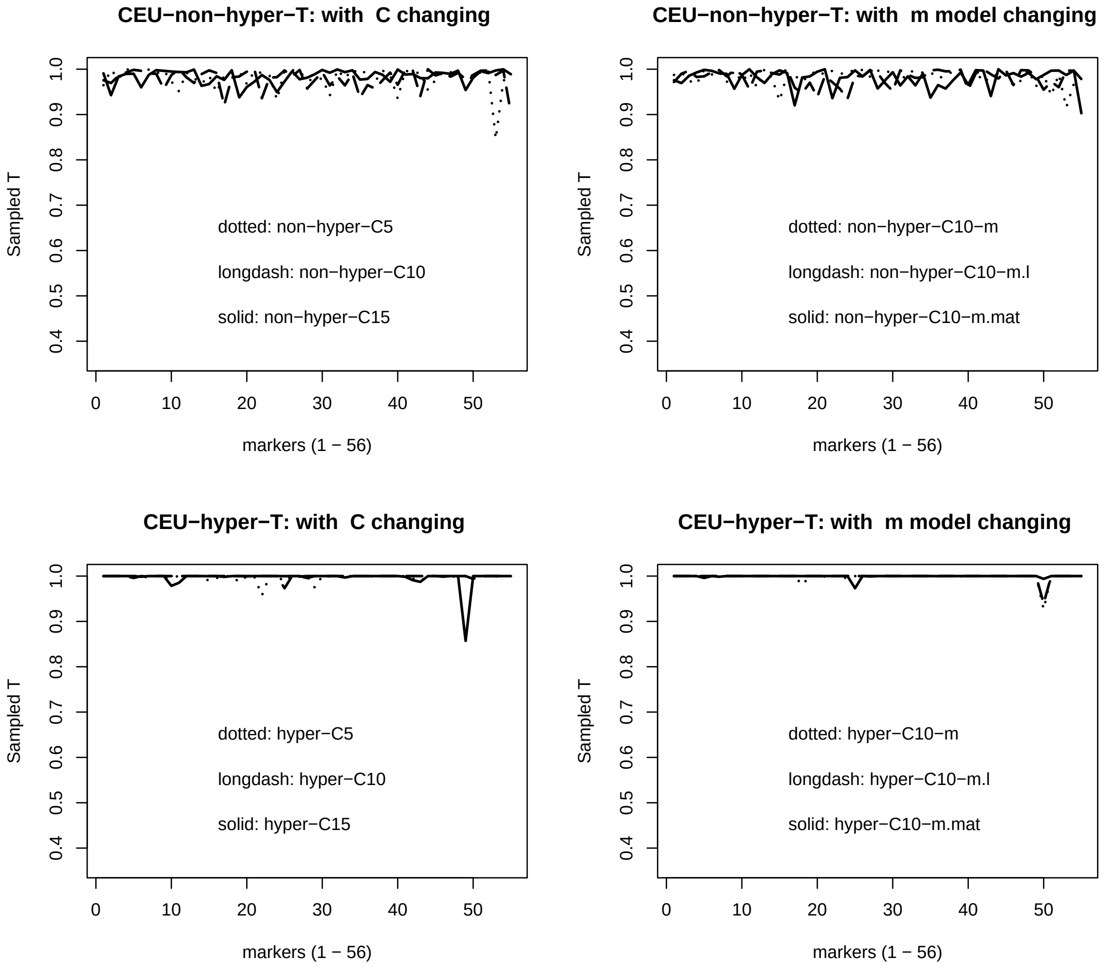


Figure 6.13: Samples of T_l ($l = 1, \cdots, 55$), for different C values (C=5, 10 and 15), different mutation models ($m.one$, $m.l$ and $m.cl$) and different hyperparameter choices (using or not using hyperparameters for T and m).

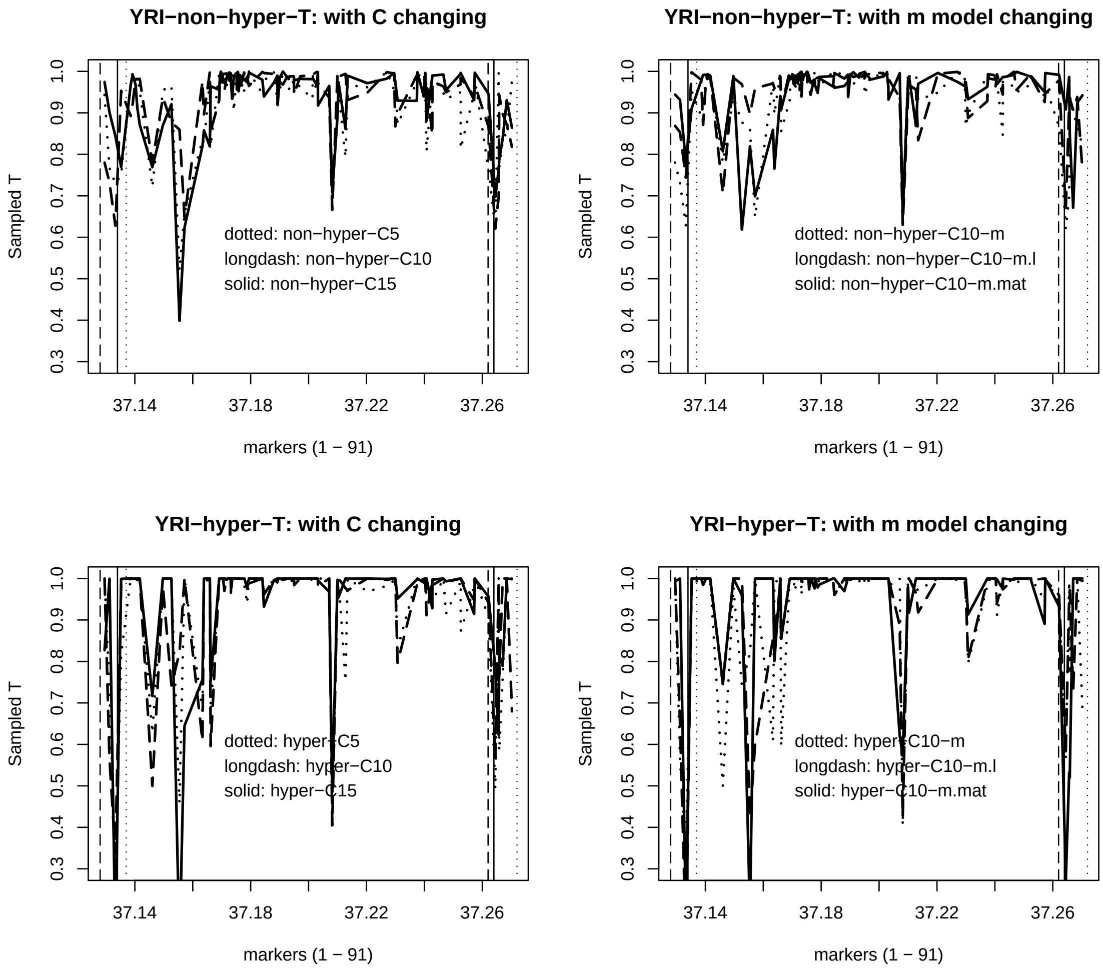


Figure 6.14: Samples of T_l ($l = 1, \cdots, 90$), for different C values (C=5, 10 and 15), different mutation models (*m.one*, *m.l* and *m.cl*) and different hyperparameter choices (using or not using hyperparameters for T and m).

Chapter 7

Discussion

In this thesis, a new Bayesian method was developed to do haplotype inference. This method is based on the assumption that present haplotypes are all inherited from ancestral haplotypes of many generations ago. Recombination and mutation processes, the two main sources of genetic variation, are included in this model. The recombinations between different ancestral haplotypes lead to a hidden Markov chain. Therefore, this haplotype inference method is based on a Hidden Markov Model. Markov Chain Monte Carlo methods were used to sample from the posterior distribution.

7.1 About using a Bayesian method

Generally speaking, there are three advantages of using a Bayesian method. First, an informative prior about a parameter can help improve the performance of the model, because inference will be based on not only the data, but also on the prior knowledge. In contrast, a non-Bayesian method would infer based only on the data itself. Second, it is much easier to examine the uncertainty of the estimate through the posterior distribution. Instead of only producing a best single estimate (a point estimate), the distribution of different haplotype combinations is also obtained. Third, it is relatively easy to incorporate other known information such as partially known haplotypes, as was done in Stephens et al. (2001).

Bayesian methods also have some drawbacks. First, if the the prior does not reflect the truth very well, then the result can be biased. Second, MCMC methods can be very slow to achieve convergence and sample from the posterior. For example, for fastPHASE (a non-Bayesian method) which is implemented using the EM algorithm, the computation time for the Daly data is about half an hour to get the presented results (with T=50 different starting values). The PHASE program is a Bayesian method using MCMC. Its running time is about 1.5 minutes per iteration, and takes about one day (25 hours) to get the results presented (1000 iterations in total). For the HMM method presented in this thesis, with C=10, it takes approximately 3 minutes to do one iteration for the Daly data, and the total amount of time to get posterior samples over 600 iterations is approximately one day (30 hours). As for the number of iterations for PHASE and the HMM method needed to get the current results, I first ran 600 iterations for the PHASE program, and found that the results varied a bit, and for the HMM method, results similar to those presented can be obtained in a run of 200 iterations. Here more iterations are run in order to make sure the Markov chains of both methods converged well. Note that the program in this thesis is implemented in R. If it were converted to C, its speed might improve from about 10 to 100 times. Both

PHASE and fastPHASE are implemented in C. The above computing time comparison is based on using the same computer for the three programs.

7.2 About the genotyping error model

Genotyping error is a very important factor in genetic studies. However, to my knowledge, genotyping error has not been considered in current haplotype inference methods, with the exception of Xing et al. (2004). One feature of my HMM-based model is the incorporation of a genotyping error parameter. A consequence of this feature is that observed genotypes are not required to match the sampled values of haplotypes (H). As shown in the result tables (Appendix C), my HMM-based method reports some potential genotyping errors. In contrast, PHASE and fastPHASE report no genotyping errors (i.e. '0' in their mis.G count), since these methods assume no genotyping errors.

Another point that is worth mentioning is that more genotyping errors are inferred when using the label method to get the best single haplotype estimate than when using the minimizing switch distance method. For the label method, the haplotype allele inference is based on the determination of mother-father labels at each chromosome. That is, the haplotype allele inference at each locus is based on the majority allele on a parental label in the samples. When the number of samples is very small, this proportion may be poorly estimated. However, when using the minimizing switch distance method to obtain a single estimate, haplotype allele inference is based on the sampled haplotype-pair proportion, which tends to be very consistent with the genotype pairs when the genotyping error rate is small. Therefore, the minimizing switch distance method tends to report fewer G-H inconsistencies than the label method. I suggest running the program a few times with different starting seeds, and then checking if those different runs report the same G-H inconsistencies. If so, there may be a potential genotyping error.

As described in section 6.2.1, when using the label method to obtain the best single estimate, the counts in the s=5, s=L and sw columns include ‘G-H inconsistent’ loci. Using these penalized error counts to compare with PHASE and fastPHASE may not be a fair comparison if the truth is that there really are genotyping errors.

7.3 About mutation

As shown in chapter 6, the effects of recombination and mutation interact. For data sets with different recombination patterns, the three mutation models perform differently. The results of those three data sets may suggest the following conclusion. When a data set has a regular recombination pattern (i.e. a strong LD pattern, or obvious haplotype block structure, like the Daly data), the mutation model ‘m.cl’ performs best. When a data set has markers close to each other with very low recombination (like the CEU data), the ‘m.cl’ model also performs best. For a data set that has an irregular recombination pattern, or many more recombinations (e.g. the YRI data), then the ‘m.one’ model performs best. Note that, these conclusions are just based on results of the three data sets. To generalize them, experiments on more similar data sets are required.

Three more points about the mutation model are worth mentioning. First, all three mutation models (m.one, m.l and m.cl) allow one allele to change to any other allele with equal probability. For microsatellite markers ($N_l > 2$) this may be unlikely. Second, the $m.cl$ and $m.l$ mutation models use the same prior for all markers. For real genetic data, it may be possible to assign a more informative prior to some of the loci, and this might help increase the accuracy. Third, in order to allow the Markov chain to move freely in the state space, at first, for all three mutation models, the starting values of the mutation rates were given a relatively large value of 0.2.

7.4 About recombination

Recombination, as one source of genetic variation, plays a very important role in the Hidden Markov Model. In fact, the essential role of recombination can be demonstrated by fitting a simpler model without ancestral haplotypes. That is, there are no ancestral haplotypes and no parameters related to the recombination (i.e. S, T, Q, m) and only haplotypes H, genotypes G and genotyping error e are kept in the model. A high-order Markov model of order d is used as the prior for H (similar to the high order Markov model for A in chapter 3). That is,

$$\begin{aligned} P(H,G,e) &= P(H|G,e)P(H) \\ &= \left[\prod_{p=1}^{P} P(G_p|H_p,e)\right]\left[P(H_{1:d})\prod_{l=d+1}^{L} P(H_l|H_{(l-d):(l-1)})\right] \end{aligned}$$

The results (not shown in this thesis) on the Daly data are increasingly good as the order of the Markov model increases from $d = 1$ to $d = 4$. When $d = 4$, the results are close to those obtained based on the Hidden Markov Model. However, for the YRI data, the results are much worse than the ones obtained from the hidden Markov model, with the error counts about two times bigger. This shows that the introduction of "recombination" and ancestral haplotypes is essential, and the hidden Markov chain is playing a very important role in this haplotype inference model.

7.5 About the best single estimate and the comparison results

A single estimate (e.g. the mode) is not an ideal way of summarizing the posterior distribution, and so may not be ideal for comparing methods. As seen for both the HMM method of this thesis and fastPHASE, two different ways of summarizing the best single estimate may produce quite different results. This is because when one best single haplotype estimate is obtained, a lot of useful information in the posterior samples is discarded. The loss of information is due to the complex dependences between the phase calls at different markers between and within all individuals (Stephens et al. 2001). A better solution is to access the uncertainty of different haplotype combinations.

In addition, one can see that the results of different runs vary even though one uses the same best single estimate method. This variation can be seen in PHASE results for the Daly data as well. This is likely due to lack of convergence of the Markov chains in these two models. It might take much longer runs to make the Markov chain converge, and produce results that do not vary with the random number seed used.

When comparing with PHASE and fastPHASE, the best parameter settings of my HMM method were used, obtained by training my HMM model on these three data sets. However, the parameters in the other two methods were not similarly optimized. Therefore, this might make my results overly optimistic. It would be more fair, perhaps, to compare the methods on a new data set where parameter choices were not previously optimized for that data set. However, the three data sets that I used have varied genetic structures, and they are real genetic data sets which represent general genetic variation patterns (i.e. with or without obvious recombination patterns). Hence, even if I use a new data set, the recombination pattern might not be dramatically different from those of the data sets I used. The parameter setting of PHASE has been improved several times in the past several years and I believe its default setting is the best setting in general. For fastPHASE, I used both a fixed number of ancestral haplotypes and the cross validation method to select the number of ancestral haplotypes. Therefore, the comparison result is reasonable.

7.6 About Hardy-Weinberg equilibrium (HWE)

If a population is in HWE, the genotype frequencies of a marker are only dependent on the allele frequencies. In the context of haplotype inference, this HWE assumption can be interpreted in a more general way, that is, the frequency of each pair of haplotypes is the product of the haplotype frequencies of those two haplotypes. The

HWE assumption might not always be satisfied. For example, when the individuals in a data set are from two different populations, suppose haplotype X only exists in individuals of one population, haplotype Y only exists in individuals of another population. The frequency of haplotype pair (X,Y) for one specific individual will be 0, but, under HWE, the frequency of (X,Y) will not be 0 since it is the product of the frequencies of X and Y. The violation of this HWE assumption might affect the accuracy of haplotype inference. However, several studies (Fallin and Schork 2000, Stephens et al. 2001, Polanska 2003) have demonstrated that departure from HWE will not affect the haplotype results dramatically, therefore, no further investigation was conducted in this study.

7.7 Future research directions

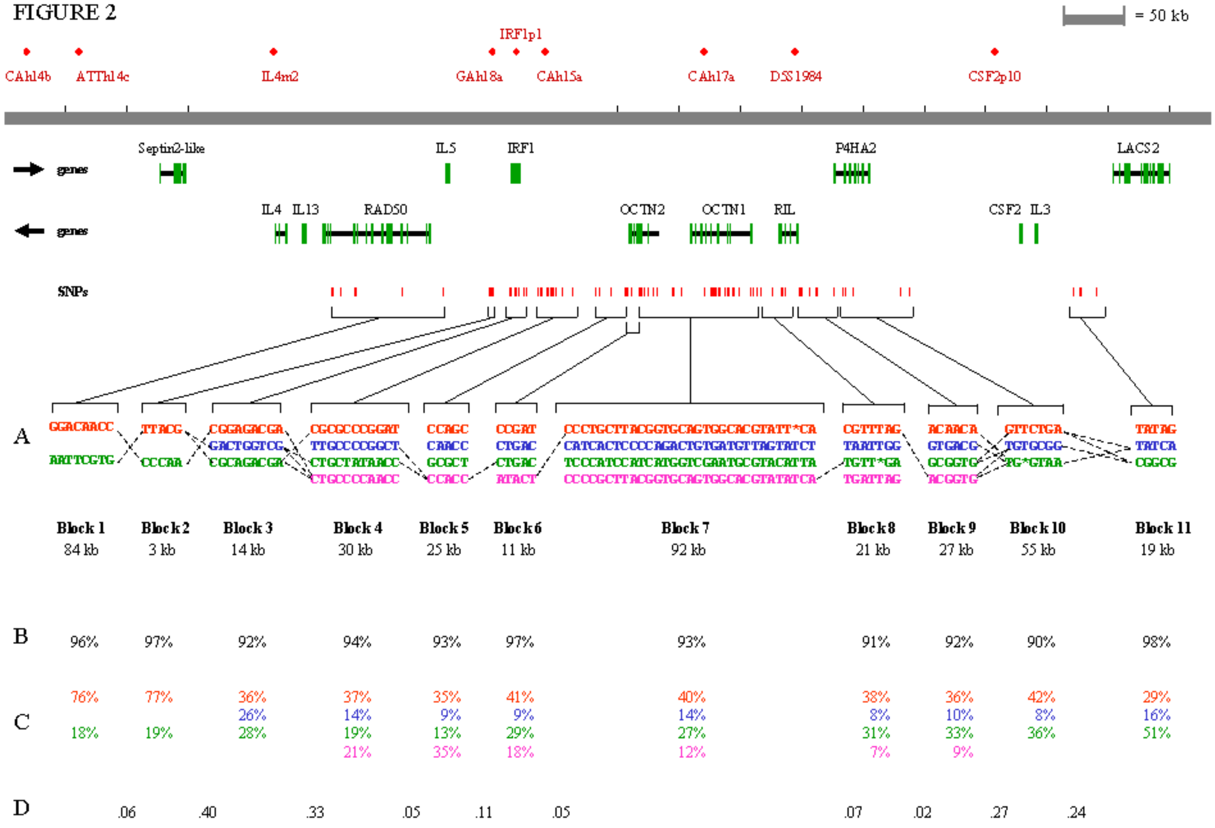
Even though this HMM method can reconstruct haplotypes relatively accurately, it would be nice if it could perform better. Hence, two potential improvements are presented. First, the current genotyping error model has only one parameter such that the error of calling the genotype AA as AB and the error of calling genotype AA as BB have the same probabilities. A potential improvement is to have different probabilities for different types of errors. For example, the probability of calling AA as BB could be much smaller. Second, for some real genetic studies, haplotypes of some individuals or at some loci may be known. This known information could be added to the current model by adjusting the scheme of sampling S (ancestral haplotype index), or by modifying $P(G|H)$ to account for it.

Besides the above improvements, this thesis research can be extended in the following aspects. First, currently the number of ancestral haplotype is fixed as a constant. The idea of using a Dirichlet process to select the ancestral haplotype number (Xing et al. 2004) may be applicable. Second, the scope of the study designs could be extended to case and control data. This may be done by using the same ancestral haplotype sets for both cases and controls, but letting the other parameters be updated differently for cases and controls. Third, to date, there have been many disease association studies using haplotype analysis, usually using a best single estimate. This approach can lead to a loss of important information among and within individuals (Stephens et al. 2001, Sham et al. 2004). Therefore, it is worthwhile to do haplotype analysis (e.g. for association studies) by incorporating the probabilities of different haplotype combinations.

Appendix A

This appendix lists Figure 2 of Daly et al. (2001) with the original figure description as follows: "Block-like haplotype diversity at 5q31. A, Common haplotype in each block of low diversity. Dashed lines indicate locations where more than 2% of all chromosomes are observed to transition from one common haplotype to a different one. B, Percentage of observed chromosomes that match one of the common patterns exactly. C, Percentage of each of the common patterns among untransmitted chromosomes. D, Rate of haplotype exchange between the blocks as estimated by the HMM. We excluded several markers at each end of the map as they provided evidence that the blocks did not continue but were not adequate to build a first or last block. In addition, four markers fell between blocks, which suggests that the recombinational clustering may not take place at a specific base-pair position, but rather in small regions".

FIGURE 2



This figure is from Daly et al. (2001). Reprinted with permisson from Nature Genetics.

Appendix B

This appendix is the parameter settings of the Hidden Markov Model.

Parameters	**Explanation**
e = 0.001	Genotyping error rate
a1 = b1 = 0.5	Prior for P(A)
hyper.T.flag="NO"	Whether using hyperparameter or not
T=rep(0.95,(L-1))	The starting value for T
alpha=9; beta=1	The starting value if using hyperprior (T)
a.0=6; b.0=1	The parameters for hyperprior (T)
alpha.beta.lower=3	The parameter for hyperprior (T)
alpha.beta.upper=16	The parameter for hyperprior (T)
hyper.m.flag="NO"	Whether using hyperparameter or not
a.m=0.8; b.m=10	Starting value if use hyperprior (m)
a.m0=1; b.m0=6	The parameters for hyperprior (m)
a.m.b.m.lower=3	The parameter for hyperprior (m)
a.m.b.m.upper=30	The parameter for hyperprior (m)
m.MAT=matrix(0.2, C, L)	The starting value
m.vector=rep(0.2,L)	The starting value
dirichlet = rep(1/C,C)	The prior for each Q_l
Q = matrix(1/C, C, L)	The starting value of Q

Appendix C

This appendix lists results of 36 runs for the three data sets. The first column of each long table is the parameter combinations, for example, "C5d0 and *m.one*", this means C=5, d=0 and the mutation model is *m.one*. The second to the seventh columns of each table are the results obtained without using hyperparameters for T and *m*. The eighth to the thirteenth columns are the results of using hyperparameters for T and *m*. For each parameter choice, there are runs with three different seeds, for each seed, the posterior samples are summarized using two methods (the label method and the minimizing switch distance method). Therefore, there are six rows in this parameter combination, the first, third and the fourth row of a parameter combination cell are the summary results obtained using the label method; the second, fourth and the sixth rows are the summary results obtained using the minimizing switch distance method. The sw.pro columns are the switch proportions.

Daly	Not using hyperparameters						Using hyperparameters					
C,d,m	mis.G	mis.M	s5	sL	sw	sw.pro	mis.G	mis.M	s5	sL	sw	sw.pro
C5d0	3	13	155	205	103	0.0397	3	13	145	180	86	0.0331
m.one	0	13	113	226	83	0.032	0	13	143	199	93	0.0358
	4	12	245	232	132	0.0509	7	14	175	201	97	0.0374
	0	12	224	266	130	0.0501	0	14	137	192	84	0.0324

	6	13	174	213	97	0.0374	2	14	169	201	96	0.037
	0	13	133	215	95	0.0366	0	14	164	233	99	0.0382
C5d0	5	13	169	213	103	0.0397	10	13	216	230	115	0.0443
m.l	0	13	161	236	100	0.0385	0	13	173	220	104	0.0401
	4	13	164	196	104	0.0401	8	13	189	239	105	0.0405
	0	13	146	251	94	0.0362	0	13	155	203	92	0.0355
	1	13	193	217	108	0.0416	9	13	203	199	93	0.0358
	0	13	171	209	103	0.0397	0	13	157	202	89	0.0343
C5d0	1	14	132	218	88	0.0339	7	13	167	219	91	0.0351
m.cl	0	14	116	201	79	0.0304	0	13	115	195	79	0.0304
	1	14	136	203	94	0.0362	11	14	189	195	96	0.037
	0	14	117	248	89	0.0343	1	13	116	194	79	0.0304
	0	13	132	192	95	0.0366	10	13	165	191	85	0.0328
	0	13	103	196	78	0.0301	0	13	117	200	81	0.0312
C5d3	5	13	198	239	119	0.0459	1	14	171	209	94	0.0362
m.one	0	13	149	230	96	0.037	0	14	162	191	93	0.0358
	8	15	179	184	91	0.0351	3	15	156	214	90	0.0347
	0	15	135	224	91	0.0351	0	15	153	219	88	0.0339
	3	14	156	211	101	0.0389	5	13	268	251	143	0.0551
	0	14	161	226	96	0.037	0	13	245	257	133	0.0513
C5d3	2	12	212	228	115	0.0443	8	13	219	215	113	0.0435
m.l	0	13	192	249	103	0.0397	0	13	187	233	105	0.0405
	1	14	178	194	100	0.0385	7	13	182	223	103	0.0397
	0	14	155	205	94	0.0362	0	13	151	222	95	0.0366
	2	13	160	201	95	0.0366	8	14	239	205	111	0.0428
	0	13	154	243	93	0.0358	0	13	191	243	107	0.0412
C5d3	1	15	143	220	91	0.0351	6	13	164	205	91	0.0351
m.cl	0	15	123	198	79	0.0304	0	13	122	221	82	0.0316
	0	13	129	190	81	0.0312	9	13	150	201	82	0.0316
	0	13	122	239	75	0.0289	0	13	108	189	79	0.0304
	0	13	124	189	80	0.0308	11	13	175	192	92	0.0355
	0	13	111	210	83	0.032	2	13	111	169	77	0.0297
C10d0	5	14	167	206	95	0.0366	3	15	190	241	111	0.0428
m.one	0	14	126	187	84	0.0324	0	15	166	206	97	0.0374
	4	15	153	199	82	0.0316	4	14	146	159	86	0.0331
	0	15	137	203	82	0.0316	0	14	128	185	81	0.0312
	6	14	172	180	92	0.0355	5	14	174	169	89	0.0343
	0	14	129	219	89	0.0343	0	14	149	203	85	0.0328
C10d0	1	13	152	180	88	0.0339	10	13	183	198	92	0.0355
m.l	0	13	131	181	84	0.0324	0	13	128	184	84	0.0324
	1	13	160	199	97	0.0374	6	13	159	226	93	0.0358
	0	13	136	200	90	0.0347	0	13	143	225	93	0.0358
	1	13	139	216	93	0.0358	5	13	200	222	104	0.0401
	0	13	115	192	82	0.0316	0	13	159	198	91	0.0351

C10d0	0	13	128	211	92	0.0355	7	13	156	200	96	0.037
m.cl	0	13	106	207	88	0.0339	1	13	112	175	83	0.032
	1	12	111	180	90	0.0347	6	12	159	221	101	0.0389
	0	12	93	156	77	0.0297	1	12	114	208	82	0.0316
	1	13	129	178	86	0.0331	7	13	163	208	92	0.0355
	0	13	105	166	75	0.0289	2	13	108	202	77	0.0297
C10d3	3	14	171	174	90	0.0347	2	14	161	188	96	0.037
m.one	0	15	139	197	88	0.0339	0	14	134	214	90	0.0347
	5	13	157	190	85	0.0328	6	14	169	166	82	0.0316
	0	14	137	191	82	0.0316	0	14	133	185	80	0.0308
	4	13	167	202	97	0.0374	7	14	174	205	91	0.0351
	0	14	156	228	91	0.0351	0	14	131	163	77	0.0297
C10d3	2	12	151	200	93	0.0358	10	13	183	192	85	0.0328
m.l	0	13	120	195	85	0.0328	0	13	143	185	83	0.032
	2	13	169	187	95	0.0366	9	13	188	219	105	0.0405
	0	13	147	200	89	0.0343	0	13	148	252	101	0.0389
	2	12	142	184	89	0.0343	5	13	152	173	84	0.0324
	0	13	116	154	82	0.0316	0	13	136	212	79	0.0304
C10d3	1	13	126	180	89	0.0343	8	13	167	191	86	0.0331
m.cl	0	13	103	189	76	0.0293	4	13	124	166	77	0.0297
	0	12	104	182	76	0.0293	3	14	143	199	85	0.0328
	0	13	101	198	76	0.0293	0	13	124	205	84	0.0324
	0	13	128	176	92	0.0355	4	13	160	205	94	0.0362
	0	13	121	186	85	0.0328	0	13	120	218	85	0.0328
C15d0	3	14	198	217	113	0.0435	2	14	156	194	91	0.0351
m.one	0	14	143	251	92	0.0355	0	14	137	194	78	0.0301
	5	15	169	208	88	0.0339	6	14	185	168	100	0.0385
	0	15	127	183	82	0.0316	0	14	137	182	86	0.0331
	3	14	160	200	86	0.0331	3	14	159	176	86	0.0331
	0	14	147	213	85	0.0328	0	14	139	186	82	0.0316
C15d0	1	13	162	192	96	0.037	2	13	144	217	93	0.0358
m.l	0	13	130	201	79	0.0304	0	13	135	220	94	0.0362
	4	13	158	168	90	0.0347	2	13	172	205	108	0.0416
	0	13	122	187	88	0.0339	0	13	162	208	103	0.0397
	3	13	199	195	106	0.0408	4	13	169	183	91	0.0351
	0	13	154	224	94	0.0362	0	13	135	210	88	0.0339
C15d0	1	11	109	177	82	0.0316	4	13	150	181	88	0.0339
m.cl	0	11	96	174	74	0.0285	0	13	115	193	76	0.0293
	1	14	149	189	91	0.0351	4	13	152	221	89	0.0343
	0	14	127	183	83	0.032	0	13	121	200	80	0.0308
	0	13	110	155	72	0.0277	7	14	184	193	91	0.0351
	0	12	110	171	74	0.0285	1	14	143	175	87	0.0335
C15d3	3	14	150	175	77	0.0297	6	13	161	171	84	0.0324
m.one	0	15	127	213	75	0.0289	0	14	135	178	79	0.0304

	3	14	142	185	80	0.0308	6	14	189	194	93	0.0358
	0	14	116	197	81	0.0312	0	14	152	168	80	0.0308
	2	14	142	174	82	0.0316	6	14	189	178	93	0.0358
	0	14	124	185	82	0.0316	0	14	158	164	86	0.0331
C15d3 m.l	2	12	154	174	94	0.0362	5	14	203	214	105	0.0405
	0	13	143	184	87	0.0335	0	14	167	196	101	0.0389
	1	12	124	184	87	0.0335	8	13	171	208	91	0.0351
	0	13	121	187	83	0.032	0	13	135	193	83	0.032
	2	12	162	178	91	0.0351	7	13	158	191	81	0.0312
	0	13	143	161	87	0.0335	0	13	114	190	76	0.0293
C15d3 m.cl	0	12	102	162	77	0.0297	4	13	148	206	93	0.0358
	0	12	96	181	78	0.0301	2	14	117	169	79	0.0304
	2	13	137	155	80	0.0308	11	12	182	170	92	0.0355
	0	12	99	170	74	0.0285	6	12	115	155	75	0.0289
	0	14	127	193	86	0.0331	4	13	153	183	82	0.0316
	0	14	119	215	81	0.0312	1	13	107	192	73	0.0281

Table A1: Results for the Daly data.

YRI	Not using hyperparameters						Using hyperparameters					
C,d,m	mis.G	mis.M	s5	sL	sw	sw.pro	mis.G	mis.M	s5	sL	sw	sw.pro
C5d0 m.one	1	2	68	99	38	0.0493	1	3	82	116	49	0.0636
	0	1	54	95	40	0.0519	0	0	77	117	51	0.0661
	3	2	79	93	43	0.0558	3	2	94	121	52	0.0674
	0	1	64	119	44	0.0571	0	0	59	94	40	0.0519
	2	2	77	88	55	0.0713	5	0	78	90	49	0.0636
	0	1	55	104	43	0.0558	0	0	56	94	45	0.0584
C5d0 m.l	0	2	82	106	46	0.0597	2	3	83	88	47	0.061
	0	2	84	126	45	0.0584	0	3	77	98	40	0.0519
	0	3	83	107	48	0.0623	3	3	90	117	50	0.0649
	0	3	94	115	53	0.0687	0	3	85	133	48	0.0623
	0	2	94	88	50	0.0649	3	6	118	128	55	0.0713
	0	2	92	92	53	0.0687	0	3	88	130	53	0.0687
C5d0 m.cl	0	4	120	129	58	0.0752	1	6	102	95	48	0.0623
	0	2	84	131	48	0.0623	0	2	72	86	45	0.0584
	0	2	81	114	48	0.0623	1	1	70	117	43	0.0558
	0	2	85	122	51	0.0661	0	0	62	98	41	0.0532
	0	4	75	96	44	0.0571	1	5	85	119	47	0.061
	0	3	74	116	43	0.0558	0	1	59	129	43	0.0558
C5d3 m.one	2	3	94	109	57	0.0739	2	2	84	170	51	0.0661
	0	0	38	87	39	0.0506	0	2	70	148	47	0.061
	2	2	81	128	52	0.0674	2	4	90	124	52	0.0674
	0	1	56	116	42	0.0545	0	0	64	146	48	0.0623
	1	3	86	115	50	0.0649	1	1	76	83	51	0.0661

	0	2	67	111	45	0.0584	0	1	68	80	44	0.0571
C5d3	0	5	104	120	53	0.0687	3	4	100	106	52	0.0674
m.l	0	1	75	112	51	0.0661	0	2	72	91	42	0.0545
	0	4	77	101	41	0.0532	4	4	114	163	61	0.0791
	0	4	76	111	45	0.0584	0	3	91	151	52	0.0674
	0	3	91	91	43	0.0558	3	3	102	111	50	0.0649
	0	2	70	86	45	0.0584	0	1	68	107	47	0.061
C5d3	0	3	85	95	45	0.0584	1	4	90	123	53	0.0687
m.cl	0	3	84	100	44	0.0571	0	3	96	128	53	0.0687
	0	5	65	106	35	0.0454	1	6	102	134	51	0.0661
	0	4	86	113	45	0.0584	0	3	72	103	45	0.0584
	0	4	97	114	57	0.0739	1	4	87	111	48	0.0623
	0	3	73	102	47	0.061	0	2	70	102	41	0.0532
C10d0	1	0	50	93	40	0.0519	5	2	87	63	40	0.0519
m.one	0	0	44	110	40	0.0519	0	0	49	78	37	0.048
	2	2	62	81	36	0.0467	1	1	83	124	47	0.061
	0	0	42	58	35	0.0454	0	1	76	90	43	0.0558
	2	0	59	115	46	0.0597	4	2	92	109	48	0.0623
	0	0	55	109	43	0.0558	0	0	63	100	43	0.0558
C10d0	0	1	82	105	54	0.07	4	2	95	121	51	0.0661
m.l	0	1	78	97	49	0.0636	0	1	71	100	45	0.0584
	1	1	69	117	46	0.0597	2	2	74	129	44	0.0571
	0	0	64	125	51	0.0661	0	0	56	122	43	0.0558
	0	3	91	85	54	0.07	4	4	110	109	50	0.0649
	0	1	77	88	50	0.0649	0	2	77	116	48	0.0623
C10d0	0	2	88	89	53	0.0687	2	3	96	103	55	0.0713
m.cl	0	2	79	110	49	0.0636	0	1	77	95	47	0.061
	0	3	76	87	44	0.0571	1	2	66	110	42	0.0545
	0	2	55	71	34	0.0441	0	0	51	108	39	0.0506
	0	2	84	113	52	0.0674	1	1	68	92	44	0.0571
	0	2	86	120	55	0.0713	0	1	71	99	44	0.0571
C10d3	1	0	43	124	34	0.0441	2	1	72	98	41	0.0532
m.one	0	0	47	120	36	0.0467	0	0	69	98	40	0.0519
	1	2	82	110	46	0.0597	2	2	100	91	47	0.061
	0	0	47	93	37	0.048	0	0	74	115	42	0.0545
	0	0	43	97	41	0.0532	3	1	77	90	47	0.061
	0	0	44	102	40	0.0519	0	0	55	105	41	0.0532
C10d3	0	3	69	51	42	0.0545	4	1	92	95	51	0.0661
m.l	0	1	63	62	41	0.0532	0	0	54	93	40	0.0519
	0	3	81	67	48	0.0623	3	0	56	98	43	0.0558
	0	1	59	65	42	0.0545	0	0	55	100	41	0.0532
	0	0	67	91	45	0.0584	2	1	69	101	43	0.0558
	0	0	63	87	42	0.0545	0	0	58	82	41	0.0532
C10d3	0	3	73	119	48	0.0623	3	3	89	102	48	0.0623

m.cl	0	2	78	117	48	0.0623	0	2	73	107	42	0.0545
	0	3	85	98	44	0.0571	1	3	90	143	54	0.07
	0	2	86	118	44	0.0571	0	2	71	131	46	0.0597
	0	3	80	81	42	0.0545	2	1	63	76	53	0.0687
	0	3	69	91	43	0.0558	0	2	61	90	44	0.0571
C15d0 m.one	2	0	62	77	41	0.0532	1	1	55	73	41	0.0532
	0	0	44	69	34	0.0441	0	2	58	91	42	0.0545
	0	0	58	106	42	0.0545	3	1	81	77	49	0.0636
	0	0	45	108	37	0.048	0	1	60	66	43	0.0558
	2	0	67	87	45	0.0584	2	1	74	91	47	0.061
	0	0	40	74	35	0.0454	0	1	63	105	47	0.061
C15d0 m.l	0	2	83	106	47	0.061	2	0	86	119	52	0.0674
	0	1	72	102	44	0.0571	0	0	68	103	43	0.0558
	0	2	74	115	49	0.0636	3	2	79	103	42	0.0545
	0	1	72	94	49	0.0636	0	0	57	80	43	0.0558
	0	4	81	78	41	0.0532	2	1	76	110	47	0.061
	0	2	88	104	53	0.0687	0	1	69	108	43	0.0558
C15d0 m.cl	0	2	90	149	55	0.0713	2	1	80	84	52	0.0674
	0	2	74	122	47	0.061	0	1	66	96	44	0.0571
	0	2	82	117	50	0.0649	2	3	82	100	45	0.0584
	0	2	82	137	50	0.0649	0	3	85	83	48	0.0623
	0	2	77	122	45	0.0584	1	1	74	127	42	0.0545
	0	2	62	110	40	0.0519	0	1	59	106	42	0.0545
C15d3 m.one	2	0	65	108	41	0.0532	2	3	92	77	47	0.061
	0	0	48	106	36	0.0467	0	0	57	66	42	0.0545
	1	0	62	91	41	0.0532	3	3	91	132	51	0.0661
	0	0	46	76	35	0.0454	0	1	84	133	55	0.0713
	2	0	58	93	36	0.0467	2	3	94	123	48	0.0623
	0	0	55	96	36	0.0467	0	1	78	107	49	0.0636
C15d3 m.l	1	4	99	84	50	0.0649	4	2	93	127	47	0.061
	0	1	62	84	41	0.0532	0	0	65	124	40	0.0519
	0	1	63	90	43	0.0558	2	2	86	109	47	0.061
	0	1	61	108	41	0.0532	0	1	70	109	46	0.0597
	0	4	93	80	42	0.0545	1	2	76	116	44	0.0571
	0	2	89	111	47	0.061	0	1	47	110	33	0.0428
C15d3 m.cl	0	4	92	88	51	0.0661	1	3	79	120	45	0.0584
	0	2	76	117	46	0.0597	0	2	60	115	40	0.0519
	0	3	86	109	47	0.061	2	3	91	118	55	0.0713
	0	3	75	126	45	0.0584	0	3	76	114	46	0.0597
	0	2	74	100	47	0.061	1	3	91	92	49	0.0636
	0	1	70	120	47	0.061	0	1	61	94	41	0.0532

Table A2: Results for the YRI data.

CEU	Not using hyperparameters						Using hyperparameters					
C,d,m	mis.G	mis.M	s5	sL	sw	sw.pro	mis.G	mis.M	s5	sL	sw	sw.pro
C5d0	0	0	6	6	4	0.0083	1	0	20	5	8	0.0167
m.one	0	0	5	1	2	0.0042	0	0	15	3	6	0.0125
	1	0	10	3	4	0.0083	2	0	20	5	7	0.0146
	0	0	5	1	2	0.0042	0	0	15	5	7	0.0146
	0	0	4	5	1	0.0021	1	0	10	2	3	0.0062
	0	0	4	7	2	0.0042	0	0	5	2	4	0.0083
C5d0	1	0	15	4	6	0.0125	1	0	25	5	9	0.0188
m.l	0	0	15	5	7	0.0146	0	0	20	6	9	0.0188
	0	0	15	5	7	0.0146	3	0	21	5	7	0.0146
	0	0	15	3	6	0.0125	2	0	10	4	5	0.0104
	0	0	15	3	6	0.0125	1	0	14	4	6	0.0125
	0	0	15	3	6	0.0125	0	0	10	2	4	0.0083
C5d0	0	0	0	1	1	0.0021	1	0	7	2	3	0.0062
m.cl	0	0	0	0	0	0	2	0	5	1	2	0.0042
	0	0	0	2	1	0.0021	1	0	10	3	4	0.0083
	0	0	0	2	1	0.0021	0	0	1	8	3	0.0062
	0	0	0	1	2	0.0042	1	0	10	3	4	0.0083
	0	0	0	2	1	0.0021	0	0	5	2	3	0.0062
C5d3	0	0	5	1	2	0.0042	3	0	20	4	5	0.0104
m.one	0	0	5	1	2	0.0042	0	0	10	9	4	0.0083
	0	0	4	12	2	0.0042	2	0	20	4	6	0.0125
	0	0	4	12	2	0.0042	0	0	20	4	8	0.0167
	1	0	10	2	3	0.0062	2	0	15	4	5	0.0104
	0	0	5	1	2	0.0042	0	0	15	4	7	0.0146
C5d3	0	0	20	4	8	0.0167	1	0	30	6	11	0.0229
m.l	0	0	20	4	8	0.0167	1	0	25	5	10	0.0208
	0	0	10	3	6	0.0125	2	0	19	4	6	0.0125
	0	0	10	4	5	0.0104	0	0	10	3	6	0.0125
	0	0	15	3	6	0.0125	2	0	24	5	8	0.0167
	0	0	15	5	7	0.0146	1	0	15	5	7	0.0146
C5d3	0	0	0	1	1	0.0021	0	0	0	0	0	0
m.cl	0	0	0	0	0	0	0	0	0	0	0	0
	0	0	0	1	2	0.0042	1	0	10	2	3	0.0062
	0	0	0	1	2	0.0042	1	0	5	1	2	0.0042
	0	0	0	2	1	0.0021	0	0	0	0	0	0
	0	0	0	2	1	0.0021	0	0	0	0	0	0
C10d0	0	0	5	1	2	0.0042	2	0	24	5	8	0.0167
m.one	0	0	5	1	2	0.0042	0	0	15	4	8	0.0167
	0	0	5	1	2	0.0042	1	0	20	4	7	0.0146
	0	0	5	1	2	0.0042	0	0	15	4	7	0.0146
	0	0	5	3	3	0.0062	2	0	20	4	6	0.0125

	0	0	5	1	2	0.0042	0	0	15	3	6	0.0125
C10d0	0	0	10	4	5	0.0104	1	0	14	3	5	0.0104
m.l	0	0	10	2	4	0.0083	0	0	10	2	4	0.0083
	0	0	15	4	7	0.0146	1	0	24	5	9	0.0188
	0	0	10	2	4	0.0083	1	0	20	5	9	0.0188
	0	0	15	4	8	0.0167	2	0	19	4	6	0.0125
	0	0	15	3	6	0.0125	0	0	10	2	4	0.0083
C10d0	0	0	0	2	1	0.0021	1	0	4	1	1	0.0021
m.cl	0	0	0	1	1	0.0021	0	0	0	1	1	0.0021
	0	0	0	1	2	0.0042	2	0	14	3	4	0.0083
	0	0	0	1	1	0.0021	0	0	5	1	2	0.0042
	0	0	0	0	0	0	2	0	14	3	4	0.0083
	0	0	0	1	2	0.0042	0	0	5	1	2	0.0042
C10d3	0	0	5	1	2	0.0042	0	0	5	1	2	0.0042
m.one	0	0	5	1	2	0.0042	0	0	5	1	2	0.0042
	0	0	5	1	2	0.0042	0	0	5	1	2	0.0042
	0	0	5	1	2	0.0042	0	0	5	1	2	0.0042
	0	0	5	3	3	0.0062	0	0	5	1	2	0.0042
	0	0	5	3	3	0.0062	1	0	5	1	2	0.0042
C10d3	0	0	20	5	9	0.0188	0	0	15	3	6	0.0125
m.l	0	0	20	6	9	0.0188	0	0	15	3	6	0.0125
	0	0	10	2	4	0.0083	1	0	15	3	5	0.0104
	0	0	10	3	5	0.0104	1	0	10	2	4	0.0083
	0	0	10	3	5	0.0104	0	0	5	1	2	0.0042
	0	0	10	3	5	0.0104	0	0	5	1	2	0.0042
C10d3	0	0	0	2	1	0.0021	0	0	5	1	2	0.0042
m.cl	0	0	0	1	2	0.0042	1	0	5	1	2	0.0042
	0	0	0	1	1	0.0021	0	0	0	0	0	0
	0	0	0	1	2	0.0042	0	0	0	0	0	0
	0	0	0	2	1	0.0021	1	0	5	1	1	0.0021
	0	0	0	0	0	0	1	0	0	0	0	0
C15d0	0	0	5	3	3	0.0062	2	0	15	4	5	0.0104
m.one	0	0	5	3	3	0.0062	0	0	5	1	2	0.0042
	0	0	5	3	3	0.0062	1	0	10	3	5	0.0104
	0	0	5	3	3	0.0062	0	0	10	4	5	0.0104
	0	0	5	1	2	0.0042	1	0	9	2	3	0.0062
	0	0	5	3	3	0.0062	0	0	5	2	3	0.0062
C15d0	0	0	10	4	5	0.0104	2	0	14	3	4	0.0083
m.l	0	0	10	4	5	0.0104	0	0	5	2	3	0.0062
	0	0	10	4	5	0.0104	2	0	29	6	10	0.0208
	0	0	10	4	5	0.0104	0	0	20	6	9	0.0188
	0	0	20	5	9	0.0188	1	0	14	3	5	0.0104
	0	0	20	4	8	0.0167	0	0	10	4	5	0.0104
C15d0	0	0	0	1	1	0.0021	2	0	9	2	2	0.0042

m.cl	0	0	0	1	1	0.0021	0	0	0	1	2	0.0042
	0	0	0	0	0	0	2	0	14	3	4	0.0083
	0	0	0	2	1	0.0021	0	0	5	2	4	0.0083
	0	0	0	2	1	0.0021	1	0	5	2	2	0.0042
	0	0	0	1	2	0.0042	0	0	0	1	2	0.0042
C15d3 m.one	0	0	5	1	2	0.0042	1	0	10	2	3	0.0062
	0	0	5	1	2	0.0042	1	0	5	1	2	0.0042
	0	0	5	1	2	0.0042	0	0	5	1	2	0.0042
	0	0	5	1	2	0.0042	0	0	5	1	2	0.0042
	0	0	5	1	2	0.0042	1	0	10	2	3	0.0062
	0	0	5	1	2	0.0042	0	0	5	1	2	0.0042
C15d3 m.l	0	0	15	3	6	0.0125	2	0	19	4	6	0.0125
	0	0	15	3	6	0.0125	0	0	10	3	6	0.0125
	0	0	10	3	5	0.0104	2	0	25	5	8	0.0167
	0	0	10	4	5	0.0104	2	0	15	3	6	0.0125
	0	0	15	5	7	0.0146	1	0	20	4	7	0.0146
	0	0	15	4	8	0.0167	1	0	15	3	6	0.0125
C15d3 m.cl	0	0	0	0	0	0	1	0	10	2	3	0.0062
	0	0	0	0	0	0	1	0	5	1	2	0.0042
	0	0	5	3	3	0.0062	1	0	7	2	3	0.0062
	0	0	5	3	3	0.0062	1	0	5	1	2	0.0042
	0	0	0	2	1	0.0021	1	0	5	3	2	0.0042
	0	0	0	0	0	0	0	0	0	1	2	0.0042

Table A3: Results for the CEU data.

Bibliography

Abecasis, G. R., Cherny, S. S. and Cardon, L. R. (2001). The impact of genotyping error on family-based analysis of quantitative traits, *European Journal of Human Genetics* **9**: 130–134.

Abecasis, G. R., Herny, S. S., Cookson, W. O. and Cardon, L. R. (2002). Merlin-rapid analysis of dense genetic maps using sparse gene flow trees, *Nature genetics* **30**: 97–101.

Akey, J., Jin, L. and Xiong, M. (2001). Haplotypes vs single marker linkage disequilibrium tests: what do we gain?, *European Journal of Human Genetics* **9**: 291–300.

Albert, J. H. and Chib, S. (1993). Bayesian inference via Gibbs sampling of autoregressive time series subject to Markov mean and variance shifts, *Journal of Business and Economic Statist* **11**: 1–15.

Bafna, V., Gusfield, D., Lancia, G. and Yooseph, S. (2003). Haplotyping as perfect phylogeny: A direct approach, *Journal of Computational Biology* **10**(3): 323–340.

Baker, J. K. (1975). The dragon system - an overview, *IEEE Transactions on. Acoustic Speech Signal Processing* **ASSP-23**(1): 24–29.

Baum, L. E. (1972). An inequality and associated maximization technique in statistical estimation for probabilistic functions of Markov processes, *Inequalities* **3**: 1–8.

Baum, L. E. and Eagon, J. A. (1967). An inequality with applications to statistical estimation for probabilistic functions of Markov processes and to a model for ecology, *Bulletin of the American Mathematical Society* **73**: 360–363.

Baum, L. E. and Petrie, T. (1966). Statistical inference for probabilistic functions of finite state Markov chains, *Annals of Mathematical Statistics* **37**: 1554–1563.

Baum, L. E. and Sell, G. R. (1968). Growth functions for transformation on manifolds, *Pacific Journal Mathematics* **27**(2): 211–227.

Baum, L. E., Petrie, T., Soules, G. and Weiss, N. (1970). A maximization technique occurring in the statistical analysis of probabilistic functions of Markov chains, *The Annals of Mathematical Statistics* **41**(1): 164–171.

Becker, T. and Knapp, M. (2005). Impact of missing genotype data on Monte-Carlo simulation based haplotype analysis, *Human Heredity* **59**(4): 185–189.

Brown, D. and Harrower, I. (2005). A new formulation for haplotype inference by pure parsimony, *Technical Report CS-2005-004, School of Computer Science, University of Waterloo.*

Buetow, K. H. (1991). Influence of aberrant observations on high-resolution linkage analysis outcomes, *American Journal of Human Genetics* **49**(5): 985–994.

Butt, C., Sun, S., Peddle, L., Greenwood, C., Hamilton, S., D., G. and P., R. (2005). Association of nuclear factor-kb in psoriatic arthritis, *Journal of Rheumatology* **32**: 1742–1744.

Chapman, N. H. and Thompson, E. A. (2001). LD mapping: the role of population history, size and structure, *Advances in Genetics* **42**(413): 435.

Chung, R. H. and Gusfield, D. (2003). Perfect phylogeny haplotyper: haplotype inferral using a tree model, *Bioinformatics* **19**(6): 780–781.

Churchill, G. A. (1989). Stochastic models for heterogeneous DNA sequences, *Bulletin of Mathematical Biology* **51**: 79–94.

Clark, A. G. (1990). Inference of haplotypes from PCR-amplified samples of diploid populations, *Molecular Biology and Evolution* **7**(2): 111–122.

Daly, M. J., Rioux, J. D., Schaffner, S. F., Hudson, T. and Lander, E. (2001). High-resolution haplotype structure in the human genome, *Nature Genetics* **29**: 229–232.

Damaschke, P. (2003). Fast perfect phylogeny haplotype inference, *14th International Symposium on Fundamentals of Computation Theory FCT* **2751**: 183–194.

Dempster, A. P., Laird, N. M. and Rubin, D. B. (1977). Maximum likelihood from incomplete data via the EM algorithm, *Journal of Royal Statistical Society Series B* **39**(1): 1–38.

Eronen, L., Geerts, F. and Toivonen, H. (2004). A Markov chain approach to reconstruction of long haplotypes, *Pacific Symposium on Biocomputing* **9**: 104–115.

Excoffier, L. and Slatkin, M. (1995). Maximum-likelihood estimation of molecular haplotype frequencies in a diploid population, *Molecular Biology and Evolution* **12**(5): 921–927.

Fallin, D. and Schork, N. J. (2000). Accuracy of haplotype frequency estimation for biallelic loci, via the expectation-maximisation algorithm for unphased diploid genotype data, *American Journal of Human Genetics* **67**(4): 947–959.

Foulkes, W. D., Thiffault, I., Gruber, S. B., Horwitz, M., Hamel, N., Lee, C., Shia, J., Markowitz, A., Figer, A., Friedman, E., Farber, D., Greenwood, C. M., Bonner, J. D., Nafa, K., Walsh, T., Marcus, V., Tomsho, L., Gebert, J., Macrae, F. A., Gaff, C. L., Paillerets, B. B., Gregersen, P. K., Weitzel, J. N., Gordon, P. H., MacNamara, E. King, M. C., Hampel, H., De La Chapelle, A., Boyd, J., Offit, K., Rennert, G., Chong, G. and Ellis, N. A. (2002). The founder mutation MSH2*1906G>C is an important cause of hereditary nonpolyposis colorectal cancer in the Ashkenazi Jewish population, *American Journal of Human Genetics* **71**(6): 1395–1412.

Gelfand, A. E. and Smith, A. F. M. (1990). Sampling-based approaches to calculating marginal densities, *Journal of the American Statistical Association* **85**: 398–409.

Gelman, A., Carlin, J. B., Stern, H. S. and Rubin, D. B. (1995). *Bayesian Data Analysis*, 1st edn, Chapman & Hall, Canberra.

Geman, S. and Geman, D. (1984). Stochastic relaxation, Gibbs distributions, and the Bayesian restoration of images, *IEEE Transactions on Pattern Analysis and Machine Intelligence* **6**: 724–741.

Greenspan, G. and Geiger, D. (2004). High density linkage disequilibrium mapping using models of haplotype block variation, *Bioinformatics* **20**(Suppl 1): 137–144.

Griffiths, A. J., Wessler, S. R., Lewontin, R. C., Gelbart, W. M., Suzuki, D. T. and Miller, J. H. (2005). *Introduction to Genetic Analysis*, 8th edn, W. H. Freeman and Company, New York.

Gusfield, D. (2003). Haplotype inference by pure parsimony, *Technical Report CSE-20032, Department of Computer Science, University of California.*

Gusfield, D. and Orzack, S. H. (2005). Haplotype inference, *CRC Handbook on Bioinformatics.*

Halperin, E. and Eskin, E. (2004). Haplotype reconstruction from genotype data using imperfect phylogeny, *Bioinformatics* **20**(12): 1842–1849.

Hastings, W. (1970). Monte carlo sampling methods using Markov chains and their applications, *Biometrika* **57**: 97–109.

Hawley, M. E. and Kidd, K. K. (1995). HAPLO: a program using the EM algorithm to estimate the frequenciesof multi-site haplotypes, *The Journal of heredity* **86**: 409–411.

Huang, Y. T., Chao, K. M. and Chen, T. (2005). An approximation algorithm for haplotype inference by maximum parsimony, *Journal of computational biology* **12**(10): 1261–1274.

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

Juang, B. H. and Rabiner, L. R. (1991). Hidden Markov models for speech recognition, *Technometrics* **33**(3): 251–272.

Kelly, E. D., Sievers, F. and McManus, R. (2004). Haplotype frequency estimation error analysis in the presence of missing genotype data, *BMC Bioinformatics*.

Kimmel, G. and Shamir, R. (2005). A block-free hidden Markov model for genotypes and its application to disease association, *Journal of Computational Biology* **12**(10): 1243–1260.

Kirk, K. M. and Cardon, L. R. (2002). The impact of genotyping error on haplotype reconstruction and frequency estimation, *European Journal of Human Genetics* **10**(6): 616–622.

Koski, T. (2001). *Hidden Markov Models for Bioinformatics*, Kluwer Academic Publishers.

Kruglyak, L., Daly, M. J., Reeve-Daly, M. P. and Lander, E. S. (1996). Paremetric and nonparametric linkage analysis: a unified multipoint approach, *American Journal of Human Genetics* **58**: 1347–1363.

Lin, S., Cutler, D. J., Zwick, M. E. and Chakravarti, A. (2002). Haplotype inference in random population samples, *American Journal of Human Genetics* **71**: 1129–1137.

Liu, J. S., Neuwald, A. F. and Lawrence, C. E. (1999). Markovian structures in biological sequence alignments, *Journal of the American Statistical Association* **94**: 1–15.

Liu, N., Beerman, I., Lifton, R. and Zhao, H. (2006). Haplotype analysis in the presence of informatively missing genotype data, *Genetic Epidemiology* **30**(4): 290–300.

Long, J. C., Williams, R. C. and Urbanek, M. (1995). An EM algorithm and testing strategy for multiple-locus haplotypes, *American Journal of Human Genetics* **56**(3): 799–810.

Mander, A. P. (2001). Haplotype analysis in population-based association studies, *Stata Journal* **1**(1): 58–75.

Marchini, J., Cutler, D., Patterson, N., Stephens, M., Eskin, E., Halperin, E., Lin, S., Qin, Z. Munro, H. M., Abecasis, G. R., Donnelly, P. and Consortium, I. H. (2006). A comparison of phasing algorithms for trios and unrelated individuals, *American Journal of Human Genetics* **78**: 437–450.

Metropolis, N., Rosenbluth, A., Rosenbluth, M., Teller, A. H. and Teller, E. (1953). Equation of state calculations by fast computing machines, *Journal of Chemical Physics* **21**: 1087–1092.

Neal, R. M. (2003). Slice sampling, *Annals of Statistics* **31**(3): 705–767.

Neuhausen, S. L., Mazoyer, S., Friedman, L., Stratton, M., Offit, K., Caligo, A., Tomlinson, G., Cannon-Albright, L., Bishop, T., Kelsell, D., Solomon, E., Weber, B., Couch, F., Struewing, J., Tonin, P., Durocher, F., Narod, S., Skolnick, M. H., Lenoir, G., Serova, O., Ponder, B., Stoa-Lyonnet, D., Easton, D., King, M. C. and Goldgar, D. (1996). Haplotype and phenotype analysis of six recurrent BRCA1 mutations in 61 families: results of an international study, *American Journal of Human Genetics* **58**(2): 271–280.

Niell, B. A., Long, J. C., Rennert, G. and Gruber, S. B. (2003). Genetic anthropology of the colorectal cancer-susceptibility allele APC I1307K: evidence of genetic drift within the Ashkenazim, *American Journal of Human Genetics* **73**: 1250–1260.

Niu, T., Qin, Z., Xu, X. and Liu, J. (2002). Bayesian haplotype inference for multiple linked Single-Nucleotide Polymorphisms, *American Journal of Human Genetics* **70**: 157–169.

Polanska, J. (2003). The EM algorithm and its implementation for the estimation of frequencies of SNP-haplotypes, *International Journal of Applied Mathematics and Computer Science* **13**(3): 419–429.

Pritchard, J. K. (2001). Are rare variants responsible for susceptibility to complex disease?, *American Journal of Human Genetics* **69**: 124–137.

Qin, Z., Niu, T. and Liu, J. (2002). Partition-Ligation Expectation-Maximization algorithm for haplotype inference with Single-Nucleotide Polymorphisms, *American Journal of Human Genetics* **71**: 1242–1247.

Quade, S. R., Elston, R. C. and Goddard, K. A. (2005). Estimating haplotype frequencies in pooled dna samples when there is genotyping error, *BMC Genetics*.

Rabiner, L. R. (1989). A tutorial on hidden Markov models and selected applications in speech recognition, *Proceedings of the IEEE* **77**(2): 257–285.

Rastas, P., Koivisto, M., Mannila, H. and Ukkonen, E. (2005). A hidden Markov technique for haplotype reconstruction, *Lecture Notes, Computer Science 3692* pp. 140 –151.

Romberg, J., Choi, H. and Baraniuk, R. (2001). Bayesian tree-structured image modeling using wavelet-domain hidden Markov models, *IEEE Transactions on image processing* **10**(7): 1056–1068.

Scheet, P. and Stephens, M. (2006). A fast and flexible statistical model for large-scale population genotype data: applications to inferring missing genotypes and haplotypic phase, *American Journal of Human Genetics* **78**(4): 629–644.

Scott, S. L. (2002). Bayesian methods for hidden Markov models: Recursive computing in the 21st century, *Journal of the American Statistical Society* **97**: 337–351.

Sham, P. C., Rijsdijk, F. V., Knight, J., Makoff, A., North, B. and Curtis, D. (2004). Haplotype association analysis of discrete and continuous traits using mixture of regression models, *Behavior Genetics* **30**: 285–293.

Siepel, A. and Haussler, D. (2004). Combining phylogenetic and hidden Markov models in biosequence analysis, *Journal of computational biology* **11**(2-3): 413–428.

Sobel, E., Papp, J. C. and Lange, K. (2002). Detection and integration of genotyping errors in statistical genetics, *American Journal of Human Genetics* **70**: 496–508.

Stephens, M. and Donnelly, P. (2003). A comparison of Bayesian methods for haplotype reconstruction from population, *American Journal of Human Genetics* **73**: 1162–1169.

Stephens, M. and Scheet, P. (2005). Accounting for decay of linkage disequilibrium in haplotype inference and missing-data imputation, *American Journal of Human Genetics* **76**(3): 449–462.

Stephens, M., Smith, N. and Donnelly, P. (2001). A new statistical method for haplotype reconstruction from population data, *American Journal of Human Genetics* **68**: 978–989.

Sun, S., Greenwood, C. and Neal, R. (2004). Haplotype inference using a hidden Markov model with efficient Markov chain sampling [abstract 2934], *Proceedings and abstracts of the American Society of Human Genetics 2004 Annual meeting.*

The International HAPMAP Consortium (2003). The international HapMap project, *Nature* **426**: 789–796.

The International HAPMAP Consortium (2004). Integrating ethics and science in the international hapmap project, *Nature Reviews Genetics* **5**: 467–475.

The International HAPMAP Consortium (2005). A haplotype map of the human genome, *Nature* **437**: 1299–1320.

Viterbi, A. J. (1967). Error bounds for convolutional codes and an asymptotically optimal decoding algorithm, *IEEE Transactions on Information Theory,* **IT-13**: 260–269.

Wang, L. and Xu, Y. (2003). Haplotype inference by maximum parsimony, *Bioinformatics* **19**(14): 1773–1780.

Xing, E., Sharan, R. and Jordan, M. (2004). Bayesian haplotype inference via the Dirichlet process, *Proceedings of the 21st International Conference on Machine Learnings.*

Yuan, Z. Q., Wong, N., Foulkes, W. D., Alpert, L., Manganaro, F., Andreutti-Zaugg, C., Iggo, R., Anthony, K., Hsieh, E., Redston, M., Pinsky, L., Trifiro, M., Gordon, P. H. and Lasko, D. (1999). A missense mutation in both hMSH2 and APC in an Ashkenazi Jewish HNPCC kindred: implications for clinical screening, *Journal of Medical Genetics* **36**: 790–793.

Zaykin, D. V., Westfall, P. H., Young, S. S., Karnoub, M. A., Wagner, M. J. and Ehm, M. G. (2002). Testing association of statistically inferred haplotypes with discrete and continuous traits in samples of unrelated individuals, *Human Heredity* **53**(2): 79–91.

Zhao, J., Lissarrague, S., Essioux, L. and Sham, P. C. (2002). Genecounting: haplotype analysis with missing genotypes, *Bioinformatics* **18**(12): 1694–1695.

Zhu, X., Zhang, S., Kan, D. and Cooper, R. (2004). Haplotype block definition and its application, *Pacific Symposium on Biocomputing* **9**: 152–163.