Bayesian Mixture Modelling of Migration by Founder Analysis

Noel Thomson

A dissertation submitted to the

University of Glasgow

for the degree of

Doctor of Philosophy

Department of Statistics

September 2009

Declaration

This thesis has been composed by myself and it has not been submitted in any previous application for a degree. The work reported within was executed by myself, unless otherwise stated.

September 2009

Acknowledgements

I would like to acknowledge the help and support of my supervisor, Dr. Vincent Macaulay. As the first PhD candidate under his supervision, I am sure I presented a unique challenge to supervise. His encouragement and support continued throughout my studies, particularly during the more difficult periods of my research.

I would like to acknowledge the help and support provided by Prof. Martin Richards (University of Leeds). Without his original work in the area of founder analysis, the ideas developed in this thesis would never have been possible. Further, his personal input on the method I propose in this thesis helped shape the method to allow many of the wishes of expert scientists to be properly considered and accommodated in my approach. Without such input this thesis would represent little more than statistical ideas and methods, with many of the practical data analysis wishes of the expert scientists being overlooked. I also would like to acknowledge Prof. Richards for providing the data that is analysed in this thesis, while additionally providing expert advice which aided prior specification.

I would like to acknowledge the funding support of The Carnegie Trust (for the Universities of Scotland), who awarded me a Carnegie Scholarship to allow this research to be undertaken.

Prof. Adrian Bowman, University of Glasgow, took on the role as my second supervisor and provided encouragement and support throughout my studies, I would like to acknowledge his continuing support.

I would like to more generally acknowledge the encouragement of all the staff in the Department of Statistics at the University of Glasgow.

Finally, I would like to acknowledge the encouragement and support of my family.

Abstract

In this thesis a new method is proposed to estimate major periods of migration from one region into another using phased, non-recombined sequence data from the present. The assumption is made that migration occurs in multiple waves and that during each migration period, a number of sequences, called 'founder sequences', migrate into the new region. It is first shown through appropriate simulations based on the structured coalescent that previous inferences based on the idea of founder sequences suffer from the fundamental problem that it is assumed that migration events coincide with the nodes (coalescent events) of the reconstructed tree. It is shown that such an assumption leads to contradictions with the assumed underlying migration process, and that inferences based on such a method have the potential for bias in the date estimates obtained.

An improved method is proposed which involves 'connected star trees', a tree structure that allows the uncertainty in the time of the migration event to be modelled in a probabilistic manner. Useful theoretical results under this assumption are derived. To model the uncertainty of which founder sequence belongs to which migration period, a Bayesian mixture modelling approach is taken, inferences in which are made by Markov Chain Monte Carlo techniques.

Using the developed model, a reanalysis of a dataset that pertains to the settlement of Europe is undertaken. It is shown that sensible inferences can be made under certain conditions using the new model. However, it is also shown that questions of major interest cannot be answered, and certain inferences cannot be made due to an inherent lack of information in any dataset composed of sequences from the present day. It is argued that many of the major questions of interest regarding the migration of modern day humans into Europe cannot be answered without *strong* prior assumptions being made by the investigator. It is further argued that the same reasons that prohibit certain inferences from being made under the proposed model would remain in *any* method which has similar assumptions.

Contents

Chapter 1

Introduction

1.1 Approaches to migration estimation

In this chapter a brief overview of some of the various approaches used to estimate migration from DNA sequence data will be presented. It is noted from the outset that although an attempt has been made to place the various approaches into categories such as likelihood-based, or Bayesian, many do not neatly belong to only a single category. It is also mentioned that this thesis will make a strong assumption about the underlying migration process, and as a consequence, most approaches discussed in this section would not be appropriate for analysing a dataset believed to have arisen from the migration process that will be later assumed. For this reason, only an overview is given in what follows and the interested reader is directed towards the referenced works for further details. The methods proposed in the literature often require very strong prior modelling assumptions, and are often based on a single approach to analysis (e.g. pure likelihood *or* Bayesian). As a result, very few investigators have been able to assess the merits of the common approaches due to issues such as differences in assumptions and parameters.

One interesting comparison, however, is the work of Kuhner and Smith [1], who compare the Bayesian and likelihood versions of LAMARC. Although it is of interest to assess the performance of the technical approaches to the inference problem when possible, such assessment is difficult and still relatively infrequent in the literature.

Phylogenetic methods

Rosenberg and Nordborg [2] make the interesting distinction between modern day genealogical methods, such as those based on coalescent theory, with phylogenetic methods stating (page 383) that "Phylogenetic methods estimate trees. They were developed to determine the pattern of species descent, which is assumed to be tree like".

The point is simply made that phylogenetic methods were designed to allow species trees to be estimated, and these methods depend on the existence of a strong correlation between species trees and gene trees. For this reason, no further comments will be made about classical phylogenetic methods and focus will be directed towards genealogical methods, where the interest is (usually) the estimation of parameters (such as migration rates) which give rise to phylogenetic trees. The actual reconstructed tree is nothing more than a (high-dimensional) nuisance parameter. Furthermore, a single reconstructed tree is an extremely difficult object about which to design statistical hypothesis tests, and this reason alone limits the usefulness of such methods if taking an approach which depends entirely on phylogenetic methods.

One particularly interesting phylogenetic approach, however, is that of Nested-Clade-Analysis (NCA), by Templeton et al. [3] This approach involves estimating the 'haplotype network' of a given sample. The algorithm used to

construct such networks attempts to use parsimony but allows nonparsimonious connections when the parsimonious reconstructions have low (≤ 0.95) probability of being true. Templeton and colleagues suggest a 95% plausible set of networks is created (which may include nonparsimonious networks). The plausible set identified is then subject to various rules which aim to split the haplotypes into specific groups (the"0-step clades, 1-step clades,..." etc.), with the members of each group being composed of members of the previous group that are only a single mutation apart. Nested groups of haplotypes are identified and a set of physical distance values are calculated which are then used to 'test' whether samples from the same population are closer to each other than would be expected by chance. This is done through permutation methods to simulate the distribution of the distance measures under the null hypothesis of no geographical associations. Once evidence of geographical structure has been identified an inference key is used to identify the demographic factor responsible. This approach is both novel and appealing on grounds of simplicity (the flow-chart type explanation of the method [3, page 781-782] is particularly unique) and it is an interesting example of an attempt at a quantitative phylogenetic approach to the inference problem. However, the method has been shown to lead to invalid conclusions [4].

Methods which rely purely on tree/network reconstructions are of limited usefulness in many areas of statistical genetics, where formal inferential methods are now generally preferred. These more formal methods attempt to make parameter estimates from a model which allows many of the stochastic features of the true evolutionary process to be accounted for.

Methods based on summary statistics

Various authors have proposed test statistics which, at certain values, can *suggest* migration, and in certain circumstances, allow *subjective* inferences to be made. Such a statistic is Tajima's D [5] which is based on the (normalised) difference between two different estimators of the (scaled) mutation rate (commonly denoted by θ). The two estimators used in Tajima's D are the average number of pairwise differences between two sequences ($\hat{\pi}$) and the Watterson estimator ($\hat{\theta}_W$), and the (normalised) difference between these two estimators is used as the test statistic. Using the notation of Hein et al. [6], the quantities e_1 and e_2 are constants depending on sample size, and S_n is the number of mutations in a sample assumed to follow the infinite sites assumption. Then D is defined by

$$D = \frac{\hat{\pi} - \hat{\theta}_W}{\sqrt{e_1 S_n + e_2 S_n (S_n - 1)}}. \tag{1.1}$$

Under the assumptions of a basic coalescent model, Tajima's D statistic should have a mean close to zero and variance close to one (although its distribution is not normal and in fact is close to that of a beta distribution), while certain departures from the basic coalescent assumptions (such as the presence of migration) can result in the distribution of Tajima's D being changed and hence a means of testing the basic coalescent.

The problem with such methods is that *completely* different demographic scenarios can have an *identical* effect on summary test statistics. For example, Tajima's D being positive (on average) happens in *any* demographic scenario that gives $E[\hat{\pi}] > E[\hat{\theta}_W]$, such as a recent population bottleneck or with limited migration between two populations (as e.g. shown in [6]).

Some authors have calculated explicit expressions for some more complicated

statistics which they then use to assess some population parameters. Wakeley et al. [7] derive expressions for the expected number of different categories of polymorphic sites for an *isolation* model, which is the term used to describe a model where a single population splits into two descendent populations, and for the *size-change model*, where the ancestral population simply changes size. Although this model involves no migration, for the isolation model, Wakeley and colleagues derive some complicated expressions for the expected values of the partitions of segregating sites in the ancestral population, which they showed to be functions of the parameters in the model. They then use numerical methods to find the values of the parameters that make the observed values closely match the expected values. Such approaches can be viewed as being similar to summary statistic methods, and more generally to *the method of moments*.

It is perhaps important to note here that methods based exclusively on summary statistics of some aspect of the data do have the advantage of ease of computation, are often simple to understand, and can provide useful insights into given datasets which can help direct an investigator towards a more appropriate, involved analysis such as those to be described in the following sections. Summary-statistic-based methods also have made their way into more formal methods in what is now commonly referred to as Approximate Bayesian Computation (ABC) [8]. Of course, using summary statistics involves loss of information: reducing what is a high dimensional dataset into a single statistic (or vector of statistics) is always going to involve loss of information from a dataset, unless such an estimator was a sufficient statistic. In almost all genetic contexts of major interest, no sufficient statistics are known to exist, with the exception of the number of segregating sites for the estimation of θ using the Ewens sampling formula [9].

Likelihood based approaches

Various authors have attempted likelihood-based inference for models with migration, the likelihood, L, being defined as the probability of the observed data, D given the parameters of the model, Λ, and any nuisance parameters, G (which could be considered as part of the parameter vector but are separated here as one may wish to integrate them out).

$$L(\Lambda) = P(D|\Lambda) = \int_G P(D|G,\Lambda)P(G|\Lambda)dG \tag{1.2}$$

In practice one would like to maximise the likelihood, which requires averaging over all possible values of the nuisance parameters. The parameters of interest may be quantities such as the effective population size, an exponential growth rate, a global migration rate, or any number of possible model-dependent parameters. In practice, the nuisance parameter is typically the phylogeny: it is this object that causes the most problems for the statistician. The space of plausible trees is extremely large and averaging over the possible phylogenies is an extremely difficult problem. It is for this reason that methods have been developed which assume a fixed tree and then make formal statistical inferences with the inherent assumption that the assumed tree is correct and the uncertainty in the tree reconstruction can be ignored. An example of such an approach is an early method of Slatkin et al. [10], which gives an estimate of the population migration rate between a pair of populations from the branching patterns that are present on the reconstructed tree.

Some attempts at the problem of averaging over gene trees have brought some success [11], [12]. The problems with models which attempt to average over all possible genealogies (or a subset of them) is that of computational complexity. Most of these methods are extremely computationally demanding,

can take long periods to run for some datasets, while there is the additional problem that these methods can only estimate parameters of well-defined models where appropriate formulas for likelihoods can be calculated (or approximated easily).

Difficult inference issues in migration models

I have touched on some of the approaches taken but some issues remain which are regularly ignored in modelling and analysis. One is that of ascertainment bias, which can be described as a departure from what one would expect to see in a *random* sample of genetic data, due to the data collection/ascertainment process. Wakeley and colleagues [13] investigate ascertainment bias and show it to have negative effects on the inference of migration rate parameters which are described as 'substantially overestimated when ascertainment bias is ignored' (as well as other population parameters such as population size changes - false signals of population expansion were even shown to result from ascertainment bias). This issue is a troublesome one which is rarely considered, but one should be aware of such bias being possible. However, it is perhaps likely to affect SNP (single nucleotide polymorphism) data more severely than sequence data, since rare alleles are less likely to be missed in the latter case.

A further, perhaps more complicated issue with migration rate estimation is that of 'ghost populations'. It is commonly assumed in models with migration that k subpopulations exist, where k is known, or that the number of populations is infinite. However, it is often the case that one does not know exactly what subpopulations exist, nor does one always have samples from every subpopulation. In other cases, one may not know of the existence of

a subpopulation and not have any data from such 'ghost' populations. Peter Beerli [14] constructs a scenario where three populations are exchanging migrants but only two are sampled. Various migration patterns are considered and it is shown that the analysis 'overestimates the [population] sizes considerably' in some cases, but perhaps more interestingly, the effect on the migration rates reveals 'no clear pattern'. It is even shown to be the case that for certain migration scenarios, the two population analysis (assuming two populations exist, ignoring the third completely) performs better than the 'ghost analysis' (the analysis that assumes a third population does exist and exchanges migrants at some rate, but is unsampled).

Other interesting conclusions reported include robustness of migration rate estimates to the number of unsampled populations. The interested reader is directed to the original paper and follow up work such as the work of Slatkin [15]. For the purposes of this thesis it is simply stated that inference of parameters often can be affected by such 'ghost' populations. This is rarely addressed by many authors and options for dealing with it are completely absent in all standard available software packages.

The issues of ascertainment bias, ghost populations and other rarely discussed factors (e.g. the consequences of DNA damage in ancient samples) that make the inference problem more difficult are very specialist areas of research at the moment. Methods for dealing with these factors within a formal inferential framework are still in their infancy. Although they are not further considered in this thesis, they should not be forgotten as potential confounding factors.

1.2 Software for estimating migration rates

Various authors have made available their software for analysing datasets where a migration parameter (or set of parameters) is believed to be appropriate. Many of these programs are suitable only when a specific demographic scenario can be assumed. In this subsection I review some of the more well-known programs and describe briefly what model they assume and what they return.

GENETREE

Genetree is a program developed by Griffiths and Tavaré [11], [16], which requires fully aligned sequence data, with each sequence being assigned to a given subpopulation, with the requirement that the sequences are compatible with the infinitely-many-sites model (although the documentation for this program does provide some advice on making an incompatible dataset compatible and states that the data should be 'close' to compatible!). This program assumes a model with migration rates between populations which are assumed constant throughout time. The program supports multiple subpopulations but closer examination of the documentation reveals that keeping the number of subpopulations down is strongly advised. It is suggested that the analysis should be constrained to two populations or that the number of free parameters in the migration matrix is 'two or three' with all others 'assumed from prior knowledge'. Additionally, it is suggested that locations should be amalgamated where possible.

The program allows a variable population size and the probability distribution of gene trees in subdivided populations is calculated through the use of complex recursions. Maximum likelihood estimation of various parameters

is the main focus of interest. These include migration matrix parameters, together with other statistics which are of particular interest to investigators assuming that a subdivided population gave rise to their data, such as probabilities of the location of the most recent common ancestors and the probabilities of mutations having occured in each of the various subpopulations. The program is well developed, but it is clear that the authors encourage keeping the number of populations small, and attempting to estimate only a few migration matrix parameters.

MIGRATE-N

Migrate [12] assumes that n populations exist which potentially are all exchanging migrants at some rate (which could be zero for some pairs of populations), and primarily, aims to estimate the migration rate between populations. The program gives the option of estimating all migration rates after scaling, while also allowing for various different migration models to be set up and parameters estimated (such as stepping stone models, source-sink models, as well as options for restrictions such as symmetric migration between demes). The program *does* allow a Bayesian approach in the estimation of the parameters in the more recent versions but the author admits that the likelihood approach is more 'mature' in MIGRATE simply because he 'started the coding with it'; for this reason the Bayesian options of MIGRATE are ignored here.

The method takes a Markov chain Monte Carlo approach with importance sampling with the aim being to bias the search through tree space to those trees with higher likelihoods, and then to correct for this feature. The integration not only involves considering possible genealogies, but also all possible

branch lengths for every edge on each genealogy. Interestingly, MIGRATE-N also allows geographic distances between the populations to be entered (or any other sensible measure of distance between populations) which allows the migration rates to be scaled further by this distance. MIGRATE also allows the mutation rate of each locus to vary according to a gamma distribution with shape parameter α.

The documentation is very honest and self-critical about the software. It breaks down the problems that can occur that can lead to incorrect inferences being made. Even the possibility of programming errors in the code is discussed! Beerli demonstrates cases where he is able to compare his program with the output from GENETREE and FLUCTUATE [17, not further discussed here as this program is not primarily designed for estimating migration between populations], and demonstrates for a few cases that the results are very similar.

As well as returning parameter estimates, MIGRATE can also return plots similar to the Skyline plots of Drummond et al. [18], although the most recent documentation suggests that this feature has not been thoroughly tested yet and is based on as-yet unpublished original work. Further, limited likelihood-ratio tests can be done to test hypothesis such as $H_0 : M_{12} = M_{21}$, that the migration rates are identical in both directions in a two-population model. In summary, MIGRATE is a well-developed, well-documented and evolving piece of software which allows inferences to be made about migration rates for some general migration models.

LAMARC

LAMARC [19] stands for Likelihood Analysis with Metropolis Algorithm using Random Coalescence. The program is an ambitious attempt at a single method which can simultaneously estimate effective population size, exponential growth rates in each population, migration rates from each population into every other population, together with a global recombination rate. In addition, restrictions can be put in place that constrain some of these parameters to be equal if desired. Additionally, the program accommodates finite sites mutation models such as the F84 model (e.g. [20]) of nucleotide substitution (which differentiates between transitions and transversions and allows for unequal base composition) and general time-reversible models are permissible.

It is interesting to consider the strong assumptions that are detailed in the LAMARC paper. Only those specifically concerning the population structure and migration process are discussed here. The method assumes that the subpopulation structure is constant across the whole depth of the tree, that the rate of migration is independent of the size of the populations, and that the migration rate between populations remains constant. The method is not suitable if populations have recently diverged from a common ancestor.

The major drawback of this program is the run times required before the sampler reaches convergence. The documentation for the program suggests that estimating a "recombination rate using 60 16 kb mtDNA sequences required 2 GB of memory and 34 weeks of workstation time". If one wished to use this program at its full capacity with multiple subpopulations, migration between each, together with recombination, the computational time required for a single run becomes prohibitive.

MDIV and IM

MDIV [21] is a program by Nielsen and Wakeley which allows simultaneous estimation of the divergence time between two populations assumed to have arisen from a common population in the past, and migration rates. The program required initially the assumption of infinite sites and no recombination, but now can accommodate the HKY finite-sites model [22]. The program also assumes equal population sizes in both populations.

The program provides testing for evidence of migration between the two populations or for evidence of shared recent common ancestry. The program provides both maximum likelihood estimates of the demographic parameters and likelihood surfaces. Rasmus Nielsen has since developed a more advanced version of the program, called IMa (Isolation with Migration model) [23] which additionally provides estimates of the joint posterior probability density of the model parameters together with log likelihood ratio tests of nested demographic models.

Other Programs

The previous list is by no means comprehensive, but illustrates some of the attempts made by researchers to make available software to the scientific community that allows sensible inferences to be made on collected datasets.

BATWING (Bayesian Analysis of Trees With Internal Node Generation) [24] assumes k subpopulations exist at the time of sampling but that they formed from population-splitting events (going forward in time) and with the very strong assumption that no subsequent migration takes place between these subpopulations. This assumption alone leads to the program documentation

making the admission that "in reality splits may be gradual and followed by migration."

BATWING allows the investigator to include various population growth models and some flexible prior choices to be made. However, as the program does not allow migration rates to be estimated, no more will be said about this program. It is sobering to note however that the program documentation makes the very general comment that "It must be recognised that some questions of interest about historic demography cannot be answered from present-day genetic data alone."

Other programs include FLUCTUATE, COALESCE and RECOMBINE, which are now effectively superseded by LAMARC.

1.3 Summary

The various approaches taken to estimate migration rates vary in their underlying model assumptions, the parameters that can be estimated, the ease with which the method can be performed and the time the analysis takes. Even the way in which the phylogeny/tree is treated varies across methods. It should be clear to the reader that no single approach is optimal for all questions of interest. In the chapter that follows the method of founder analysis will be presented. This is an interesting approach which starts like a phylogenetic method to produce a tree on which all further analysis is conditioned. The method then deviates from the route most analysis methods take by making the assumption that migration occurs in waves which give rise to migration events involving migrant sequences known as founder sequences and that the migration is approximately unidirectional. The approach is unique

as it is less concerned with the estimation of migration rates between populations, and more concerned with estimation of the times when the assumed migration waves occurred, together with the additional aim of attempting to identify the sequences involved in migration events. The method can be viewed as a hybrid approach which allows a unique model of migration to be considered and set within a statistical framework that allows inferences of interest to be made.

Chapter 2

Founder analysis

2.1 Founder analysis description

2.1.1 Introduction

Torroni et al. [25] analysed the mtDNA sequences of 167 American Indians by restriction analysis and observed 50 distinct haplotypes, of which 48 of these haplotypes separated nicely into four distinct clusters after a parsimony analysis. Torroni et al. label these clusters as A-D (figure 2.1) and describe the mutations that define them and the additional subclusters. They then go on to argue that various haplotypes are likely to be the "founding haplotypes" with justification being that these haplotypes are the most common within the cluster, and/or that their position within the cluster of the reconstructed phylogeny is "nodal within the cluster", with some additional reasoning given such as that the haplotypes that are probable founders are found in larger populations, while other members of the cluster are not - the argument here is that this indicates that such sequences are older and points towards such sequences being those that define a given cluster.

Figure 2.1: Reproduction of Figure 1 of Torroni et al. (1992).

Torroni et al. suggest further that the reconstruction indicates that two independent migrations took place with "two or three" haplotypes (which they have numbered haplotypes 1, 9 and 13) being flagged as *founders*, essentially because they are located deep in the tree, coinciding with nodes that define two of the clusters they are interested in, and importantly, these haplotypes/nodes essentially define the parts of the tree from which all Nadene haplotypes derive. This work, although little more than a basic phylogenetic reconstruction with some sensible *subjective* interpretation, laid the ground-

work for what is now generally referred to as "founder analysis". The work by Torroni et al. was only possible due to the fact that the reconstructed phylogeny displayed nice clustering which almost partitioned the mtDNA into distinct classes, each of which contained haplotypes from only a select group of individuals suggesting the existence of 'founder sequences' involved in various migrations into new areas.

Richards et al. [26] formalised the method of founder analysis by using objective methods to a) reconstruct a suitable phylogeny when the data set (typically mtDNA sequences) may be difficult to resolve due to reccurent mutations, with the possibility that a very high-dimensional genealogical *network* may exist, b) define formal methods to identify the 'founder sequences', and finally, c) use statistical concepts to try and estimate the age of the founder sequences and other related quantities of interest, while providing some estimates of the uncertainties related to these estimates.

It is perhaps somewhat unfortunate that exactly what a 'founder' is has not been formally defined in the scientific literature anywhere as yet. Only when working on extending the method of founder analysis (as described by Richards et al.) in this thesis did the clarification of some terms such as 'founder sequence' become necessary.

In the work of Torroni et al. and Richards et al. the terms 'founder' and 'founder sequence' were used loosely to mean a sequence involved in a migration event into a new area. However, in terms of the methodology (particularly the more formal parts of [26]), the terms 'founder' and 'founder sequence' are ambiguous. The assumption was made that one is talking about the founder sequence as being *one of the internal nodes* on the reconstructed phylogeny, which may or may not be the same sequence involved in

the migration event of interest.

To clarify, figure 2.2 displays part of a tree that contains an edge which carries a single migration event; mutations are marked on the edge (one can assume for now that they represent the exact times of each mutation). From this diagram I now will define some terms:

- "founding event": *A migration of a sequence into a new area.*
- "founder": *The DNA sequence that was brought into the new area by the founding event.*
- "founder sequence type": *The DNA sequence that corresponds to the node on the reconstructed tree that defines the new cluster.* The "founder sequence type" *may* be identical to the founder, but it may have been subject to additional mutations which make it different from the sequence of the founder. The distinction between "founder" and "founder sequence type" may seem a little unnecessary but the distinction will matter in what follows.
- "founder cluster": *All branches of the phylogeny descending from the founder sequence type.*

At the simplest level, founder analysis attempts to identify and date migrations into a new area by inferring 'founder sequence types' (using a set of selection criteria) in potential source populations. Identified founder sequence types have associated with them a cluster of descendent sequences in the settlement region that are derived from them, and the method attempts to estimate the age of such clusters. The method as proposed then

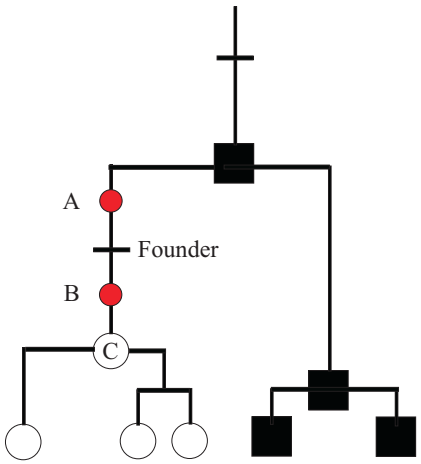

Figure 2.2: The "founder" is marked on the edge between mutations 'A' and 'B': this is the sequence involved in the "founding event". Node 'C' represents the "founder sequence type", which is not identical to the founder in this case as it carries the additional mutation 'B' that is absent on the founder. The cluster of the tree defined by node C is the "founder cluster".

tries to 'estimate the proportion of modern lineages whose ancestors arrived during each major phase of settlement' [26, page 1251]. That is, under the *assumption* that migration occurred in short duration waves, the proportion of the modern-day sample that is derived from each wave is estimated. The primary application of the method was to allow a sample of modern mtDNA sequence data to be collected and analysed to provide a *quantitative* estimate of the demic component to the spread of agriculture into Europe from the Near East (for maternal lineages). The method made use of reduced median networks [27] (together with some use of RFLP typing and some extensively discussed rules) to reconstruct the phylogenetic relationships between the sequences, and to reduce the sample data from a network to a tree.

The major aim of this thesis is to investigate some properties of this method

of analysis through the creation of appropriate data simulation procedures, and attempt to identify the performance and limitations of the method. It will be shown that founder analysis as proposed in the original paper is likely to suffer from some fundamental problems that cannot be *easily* overcome through improved statistical modelling, as the problems identified are essentially problems that arise due to the *nature and size of available datasets*. The primary issue is a systematic *underestimation* of the migration times (estimates in [26] and derived work are likely to be too *young*). This is in contrast to critiques of the method that assume that the estimates are likely to be too old [28, introducing the famous 'martian analogy']. I shall address this bias through the development of an improved version of founder analysis, which generalises the original method in a way which ameliorates such a problem and makes the method of founder analysis fully Bayesian.

The new method will be shown to perform well with simulated data for datasets of appropriate size, overcoming the problem identified at the simulation stage. An application of the full Bayesian method to re-analyse the dataset used in the Richards et al. paper then leads to an argument that current datasets used to investigate the migration of humans are likely to be too small to allow any strong inferential statements to be made, and that without the *very* strong assumption that migration events *almost always coincide with events that can be accurately identified from a reconstructed phylogeny*, the dating of migration events/periods is always likely to be imprecise for datasets of current sizes, when the number of migration events/periods is not trivially small.

2.1.2 The method of founder analysis

This section reviews the major assumptions and methods used in the original work of Richards et al. [26]. The reduction of phylogenetic networks to phylogenetic trees is not discussed here. One simply notes that the mtDNA sequence data (almost 300 base pairs of the first hypervariable segment of the control region), augmented with some additional RFLP typing (at diagnostic positions in the coding region of mtDNA) are used to reduce the space of possible networks which described the data, down to a resolved phylogenetic tree. While it is difficult to ignore the complications of the tree reconstruction process, and the variability that is lost through assuming a single reconstructed tree as even being close to the 'truth', one can at least assume the reconstruction to be reasonable and investigate the method of founder analysis *conditional on the assumed reconstructed tree*, which is what is done in this thesis. It is also acknowledged now that, with the existence of complete mtDNA genome datasets, future analysis will be able to reconstruct the phylogeny with much more precision than was possible when the founder method was first applied.

It also should be mentioned briefly here that, in all of what follows, discussions about sequence types, haplogroups, haplotypes and the associated nomenclature used to denote such objects will follow that used in the original papers (described in [29] and [30]) unless it is explicitly stated otherwise. Some unresolved branching orders in the phylogeny have been resolved since the Richards et al. paper (for example [31, Resolution of haplogroup U] and [32, Further resolution of H]), and as a consequence some nomenclature to denote haplogroups has changed. In the interests of consistency and so that comparisons can be made more fairly between the original work and what

follows in this thesis, attempts have been made to keep everything consistent with the original paper unless explicitly stated otherwise.

Given the reconstructed tree, criteria were established which were used to identify what are termed in the original paper as "candidate" founders (it should be noted here that this candidate list is a list of founder sequence types i.e. corresponding to nodes on the tree, and *not* founders). The first criterion is perhaps the most natural. It is the presence of identical sequence types in both the source and settled populations, since each such match suggests that an individual or individuals with that sequence type was involved in a migration event into the settled region (under a crucial assumption of unidirectional migration).

Three other criteria are described, which identify inferred matches within the Near Eastern and European phylogeny. They are either:

1. "unsampled types with both European and Near Eastern derivatives; or

2. sequence types sampled only in the Near East and whose immediate derivatives include at least one European; or

3. sequence types sampled only in Europe and whose immediate derivatives include at least one Near Eastern individual." [26, page 1255]

Criterion 1 can be justified by considering that the existence of sequences in both the Near East and Europe, that are each only mutational steps away from some other (unsampled) sequence, does suggest that the unsampled sequence in question could indeed be a possible candidate founder that simply is not represented in the current dataset under investigation (figure 2.3 shows

an example initial tree (left) and how the founder sequence type is determined (right)). Inferred criterion 2 can be justified by realising that the existence of a sequence type from the Near East that has European derivatives that are only mutational steps away from that Near Eastern sequence are possible candidate founders, and that one simply has not observed the same sequence in the European sample of data, instead having observed a descendent of such a sequence (figure 2.4). Criterion 3 represents sequence types found in Europe that have Near Eastern derivatives, which suggests the possibility that such a sequence could have migrated from the Near East (although such a Near Eastern sequence is absent from the sample) and existed in both locations. The existence of sequence derivatives in the Near East suggests that the sequence *was* present in the Near East (giving rise to its derivatives there), while its existence in Europe makes it an obvious candidate founder sequence (figure 2.5).

The previous paragraph described the criteria for the selection of potential candidate founders. However, recurrent mutation and back migration could easily result in candidate founders falsely being identified. The original founder analysis paper introduces 'three levels of stringency to identify founder candidates' [26, page 1255], denoted by the $f1$, $f2$ and fs criteria, with the primary aim being to reduce the effects of recurrent mutation on the candidate founder list. Additionally, the candidate list as initially constructed, subject to no stringency checks, was denoted by $f0$, forming the largest founder set, but presumably containing the largest number of *false* founders. The candidate list after the application (or not) of some stringency check will be referred to as the *founder pool* in what follows. To re-iterate, references to the list of founder sequence types *after* application of the $f0, f1, f2$

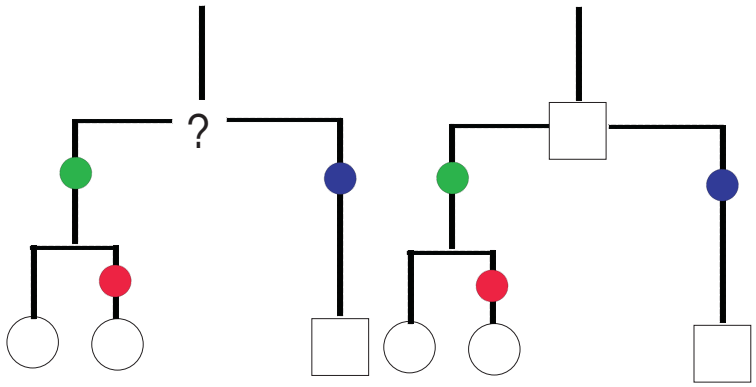

Figure 2.3: The question mark displays an unsampled sequence with derivatives in both the Near East (square) and Europe (circles), suggesting the node '?' did exist, while the assumption of no back migration gives the node a Near Eastern assignment, most parsimoniously. Filled circles represent mutations.

or fs criteria will be called the *founder pool*.

The $f1$ and $f2$ criteria were implemented to reduce the possibility of recurrent mutational events resulting in the identification of *false* candidate founders. To this end, sequence matches (either sampled directly or inferred from the previously described criteria) were required to have either one ($f1$ criterion) or two ($f2$ criterion) branches deriving from them in the Near East, while the derived types must not connect to the founder candidate via sequence types found only in Europe. Essentially, the $f1$ and $f2$ criteria allow the investigator to filter out sequence matches that have arisen merely as a consequence of parallel mutations (especially those that occur at 'fast positions' e.g. see [33, page 62]) in both settlement regions, generating identical sequences that, however, are not identical by descent. An interesting consequence of this criterion noted in the original work [26, page 1255] is that it

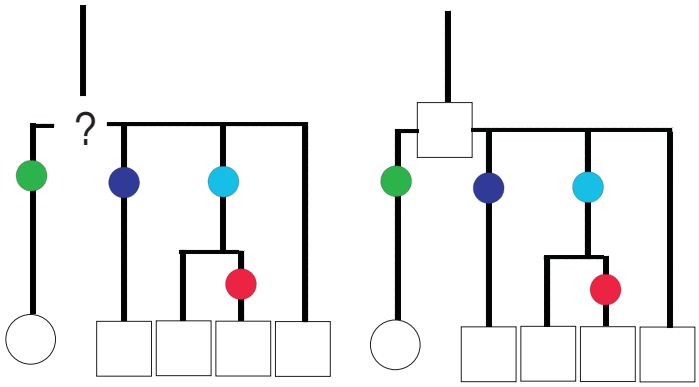

Figure 2.4: In this case, one has a sequence only observed in the Near East, with multiple sequence types that are only a few mutational steps away, one of which is European. The node marked with a question mark is assumed to be Near Eastern due to the assumption of no back migration.

brings with it some additional screening against back-migration as recently back-migrated types from Europe should lack derivatives in the Near East, which is exactly what the $f1$ and $f2$ criteria are screening for. This screening against back-migration is welcome as more recent work [34] has provided some evidence of back-migration from Europe.

The fs screening criterion was also discussed, which was an attempt to correct for the fact that the success of the $f1$ and $f2$ criteria is 'dependent on the frequency of the founder cluster candidates in Europe' [26, page 1255]. The frequency-based correction used in the fs criterion is *extremely difficult* to justify formally. It should be viewed as little more than an interesting idea which perhaps could be developed further in the future. Regardless, due to the ad hoc nature of the correction (particularly the $\log_{10}$ calculation used), no more will be said about the fs criterion in what follows.

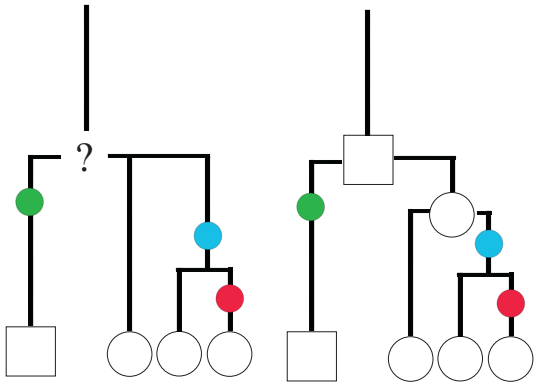

Figure 2.5: The '?' in this case has multiple European derivatives and only a single Near Eastern derivative. The assumption of no back migration automatically requires this sequence to be assigned as Near Eastern.

2.1.3 Statistical details of founder analysis

With a suitable founder list selected (the *founder pool*), the statistic, ρ, was used to provide an (unbiased) estimate of the age of each founder cluster in mutational time units, subsequently converted to years based on an assumed mutation rate of 1 transition per $20,180$ years [35]. Again, it needs to be stated here that what is being estimated is actually the age of a founder sequence type (*node* on the tree), which is assumed to correspond to the age of that cluster. The ρ estimator is itself an interesting object which will be discussed in more detail in a later chapter, but, for the moment, one simply notes that any node on a given phylogenetic tree can be dated in an unbiased manner using this estimator, and, importantly, it is inherently assumed in the original method that dating of the founder sequence type on the phylogenetic tree closely matches that of the migration time of that founder. In this section the mathematical details of the method are described in some detail.

For notational consistency with the original work, this section follows the exact notation used by Richards et al. [26]. It should be noted here however that in following chapters similar notation is used to represent slightly different statistical quantities for convenience.

It is assumed that there exists a pre-determined number, M, of migration periods, with migration period m ($1 \leq m \leq M$) occurring precisely at time t_m. As the ρ estimator of divergence time requires time to be measured in mutational time units, the notation τ_m is introduced to represent mutational time, so that $\tau_m = \mu t_m$, with μ being the mutation rate (of the full sequence typed).

Assume a uniform prior distribution for the time to the most recent common ancestor (MRCA) of a founder cluster. One also assumes the mutation process along tree edges to be Poisson. For a given founder cluster, under the assumption of a star-like phylogeny (figure 2.6), and assuming the founder cluster arose from migration period m, the number of mutation events present in the founder cluster would be distributed as a Poisson random variable with parameter $n_i\tau_m$, where n_i denotes the number of descendent sequences ('tips') arising from founder cluster i ($1 \leq i \leq I$). The indicator variables a_{im} identify whether founder i is associated with migration period m (in which case, $a_{im} = 1$). Similarly, $a_{im} = 0$ when founder i is *not* associated with migration period m. Of course, a priori, one does not know which migration period a given founder belongs to, and an uninformative, discrete uniform prior distribution is assigned, so that $P(a_{im} = 1) = 1/M$. With the notation now defined, one can derive the formula given in the original paper [26, page 1257, unnumbered formula] using Bayes' theorem.

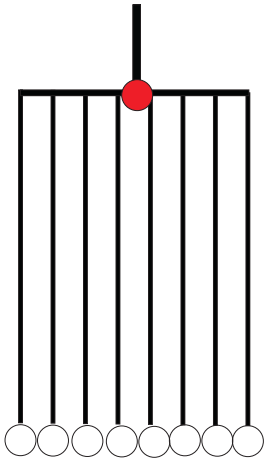

Figure 2.6: Example of a basic star tree. The founder sequence type defines a cluster which is composed of edges leading to external nodes which all are assumed to have the same length, with no further bifurcations present below the founder sequence type.

Statistical details - original derivation

Consider, for a given founder, founder i, the average number of mutations, ρ_i, on a given star tree of depth τ_m, with prior uniform allocation to migration period m. Then

$$n_i\rho_i|\tau_m, a_{im}=1 \;\sim\; \text{Po}(n_i\tau_m) \text{ and } P(a_{im}) = \frac{1}{M}\forall m, i.$$

Then, applying Bayes' Theorem, we have

$$P(a_{im}=1|\rho_i,\tau_m) \;=\; P(\rho_i|\tau_m, a_{im}=1)P(a_{im}=1)/K_1,$$

where $K_1 = \sum_{m=1}^{M} P(\rho_i|\tau_m, a_{im})P(a_{im})$. Then,

$$\begin{aligned} P(a_{im}=1|\rho_i,\tau_m) &= \frac{(n_i\tau_m)^{(n_i\rho_i)}e^{-n_i\tau_m}}{(n_i\rho_i)!}.\frac{1}{M}/K_1 \\ &= e^{-n_i\tau_m}\tau_m^{n_i\rho_i}\left\{\frac{n_i^{n_i\rho_i}}{(n_i\rho_i)!M}\right\}/K_1 \\ &= e^{-n_i\tau_m}\tau_m^{n_i\rho_i}/K_2, \end{aligned} \tag{2.1}$$

where $K_2 = \sum_{m=1}^{M} e^{-n_i\tau_m}\tau_m^{n_i\rho_i}$, and noting that the contents of the curly brackets in (2.1) are independent of m. Thus

$$\begin{aligned} P(a_{im} = 1|\rho_i, \tau_m) &= e^{-n_i\tau_m} e^{\ln[\tau_m^{n_i\rho_i}]}/K_2 \\ &= \frac{e^{-n_i[\tau_m - \rho_i \ln \tau_m]}}{\sum_{m=1}^{M} e^{-n_i[\tau_m - \rho_i \ln \tau_m]}} \end{aligned} \tag{2.2}$$

Equation (2.2) is that expressed on page 1257 of the original founder paper. This equation allows the investigator to attribute a probability to the event that the cluster deriving from candidate founder i [with some given mtDNA sequence] is associated with the m^{th} migration event/period.

A final quantity that is calculated is the proportion of the total sample that is associated with the m^{th} migration period, denoted by S_m and calculated using the following formula, with n denoting the total sample size:

$$S_m = \frac{1}{n}\sum_{i=1}^{I} a_{im} n_i. \tag{2.3}$$

S_m is a more interesting expression than it appears to be at first glance, since it relates what I have called the founder pool to the original data sample. One can regard the founder pool as the 'data' once the founder assignments have been made, and it is tempting to think of each member of the founder pool as somehow being *equal* or having common properties shared with all other founder sequence types. However, an identified founder sequence type whose founder cluster has a large number of descendants is not the same object as a founder cluster that may only be associated with a small descendant cluster of only a handful of sequences. Furthermore, it may be the case that the strength of belief that a given sequence is in fact a *genuine* founder differs between founders. This relationship is not one that is described in much detail by Richards et al. but it is extremely important to note that *all founders are not equal*, and the founder pool represents a set of sequences

that necessarily have differing numbers of descendants in the present-day sample.

The issue described in the previous paragraph resurfaces later and becomes important for understanding limitations that can arise for methods which reduce a modern-day sample of sequences to a reduced founder pool. It is always going to be the case due to the phylogenetic nature of the data that the number of potential candidate founder sequence types (nodes in the tree) is non-increasing going back in time (and decreasing to 1, the MRCA of the sample). This brings with it the consequence that the older founder sequence types are likely to have more descendants in the present-day sample than the recent founder sequence types, which are likely to represent founder clusters with only a small number of modern-day descendants. Formula 2.3 actually can be viewed as a way of circumventing this problem by re-establishing the link between founder sequence types and the *members of the clusters that each founder defines*.

The model used by Richards et al. [26] assumed five migration periods representing major prehistoric migrations from the Near East to Europe, the Neolithic at $9,000$ YBP, the Mesolithic at $11,500$ YBP, the late Upper Palaeolithic (LUP) at $14,500$ YBP, the middle Upper Palaeolithic (MUP) at $26,000$ YBP, and the early Upper Palaeolithic (EUP) at $45,000$ YBP, with a final period being assigned at $3,000$ YBP, simply to mop up recent migration events that are of only minimal interest.

Statistical details - dating the founder clusters

In the original paper, it was mentioned [26, page 1256] briefly that the dating of founder clusters was done from the "(gamma-distributed) posterior" with-

out derivation. Below I derive this result which was used to date individual founder clusters.

We assume that a star tree describes the founder cluster, with n tips in this founder cluster. The founder cluster sample size n is assumed known (observed) and fixed (it is not a random variable). Let t_k denote the migration time of founder k. One wishes to determine the distribution of t_k given n and the ρ value for that founder cluster. Under the star tree assumption,

$$n\rho|t_k \sim \text{Po}(\mu t_k n), \tag{2.4}$$

where μ is the mutation rate of the sequence under consideration.

t_k is assumed to be uniformly distributed over the entire allowable time period, that is to say,

$$t_k \sim \text{Un}(0, \infty). \tag{2.5}$$

The (improper) distribution (2.5) is important as it indicates that t_k is a random variable whose value may not necessarily coincide with any of the assumed migration periods (which were assumed to be point masses in the same paper [26] and were chosen by the investigators). Then,

$$\begin{aligned} P(t_k|n,\rho) &= \frac{P(t_k,\rho|n)}{P(\rho|n)} \\ &\propto P(\rho|t_k,n)P(t_k|n) \\ &\propto \frac{e^{-\mu t_k n}(\mu t_k n)^{n\rho}}{(n\rho)!}. \end{aligned} \tag{2.6}$$

Note, (2.6) follows since $P(t_k|n)$ is uniform in t_k, and the distribution of $n\rho$ from (2.4) can be used since, with n known, it is simply a 1-to-1 transformation of the discrete random variable. Thus,

$$t_k|n,\rho \sim \text{Ga}(n\rho + 1, n\mu). \tag{2.7}$$

2.1.4 Performance of the original method

The method is applied and gives rise to age estimates for some of the major founder clusters, and the proportion of lineages in each cluster is reported. An important feature of the results not discussed in much detail is the dating of various founder clusters not coinciding with *any* of the assumed migration periods. For example, Figure 1, page 1266 of Richards et al. [26] shows founder clusters HV, U4 and H to have 95% credible regions for the age estimates that do not overlap with any migration period; this figure is reproduced below (figure 2.7). This could simply be attributed to some of the uncertainty in the age estimates being lost as a result of the model assumptions (e.g. uncertainty exists in tree reconstruction: a perfect star tree assumption for each founder cluster is unrealistic but necessary for the model). Furthermore, the major migration periods were assigned a single date. The idea that prehistoric migration periods can be reliably assigned to a single date with any certainty is unrealistic. Such estimates themselves would have uncertainty (which would not even be the same for each period) and this is not at all represented. However, it is notable that some founder clusters are assigned intervals that do not lie even close to any of the assumed migration periods, regardless of whether it is the intervals that are not wide enough or the dates of the migration periods which are unsuitable.

Table 4 of the original work (figure 2.8) displayed the posterior estimates of S_m for all the criteria across the migration periods, where the Late Upper Palaeolithic is seen to consistently contain, on average, the largest proportion of the sample. In the following chapter this result will be revisited and it will be argued that interpretation of a single estimate of this statistic, instead of the complete posterior *distribution* of S_m, is highly problematic and hides an

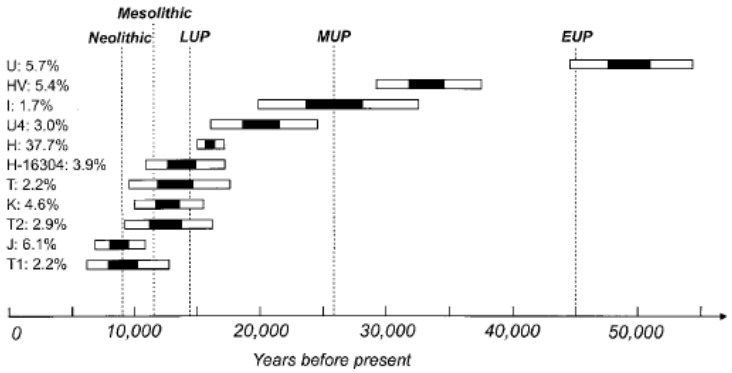

Figure 2.7: Reproduction of figure 1 of the original founder paper [26].

important property of the S_m statistic.

Founder analysis can be summarised as being a novel method of reducing a modern-day sample of sequences down to a much reduced pool of founder sequence types, assumed to represent the founder sequences involved in the major migration periods of prehistory. Under some assumptions explicitly stated in the original work, the founder pool can be used to date major founder clusters and provide a quantitative estimate of the proportion of a modern-day sample that can be attributed to migration periods that are assumed to have taken place. The method however has not been subjected to any testing with appropriate simulation studies. The original work does not attempt to hide the assumptions required for such a method of analysis to produce valid conclusions, and in the following chapter a simulation study will be developed to investigate some important properties of the method and will indicate that one of the assumptions in particular is highly problematic and needs to be dealt with.

Table 4

Percentage, of Extant European mtDNA Pool, Derived, in Each Migration Event, from Near Eastern Founder Lineages

	Mean ± Root-Mean-Square Error, of Contribution, for Criterion* (%)					
Migration Event	*f0*	*f1*	*f2*	*fs*	*fs'*	*fsr*
Basic model:						
Bronze Age/recent	16.3 ± 1.2	5.9 ± 1.2	2.6 ± .7	4.0 ± .9	2.7 ± .7	7
Neolithic	48.5 ± 3.5	21.8 ± 3.1	12.4 ± 1.6	13.3 ± 2.0	11.9 ± 1.9	23
LUP	25.1 ± 3.5	58.8 ± 3.4	63.7 ± 2.6	58.8 ± 2.8	55.4 ± 1.9	36
MUP	5.8 ± 1.5	9.3 ± 2.1	12.8 ± 2.3	14.6 ± 2.2	11.0 ± .9	25
EUP	1.8 ± 1.0	1.7 ± 1.0	6.0 ± .7	6.9 ± .5	16.5 ± .5	7
Extended model:						
Bronze Age/recent	15.2 ± 1.2	5.1 ± 1.2	2.4 ± .8	3.6 ± .9	2.5 ± .7	6
Neolithic	41.5 ± 3.0	16.5 ± 2.8	10.1 ± 2.5	10.7 ± 2.6	9.7 ± 2.5	18
Mesolithic	18.5 ± 4.1	45.2 ± 5.9	9.6 ± 4.6	10.8 ± 4.0	9.5 ± 4.0	19
LUP	15.4 ± 3.6	20.3 ± 5.8	56.9 ± 4.5	51.2 ± 4.0	48.6 ± 3.5	23
MUP	5.3 ± 1.5	8.8 ± 2.1	12.6 ± 2.3	14.4 ± 2.2	10.8 ± .9	25
EUP	1.7 ± 1.0	1.6 ± 1.0	6.0 ± .7	6.8 ± .5	16.5 ± .5	7

* Calculated as described in the Subjects and Methods section. No error estimates are shown for *fsr* because the (intuitively large) uncertainty introduced by the repartitioning process is hard to assess. Some (2.4%) of lineages ("erratics") were not assigned to founders and account for the remainder for each criterion; they are either the result of recent east Eurasian or African admixture or are rare types that could not be classified.

Figure 2.8: Reproduction of Table 4 of the original founder paper [26].

2.1.5 Criticisms of founder analysis

The method of founder analysis has been criticised by various authors. Barbujani and Dupanloup [36, Chapter 33] describe some of their problems with the method. They appreciate the work making its modelling assumptions explicit, but express concern that the mutations generating a new haplogroup may not necessarily be followed by population expansion, one of the strong assumptions of the original method. The authors suggest that the dating of founding events via the founder analysis method is unsatisfactory [36, page 423] and that the idea of inferring a largely Palaeolithic origin of the Europeans is likely to be incorrect. This view was expressed in a reply [28] to earlier work [37] when founder analysis was not yet formalised. In that reply Barbujani et al. construct an imaginary scenario where Europeans colonise Mars and say that "It would not be wise for a population geneticist of the future to infer from that a Paleolithic colonisation of Mars". They then ex-

plain in some detail how using MRCA dates as an estimate of migration time leads to overestimates.

The problem with this criticism is that Barbujani and colleagues are wrongly assuming that the age of the most recent common ancestor of two populations is what founder analysis is trying to estimate at some point. It is indeed correct that, if one were to use the date of the MRCA of sequences from two different populations as an estimate of the date at which migration between these populations occurred, then a ridiculously large and incorrect date would be obtained. However, the founder analysis method is not at any point trying to estimate the time of the MRCA of sequences from two populations, or any related quantities. It never at any point is concerned with the divergence time or MRCA's of the populations, but rather the divergence times of founder clusters.

Chapter 3

Founder analysis simulation study

3.1 Introduction

The following section introduces a structured coalescent model which incorporates population expansion in each of the demes at a common scaled growth rate together with a migration process which varies discontinuously over time. The idea here is to extend a coalescent model to create a model that will generate sequence data appropriate to what the method of founder analysis assumes, namely a model where migration periods occur at various time points, generating founder sequence types with associated founder clusters. The aim here is to build up a model which resembles that assumed in founder analysis gradually, starting with results from the structured coalescent theory.

The simulation requires short periods/bursts of migration to mimic the assumed prehistoric demography. However such a simulation procedure does

raise some theoretical issues related to the strong migration limit [38], which is nicely summarised by Wakeley [39], who summarises that "if 'Nm' is large a subdivided population behaves like a well-mixed, or panmictic, population". During the migration periods, Nm will need to be large to generate any appreciable number of migrants. However, since the migration periods will be very short I do not believe the strong-migration limit to be a problematic issue here. Nm could be reduced and the length of the migration periods increased to compensate for the reduction in all of what follows, although this would result in a simulation which did not closely resemble that assumed by the founder analysis method. The aim here is simply to *approximate* a migration process, with which one can start to look at the properties of the founder analysis method.

3.2 The model

The model described here builds on that described by Nordborg and Krone [40, chapter 12].

One envisages a subdivided population consisting of d demes, exchanging migrants forward in time from deme j to deme i in a single generation, with probability m_{ji}. Note the order of the indexes here. Recalling that the coalescent is a *backwards*-in-time process, one reserves the natural ordering of the index for the backwards in time process. Deme k, $k \in \{1, 2, \ldots, d\}$ has initial (present-day) population size of $N_k = N/d$ (this convenient assumption of equal population size in every deme is not necessary, and is relaxed later). For now, assume the population sizes and migration probabilities within each deme are not varying as a function of time.

Let b_{ij} denote the probability of migration backwards in time from deme i

to deme j, in a single generation. It is necessary to express b_{ij} as a function of the forward-in-time migration probabilities so that coalescent arguments can be invoked and the scaled backwards-in-time migration probabilities be established.

3.2.1 The relationship between the forward and backward migration probabilities

Recall,

$$\begin{aligned} m_{ji} &= P[\text{lineage in } j \text{ migrates from } j \text{ to } i] \\ &= P[\text{`parent' in } j \text{ has `child' in } i]. \end{aligned}$$

The forward migration probabilities can be represented by the $d \times d$ matrix:

$$M = \begin{bmatrix} m_{11} & m_{12} & \dots & m_{1d} \\ m_{21} & m_{22} & \dots & m_{2d} \\ \vdots & \vdots & \ddots & \vdots \\ m_{d1} & m_{d2} & \dots & m_{dd} \end{bmatrix}.$$

Note that in the above, the artificial migration from a deme back into itself is assigned a probability. This artificial construction allows the constraint $\sum_i m_{ji} = 1$ to be imposed, since one now can conceptually view the entire deme migrating each generation, with a meaningful migration fraction (from one deme to a different deme that changes the state of the system) of $\sum_{i,i\neq j} m_{ji} = 1 - m_{jj}$.

The backwards-in-time probabilities can be described in a variety of ways:

$$\begin{aligned} b_{ij} &= P[\text{lineage in } i \text{ migrates backwards in time from } i \\ &\quad \text{to } j \text{ in a single generation}] \end{aligned}$$

$=$ P[‘child’ in i has ‘parent’ in j in a single generation]

$=$ P[‘child’ in i descended from ‘parent’ in j in a single generation]

$=$ number of children (of parents from j) that were sent to i/
number of children (of parents from j) that were sent to anywhere

$=$ proportion of children in i that originated from parents in j,
in a single generation.

The above description is instructive in understanding what the backwards migration probabilities represent. However, a simple application of Bayes Theorem gives

$$b_{ij} = \frac{\frac{N_j}{N} m_{ji}}{\sum_k \frac{N_k}{N} m_{ki}} = \frac{N_j m_{ji}}{\sum_k N_k m_{ki}}. \tag{3.1}$$

Figure 3.1 is instructive for understanding this result.

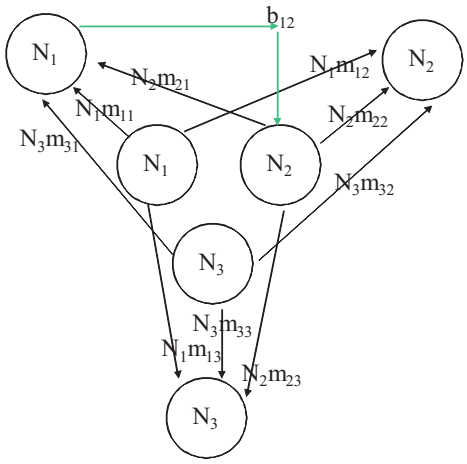

Figure 3.1: The backwards migration probabilities. Inner circles represent the current generation. Outer circles represent the next generation.

3.2.2 Simple example

In a simple case, let $d = 3$. Then $N_i = N/d = N/3$. Also, suppose an isotropic migration process, giving $m_{ji} = \begin{cases} \frac{m}{d-1}, & i \neq j \\ (1-m), & i = j \end{cases}$, resulting in

$$M = \begin{bmatrix} (1-m) & m/2 & m/2 \\ m/2 & (1-m) & m/2 \\ m/2 & m/2 & (1-m) \end{bmatrix}.$$

Then we obtain $b_{ij} = \frac{N_j m_{ji}}{\sum_k N_k m_{ki}} = \frac{m_{ji}}{\sum_k m_{ki}} = \begin{cases} \frac{m}{d-1}, & i \neq j \\ (1-m), & i = j. \end{cases}$

In general, the relationship between b_{ij} and m_{ji} can be more complicated.

3.2.3 The rates of events

Suppose now we have a sample of lineages from the present and wish to trace the ancestry of the sample going back in time. Denote the number of lineages in deme i by k_i. In a single generation the probability of a coalescent event and hence the rate per generation is approximately given by $\binom{k_i}{2}\frac{1}{N_i}$. In one generation in deme i, the probability of migration into i from j is $k_i b_{ij}$.

In time units of N generations (continuous time), the coalescence rate in deme i is

$$\binom{k_i}{2}\frac{N}{N_i} = \binom{k_i}{2}\frac{1}{c_i}, \qquad \left(\text{where } c_i = \frac{N_i}{N}\right)$$

and the migration rate into i from j is $k_i b_{ij} N$.

3.3 Time-dependent migration and population size

Suppose now N_i and m_{ji} are time-dependent. Let $N(t)$ and $m_{ji}(t)$ denote the time-dependent population size and migration rates. In one generation, the probability of coalescence is $\binom{k_i}{2}\frac{1}{N_i(t)}$ and the probability of migration into i from j is $k_i b_{ij}(t)$, where $b_{ij}(t) = \frac{N_j(t)m_{ji}(t)}{\sum_k N_k(t)m_{ki}(t)}$. In units of $N(0)$ generations, the rate of coalescence in deme i is

$$\binom{k_i}{2}\frac{N(0)}{N_i(t)} = \binom{k_i}{2}\frac{1}{c_i(t)}, \qquad \left(\text{where } c_i(t) = \frac{N_i(t)}{N(0)}\right), \tag{3.2}$$

and the rate of migration into i from j is

$$k_i b_{ij}(t) N(0). \tag{3.3}$$

3.4 Simulation

Suppose we have currently moved t time units into the past. Let t_{ci} be a simulated waiting time for a coalescent event in deme i. If $X \sim \text{Ex}(1)$, then [41, Chapter 11]

$$X = \int_t^{t+t_{ci}} \lambda_{ci}(u)\, du, \qquad \text{where } \lambda_{ci}(t) = \binom{k_i}{2}\frac{1}{c_i(t)}, \tag{3.4}$$

induces the correct distribution for t_{ci}. Similarly, let $t_{m_{ji}}$ be a simulated waiting time for migration into deme i from deme j. If $X \sim \text{Ex}(1)$, then

$$X = \int_t^{t+t_{m_{ji}}} \lambda_{m_{ji}}(u)\, du, \qquad \text{where } \lambda_{m_{ji}}(t) = k_i b_{ij}(t) N(0). \tag{3.5}$$

These results need extending however to a model which incorporates migration periods, which is covered in the next section.

3.5 Piecewise constant migration

In this section the simulation process for generating the time of migration events is described. We assume the migration probabilities between demes are constant within given time intervals (epochs). This artificial scenario is considered as it allows analytic formulae for the times to the next migration events to be created and is clearly a first approximation to a process which has migration rates varying continuously over time. From previous sections it has been stated that if $X \sim \text{Ex}(1)$, then $X = \int_t^{t+t_{m_{ji}}} \lambda_{m_{ji}}(u)\ du$, where $\lambda_{m_{ji}}(t) = k_i b_{ij}(t) N(0)$. From this it is possible to find explicit analytic formulas for $t + t_{m_{ji}}$, under certain assumptions about the underlying migration and demographic processes. The assumptions considered in this section are that the population sizes in all demes are equal, so that $N_j(t) = N(t)/d, \forall j, t$, so that

$$\begin{aligned} X &= \int_t^{t+t_{m_{ji}}} k_i b_{ij}(u) N(0)\ du \\ &= k_i N(0) \int_t^{t+t_{m_{ji}}} \frac{N_j(u) m_{ji}(u)}{\sum_k N_k(u) m_{ki}(u)}\ du \qquad (3.6) \\ &= k_i N(0) \int_t^{t+t_{m_{ji}}} \frac{(N(t)/d) m_{ji}(u)}{(N(t)/d) \sum_k m_{ki}(u)}\ du \\ &= k_i N(0) \int_t^{t+t_{m_{ji}}} \frac{m_{ji}(u)}{\sum_k m_{ki}(u)}\ du. \qquad (3.7) \end{aligned}$$

3.5.1 Modelling notation

I will now consider a model which allows the migration rates to vary in a piecewise-constant fashion (the change points separating so-called 'epochs'), with exponential population expansion within each deme. This provides migration rates which are varying between epochs together with a reasonable model of population expansion.

One envisages a two-deme ($d = 2$) population scenario (e.g. representing the Near East and Europe). We envisage epochs corresponding to important periods of demographic pre-history. The simplest model would contain four epochs of non-zero migration corresponding to the periods Early Upper Palaeolithic, Middle Upper Palaeolithic, Late Upper Palaeolithic and Neolithic. By assigning zero migration rates to each of these periods but allowing migration for a short epoch between these periods, one can model the process of migration into Europe. Figure 3.2 shows an example of the periods of migration and no migration in relation to the epochs of this model. For simplicity, the plot below shows only three epochs: this is sufficient to demonstrate the model and is the model used to test code correctness and to explore some simple, but important, aspects of the model. Of course, the true underlying process would be more complicated than this.

Now, define the epoch boundaries by T_j, $j \in (0, 1, \ldots, E)$, $T_0 = 0$ and let $\epsilon_r = (T_{r-1}, T_r)$, $r \in (1, 2, \ldots, E)$. Generating the migration event times $t + t_{m_{ji}}$ involves (3.7).

Recall that (3.7) assumes that $N_j(t) = N(t)/d, \forall j, t$, which eliminates the $N_j(t)$ component of (3.6). The value of the next migration event time $t+t_{m_{ji}}$ can fall within any of the epochs (provided it is a time greater than t). Noting that the integral in (3.7) is an integral of a piecewise constant function, we

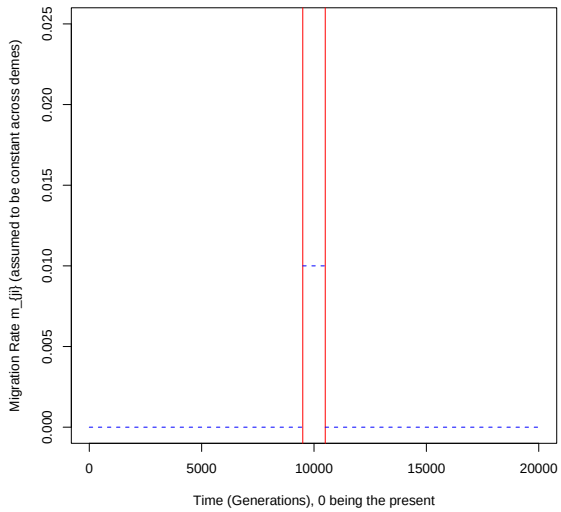

Figure 3.2: Plot demonstrating two epochs with zero migration, separated by a single epoch where migration between demes is allowed.

can calculate a value for $t+t_{m_{ji}}$ assuming we pass through $1, 2, \ldots$ epochs. In what follows it is shown that under the model of piecewise linear migration only a single $t + t_{m_{ji}}$, calculated when assuming the migration event falls in epoch ϵ_r, will be a valid migration event. All other $t + t_{m_{ji}}$ will be shown to be invalid as they do not belong to the epoch that the migration event was assumed to occur in.

3.5.2 Deriving $t + t_{m_{ji}}$ when assuming the migration event occurs in the current epoch

Let $t_{\alpha-1}$ be the current time (time of last event that changed the configuration of the system). The reader is reminded that the event $t_{\alpha-1}$ may have been associated with *either* a coalescent or migration event, whereas in what follows we are implicitly assuming t_α is the time associated with the next migration event. Suppose we are considering a backwards migration event from deme i to j, and, at time $t_{\alpha-1}$, k lineages are present in deme i. Let $t_\alpha = t_{\alpha-1} + t_{m_{ji}}$.

Assuming that t_α occurs during the same epoch as $t_{\alpha-1}$, epoch p say, then the integral in (3.7) involves integration of a single continuous function, so t_α can be calculated by directly integrating (3.7) and solving for t_α as follows:

$$X = k_i N(0) \int_{t_{\alpha-1}}^{t_\alpha} \frac{m_{ji}(u)}{\sum_k m_{ki}(u)}\, du \tag{3.8}$$

$$= k_i N(0) \frac{m_{ji}(p)}{\sum_k m_{ki}(p)} [t_\alpha - t_{\alpha-1}]. \tag{3.9}$$

Note, (3.9) follows from (3.8) because *within a given epoch*, epoch p say, the ratio $\frac{m_{ji}(u)}{\sum_k m_{ki}(u)}$ is a *constant*. Thus, the integral (3.8) is simply the area of a rectangle of dimension $\frac{m_{ji}(p)}{\sum_k m_{ki}(p)}$ by $[t_\alpha - t_{\alpha-1}]$. Solving (3.9) for t_α,

$$t_\alpha = \frac{X}{k_i N(0)} \frac{\sum_k m_{ki}(p)}{m_{ji}(p)} + t_{\alpha-1}. \tag{3.10}$$

3.5.3 Deriving $t + t_{m_{ji}}$ when assuming the migration event occurs in the subsequent epoch

Using the notation of the previous section, we further define $l_{\alpha-1}$ to be the index of the epoch boundary prior to (to the left of) $t_{\alpha-1}$. For example, index the epoch boundaries by $0, 1, \ldots$ and suppose $t_{\alpha-1} \in \epsilon_1$, i.e. the first epoch.

Then $l_{\alpha-1} = 0$. Further, define l_α to be the index of the epoch boundary prior to t_α. Note that, since we are always *assuming* we know the epoch within which the migration event occurs in these calculations, l_α is always known. Then, the integral in (3.7) when assuming the migration event occurs in the subsequent epoch is the sum of the area of two rectangles, one of dimension $\frac{m_{ji}(l_{\alpha-1}+1)}{\sum_k m_{ki}(l_{\alpha-1}+1)}$ by $[T_{l_\alpha} - t_{\alpha-1}]$ and the second of dimension $\frac{m_{ji}(l_\alpha+1)}{\sum_k m_{ki}(l_\alpha+1)}$ by $[t_\alpha - T_{l_\alpha}]$, where we have made use of the fact that, given $t_{\alpha-1} \in \epsilon_p$, then $l_{\alpha-1} = (p-1)$, and hence $m_{ji}(p) = m_{ji}(l_{\alpha-1}+1)$. Although the l notation seems unnecessary, it will be seen to be most useful in the next section when we consider generalising to passing through an arbitrary number of epochs. However, for the case under consideration, (3.7) is evaluated as follows:

$$\begin{aligned} X &= k_i N(0) \int_{t_{\alpha-1}}^{T_{l_\alpha}} \frac{m_{ji}(l_{\alpha-1}+1)}{\sum_k m_{ki}(l_{\alpha-1}+1)}\, du + k_i N(0) \int_{T_{l_\alpha}}^{t_\alpha} \frac{m_{ji}(l_\alpha+1)}{\sum_k m_{ki}(l_\alpha+1)}\, du \\ &= k_i N(0) \frac{m_{ji}(l_{\alpha-1}+1)}{\sum_k m_{ki}(l_{\alpha-1}+1)} [T_{l_\alpha} - t_{\alpha-1}] + k_i N(0) \frac{m_{ji}(l_\alpha+1)}{\sum_k m_{ki}(l_\alpha+1)} [t_\alpha - T_{l_\alpha}]. \end{aligned} \tag{3.11}$$

Solving (3.11) for t_α,

$$t_\alpha = \left\{ \frac{X}{k_i N(0)} - \frac{m_{ji}(l_{\alpha-1}+1)}{\sum_k m_{ki}(l_{\alpha-1}+1)} [T_{l_\alpha} - t_{\alpha-1}] \right\} \frac{\sum_k m_{ki}(l_\alpha+1)}{m_{ji}(l_\alpha+1)} + T_{l_\alpha}. \tag{3.12}$$

3.5.4 Deriving $t + t_{m_{ji}}$ when assuming the migration event occurs after ≥ 2 epochs

Suppose now the general case where at least a single *full* epoch is passed through before the migration time t_α. In this case the integral in (3.7) is the sum of the area of two rectangles of dimensions $\frac{m_{ji}(l_{\alpha-1}+1)}{\sum_k m_{ki}(l_{\alpha-1}+1)}$ by $[T_{l_{\alpha-1}+1} - t_{\alpha-1}]$ and $\frac{m_{ji}(l_\alpha+1)}{\sum_k m_{ki}(l_\alpha+1)}$ by $[t_\alpha - T_{l_\alpha}]$, together with the sum of the area of the rectangles corresponding to the epochs which are *fully* passed through when going from $t_{\alpha-1}$ to t_α. (3.7) is evaluated as follows:

$$\begin{aligned} X &= k_i N(0) \int_{t_{\alpha-1}}^{T_{l_{\alpha-1}+1}} \frac{m_{ji}(l_{\alpha-1}+1)}{\sum_k m_{ki}(l_{\alpha-1}+1)}\, du \\ &+ k_i N(0) \int_{T_{l_\alpha}}^{t_\alpha} \frac{m_{ji}(l_\alpha+1)}{\sum_k m_{ki}(l_\alpha+1)}\, du \\ &+ \sum_{s=l_{\alpha-1}+1}^{l_\alpha-1} k_i N(0) \int_{T_s}^{T_{s+1}} \frac{m_{ji}(s+1)}{\sum_k m_{ki}(s+1)}\, du \\ &= k_i N(0) \frac{m_{ji}(l_{\alpha-1}+1)}{\sum_k m_{ki}(l_{\alpha-1}+1)} [T_{l_{\alpha-1}+1} - t_{\alpha-1}] \\ &+ k_i N(0) \frac{m_{ji}(l_\alpha+1)}{\sum_k m_{ki}(l_\alpha+1)} [t_\alpha - T_{l_\alpha}] \\ &+ k_i N(0) \sum_{s=l_{\alpha-1}+1}^{l_\alpha-1} \frac{m_{ji}(s+1)}{\sum_k m_{ki}(s+1)} [T_{s+1} - T_s]. \end{aligned} \tag{3.13}$$

Solving (3.13) for t_α,

$$\begin{aligned} t_\alpha &= T_{l_\alpha} + \frac{\sum_k m_{ki}(l_\alpha+1)}{m_{ji}(l_\alpha+1)} \Bigg\{ \frac{X}{k_i N(0)} - \frac{m_{ji}(l_{\alpha-1}+1)}{\sum_k m_{ki}(l_{\alpha-1}+1)} [T_{l_{\alpha-1}+1} - t_{\alpha-1}] \\ &- \sum_{s=l_{\alpha-1}+1}^{l_\alpha-1} \frac{m_{ji}(s+1)}{\sum_k m_{ki}(s+1)} [T_{s+1} - T_s] \Bigg\}. \end{aligned} \tag{3.14}$$

3.6 Simulation of the migration process and the consequences

The previous section details the assumed migration process, and various formulae derived for t_α, the time of the next (migration) event. In this section the process is simulated and it is shown that only one migration event occurring during a single epoch can be a valid migration event for a given random draw from an exponential distribution with rate 1.

Consider a model with four epochs, assuming the migration rates within each epoch are $m_{21} = (0.00001, 0.00005, 0.00001, 0.00005)$. For the purposes of this simulation we will denote by T.vec, the vector of break points (in continuous time) which separate the epochs, T.vec $= (T_0, T_1, T_2, T_3, T_4)$, $T_0 = 0$, $T_4 \to \infty$.

Let T.vec $= (0, 0.1, 0.2, 0.3, \infty)$. Assume a two-deme model with migration in only a single direction, from deme $2 \to 1$ backwards in time. Assume also a constant population size of 5000 in each deme (so scaling in units of $10,000$ generations), and assume further that $k = 10$ lineages exist in each deme at time 0. Note that all of the previous assumptions have been made as simple as possible, although this is not required. The purpose of this section is to demonstrate that the mathematics of the migration process yields simulation results that are reasonable. We now imagine drawing a realisation of an exponential random variable with rate 1, and, with this, calculate the time to the next migration time.

Using R [42], $10,000$ simulations of t_α were undertaken, each calculated from a draw from an exponential distribution with rate 1. The values of t_α were

calculated, assuming the events occur during each specific epoch. Table 3.1 shows the first 10 rows of output from this simulation. It can be seen that only a single t_α falls within the assumed epoch under which it was calculated. In all $10,000$ simulations, only a single migration time was valid for each draw.

Table 3.1: t_α assuming event occurs in each epoch.

	Epoch 1	Epoch 2	Epoch 3	Epoch 4
1	0.76	0.23	0.36	0.31
2	1.87	0.45	1.47	0.53
3	1.73	0.43	1.33	0.51
4	0.17	0.11	−0.23	0.19
5	1.99	0.48	1.59	0.56
6	0.21	0.12	−0.19	0.20
7	0.05	0.09	−0.35	0.17
8	0.84	0.25	0.44	0.33
9	1.40	0.36	1.00	0.44
10	0.01	0.08	−0.39	0.16

3.6.1 Proof that only a single time corresponding to a single epoch is a valid migration time

This section requires some mathematical analysis and basic measure theory. Minimal reference to the concept of measure has been made in this thesis, and this approach will be continued in later sections. For the required proof here, however, one cannot avoid measure and the concept of *almost surely*.

Recall the Intermediate Value Theorem, e.g. see [43]:

Consider a function $f(x)$ continuous at every point of an interval. Let a and b be any two points of the interval and let η be any number between $f(a)$ and $f(b)$. Then there exists a value ξ between a and b for which $f(\xi) = \eta$.

Let Λ denote the integrated intensity function, which is continuous. Recall that, in the standard coalescent, an infinite sample of sequences finds a common ancestor in finite time e.g. see [6]. Consequently, a coalescent process with (non-zero) migration between two populations will also find a common ancestor in finite time. Suppose for the moment the integrated intensity function is also strictly increasing, which will be the case provided the migration rate is always non-zero. Let $\Lambda(t_\alpha)$ be the value of the function such that the integral from $t_{\alpha-1}$ to t_α equals X, the value of the draw from the Ex(1) distribution, i.e $\int_{t_{\alpha-1}}^{t_\alpha} \lambda(u)du = \Lambda(t_\alpha) - \Lambda(t_{\alpha-1}) = X$. By the Intermediate Value Theorem, a t_α exists that gives the required $\Lambda(t_\alpha)$. Uniqueness of t_α follows from the assumed strictly increasing assumption of the $\Lambda(.)$ function.

In the more general case when $\Lambda(.)$ is not assumed to be strictly increasing, but only monotonically increasing, we require the concept of a result being

true *almost surely*. Recall the elementary probability object, a probability space $(\Omega, \mathcal{F}, P)$, where Ω is the sample space, $\mathcal{F}$ is a σ-field of subsets of Ω, and P is a probability measure, a mapping from $\mathcal{F}$ to the real numbers such that $0 \leq P(A) \leq 1$ for all $A \in \mathcal{F}$, with $P(\emptyset) = 0$, $P(\Omega) = 1$, and P countably additive. It can be shown (e.g. see [44]) that for a continuous random variable, Y, $P(Y = y) = 0$.

Consider now the possibility that the migration rate could be zero within some epoch. By the Intermediate Value Theorem, a value of t_α which satisfies (3.5) still exists. However, a zero migration rate within an interval now means that the t_α which satisfies (3.5) is no longer guaranteed to be unique. In fact, as soon as a migration epoch is entered with a zero migration rate, the function $\Lambda(.)$ is constant within that epoch. Suppose this epoch corresponds to times (T_p, T_{p+1}), with the value of $\Lambda(.)$ within this interval being q. Then every $t \in (T_p, T_{p+1})$ satisfies (3.5), when $\Lambda(t_\alpha)$ takes on the *single specific value* q. However, $P(\Lambda(t_\alpha) = q) = 0$. Thus, the event that $\Lambda(t_\alpha) = q$ has zero measure. Thus, a value of t_α which satisfies (3.5) exists by the intermediate value theorem and is unique *almost surely*.

3.7 Coalescent event rates

This section briefly describes the process of simulating coalescent event times under a model with constant population size in each deme and that consists of possibly different initial (present day) population sizes. This section also helps relate the models (notation) described in chapter 4 of [6] and pages 231-255 of [45]. Structured coalescent processes such as those described in ([6],[45]) are often scaled in the size of the total global population, here denoted by N (in many diploid applications it is common and convenient to introduce a factor of 2 and use quantities such as $2N$, but not here). The consequence of this is that the coalescent rate within a deme (subpopulation) of size N_i is larger than the coalescent rate that would apply to a *single* population of size N_i. Letting N denote the size of the total global population in a d-deme model and supposing the size of the population at time $t = 0$ in each deme is $\frac{N}{d}$, and that deme i has size $N(t) = \frac{N}{d}, \forall t$, then the rate of coalescence in deme i becomes:

$$\binom{k_i}{2}\frac{N}{N_i(t)} = \binom{k_i}{2}\frac{d\frac{N}{d}}{\frac{N}{d}} = d\binom{k_i}{2}. \tag{3.15}$$

The factor of d arises due to the time scaling in the size of the global population size. This is similar to the notation used in [6]. [45] generalises this slightly by allowing the population sizes to vary between demes according to some c_i values. These can be viewed as the fractions of the total population that is present in deme i, and can be related to the model of [6] by noting that, in [45], N is still the global population size, with $N = \sum_i c_i N$, with $\sum_i c_i = 1$. Now, assuming $N_i(t) = c_i N, \forall t$, then within a single deme the rate is just:

$$\binom{k_i}{2}\frac{N}{N_i(t)} = \binom{k_i}{2}\frac{N}{c_i N} = \frac{1}{c_i}\binom{k_i}{2}. \tag{3.16}$$

To relate this back to the model in [6], note that, for a d-deme model, when each deme is of equal size $c_i = \frac{1}{d} \forall i$, the coalescent rate (3.16) becomes $d\binom{k_i}{2}$, as in (3.15).

3.7.1 Coalescent event times under exponential expansion

(3.15) describes the rate of coalescence in deme i for the model under consideration (specifically with the assumption that each deme has the same initial population size). Suppose further that the populations are decreasing in size going back in time at the rate β (measured on the coalescent time-scale), corresponding to exponential growth forward in time. Then (3.15) becomes

$$\binom{k_i}{2} \frac{N}{N_i(0)e^{-\beta t}} = \binom{k_i}{2} \frac{d\frac{N}{d}}{\frac{N}{d}e^{-\beta t}} = d\binom{k_i}{2} e^{\beta t}. \tag{3.17}$$

Thus, the time to the next coalescent event t_α in deme i is obtained from solving a slightly modified version of (3.4) with the appropriate rate function from (3.17) above:

$$\begin{aligned} X &= \int_{t_{\alpha-1}}^{t_\alpha} d\binom{k_i}{2} e^{\beta u}\, du & (3.18) \\ &= d\binom{k_i}{2} \frac{1}{\beta} \left[e^{\beta t_\alpha} - e^{\beta t_{\alpha-1}} \right]. & (3.19) \end{aligned}$$

Solving for t_α yields

$$t_\alpha = \frac{1}{\beta} \log \left\{ X\beta \frac{1}{d} \binom{k_i}{2}^{-1} + e^{\beta t_{\alpha-1}} \right\}. \tag{3.20}$$

3.8 Simulation

Equations (3.10), (3.12), (3.14) and (3.15) are all that are required to simulate the structured coalescent process without exponential population growth (forward in time), together with a migration process which is constant within each given epoch. The purpose of the first simulations is to investigate how close the founder sequence types are to the migration epoch boundary; this is something which can only be determined by simulation techniques and is something that has not been discussed in the literature, with no effort directed so far to estimate any discrepancy between these two dates.

3.8.1 First simulation output (no exponential growth)

The first simulation was designed to make use of the structured coalescent process without exponential growth and was undertaken to investigate how close the founders occurred in relation to the start of the designated migration boundary. This is important as, given sequence data, the inferred founder sequence types can only be taken to be any of sequences present or common ancestors of a subset of the sequences. However, the founder that actually took part in the migration need not match the sequence of the node we infer from any given tree as being the founder sequence type. The migration event could have occurred much further back in time and we can only *infer* a derived sequence as being the founder (recall figure 2.2).

3.8.2 Parameter values

The starting number of ancestors in each deme was varied from 250 to 1000 in increments of 250. We prohibit migration for 7500 years and then a hypothetical migration period is envisaged 7500 years ago from the present, which

lasts 1000 years. During this period one-way migration is permitted. After this period no migration occurred, until a period of fast migration at 13500 years, corresponding to the settlement of the descendent population. The total population size was set at 5000, with a generation assumed to last 25 years. 25 trees were simulated for the cases where the number in each deme was 250 or 500, and 10 trees were simulated for the cases where the sample size in each deme was 750 or 1000. The forward migration rate, denoted by m, was varied from 0.0001 to 0.01 per generation.

The time 7500 years ago corresponds to 300 generations, after which a migration period occurs which lasts for 40 generations. The main point of interest here is the difference between the time of the migration period, and the time of the founder sequence types. The complete output from these simulations is not shown (but was retained), as the conclusions were similar across all runs. The most illustrative sets of summaries are presented below (those simulations with the largest migration rates and, subsequently, largest numbers of founder sequence types), in figures 3.3 and 3.4.

From the plots, it is clear that the difference between the time of the migration boundary and the time of the founder sequence types can be very different. This is problematic and suggests that the method of founder analysis may be estimating founders as being too young i.e. occurring too close to the present. However, the method of founder analysis assumes a star tree topology for founder clusters, which is more probable under situations of population expansion. The previous model does not include expansion and this could be the reason that the discrepancies between the migration time and founder sequence type times are so large. Furthermore, for the simulation to be relevant to the results that one may obtain from analysis of a

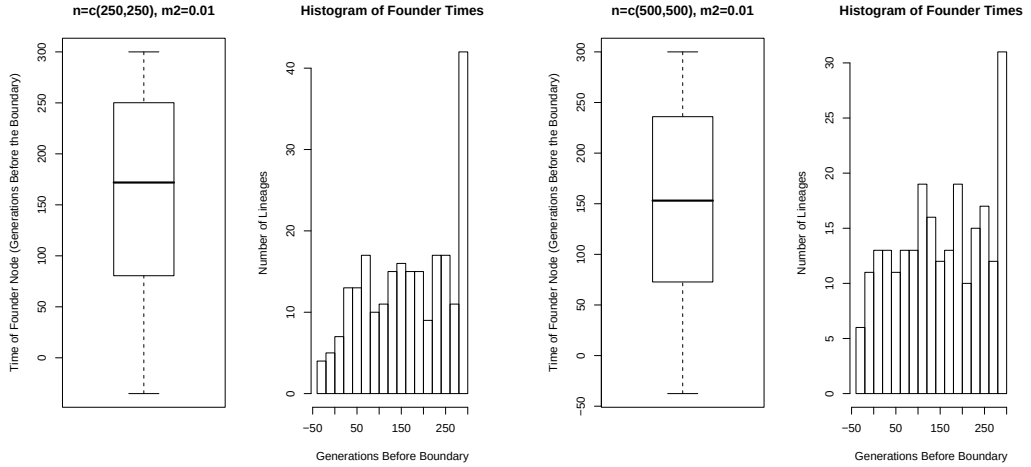

Figure 3.3: Summaries of the number of generations before (in reversed time) the migration boundary that the founder sequence types *actually* were located across all simulations. Note that negative values indicate that the founder sequence type was located within the designated migration period. The migration period starts 300 generations before present. Summaries for sample sizes 250 (left) and 500 (right) are presented in this figure, both for $m = 0.01$.

real dataset, it is necessary that the tree depths from the simulations are in some way 'similar' to the tree depths that would be expected for real human mtDNA. This forms the focus of the next section.

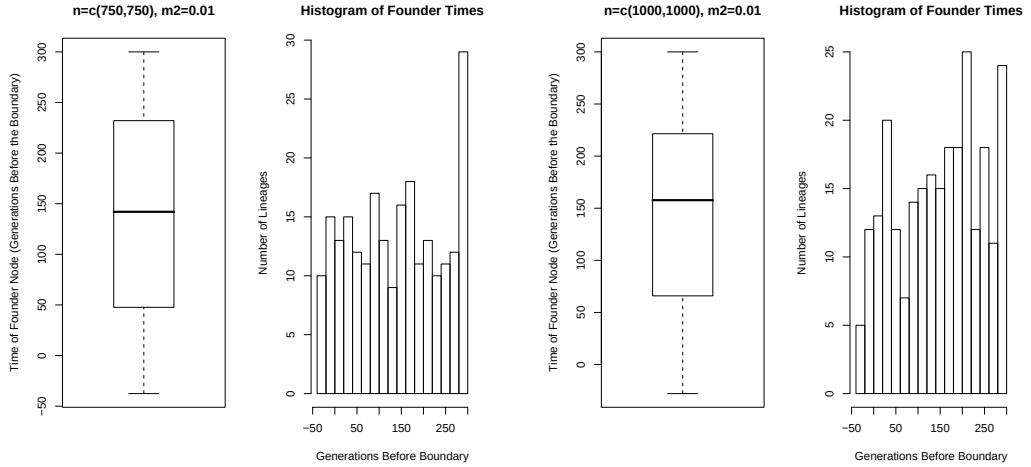

Figure 3.4: Summaries of the number of generations before (in reversed time) the migration boundary that the founder sequence types *actually* were located across all simulations. Note that negative values indicate that the founder sequence type was located within the designated migration period The migration period starts 300 generations before present. Summaries for both sample sizes 750 (left) and 1000 (right) are presented in this figure, both for $m = 0.01$.

3.9 Considering tree depth

Before embarking on some more detailed investigation of the most basic properties of founder analysis, it seemed sensible to look at the range of tree lengths that were obtained for various values of N and β. It is well known that, as the population expansion rate increases, the expected total tree depth decreases for a population of fixed initial size N, although no analytical formula is available for its expected value. This could be problematic as a tree with star-like founder clusters is assumed in founder analysis, and to obtain such a tree, if this is at all possible, the population expansion rate may need to be high. This potentially could result in trees being obtained of unreasonable

total depth. To compensate for the increasing values of population expansion, N can be increased. The following section looks to vary N, and, for each N value, vary the population expansion rate so that trees of sensible depth are obtained. For the moment, a sensible range is defined to be trees of depth ranging from $60,000 - 90,000$ years [46]. Although this is a fairly vague choice, it will allow some investigation of the relationship between N and β.

Table 3.2 shows the summary of the investigation. Although the migration model parameters are not of particular relevance within this setting, a model with epoch boundaries at $(0, 19500, 20500, 40000, \infty)$ years back in time from the present day was selected with a forward migration rate of $m = 0.001$ between $19,500$ and $20,500$ years (a period of 40 generations starting from 780 generations from present), and a 'fast' forward migration rate of $m = 0.05$ in the final epoch to bring the lineages into a single deme. Epochs 1 and 3 were assigned zero migration rates. As before a two-deme model was used with one-way migration from population 1 to 2 forward in time, with 250 lineages in each deme at the start. Some plots of trees at various parameter combinations are also shown (figures 3.5 and 3.6). Note however that the y axis scale is not constant across plots. Four trees which have lengths close to $60,000 - 90,000$ years were randomly selected for presentation and a set of illustrative figures has been produced, figures 3.5 and 3.6, which demonstrate the visible change in the trees obtained when the expansion rate is increased but the tree depth is held approximately constant. The complete set of figures is not presented here (although were retained).

Table 3.2: Investigating tree depth for various N and β combinations. Note that, for each of the combinations selected, 10 trees were initially generated, and if any of the tree lengths fell within the correct range then a further 10 were generated, and the β value classed as 'accepted' if at least 3 of the 20 trees were accepted; otherwise the parameter combination was rejected.

N	β Rejected	β Accepted	Notes
1000	0.000001,0.1	NA	$N = 1000$ too small
2000	5	0.000001,0.1	$\beta = 5$ too large
5000	10	0.1,5	3 trees accepted for $\beta = 5$
10000	0.1,20	5,10	
100000	50,100,300	150,200,250	
1000000	≤ 1750	2000,3000	
100000000	≤ 20000		All trees too deep

With parameter combinations which give rise to trees of appropriate depth now known, it is now possible to investigate the discrepancy between the date of the founder sequence type and the migration epoch date.

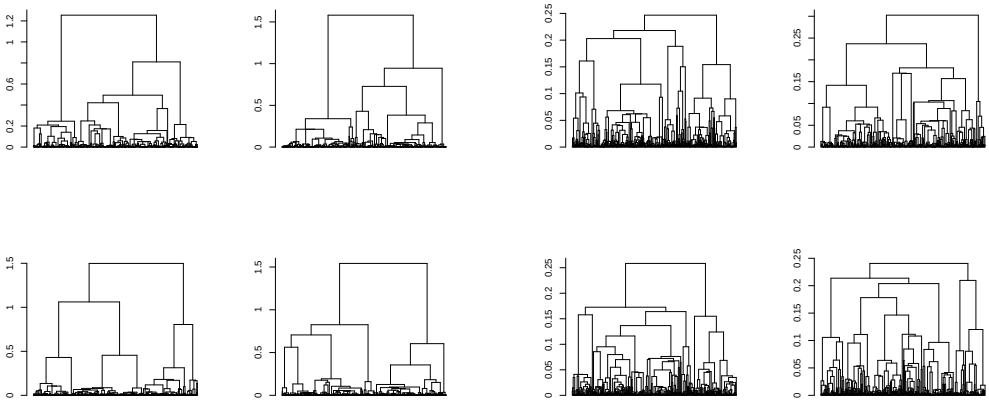

Figure 3.5: Some examples of the trees obtained for two parameter sets (left: $N = 2,000$, $\beta = 0.1$, right: $N = 10,000$, $\beta = 10$). Note the increasing length of the external branches as the expansion rate increases.

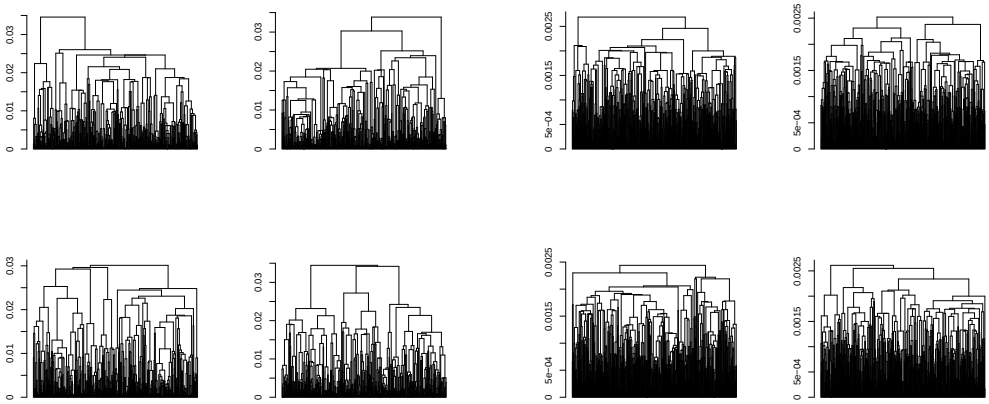

Figure 3.6: Some examples of the trees obtained for two parameter sets (left: $N = 100,000$, $\beta = 150$, right: $N = 1,000,000$, $\beta = 3000$). Note how these trees start to display more obvious star-like structure.

3.10 Second simulation

The founder simulation was run again using parameter values that gave trees which both looked star-like and had a sensible time to the most recent common ancestor. 250 lineages initially were present in each deme with $\beta \in \{150, 200, 250\}$ for $N = 100,000$ and $\beta \in \{2000, 2500\}$ for $N = 1,000,000$. The migration rate, m, in the migration epoch (Epoch two) was set to either 0.1 or 0.001 so the difference in the number of founders could be inspected, and the migration period started at $19,500$ years from present, and concluded at $20,500$ years from present. The output for four parameter combinations are presented below (Figures 3.7 and 3.8). The results from the other simulations are omitted (but retained) since they offer no additional insight.

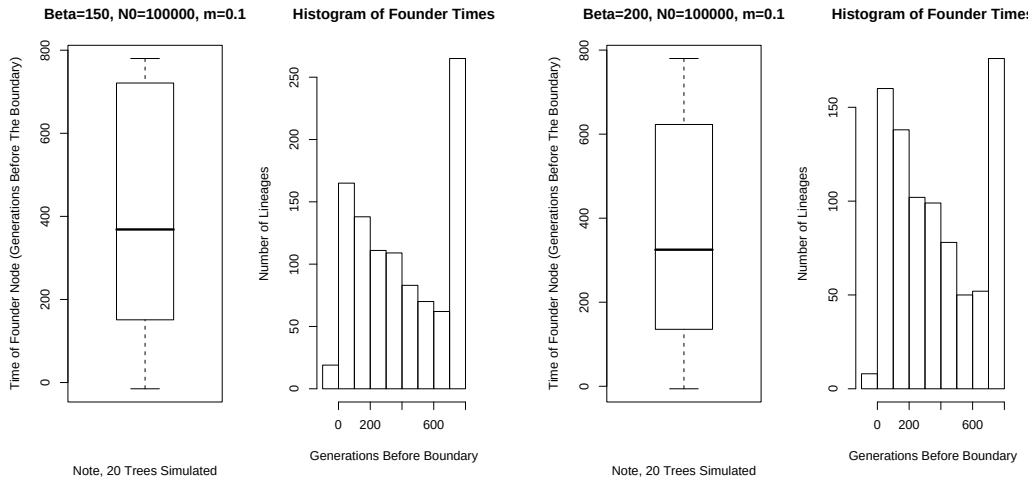

Figure 3.7: Summary plots of the discrepancy between the founder sequence type dates and the migration period.

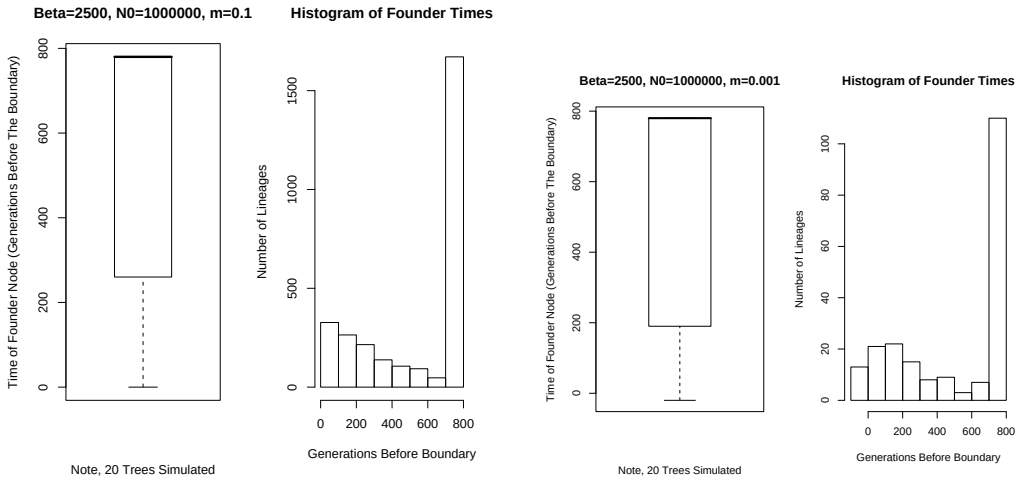

Figure 3.8: Summary plots of the discrepancy between the founder sequence type dates and the migration period for particularly large β cases.

Figures 3.7 and 3.8 display the same troubling features seen previously, namely that the founder sequence types are often located on the tree many generations away from the migration period. Some of these founder sequence types actually even occur at the tips of the tree, indicating that only allowing founders to occur on nodes of a tree may introduce significant bias in the age estimates of founders, while suggesting that the sequence actually involved in the migration/founding event may be markedly different from the sequence of the founder sequence type.

Of course, one can argue that this simulation is still far from appropriate. Firstly, the migration rates have simply been selected for the migration epoch (which itself has its duration simply defined), and this may be far from the true rate, or the migration periods may be too short (or long); all that this affects however is the *number* of migration events/founders; it does not change the fact that the founder sequence types are located far from the migration epoch. Secondly, one could argue that any simulation should also try to in-

corporate the assumed idea that founding events are followed by expansion. However, tying migration with demography at the level of founders is very troublesome from a technical point of view. Coalescent theory applies to the entire history of a sample, so working in ideas to allow subsets of the sample to be subject to different demographic processes is not possible. Perhaps a more important reason that demography should not be tied tightly to migration events at the founder level is the simple fact that the assumption of migration followed by instantaneous expansion is an extremely strong (and unrealistic) one, so trying to develop a simulation procedure for such a very artificial scenario does not seem ideal.

It is necessary however to point out that the above problem of the large numbers of generations between founder sequence types and the migration periods would arise under certain simulation conditions in a perfectly natural manner depending on the combinations of parameter values selected and the locations of the migration periods. A good example is to consider what would happen in the above simulations if the migration period were brought closer to the present. In this case the lower parts of the tree would remain unchanged, so one would still see the long external edges. However the migration period being closer to the present would result in more founder sequence types occurring on external edges of the tree. Similarly, a very old migration period would reduce the number of founder sequence types on the external edges as fewer edges on the tree exist at the time of the migration period which terminate at the bottom of the tree. It is simply noted here that this fact does *not* mean that this problem can be ignored, and it is shown in future chapters that the problem of founder sequence types existing on the external edges of a tree is very real and presents itself in *real* datasets.

3.11 Third simulation

Developing the model further, it is possible to give the different populations different initial starting sizes and different expansion rates. Define $N_i(0) = c_i N(0)$, with $\sum_i c_i = 1$ so that c_i is the fraction of the total initial population in deme i. Assuming populations experience expansion at different rates, then

$$N_i(t) = c_i N(0) e^{-\beta_i t}, \tag{3.21}$$

where $\beta_i = b_i N(0)$ and

$$N(t) = \sum_{i=1}^{d} c_i N(0) e^{-\beta_i t}. \tag{3.22}$$

In the special case when $d = 2$,

$$N(t) = [c_1 e^{-\beta_1 t} + c_2 e^{-\beta_2 t}] N(0). \tag{3.23}$$

Now, generalising (3.1), we have

$$b_{ij}(t) = \frac{N_j(t) m_{ji}(t)}{\sum_k N_k(t) m_{ki}(t)}. \tag{3.24}$$

Inserting (3.21) into (3.24) gives

$$b_{ij}(t) = \frac{c_j e^{-\beta_j t} m_{ji}(t)}{\sum_k c_k e^{-\beta_k t} m_{ki}(t)}, \tag{3.25}$$

and, for the $d = 2$ case,

$$b_{ij}(t) = \frac{c_j e^{-\beta_j t} m_{ji}(t)}{c_1 e^{-\beta_1 t} m_{1i}(t) + c_2 e^{-\beta_2 t} m_{2i}(t)}. \tag{3.26}$$

Consider a backwards migration from deme r to deme $s(s \neq r)$, for which (3.26) becomes

$$\begin{aligned} b_{rs}(t) &= \frac{c_s e^{-\beta_s t} m_{sr}(t)}{c_r e^{-\beta_r t} m_{rr}(t) + c_s e^{-\beta_s t} m_{sr}(t)} \\ &= \frac{1}{1 + \frac{m_{rr}(t)}{m_{sr}(t)} \frac{c_r}{c_s} e^{-(\beta_r - \beta_s)t}}. \end{aligned} \tag{3.27}$$

(3.27) features in (3.5) and needs to be integrated appropriately. To make the integration process symbolically easier, some notation is introduced which nicely emphasises the fact that parts of this formula are constant within an epoch. Let

$$\gamma_\epsilon = \frac{m_{rr}(t)}{m_{sr}(t)}, \tag{3.28}$$

$$\beta = \beta_r - \beta_s, \tag{3.29}$$

$$\delta = \frac{c_r}{c_s}. \tag{3.30}$$

It is worth noting that β and δ are constants, and that γ_ϵ is constant within each epoch (the ϵ subscript is in place to acknowledge the fact that γ may be different in each epoch). Now, (3.5) becomes

$$X = k_r N(0) \int_{t_{\alpha-1}}^{t_\alpha} \frac{1}{1 + \frac{m_{rr}(u)}{m_{sr}(u)} \frac{c_r}{c_s} e^{-(\beta_r - \beta_s)u}} \, du \tag{3.31}$$

$$= k_r N(0) \int_{t_{\alpha-1}}^{t_\alpha} \frac{1}{1 + \gamma_\epsilon \delta e^{-\beta u}} \, du. \tag{3.32}$$

The solution to the integral in (3.32) when assuming one stays within a single epoch (so that γ_ϵ does not change) is readily seen to be (Appendix B)

$$\left[\frac{1}{\beta} \log \left(\gamma_{\epsilon_w} \delta + e^{\beta u} \right) \right]_{t_{\alpha-1}}^{t_\alpha}. \tag{3.33}$$

3.11.1 Deriving t_α when $t_{\alpha-1}, t_\alpha$ belong to same epoch

Assume $t_{\alpha-1}, t_\alpha \in \epsilon_w$, i.e. the previous and next events both lie in the same epoch, w. Then

$$X = k_r N(0) \left[\frac{1}{\beta} \log \left(\gamma_\epsilon \delta + e^{\beta u} \right) \right]_{t_{\alpha-1}}^{t_\alpha}. \tag{3.34}$$

Solving for t_α gives

$$t_\alpha = \frac{1}{\beta} \log \left\{ \left[\gamma_\epsilon \delta + e^{\beta t_{\alpha-1}} \right] \exp \left[\frac{X\beta}{k_r N(0)} \right] - \gamma_\epsilon \delta \right\}. \tag{3.35}$$

3.11.2 Deriving t_α when $t_{\alpha-1} \in \epsilon_z, t_\alpha \in \epsilon_{z+h}, h \geq 1$

Suppose $t_{\alpha-1} \in \epsilon_z, t_\alpha \in \epsilon_{z+h}, h \geq 1$. Then

$$\begin{aligned} X = k_r N(0) \Bigg\{ & \int_{t_{\alpha-1}}^{T_{l_{\alpha-1}+1}} \frac{du}{1 + \gamma_{\epsilon_{l_{\alpha-1}+1}} \delta e^{-\beta u}} \\ & + \int_{T_{l_\alpha}}^{t_\alpha} \frac{du}{1 + \gamma_{\epsilon_{l_\alpha+1}} \delta e^{-\beta u}} \\ & + \sum_{s=l_{\alpha-1}+1}^{l_\alpha - 1} \int_{T_s}^{T_{s+1}} \frac{du}{1 + \gamma_{\epsilon_{s+1}} \delta e^{-\beta u}} \Bigg\}. \end{aligned}$$

Integrating and solving for t_α, after some tidying (Appendix B), gives

$$t_\alpha = \frac{1}{\beta} \log \left\{ \exp \left[A - E_1 \right] \frac{CD}{B} - F \right\} \tag{3.36}$$

where $A - F$ are defined as follows:

$$\begin{aligned} A &= \frac{X\beta}{k_r N(0)}, \\ B &= \gamma_{\epsilon_{l_{\alpha-1}+1}} \delta + e^{\beta T_{l_{\alpha-1}+1}}, \\ C &= \gamma_{\epsilon_{l_{\alpha-1}+1}} \delta + e^{\beta t_{\alpha-1}}, \end{aligned}$$

$$
\begin{aligned}
D &= \gamma_{\epsilon_{l_\alpha+1}}\delta + e^{\beta T_{l_\alpha}}, \\
E_1 &= \sum_{s=l_{\alpha-1}+1}^{l_\alpha-1} \left[\log\left(\gamma_{\epsilon_{s+1}}\delta + e^{\beta T_{s+1}}\right) - \log\left(\gamma_{\epsilon_{s+1}}\delta + e^{\beta T_s}\right)\right], \\
F &= \gamma_{\epsilon_{l_\alpha+1}}\delta.
\end{aligned}
$$

An equivalent way of computing (3.36) can also be established by further rearrangements and simplifications shown in Appendix B.

3.11.3 Simulating the model

The model described in the previous section was used to investigate the likely effect of increasing β on the founder sequence type times. Although it has already been noted that care needs to be taken to ensure that the resulting trees are of sensible length, the first simulation undertaken simply set N to $20,000$, and m during the migration period to 0.01 (a preliminary run indicated that this would result in a reasonable number of migrations occurring). 250 lineages were present in each deme at the start and 20 trees were simulated for varying values of β. The β parameter in deme 2 in this simulation took on the values $5, 10, 20, 50, 100, 200$. It was found in previous simulations, for $N = 10,000$ and $\beta = 5 - 20$, that trees of reasonable depth (between $60,000$ and $90,000$ years) were produced. Although these previous simulations were done using a model where both demes were experiencing population expansion at equal rates, the β range covered here is likely to span at least some of the range of values which result in trees being obtained which are of sensible depth. Regardless, the main purpose of this simulation was to investigate the founder sequence type times and how they varied as β increased in a single deme. Figure 3.9 shows the difference in the dates of the founder sequence type and the migration boundary across all trees.

From figure 3.9 it can be seen that an increase in β does indeed seem to reduce

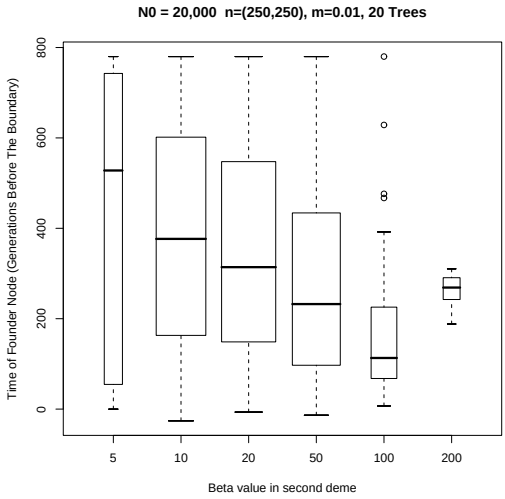

Figure 3.9: Boxplots of the number of generations between the founder sequence type and start of the migration period.

the difference between the founder sequence type date and the start of the migration period, with the exception of $\beta = 200$. However, $\beta = 200$ produced the fewest founders. This is due to the fact that the large β value necessarily forces the tree to reach a most recent common ancestor very quickly. In fact, by the time the migration boundary is reached (780 generations ago), the population size in deme 2 is approximately $c_2 Ne^{-200*(0.039)} \approx 4$. Under such conditions the coalescent process is a poor approximation to the underlying process as the population size is no longer large. In contrast, for the $\beta = 100$ case, the same calculation yields a population size of just over 200. For large β values the descendent population is very likely to have coalesced before the migration boundary.

It has also been mentioned previously that how far founder sequence types are from the migration boundary also depends on where the migration period is in relation to the parts of the tree that display large numbers of bifurca-

tions. If one were artful it would be possible to find parameter sets (N, β_1, β_2) that have many coalescent events 'close' to the migration period. In such instances the difference between the founder sequence type dates and the true migration period dates would be small. This would be an artificial way of making the method of founder analysis look to be doing something sensible by assuming that the date of the founder sequence closely matches that of the founder sequence type!

The above observation regarding the decrease in population size going back in time necessitates a more careful approach to investigating the method of founder analysis. Ideally, it would be best to ensure that the population size up until a given (chosen) point in the past is not too small. A little thought is all that is required to realise that such an approach, under the model of much larger population expansion in deme 2 than deme 1 (deme 1 possibly not expanding), would require the population size of deme 2 potentially to be much larger than the population size of deme 1 at the present. Fortunately, the model already developed is flexible enough to allow such an idea to be incorporated.

3.12 Simulation 4: Regulating the initial population size and c_i fractions

3.12.1 The model

In all previous models the initial population sizes in each deme have been equal. In this section a method is described which, in the case of unequal exponential expansion in each deme, will result in the initial population sizes of each deme being different.

As before, we denote the total initial population size as N. This is now an unknown quantity, but we always have

$$\begin{aligned} N &= N_1(0) + N_2(0) \\ &= N_1 + N_2(0), \text{ if } N_1(t) = N_1(0) \equiv N_1 \forall t. \end{aligned} \tag{3.37}$$

Define N_e to be the effective population size of deme 2 (the deme that is receiving the migrants going forward in time). It is worth noting here that the effective population size for a subdivided population is not the same as the effective population size of a single population that we are using here. [47] gives a detailed explanation of some of the different effective population sizes that arise in different settings, and it is proved that (page 95) 'the effective is always larger than the actual population size and can be much greater when $4Nm$ is small', this proof being based on work of Nei and Takahata [48]. However, it is important to realise we are defining N_e to be the effective population size in a *single* deme and looking at this in isolation (and not the effective population size in the full subdivided population model), but accept that this is an *approximation*, as the arrival of immigrants from deme 1 does make it more likely that it will experience more variability than a

single isolated population of the same size. Then, measuring time in units of generations,

$$\begin{aligned} \frac{1}{N_e} &= \frac{1}{T}\sum_{t=0}^{T}\frac{1}{N_2(t)} \qquad (3.38) \\ &\approx \frac{1}{T}\int_0^T \frac{1}{N_2(t)}dt \\ &= \frac{1}{T}\int_0^T \frac{1}{N_2(0)e^{-bt}}dt \\ &= \frac{1}{T}\frac{1}{c_2 N}\left[\frac{1}{b}e^{bt}\right]_0^T, \qquad (3.39) \end{aligned}$$

Since $N_2(0) = c_2 N$, where c_2 is the fraction of the initial total population size in deme 2 (c_1 is analogously defined). So,

$$\begin{aligned} \frac{1}{N_e} &\approx \frac{1}{T}\frac{1}{c_2 N}\left[\frac{1}{b}e^{bt}\right]_0^T \\ &= \frac{1}{T}\frac{1}{c_2 N}\frac{1}{b}\left(e^{bT}-1\right). \qquad (3.40) \end{aligned}$$

It is worth noting that, as $b \to 0$,

$$\frac{\left(e^{bT}-1\right)}{b} \to \frac{(1+bT)-1}{b} = T$$

and thus, $N_e \to c_2 N$, as one would hope.

We now fix the initial size of deme 1, i.e. $Nc_1 = \Omega$. Then, from (3.40), we have (see Appendix B for detailed derivation),

$$N \approx \Omega + N_e\frac{1}{T}\frac{1}{b}\left(e^{bT}-1\right). \qquad (3.41)$$

Now, from (3.41), we have a way to estimate N, the initial combined population size, given a value for b, the population expansion rate per generation in deme 2, and an arbitrary time point in the past, T, in generations. From the obtained N, one can then calculate the relevant $\beta = bN$, and the initial population sizes in each deme using $N_2(0) = N - \Omega$ and $N - N_2(0) = N_1(0) = N_1$.

Once $N_1(0)$ and $N_2(0)$ are known, the fraction of the initial starting population that is present in each deme can be calculated using $c_1 = N_1/N$ and $c_2 = N_2(0)/N$.

So, by setting values for Ω and b and N_e, all of the variables necessary for the process to be simulated can be derived, and the procedure described above ensures the effective population size of the second deme is N_e, while at the same time taking into account that the population has been expanding at rate b per generation from T generations ago. This ensures that the population size at generation T is still 'large', and that the assumption for the coalescent process to be valid, that the population size is much larger than the sample size, is upheld, at least up to a point T generations into the past. (Although one might expect that violation of this assumption would invalidate the coalescent approximation, it has been shown [49] that, in some situations, inference that cannot usually be done under the normal coalescent framework with the population size assumptions can in fact be done when the sample size equals the population size.)

However, the main benefit of the above approach is that one is now confident that, up until a time T generations in the past, the population size in deme 2 will remain large and the situation experienced in the previous model, where the population size at the start of the migration period was as low as four individuals, should not be encountered.

Investigating founder times under the new model

The value of $\Omega = Nc_1$ was set to $10,000$, N_e to $10,000$, and T, the time point in generations in the past which would be used to calibrate the parameters was set to 820 (or $20,500$ years, the date used in the past that represented

the end of the migration period). The choice of T, for the moment, is fairly arbitrary. The expansion rate b for deme 2 was set to values between 10^{-2} and 10^{-8}. These values and the values of the derived parameters are shown in table 3.3. Note that the b value was varied initially to try to obtain β values that would allow the founder times to be investigated across a sensible range of β values, and that, for the low b case, we obtain $N \approx 20,000$ and $c_1 \approx c_2 \approx 0.5$, which is what we would expect to see in the case where both demes were experiencing almost identical demographic histories.

Table 3.3: Parameter settings for simulations.

	Nc_1	N_e	T	b	β	c_1	c_2	N
1	10000	10000	820	0.010000	44489	0.002248	0.9978	4448963
2	10000	10000	820	0.005324	1000	0.0532	0.9468	187985
3	10000	10000	820	0.003300	203.36	0.1623	0.8377	61623
4	10000	10000	820	0.003000	160.55	0.1869	0.8131	53515
5	10000	10000	820	0.001000	25.494	0.3923	0.6078	25494
6	10000	10000	820	0.000500	11.181	0.4472	0.5528	22361
7	10000	10000	820	0.000100	2.042	0.4897	0.5103	20421
8	10000	10000	820	10^{-8}	0.00002	0.5	0.5	20000

For each parameter set 50 trees were simulated, starting with 250 samples in each deme. The forward migration rate m was set to 0.01 and the difference between the founder sequence type times and the migration boundary recorded as before. Figure 3.10 shows boxplots of the difference between the founder sequence type time and the migration boundary together with histograms of the actual distributions of the difference between the founder

sequence type times and the start of the migration period.

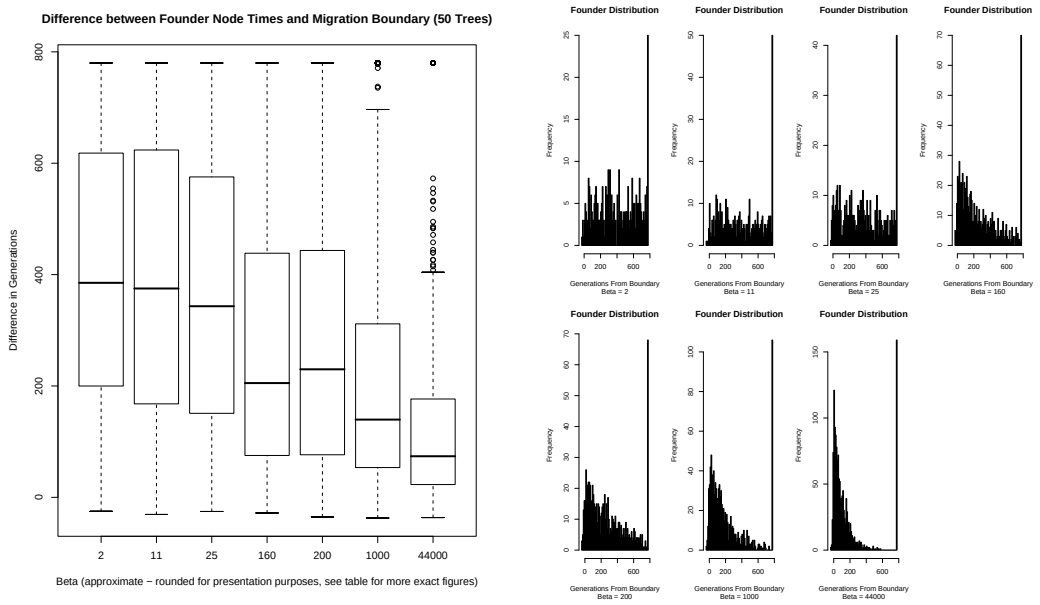

Figure 3.10: Summary plots of the discrepancy between the founder sequence type dates and the start of the migration period when N_e is $10,000$.

It is important to notice that, even when the population expansion rate in deme 2 is extremely high, founder nodes still exist which are the maximum time away from the migration boundary that they could be (e.g. $\approx$ 800 generations). These founders account for the spike at the right-hand side of each of the distributions. It is notable however that the distribution of founder times appears to move from being approximately uniform (ignoring the spike discussed already) when the expansion rate is very low, to becoming right skewed when the population expansion rate in deme 2 increases. This move *could* suggest that the method of founder analysis will indeed perform better as the phylogeny becomes more 'star like', and, regardless of whether this is the case or not, it is clear that in all instances the number of founder

sequence types that occur *far* from the migration periods is particularly large and worth attempting to address.

3.12.2 Changing N_e

The effect of changing N_e was briefly investigated for a more limited range of parameter values. N_e was set to be 1000, while Nc_1 was held at $10,000$ as before. Some limited evidence [37, Table 1, page 188] suggests that migration from the Near East to Europe perhaps involved a larger population in the Near East than in Europe, as migration into Europe was possibly then followed by expansion of the founders as they colonised Europe through breeding. This is some justification for looking at a reduced N_e with all other things being equal. Table 3.4 shows the parameter values used for this simulation. The interest here in this final simulation model is to bring the model to a level that it could be considered to resemble one possible view about the migration of modern humans into Europe, and to demonstrate that the method of founder analysis is likely to be biased with its inherent assumption that the dating of founder sequence types is representative of the date of the founder events.

Table 3.4: Parameter settings for simulations.

	Nc_1	N_e	T	b	β	c_1	c_2	N
1	10000	1000	820	0.0001	1.1042	0.9056	0.0944	11042
2	10000	1000	820	0.003	43.0547	0.6968	0.3032	14351
3	10000	1000	820	0.01	4538.964	0.0220	0.9780	453896

Figure 3.11 shows the results and should be compared to figure 3.10. The

first thing to notice about the $N_e = 1000$ case is the decrease in the number of founders. This is expected simply because we have reduced the effective population size 10-fold. As the population size decreases, the time until two lineages share a common ancestor decreases. Thus, the number of lineages available for migration when N_e is reduced is stochastically smaller than when N_e is larger.

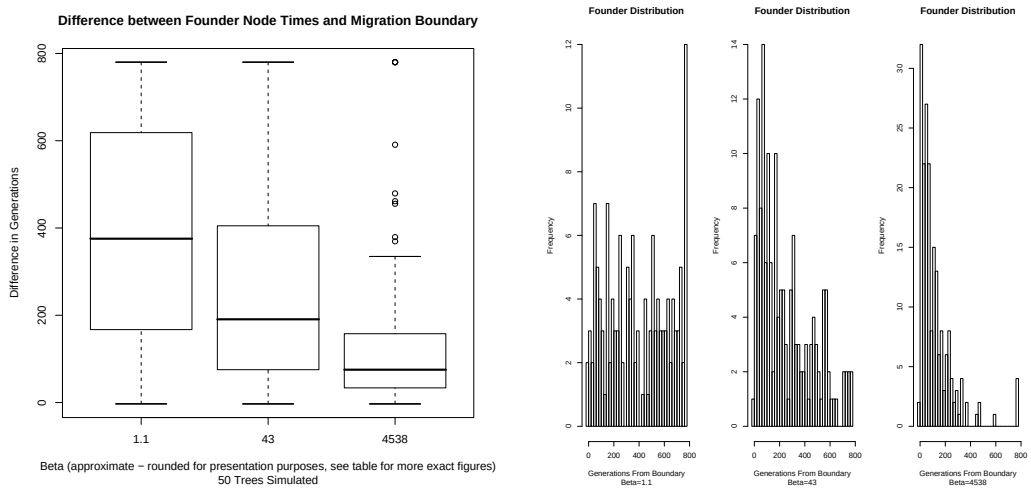

Figure 3.11: Summary plots of the discrepancy between the founder sequence type dates and the start of the migration period when N_e is reduced to $1,000$.

It also appears to be the case that the spike at ≈ 800 generations is less evident for the $N_e = 1000$ case, for (non-zero) expansion rates. This can be attributed to the faster coalescence rate within deme 2 due to the smaller effective population size. As the coalescence rate increases, the number of lineages available for migration *that have not been involved in a coalescent event already* (and thus, would give rise to a difference of approximately 800 generations between the founder sequence type time and the migration period) is decreased, meaning that the spike is less evident as the expansion rate increases. It is apparent for these parameter values and model that an

increase in b from 0.0001 to 0.03 results in the median difference between the founder sequence type time and migration time decreasing by about 200 generations, although there still is a visible problem that one would wish to address in any inferential process based on the ideas of founder analysis.

It has already been mentioned that obtaining trees of reasonable depth is an important consideration, so the depths of the trees for the two expansion cases described previously are shown in figure 3.12. It can be seen that the point T, chosen to be the time in generations to calibrate the other parameters, also gives some indication of the likely depths of the tree, although the huge variability in the coalescent process can be seen from the figures. The well known fact that, as the exponential growth rate increases, the expected length of the tree decreases is clearly visible from figure 3.12, as, going back in time, as b increases, the population is getting smaller faster, this forces the process to end sooner. The reader may be a little concerned about the depth of the trees for the $\beta \approx 4500$ case, with the median of the tree depths across the 50 simulated trees being close to the time of the migration period. This, in part, explains why the difference between the founder sequence type dates and the start of the migration period is smaller for this parameter set. By the time the migration period is entered a large part of the tree has coalesced, and very few founder sequence types will occur on tree edges which extend to the bottom of the tree, so differences in the order of hundreds of generations as seen in other cases are not likely to occur.

3.12.3 What this process is actually doing

It is worth suggesting at this point what our current model may be doing in terms of how close the founder sequence types will be to the designated

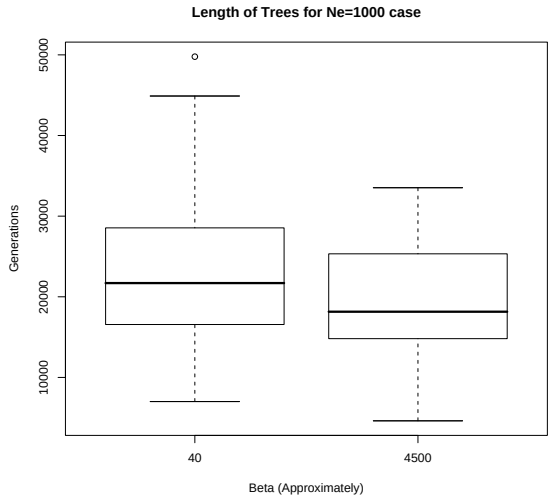

Figure 3.12: Boxplot of depth of Trees when $N_e = 1000$.

migration boundaries. The time T can be viewed as roughly defining some sort of approximate end-point of the process (accepting that this is a vast over-simplification, as for the process to end all lineages must be in the same deme to allow coalescence of the entire sample), as the effective population size is largely determined by the times during which the population is small. For a given fixed T and N_e, an increase in b will cause the tree to become more star-like. This should reduce the difference between the founder sequence type date and the migration boundary time, on average, assuming that the coalescent events still take place within the same epoch - in essence, the expansion rate increasing simply reduces the number of tree edges which extend down to the present day, which in turn reduces the number of founder sequence types that occur at the maximal time from the migration period. This is satisfactory (assuming the method of generating N by choosing values for N_e and Nc_1 is appropriate) for the case of two epochs of no migration separated by an epoch of migration. What is unclear however is how such

a model will behave when more than a single migration epoch is permitted. This concern is what is looked at in the following final simulation.

3.13 Two migration epochs

Assume now that we have five epochs, an epoch of $19,500$ years, during which there is no migration, followed by $1,000$ years of migration, followed by a further $19,000$ years of no migration, then another period of $1,000$ years of migration, followed again by $19,000$ years of no migration, then a final burst of migration corresponding to initial settlement. Thus, the epoch boundaries are at 0, 19500, 20500, 39500, 40500, 59500, ∞. Set $T = 1620$ generations ($40,500$ years, the end of the second migration period) as being the point to calibrate N, and suppose $\Omega = Nc_1 = 10,000$. The parameter settings for this simulation are tabulated in table 3.5.

Table 3.5: Parameter settings for two epoch simulation.

Nc_1	N_e	T	b	β	c_1	c_2	N
10000	10000	1620	0.000001	0.02	0.4998	0.5002	20008
10000	10000	1620	0.001	35.02	0.2856	0.7144	35019
10000	10000	1620	0.0015	78.94	0.19	0.81	52629
10000	10000	1620	0.002	171.44	0.1167	0.8833	85721
10000	10000	1620	0.003	820.27	0.037	0.963	273424
10000	10000	1620	0.01	66997130	10^{-6}	≈ 1	6699713024

This simulation serves two purposes. Most importantly, it demonstrates that using founder sequence type dates in a model with multiple migration

periods can lead to very unfortunate situations where the natural conclusions an investigator might like to believe are incorrect, and the dating based on founder sequence types leads to the date estimates of founding events being *completely inappropriate*, with the migration event being wrongly assumed to have occurred during a more recent migration period than it actually did.

The assumption that the founder sequence type should in some way be close to the founding event is seen in this simulation (figures 3.13 and 3.14) to lead to unfortunate situations where the founding event takes place during the later migration period ($39,500$ - $40,500$ years), but occurs on an edge that can extend to near the very bottom of the tree. Assuming that one has a sensible way to date such founder sequence types (which is discussed in the next chapter, and merely assumed for now), it is clear that even any unbiased estimator of the date of such founder sequence types does not necessarily reflect the true date of the founding event, and in many cases may be tens of thousands of years from the true founding event.

A second purpose of this simulation is to demonstrate that, although the magnitude of the discrepancy between the founder sequence type date and the true founding event date is something which is often heavily dependent on the choice of parameters made by the investigator, together with the location of the migration boundaries, in any real example one would wish to consider multiple migration periods. In this more general case, the problem seen in the previous simulations is one which will always be present, and without a doubt it is a problem which should be addressed in any inference method based on the idea of founder events which do not nicely occur on the nodes of any reconstructed phylogeny. Assuming this in any inference procedure does seem unreasonable.

It is interesting that Richards et al. [26] did not worry about this problem, instead focussing on the consequences of issues such as back-migration and recurrent mutation. I was fortunate to be able to talk to Prof. Richards at various points throughout my research and it became clear that this problem was never considered at all, and even the distinction between founder sequence type and founder (which I have since defined) was not ever considered necessary. The consequences of this problem is bias in the founder age estimates, and this bias can only result in the founder dating being too *young*. It is of interest to note that, although bias is an undesirable property, there is an irony in the *direction* of the bias as criticism about the date estimates obtained from the original founder method assumes the estimates are too *old*, as described in Chapter 2. The simulations undertaken and the problem identified show further evidence that the criticisms about the method are related to a misunderstanding about what it is actually doing, and one would hope that work such as these simulations and the extensions to founder analysis which I will propose in the following chapters will help clear up any misunderstandings.

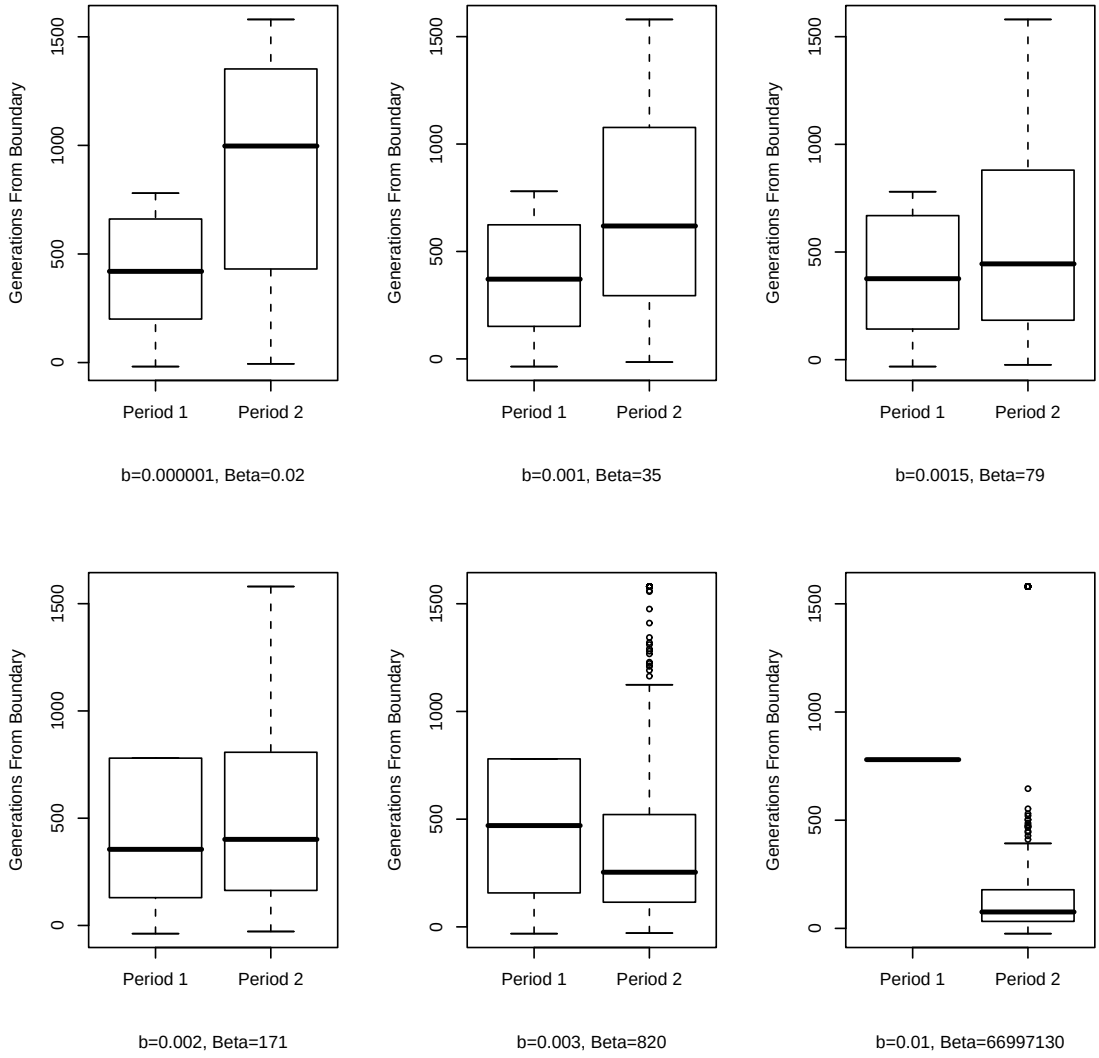

Figure 3.13: Boxplots of the difference between the founder sequence type dates and the migration boundaries for the five epoch case.

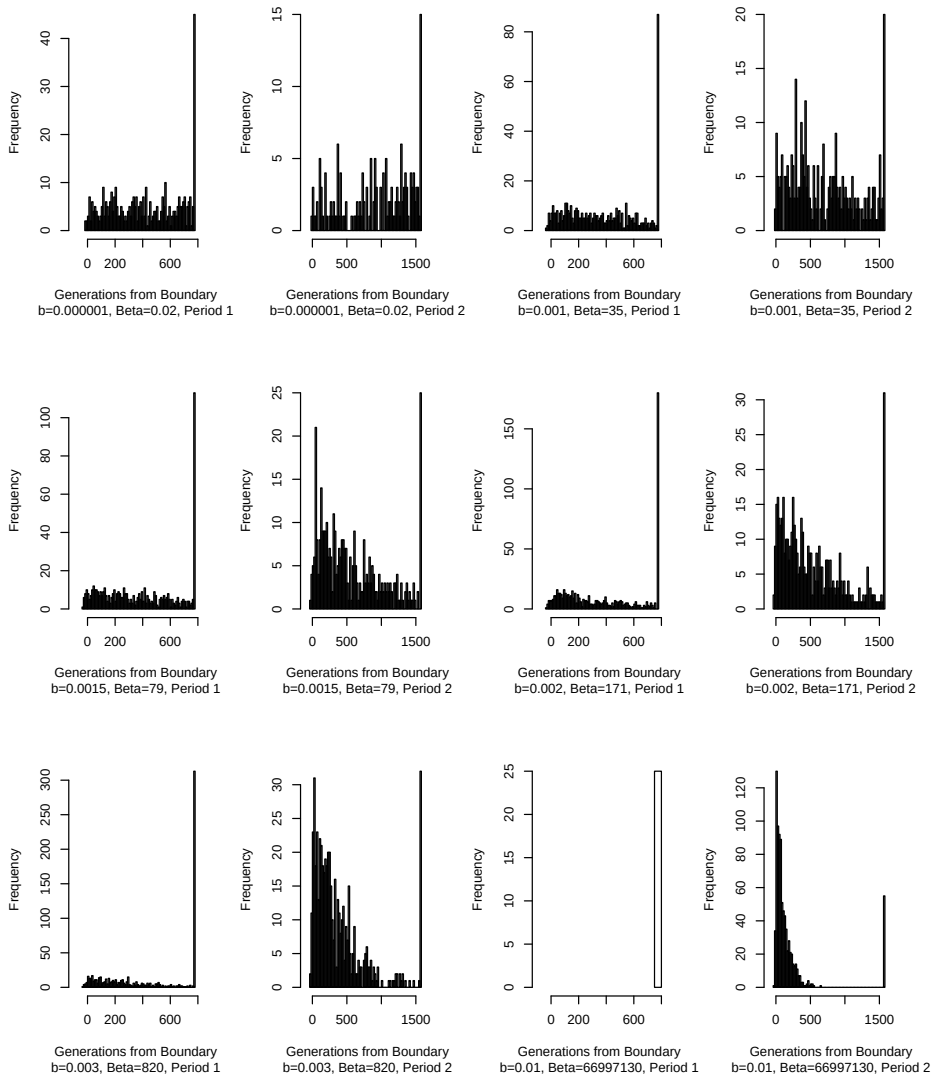

Figure 3.14: Histograms of the difference between the founder sequence type dates and the migration boundaries for the five epoch case.

Chapter 4

Founder analysis extension

4.1 The ρ estimator of divergence time

Forster et al. [35] describe an estimator of the 'arrival time of each founding control region sequence' which they define as ρ, the average number of sites differing between a set of sequences and a specified common ancestor. This estimator is simply the 'average distance to the root', as used in previous work [50]. In this section some properties of the ρ estimator are derived. The formulae, notation and derivation of properties below do not follow that of [35], but incorporate the ideas and notation of [51] and [52], with some modifications to ensure consistency in what follows. It also should be noted that some of the properties described below are not formally derived in the literature, but are routinely assumed. Further, 'paths' are defined and the ρ estimator expressed using the concept of 'paths up a tree'. Furthermore, the concept of a ρ *mutation* is something I wish to introduce. The ρ statistic is important in founder analysis as it is the estimator that was used to date the founder sequence types. Although it has been shown through simulation that this estimate does not always coincide with the event one wishes to estimate

(the actual founding date), the properties of this estimator are of interest in their own right.

Define u to be the expected number of scored mutations per coalescent time unit across the length of a sequenced segment (the mutation rate). Further, assume the number of mutations on a given tree edge of length t_i coalescent time units is a Poisson distributed random variable, T_i, with parameter $\mu_i = t_i u$. A bifurcating tree with n external nodes has $k = 2n-2$ edges of lengths $t_1, t_2, \ldots, t_k$. Edge i defines a clade which has n_i descendants in total. The coalescent time of the sample, t, can be expressed as

$$t = \frac{1}{n}\sum_{i=1}^{k} n_i t_i. \tag{4.1}$$

Now, consider the random variable $T = \frac{1}{n}\sum n_i T_i$. Then, given the t_i,

$$\begin{aligned}
E[T] &= E\left[\frac{1}{n}\sum_{i=1}^{k} n_i T_i\right] \\
&= \frac{1}{n}\sum_{i=1}^{k} n_i E[T_i] \\
&= \frac{1}{n}\sum_{i=1}^{k} n_i t_i u, \qquad (\text{since } T_i \sim \text{Po}(t_i u)) \qquad (4.2)\\
&= u\left[\frac{1}{n}\sum_{i=1}^{k} n_i t_i\right] \\
&= ut, \quad (\text{from } (4.1)). \qquad (4.3)
\end{aligned}$$

Further,

$$\begin{aligned}
V[T] &= V\left[\frac{1}{n}\sum_{i=1}^{k} n_i T_i\right] \\
&= \frac{1}{n^2}\sum_{i=1}^{k} n_i^2 V[T_i] \qquad (\text{since } T_i \text{ independent, given } t_i)
\end{aligned}$$

$$= \frac{1}{n^2}\sum_{i=1}^{k} n_i^2 t_i u. \tag{4.4}$$

One sees from (4.3) that the random variable T has an expected value equal to the coalescent time, t, multiplied by the mutation rate, u. In practice, however, the number of mutations on a given edge of the tree will rarely be equal to its expected value. Denoting the number of *observed* mutations on edge i from an inferred phylogeny by l_i, and using this as an estimate of $t_i u$, the statistic ρ can be calculated for any given internal node, node q say, by the formula

$$\rho = \frac{1}{n}\sum_{i \in D_q} n_i l_i, \tag{4.5}$$

where n is the number of external nodes (or external edges) below node q, n_i is the number of descendants of node i, and D_q is the set which contains all labels of edges below q, both internal and external. In summary, the estimator ρ is simply the random variable defined above as T, when it is evaluated with $t_i u = l_i, \forall i$.

Further, an estimator for the variance of ρ [51] follows by replacing $t_i u$ with l_i in equation (4.4):

$$\hat{\sigma^2} = \frac{1}{n^2}\sum_{i=1}^{k} n_i^2 l_i. \tag{4.6}$$

An alternative expression for ρ is possible by defining 'paths up a phylogeny'. Consider a tree with n external nodes/edges. Denote the path up the tree from external node $j, j = 1, 2, ..., n$, to the common ancestor of the nodes by 'path' $\wp_j$. Further, denote the number of mutations on path $\wp_j$ by $M_{\wp_j}$. Then the ρ statistic can be re-expressed with this new notation as

$$\rho = \frac{1}{n}\sum_{j=1}^{n} M_{\wp_j}. \tag{4.7}$$

Although this description of ρ in (4.7) may be more intuitive than that of (4.5) (nicely demonstrating why this estimator can be described as the average distance to the root/node), it does not allow an estimator of the variance of ρ to be derived. However, (4.7) has been described, and 'paths' defined, as they allow much simpler descriptions in some situations which will follow.

4.1.1 Further properties of the ρ statistic

One can re-express (4.4) as

$$\begin{aligned} V[T] &= \frac{1}{n^2}\sum_{i=1}^{k} n_i^2 t_i u = \frac{1}{n}\left\{\frac{1}{n}\sum_{i=1}^{k} n_i^2 t_i u\right\} \\ &\geq \frac{1}{n}\left\{\frac{1}{n}\sum_{i=1}^{k} n_i t_i u\right\}, \quad \text{since } n_i \geq 1. \\ V[\rho] &\geq \frac{1}{n}\left\{\frac{1}{n}\sum_{i=1}^{k} n_i l_i\right\} = \frac{1}{n}\left\{\frac{1}{n}\sum_{i\in D_q} n_i l_i\right\} \\ &= \frac{1}{n}\rho, \qquad \text{by recognising the formula for } \rho. \end{aligned} \tag{4.8}$$

Saillard et al. [51] note that, in the case of a perfect star phylogeny, (4.8) becomes an equality. Torroni et al. [53] define a 'star index', which is a score between 0 and 1, essentially a frequency measure of how often pairs of sequences coalesce in the root of the tree. A star index score of 1 (which arises for a perfect star tree) yields an equality for (4.8). Formula (4.8) is important as it allows a lower bound on the variance of the estimator. Although a minimum bound on the variance may not seem immediately useful, how

close this minimum bound is to the true variance can be (albeit subjectively) assessed by computing the star index for a given observed phylogeny. Saillard et al. [51] define ρ/σ^2, rounded to the nearest integer, to be the "effective star size"; this is 'the size of a perfect star sample with approximately the same values of ρ and σ as the given [observed] sample' [51, page 721]. They further define the "efficiency" of the sample as $\rho/(n\sigma^2)$.

An approximate upper bound on the variance is provided by Thomson et al. [52], who use different notation, $\hat{T}$ instead of ρ, and work within a framework similar to that described previously concerning 'paths', although they do not specifically describe the estimator in this manner. Thomson notes that the variance of $\hat{T}$ would be less than the variance one would obtain by picking a single random sequence and using that alone to estimate the variance of the time to the given node in question.

4.1.2 Consistency of the ρ statistic?

At present, no work has been published regarding the consistency of the ρ estimator. Proving consistency (or lack of) is a difficult problem due to the dependency of the estimator. To see this problem clearly, one needs to distinguish between a mutation and a ρ *mutation*. Define a ρ *mutation* to be a mutation present on an internal branch of a phylogenetic tree (see figure 4.1).

This distinction between mutations on the internal and external branches of a tree is necessary to allow clear explanations below. These ρ mutations are the reason why establishing consistency (or not) of the ρ estimator is difficult. Consider, for simplicity, the standard coalescent with no migration or population expansion. Kingman [54] has shown that the time to the

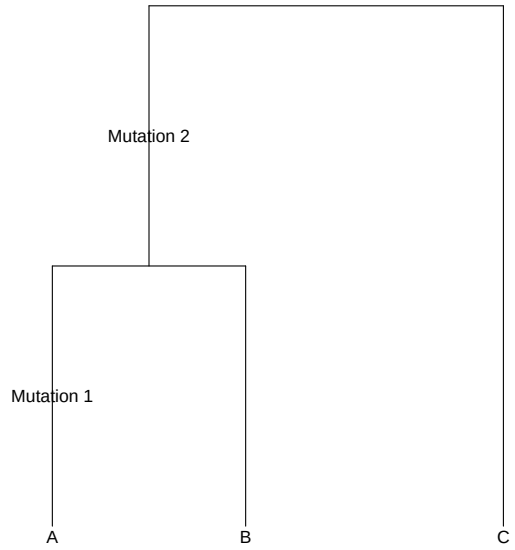

Figure 4.1: Plot showing both a standard mutation (Mutation 1), and a 'ρ mutation' (Mutation 2).

most recent common ancestor is finite, even for a sample of infinite size. As the sample size tends to infinity, i.e. as the number of external edges of a tree grows to infinity, the ρ mutations on the innermost branches of the tree contribute more and more to the ρ estimator. Certainly, in the case of the standard coalescent, increasing the sample size results only in the tree displaying a larger and larger number of (small) external or near external branches (and thus, carrying few mutations). However, these small external edges can in fact dramatically alter the ρ value, particularly when the number of sequences within the cluster of interest is low (as will be the case for some founder clusters). This is shown in figure 4.2.

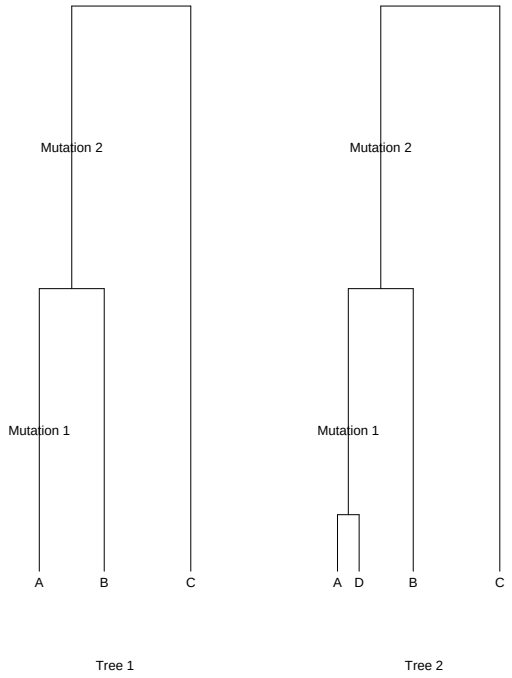

Figure 4.2: Plot showing the addition of a single external edge to a phylogenetic tree. Tree 1 has a ρ value of 1. However the addition of a new sequence, D, which coalesces with A close to the present, results in a large increase in ρ, becoming 5/4.

At a more fundamental level though, consistency is a difficult question as even what is the 'sample size' that is to be considered is unclear. One could argue that the sample size increase that should be considered is not that of the number of external nodes/sequences, but that of the number of sites or length of the DNA sequence under consideration. Conceptually, it is feasible to suggest that, as the number of sites sequenced increased to infinity, the tree would become better resolved, and the variance of the estimator would decrease. Consistency of this estimator is something which has not been

explored in the scientific literature. Despite the consistency of this estimator being in doubt, the ρ estimator provides a useful tool in estimating, in an unbiased manner, the time of given nodes in any phylogenetic tree. In the sections that follow the estimator is used as the starting point of a Bayesian method to estimate the times of sequence migration.

It is perhaps interesting to note that one may derive an analytic formula for the distribution of the number of mutations along a single path (as defined previously), under the standard n-coalescent, using the theory of 'Phase-Type Distributions' [55]. The derivation I considered during my research allowed all moments to be calculated analytically for a single path under some specific assumptions, but did not help in establishing consistency, as multiple paths overlapped. The distribution of the number of mutations along the entire length of the tree is a much more complicated object as a result of this dependency.

4.1.3 Recent criticisms of the ρ estimator

Recent work [56] tries to cast some doubt on the 'accuracy' of the estimator and attempts a 'validation exercise'. A reasonable-seeming simulation procedure based on various demographic models (as well as a basic n-coalescent) is presented. It is unfortunate however that equations (2) and (3) [56, page 339], which are supposed to be the expected value and variance of ρ, are incorrect due to a misunderstanding of what the sum is over, which should be all *edges* in the tree, not 'unique haplotypes sampled from n individuals', as described. The tone of the paper is unfortunate and seems to suggest that Cox is unconvinced about the mathematical properties of the ρ estimator. It is stated that 'Forster et al. (1996) suggest that multiplication of

the ρ statistic with a known mutation rate scalar yields an unbiased estimator of molecular age for the given ancestral node in real chronological time.' Cox proceeds to examine the 'bias and variance of point estimates of dates obtained from a simple constant-size population'.

In the simulation results presented, it is claimed that the ρ statistic is (among other things) biased. This is mysterious when the simple result, my (4.3), invalidates Cox's analysis and conclusions. It is unfortunate that this paper starts by giving incorrect equations for the expected value and variance of this estimator, and I believe that some of the results in this paper have been created using incorrect formulae.

However, Cox does make some important notes in his discussion section that are true and often overlooked, namely that the *date* estimates that arise from ρ calculations are totally dependent on the mutation rate used. However, the ρ estimate itself (non-scaled by mutation rate) has the same meaning in every case. It is also mentioned that mutation rate can change over time, which would indeed make date estimates from ρ extremely difficult to justify and interpret. A third point is that of different mutation rates in different areas of the region sequenced. However, by a very simple property of the Poisson distribution it is only the average rate that is relevant. Cox mentions as a fourth problem the fact that often a single tree is not found, and instead a network due to recurrent mutations may arise. Indeed, the ρ estimates obtained are conditional on the assumed tree. A more questionable part of the discussion is that regarding choice of locus. I would argue that, although the choice of locus could affect the date estimates obtained, any dating estimate is conditional on the tree that results from the region that was sequenced, and this does not have any bearing on the statistical properties

of the estimator.

Note that I am not claiming that this estimator cannot be improved upon (for example, any estimator that down-weighted the mutations on the internal branches would presumably have smaller variance), but hold the position that it is a good starting point with which to date nodes of a tree in an unbiased manner.

4.1.4 Estimating migration time using ρ

Recall from the previous simulation chapter that the migration time of a given sequence was not always close to any given node on a phylogenetic tree. This problem exists simply because migration events do not necessarily coincide with any coalescent events.

4.1.5 Connected star trees

The method of founder analysis assumed star-tree topologies for founder clusters, together with the implicit assumption that the migration events were followed by a period of rapid expansion. From previous simulations it has been seen that the time of migration events may in fact not coincide with the time of internal nodes in the phylogenetic tree (these nodes denote possible founder sequence types). This section introduces the concept of a connected star tree, which is used in what follows to *bound* the migration time of a single sequence (founder) between two (unbiased) estimates.

Consider the tree shown in figure 4.3, where we imagine a migration event on the edge connecting the *subtree* consisting of $n_a = 4$ sequences (itself a perfect star tree), with the other $n_b - n_a = 9 - 4 = 5$ edges, which form the

'comb' of the tree. Note that in general this whole tree is itself a sub-tree of the full tree of the sample.

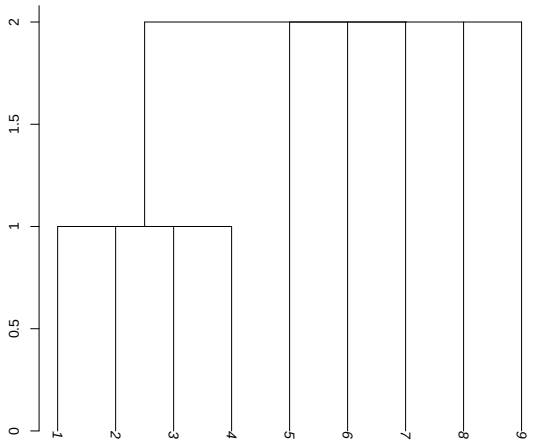

Figure 4.3: A 'connected star tree' with $n_a = 4$, $n_b = 9$, $\tau_A = 1$, $\tau_B = 2$.

Assume that the migration event of interest takes place (at time τ) on the connecting edge, that the time of the coalescence of the first n_a sequences on the subtree is τ_A, and that the time of coalescence of the whole tree is τ_B. Founder analysis attempts to estimate the time to a given node that was in some sense 'close' to the migration event (τ_A) and use this as an estimate of the migration time (τ). When estimating migration times, one is actually trying to estimate a time which falls *between* two estimated sequences (τ in the current notation), not the time that a given inferred DNA sequence arose or the time to some most recent common ancestor of a subset of sequences.

To this end, a model which assumes what shall be called a 'connected star tree' is proposed, an example of which is shown in figure 4.3. This improves

on what is considered in founder analysis by directly addressing the issue that one wishes to estimate a time that does not directly correspond to any inferred DNA sequence on a phylogenetic tree, while allowing the star-tree assumption to be relaxed slightly. The strength of this approach will be seen shortly when an analytic formula is derived for the joint probability distribution of the quantities we are interested in ($P[\tau, \tau_A, \tau_B]$). The method is based on the fact that, for a given connected star tree, we can use the ρ estimator for two internal nodes to be bounds on the true migration time we wish to estimate (τ, in mutational units). One then can derive the joint probability density of the τ's given the ρ's. The exact details are derived in the next section.

4.1.6 Deriving the joint distribution of interest

Suppose one calculates ρ_A and ρ_B for a given founder cluster and its containing cluster respectively, which are our (unbiased) estimates of τ_A and τ_B. The full joint density can then be derived as follows:

$$\begin{aligned} P[\tau, \tau_A, \tau_B, \rho_A, \rho_B] &= P[\tau|\tau_A, \tau_B, \rho_A, \rho_B] P[\tau_A, \tau_B, \rho_A, \rho_B] \\ &= P[\tau|\tau_A, \tau_B, \rho_A, \rho_B] P[\rho_A, \rho_B|\tau_A, \tau_B] P[\tau_A, \tau_B] \\ &= P[\tau|\tau_A, \tau_B] P[\rho_A, \rho_B|\tau_A, \tau_B] P[\tau_A, \tau_B], \end{aligned} \tag{4.9}$$

since τ is conditionally independent of ρ_A and ρ_B, given τ_A and τ_B.

Furthermore,

$$P[\tau, \tau_A, \tau_B, \rho_A, \rho_B] = P[\tau, \tau_A, \tau_B|\rho_A, \rho_B] P[\rho_A, \rho_B]. \tag{4.10}$$

Thus,

$$P[\tau, \tau_A, \tau_B|\rho_A, \rho_B] = \frac{P[\tau|\tau_A, \tau_B] P[\rho_A, \rho_B|\tau_A, \tau_B] P[\tau_A, \tau_B]}{P[\rho_A, \rho_B]} \tag{4.11}$$

$$\propto P[\tau|\tau_A, \tau_B] P[\rho_A, \rho_B|\tau_A, \tau_B] P[\tau_A, \tau_B], \tag{4.12}$$

by an application of Bayes' theorem.

At this point one notes that, within a Bayesian framework, $P[\tau|\tau_A, \tau_B]$ can be viewed as a prior distribution on τ, given τ_A and τ_B, while $P[\tau_A, \tau_B]$ can be viewed as a prior distribution on τ_A and τ_B. This leaves only $P[\rho_A, \rho_B|\tau_A, \tau_B]$ unspecified.

4.2 The distribution of $\rho_A, \rho_B | \tau_A, \tau_B$

Assuming conditional independence, and working with $n_a\rho_A$ (the product of the number of edges in the subtree multiplied by the value of ρ_A, which is simply the total number of mutations on the subtree), as opposed to ρ_A, $P\left[n_a\rho_A, n_b\rho_B|\tau_A, \tau_B\right]$ is the product of a standard Poisson random variable (under the infinite-sites model) coming from the smaller subtree:

$$P\left[n_a\rho_A = \gamma|\tau_A\right] \;=\; \frac{\left(n_a\tau_A\right)^{\gamma}\exp\left\{-n_a\tau_A\right\}}{\gamma!}, \tag{4.13}$$

multiplied by the contribution from $P\left[n_b\rho_B|\tau_A, \tau_B, n_a\rho_A\right]$. Note that (4.13) is simply a Poisson random variable with parameter $n_a\tau_A$, the total length of the subtree.

Deriving an analytic formula for the distribution of $n_b\rho_B$ given $\tau_A, \tau_B, n_a\rho_A = \gamma$ is more difficult due to the connecting edge which carries 'ρ mutations' (all other edges on the tree can carry mutations, but each mutation is counted only once in the current connected star tree framework). Below shows the formula for the probability that $n_b\rho_B = k$ given τ_A and τ_B, together with the observed number of mutations (γ) on the subtree. The proof of this result is covered in the following sections.

$$\begin{aligned} & P\left[n_b\rho_B = k|\tau_A, \tau_B, \gamma\right] \\ = & \sum_{j=0}^{\left\lfloor\frac{k-\gamma}{n_a}\right\rfloor} \frac{\left[\left(n_b - n_a\right)\tau_B\right]^{k-\gamma-n_aj}\left(\tau_B - \tau_A\right)^j e^{-\left[\left(n_b-n_a\right)\tau_B + \left(\tau_B-\tau_A\right)\right]}}{j!\left(k-\gamma-n_aj\right)!} \\ = & \sum_{j=0}^{\left\lfloor\frac{k-\gamma}{n_a}\right\rfloor} \frac{\mu^{k-\gamma-n_aj}\eta^j e^{-(\mu+\eta)}}{j!\left(k-\gamma-n_aj\right)!}, \end{aligned} \tag{4.14}$$

where $\eta = \tau_B - \tau_A$, $\mu = \left(n_b - n_a\right)\tau_B$ and $\lfloor x \rfloor = \max\left\{n \in \mathbb{Z} | n \leq x\right\}$. The above can be used to give an analytic representation of the joint distribution of τ, τ_A, τ_B:

$$
\begin{aligned}
& P\left[\tau, \tau_A, \tau_B | \rho_A, \rho_B\right] \propto P\left[\tau | \tau_A, \tau_B\right] P\left[\tau_A, \tau_B\right] P\left[n_a\rho_A, n_b\rho_B | \tau_A, \tau_B\right] \\
= \quad & P\left[\tau | \tau_A, \tau_B\right] P\left[\tau_A, \tau_B\right] P\left[n_a\rho_A | \tau_A\right] P\left[n_b\rho_B | \tau_A, \tau_B, n_a\rho_A = \gamma\right],
\end{aligned}
$$

where $P\left[n_a\rho_A | \tau_A\right]$ and $P\left[n_b\rho_B | \tau_A, \tau_B, n_a\rho_A = \gamma\right]$ are as stated previously.

4.3 Consequences of the factorisation

Although the ρ estimator has been shown to be unbiased in its unconditional form, factorising has the consequence of introducing ρ conditional on other random variables. This section demonstrates some consequences of such a factorisation.

Recall one of the fundamental probability objects, the probability generating function (p.g.f.). A random variable, X, taking nonnegative integer values, has probability generating function $G_X(z)$, defined to be:

$$
G_X(z) = \sum_{l=0}^{\infty} p_X(l) z^l, |z| \leq 1. \tag{4.15}
$$

Further, recall that, if such a random variable, X, is a linear combination of other independent random variables $Y_i, i = 1, 2, \ldots, M$, then the probability generating function is computed as follows. If

$$
X \quad = \quad a_1 Y_1 + a_2 Y_2 + \ldots + a_M Y_M,
$$

then

$$
G_X(z) \quad = \quad \prod_{i=1}^{M} G_{Y_i}(z^{a_i}). \tag{4.16}
$$

Now, $n_b\rho_B$ is a random quantity which is the sum of three independent random quantities, the number of mutations that fall on the subtree, the number

of mutations that fall on the connecting edge, and the number of mutations that fall on the comb. However, conditioning on γ in $P\left[n_b\rho_B|\tau_A, \tau_B, n_a\rho_A = \gamma\right]$ has the effect of *removing* some of the randomness, as the computation is conditioning on γ being fixed/already observed. Thus, $n_b\rho_B$ can be expressed in the following manner:

$$n_b\rho_B \quad = \quad \gamma Y_1 + n_a Y_2 + Y_3, \tag{4.17}$$

where $Y_2 \sim \text{Po}(\tau_B - \tau_A), Y_3 \sim \text{Po}((n_b - n_a)\tau_B)$ and Y_1 is a random variable taking on the value 1 with probability 1 (i.e. a constant).

We now go on to calculate the first two moments of this random quantity using probability generating functions, to investigate the effect of conditioning on $n_a\rho_A = \gamma$.

From (4.16) and (4.17), the p.g.f of $n_b\rho_B|n_a\rho_A = \gamma$ is
$G(z) = G_{Y_1}(z^\gamma)G_{Y_2}(z^{n_a})G_{Y_3}(z)$. Hence,

$$\begin{aligned}
G(z) &= \sum_{l_1=0}^{\infty} p_{Y_1}(l_1)(z^\gamma)^{l_1} \sum_{l_2=0}^{\infty} p_{Y_2}(l_2)(z^{n_a})^{l_2} \sum_{l_3=0}^{\infty} p_{Y_3}(l_3)(z^{l_3}) \\
&= 1(z^\gamma) \sum_{l_2=0}^{\infty} \frac{\eta^{l_2} e^{-\eta}}{l_2!} z^{n_a l_2} \sum_{l_3=0}^{\infty} \frac{\mu^{l_3} e^{-\mu}}{l_3!} z^{l_3} \qquad (4.18) \\
&= z^\gamma e^{-\eta} e^{-\mu} \sum_{l_2=0}^{\infty} \frac{\left(\eta z^{n_a}\right)^{l_2}}{l_2!} \sum_{l_3=0}^{\infty} \frac{(\mu z)^{l_3}}{l_3!} \\
&= z^\gamma e^{-\eta} e^{-\mu} e^{\eta z^{n_a}} e^{\mu z} \\
&= z^\gamma \exp\left\{\eta\left(z^{n_a} - 1\right) + \mu\left(z - 1\right)\right\}. \qquad (4.19)
\end{aligned}$$

With the p.g.f. established, one can calculate the falling factorial moments. Here, the mean of $n_b\rho_B|n_a\rho_A = \gamma$ is computed. Recall that a random variable, X, with probability generating function $G_X(z)$, has mean $E[X]$ equal to the value of the first derivative of its probability generating function, evaluated

when $z = 1$. So, from (4.19),

$$\begin{aligned} G(z) &= z^{\gamma} \exp\left\{\eta\left(z^{n_a} - 1\right) + \mu\left(z - 1\right)\right\} \\ G^{(1)}(z) &= \gamma z^{\gamma-1} \exp\left\{\eta\left(z^{n_a} - 1\right) + \mu\left(z - 1\right)\right\} \\ &+ z^{\gamma}[\mu \exp\left\{\eta\left(z^{n_a} - 1\right) + \mu\left(z - 1\right)\right\} \\ &+ n_a \eta z^{n_a - 1} \exp\left\{\eta\left(z^{n_a} - 1\right) + \mu\left(z - 1\right)\right\}] \\ G^{(1)}(1) &= \gamma + \mu + n_a\eta \\ &= \gamma + n_b\tau_B - n_a\tau_A. \end{aligned} \tag{4.20}$$

Expression (4.20) shows that $n_b\rho_B|n_a\rho_A = \gamma$ is unbiased only when $\gamma = n_a\tau_A$, i.e. only when the number of mutations on the subtree is equal to its expected value.

One could proceed further and calculate the falling factorial moments. It is simply stated here that

$$V[n_b\rho_B|n_a\rho_A = \gamma] = n_a^2\left(\tau_B - \tau_A\right) + \left(n_b - n_a\right)\tau_B, \tag{4.21}$$

whereas the equivalent formula for the variance, under the connected star tree assumption, but *not* conditioning on $n_a\rho_A = \gamma$, is

$$V[n_b\rho_B] = n_a^2\left(\tau_B - \tau_A\right) + \left(n_b - n_a\right)\tau_B + n_a\tau_A. \tag{4.22}$$

Conditioning on γ reduces the variance as the variability in the subtree is lost.

A further interesting result is shown in Appendix B. It is shown there that the covariance of $n_a\rho_A$ and $n_b\rho_B$ is equal to the variance of $n_a\rho_A$.

4.4 Single founder case

In this section, (4.14) is proved for a special (but important) case. For a single migration event on a connected star tree, the proof is given here for

the degenerate case, $n_a = 1$, by making use of the standard probability generating function approach. It was shown in (4.19) that the probability generating function of $n_b\rho_B|n_a\rho_A$ is given by $z^\gamma \exp\{\eta(z^{n_a}-1)+\mu(z-1)\}$.

Deriving the required result using probability generating functions is a non-standard induction problem, because the result required is the general formula for the k^{th} derivative of a function that contains a variable z, that needs to be evaluated with $z = 0$ for each derivative. The following lemma concerning the k^{th} derivative of the probability generating function is the starting point, noting here that $(x)_{(y)}$ is the falling factorial $(x)_{(y)} = x(x-1)(x-2)\ldots(x-y+1)$ and $S = \exp\{\eta(z-1)+\mu(z-1)\}$.

Lemma.

$$G^{(k)}(z,\gamma,n_a=1) = \sum_{j=1}^{k+1}\binom{k}{j-1}(\gamma)_{(j-1)}z^{[\gamma-j+1]}(\eta+\mu)^{[k-j+1]}S. \tag{4.23}$$

Proof (by induction).

True for $k = 1$?

$$\begin{aligned} G(z) &= z^\gamma\exp\{\eta(z-1)+\mu(z-1)\} = z^\gamma S \\ G^{(1)}(z,\gamma,n_a=1) &= \gamma z^{\gamma-1}S + z^\gamma S(\eta+\mu) \\ &= \sum_{j=1}^{2}\binom{1}{j-1}(\gamma)_{(j-1)}z^{[\gamma-j+1]}(\eta+\mu)^{[2-j]}S, \end{aligned} \tag{4.24}$$

as required.

Now suppose (4.23) is true for the $(k-1)^{\text{th}}$ derivative, i.e.

$$G^{(k-1)}(z,\gamma,n_a=1) = \sum_{j=1}^{k}\binom{k-1}{j-1}(\gamma)_{(j-1)}z^{\gamma-j+1}(\eta+\mu)^{k-j}S.$$

$$
\begin{aligned}
\text{Then } G^{(k)}(z,\gamma,n_a=1) &= \frac{d}{dz}G^{(k-1)}(z,\gamma,n_a=1)\\
&= \frac{d}{dz}\left\{\binom{k-1}{0}(\gamma)_{(0)}z^{\gamma}(\eta+\mu)^{k-1}S\right\}\\
&\quad+\frac{d}{dz}\left\{\binom{k-1}{1}(\gamma)_{(1)}z^{\gamma-1}(\eta+\mu)^{k-2}S\right\}\\
&\quad+\ldots+\\
&\quad+\frac{d}{dz}\left\{\binom{k-1}{k-2}(\gamma)_{(k-2)}z^{\gamma-k+2}(\eta+\mu)^{1}S\right\}\\
&\quad+\frac{d}{dz}\left\{\binom{k-1}{k-1}(\gamma)_{(k-1)}z^{\gamma-k+1}(\eta+\mu)^{0}S\right\}\\
&= \frac{d}{dz}\left\{1z^{\gamma}(\eta+\mu)^{k-1}S\right\}\\
&\quad+\frac{d}{dz}\left\{\binom{k-1}{1}(\gamma)z^{\gamma-1}(\eta+\mu)^{k-2}S\right\}\\
&\quad+\ldots+\\
&\quad+\frac{d}{dz}\left\{\binom{k-1}{k-2}(\gamma)_{(k-2)}z^{\gamma-k+2}(\eta+\mu)S\right\}\\
&\quad+\frac{d}{dz}\left\{1(\gamma)_{(k-1)}z^{\gamma-k+1}S\right\}\\
&= \gamma z^{\gamma-1}(\eta+\mu)^{k-1}S+z^{\gamma}(\eta+\mu)^{k}S\\
&\quad+\binom{k-1}{1}(\gamma)_{(2)}z^{\gamma-2}(\eta+\mu)^{(k-2)}S\\
&\quad+\binom{k-1}{1}\gamma z^{\gamma-1}(\eta+\mu)^{k-1}S\\
&\quad+\ldots+\\
&\quad+\binom{k-1}{k-2}(\gamma)_{(k-1)}z^{\gamma-k+1}(\eta+\mu)S\\
&\quad+\binom{k-1}{k-2}(\gamma)_{(k-2)}z^{\gamma-k+2}(\eta+\mu)^{2}S\\
&\quad+(\gamma)_{(k)}z^{\gamma-k}S+(\gamma)_{(k-1)}z^{(\gamma-k+1)}S(\eta+\mu).
\end{aligned}
$$

So,

$$\begin{aligned} G^{(k)}(z,\gamma,n_a=1) &= z^{\gamma}(\eta+\mu)^k S \\ &\quad + \left[\gamma z^{\gamma-1}(\eta+\mu)^{k-1}S\right]\left[\binom{k-1}{0}+\binom{k-1}{1}\right] \\ &\quad + \left[(\gamma)_{(2)}z^{\gamma-2}(\eta+\mu)^{k-2}S\right]\left[\binom{k-1}{1}+\binom{k-1}{2}\right] \\ &\quad + \ldots + \\ &\quad + \left[(\gamma)_{(k-1)}z^{\gamma-k+1}(\eta+\mu)S\right]\left[\binom{k-1}{k-2}+\binom{k-1}{k-1}\right] \\ &\quad + (\gamma)_{(k)}z^{\gamma-k}S \\ &= \sum_{j=1}^{k+1}\binom{k}{j-1}(\gamma)_{(j-1)}z^{\gamma-j+1}(\eta+\mu)^{k-j+1}S, \end{aligned}$$

as required by (4.23). Thus, the lemma follows by induction.

Having established the formula for the k^{th} derivative, one could recover $P[n_b\rho_B|n_a=1,\rho_A=\gamma]$ using the standard result for probability generating functions, since $P[n_b\rho_B=k|n_a=1,\rho_A=\gamma]=G^{(k)}(0)/k!$. But,

$$\begin{aligned} \frac{G^{(k)}(z)}{k!} &= \sum_{j=1}^{k+1}\binom{k}{j-1}(\gamma)_{(j-1)}z^{\gamma-j+1}(\eta+\mu)^{k-j+1}S/k! \\ &= \sum_{i=0}^{k}\binom{k}{i}(\gamma)_{(i)}z^{\gamma-i}(\eta+\mu)^{k-i}S/k!. \end{aligned} \tag{4.25}$$

So,

$$\begin{aligned} \frac{G^{(k)}(0)}{k!} &= \binom{k}{\gamma}(\gamma)_{(\gamma)}0^0(\eta+\mu)^{k-\gamma}S/k! && (4.26) \\ &= \frac{k!\gamma!(\eta+\mu)^{k-\gamma}S}{k!\gamma!(k-\gamma)!} \\ &= \frac{(\eta+\mu)^{k-\gamma}\exp\{-(\eta+\mu)\}}{(k-\gamma)!}. && (4.27) \end{aligned}$$

Note that (4.26) follows since all terms of the sum disappear, except when $i=\gamma$. One recognises this as simply the probability of a draw of $k-\gamma$ from

a Poisson distribution of rate $(\eta+\mu) = (n_b\tau_B - 1.\tau_A) = (n_b\tau_B - 1.\tau_B)$ (under the assumption that $n_a = 1$, and, as such, the connecting edge is of length 0, so that $\tau_A = \tau_B$), which simply says the probability of $k-\gamma$ mutations on the remaining $(n_b - 1)$ edges of total length $\tau_B(n_b - 1)$ is a Poisson random variable, which is what one would expect in this special case.

Although the above method of proof is very satisfying from a technical point of view, demonstrating the power of the method of generating functions, the approach for the general case (n_a arbitrary) is far more involved. Instead, one uses a more intuitive method of proof in the following section, which shows from where each part of (4.14) originates.

4.4.1 Single founder case: general proof

Consider a general connected star tree with $n_a\rho_A = \gamma$ mutations on the subtree. Suppose now that one wishes to calculate $P[n_b\rho_B = k|\tau_A, \tau_B, \gamma]$. The tree under consideration has both mutations and ρ mutations (on the connecting edge). The total mutation count is k, a random quantity (being careful to note here that this count is not simply the number of unique mutations since each ρ mutation contributes n_a to the mutation count). After accounting for the γ mutations on the subtree (which *are* unique mutations), $k-\gamma$ of the mutation count is left to be placed on the connecting edge and/or the additional $(n_b - n_a)$ edges (the 'comb').

Each mutation on the connecting edge contributes n_a to the mutation count, while the total mutation count cannot exceed k. Suppose j 'ρ mutations' occur on the connecting edge. These contribute n_aj to the ρ count. This leaves $k - n_aj - \gamma$ mutations that must have occurred on the comb. The previous intuitive reasoning immediately provides the maximal number of ρ

mutations, which is given by $\left\lfloor \frac{k-\gamma}{n_a} \right\rfloor$. The number of mutations, j, on the connecting edge is simply a Poisson random variable with rate $(\tau_B - \tau_A) = \eta$, while the number of mutations, $k - n_a j - \gamma$, on the comb is simply a Poisson random variable with rate $(n_b - n_a)\tau_B = \mu$.

With the previous work established by intuitive reasoning, the proof of (4.14) can be neatly expressed as follows, avoiding the need to determine by induction a formula for the k^{th} derivative of the relevant probability generating function.

Let j be the number of ρ mutations on the connecting edge, which will be a positive integer, $j = 0, 1, 2, ..., \left\lfloor \frac{k-\gamma}{n_a} \right\rfloor$.

Now,

$$P\left[n_b\rho_B = k|\tau_A, \tau_B, \gamma\right] = \sum_{j=0}^{\left\lfloor \frac{k-\gamma}{n_a} \right\rfloor} P\left(n_b\rho_B = k, j|\tau_A, \tau_B, \gamma\right), \tag{4.28}$$

where $P\left(n_b\rho_B = k, j|\tau_A, \tau_B, \gamma\right)$ is the probability of j mutations on the connecting edge and $k - \gamma - n_a j$ mutations on the comb, i.e.

$$\begin{aligned} P\left(n_b\rho_B = k, j|\tau_A, \tau_B, \gamma\right) &= \frac{e^{-\eta}\eta^j}{j!}\frac{e^{-\mu}\mu^{k-\gamma-n_a j}}{(k-\gamma-n_a j)!} \\ &= \frac{e^{-(\eta+\mu)}\eta^j\mu^{k-\gamma-n_a j}}{j!(k-\gamma-n_a j)!}. \end{aligned}$$

Thus, (4.28) becomes

$$P\left[n_b\rho_B = k|\tau_A, \tau_B, \gamma\right] = \sum_{j=0}^{\left\lfloor \frac{k-\gamma}{n_a} \right\rfloor} \frac{e^{-(\eta+\mu)}\eta^j\mu^{k-\gamma-n_a j}}{j!(k-\gamma-n_a j)!} \tag{4.29}$$

as required.

It is not immediately obvious that this formula is identical to the result derived using probability generating functions for the case when $n_a = 1$. This is shown below.

When $n_a = 1$, the 'connecting edge' has length zero. This has the consequence of restricting the number of mutations on the connecting edge to be zero. All the additional $k-\gamma$ mutations must have occurred on the additional $n_b - 1$ edges which formed the comb. This observation removes the sum in formula (4.29), as all terms disappear except for the $j = 0$ case. Setting $j = 0$ and $n_a = 1$ gives rise to

$$P\left[n_b\rho_B = k|\tau_A, \tau_B, \gamma, n_a = 1\right] = \frac{e^{-(\eta+\mu)}\mu^{k-\gamma}}{(k-\gamma)!}. \tag{4.30}$$

This is equivalent to (4.27) when one realises that $\eta = (\tau_B - \tau_A) = 0$ when the connecting edge is of length zero, and (4.30) can simply be re-expressed to agree with (4.27) as

$$P\left[n_b\rho_B = k|\tau_A, \tau_B, \gamma, n_a = 1\right] = \frac{e^{-(\eta+\mu)}(\mu+\eta)^{k-\gamma}}{(k-\gamma)!}.$$

4.5 MCMC estimation of a single migration time

The parameter of interest here is τ, the time of the migration event, which is assumed to fall somewhere on the connecting edge of a connected star tree. It was previously shown (4.12) that

$$P\left[\tau, \tau_A, \tau_B|\rho_A, \rho_B\right] \propto P\left[\tau|\tau_A, \tau_B\right] P\left[\rho_A, \rho_B|\tau_A, \tau_B\right] P\left[\tau_A, \tau_B\right].$$

Integrating out τ_A and τ_B gives rise to the density of τ, noting carefully the implicit inequality $\tau_A \leq \tau \leq \tau_B$ that gives rise to the integral limits shown

below:

$$\begin{aligned} P \quad [\tau|\rho_A, \rho_B] &\propto \int_{\tau}^{\infty} \int_{0}^{\tau} P\left[\tau, \tau_A, \tau_B|\rho_A, \rho_B\right] d\tau_A d\tau_B \\ &= \int_{\tau}^{\infty} \int_{0}^{\tau} P\left[\tau|\tau_A, \tau_B\right] P\left[\rho_A, \rho_B|\tau_A, \tau_B\right] P\left[\tau_A, \tau_B\right] d\tau_A d\tau_B \\ &= \int_{\tau}^{\infty} \int_{0}^{\tau} P\left[\tau|\tau_A, \tau_B\right] P\left[\rho_A|\tau_A\right] P\left[\rho_B|\tau_A, \tau_B, \rho_A\right] P\left[\tau_A, \tau_B\right] d\tau_A d\tau_B. \end{aligned} \tag{4.31}$$

It is interesting to note that τ only appears above on the limits of the relevant integrals. Using MCMC, one can sample from the joint density of τ_A and τ_B, while τ can be sampled with only a small extension to the MCMC algorithm. This algorithm is implemented in R [42], assuming the prior distribution of $\tau|\tau_A, \tau_B$ is $\mathrm{Un}(\tau_A, \tau_B)$, and that the prior on (τ_A, τ_B) is uniform in a finite region of (τ_A, τ_B) space (such that $0 \leq \tau_A \leq C_1, \;\; 0 \leq \tau_B \leq C_2, \;\; C_1, C_2 \in \mathbb{R}$).

4.6 Examples of estimation of a single migration time by MCMC

In this section pseudocode for the algorithm is described in some detail and examples are given of the code's operation. The method uses the Metropolis-Hastings algorithm ([57], [58]).

1) Set $\tau_A^{(1)} = \rho_A$ and $\tau_B^{(1)} = \rho_B$.

2) Preliminary check that $\tau_A^{(1)} \leq \tau_B^{(1)}$. If this inequality is not satisfied, adjust $\tau_A^{(1)}$ and $\tau_B^{(1)}$ to obtain appropriate starting values which obey the inequality.

3) Create a matrix to store the (τ_A, τ_B, τ) values for each retained iteration.

4) Set a counter to 0 which will store the number of successful moves made by the MCMC algorithm for the (τ_A, τ_B) move proposals.

5) Compute $P[\tau, \tau_A, \tau_B | \rho_A, \rho_B]$ using formula (4.12) (i.e. up to a normalising constant) with $\tau_A = \tau_A^{(1)}, \tau_B = \tau_B^{(1)}$.

6) Enter loop (set $i = 1$). Loop through (7)-(12) until enough burn-in and real draws have accumulated.

7) Propose new (τ_A, τ_B) combination using $\tau_A^{(i+1)} = \tau_A^{(i)} + N(0, \sigma_1^2)$, and $\tau_B^{(i+1)} = \tau_B^{(i)} + N(0, \sigma_2^2)$, where σ_1^2 and σ_2^2 are set by the user.

8) Check that the new $\tau_A^{(i+1)}, \tau_B^{(i+1)}$ proposals obey the necessary constraints $\tau_A^{(i+1)} \leq \tau_B^{(i+1)}, \tau_A^{(i+1)} \geq 0$. If not, reject proposals and set $\tau_A^{(i+1)} = \tau_A^{(i)}$ and $\tau_B^{(i+1)} = \tau_B^{(i)}$.

9) If necessary constraints are satisfied, calculate $P[\tau, \tau_A, \tau_B | \rho_A, \rho_B]$ using formula (4.12) with $\tau_A = \tau_A^{(i+1)}, \tau_B = \tau_B^{(i+1)}$.

10) Form an acceptance ratio $\frac{P\left[\tau,\tau_A^{(i+1)},\tau_B^{(i+1)}|\rho_A,\rho_B\right]}{P\left[\tau,\tau_A^{(i)},\tau_B^{(i)}|\rho_A,\rho_B\right]}$.

11) Accept move with probability $\min\left(1, \frac{P\left[\tau,\tau_A^{(i+1)},\tau_B^{(i+1)}|\rho_A,\rho_B\right]}{P\left[\tau,\tau_A^{(i)},\tau_B^{(i)}|\rho_A,\rho_B\right]}\right)$.

12) Store the final values of τ_A and τ_B for iteration $i+1$ and update the move counter if the proposal was accepted at this iteration and the burn-in period has passed. Draw $\tau^{(i+1)}$ uniformly between $\tau_A^{(i+1)}$ and $\tau_B^{(i+1)}$. If the move was accepted, store the value of $P\left[\tau,\tau_A^{(i+1)},\tau_B^{(i+1)}|\rho_A,\rho_B\right]$ for use at the next iteration. Otherwise, store the previous value.

13) Calculate the acceptance rate for (τ_A,τ_B) by dividing the move counter by the number of iterations.

Two simple simulations are shown below to demonstrate the previous algorithm's usefulness and to demonstrate code correctness. In case 1, $n_a = 10, n_b = 20, \tau_A = 2, \tau_B = 10$, and suppose the number of mutations on each edge is set to its expected value. This situation would give rise to $\rho_A = 2$ and $\rho_B = 10$, with 8 mutations falling on the connecting edge. Using the approach detailed previously, one can simulate the joint (posterior) distribution of τ_A and τ_B, while, adding the assumption that τ lies uniformly between τ_A and τ_B, one can investigate the distribution of τ. From the output shown, figure 4.4, one can see that the distribution obtained is centred around the correct values. This procedure involved $105,000$ iterations, starting values of $\tau_A = 5, \tau_B = 15$, with a burn-in of $5,000$, and $\sigma_1 = \sigma_2 = 0.2$.

In cases where the connecting edge is short the method performs better (in terms of the posterior distribution of τ), giving a posterior distribution which is peaked around a small range of values (figure 4.5) for the case $n_a = 10, n_b = 20, \tau_A = 2, \tau_B = 2.05$, where the number of mutations on each

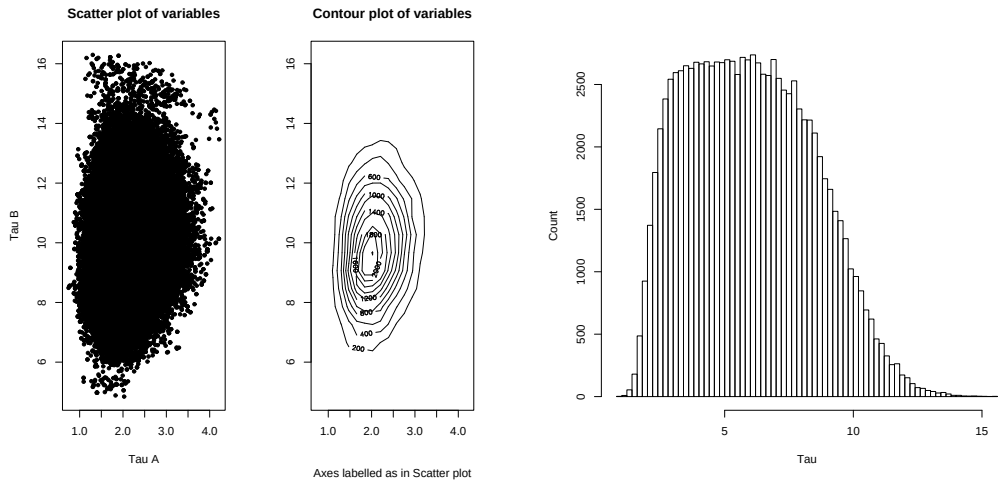

Figure 4.4: Long connecting edge case. Posterior distribution of (τ_A, τ_B) illustrated by $100,000$ draws (left) and a contour plot (middle), together with the (unnormalised) posterior distribution of τ (right).

edge is set to its (rounded) expected value. This situation would give rise to $\rho_A = 2$ and $\rho_B \approx 2$, with no mutations falling on the connecting edge. This procedure involved $105,000$ iterations (after thinning), starting values of $\tau_A = 5, \tau_B = 5$, with a burn in of $5,000$. In this case, $\sigma_1 = \sigma_2 = 0.1$, and some thinning was done since the (τ_A, τ_B) region was smaller than the previous case, with every fifth draw being retained.

Figures 4.4 and 4.5 demonstrate that the procedure is giving posterior densities which look as one would expect under the hypothesised connected star trees with the number of mutations on each edge set at its expected value under an infinite sites model. In the next section the problem of combining the set of all estimated migration times across many founder clusters is addressed, and a model is proposed which not only allows Bayesian estimation of the time of specific founding sequences, but also potentially allows Bayesian estimation of the dates of the main periods of migration, while ad-

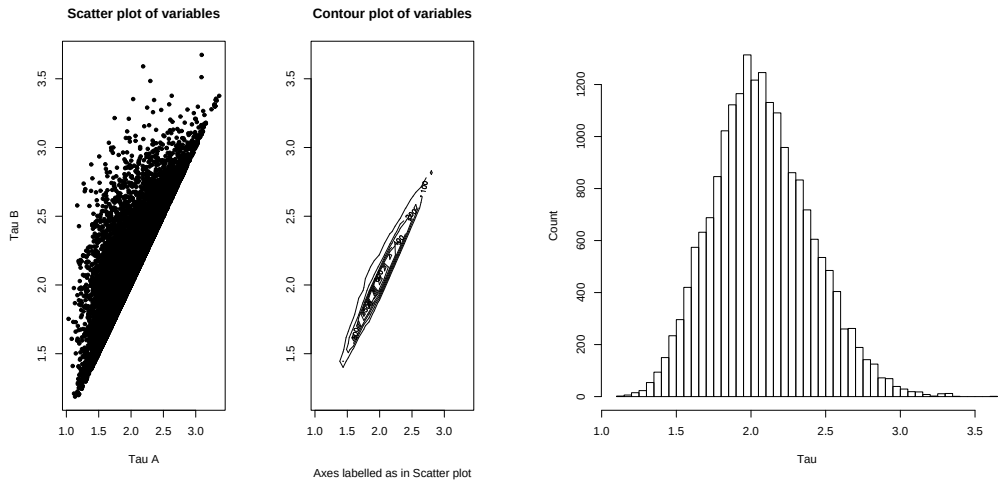

Figure 4.5: Short connecting edge case. Posterior distribution of (τ_A, τ_B) illustrated by $100,000$ draws (left) and a contour plot (middle), together with the (unnormalised) posterior distribution of τ (right).

ditionally providing an objective way of estimating the probability that a given founder sequence belongs to any one of the specific migration periods.

4.7 Estimating migration periods with a mixture model

The previous section detailed Bayesian inference of the migration time of a *single* migration/founding event. In this section, a hierarchical framework is used to combine the information from *every* migration/founding event, to attempt to estimate quantities that relate to the migration history of an entire sample of sequences and the derived founder sequences.

While it would be possible at this point to go straight to formally defining a full Bayesian model, defining prior distributions and such, one feels the need to justify the reasoning behind the model choice. An attempt to give the reader some context-specific explanations of what the parameters of the Bayesian model represents is first provided. To this end, in this section, the model is first described at an intuitive level, introducing the parameters in a way which is intended to offer the reader some insight into the reasoning behind the model choice, and simultaneously giving an explanation of what types of parameters the model ideally should be able to estimate, in the context in which the model will be implemented. Once this introduction is complete, the model is defined in a formal statistical manner with only brief comments about what the parameters represent. The penultimate part of this section discusses the model parameters and what they represent in the context-specific case of interest, in some detail. The final part of this section then demonstrates the model's operation on some simulated data, with attempts to display both the positive features of the model, and the problems which arise in some important cases.

4.7.1 Modelling the migration history of a sample

It is assumed in what follows that a sample of sequences has a migration history which is composed of a number of major migration periods, during which multiple founding events would occur. The correctness of this assumption is, of course, open to question. However, in what follows, it is the assumption that is being made. Under the given assumption, natural quantities of interest arise that one may wish to estimate. The first is simply the times and durations of these major migration periods. In the human context, the dating of the migration periods may be linked to periods of pre-history such as the Neolithic and (Upper, Middle and Lower) Palaeolithic. One may wish to ask at what time did migration periods occur, and how long did these migration periods last? The times and duration of major migration periods are thus natural parameters that should be represented and estimated in any sensible model.

One may also be interested in the relative differences in the number of migration events that occurred in each of the migration periods. This is perhaps not a immediately natural parameter to wish to estimate, but with some thought one can see that a question such as 'Is there evidence from our present day sample that some migration period occurring during the Neolithic involved a larger number of migration/founder events than some migration period during the Upper Paleolithic?' could be of interest. These types of questions are indeed more difficult to answer for a variety of reasons (namely that population sizes are likely to be different in both periods, so quantities such as the number or fraction of migration/founder events are difficult to define clearly, while further complications arise because we know that migration periods occurring further back in time are likely to have fewer associated/inferred

founder sequences than those closer to the present). Regardless, it is desirable that a model of the migration history should provide at least some basis to attempt to answer such questions.

A more fundamental question is simply how many major migration periods can be inferred from a given sample. This is again a non-trivial problem as such an inference will be affected by many factors, some of which will be shown in what follows to be non-statistical and independent of model choice, prohibiting such inferences from being made.

Finally, while a global model for inference of the migration process is the primary aim here, one may in fact be more interested in estimating *exactly which* migration period a *specific* migrant/founder sequence belongs to. So, given an inferred founder sequence, one may wish to estimate the probability that the founder sequence originated from each of the specific migration periods. This is of course, conceptually, another extremely difficult question to address; answering such a question relies on the founder sequence of interest being assigned in some systematic manner to a specific migration period. Assigning founder sequences to migration periods requires the migration periods to have been defined, but, as stated earlier, the times, lengths and even number of migration periods may not be specified from the outset and are in fact items one wishes to estimate.

In summary, one can build up a picture of what parameters a useful model should have, and what questions any such model should allow to be investigated. Furthermore, from previous sections, a method to infer the migration time of a single founder event has been described. In what follows, a hierarchical Bayesian mixture model is described which performs simultaneous estimation of the individual founder times as described in the previous sec-

tion, and the set of founder times is then used within a mixture model to estimate probability densities which represent the quantities described above.

4.7.2 Model specification

The mixture model is parameterised following Roberts' notation [59, page 319], with only some minor changes (which follow Gelman et al. [60]) to clear up some ambiguity. The migration time of founder j ($j = 1, \ldots, J$) shall be denoted by τ_j in what follows, and we shall initially assume the τ_j's are known, i.e. data. The joint distribution of the data given the parameters θ is taken as

$$\begin{aligned} p(\tau|\theta) &= \prod_{j=1}^{J} p(\tau_j|\theta) \\ &= \prod_{j=1}^{J} \sum_{i=1}^{k} p_i \varphi(\tau_j; \mu_i, \sigma_i^2), \quad \sum_i p_i = 1,\ 0 \leq p_i \leq 1\ \forall i. \end{aligned} \tag{4.32}$$

Equation (4.32) represents the founders as coming (independently) from a mixture distribution [61], with the migration periods represented by the distributions $\varphi(\tau; \mu_i, \sigma_i^2), i = 1, \ldots, k$ (assumed normal distributions with means μ_i and variances σ_i^2), with the fractions p_i representing the *a priori* probability that an arbitrary founder sequence originates from migration period i. In what follows, the collection of means and variances $(\mu_i, \sigma_i^2, i = 1, \ldots, k)$ of every component will be denoted by θ for notational convenience.

The above specification immediately provides parameters which represent some of the primary quantities of interest described previously. One views the normal distributions as representing each of the migration periods, and the parameters of the distributions represent estimates of both the time and spread of the migration periods. Furthermore, the p_i can be thought of as an

approximate measure of the proportion of the founder sequences that belong to each of the migration periods. It is noted here however that it is difficult to assign a precise interpretation to these p_i parameters for reasons that will be explained later.

One of the main benefits of the mixture-model framework is the fact that each data point's component membership can be represented by an indicator vector, a 'missing variable' which can be estimated. One does not know which migration period each founder belongs to. Within the mixture model framework an indicator variable (a vector) is assigned to each data point (founder), and this provides a means to estimate the probability that a given founder sequence originated from a specific migration period.

Define the indicator vector for founder j to be z_j, with its ith element,

$$z_{ij} = \begin{cases} 1 & \text{if } \tau_j \sim \varphi(\tau_j; \mu_i, \sigma_i^2), \\ 0, & \text{otherwise.} \end{cases} \tag{4.33}$$

That is to say, each founder has a k-element indicator vector with element i ($i = 1, \ldots, k$) being 1 if founder j is assigned to component (migration period) i, with all other elements of the indicator vector being 0.

At this point, the parameters of interest $(\mu_i, \sigma_i^2, p_i), i = 1, \ldots, k$, and the assignment indicator variables $z_j, j = 1, \ldots, J$ have been described. One now assumes that given the component mean and variances, θ, the founder assignment indicator vectors are a draw of size 1 from a multinomial distribution with parameters p_i. Further, given that founder j belongs to component i (i.e. given the indicator vector z_j), and given the component means and variances, θ, one assumes that the founder migration time τ_j comes from a normal distribution with mean μ_i and variance σ_i^2, as shown below:

$$z_j \sim \text{Mult}(1; p_1, \ldots, p_k), \tag{4.34}$$

$$\tau_j|z_j,\theta \sim N\left(\prod_{i=1}^{k}\mu_i^{z_{ij}},\prod_{i=1}^{k}\sigma_i^{2z_{ij}}\right). \tag{4.35}$$

A hierarchical model is gradually being built up here. Under a connected star-tree assumption, one can estimate a (τ_A,τ_B) pair for each of the J founders. Using the result in the previous section, a Bayesian estimate of the actual migration time, τ, can be made (under reasonable prior assumptions) for each of the J founders. One is now adding on top of this the assumption that the set of actual migration times arise from a mixture model. The assignment of founders to a specific component is done using indicator variables, which are determined by a multinomial probability model depending on parameters p_i. Once the assignments of founders to the k components has been made, the migration event times are assumed to follow normal distributions with means and variances $(\mu_i,\sigma_i^2), i=1,\ldots,k$. Now that the parameters of interest have been described and the mixture model framework introduced, appropriate prior distributions for the parameters are given.

It is hoped that the previous details have convinced the reader that the model specification and framework as described are indeed suitable for the problem at hand, and not merely an *artificial* parameterisation that involves parameters that do not represent quantities of real interest to anyone investigating migration processes (with the assumption that the migration process did indeed involve distinct periods of migration).

Denote by $\pi(.)$ a prior distribution to be defined by the investigator. The natural prior distributions are the conjugate priors:

$$\pi_i(\mu_i,\sigma_i^2) = \pi_i(\mu_i|\sigma_i^2)\pi_i(\sigma_i^2), \tag{4.36}$$

$$\mu_i|\sigma_i^2 \sim N(\xi_i,\sigma_i^2/\kappa_i), \tag{4.37}$$

$$\sigma_i^2 \sim IG(\nu_i/2, S_i^2/2), \tag{4.38}$$

$$p \sim Dir(\alpha_1, \ldots, \alpha_k). \tag{4.39}$$

Note that the prior for the component means *depends on the component variance*. This is discussed by Gelman et al. who state that "it often makes sense for the prior variance of the mean to be tied to σ^2, which is the sampling variance of the observation y [τ in this work]". The conjugate prior for the p_i's is the Dirichlet distribution. IG is the inverse gamma distribution.

Specifying the hyperparameters is an additional complication that the investigator must undertake. It is usual in Bayesian statistical applications to ensure that prior specification is suitably vague to allow the data to be the primary factor in determining the posterior densities of the parameters of interest. The model however does provide an opportunity for informative priors to be selected in the event that the investigator has strong reason to believe in his/her prior beliefs in the migration history of a sample. The hyperparameters ξ represent the prior component means of the normal distributions (the prior mean times of the migration periods of interest). The κ parameters can be viewed to represent the strength of one's belief in the prior component mean values, noting that, as κ_i increases, the prior component mean density becomes tightly peaked around ξ_i.

4.7.3 Full conditional distributions of (μ, σ^2)

In this section, the full conditional distributions of the parameters are derived. These distributions allow one to make draws from each variable in turn, conditional on the other variables which will be known at every stage of the algorithm. The existence of such conditional distributions allows one to use Gibbs sampling to investigate the parameters of interest. The posterior densities of the parameters of the normal distributions are calculated by first deriving an expression for the product of the prior density and the likelihood, as shown below. For simplicity in notation the derivation is done for the single normal case. This is acceptable since, in the finite mixture model case, once the allocation vectors are assigned and the data points belong to a single component, the mixture model essentially simplifies to estimating the parameters of k independent normal distributions.

The prior density is

$$\begin{aligned}
\pi(\mu, \sigma^2) &= \pi(\mu|\sigma^2)\pi(\sigma^2) \\
&= \frac{1}{\sqrt{2\pi(\sigma^2/\kappa)}} \exp\left\{-\frac{1}{2}\frac{(\mu-\xi)^2}{(\sigma^2/\kappa)}\right\} \\
&\quad \times \frac{[S^2/2]^{[\nu/2]}}{\Gamma(\nu/2)} (\sigma^2)^{-[\nu/2+1]} \exp\left\{-\left[S^2/2\sigma^2\right]\right\} \\
&\propto (\sigma^{-1})(\sigma^2)^{-[\nu/2+1]} \exp\left\{-\frac{1}{2\sigma^2}\left[S^2 + (\mu-\xi)^2\kappa\right]\right\}.
\end{aligned}$$

The likelihood is a product of normals (e.g. see [62]):

$$\begin{aligned}
L(\mu, \sigma^2) &= P(\tau|\mu, \sigma^2) \\
&= \prod_{j=1}^{J} \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left\{-\frac{(\tau_j-\mu)^2}{2\sigma^2}\right\} \\
&\propto (\sigma^2)^{-J/2} \exp\left\{-\frac{1}{2\sigma^2}\left[J(\mu-\bar{\tau})^2 + \sum_j (\tau_j-\bar{\tau})^2\right]\right\}.
\end{aligned}$$

Multiplying the prior density by the likelihood yields the posterior density (up to a normalising constant):

$$\begin{aligned} p(\mu,\sigma^2|\tau) &\propto (\sigma^{-1})(\sigma^2)^{-(\nu+J+2)/2} \\ &\quad \times \exp\left\{-\frac{1}{2\sigma^2}\left[S^2+(\mu-\xi)^2\kappa+\sum_j(\tau_j-\bar{\tau})^2+J(\bar{\tau}-\mu)^2\right]\right\} \end{aligned} \tag{4.40}$$

Considering the terms in the square bracket of the argument of the exponential in isolation, (4.40) can be re-expressed (after some tedious algebra) as

$$\begin{aligned} p(\mu,\sigma^2|\tau) &\propto (\sigma^{-1})(\sigma^2)^{-[(\nu+J)/2+1]} \\ &\quad \times \exp\left\{-\frac{1}{2\sigma^2}\left[S^2+\sum_j(\tau_j-\bar{\tau})^2+(J+\kappa)\left[\mu-\frac{(J\bar{\tau}+\xi\kappa)}{(J+\kappa)}\right]^2\right.\right. \\ &\quad \left.\left.+\frac{J\kappa}{(J+\kappa)}(\xi-\bar{\tau})^2\right]\right\}. \end{aligned} \tag{4.41}$$

The full conditional distribution of μ (given σ^2 and τ) is proportional to the above:

$$p(\mu|\sigma^2,\tau) \propto \exp\left\{-\frac{(J+\kappa)}{2\sigma^2}\left[\mu-\frac{(J\bar{\tau}+\kappa\xi)}{(J+\kappa)}\right]^2\right\} \tag{4.42}$$

One recognises the above as a quantity proportional to a normal density $\mathrm{N}(\alpha,\beta)$ with parameters

$$\begin{aligned} \alpha &= \frac{(J\bar{\tau}+\kappa\xi)}{(J+\kappa)}, \\ \beta &= \frac{\sigma^2}{\kappa+J}. \end{aligned}$$

Similarly, the full conditional of σ^2 (given μ and τ), is seen to be an inverse gamma, $IG(\gamma,\delta)$, with parameters

$$\gamma = \frac{\nu+J}{2}, \tag{4.43}$$

$$\delta = \frac{1}{2}\left[S^2+\sum_j(\tau_j-\bar{\tau})^2+\frac{J\kappa}{(J+\kappa)}(\xi-\bar{\tau})^2\right]. \tag{4.44}$$

4.7.4 Full conditional distributions of p_i, z_j, τ_j, (τ_A^j, τ_B^j)

With a Dirichlet prior on the p_i, $p \sim Dir(\alpha_1, \ldots, \alpha_k)$, and noting that the p_i are connected to the other parameters in the model only via the hyperparameters (the α's) and the allocation vectors (the z_j), due to the hierarchical nature of the mixture model, one can derive the full conditional of the p_i. In the derivation that follows, $\{p_i\}$ denotes the set of mixing fractions, $\{\alpha_i\}$ denotes the set of α hyperparameters, and $\{z\}$ denotes the complete set of indicator vectors, while $m_i(z) = \sum_{j=1}^{J} z_{ij}$ denotes the number of data points currently assigned to component i. Then,

$$\begin{aligned} p(\{p_i\} \,|\, \{\alpha_i\}, \{z\}) &\propto p(\{p_i\} \,|\, \{\alpha_i\}) p(\{z\} \,|\, \{p_i\}) && (4.45) \\ &\propto \prod_{i=1}^{k} p_i^{\alpha_i - 1} p_i^{m_i(z)} && (4.46) \\ \text{i.e. } \{p_i\} \,|\, \{\alpha_i\}, \{z\} &\sim Dir(\{\alpha_i + m_i(z)\}). && (4.47) \end{aligned}$$

The full conditional distribution of the allocation vectors, the z_j, is calculated as follows. Again, one uses the hierarchical nature of the mixture model, which is helpful since the allocation vector for founder j depends only on the founder migration times (the current τ value for founder j, τ_j), together with the p_i's:

$$\begin{aligned} p(z_{ij} = 1 | \tau_j, \{p_i\}, \theta) &= \frac{p(z_{ij} = 1 | \{p_i\}) p(\tau_j | z_{ij} = 1)}{\sum_{l=1}^{k} p(z_{lj} = 1 | \{p_l\}) p(\tau_j | z_{lj} = 1)} && (4.48) \\ &= \frac{p_i \varphi(\tau_j; \theta_i)}{\sum_{l=1}^{k} p_l \varphi(\tau_j; \theta_l)}, && (4.49) \\ \text{i.e. } z_j | \tau_j, \{p_i\}, \theta &\sim \text{Mult}\left(1; \left\{ \frac{p_i \varphi(\tau_j; \theta_i)}{\sum_{l=1}^{k} p_l \varphi(\tau_j; \theta_l)} \right\}\right), && (4.50) \end{aligned}$$

where $\varphi(\tau_j; \theta_i)$ is the value of the pdf of a normal density with mean μ_i and variance σ_i^2, evaluated at τ_j.

The full conditional distribution of τ_j, given the allocations (the z_j), the parameters of the normal distributions for each component (the θ_i's), together with the (τ_A, τ_B) for founder j (where, for clarity, the founder index now appears as a superscript) is seen to be simply a truncated normal distribution since, once the allocation for a given founder is known and the parameters of the normal distribution to which it belongs are determined, the distribution of $\tau_j|z_j, \theta, (\tau_A^j, \tau_B^j)$ is normal with mean μ_i and variance σ_i^2, subject to the additional constraint that $\tau_A^j \leq \tau_j \leq \tau_B^j$, which is a truncated normal, i.e.

$$\begin{aligned} p(\tau_j|z_j, \theta, (\tau_A^j, \tau_B^j)) &= \frac{p(\tau_A^j, \tau_B^j|\tau^j)p(\tau^j|z_j, \theta)}{\int_{\tau_j} p(\tau_A^j, \tau_B^j|\tau^j)p(\tau^j|z_j, \theta)d\tau_j}, \\ p(\tau_j|z_j, \theta, (\tau_A^j, \tau_B^j)) &= \frac{I(\tau_A^j \leq \tau_j \leq \tau_B^j)\varphi(\tau_j; \theta_i)}{\int_{\tau_A^j}^{\tau_B^j} \varphi(\tau_j; \theta_i)d\tau_j}, \end{aligned} \tag{4.51}$$

where I denotes the indicator function. The denominator of (4.51) follows since $p(\tau_A^j, \tau_B^j|\tau^j)$ is a constant in the range space.

Finally, given τ_j and (ρ_A^j, ρ_B^j) for each founder, the full conditional of (τ_A^j, τ_B^j) is calculated in a similar manner:

$$p(\tau_A^j, \tau_B^j|\tau_j, \rho_A^j, \rho_B^j) = \frac{p(\tau_A^j, \tau_B^j|\tau_j)p(\rho_A^j, \rho_B^j|\tau_A^j, \tau_B^j)}{\int\int p(\tau_A^j, \tau_B^j|\tau_j)p(\rho_A^j, \rho_B^j|\tau_A^j, \tau_B^j)d\tau_A^j d\tau_B^j}. \tag{4.52}$$

Under the assumption that $\tau_A^j, \tau_B^j|\tau_j$ is uniform in $\tau_A^j \leq \tau \leq \tau_B^j$, one realises that this is simply a truncated form of the distribution previously determined when considering only a single founding event (implicitly assuming that it migrated during the *only possible* migration period, i.e. a mixture model with a single component).

At this point the full conditionals for every parameter in the model have been explicitly evaluated, and an appropriate Bayesian procedure (Gibbs sampling) can be used to create samples from the posterior distributions of

the parameters, given appropriate prior choices and data (data here meaning ρ_A, ρ_B estimates for a set of J founder sequences). Before doing this however, I shall summarise the statistical features of the mixture model and demonstrate proof of concept at the mixture model level (and code correctness at the computational level) in a similar manner to what was done when considering estimating the migration time for a single founding event.

4.7.5 Mixture model summary

Assume for the moment that the data are actually the set of actual migration event times for each founder, and not the set of (ρ_A, ρ_B) values for every migration event. With this assumption, which is made purely for model testing purposes here, one has removed the additional uncertainty introduced through the (τ_A, τ_B) estimation process, and reduced the hierarchical structure of the model down to a more standard finite mixture model. Further, assume that the number of migration periods is fixed and known.

The prior distributions and the resulting posteriors which were derived previously for the mixture model level of the complete hierarchical model are summarised below.

Priors:

$$\begin{aligned} \pi_i(\mu_i, \sigma_i^2) &= \pi_i(\mu_i|\sigma_i^2)\pi_i(\sigma_i^2), \\ \mu_i|\sigma_i^2 &\sim N(\xi_i, \sigma_i^2/\kappa_i), \\ \sigma_i^2 &\sim IG(\nu_i/2, S_i^2/2), \\ p &\sim Dir(\alpha_1, \ldots, \alpha_k). \end{aligned}$$

The resulting full conditionals were shown to be

$$\mu_i|\tau, z, \sigma_i \sim N(\xi_i(\tau, z), \sigma_i^2/(\kappa_i + m_i(z))), \tag{4.53}$$

$$\sigma_i^2|\tau, z \sim IG\left(\frac{\nu_i + m_i(z)}{2}, \frac{1}{2}\left[S_i^2 + \hat{S}_i^2(\tau, z) + \frac{\kappa_i m_i(z)}{\kappa_i + m_i(z)}(\bar{\tau}_i(z) - \xi_i)^2\right]\right), \tag{4.54}$$

$$p|\tau, z \sim Dir\left(\alpha_1 + m_1(z), \ldots, \alpha_k + m_k(z)\right), \tag{4.55}$$

where

$$\xi_i(\tau, z) = \frac{\kappa_i \xi_i + m_i(z)\bar{\tau}_i(z)}{\kappa_i + m_i(z)},$$

$$\begin{aligned} m_i(z) &= \sum_{j=1}^{J} z_{ij}, \\ \bar{\tau}_i(z) &= \frac{1}{m_i(z)} \sum_{j=1}^{J} z_{ij}\tau_j, \\ \hat{S}_i^2(\tau, z) &= \sum_{j=1}^{J} z_{ij}(\tau_j - \bar{\tau}_i(z))^2. \end{aligned}$$

The four functions above can be recognised (respectively) as a weighted mean of prior and actual migration times in component i, the number of founders assigned (currently) to component i, the mean migration time in component i and the sum of squares of deviations of migration times from their relevant component means.

The posterior distribution of the allocation vectors is

$$z_j|\tau_j, \theta \sim \text{Mult}(1; p_1(\tau_j, \theta), \ldots, p_k(\tau_j, \theta)), \tag{4.56}$$

where

$$p_i(\tau_j, \theta) = \frac{p_i \varphi(\tau_j; \mu_i, \sigma_i)}{\sum_{l=1}^{k} p_l \varphi(\tau_j; \mu_l, \sigma_l)}.$$

An MCMC sampler to produce draws from the posterior distributions in the model is now described, assuming appropriate data is provided and suitable priors selected. Note here that, within the finite mixture model level of the hierarchical model, all of the parameter updates will be Gibbs updates, i.e. they are draws from a full conditional probability distribution, and *not* a move which depends on an acceptance ratio. This is in contrast to the (τ_A, τ_B) update step seen earlier which was a Metropolis-Hastings move and which, even in the presence of thinning, could result in the parameter updates remaining constant over short periods of the chain. One could argue that

Gibbs updates are more attractive, since the parameters should always be updating. However, small movements from Gibbs updates can actually result in poor mixing which could be harder to detect than in cases when moves were based on acceptance ratios. In such cases bad mixing is easier to determine as it is clear to the investigator that mixing is taking place at a very slow rate. A further disadvantage of Gibbs moves is that the moves cannot be tweaked by altering proposal distributions.

4.7.6 A note about identifiability

Up until this point, the issue of identifiability has been ignored. Finite mixture models suffer from identifiability issues in general due to the fact that the components are exchangeable unless some additional constraints are imposed on them. To see this problem, consider a two component mixture model with the true component mean values being equal to 5 and 10 (in some appropriate units). In the absence of any further information, one can see that it should make no difference whether the component with the smaller mean is labelled as the first or second component and a permutation of the labels should not affect any posterior densities of interest if an appropriate algorithm was devised and ran to convergence.

To avoid this label-switching problem, an ordering is imposed on the component means, and it is assumed that the component designated as the *first* component is that component with the *smallest* mean, that is, $\mu_1 \leq \mu_2 \leq \ldots \leq \mu_k$.

4.7.7 Pseudocode for the mixture model

Recall for the moment that the data here is still for the time being assumed to be the τ values. The following describes the process of the Bayesian estimation of the parameters of the mixture model.

1) Choose hyperparameters appropriately so that $\xi, \kappa, \alpha, \nu, S^2$ and number of components are defined. An additional parameter is required, μ_{MAX}, which represents some maximal value that the largest component mean cannot exceed. This 'parameter' is required only for computational/coding purposes and is chosen to be large enough so that no component mean will ever be even of the same order of magnitude as it. Essentially, this is a way of making the prior on the means proper (normalisable).

2) Generate sensible starting values for the parameters. It is worth noting here that the starting parameter values should not affect the posterior densities, and, regardless of the starting location, the same posterior densities should be obtained subject to satisfactory mixing with some burn-in. Let the index used to denote the iteration one is at be u, initialised at $u = 1$. The initial component mean vector, $\underline{\mu}(1)$ is set to equal the prior component mean vector, $\underline{\xi}$. The component variance vector, $\underline{\sigma}^2(1)$, is set to equal the expected value of the prior distributions on the variances i.e. $\sigma_i^2(1) = S_i^2/(\nu_i - 2)$. The $\underline{p}(1)$ vector is initialised so that the starting prior probability that a founder belongs to each of the k components is $1/k$. The starting values of the allocation matrix $z(1)$ are obtained by assigning each data point to the component that its τ value is closest to in terms of absolute value (with ties broken at random).

3) Create storage variables to store the values of $\underline{\mu}, \underline{\sigma}^2, \underline{p}$ at every iteration, as well as the number of data points assigned to each component at each

iteration. Storing the allocation matrix, z, in its entirety at every iteration is not done due to its size (a model with 4 components, containing even just 100 founders, with $50,000$ iterations after burn-in, would require an array with dimensions $(100, 4, 50000)$ for complete storage). Instead, create one J by k matrix which will store the *sum* of the z matrix across all iterations, so that row j of the matrix represents a vector that displays the number of times that founder j was assigned to each of the components.

4) Loop through the following until 'burn.in (B) + number.draws (I)' is reached.

5) Update the mixing fractions. Draw $p(u+1)$, using (4.55). If $(u+1) > B$, store p.

6) Update the z matrix (allocation of founders). Draw $z(u+1)$ using (4.56). If $(u+1) > B$, add $z(u+1)$ to the cumulative z matrix.

7) Update the component means. Draw $\mu(u+1)$ using equation (4.53). If $(u+1) > B$, store μ.

8) Update the component variances. Draw $\sigma^2(u+1)$ using (4.54). If $(u+1) > B$, store σ^2.

9) Increment u

10) Restart loop provided $u < B + I$.

11) Return storage objects.

4.7.8 Some examples

In this section, the algorithm is demonstrated in some constructed cases where data (the τ's still) are simulated and known, and the priors are selected to be appropriate for the simulation. One notes here that the scale of the parameters is set to loosely match what will be used in real data analysis, namely that time will be measured in units where 1 unit corresponds to approximately $20,000$ years, matching the mutation rate of the segment of mitochondrial DNA to be analysed later in this thesis. This time scaling knowledge aids hyperparameter selection since one can be confident that the migration periods of interest (in the case of humans) all would have occurred within the last $100,000$ years, or certainly within the last $200,000$ years. This allows the scale of time measurement to be safely constrained within $(0, 10)$, corresponding to $(0, 200,000)$ years before present (YBP).

The first example considered shows the method's performance for a case with two hypothetical components which have some degree of overlap in their tails. Namely, one imagines two migration periods, corresponding to two normal distributions with means $(0.45, 1.3)$. The components are assumed to have standard deviations of 0.2, and, from this, 100 data points are simulated from both distributions. Figure 4.6 shows the two distributions which were used for this simulation.

It should be noted here, before considering the model's performance, that this case, although very artificial, does demonstrate what could be considered a problem with using a mixture model to estimate the dates of migration periods. One can conceptually imagine a data point being simulated from the normal distribution centred at 0.45 in figure 4.6, and being found to have come from the right tail of the distribution, for instance a value of 1.1. The

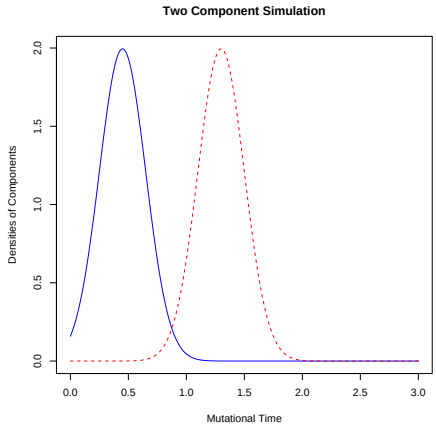

Figure 4.6: Plot showing the theoretical normal distributions (components) used to generate the data. Two components are envisaged, with means at 0.45 and 1.3 ρ units, both with standard deviation 0.2.

mixture model set-up makes use of the allocation vectors (the z) to assign each data point to a component at every stage of the process with a Gibbs update. One could argue here that it makes little sense that this hypothetical data point could be (*correctly*) assigned to the first component with mean 0.45, yet its actual value is close to the mean of the second component (1.3). This is indeed a troublesome issue conceptually. If the model was correctly identifying this hypothetical point to belong to the first component consistently, but also correctly identifying the component means, one is in the unsatisfactory situation where the investigator would be forced to report a migration event consistently associated with a migration period despite the fact that its actual migration time (here assumed known, as it is simulated, but would be otherwise inferred) suggests that it belongs to a different migration period.

The problem described above though is not one that is consistent with the

assumed migration process which generates the data. The migration process is assumed to have occurred in short bursts, with each migration period possibly even involving a large number of migration events over a relatively small time scale. The assumptions about the migration process mean that such a 'problem' that may be seen in a real data case is merely a consequence of the data being relatively uninformative. It is important however to realise that the model as discussed so far may in fact display such unfortunate cases when a data point (inferred migration time) is assigned to a component even although its migration time suggests such a component membership to be unlikely. It is perhaps best to consider such problem cases as merely indicating that the data point in question belongs to a founder sequence that cannot be *reliably* assigned to any one specific component (such problem cases generally will lie in the tails between consecutive components).

4.7.9 Simulation 1

The data as simulated is graphed in figure 4.7.

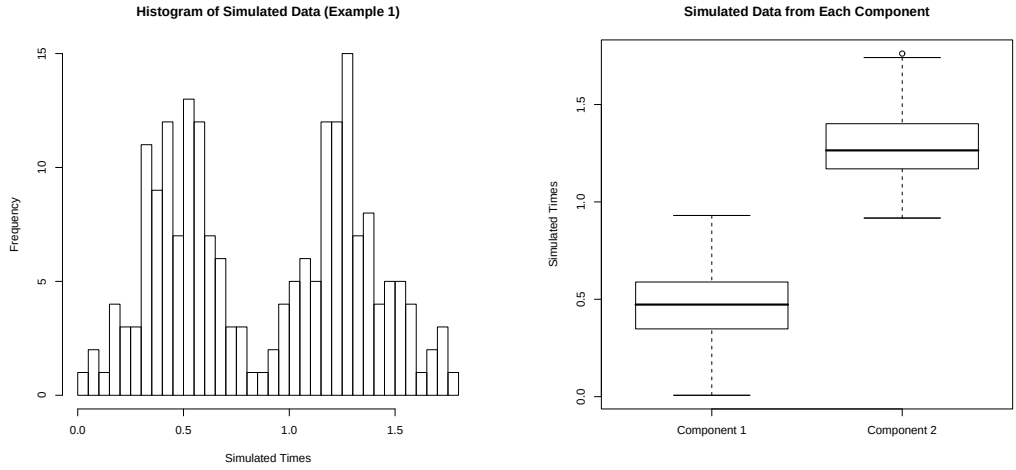

Figure 4.7: Histogram (left) and boxplot (right) of the 200 simulated data points, 100 from each component. The standard deviations of both components was set to 0.2.

With the data simulated, appropriate priors are chosen to match the data well. In reality of course, this luxury is one which the investigator does not have, but is of interest in this case to see the posterior densities under such a case. To this end, priors are set as follows: $\xi = (0.45, 1.3)$, $\nu = (4, 4)$, $S^2 = (0.1, 0.1)$, $\kappa = (1, 1)$, $\alpha = (1, 1)$.

The choice of ν and S^2 gives rise to an identical prior density for both σ_1^2 and σ_2^2 (figure 4.8), which has an expected value equal to 0.05, which corresponds to a prior standard deviation of ≈ 0.22, and *infinite* variance.

One now looks at the posterior densities that are obtained. These examples involved retaining $5,000$ iterations after first discarding $2,000$ iterations for

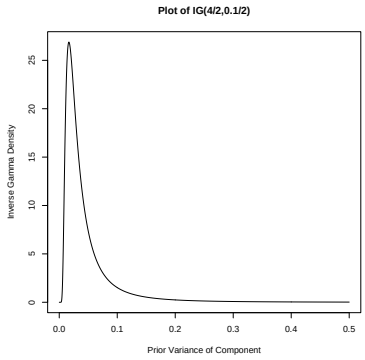

Figure 4.8: Plot showing the prior distribution of σ^2.

burn-in, with thinning so that every fifth draw was retained (resulting in a total of $25,000$ iterations being undertaken after burn-in).

Figure 4.9 shows that over the course of the inference process, on average, the correct number of data points are assigned to each component.

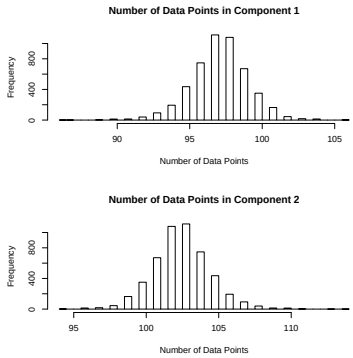

Figure 4.9: Posterior distribution of number of data points in each component.

Figure 4.10 shows that the posterior means of the components are *slightly* shifted from their true values. This can be explained by considering the sys-

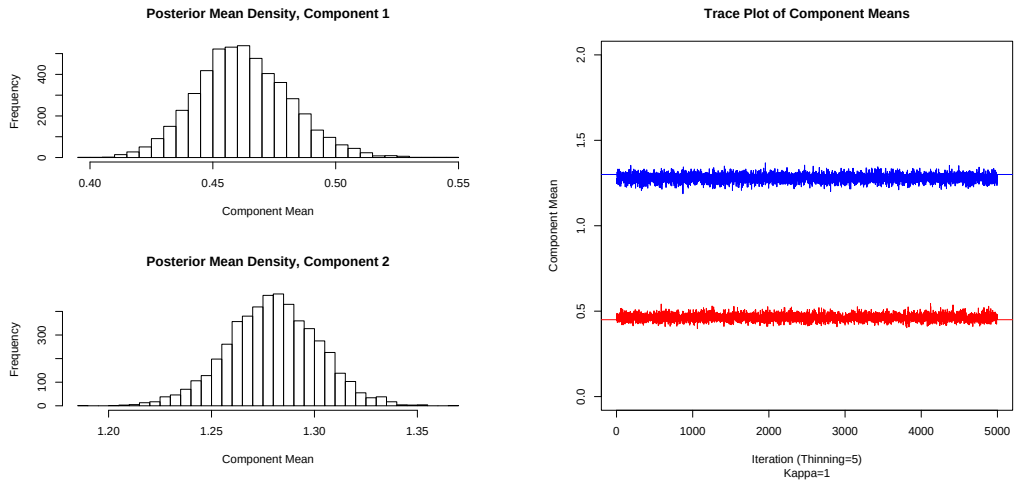

Figure 4.10: Unnormalised posterior distribution of μ (left), together with trace plot of component means (right).

tematic way that each component has data points incorrectly assigned to it. The earliest component (which involved data simulated from a distribution with mean 0.45) can only have data points incorrectly assigned to it from those that belong to component 2, and these are likely to be data points that when incorrectly assigned to the first component, would be found in the right tail of the first component. Similarly, any incorrect assignments of data points to component 2 that should belong to component 1 are likely to be found in the left tail of the second component. Whether this issue is a substantial problem or not is going to depend on the uncertainty in the data from the outset. One notes that the true values of the component means are not too far into the tails of the posterior density, and one can see that 95% highest posterior density intervals would contain the true component mean values.

One would hope to recover the correct standard deviation, which was set to be 0.2 (with the sample standard deviations found to be 0.1796 and 0.1916

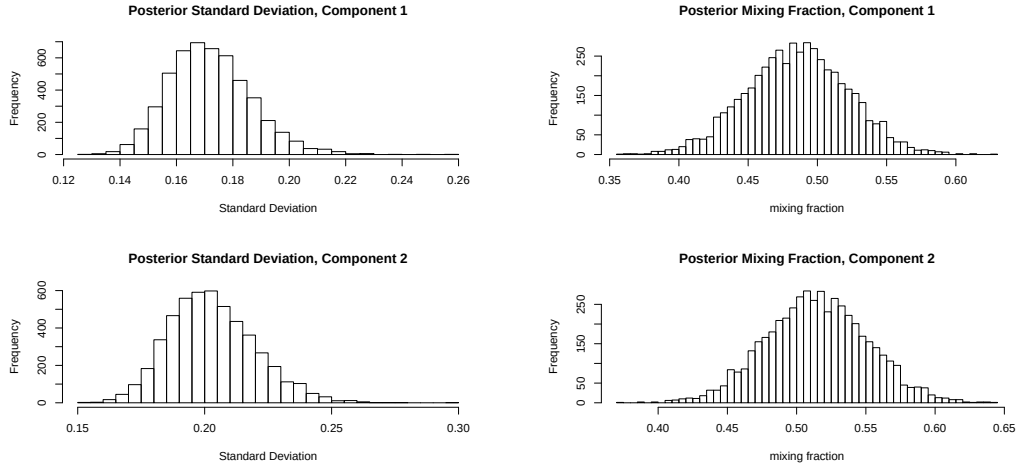

Figure 4.11: Posterior distribution of σ (left) and the p_i fractions (right). Both unnormalised.

for components 1 and 2, respectively), which is the case (figure 4.11, left). Figure 4.11 also shows the posterior distribution of the p_i fractions, which is seen to reflect what one would hope to see, in that the densities are centred approximately on 0.5.

It should be clear that the method is performing reasonably well.

4.7.10 Simulations with badly-chosen priors

In this section, the previous simulation is repeated except that the prior hyperparameters are modified to ill-match the parameter choices in the simulation of the data. This section can be viewed as a test of robustness of the method under prior misspecification. The first simulation involves a simple change in the ξ vector, which was previously set at the true values of the μ's, $(0.45, 1.3)$. One would hope that changing this vector so that it differs from the truth would not have serious consequences on the posterior densities obtained, particularly when the number of data points in each component is as high as 100. Figure 4.12 shows the posterior densities to be little changed when the ξ vector is set to $(0.9, 0.95)$, with everything else remaining identical to the previous case.

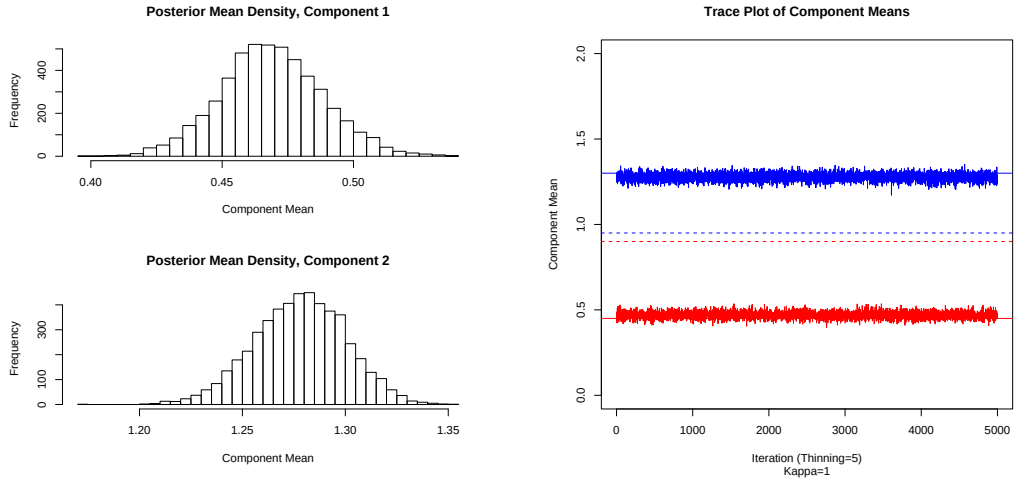

Figure 4.12: Posterior distribution of μ after modifying the ξ hyperparameter values (left) and the associated trace plot with the ξ hyperparameter values being modified to $(0.9, 0.95)$, shown as the broken lines in the figure (right).

While the ξ values only express one's prior belief about the location of the

component means, the κ hyperparameters can be viewed as being related to the *strength* of one's belief about the location of the components. Recall that the prior component mean, conditional on the component variance, is distributed as $N(\xi_i, \sigma_i^2/\kappa_i)$. Setting κ to 1 is an obvious choice as it makes setting (and interpreting) the prior distributions of the component means more straightforward. Increasing κ_i above 1 tightens the prior distribution of the component mean around its ξ_i value. Thus, as κ increases the hyperparameter ξ is given more weight: the parameters of the distributions which lead to the posterior draws start to become dominated by the ξ_i prior, as κ increases. This effect is shown in the next three brief simulations.

Holding $\xi = (0.9, 0.95)$, but increasing κ_1 and κ_2 to 10, with everything else held fixed, one obtains the posterior density and trace plot for the means as shown in figure 4.13. Note that the component mean estimates have been pulled closer to the ξ values (shown as the broken line on the trace plot). It is also worth noting here that the posterior standard deviation starts to show deviations from the true value, with both components demonstrating slightly inflated posterior mean standard deviations of 0.221 and 0.226.

When increasing κ further one starts to see the prior having a significant effect on the posterior densities obtained. Figure 4.14 shows the posterior mean densities and associated trace plots when κ is increased to 20. It is clearly visible now that the posterior mean densities are being strongly pulled towards the ξ values. It is notable that other recorded variables, particularly the number of data points assigned to each component, start to show large departures from the previous cases seen with the less informative priors. Figure 4.15 shows the posterior density estimates of the number of founders in each component and the standard deviations.

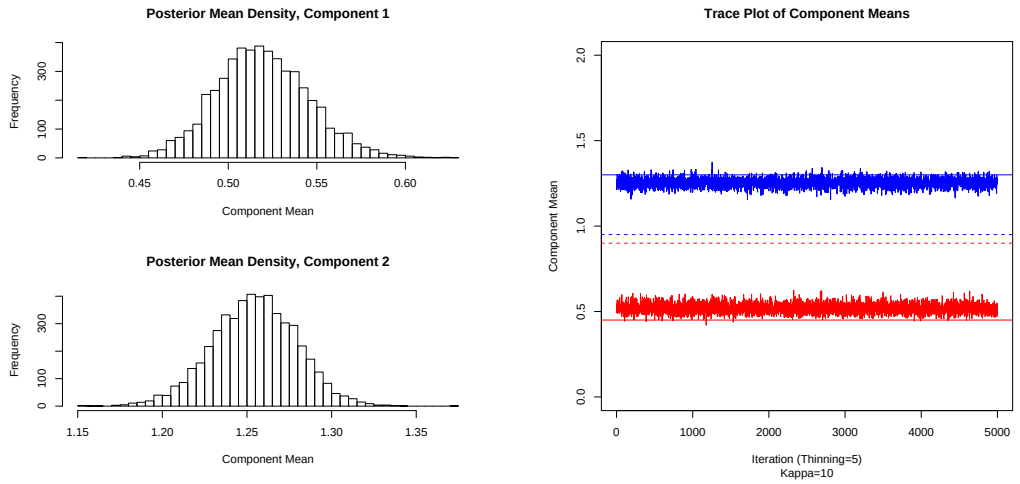

Figure 4.13: Posterior distribution of μ after modifying the κ hyperparameter values to be 10 (left) and the associated trace plot (right).

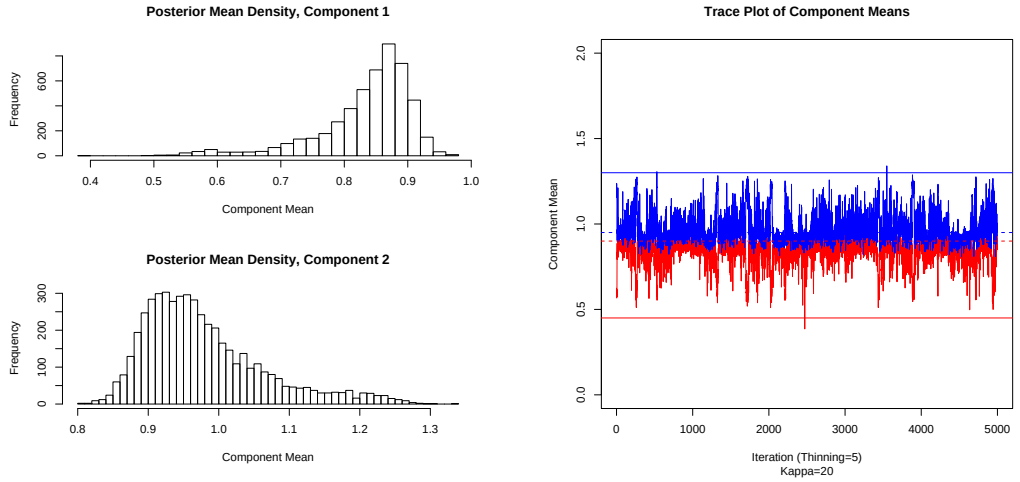

Figure 4.14: Posterior distribution of μ after modifying the κ hyperparameter values to be 20 (left) and the associated trace plot (right).

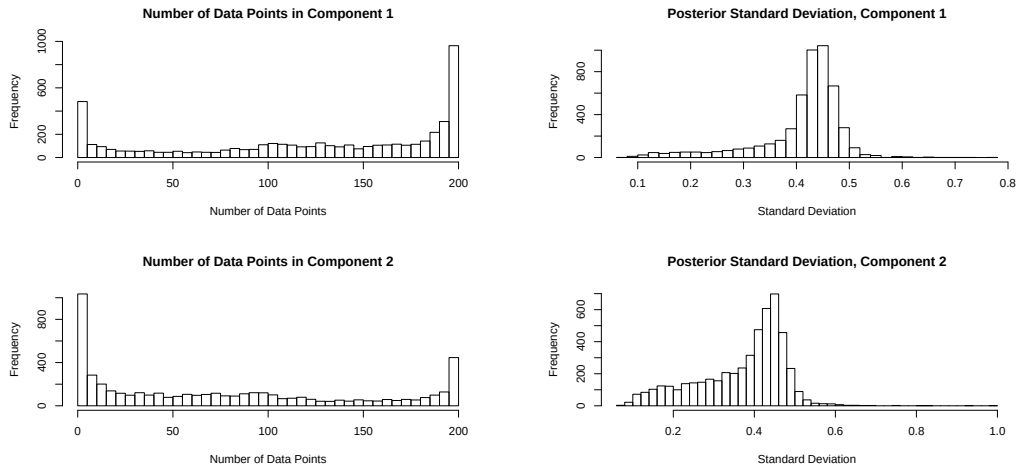

Figure 4.15: Posterior distribution of number of data points in each component with $\xi = (0.9, 0.95)$ and $\kappa = (20, 20)$ (left) and the distribution of σ (right).

Across the simulation the number of data points assigned to each component spans the entire possible range of values. What is happening here is that the large κ value is forcing the posterior component means to take on values which are not supported by the data originating from *either* component. This results in the allocations to components being fairly arbitrary (assuming that both component means are far from the truth); the consequence of this is that components can become empty. Empty components are problematic as the next component mean (and other) updates then essentially become draws from the priors, which are strongly peaked on the wrong values with large κ. Further, the data points must be assigned to *some* component, and this is why the number of data points in a given component is fairly uniform when the component is not empty or containing all the data points. This example shows that inappropriate prior choices can lead to problem cases such as empty components, that is when the hyperparameters are chosen

to be strongly informative with large mass at areas that the data does not support. The posterior standard deviations show an incremental increase again, which is to be expected when the posterior mean estimates are forced (through prior choice) to occupy areas of migration time that the data does not support.

4.7.11 The sample size effect

An important issue is that of sample size. The luxury of a simulated data set is one that an investigator does not have, and components are likely to exist with a relatively small number of members. In this section, the number of members in a given component is reduced to try to gain some insight into the effects on the posterior density estimates.

The first example considered involved reducing the number of members of the second (older) component to 50, while retaining the 100 members of the first component. The means of the simulated data from each component were found to be 0.456 and 1.293, respectively, with standard deviations 0.189 and 0.222 respectively.

With identical (the original reasonable) prior choices as described previously one simply needs to inspect the plots that arise from the simulation. Figure 4.16 shows the number of data points assigned to each component, together with a histogram of the stored values of the mixing fractions.

From figure 4.17, one can see that the component mean histograms do certainly contain the true values. What is perhaps more interesting though is the trace plot of the posterior means, which displays greater variability in the posterior mean for the second component (the one which contained only 50 observations).

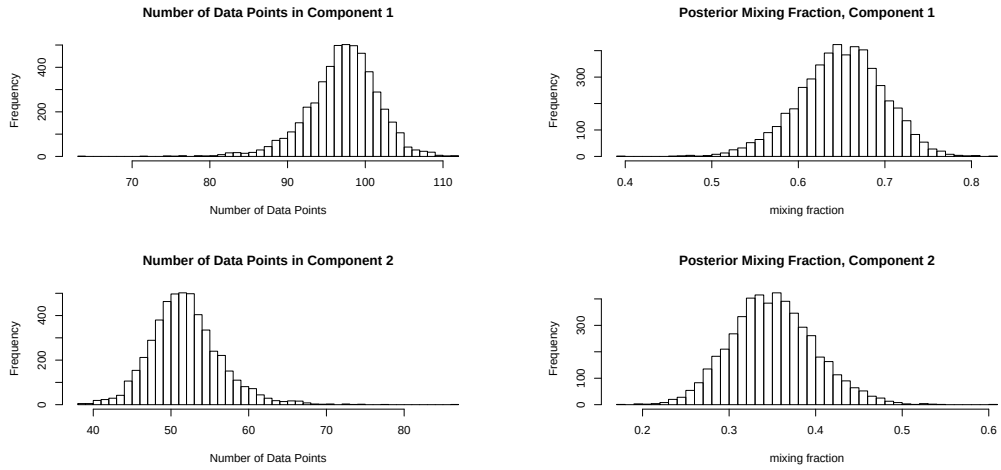

Figure 4.16: Posterior distribution of number of data points in each component (left) and the histograms of the mixing fractions when component 2 only has 50 members (right).

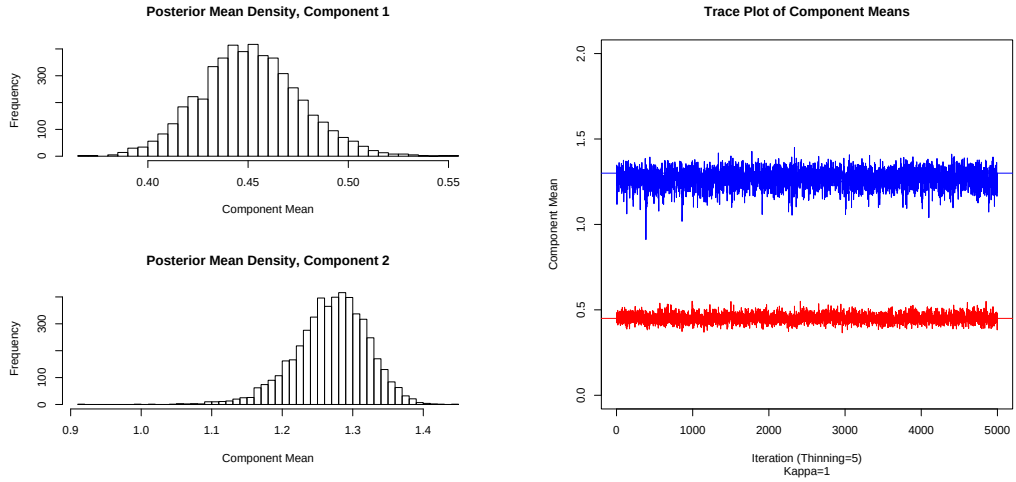

Figure 4.17: Posterior distribution of μ when sample size in component 1 is 100 and in component 2 is 50 (left) and the associated trace plot(right).

4.7.12 The α hyperparameters

Recall from equation (4.55) that the full conditional distribution of the mixing fractions is distributed as a Dirichlet distribution with parameters $\alpha_i + m_i(z)$. Within Bayesian mixture models, it is normal for the hyperparameters to be chosen so that the prior on the mixing fractions is uniform on the simplex $\sum_i p_i = 1$, i.e. with $\alpha_i = 1, \forall i$. While this seems reasonable, in some instances this hyperparameter choice leads to results with some notable consequences for cases that could be important in practice.

As discussed briefly previously, the process of reconstructing a phylogeny from a modern day sample with the aims of attempting a founder analysis necessarily brings with it the issue of a decreasing pool of sequences which could be founder sequence types as one goes back further in time. This has consequences in the allocation of founder sequence types for founders that are estimated to have originated from periods which lie between two components. The problem of allocating founder sequences that lie in the tails of two components is actually more difficult than one would first think; it turns out that founders lying exactly between 2 components with equal variances, are more often assigned to the component with the largest number of members. This can be explained by considering the following theoretical example. Of course, such an artificial construction is not likely to occur as clearly as shown below in practice. However, the case discussed is instructive in explaining the issue at hand.

Assume a two-component model with component densities which have little overlap in the tails of the distributions, for example two normal densities which are fairly well separated and sharply peaked so that little mass is contained in the tails. Now, suppose the data is informative enough so

that at every stage of the inference process, the posterior component means, variances and all other parameters of the model are accurate (within some acceptable range of values since these will be varying at every iteration).

Further, consider the allocation update (4.56):

$$z_j|\tau_j,\theta \;\sim\; M_2(1;p_1(\tau_j,\theta),p_2(\tau_j,\theta)),$$

where

$$p_i(\tau_j,\theta) \;=\; \frac{p_i\varphi(\tau_j;\mu_i,\sigma_i)}{\sum_{l=1}^{2}p_l\varphi(\tau_j;\mu_l,\sigma_l).}$$

One can see that, in the allocation equation, in the case where the τ_j (migration time) does not give any information about component membership through its probability density value (i.e. it lies between two components in such a manner that the normal density part of (4.56) contributes the same for both components), all that remains that determines the parameters of the allocation update are the p_i fractions. This leads to a problematic situation where a founder is assigned more often to one component over another, simply because the distribution which determines the assignment has parameters that are strongly influenced by the number of members of each component.

Formally, the full conditional distribution of the p vector is $p|\tau,z \sim Dir(\alpha_1 + m_1(z),\alpha_2 + m_2(z))$. When the number of members of both components are finite (and 'small'), and one component has twice the members of the other (call the sizes N and $2N$), with $\alpha_i = 1$, the full conditional distribution that is drawn from becomes $Dir(1+2N,1+N)$. This distribution tends to give p vectors which can assign relatively large values to the entry that corresponds to the component with the largest number of members. Figure 4.18 demonstrates this property for the specific case when $N = 50$. Now, when one

returns to the allocation update equation, in the absence of any information about component membership from the respective τ value of the founder, the multinomial draw which assigns the founder to a given component can be viewed as almost exclusively being determined by the p vector. The end result of this is when the founder time is very uninformative with regards to component membership, the allocations of founders to components can be strongly influenced by the number of founders in each component, as this determines the p vector's contents. Of course, one may argue that the above

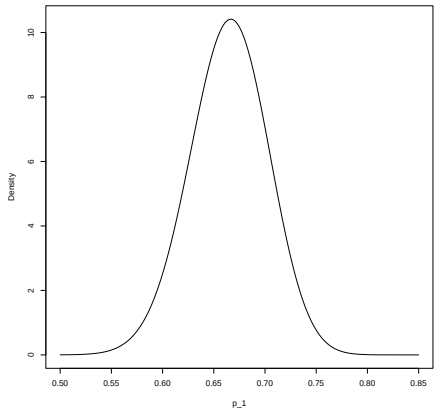

Figure 4.18: Density curve of the Dirichlet distribution, $Dir(1+2N, 1+N)$, with $N = 50$.

is in fact not a problem with the model or the updates, and instead argue that this is in fact one of the strengths of the Bayesian approach. To this end, it can be argued that one is using the information in the complete data set which has determined (we can assume for the moment, correctly) that one component does indeed have more members than another. In the cases when the migration times are unclear or even completely uninformative with regards to an assignment to one of two possible components, one can argue

that the complete dataset has identified that it is more likely to have come from one component (the larger) over another. Taking this hypothetical argument to an extreme, one could ask the question 'In the absence of *any* migration time data, in a two-component model in which the experimenter was satisfied that one component contained double the number of data points of the other, where would such an experimenter assign such an uninformative migration time if such an assignment had to be made?'.

While this discussion is of interest in its own right, the issue is slightly more complicated within population genetic models where one knows from the outset that the pool of sequences from which one could identify founders is non-increasing as one goes back in time (since the number of ancestral lineages is a death process). Relating this to the discussion at hand, one could argue that the complete data set may not always be *correctly* identifying that one component contains more than the next, and instead may just be reflecting the fact that the number of coalescent events decreases going back in time, as does the number of potential founder sequence types at each migration period. The consequence of this is that it is plausible that the number of founders belonging to each component decreases as one moves from the most recent migration period to the oldest only because of the way in which a phylogeny is reconstructed. It is in fact very plausible that the latest migration period considered may in fact only contain a very small number of founder sequences. In such a situation it is difficult to support the idea that in cases where the migration times are very uninformative one should put significant weight on the number of founders assigned to each component.

With the previous discussion complete, one returns to specifying the α vector

hyperparameter. The first question of interest is whether setting $\alpha_i = 1, \forall i$, is a sensible, or even 'safe', hyperparameter choice. The answer to this question turns out more straightforward when considering the possible alternatives.

One could attempt to incorporate the prior knowledge that the pool of possible founder sequences types is decreasing as one moves from the most recent migration period (component) to the latest migration period and try to manipulate the prior distribution in such a way as to model this effect. This would involve a non-symmetric α vector, with $\alpha_i \leq \alpha_j, i < j$. Essentially, this option puts more weight on assignments to older components (compared to the $\alpha_i = 1, \forall i$, case) when the τ value is relatively uninformative. The problem with such a hyperparameter choice is that it relies on a prior judgement being made by the investigator about the relative rate of decrease in component size due to the ancestral death process. This is a difficult problem in its own right and such an approach would be extremely difficult to justify in practice.

An intermediate hyperparameter choice would be one which involved a Dirichlet prior with an alpha vector consisting of a single value greater than 1. This option brings with it the nice feature that, for components with very few founders assigned to them, in cases where the τ value is uninformative for assignment purposes, the p vector is moved closer to being a symmetric vector with all entries equal to 1/(number of components). The problem of course in this case is that, as $\alpha_i = \alpha \to \infty$, the model is essentially throwing away any information in the data that exists about the relative number of founders contained in each component.

The final option is a compromise that attempts to reduce the effect of the death process that is likely to cause the number of founders to decrease as one

goes back further in time, while, at the same time, attempts to reduce the number of allocations that are made based on only the number of founders that belong to different components. A possible option is to repeat the analysis for a range of α values, including 1, and up to the α value which results in no single element of the p vector falling below a given threshold (ω) set by the investigator, on average (e.g., in a two-component model, the investigator may wish to increase α in increments of 5, until such a time that the minimal element of the p vectors, on average, is no less than ω).

Chapter 5

Data analysis - preparing the dataset

5.1 Extracting the data

With the method of analysis previously described, I set out to prepare a suitable dataset to analyse. Fortunately, the original database used in the work of Richards et al. [26] was available to me. This automatically provided the necessary data/founder sequence type age estimates (ρ_A and n_a) under each of the $f0$, $f1$, $f2$ and fs criteria. However, this was no longer sufficient since my method requires the ρ data for the enclosing/containing clusters. In what follows the cluster that is defined by the ancestor of a founder sequence type will be referred to as the 'containing cluster'. To obtain the required data (n_b and ρ_B), the original networks as constructed by Prof. Richards were necessary to enable the trees to be re-created and the n_b and ρ_B data calculated.

Before discussing the data preparation process in some detail it is worth not-

ing here an important difference in the containing cluster data compared to the founder cluster. The original cluster (defined by the identified founder sequence type) contains only European sequences, while the containing cluster almost always contains at least a single Near Eastern sequence (the presence of such sequences was necessary to identify the founder; however recall that some founders were *inferred* founders). In many instances it was the case that the containing cluster was much larger than the founder cluster (perhaps twice the size or more). The initial reaction one has is to assume that this is a good feature as (under the assumption of a connected star-tree model) a larger number of descendants should be good for inferential purposes when the ρ_B calculation is done. This raises some other statistical issues which are open to discussion, e.g. those of sampling.

5.1.1 Sampling considerations.

Sampling issues in this context encompass some standard statistical problems such as sample size, but unique sampling questions arise which are specific to genetic data, and some of which are particular to any method which is based on identifying sequences which are likely to be involved in migration events. It is somewhat unfortunate that a present-day sample of thousands of sequences may give rise to only $100 - 200$ inferred founder sequence types. This problem is one which can be easily appreciated by considering that the most common European sequence types are necessarily sampled most often in a random sample, and these common sequences contribute almost nothing in defining more founder sequences, since, under the assumption that the correct founder(s) have already been identified, most common European sequence types will simply add one to the n_a value of that founder cluster, essentially contributing *nothing* to the dataset that is actually *used* in the

analysis, other than a slight refinement of the n_a and ρ_A values for that cluster.

Increasing the sample size of Near Eastern sequences is a more difficult concept to evaluate, as founders can be inferred based on only a single Near Eastern sequence. With a finite European sample, fixed and unchanging, increasing the number of Near Eastern sequences in the sample is useful only up to the point where the founder list is saturated (every founder sequence type is identified). As far as I am aware, no work has been published to try and model the number of inferred founder sequences as a function of the number of Near Eastern sequences sampled. It is my belief that the number of inferred founder sequences would increase relatively quickly as the number of Near Eastern sequences increased from zero to some small value as each newly introduced sequence would have a high probability of defining some new founder cluster. However, once the number of Near Eastern sequences sampled reaches more moderate levels, the rate at which new founder sequences would be identified would decrease (as some Near Eastern sequences would not define new clusters), and, given a finite European sample, adding more Near Eastern sequences would eventually result in no change to the founder list.

Of course, in reality, the number of European and Near Eastern sequences will be finite and not extremely large. The sampling issue that one has to deal with initially is that of sampling proportion. Does one sample more Europeans with the aim of obtaining relatively good ρ_A values (especially for the founders of the most common sequence types), but accept that this could result in few inferred founder clusters, some with a large number of members? Or, does one sample more Near Eastern sequences with the aim

of identifying more founder clusters, but accepting that some may be very poorly defined with perhaps only one or two European sequences contained in many clusters? A more difficult sampling question, but perhaps a more important one, brings cost into the problem: even assuming that the optimal sampling proportion had been identified, given a finite amount of money, does one sample more sequences (in the 'optimal' proportion) or does one sequence more sites on the sequences of a smaller present-day sample?

Sampling issues are unfortunately, by their very nature, issues which should be worried about *before* any data is collected, and as a consequence no solution or any suggestions as to a proper sampling procedure will be put forward here. However I would hope that any future studies that may be undertaken using methods such as founder analysis will think harder about such issues at the data-collection stage.

5.1.2 Reconstructing the networks

The original hand-drawn networks of Richards et al. were obtained through personal communication and formed two folders of drawings which were used as the starting point in the original work. Before continuing it is necessary to clarify the original notation. The notation r was used to define the number of mutations on the (assumed star) tree of that founder cluster. Such founder clusters often did not resemble perfect star trees and had ρ mutations which contributed more than a single count to the r value. Thus, I now simply extend this notation a little and introduce r_A as the number of mutations (being careful to remember that ρ mutations contribute more) on the subtree defined by the founder sequence type, and r_B as the total mutation count on the full part of the tree defined by the founder sequence type.

The founder criterion $f1$ was selected for analysis, the primary consideration being that of sample size: the $f1$ criterion identified 134 founder sequence types compared to $f2$ which identified only 58 founder sequence types. Although use of the $f0$ criterion would have involved a larger founder pool (210 founder sequence types), it undoubtedly is subject to the largest number of false positives as it involves no filtering of the founder candidate list: inflating the size of the founder list is difficult to justify when one knows that the additional founders do not pass the more stringent criteria for inclusion under the $f1$, $f2$ and fs criteria. The fs criterion (106 founders identified) was rejected due to the rather arbitrary formula used to identify founders as has been explained earlier. It is perhaps worth noting that [26] put more weight on the fs list; it is my belief that, until this criterion is subject to more thorough investigation, its performance is open to question. The $f1$ criterion filters out the most likely false positives, provides a relatively large number of founder sequence types and the method it uses to select founders is very well determined and clear. It is noted here that 31 founders identified under $f1$ were not identified under fs, while only 3 founders identified under fs were not identified as founders under $f1$. This observation is of interest as it shows that the fs list contains the same core inferred founder sequences as that of the $f1$ list.

5.1.3 The data extraction process

The reader is encouraged to consider the data preparation and cleaning described here as it gives insight into some simplifications that are made in the method. Before worrying about any n_b or ρ_B calculations, the database and diagrams were used to re-construct the phylogenies. As the n_a and ρ_A values were calculated from European sequences, it became clear that the assumption of a star tree and no (or little) recurrent or parallel mutation actually was an extremely strong one, as often one would see identical European and Near Eastern sequence types (defining the founder cluster), but yet the necessary reconstruction that gave a founder cluster with only European sequences required parallel mutations, and in some instances involved sites which were not 'fast'.

An example of this is founder sequence type h10 (256), which was a sequence seen twice in Europe and once in the Near East. The problem with reconstructing h10 (and the other similar founder clusters) was that it had founder sequence type h73 (148 256 319) as a subcluster (although this was a founder only under the $f0$ criterion). Figure 5.1 shows this cluster. The resolution of this h10 cluster provides some insight into a problem which was not obvious from the original paper. Dealing with h73 first (although this would not feature in the $f1$ dataset), this $f0$ founder sequence type consists of a cluster only with 2 members having the same mutations (148 256 319).

Reconstructing h73 was indeed trivial, but the problems start appearing when one considers reconstructing h10 which also contains both sequences from h73. At first glance, h10 looks equally as trivial to reconstruct, it contains only a single Near Eastern sequence that shares the same sequence as the founder sequence type (256). All of the additional European sequences in

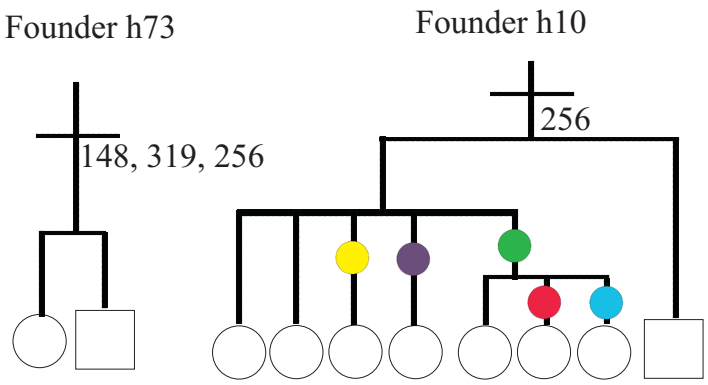

Figure 5.1: Founder sequence types h73 (left) and h10 (right). Mutations which feature in both founder clusters are numbered, other mutations are represented by solid circles. The two members of h73 are part of h10 and need to be added to it.

h10 have a mutation at 256 and some other mutations that are easily resolved (no additional shared mutations at all). However, the European sequence (148 256 319) that formed part of h73 now needs to be considered part of this founder cluster. It is natural to assume that one can just place an extra branch on the tree with the 148 and 319 mutations to represent this European sequence in this founder cluster. This is problem-free as these mutations do not feature elsewhere in the reconstruction of h10. The issue that presents itself is how to deal with the Near Eastern sequence (148 256 319). None of these sites is 'fast', so one does not really wish to add an identical sequence to the Near Eastern side of the tree: such a parallel mutation is unlikely.

Ideally, one would wish to add this sequence to the same part of the tree as the identical European sequence just added (a back-migration into the Near East). This, however, violates the assumption that the original migration occurred, forming a founding sequence (assumed here as 256), which then

dispersed in *Europe* giving rise to derived sequence types. The only reconstruction that gives *only* European sequences below the h10 founder sequence (256) requires a parallel mutation in Europe and the Near East that gives rise to the (148 256 319) sequence seen in both areas. If one allows Near Eastern sequences in the founder cluster (perhaps arising due to back migrations) and one simply wishes to include only the European sequences in the ρ calculation then this is fine and such a reconstruction would give values which agree with the original database. In all of the reconstructions undertaken, the trees were resolved by parallel mutations in such circumstances, leading to values which agreed with the original work, although this does raise some questions about likely back-migration.

Taking the previous issue further (perhaps a more obvious complication that arises when considering the need now for n_b and ρ_B data), the containing clusters which make up the connected star tree from which the n_b and ρ_B values are determined often contain multiple $f1$ founder clusters as part of the comb, and in some rare cases the containing cluster coincides with a major cluster/haplogroup. An example of this is founder u22 (reference sequence in U) which had as its containing cluster all of U, which amounts to 1296 sequences. Further, founder hv06 ([067]) has as its containing cluster all of HV, which includes the most frequent haplogroup in Europe, H, as well as V. It is perhaps worth naming the founders here that coincide with the major haplogroups, as the n_b and ρ_B data that arise from them are of some interest. They are u22 (reference sequence in U), w01 (W, 223 292), v01 (V, 298), i01 (I, 129 223), k01 (K, 224 311), ph01 ((pre-HV)1, now R0a [63], 126 362), x01 (X, 189 223 278), n05 (N1a, 172 147t 223 248 355), n01 (N1b, 145 176t 223), n07 (N1c, 201 223 265), j00 (J, 69, 126), hv06 (HV1, [067]), t01 (T, 126 294), and h00 (reference sequence in H).

Perhaps a more troublesome issue results from the fact that the reconstructed phylogenies occasionally had common mutations (at fast sites) occurring more than once in different parts of the tree, and occasionally as back mutations within a given cluster. The definition of ρ essentially is the average number of mutational differences between the node in question and those forming the external edges. However, when calculating the ρ_B data it was seen in some instances that the containing cluster could have a mutation at some fast site (e.g. 189), and then much further down towards the tips of the comb, that same mutation could occur again (perhaps for more than one sequence). The site in question could then agree with the sequence it is being compared to, but only because two transitions have been assumed to have occurred at that site (for each of the sequences that have had the back-mutation).

The question is whether one counts these as being an extra two mutations different from the node they are to be compared with. I have counted these as multiple mutations. This is motivated by considering a (hypothetical) reasonable tree reconstruction, for example, done by a reputable geneticist. The reconstruction has a node labelled A carrying no mutations (some sort of reference that we wish to date). This node has many descendants, and one major branch of its descendants carries the mutation, mutation 1 occurring at site α. Within this major branch a minor branch is reconstructed which contains a new mutation, mutation 2 at site β (defining this minor branch). However, within this minor branch a sequence exists (node B) which has a back mutation at site α. Assuming the reconstruction to be correct, it seems perfectly reasonable that, when trying to date node A using ρ, both mutations at site α are counted. Failing to do so suggests that one is unhappy with the reconstruction or is being selective in the use of the data provided

by the geneticist's reconstruction.

Figure 5.2 shows a summary of the complete dataset as will be analysed with the extended founder analysis method and table A.1 of Appendix A shows the complete dataset. More detailed figures (figures A.1-A.10) can be found in Appendix A which explicitly detail the founder label, as well as the n_a and n_b values for each founder. These additional figures do warrant some inspection as they do contribute to one's understanding of some of the problems of both the original founder analysis method and the proposed extended founder analysis method.

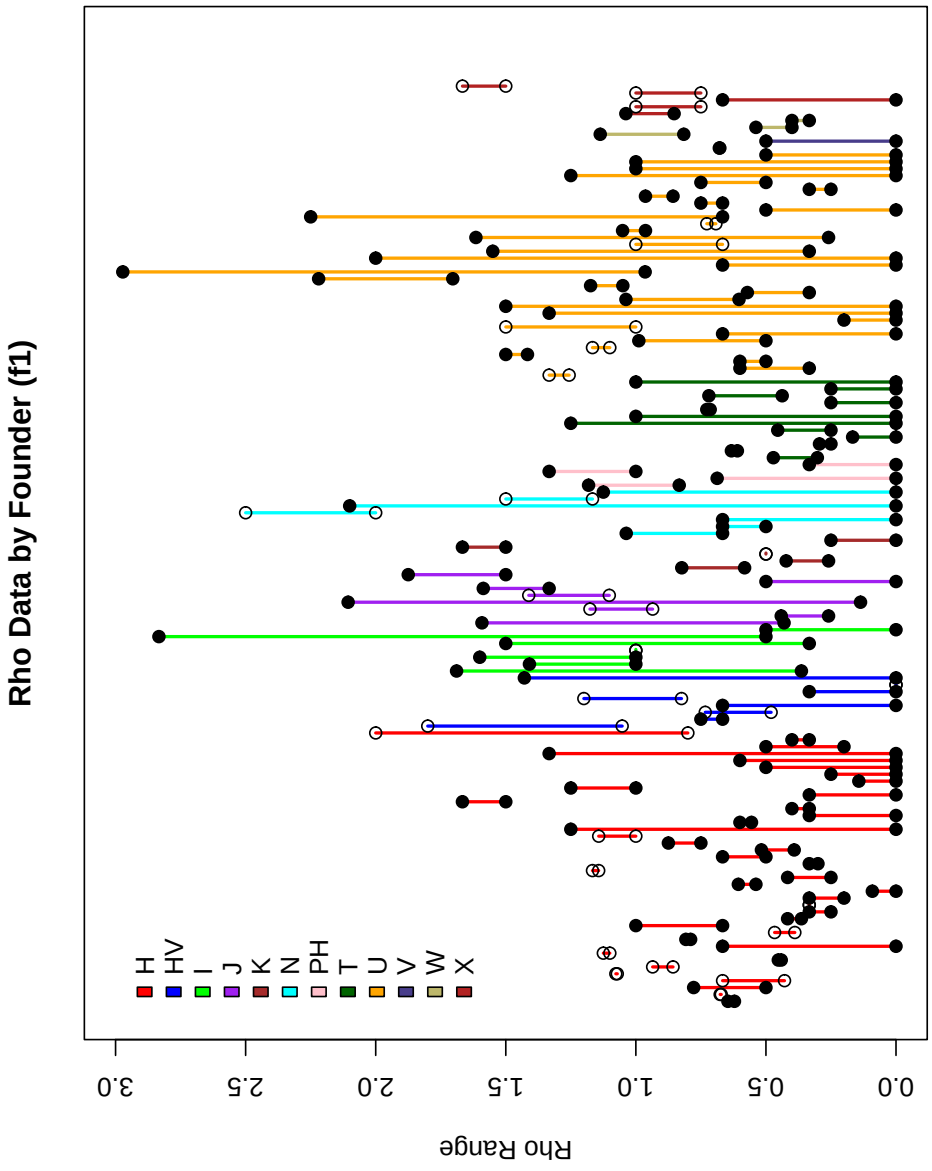

Figure 5.2: Plot of the final dataset. Hollow points indicate ρ_A being greater than or equal to ρ_B.

5.1.4 Observations on the prepared dataset

The additional figures in the Appendix use hollow points to signify when the ρ_A and ρ_B values exhibit the property that $\rho_A \geq \rho_B$, which is particularly problematic for any method (such as my proposed method) which uses this data to estimate the relevant τ_A and τ_B values. It should be noted here however that this issue, where using a dating method to date the ages of nodes on an assumed phylogeny can suggest nodes having the 'wrong' date ordering, is a problem which never manifested itself in the original method [26] since only a single ρ estimate was calculated for each founder sequence type. The only new problem here is that of ensuring this feature does not disrupt the statistical analysis (in particular, the mixing).

It is of substantial interest to note that many (41) of the dataset ρ_A values are zero. This is unsatisfactory if the investigator wishes to relate these estimates to the migration time of that founder. Almost every major haplogroup has at least a single founder sequence with a zero ρ_A value - the reconstructed cluster that belongs to that founder sequence type does not display any mutations at the sites that have been sequenced (usually due to the cluster being tiny, perhaps of only 1 or 2 lineages).

On a related point, it is notable in many instances (e.g. founders j00, j03, n05, u22, u31, to name a few) that the ρ_A and ρ_B values can differ quite markedly. The issue which then arises is which estimate is likely to be closest to the true unknown migration time/founding event. This is something that one cannot say with any certainty.

A final note is that, in the cases where the ρ_A and ρ_B values do not display the natural ordering one would expect, it is extremely rare in such cases to see large discrepancies (≥ 0.25) between the ρ values, and in almost all cases the

sizes of the founder clusters (the n_a and n_b values) in question are small. An exception is h01 which has $n_a = 108, n_b = 131$ with ρ estimates in the wrong order (although very close), while hv06 has $n_a = 5, n_b = 1254$ (due to the containing cluster containing all of H and V), with the ρ estimates again in the wrong order. These odd-seeming cases actually are helpful in evaluating and understanding the method's performance and for investigating mixing: these issues will be revisited in the data analysis section.

Chapter 6

Data analysis

6.1 Re-analysing the original dataset using the original method of analysis

The dataset (in terms of the ρ_A and n_a values) as reconstructed was only slightly different from that which was used previously [26]. These differences arise because of the way in which mutations on singleton founders were allocated. This change however has the effect of making some founder clusters appear *older*, as well as possibly having implications for the S_m proportions (recall equation 2.3). Regardless of these (minor) changes, it is desirable to re-analyse the data in the identical manner to that of the original paper so that comparisons can be made between the old and improved methods. To this end the n_a and ρ_A data was used in isolation and the original method of analysis was re-coded in R [42]. Using the same assumed mutation rate of 1 transition per $20,180$ years [35] (between positions 16090 and 16365), the age estimate of each founder cluster was evaluated (note however that a recent recalibration [64] suggests that a faster rate is likely). The re-analysis gives

50% and 95% credible regions for the age of various founder clusters as shown in figure 6.1 (which requires 25 or more members in the founder cluster for inclusion, approximately 1% of the European sample), while figure A.11 in Appendix A shows a similar plot (which requires 40 or more members in the founder cluster for inclusion, approximately 1.5% of the European sample). These figures display the founders in order of age as measured by the lower end of the 50% credible region.

The dashed vertical lines at 9000, 14500, 26000 and 45000 YBP represent estimates of the ages of the Neolithic, late Upper Palaeolithic (LUP), middle Upper Palaeolithic (MUP) and early Upper Palaeolithic (EUP) respectively, with an additional period added at 3000 YBP to 'mop-up' sequences arising from more recent migration periods/events as described in the original paper. A figure (Figure 1, page 1266) of the original paper displayed similar information (but for $\geq$ 40 lineages). It should be noted however that the original paper used the fs founder-identification criterion, so direct comparisons cannot be made without considering the consequences of using $f1$ instead of fs. The entire set of credible region values can be found in table A.2 of Appendix A.

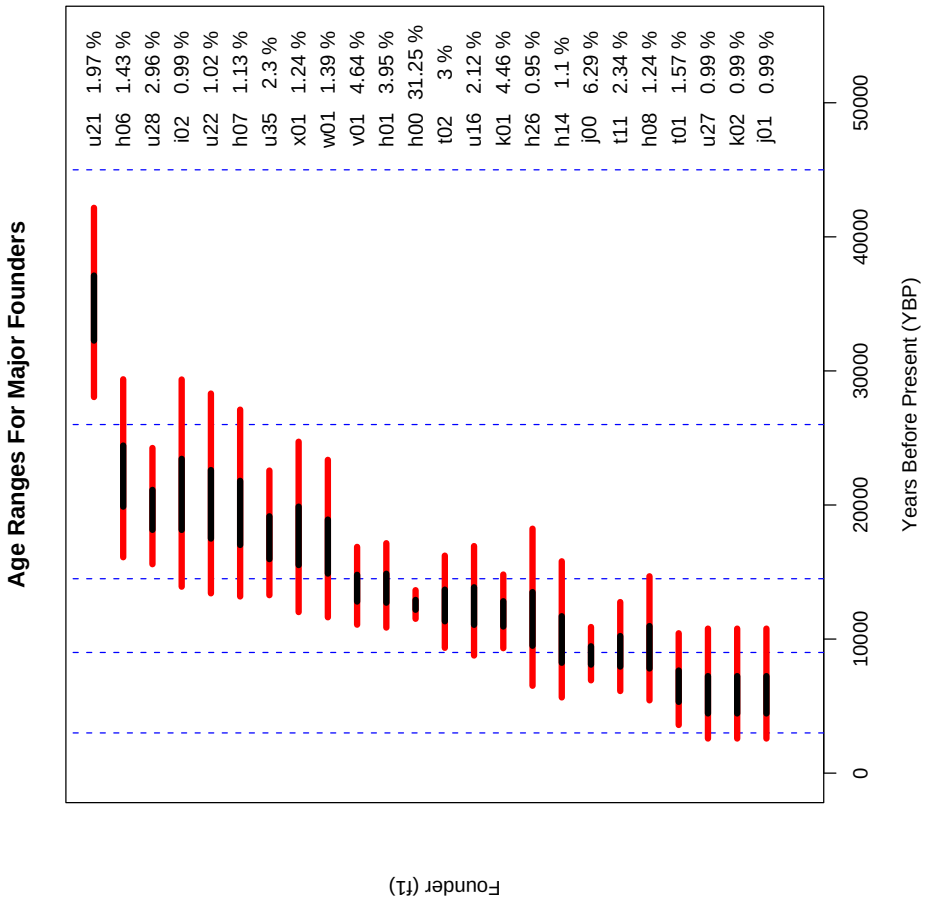

Figure 6.1: Credible regions for ages of major founder clusters using the original method (gamma distribution on ages). The inner bars represent the 50% credible regions while the outer bars represent the 95% credible regions.

The S_m values returned from this analysis can be seen in table 6.1, which very closely matches the original values [26, table 4, page 1267], with some minor differences due to minor dataset modifications. A graphical representation of table 6.1 is presented in figure 6.2.

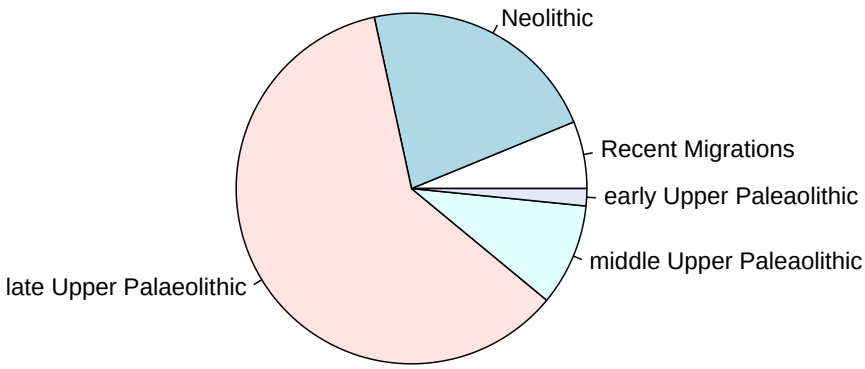

Figure 6.2: Pie chart of the S_m proportions for the $f1$ founder list using the original founder analysis method.

Table 6.1: Percentage of the European sample assigned to each period under the assumed migration model, together with the root-mean-square error.

Period	Mean posterior percentage	RMS Error
Recent Migrations	6.20	1.25
Neolithic	22.22	3.10
late Upper Palaeolithic	60.60	3.42
middle Upper Palaeolithic	9.40	2.14
early Upper Palaeolithic	1.59	1.03

6.1.1 Some observations

It is reassuring from a code correctness perspective that the results obtained do not differ to any significant degree from those presented in the original work. The S_m proportions obtained and the dates of founder clusters closely match those presented in the relevant figures and tables detailed in the previous sections.

What is more interesting to note is that the $f1$ criterion provides dates for founder clusters which are, in general, *more recent* than those obtained by the fs analysis. This information was essentially available for extraction in the original paper but it is slightly disconcerting that substantial differences in the age of clusters can occur when the criteria used to define the candidate founder list is changed. The selection of the $f1$ founder list has been justified previously. However the reader is reminded that the extended founder method which I am proposing in this thesis is likely to place the age estimates of founders further back, essentially allowing them to appear older due to the removal of the assumption that the sequence involved in the migration event

coincides with a node on the reconstructed phylogeny and then immediately disperses (forming the assumed star tree).

It is reasonable to assume that the original method is assigning dates which are too recent, and the reader with some concerns as to the dating of some clusters under the $f1$ criteria, which may appear too *young*, is reminded that these can now be viewed as an estimate of the youngest age of such founders. A specific example is founder u22, the reference sequence in U; this has a 95% CR with an upper limit of $28,312$ YBP. However, the ρ_B estimate for this founder (which is ignored in the re-analysis using the original method) is just below 3, which corresponds to a date of just under $60,000$ YBP. It is acknowledged here that the choice of founder criterion used does affect the results when using the original method and this leads to the acknowledgement that the choice of criterion would almost certainly affect the results of the extended founder analysis method in the coming sections. Regardless, for the purposes of investigating the method and its performance, and comparing it with the previous method, the choice of a single criterion ($f1$) is sufficient and reduces the need for duplicate analyses that contribute very little towards gaining a better understanding at a statistical level of the extended method of founder analysis that I am proposing.

A thorough re-reporting of the conclusions relating to the S_m proportions already published in the original work is not appropriate here, and instead it is perhaps best to hypothesise how the S_m estimate (or the equivalent estimator) is likely to work in the extended method. Recall that, at every stage of the extended founder analysis method, every founder is assigned to a component (period); from this, the *distribution* of the proportion of the European sample that is assigned to each period can be produced.

6.1.2 Some limitations of the analysis

Inferred founder sequences, age estimates of such founders and an estimate of the proportion of the European sample assigned to each migration period of interest are useful and informative, but one may ask what is missing from this analysis. It is of concern that the age estimates of some founders do not span *any* of the proposed migration periods. A problem with the age estimation procedure used in the original work (based on a gamma distribution) is that the investigator is led to think that the 'best' (most probable) age estimates lie somewhere near the middle of the age bands shown in figure 6.1. For instance, the 95% CR for founder u22 (reference sequence in U) is seen to span both the early and middle Palaeolithic, with the 50% CR being uncomfortably centred between both. If the assumption that the early and middle Palaeolithic define significant periods of migration (*with little migration in between*), it is reasonable to suggest that one would wish the most probable ages of any given founder cluster to be close (in some sense) to these assumed significant periods of migration, and *not* instead centred directly between them.

Taking the previous point a step further one can see a related problem, in the presence of uninformative data subject to large variability (as is the case in many genetic datasets), but where the investigator is prepared to make the assumption that the major migration periods defined certainly existed, were at least 'close' to the assumed dates, and were separated by periods of little migration. Then, one would hope that an appropriate analysis would acknowledge this uncertainty with perhaps disjoint credible intervals, one for the situation where the cluster was assigned to the first period, and one when assigned to the second period. It is difficult to argue for a method that returns

age estimates which in many respects contradict the model assumptions that allow the analysis to be possible.

Indeed, it may be the case that some of the problems just described arise simply because of the uncertainty that is underestimated due to simplifications and assumptions (e.g. assuming a single phylogeny being 'correct', ignoring other possible reconstructions, greatly underestimates the variability). Accounting for such sources of uncertainty certainly would result in date estimates which had more variability associated with them (and thus founders which span *no* migration periods become less likely). Regardless of this, it is undesirable that the original method cannot return age estimates in a way which attempts to respect the defined migration periods. It is *never* the case that an $\alpha\%$ CR for a founder age estimate can be composed of disjoint intervals, with each interval being close to the assumed migration periods. The gamma distribution on the ages of founders which do not lie close to a given migration period is basically inappropriate due to the contradictions it introduces with the original modelling assumptions that are believed to give rise to the data.

A final concern with the original method's conclusions arises due to the fact that over 50% of the European sample of sequences is assigned to only six founder clusters (from the total set of 134 founders). One could argue that a quantity such as the proportion of the European sample assigned to each migration period is highly influenced by these six founder clusters. Indeed, founder cluster h00 contains 31.25% of the European sequences and as a result is highly influential in the S_m proportions obtained. Removing this founder from the analysis changes the values obtained dramatically, and one could argue that this makes the method extremely sensitive to a small number of

data points.

It will be shown in the remainder of this thesis that the extended version of founder analysis that is based on the use of Bayesian mixture modelling allows some (but not all) of these issues to be addressed, although the cost of extending the analysis to relax the star-tree assumption is that of lost precision in the estimates obtained. It will be argued however that the estimates which arise from the extended model are more consistent with the underlying assumptions about the migration process. Additionally, the extended model will be shown to return useful parameter estimates which could not be obtained with the original method.

6.1.3 Extended founder analysis - fixed components

In this section, the extended founder analysis model will be used to re-analyse the dataset once the new issue of MCMC mixing has been investigated. Recall from the previous chapter, which described the extended founder analysis method, that the τ_A and τ_B updates arise from a Metropolis move with proposals being the previous values with a small normally distributed deviation on top of these. Moving through the (τ_A, τ_B) space for each founder is the practical problem of mixing. In this section it will be assumed that the migration periods of interest are well defined and composed of $k = 5$ periods, centred on some appropriate times which can be defined by the investigator, with normal distributions with some appropriate variance representing each migration period. Although this may seem restrictive, it will be shown that this version of the model allows some estimates to be obtained which are well-defined and interesting, and that are more difficult to interpret under the more general extended method, and, as far as I am aware, no alternative

methods have been proposed in the literature which allow such estimates to be obtained in such an objective manner as part of a more general method.

I feel it is necessary here to *very* briefly summarise some main features of the extended founder analysis model already presented, together with some features of the dataset that complicate any analysis (and particularly what these mean for mixing).

It has been assumed throughout that the migration periods defined by the investigator gave rise to founder sequences. As a consequence, the assumed migration process that gave rise to the data does not really support large numbers of founders that date between the assumed major migration periods. Unfortunately, the ρ_A and ρ_B data does not always respect the assumed dates of the major migration periods. The extended founder analysis method provides a compromise to the fact that the data appears to contradict the assumed model by putting distributions which are centred on the dates that are believed to represent the major migration periods, while still having probability mass on the periods between these assumed peaks of migration.

If the data were extremely informative, one would hope that each founder would have ρ data which would allow its dating (τ values) to be close (in some sense) to the centre of one of the migration periods. In the absence of informative data, such as the case of a founder with ρ values which span the entire range of conceivable values, it is of particular interest to ask what one would wish to see in such date estimates - this issue has been touched on very briefly already. One would *not* wish to give such a founder a date estimate centred on the middle of the allowable region, together with an estimate that put least mass on the extreme lower and upper ends of the allowable dates. What one would like to see is an uninformative estimate

of the date of the founder, that in some sense *respected the model that was assumed to give rise to the data*. Of course, the completely uninformative data point case is not the best case to aid any data analysis procedure, but it is important that, in the completely uninformative case, the scope for a (false) informative-seeming inference is low and, more generally, one would hope that the parameter estimates obtained would be uninformative while still respecting the assumed underlying migration process.

With a little thought, one realises that such an uninformative estimate should consist of date estimates that are disjoint and close to the peaks of migration. Assuming a fixed component model, and then given an additional single founder sequence with, e.g. $\rho_A = 0, \rho_B = 3.5$, a satisfactory uninformative estimate of the date of such a founding event would simply be one which respected the current components, namely that the new founder belonged to component i ($i = 1, 2, \ldots, k$) with some probability P_i, and the only sensible 'interval' for the age of that founder that one could envisage is a sequence of perhaps disjoint intervals $(I_1, I_2, \ldots, I_k)$, with each representing the age estimate *if that founder originated from migration period/component* i.

6.1.4 Fixed mean and variance case

With the previous discussion in mind, a k-component model with fixed means and variances can be set out. Overlapping components which cover the entire space of allowable dates will be selected so that all possible ρ estimates are supported in the model. Having such overlapping periods is not inconsistent with the underlying assumption of major peaks of migration that define the periods of interest. It simply represents our knowledge that no dataset is ever going to strongly support the model of point masses at k distinct dates.

k will be chosen to be 5 as in the original founder analysis work, and the components will be defined in what follows.

If the MCMC chain for the extended founder analysis method could be run for an infinite number of iterations, it would be the case that the proposal distributions on the (τ_A, τ_B) updates would be largely irrelevant (under some mild conditions). However in reality one needs to worry about the acceptance rates of the proposed updates. Ideally, one does *not* wish to have an acceptance rate that is too low, as low rates mean that the parameters are not mixing well, and can even result in cases where the procedure does not reach a state of stationarity (and so does not in fact return draws from the posterior distributions of interest). The converse is when the acceptance rates are too high, which usually reflects not making large enough moves to explore the parameter space. One wishes to explore the *entire* allowable region, including the boundaries, and it is therefore expected that proposals will be made which give parameter values that are not accepted. An acceptance ratio of 1 can often simply mean that the proposals are almost identical to current values and, as such, the procedure is not mixing well.

In many MCMC applications, one can suggest various parameter values for the proposal distributions and then use shorter runs of the code to help select the appropriate proposal distributions which give satisfactory mixing/acceptance rates. The issue of mixing is complicated further for the dataset I will analyse. It has already been shown for some founders that the ρ estimates obtained do not obey the natural ordering one would hope they would. This is problematic as the τ_A and τ_B updates which depend on the ρ data turn out to be rejected more often (on average) than those cases where the ρ data do obey the natural ordering. It is fortunate that

the proposal distributions do not need to be identical for every founder. The proposals will remain fixed as being normally distributed with a mean equal to the previous τ_A or τ_B value, with some standard deviation, σ, that is to be defined. One could attempt to use some appropriate estimate of σ for each founder computed from the data (eg the Saillard estimator [51]). Such an estimate may be appropriate for a method which respects the true (binary) tree structure of each founder cluster. Under the assumption of a connected star tree for founder clusters, however, it is difficult to justify the use of such involved estimators, when *any* σ value that gives reasonable mixing will lead to *identical* posterior distributions. Alternatively, adaptive MCMC schemes exist which could also be considered, which involve tuning the proposal distributions over runs (see [65]).

To investigate mixing, components will be assumed to have means centred at $0.15, 0.45, 0.725, 1.3$ and 2.25 units (roughly corresponding to 3000, 9000, 14500, 26000, and 45000 YBP). The standard deviations of the components are more difficult to select and ideally should be elicited from researchers working in the area. For the purposes of establishing mixing however, only two values will be selected, 0.2 (representing very weakly defined components with some large overlap), and 0.1 (representing better defined components with some overlap, with only the oldest component being fairly isolated from the rest). Graphical representations of these scenarios are shown in figure 6.3. It is acknowledged here that both selected values are perhaps inappropriate for the earliest component as *large* amounts of the component's mass are lost due to 0 YBP being the minimal allowable date; the final fixed mean and variance case analysis will not have this problem as it will involve component distributions that have been chosen appropriately: the cases considered here are to investigate mixing only and in order to demonstrate a particularly

interesting feature of this model.

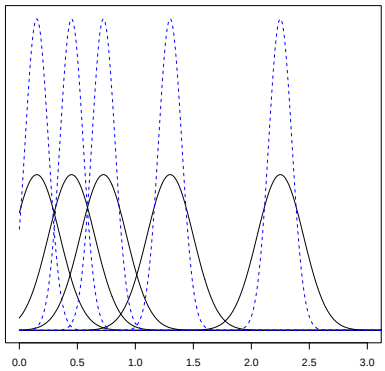

Figure 6.3: Hypothetical probability distributions of migration times for 5 components. Standard deviations 0.2 (solid lines), and 0.1 (dashed lines).

With the migration periods so defined, the variance of the proposal on the τ_A and τ_B updates can be varied and the acceptance rates for each founder determined. The σ values used here are $0.05, 0.1, 0.2, 0.3$ and 0.5. The only optional parameter values here were chosen to be $\alpha_i = 1, \forall i$, and the component means were fixed (assumed known) as mentioned previously. Burn-in was set to $2,000$ iterations, and $5,000$ iterations were stored with no thinning (thinning would be meaningless here as only the acceptance rates are of interest, which do not change with thinning).

Tables 6.2 and 6.3 show the acceptance rates on the (τ_A, τ_B) moves for each of the values of σ under the model with components of fixed standard deviation equal to 0.2 and 0.1, respectively. It is notable that the standard deviation of the components changing from 0.2 to 0.1 does not appear to have a large effect on the number of founders in each band of (τ_A, τ_B) acceptance rates.

Other runs (not shown) at the same parameter values gave almost identical tables in all instances. It is clear however from the tables that a proposal σ of 0.05 is too low as over half the founders have (τ_A, τ_B) acceptance rates of 0.7 or greater, which is large (especially considering this is an acceptance rate on a 2-dimensional parameter vector, and not a single parameter). Similarly, σ values of 0.3 and 0.5 result in large numbers of founders having unacceptably low rates. The decision between using σ being equal to 0.1 or 0.2 (or some different intermediate value) is more of a subjective one. In what follows I select 0.1 simply because further inspection of the acceptance rates at 0.2 revealed acceptance rates for some founders as low as 3%, while the high acceptance rates of 0.6 (when $\sigma = 0.1$) or more which account for just over 40% of the founders can be dealt with using sufficient thinning to ensure that the parameters have moved reasonably far from the previous values at each *stored* iteration.

It is important to remember that the choice of the standard deviation of the proposal distributions would be irrelevant if the method could be run indefinitely. The previous provisional runs simply help to ensure that the process mixes at a reasonable rate, and this in turn helps ensure that the chain has indeed reached stationarity by the end of the burn-in period, and that the stored parameter values represent draws from the posterior distributions. It is clearly beneficial though to use pilot runs on the actual dataset to be analysed to help select appropriate proposal distributions. While it would be possible to attempt to give general proposal distributions for any dataset to be analysed using this method, it is my strong belief that datasets should be dealt with on a case-by-case basis.

Table 6.2: Number of founders with acceptance rates in each band when the component standard deviation is set to 0.2.

Band	$\sigma = 0.05$	$\sigma = 0.1$	$\sigma = 0.2$	$\sigma = 0.3$	$\sigma = 0.5$
$0.0 - 0.1$	0	0	6	14	48
$0.1 - 0.2$	0	5	11	33	56
$0.2 - 0.3$	0	5	31	45	19
$0.3 - 0.4$	6	9	30	23	9
$0.4 - 0.5$	4	25	33	10	2
$0.5 - 0.6$	8	33	10	9	0
$0.6 - 0.7$	33	34	11	0	0
$0.7 - 0.8$	53	16	2	0	0
$0.8 - 0.9$	24	7	0	0	0
$0.9 - 1.0$	6	0	0	0	0

6.1.5 Extended founder analysis - fixed mean and variance case analysis

In this subsection, a complete re-analysis of the dataset will be undertaken, using the same parameter choices, with a proposal distribution for the τ_A, τ_B updates having a standard deviation of 0.1 (in light of the previous section). Particular emphasis will be on displaying what is returned by the method.

Recall that the lowest acceptance rates seen when the proposals involved a standard deviation of 0.1 were within the $0.1-0.2$ band. A closer examination of the runs for this proposal distribution revealed the lowest acceptance rate to be ≈ 0.15. Acceptance rates are often misleading quantities as thinning of MCMC chains is normal. An acceptance rate of 0.1, say, is not a major prob-

Table 6.3: Number of founders with acceptance rates in each band when the component standard deviation is set to 0.1.

Band	$\sigma = 0.05$	$\sigma = 0.1$	$\sigma = 0.2$	$\sigma = 0.3$	$\sigma = 0.5$
$0.0 - 0.1$	0	0	6	13	48
$0.1 - 0.2$	0	4	9	33	55
$0.2 - 0.3$	0	7	27	41	19
$0.3 - 0.4$	6	6	35	27	10
$0.4 - 0.5$	6	18	31	10	2
$0.5 - 0.6$	5	41	12	10	0
$0.6 - 0.7$	33	35	13	0	0
$0.7 - 0.8$	47	14	1	0	0
$0.8 - 0.9$	31	9	0	0	0
$0.9 - 1.0$	6	0	0	0	0

lem if the procedure is thinned substantially. That is, if no thinning was in place, such a parameter's trace plot would display regions where it was stuck at a particular value, while thinning so that every jth iteration is stored, where j is of moderate size, would still give the identical 10% acceptance rate (approximately), but the trace plot would not display the same large regions where the chain had not moved from previous iterations. In some sense, thinning improves the distributions one obtains from an MCMC procedure, while at the same time reducing the dependence between the stored iterations. Of course, the more thinning one does the longer the chain must be run in order to obtain the same number of stored iterations.

Subjective inspection of autocorrelation plots of some of the parameters (not shown) from the pilot runs suggested that storing every 4th or 5th iteration

would be sufficient for removing the obvious correlation between iterations. However, I opted to store every 7th iteration, simply because it further reduced any worry about dependence issues, while at the same time meaning that the (τ_A, τ_B) values were likely to have moved between each stored iteration, reducing the 'stickyness' of trace plots. With the lowest observed acceptance rate being 0.15, storing every seventh iteration meant that, on average, the trace plots obtained should not display many regions where (τ_A, τ_B) appear stuck.

Burn-in was set to $5,000$ iterations. Note that, for the fixed component case, the (sensible) starting values for the parameters should mean that the process reaches stationarity long before the end of the burn-in period, and the burn-in here is, in fact, generous. A much larger number of iterations will be stored here, and the number chosen to be stored was $25,000$, meaning that $5,000$ burn-in iterations would take place, followed by $175,000$, of which $25,000$ would be stored after thinning. A run of this size takes approximately 8 hours on a Pentium 4 (3GHz) processor with 1Gb of RAM.

At the founder level, the analysis provides trace plots of τ_A, τ_B and τ, summarised by a posterior density estimate of the date of the migration/founding event (τ). An example of such output is shown in figure 6.4, where the output for founder v01 is displayed. This plot is for the case when the components were assumed to have a standard deviation of 0.1.

Such trace plots of quantities such as τ are informative as they often demonstrate cases where the founder does not always get assigned to a single component and this can often be clearly visible on the plot, with obvious jumps where the founder has been re-allocated to a different component as the sampler progresses. A more direct measure of component membership for a given

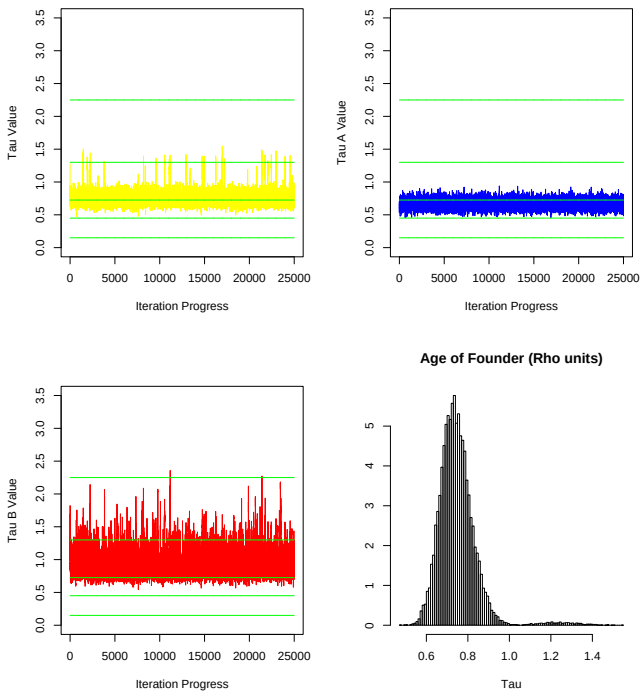

Figure 6.4: Trace plots of the three τ values for founder v01, together with a histogram estimate of the posterior density of τ.

founder is the z proportion matrix, which contains, for every founder, the proportion of the stored iterations that each founder was assigned to each component. The first few lines of such a return is shown in table 6.4.

Table 6.4: Extract of z proportion matrix. Note rounding means that some rows do not sum to one.

Founder	Comp. 1	Comp. 2	Comp. 3	Comp. 4	Comp. 5
1	0.00	0.01	0.99	0.00	0.00
2	0.00	0.01	0.99	0.00	0.00
3	0.00	0.04	0.94	0.01	0.00
4	0.01	0.10	0.89	0.00	0.00
5	0.00	0.00	0.30	0.70	0.00
6	0.00	0.00	0.87	0.12	0.00
7	0.00	0.09	0.90	0.01	0.00
8	0.00	0.00	0.55	0.45	0.00
9	0.01	0.05	0.84	0.10	0.00
10	0.00	0.01	0.91	0.08	0.00

The proportion matrix, together with the trace and density plots provide useful information at the founder level. The z proportion matrix provides some measure of how strongly to believe that a given founder does belong to a given component. Similarly, some density plots are multi-modal, which addresses the issue posed earlier regarding what the investigator would wish to see when the data is uninformative.

Moving from the parameters at the level of single founders to those parameters and summaries that are global to the model, one obtains densities for

the p_i fractions, the probability that a random founder belongs to component i (Figure 6.5, left), the number of founders assigned to each component (Figure 6.5, right), and the proportion of the European sample assigned to each component (Figure 6.6).

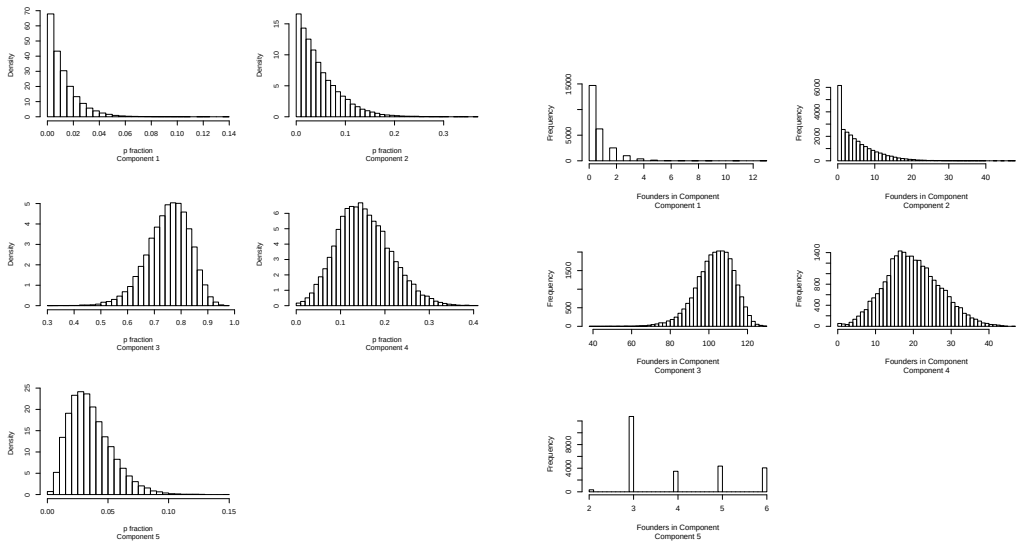

Figure 6.5: Posterior density plots for the p_i fractions (left), and the posterior distribution (unnormalised) of the number of founders in each component (right).

One of the major benefits of the fixed component case is the fact that the global densities obtained have a consistent meaning, e.g. if $\approx 90 - 110$ founders are being assigned to a given component and the component has a fixed mean and variance, then interpretation is relatively straightforward. Looking ahead, a density plot displaying identical features is much harder to interpret if it is the case that the component's location and spread are varying throughout the iterations.

I now wish to touch on a troublesome issue relating to prior choice of the

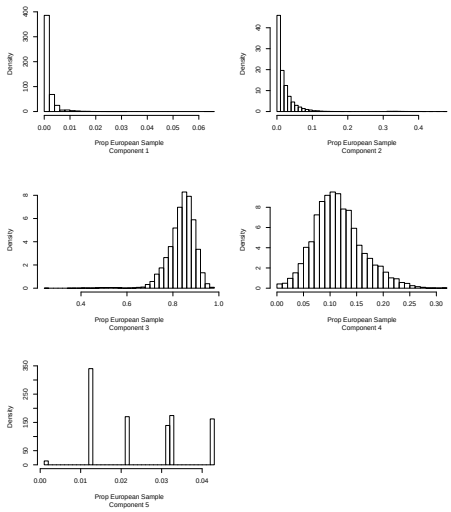

Figure 6.6: Posterior distribution of the proportion of the European sample in each component.

standard deviations of the components. In cases where a founder's ρ values are far apart, one would expect conflicting signals from the density plots for the τ values of such founders - the age of such a founder *should* be unclear. One would wish such uncertainty to be reflected in the density obtained, and in previous sections some effort has been directed to trying to justify the existence of a density plot that both respected the uncertainty in the age *and agreed to some extent with the process that is being assumed to have given rise to the data, that is, waves of migration periods.*

It is comforting to see density plots for some founders which display such densities. Figure 6.7 (left) shows trace and density plots for such a founder, u27, when components have an assumed standard deviation of 0.1. This founder has $\rho_A = 0.259, \rho_B = 1.615$, and the dating under the original founder analysis method gave a 50% CR of $(4451.64, 7238.21)$ which can be back-translated into ρ units to become $(0.221, 0.359)$. Although interpretation is not the ma-

jor issue here, it is worth noting that the 50% CR obtained from the basic founder analysis method lies almost centrally between two assumed periods of migration, and as such is *inconsistent with the underlying model of migration.*

The extended founder method gives older dates (reflecting the use of the additional information that $\rho_B = 1.615$, which suggests that this founder's age is uncertain, and possibly much older than the ρ_A data alone suggests), while the posterior density obtained is multi-modal, with most mass in the third component, although the founder has been assigned to an older component a smaller fraction of the time. The important point here is that the density is no longer inconsistent with the assumed underlying migration process - the modes in the density plot roughly correspond with two of the assumed migration periods, while the extended tail at the left-hand side of the plot represents the fact that this founder has been allocated to a more recent component an even smaller fraction of the time. This is reflected in the z proportion vector for this founder which displays $(0.002, 0.043, 0.794, 0.161, 0.000)$.

It is difficult to argue that this is not an improvement over the original founder method. The identical founder, but for the case when the component standard deviation is assumed to be 0.2, gives rise to a trace and density plot also shown in figure 6.7 (right), and a corresponding z proportion vector $(0.003, 0.028, 0.836, 0.132, 0.001)$. The issue to note is that the posterior density is not now multi-modal. Although this is perhaps a minor point (the densities are not inconsistent, and one can see that they both display similar information), but it should be clear that the prior choice of standard deviation of the components has to be done with some care. Note that the τ_A and τ_B densities seem less sensitive to changes in the component standard

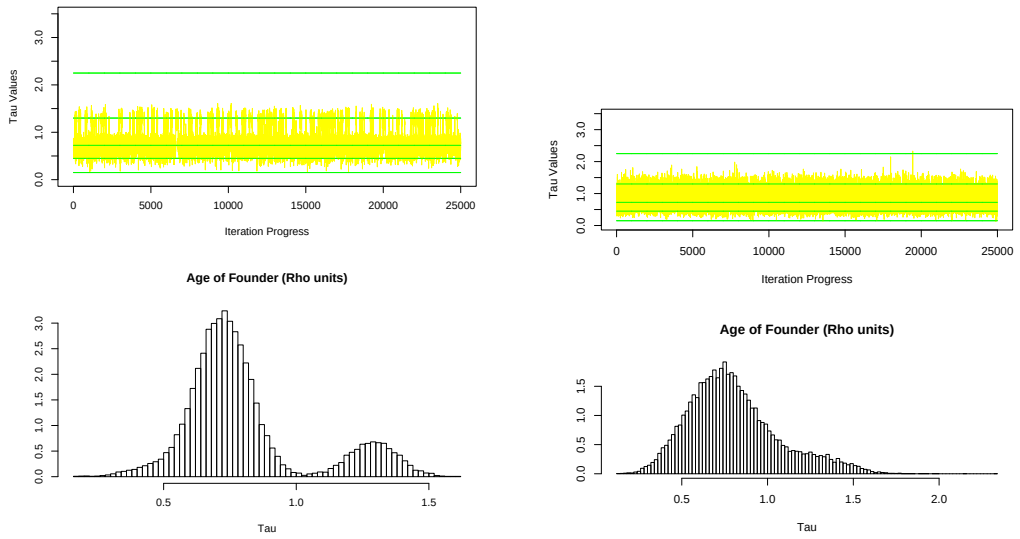

Figure 6.7: Trace and density plots for founder u27 when assuming the components have a standard deviation of 0.1 (left) and 0.2 (right).

deviations, particularly when ρ_A and ρ_B are not close together.

6.1.6 Some observations

The dating of founding/migration events now respects the assumed underlying model: every τ estimate draws on the location and spread of the components (currently fixed by the investigator). The original founder method based its estimate of the date of each founder on the data provided by that founder *alone*. In contrast, the extended method provides estimates which are not inconsistent with the assumed underlying migration process.

Multi-modal density plots appear for founders whose ρ_A and ρ_B values are far apart. The uncertainty is now represented in two ways: (i) in the migration period that the founder should be assigned to, and (ii) within a migration period, the uncertainty of that founder age *conditional on it belonging to a*

given migration period.

The extended method appears to be mixing well for most parameters. However this has been helped by appropriate provisional examination of acceptance rates for (τ_A, τ_B) and adequate thinning. It needs to be stated however that one parameter where mixing is potentially unsatisfactory for a very small selection of founders is the z matrix. The z matrix is a quantity for which mixing is quite difficult to evaluate. In some instances the ρ_A and ρ_B data are extremely informative, in the correct order, and lie close to an assumed peak of migration. In such cases one would expect the appropriate row of the z-matrix rarely to change. This would not imply poor mixing. The data *supports* a single allocation with movement to other components not expected very often.

In other instances the ρ data span a range between two migration periods, and inspection of the trace plots revealed that the sampler was jumping between two components, but only *very rarely*, with large numbers of iterations between each 'jump'. This suggests poor mixing and one cannot really trust the relative heights given to each period in the posterior density plots (although the shape within each period should be more reliable).

It turns out that, if one ensures sufficient overlap in the tails of the component distributions, the z matrix mixing greatly improves and no longer do trace plots display only rare jumping between components. This is the first fix for this mixing issue. A second fix exists which will briefly be discussed once the problem has been illustrated. Figure 6.8 demonstrates the problem for founder u21, which has $\rho_A = 1.70, \rho_B = 2.22$, an obvious problem case as these values lie between the assumed migration periods centred at 1.300 and 2.250 ρ units.

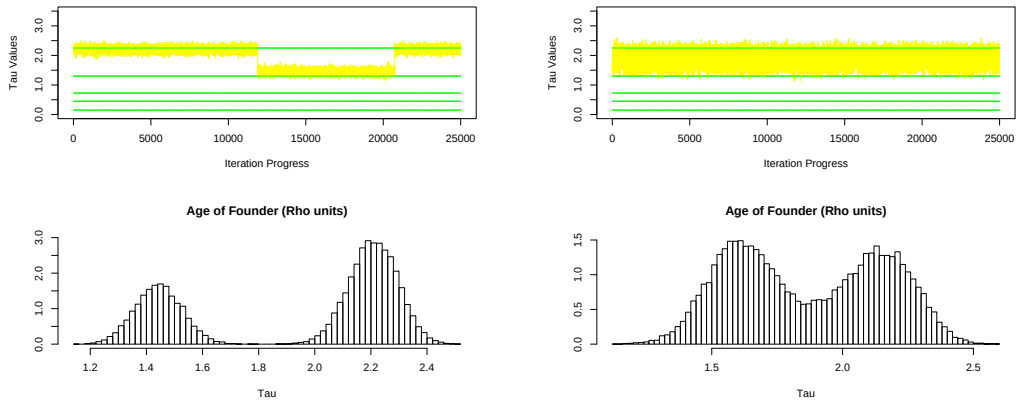

Figure 6.8: Trace and density plots for founder u21 when assuming the components have a standard deviation of 0.1 (left) and 0.2 (right). Note the two jumps in the trace plot when component membership changes when the standard deviation was set to 0.1.

The two densities do support slightly different age estimates. However there is no real contradiction. Dates that are closer to the assumed peaks of migration are supported more strongly when the standard deviation is set to 0.1 simply because the prior component distributions were set up to give much stronger support for these dates. When the standard deviation is increased to 0.2, the date estimates that lie firmly between the two assumed migration periods have more support, which allows the data (via the ρ estimates) to have a much larger influence on the dating, as it now conflicts less with the locations and spreads of the components; essentially the data are now supported by the prior component distributions instead of conflicting with them and the founder's migration date is no longer forced to jump uncomfortably between two periods when its ρ data support neither of them very strongly.

A mention is made here of the previous theoretical example given in Chapter

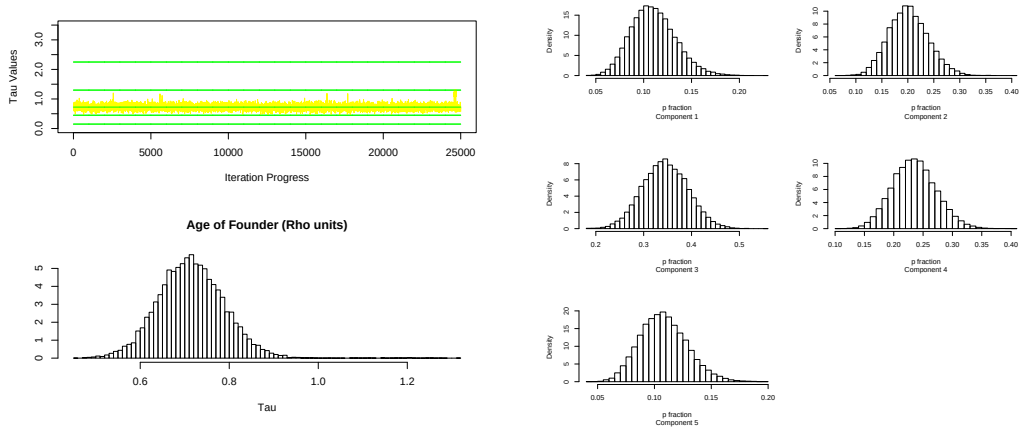

Figure 6.9: Trace and density plot for founder h01 when assuming the components have a standard deviation of 0.1 and prior alpha vector elements equal to 20 (left). Marginal p_i fractions when the elements in the α vector are increased to 20 (right).

4, namely the effect of changing the α parameters so that the p_i fractions (the mixing fractions) are more balanced. It was suggested that the α values could be increased so that the p_i fractions would not display negligible support for the components with least founders (particularly the oldest component, reflecting the thinning out of the tree). It turns out that such an increase in the values in the α vector also increases the number of jumps between components in many cases, even for components which display almost no jumps when α is 1, as figure 6.9 demonstrates for founder h01, when the standard deviation is 0.1, but the α vector was $(20, 20, 20, 20, 20)$, rather than $(1, 1, 1, 1, 1)$. This founder experienced no moves in the latter case.

There is one consequence of this, beyond that of a mere sensitivity analysis (how robust is the inference to changes in priors). Increasing the elements in the α vector makes the number of founders in each component more balanced

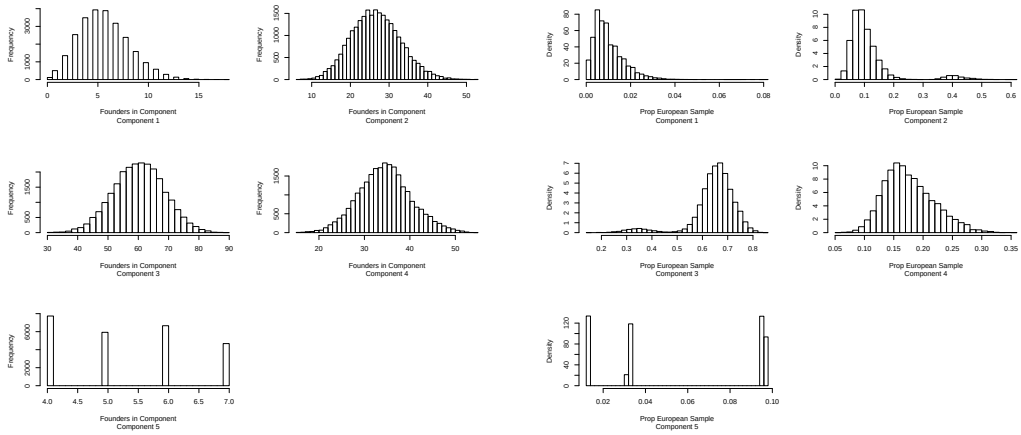

Figure 6.10: Number of founders in each component when assuming the components have a standard deviation of 0.1 and prior alpha vector elements equal 20 (left), together with the proportion of European lineages in each component (right).

by making the mixing fractions more balanced. This in turn has the effect of making our equivalent of the S_m fraction, the proportion of the European sample assigned to each component, display a troublesome feature: some of the components with largest n_a values now change membership, whereas before they rarely moved and contributed their n_a to a single component at almost every iteration. As a result the density plot of S_m displays some bi-modality. Figure 6.10 illustrates this.

With the previous exploratory runs undertaken, together with some discussion of the output and some important observations, one requires sensible prior choices for a 'final' run to be possible. The previous runs are important however not only for investigating performance, but also as a fair comparison with the conclusions of the original founder analysis work, with component locations set to match those used in the original work [26].

6.1.7 Prior elicitation

The process of selecting the appropriate locations and scales of the prior component distributions, together with suitable choices for the α hyperparameters is one which ideally should be guided by expert opinion in the subject area. Prof. M. Richards (University of Leeds) provided his opinions on some features of the extended method. This section would not be meaningful without his input: the informative priors that were based on his input allow this run of the analysis to be a proper data analysis run, where conclusions can be made and some interpretation made beyond those purely of a statistical nature.

After a day of discussing the extended founder analysis method, Prof. Richards' view on some features of the model became clear. The issue of bimodal densities for founder migration time estimates was troublesome, as interpretation was more difficult under the new method. The possibility of bimodal densities for the proportion of the European sample belonging to each component would be a difficult concept to interpret/report within the archaeogenetics community.

It is perhaps more interesting to report his views on the α parameters, arguably the most difficult hyperparameters to select and justify. The view emerged that this parameter is indeed important, but deciding *a priori* on its value was not judged as the best approach. Instead, the view emerged that the model should be analysed for a few values of α and any notable differences in conclusions openly reported. Selecting some (average) minimal number of founders that should be allocated to each component and fixing α to ensure this was too artificial. It is likely to be the case that very few founders originate from the oldest component so 'forcing' more into it was

unsatisfactory. The conclusion reached was to keep the components of the α vector equal, but explore analyses where this value was changed. If any judgement was to be made on minimal requirements for a single component it should not be based on number of founders (but perhaps on the average proportion of the European sample assigned to a component - since this quantity reflects the raw sequence level of the data, and not the founder level data).

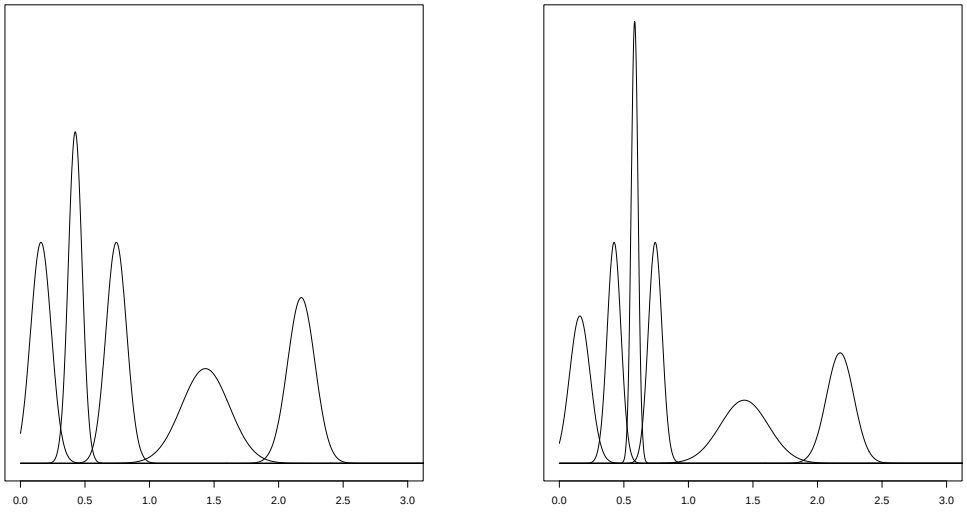

Figure 6.11: Prior component distributions (set 1), as proposed by Professor Martin Richards (left). Prior component distributions (set 2, with an additional period) (right).

Prof. Richards gave two models, one with five components, a second with six components. The first model had components centred at 3000, 8000, 14000, 27000, 41000 YBP, representing updated time estimates of the same periods described in the original founder work. It is notable that the latest component is now centred at a more recent time than previously, which should make

more founder ρ estimates consistent with it. A second set of times included an additional period being inserted at 11000 YBP. The standard deviations of these components was chosen by Prof. Richards, giving rise to the plots shown in figure 6.11. The analysis that follows focuses on the five component model, as the six component model offered no additional insight into the method's performance.

6.1.8 Five component model

The means of the components were set to 0.159, 0.425, 0.743, 1.433, 2.176 (by scaling the times in the previous section by a recent calibration of 18845 years per ρ unit in the first hypervariable segment of the control region [64]). $5,000$ iterations of burn-in were set, $20,000$ iterations were to be stored, with every 7th iteration stored (thinning). The elements of the α vector were set initially to 1. The output for a few founder sequence types is discussed below, which display some important features of the model. Interpretation of the results in terms of the prehistory of Europe is outside the scope of this work.

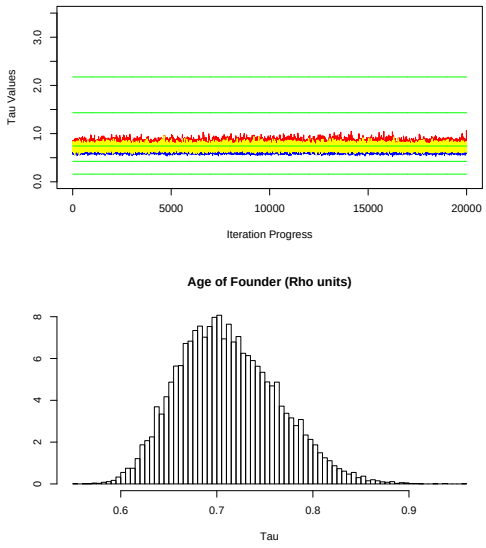

Figure 6.12: h00 trace and density plots of τ.

Founder sequence type h00 is an interesting case ($n_a = 855, n_b = 1017, \rho_A = 0.621, \rho_B = 0.646$), by far the most frequent founder cluster in Europe. It provides very informative ρ data that lie close to an assumed component (component 3). One would hope that the output for this founder would not

be bimodal and that it would be reasonably well defined close to component 3. This is indeed the case (figure 6.12).

Founder u22 provides a contrasting scenario ($n_a = 28, n_b = 1296, \rho_A = 0.964, \rho_B = 2.972$). It has very uninformative ρ data that span multiple components. One would hope that the output for this founder would display some considerable uncertainty while respecting the assumed migration model. This is indeed shown in the output (figure 6.13). This sort of output, together with z proportion vector (0.00000, 0.00020, 0.36295, 0.56085, 0.07600) demonstrates the tricky interpretation that needs to be undertaken for some founders. However, after some discussion it was generally accepted that in cases of large uncertainty, such plots are indeed desirable, and more honest than the dating obtained by the original founder method.

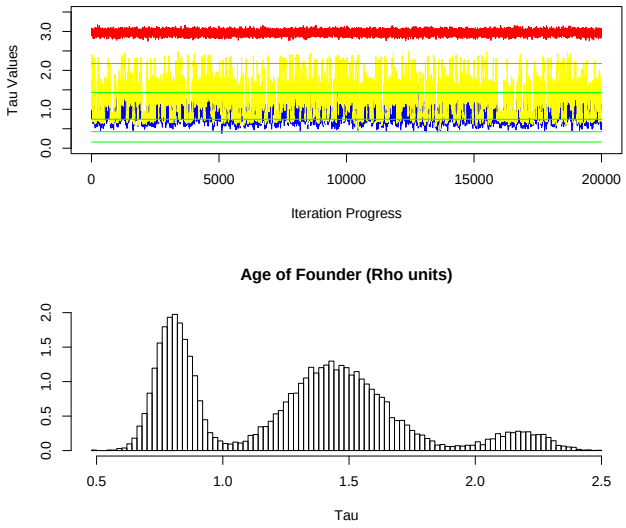

Figure 6.13: u22 trace and density plot.

Similarly, u21 provides another multimodal plot (figure 6.14). The ρ data

for this founder $n_a = 54, n_b = 278, \rho_A = 1.70, \rho_B = 2.22$ support both the fourth and fifth components, and the z proportion vector confirms it indeed spends considerable time in both of these (0.00000, 0.00000, 0.00000, 0.62115, 0.37885). It is also satisfying (and important to note) that the peaks of the bimodal density obtained for this founder age are shifted relative to the prior densities. This is a good feature of the model as it shows that, in cases where the data do *not* agree with the component means, but are informative as much as indicating which of the components the founder is likely to lie *between*, the data can define the peaks and the posterior density does not simply resemble the prior.

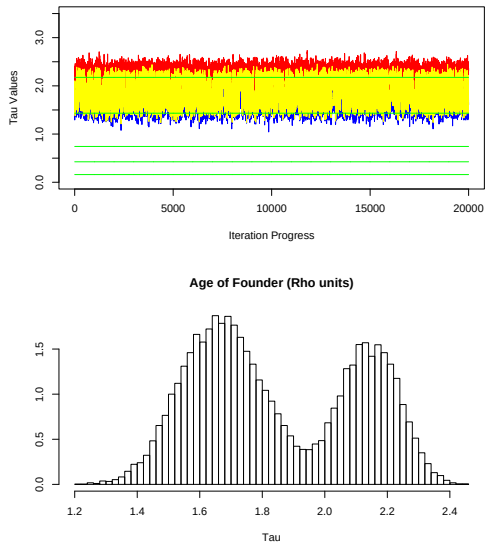

Figure 6.14: u21 trace and density plot.

Founder w01 ($n_a = 38, n_b = 73, \rho_A = 0.82, \rho_B = 1.14$) is another very interesting case, with data that generally support the third component with some possible minor support for the fourth component (figure 6.15). The

density plot obtained very nicely displays what the ρ data suggest, while the relevant row for the z proportion vector (0.00000, 0.00015, 0.70600, 0.29375, 0.00010) confirms this.

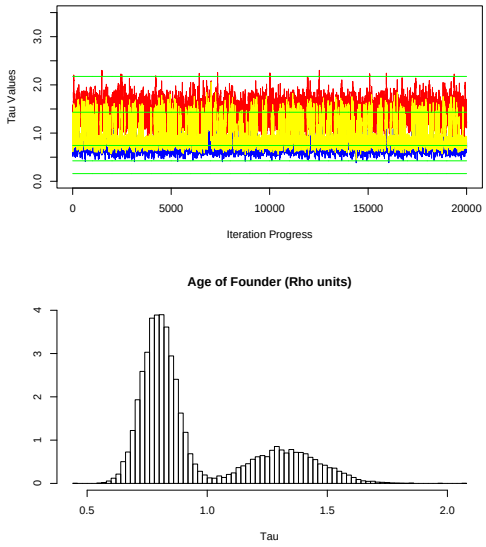

Figure 6.15: w01 trace and density plot.

It is worth noting that in the cases seen so far the mixing generally appears good and no extremely rare jumping between components of the type displayed earlier (for founder u21) is apparent. Founder v01 further demonstrates this with data that are very informative ($n_a = 127, n_b = 134, \rho_A = 0.6771, \rho_B = 0.6791$), yet this founder is still occasionally assigned to other components, with a z proportion vector of (0.00000, 0.00005, 0.99110, 0.00885, 0.00000) (figure 6.16).

It is at this point that the reader may be starting to see one of the major problems: the third component has considerable support for most founders due to the fact that many founders have ρ data that span across it. This is

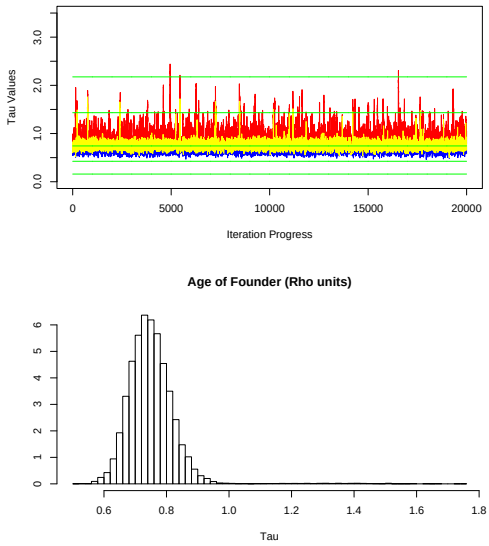

Figure 6.16: v01 trace and density plot.

problematic as it results in a very large p_i fraction for the third component, which has the effect of putting founder sequences whose ρ data are very uninformative in this component more often than one would perhaps wish (recall that the p_i fraction is, on average, larger for those components with the largest number of members).

To see founders which are assigned to the most recent components it is best to look at the runs which have increased α elements. It was decided to do a run with α being a vector of elements all of value 20. Figure 6.17 shows the resulting trace and density plots for founder j00, which has substantial support for the more recent components, as indicated in the original founder paper ($n_a = 172, n_b = 382, \rho_A = 0.43, \rho_B = 1.59$).

It is interesting to inspect the posterior probability plots of the number of founders assigned to each component and the proportion of the European

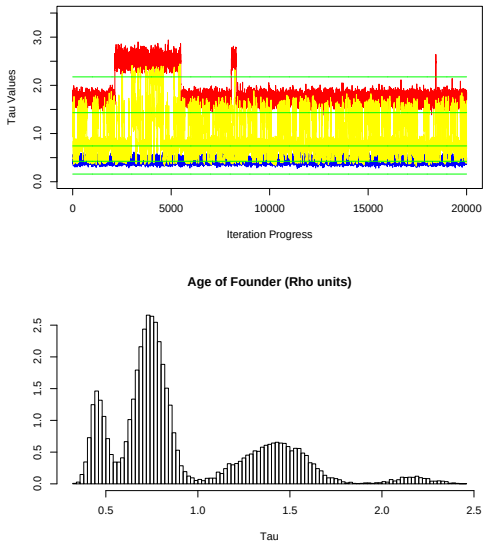

Figure 6.17: j00 trace and density plots when α is 20.

sample derived from each of the migration periods for the two α cases (figures 6.18 and 6.19). They illustrate what happens as the elements of the α vector are increased. It is clear that the earliest and latest components see large increases in the number of founders assigned to them as α is increased. Figure 6.20 demonstrates why this is happening by showing the marginal posterior densities of the p_i fractions. No attempt is made here to suggest the 'correct' value of α to select. This parameter is perhaps the one that requires most input from the subject expert; selection of it involves much deeper consideration of the underlying process, together with an understanding of the implications of the death process on lineages in the phylogeny.

Note that figure 6.19 does *not* display the bimodality properties seen earlier when using the dates very similar to that of the original founder method. The large clusters (in terms of n_a values) are not moving around as much under

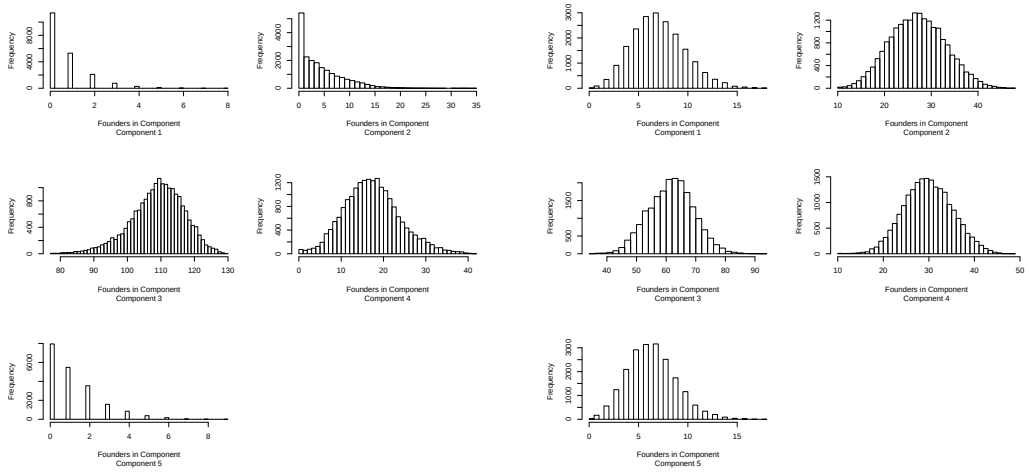

Figure 6.18: (Unnormalised) Posterior probability distribution of the number of founders assigned to each component, when α was 1 (left), and 20 (right).

the new migration dates provided by Prof. Richards. This is in many ways satisfactory from a reporting of conclusions perspective but it is important to recall that the proportion of the European sample assigned to each component is a potentially unstable quantity when large founder clusters exist which are not consistently assigned to the same component at all iterations of the process.

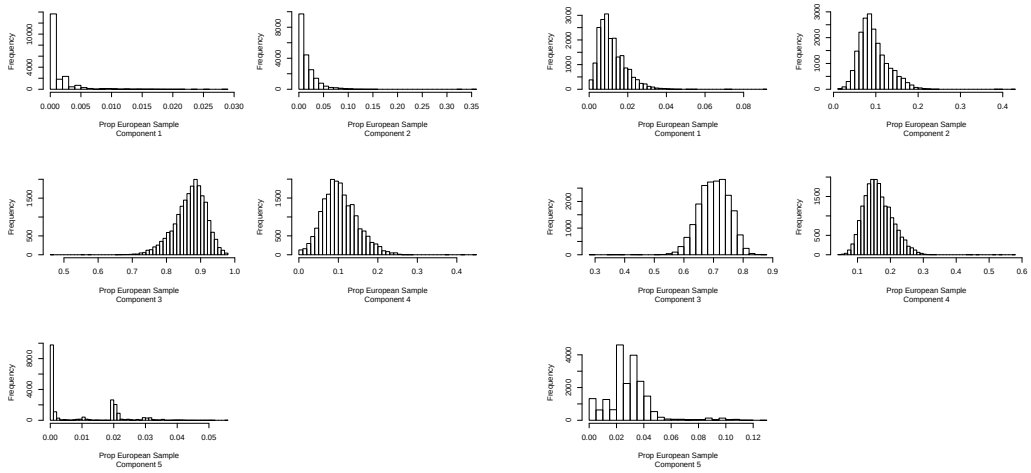

Figure 6.19: (Unnormalised) Posterior probability distribution of the proportion of the European sample assigned to each component, when α was 1 (left), and 20 (right).

6.1.9 Summary and conclusions

The fixed components model provides an improved method of dating founder sequences using the assumption of a connected star tree for each founder, to model the migration dates on the connecting edges.

The method has all the strengths of the original founder method. That is, it allows the inclusion of an expert's beliefs on the locations of the major migration periods, provides dating of each founder cluster and provides an equivalent estimator of the S_m fraction, the proportion of the European sample assigned to each migration period. In addition, though, many of the weaknesses and problems present in the original method are overcome by this extended method.

By using the idea of a connected star tree, one is no longer assuming that migration/founding events coincide with a specific node (the founder sequence

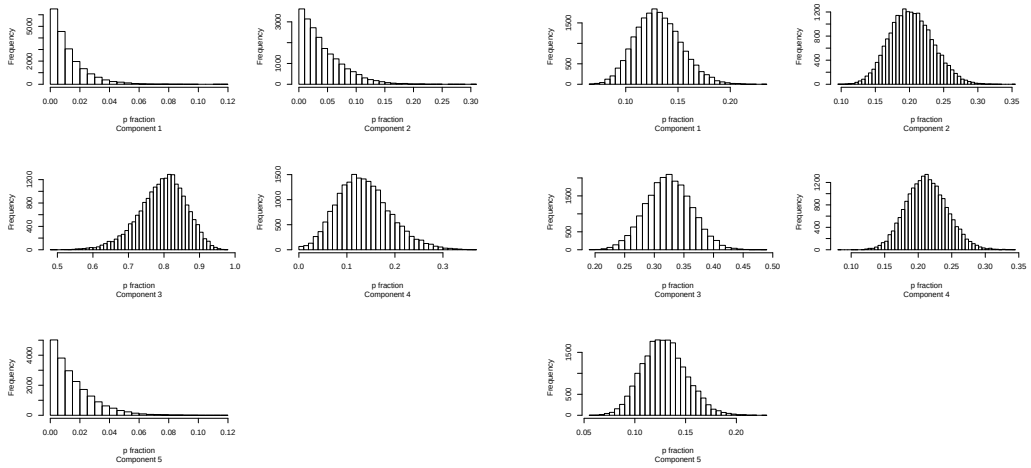

Figure 6.20: (Unnormalised) Marginal p_i fraction densities, when α was 1 (left), and 20 (right).

type) on the reconstructed phylogeny. In essence, the extremely strong assumption that every migration event results in the *instantaneous* expansion of that sequence in Europe is relaxed, and one is now adequately allowing for the migration events to occur anywhere along the entire depth of the reconstructed tree, not merely at the finite number of points where a node of the tree is located. The reasonable criteria for identifying possible founder sequences still remain in effect. However, they are now being used to identify *edges* where a migration event has occurred and the uncertainty in the location of the event on that given edge is now being modelled appropriately.

The mixture modelling approach provides dating that is fully consistent with the assumed underlying migration process, founder sequences which do not have informative data being allowed to be assigned to different components over the MCMC steps, which results in such components being uninformatively dated over multiple migration periods. This feature is indeed desirable: cases where the assignment of a sequence to a migration period is in doubt

should be reflected in the output obtained instead of reported by a misleading estimate that may not even represent a period in which the expert believes any migration was taking place.

The strength with which one should believe a founder to belong to a given component can be readily estimated by the relevant row of the z proportion matrix, while the assignments of founders to components at each iteration provides the necessary information to estimate the proportion of the European sample assigned to each component. The quantity S_m of the original method can be thought of as a single point estimate/summary of the marginal posterior distributions obtained in the extended method. It has been shown that this distribution can be multimodal, a fact that the single S_m estimate cannot capture. The *distribution* of the proportion of the European sample assigned to each component is now being estimated.

The method allows the interested investigator to use his/her own prior choices for the distributions of the components and gives the dating estimates and assignments conditional on those prior choices. This, of course, leads to conclusions which depend on informative prior choices, but this is in fact a strength of the method, provided one is prepared to make such assumptions (which many archaeologists and palaeodemographers *are* prepared to do). On the issue of prior selection, the interesting consequence of varying the α vector was demonstrated. This vector poses interesting questions for both the statistician and the expert. On the one hand, the idea of setting the α vector equal to a vector whose elements are all one is appealing as it allows the data to determine the likely mixing fractions. On the other hand, one knows that lineages will be lost for reasons such as genetic drift, while it is also well understood that fewer lineages will be available for designation

as being a founder sequences type as one goes further back into the past (towards the root of the tree). In such instances one does need to question whether increasing the elements of the α vector to keep all the p_i fractions 'non-negligible' is desirable. This is a difficult issue and one for which no definite answer is provided. It could even be argued that an α vector whose elements were not all equal could be in some sense optimal, but defining and justifying such a choice is difficult.

6.2 Extended founder analysis - full

The remaining issue which is covered in this small penultimate section is the extension of the method to allow the identified founder sequences to define the components. The theory for this section has already been covered and it has been demonstrated in a previous chapter that *in the presence of informative τ data* this wish can indeed be satisfied.

It is stated here from the outset that this wish is one which cannot be realistically achieved with the dataset at hand. It turns out that to define components to any reasonable level of resolution, a non-negligible number of founder sequences needs to be assigned to each component. It has already been demonstrated in the previous chapter that one component, the third (fixed) component takes in the majority of the founder sequences, while the ρ data do not regularly support any component whose date lies $> 35,000$ YBP, or $< 5,000$ YBP. This is unfortunate - having only a very small number of founders which plausibly originate from very recent migration periods or very old migration periods means that these components are unable to be clearly defined. The problem is made more troublesome by the fact that the ρ data for many founders span a large range of the allowable dates, so even

in the cases where the ρ values appear to offer some limited support for very recent or old dating, it is not uncommon for the data to support a wide range of dates. As a result, it is very rare to see a founder assigned *exclusively* to very recent or very late components.

It is important, however, to demonstrate that the estimation of migration periods using the idea of founder sequences is an extremely difficult problem, and that the difficulties arise because the number of founder sequence types will always be very low relative to the number of raw sequences used in the phylogeny reconstruction. The result of this is that the number of founder sequence types in many of the components is likely to be extremely low. These are problems which would affect *any* method which attempts to identify founder sequences.

6.2.1 Extended founder analysis - estimating component means and variances

The extended founder model is now supplemented with the code that estimates the component means and variances at every iteration, in addition to all the parameters which were sampled in section 6.1.3. If the data were informative enough, one would hope that it would be possible to use the posterior density estimates as an indication of the locations of the major periods of migration. Extensive simulations for various parameter sets were undertaken. In this section the results of a small number are reported, which demonstrate some of the posterior distributions obtained under the method, but more importantly show the limitations with the method that arise due to the large uncertainty in the date estimates of each founder and the unfortunate problem of empty components.

The ξ hyperparameter was set to match values used in the original founder work [26, page 1256], for dates that roughly correspond to 3000, 9000, 14500, 26000, 45000 YBP. The choice of hyperparameters on the variance gives an expected prior variance of ≈ 0.05, for each component. In summary, $\xi = (0.15, 0.45, 0.725, 1.3, 2.25)$, $\nu = (4, 4, 4, 4, 4)$, $S^2 = (0.1, 0.1, 0.1, 0.1, 0.1)$, $\kappa = (1, 1, 1, 1, 1)$, and $\alpha = (1, 1, 1, 1, 1)$.

The first analysis that is reported represents an ideal case which one would wish to investigate if the data allowed informative posterior distributions to be obtained. The code was run for $100,000$ iterations with $2,000$ burn-in (inspection of multiple trace plots that involved no burn-in appeared to indicate that this was more than adequate). Of the $100,000$ iterations, $10,000$ were stored after thinning (every 10^{th} iteration was stored).

Focusing on the level of components (rather than individual founders), inspection of the trace plot of component means, figure 6.21, at first glance, suggests that some separation into components has occurred.

Inspection of figure 6.21 in isolation possibly could lead to the view that a late component and perhaps even an early component have been determined by the model. This view is, however, unfounded: the posterior distributions of the component means for the latest and earliest components are seen to be estimated very often from the prior *alone*. In the run reported here, the z proportion matrix indicates that only five founders are placed in the latest component in more than 5% of the stored iterations. Further, only founders u21 and u22 are allocated to the latest component in over 10% of the iterations (82% and 11%, respectively. Note that u21 has a lower ρ_B value than u22, yet is assigned to this latest component more often - this is due to the ρ_A value of u21 being much larger than that of u22, at 1.7

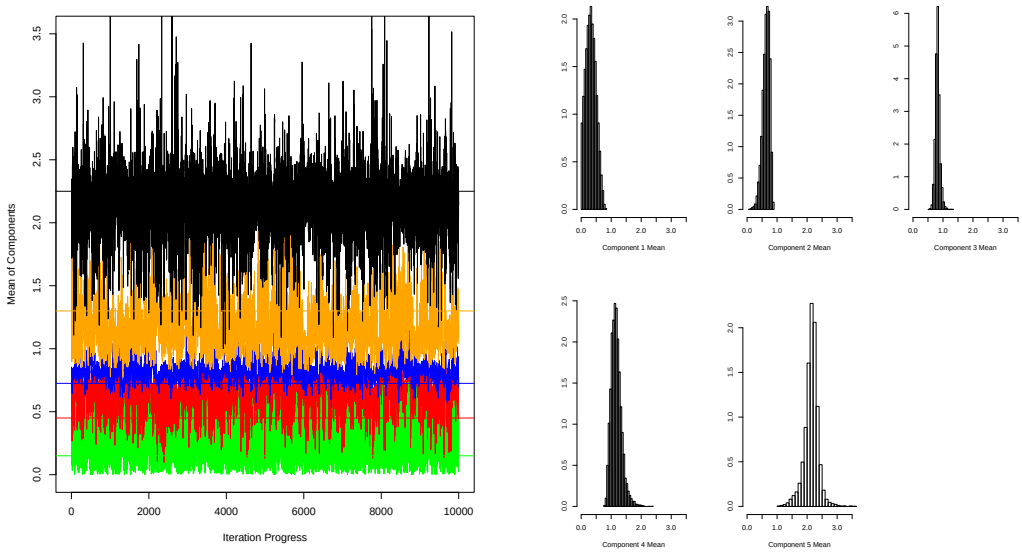

Figure 6.21: Trace plot of component means in the 5-component model. Component prior means are shown as solid lines. (left), together with the posterior densities of the component means (right).

compared to u22's 0.96). In approximately a tenth of the stored iterations, however, this later component is empty, and it is extremely rare to see more than five founders allocated to it (in this simulation $> 97\%$ of iterations had five or fewer founders in this component). Similarly, for the earliest component, in $\sim 83\%$ of iterations the number of founders allocated to the earliest component was five or fewer, and indeed the component was empty $\sim 34\%$ of the time.

The issue of empty or almost empty components is a troublesome one, and one for which no easy solution is available. The founder data are particularly bad for defining late and early components because very few founders' ρ data support *only* very late or very early dates; in most cases the range of ρ values supported is large. A section discussing the issue of estimating the number of

components will follow shortly, but for now, one simply notes that the founder ρ data does not lend itself well to establishing early or late components because these components are often near-empty, and the prior values are in many cases contributing as much weight to the posterior inference as the data are. It is also worth recalling here that fewer founder sequence types (on average) can originate from the oldest periods, as a natural consequence of the tree structure and the *decreasing number of nodes present as one moves towards the root of the tree.* Of course, there is always the alternative view that these components are useful as they mop up the 'outliers'/extreme founder sequence types, allowing the other components to be better defined. The main point I wish to make here is that the earliest and latest components are largely being defined by only a few founder sequence types that are competing with the prior, and by competing one is admitting that the prior is often playing a *crucial* role in the posterior estimate of the parameter, which is far from ideal.

A further and perhaps less obvious problem is when multiple components are essentially the same. Further inspection of figure 6.21 reveals that the posterior means of multiple components are *extremely* similar. The problem with this is simply that one may not know whether a model is identifying multiple components, each of which has its own members, or if a single component is all that is required to contain these founder sequence types. It is difficult to argue against the view that components 2 and 3 appear extremely similar since there is a large overlap in the posterior distributions of their component means. Components 3 and 4 appear better defined with less overlap in posterior mean estimates, but one could still argue that there is the possibility that a single component could do instead of three separate components.

In light of the observations above, the method was re-run after removing the earliest and latest components to look at the effect of this exclusion on the component means one obtains. In addition, some checks were taken to ensure that the posterior distributions obtained were robust when the prior values on the component mean were varied. Provided that the prior means were varied only within the range $(0.25, 1.5)$, then the posterior densities obtained were insensitive to all such sensible choices. Some output from the run when the earliest and latest components are removed and all other hyperparameters are as before is shown in figure 6.22.

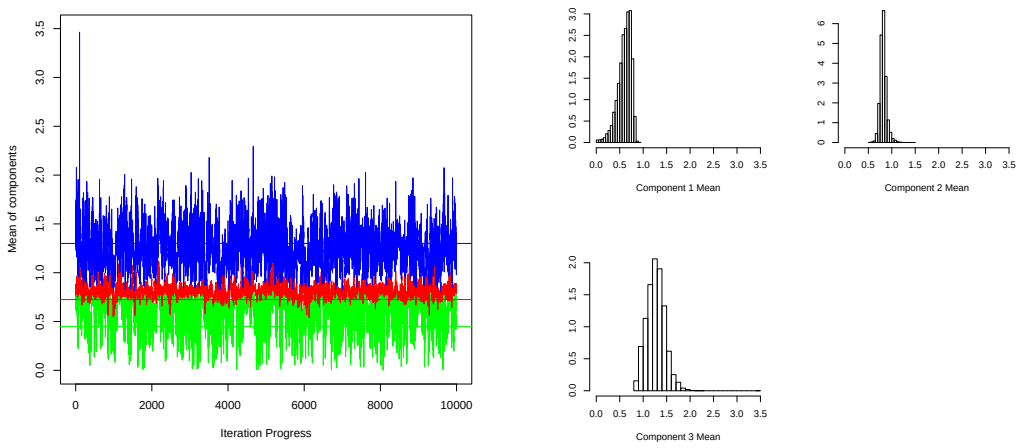

Figure 6.22: Trace plot of component means in the three-component model. Component prior means are shown as solid lines (left), together with the posterior densities of the component means (right).

Choosing three components with one having a very large prior mean (e.g. > 2), however, did result in behaviour similar to that of the five-component case reported above, with this one component often being empty and only containing a very few founder sequence types a minority of the time. In essence, very late components are either empty and estimated by the prior,

or near empty but with some founders (such as u21) in them, with almost every other founder sequence type being consistently allocated to more recent components, at around $1.2 - 1.5$ ρ units. It is still apparent in figure 6.22 that components 1 and 2 have considerable overlap. It is also important to note that the variability in the mean value of component 2 is less than that of component 1 partly because it is constrained between the current values of the means of components 1 and 3 at every iteration. While some formal methods do exist to try and estimate the number of components, such as reversible jump MCMC ([66] and [67]), this method (or alternative simpler methods using ideas such as those in [68], which were explored in my research but were largely unsuccessful in this application) will not be presented in this thesis. Instead, more subjective and intuitive arguments are used in what follows to demonstrate that such formal methods are unnecessary and would not help much in the problem at hand due to the specific nature of the data used.

Recall that the initial model with five components was rejected because the latest and earliest components were often empty, or else contained only a very small fraction of the founder sequence types. Removing these components gave rise to a three-component model with components that rarely emptied, but components 1 and 2 still share similar posterior means, which could suggest that they should be merged into a single component. Although this argument is rather heuristic, it is further strengthened when one considers the entries in the z proportion matrix for these two components.

It is seen for this simulation (and repeated in all others) that it is uncommon for any founder sequence type to be allocated to either component 1 or component 2 *exclusively*, and in fact, for the simulation presented, 110 founder

sequence types (of the 134) were assigned to the earliest two components in over 90% of the stored iterations, and, of these 110 founders, the number of founders which were allocated to both of the early components in almost equal proportions was 89 (if 0.2 is used as the maximum difference between the allocation proportions for a founder sequence type) or 46 (if 0.1 is used instead). What is happening here is that the model essentially is not treating these two components as being different and the founders which reside in them are spending relatively equal proportions of their time moving between them. This suggests that one may wish to remove one of these components and go down to a two-component model.

It is at this time that one may wish to consider what this is saying in a broader perspective. The extended model, which *correctly* tries to model the fact that migration events occur on edges that connect nodes of the tree, is unable to reliably infer a non-trivial number of components. I would argue that this is not a weakness of the method, but is in fact little more than an honest reflection of how vague the data are. The estimation of the components is based on τ estimates which are changing at every iteration. This alone makes it extremely difficult to guess where the majority of the τ values for each founder sequence type lie in the space of allowable τ values. This idea and the problems that result from it are explored in the discussion section to follow. For now, I simply go on to present the reduced two-component model and show that it leads to components being identified which agree with what is intuitively suggested from the previous three-component case, namely that the extended founder model does appear to be able to identify two major components centred at around 0.8 and 1.2 ρ units.

When moving down to just two components, the z proportion matrix starts

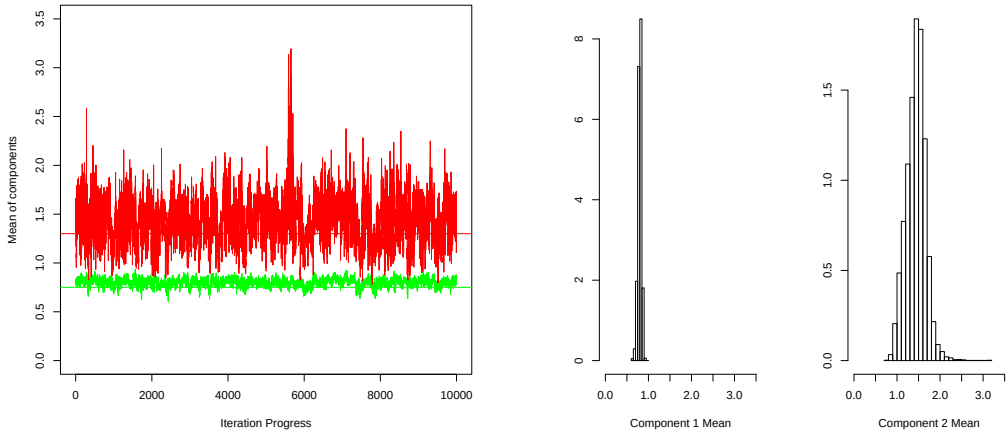

Figure 6.23: Trace plot of component means in the two-component model. Component prior means are shown as solid lines (left), together with the posterior density estimates of the component means (right).

to display rows which indicate that the founder sequence types are being relatively unambiguously assigned to a single component, although movement to the other component is visible for every founder sequence type (none of the z proportion matrix entries is zero), suggesting reasonable mixing. Out of the 134 founder sequence types, 100 are assigned to a single component $> 90\%$ of the time (which is stable across runs), and this number rises to 122 if using $> 80\%$ as the cut-off figure. Although this is not in any way a formal method to justify a two-component model, it is difficult to argue against the idea that, in this case, the components are being *defined* by the data. Both components are non-empty (although the most recent component has the larger proportion of the founders in it, approximately 85%), the majority of the founder sequence types are spending a large proportion of the time in a single component, and the posterior estimates obtained were seen to be insensitive to prior hyperparameter values on component means

(not shown). Further, reasonable mixing is demonstrated by the fact that the founder sequence types are moving between the components and are not simply stuck in a single component across all iterations.

Table 6.5 displays the mean values of the posterior component means for the previously discussed runs, with the addition of the single component model. The single component case is rather uninteresting from both a modelling and an interpretation perspective, but it is included for completeness.

Table 6.5: Mean values of the posterior component means for each model considered. Standard deviations of the posterior mean are given in brackets.

Comps.	Mean 1	Mean 2	Mean 3	Mean 4	Mean 5
5	0.32(0.324)	0.63(0.119)	0.82(0.075)	1.17(0.177)	2.14(0.238)
3		0.60(0.144)	0.81(0.070)	1.27(0.193)	
2		0.79(0.041)		1.44(0.225)	
1			0.86(0.037)		

Of course, all the models above provide posterior distributions for the component variances. However, these provide limited additional insight. The posterior distributions of components' σ^2 for the latest and earliest components (which were either empty or close to being empty) display distributions which closely resemble the prior, with some extra variability often visible due to the fact that the handful of founder sequences that often make it into these components have τ values which can sometimes be quite far from the component mean. An example of this is the five-component model's oldest component which often only has a few founders in it, whose τ values can be

very varied and are changing at every iteration. Components which contain particularly large numbers of founders are seen to have posterior distributions with support for smaller standard deviations, and this is particularly true for components which lie between other components and do not see the same variability in the τ values of their members as do those on the extreme ends of the ρ scale. An example of the posterior distributions for the standard deviations of the components in the two-component model case is shown in figure 6.24.

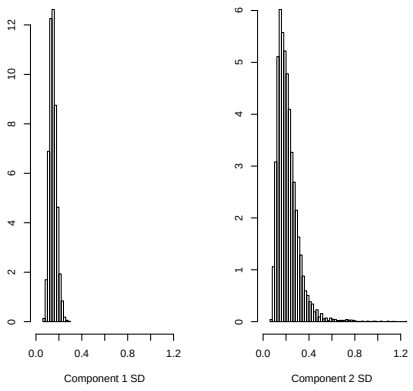

Figure 6.24: Posterior distribution of σ for each component in the two-component model.

Additionally, one may be interested not just in the posterior distributions of the parameters of the components, but in parameters at the founder sequence type level. The same posterior distributions (such as the τ values) are estimated for every founder as were estimated in the more restricted case when components were assumed to be fixed, as are the more global parameters such as the proportion of the European sample assigned to each of the components under each model considered. It is here that a problem lies with

the very general extended founder case that involves estimating the locations of components.

The problem is best demonstrated by a particular example, founder sequence type u22. This founder is unusual as it has ρ values of $(0.96, 2.97)$, and so is one of the few components with ρ data that obviously suggest that this founder sequence type could plausibly originate from an older migration period/component. This founder sequence type differs from u21 however (which also has a large ρ_B value) by having a ρ_A value which suggests the possibility that it originated not from an older migration period, but a more recent period. As such, u22 can be viewed as an interesting 'problem' case. It then becomes interesting to investigate the posterior distributions of the τ values of this founder sequence type under each of the models as the number of components changes.

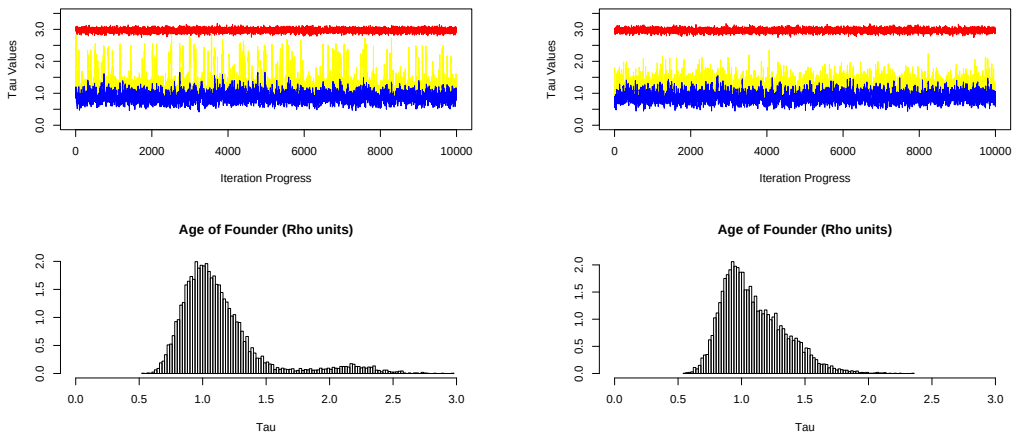

Figure 6.25: Posterior distribution of founder u22 and trace plot under the five-component model (left) and three-component model (right).

Figures 6.25 and 6.26 show the posterior distribution of τ for founder se-

quence type u22 under each of the considered models. Firstly note how the τ_A and τ_B estimates are almost unchanged across the models. It is very interesting to see the change that takes place in the τ estimates when moving from five to three components. Removal of the oldest component results in a clear visible change in the distribution of the τ density for this founder. It is clear that the five-component model results in some support for this founder having a τ value of perhaps ≈ 2.25 ρ units. The change from three to two components has a less dramatic effect, while, with a single component, it is seen that the τ value is very close to τ_A which defines a lower limit on the allowable τ's.

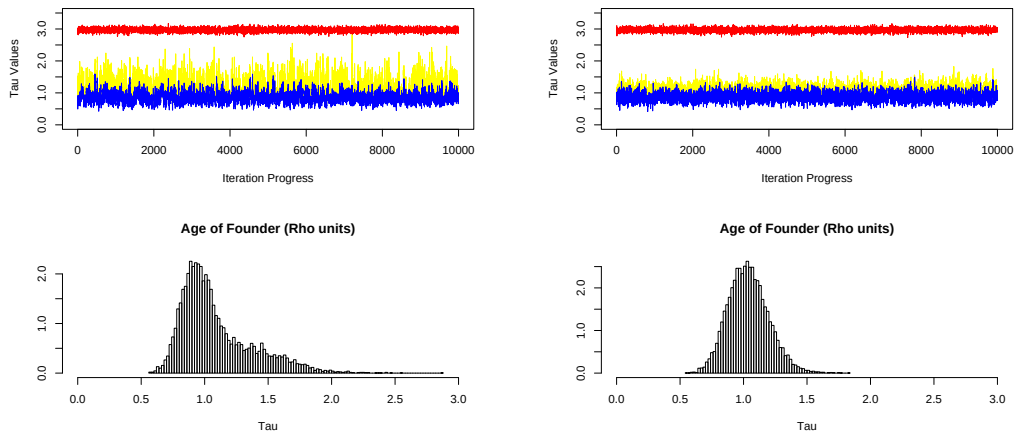

Figure 6.26: Posterior distribution of founder u22 and trace plot under the two-component model (left) and one-component model (right).

It might not be immediately obvious from the u22 case that this is in fact a problem. In the fixed mean and variance case, posterior distributions for the τ values nicely represent the distribution of the migration times of that founder, conditional on the *well defined* components. Quantities such as the z proportion vector had clear interpretations that allowed the investigator

to make statements such as "The fraction of the time founder sequence type X was assigned to component Y centred at Z ρ units was approximately...". Such interpretations are now much more difficult as the components are not so well defined, and in some instances, such as the u22 case, a component we would hope the method would allocate them to (e.g. a very late component) may not even be in the model as it had been removed for regularly containing only a few members!

Further, all of the problems of the fixed mean and variance case remain, particularly specification of the α hyperparameters. In summary, the more general method which attempts to estimate the locations and spreads of the components brings with it substantial extra uncertainty and difficult questions relating to the number of components, and, even deeper, the question about when is a component worth keeping remains.

Prof. Richards suggested (pers. comm.) that you would not want to remove a migration period just because it only had a trivial number of founders in it, especially if this meant the model was reduced to one which had few migration periods. This would especially be the case if fewer migration periods remained than the number of periods that are widely believed to have the *potential* for *some* migration (e.g. the LUP, MUP, EUP etc). An analogy can be made with the statistical idea of leaving terms in a regression model that are not found to be statistically significant. This idea goes against the idea of reducing a model down to what can be shown to be statistically significant, but allows expert views to guide a model, even when the statistical evidence does not necessarily demonstrate agreement with beliefs commonly held by experts in the area of work.

Accepting the expert view that a late (and early) migration period should

be left in the model, even if the statistical evidence is not strong in favour of this idea, one is led to ask the question "If a component is left in the model because the general expert opinion in the field supports it, but one does not have the informative data required to accurately infer such components, is it best to allow the model to make such an inference on component location and variance, or is it best, if accepting the expert view, to put all of one's faith in the expert opinion and simply allow them to define these components completely (i.e. the fixed mean and variance case)?".

The answer to this question is one about which I can only provide my opinion. It is my strong belief that the general extended founder method that estimates the component means and variances is likely to be unsuccessful while the number of founder sequence types is relatively small, and, more importantly, very uninformative. Even in the case of a perfectly resolved phylogeny, with perfect dating of all the nodes on the tree, the extended founder analysis method will always have problems defining components, simply because it honestly attempts to model the discrepancy between the date of the founder sequence type and the founder that was involved in the migration event.

The general full model may help refine expert scientists' beliefs in the location of some major migration periods, but not those of much older migrations for which very limited evidence exists based just on reconstructions from modern-day DNA sequences. The substantial uncertainty that exists in the dating of the nodes of the phylogeny, combined with the small number of founder sequence types that can be identified, is prohibitive to formal inferential methods, and I believe one must instead put trust in the expert scientist by allowing him/her to define the components of interest.

The inference that can be made for the fixed mean and variance extended founder analysis case is a step closer to giving scientists the tools they require to investigate the periods of migration of modern humans. The method allows informative statements to be made about parameters of interest while at the same time takes the method of founder analysis to the next level by removing the dating/conclusions that were inconsistent with the migration model and introducing connected star trees, which allows one to account for the fact that the migration event of interest does not in fact coincide with any event we can reconstruct reliably on a phylogenetic tree.

Chapter 7

Summary and discussion

7.1 Summary of the work undertaken

In this thesis I have investigated and extended founder analysis, a popular method for analysing modern-day DNA sequences with the aim to identify the sequences involved in migration events on a reconstructed phylogeny, and to date such migration events. Through simulation methods the method was shown to display a major weakness in that it was actually *not* estimating the time of any migration event of interest, and instead estimating a more recent time that corresponded to a coalescent event on the phylogeny. My model was built up appropriately to try to mimic the migration process that was assumed in the original founder analysis work, and it was seen that the problem identified remained throughout. Once this problem was identified it became apparent that it was one which was very real in the original dataset [26], with founder sequences having ρ estimates arising which suggested they could be dated close to the present day which was obviously a problem, and suggested the possibility of large bias in the dating.

The problem identified required the definition of a 'founder' to be revis-

ited, and the distinction between 'founder' and 'founder sequence type' being made. This distinction made it clear that the method put forward in the original work [26] could be improved upon by developing a method which was designed to estimate the date of founders, and not founder sequence types.

With the distinction between founder and founder sequence type made, the concept of a connected star tree was introduced, a simple form of phylogeny which contains a single edge that is assumed to carry a migration event of interest that one wishes to date. The complete phylogeny of the sample was then assumed to be composed of a set of connected star trees. This object, although still much simpler than a general bifurcating tree, allowed me to model the actual migration time of interest by bounding this event time between two nodes on the connected star tree which can be dated by a simple estimator.

The revisiting of what actually constituted a founder, together with the new concept of a connected star tree, formed the necessary foundations for an extension of the founder analysis method to be made. This extension was done by clearly defining parameters (τ_A, τ_B) which represented boundaries on each migration time of interest (and coincided with coalescent events on the tree), with additional parameters τ which represent the actual migration times one wished to estimate. Appropriate probability results were derived under a connected star-tree assumption which allowed one to formally describe the probability distribution of these parameters given the data (in the form of the ρ estimates).

The idea was developed and it was shown that it was possible to accurately estimate the posterior distribution of a single migration time on a connected star tree (the τ's) using the probability results derived. By using only a small

amount of extra data that is obtainable from any dataset that the original founder method could be applied to, the additional uncertainty in the founder migration time could be accounted for. Perhaps more importantly, the use of connected star trees with the new results removed the (inherent) assumption that founders/founding events coincide with founder sequence types. Such an assumption meant that the original founder method was in fact giving date estimates to founders that were too young, as the original method was actually trying to date the founder sequence type.

After showing that an improvement in the dating of a single founding event was possible with the use of connected star trees and the results derived, the idea was extended to allow the set of τ values from *every* founder to form the input to a mixture model. A hierarchical mixture model was built up with components which represented periods of migration. It was shown through simulation of τ data (ignoring the (τ_A, τ_B) level of the model) that, in the presence of informative τ data, such a mixture model would allow sensible inferences to be made about the components.

The dataset used in the original analysis of Richards et al. [26] was revisited and the additional data required (n_b, ρ_B) were calculated from the original diagrams and network reconstructions (Prof. Richards, pers. comm.). The extended founder analysis method with fixed component means and variances was used to re-analyse the (supplemented) dataset. The method was seen to remove some of the contradictions seen in the conclusions of the original founder method; the estimates of founder migration times were shown to be consistent with the underlying migration model which was assumed to have given rise to the data, whereas the original method was not.

Further, it was demonstrated that the extended method presented in this

thesis provided additional posterior density estimates of parameters that had no equivalent in the original founder work. Of particular interest was the posterior probability that a given founder sequence was allocated to each component. This vector provides some insight into the migration history of sequences which do not nicely belong to a single component. Until now, such an inference has not been possible in such an objective manner, and, within the extended founder method, this is returned routinely for every founder.

The proportion of the European sample assigned to each component was considered in this thesis, and it was shown that, under the extended founder analysis method, the *distribution* of this estimator can be multimodal and is very sensitive to large founder clusters which are not unambiguously assigned to a single component. How informative the average of this distribution is, as used in the original founder method, after such a discovery, is open to question.

One of the strengths of the extended founder method is that the model is flexible enough to allow expert scientists to incorporate their views through prior choices, and provides parameter estimates that arguably represent natural quantities that have an interpretation in terms of the migration history of a sample. In fact, one of the best features of the model presented is that it is not merely an artificial model with parameters which do not have an interpretation that the expert scientist would understand. Even the z matrix, which is typically a nuisance parameter in the context of mixture models, has a natural interpretation in terms of the probability a given founder originated from each migration period.

The α hyperparameter vector is more difficult than usual to specify in this context. The unique nature of founder sequence types which arise on the

nodes of a phylogeny result in this hyperparameter being of increased importance. It was demonstrated that assigning the α vector to be suitably vague was not a trivial matter since it is not clear that the number of founder sequence types in each component should be, *a priori*, assumed to be similar. It was in fact argued that the pool of founder sequence types is a non-increasing quantity as one goes from the tips of the tree (present day) to the root, and as a result one would expect (on average) a very old component to have fewer founder sequence types in it than a very recent component. This issue was explored a little and it was argued that this hyperparameter required special consideration because of the nature of the data the method operates on.

The extended founder analysis method was run with suitable prior distributions set after the model was discussed with a subject expert. The output obtained gave posterior density estimates for the migration time of each founder sequence type that was consistent with what one would expect from the ρ values, while at the same time consistent with the underlying migration model. The posterior estimates for various founder sequence types were presented to demonstrate the workings of the method.

An attempt was made to demonstrate the more general analysis on the original founder dataset which incorporated the additional task of estimating the locations and spreads of the components. The work presented in this section was purposely brief as the data available were not informative enough for the task, and actually made interpretation of other parameters more difficult. This task, when one is aiming to estimate a non-trivial number of components (ideally, five or six, according to the expert) is prohibitively difficult because of the small number of founder sequence types. Although there were 134 of these, many are uninformative since their ρ data spanned a

large fraction of the allowable range. The small number of founder sequence types, many providing very uninformative data, resulted in early and later components being extremely difficult to determine. The method struggled with issues such as empty components, while on other occasions it appeared to be the case that some components were unnecessary.

It was then argued that the complete extended founder analysis that estimated the migration times of each founder, the component means and variances, and all other quantities of interest (such as the proportion of the European sample that originated from each period), was in fact *less* useful than the fixed mean and variance case, as interpretation of parameters at the founder level becomes much more difficult when the components are not as clearly defined.

7.2 Criticisms and areas for improvement

Although a step forward, the work of thesis still has room for improvement. In this section I suggest some obvious areas where the method presented could be improved, and discuss other ideas and concepts that one may wish to build into further attempts at developing founder analysis.

7.2.1 Direct additions to the extended founder method

Some areas for improvement in the method I have proposed in this thesis have been mentioned already. I summarise them here.

In terms of testing methods like founder analysis that assume specific migration histories such as those involving bursts of migration, further mathematical work on the structured coalescent would be welcome to investigate

formally the validity of the approach taken in this thesis. It is unfortunate that the structured coalescent theory does not aim to accommodate short bursts of strong migration. It is my belief that the approach taken in this thesis was reasonable for what the simulations were trying to achieve, even if the simulations are viewed merely as a crude approximation to the migration process of interest. However, I believe that the types of migration histories that one would wish to simulate may be better done through other means (perhaps forward in time simulations). Further work on *appropriate* simulation methods would be welcome.

Further, one could argue that a larger mathematical problem is to develop the structured coalescent process so that expansion is allowed to occur soon after every migration event (what was assumed in the original founder analysis) for each founder cluster. This is not easily accommodated in the structured coalescent which only allows global parameters at the level of subpopulations - developing a mathematical structured coalescent model which allows this type of behaviour at the level of single founder clusters seems a tough mathematical problem and I imagine one for which no easy solution may exist.

Formally estimating the number of components in the full extended mixture model is certainly a statistical problem that could be attempted in the future. Some standard statistical approaches to this problem were touched upon briefly in Chapter 6 of this thesis. I would suggest this problem is best left until the method of founder analysis can be shown to perform better in the case where the number of components is non-trivial. Currently this standard statistical problem appears to be unnecessary due to the uninformative data and the small number of founders assigned to the early and later components.

Within the extended founder model presented, the α hyperparameter vector was discovered to take on a unique role due to the fact that the proportion of founder sequence types expected (*a-priori*) to be assigned to each component is a very difficult quantity to estimate. It would be of interest, to start with, to try to model (or even just simulate) this vector of proportions, to attempt to gain an understanding of the way in which founder sequence types are distributed throughout the migration periods under different migration models. For such a problem, one could even start with a very general structured coalescent model with a migration rate matrix which was fixed and unchanging throughout the entire depth of the tree. Once an understanding of this developed, one could perhaps justify and decide on suitable prior values of the α vector that may not be equal in all elements. Such a decision would be difficult to justify unless it also was accepted by experts in the application of the method.

7.2.2 Design issues

At the data collection level, one could undertake some work to try to estimate the optimal sampling proportions of source (e.g. Near Eastern) and descendent (e.g. European) populations. It was argued earlier that sampling more European sequences may give diminishing returns if the number of Near Eastern sequences is held fixed. It was also suggested that sampling more Near Eastern sequences was also likely to result in only a limited number of additional founder sequence types being identified once the majority of the founder sequence types were found. The relationship between the number of founder sequence types identified and the proportion of Near Eastern sequences sampled is likely to be a complicated one, and this relationship is also most probably going to depend on total sample size, which makes such

a task more difficult. Even the question "Does one prefer having large numbers of small founder clusters, or small numbers of large founder clusters?" remains open to debate. The answer to this question, however, is an essential consideration if the larger problem of trying to decide on the proportions of our sample to be taken from each population is to be investigated.

Another design issue that one may wish to investigate is that of the sampling locations within each subpopulation. Attempting to select which parts of Europe or the Near East to draw samples from, in order to gain the most information from a sample of fixed size, is not a trivial issue. It is difficult to suggest how such a sampling strategy could be determined, but the convenient approach of simply using all of the available data (the approach taken by Richards et al. [26]) is unlikely to give as good information about the migration history of a sample as that which would be available from a sample (of identical size) that had been sampled according to some sensible sampling strategy. If new sequence data were to be collected the question of where to sample from would be one that should be considered before any data was collected.

7.2.3 Phylogenetic improvements

The connected star-tree idea presented in this thesis provides a starting point in modelling a migration time which can be assumed to have occurred between two nodes on a reconstructed tree. One could envisage taking the idea further and developing an approach based on a more general (perhaps multifurcating) phylogeny. Such an approach would allow the investigator to better accommodate the large amount of dependence that actually exists on a general reconstructed tree. Using a more general tree introduces

more ρ mutations which would complicate any likelihood calculations and deriving expressions for quantities similar to $P(\tau_A, \tau_B|\rho_A, \rho_B)$ would become more difficult. The benefit of such a generalisation though would be that of more accurate posterior distributions on parameters such as (τ_A, τ_B) being obtained.

An important inherent assumption of founder analysis is that it assumes a *single* migration event occurs with each founding event. Essentially, a parsimony approach to this problem is suggested. In reality, however, a founding event may have involved multiple migration events which occurred at various points along a single edge of a tree. The consequence of this parsimony approach is clearly an underestimation of the number of founder sequence types, and, interestingly, this adds even more support to the idea that not all founders are equal, and that perhaps one may wish to weight the information from the founder sequence types in some systematic way. Both the information provided by a founder cluster (e.g. its size and length of connecting edge) and the number of migration events involved in a given founding event are factors which make the information from founder clusters highly variable between clusters. Estimating the number of migrations involved in a given founding event, together with weighing the information appropriately from given founder sequence types, is another example of a difficult statistical issue which researchers may wish to attempt to tackle in the future.

Little mention was made in this thesis about the conditioning on the *single* reconstructed phylogeny and how this would affect any founder analysis. One of course acknowledges the variability that is lost through such a process. It is difficult to argue against the view that this uncertainty ideally should be dealt with in some formal statistical manner. However, in light of the

considerable uncertainty already present and the lack of information in the data, this area of 'improvement' is likely to achieve little more than massively complicating the method to obtain an inference which is more uninformative than the current approach (although it would be a move towards a more honest inference procedure). At this time I believe the effort required to add in this level of detail would be better directed elsewhere.

Perhaps an area worth exploring in the future would be the differences between the $f0$, $f1$, $f2$ and fs criteria on the founder sequence types identified. This could be done through simulation procedures to generate founder sequence types and then each of the criteria could be used to form a candidate founder list. This would be interesting as it would allow some understanding of the numbers of founder sequence types that were being wrongly identified under each of the criteria. Further, it may even be possible through such an investigation to determine how the age of the migration period affects the number of falsely identified founder sequence types under each of the criteria. Currently the consequences of selecting, say, the $f1$ candidate founder list over the $f2$ candidate list is unknown.

7.2.4 Better use of genetic/other data

mtDNA has been the primary type of data used in founder analysis work. One uses such data primarily because it is non-recombining and relatively fast mutating. However, there is no reason why other parts of the genome cannot be used in any founder analysis approach. If appropriate care were taken to ensure that multiple independent parts of the genome were selected (e.g. on different chromosomes), and that each part was not severely disrupted by recombination (e.g. a haplotype block between recombination hotspots),

then one could obtain independent phylogenies from each of the independent sequence data.

It is possible (in the ideal world where the data at each part of the genome are very informative) that different parts of the genome could display similar numbers of founder sequence types from similar numbers of migration periods. In contrast, one could imagine a situation where different parts of the genome gave rise to vastly different numbers of founder sequence types and/or founder sequence types which appear to indicate different migration histories for the sample under consideration. Combining information from different sites in a formal statistical manner would be a difficult task, but one which would allow a stronger inference to be made, particularly if the data at each independent part of the genome were essentially telling the same story. Even the case where every location suggests a different migration history would be interesting as it a) would suggest that founder analysis is not an ideal method to use, and b) suggests that further work should be directed to establish why the reconstructed migration history of a single sample should differ greatly when using different independent parts of the genome.

Perhaps a more difficult task a statistician would be interested in would be the problem of adding non-sequence data into a statistical founder analysis. The beliefs held by expert scientists in the field are partly driven by archaeological data. This of course would be a non-trivial problem. However, one would hope that the information in any suitable archaeological data would supplement the genetic data. It is reasonable to argue that one is already relying on non-genetic data in a founder analysis as prior specifications (even the locations of the migration bursts in Richards et al. [26]) typically encompass the expert's view based on his/her exposure to multiple different sources

of information, one of which is the archaeological data.

These open questions and problems remain. This thesis presents a first step at putting founder analysis on a firm statistical foundation.

Appendix A

Figures, tables and miscellaneous output

Dataset to be analysed

	Founder	n_a	r_A	n_b	r_B	ρ_A	ρ_B
1	h00	855	531	1017	657	0.62	0.65
2	h01	108	73	131	88	0.68	0.67
3	h02	2	1	18	14	0.50	0.78
4	h05	6	4	14	6	0.67	0.43
5	h06	39	42	42	45	1.08	1.07
6	h07	31	29	35	30	0.94	0.86
7	h08	34	15	42	19	0.44	0.45
8	h10	8	9	10	11	1.13	1.10
9	h12	1	0	3	2	0.00	0.67
10	h13	19	15	26	21	0.79	0.81
11	h14	30	14	36	14	0.47	0.39

12	h15	3	2	4	4	0.67	1.00
13	h16	11	4	12	5	0.36	0.42
14	h18	4	1	6	2	0.25	0.33
15	h21	6	2	9	3	0.33	0.33
16	h23	5	1	6	2	0.20	0.33
17	h25	7	0	11	1	0.00	0.09
18	h26	26	14	33	20	0.54	0.61
19	h28	8	2	12	5	0.25	0.42
20	h29	6	7	7	8	1.17	1.14
21	h30	10	3	12	4	0.30	0.33
22	h32	8	4	12	8	0.50	0.67
23	h35	23	9	29	15	0.39	0.52
24	h36	4	3	8	7	0.75	0.88
25	h37	7	8	11	11	1.14	1.00
26	h38	1	0	4	5	0.00	1.25
27	h39	9	5	10	6	0.56	0.60
28	h40	1	0	3	1	0.00	0.33
29	h41	3	1	5	2	0.33	0.40
30	h43	2	3	3	5	1.50	1.67
31	h44	1	0	3	1	0.00	0.33
32	h45	2	2	8	10	1.00	1.25
33	h46	4	0	7	1	0.00	0.14
34	h52	2	0	8	2	0.00	0.25
35	h53	1	0	2	1	0.00	0.50
36	h62	2	0	5	3	0.00	0.60
37	h76	1	0	3	4	0.00	1.33
38	h78	5	1	6	3	0.20	0.50

39	h79	3	1	5	2	0.33	0.40
40	h81	2	4	5	4	2.00	0.80
41	hv01	5	9	38	40	1.80	1.05
42	hv02	3	2	4	3	0.67	0.75
43	hv03	15	11	25	12	0.73	0.48
44	hv04	1	0	6	4	0.00	0.67
45	hv06	5	6	1254	1034	1.20	0.82
46	hv07	1	0	3	1	0.00	0.33
47	hv08	5	0	6	0	0.00	0.00
48	hv09	1	0	7	10	0.00	1.43
49	i01	22	8	74	125	0.36	1.69
50	i02	27	27	44	62	1.00	1.41
51	i03	2	2	5	8	1.00	1.60
52	i05	2	2	5	5	1.00	1.00
53	i06	3	1	4	6	0.33	1.50
54	i07	2	1	6	17	0.50	2.83
55	i08	1	0	2	1	0.00	0.50
56	j00	172	74	382	608	0.43	1.59
57	j01	27	7	34	15	0.26	0.44
58	j02	17	20	31	29	1.18	0.94
59	j03	22	3	132	278	0.14	2.11
60	j04	17	24	49	54	1.41	1.10
61	j05	3	4	104	165	1.33	1.59
62	j13	1	0	4	2	0.00	0.50
63	j18	2	3	8	15	1.50	1.88
64	k01	122	71	221	182	0.58	0.82
65	k02	27	7	45	19	0.26	0.42

66	k03	6	3	8	4	0.50	0.50
67	k04	2	3	3	5	1.50	1.67
68	k09	1	0	4	1	0.00	0.25
69	n01	3	2	27	28	0.67	1.04
70	n02	2	1	3	2	0.50	0.67
71	n03	1	0	3	2	0.00	0.67
72	n04	2	5	3	6	2.50	2.00
73	n05	1	0	10	21	0.00	2.10
74	n06	4	6	6	7	1.50	1.17
75	n07	1	0	8	9	0.00	1.13
76	ph01	6	5	55	65	0.83	1.18
77	ph02	2	0	16	11	0.00	0.69
78	ph03	2	2	3	4	1.00	1.33
79	ph05	2	0	3	1	0.00	0.33
80	t01	43	13	70	33	0.30	0.47
81	t02	82	50	98	62	0.61	0.63
82	t03	12	3	17	5	0.25	0.29
83	t04	6	0	12	2	0.00	0.17
84	t05	8	2	11	5	0.25	0.45
85	t06	1	0	4	5	0.00	1.25
86	t07	1	0	3	3	0.00	1.00
87	t08	7	5	11	8	0.71	0.73
88	t09	1	0	4	1	0.00	0.25
89	t11	64	28	114	82	0.44	0.72
90	t13	3	0	4	1	0.00	0.25
91	t14	1	0	3	3	0.00	1.00
92	u01	9	12	39	49	1.33	1.26

93	u02b	3	1	5	3	0.33	0.60
94	u04	2	1	10	6	0.50	0.60
95	u06	12	17	28	42	1.42	1.50
96	u07	6	7	10	11	1.17	1.10
97	u09	16	8	88	87	0.50	0.99
98	u10	3	0	9	6	0.00	0.67
99	u11	2	3	4	4	1.50	1.00
100	u12	3	0	5	1	0.00	0.20
101	u13	1	0	3	4	0.00	1.33
102	u14	1	0	2	3	0.00	1.50
103	u16	58	35	105	109	0.60	1.04
104	u17	6	2	7	4	0.33	0.57
105	u18	20	21	23	27	1.05	1.17
106	u21	54	92	278	617	1.70	2.22
107	u22	28	27	1296	3852	0.96	2.97
108	u23	1	0	3	2	0.00	0.67
109	u24	1	0	2	4	0.00	2.00
110	u25	3	1	20	31	0.33	1.55
111	u26	3	3	9	6	1.00	0.67
112	u27	27	7	135	218	0.26	1.61
113	u28	81	78	98	103	0.96	1.05
114	u29	11	8	13	9	0.73	0.69
115	u31	3	2	8	18	0.67	2.25
116	u33	1	0	2	1	0.00	0.50
117	u34	3	2	4	3	0.67	0.75
118	u35	63	54	81	78	0.86	0.96
119	u36	16	4	18	6	0.25	0.33

120	u38	2	1	4	3	0.50	0.75
121	u40	1	0	4	5	0.00	1.25
122	u41	1	0	5	5	0.00	1.00
123	u42	1	0	2	2	0.00	1.00
124	u45	1	0	4	2	0.00	0.50
125	v01	127	86	134	91	0.68	0.68
126	v04	1	0	2	1	0.00	0.50
127	w01	38	31	73	83	0.82	1.14
128	w02	10	4	13	7	0.40	0.54
129	w04	6	2	10	4	0.33	0.40
130	x01	34	29	78	81	0.85	1.04
131	x02	2	2	4	3	1.00	0.75
132	x03	1	0	3	2	0.00	0.67
133	x04	2	2	4	3	1.00	0.75
134	x09	3	5	4	6	1.67	1.50

Table A.1: Table of the founder sequence type data.

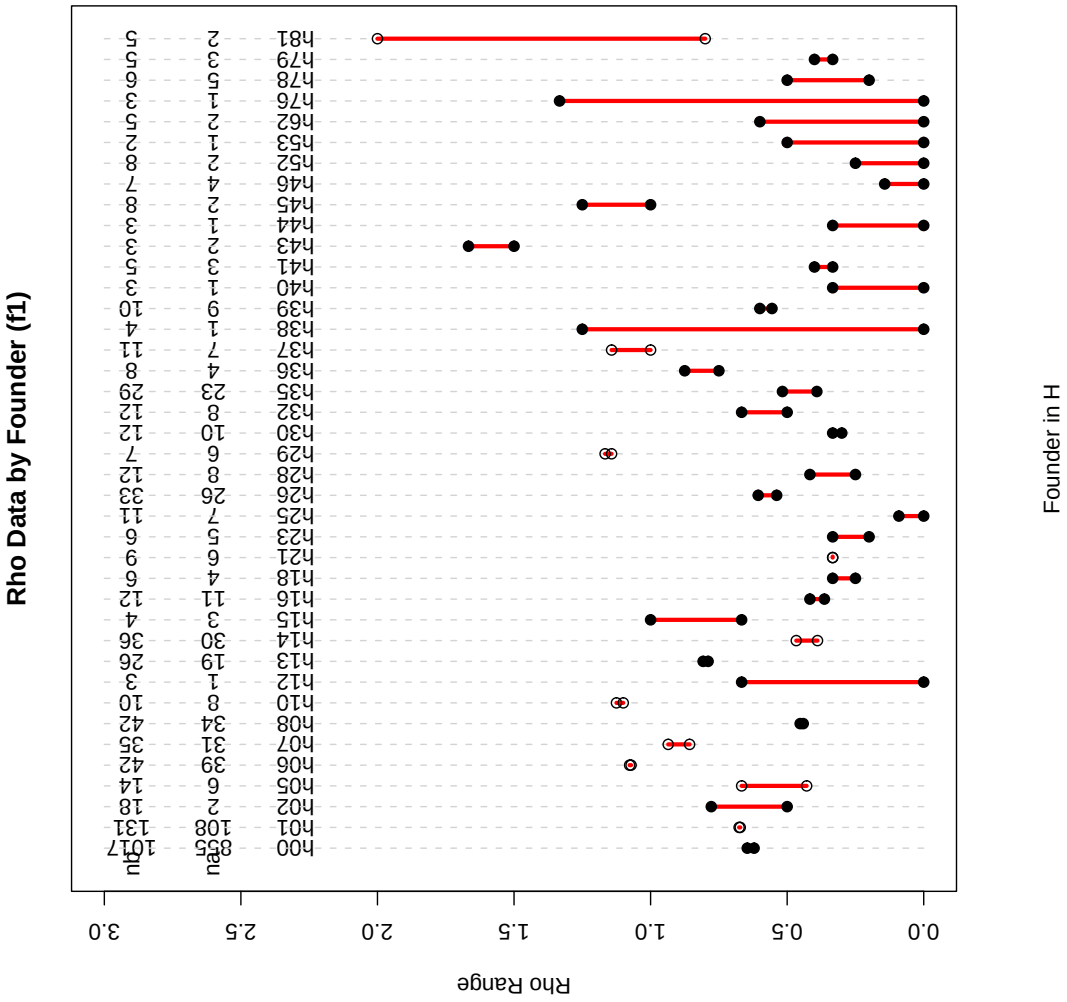

Figure A.1: Plot of those founders located in haplogroup H.

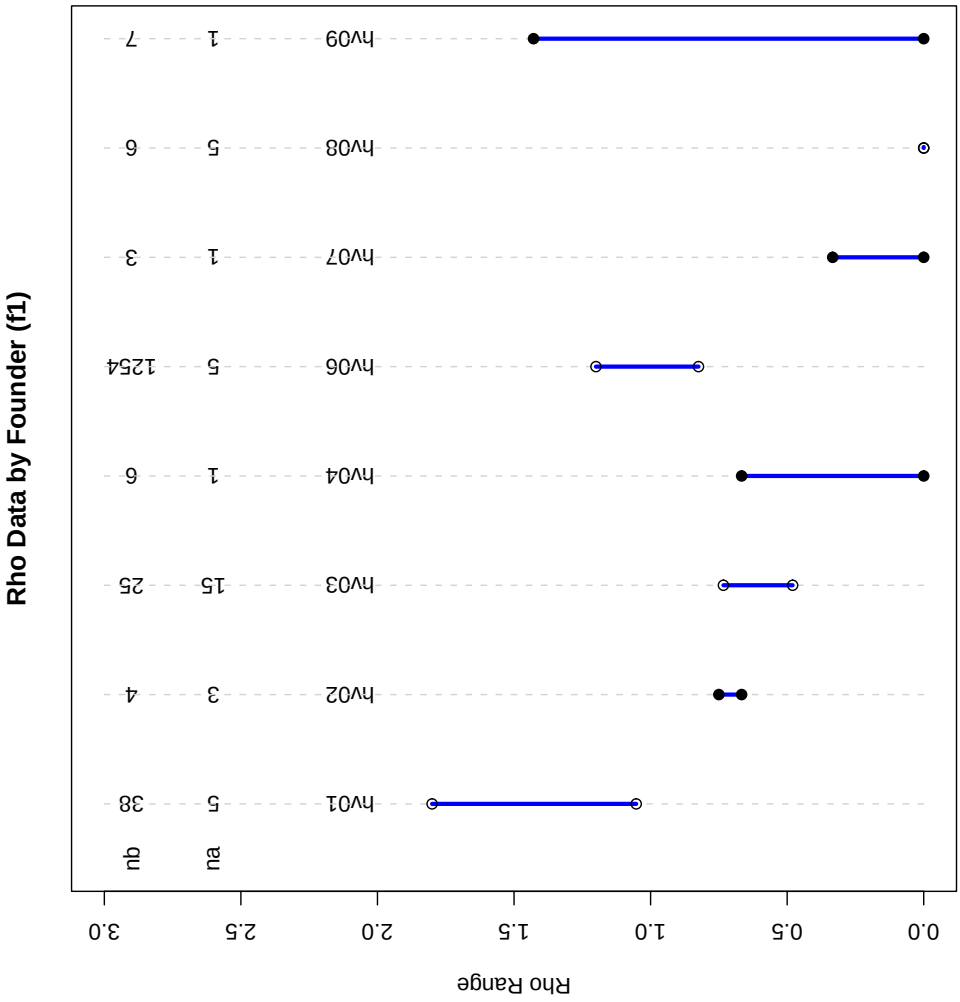

Figure A.2: Plot of those founders located in haplogroup HV.

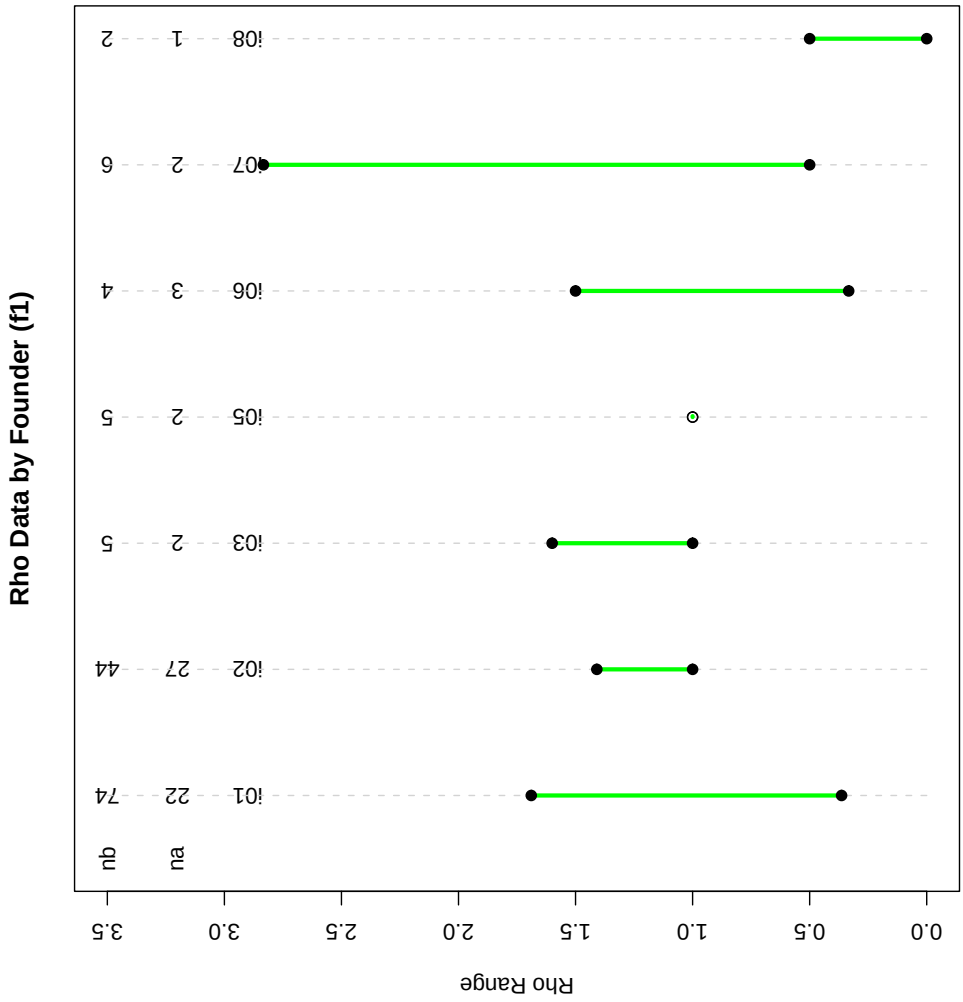

Figure A.3: Plot of those founders located in haplogroup I.

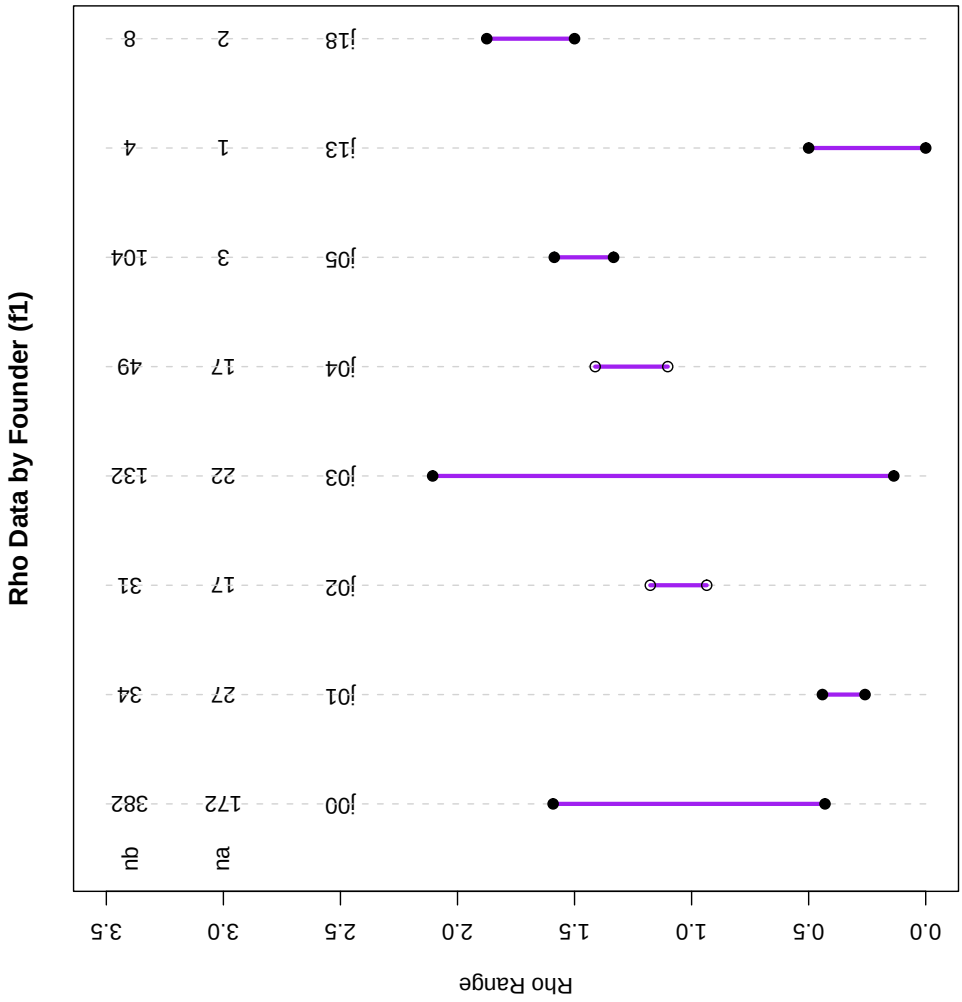

Figure A.4: Plot of those founders located in haplogroup J.

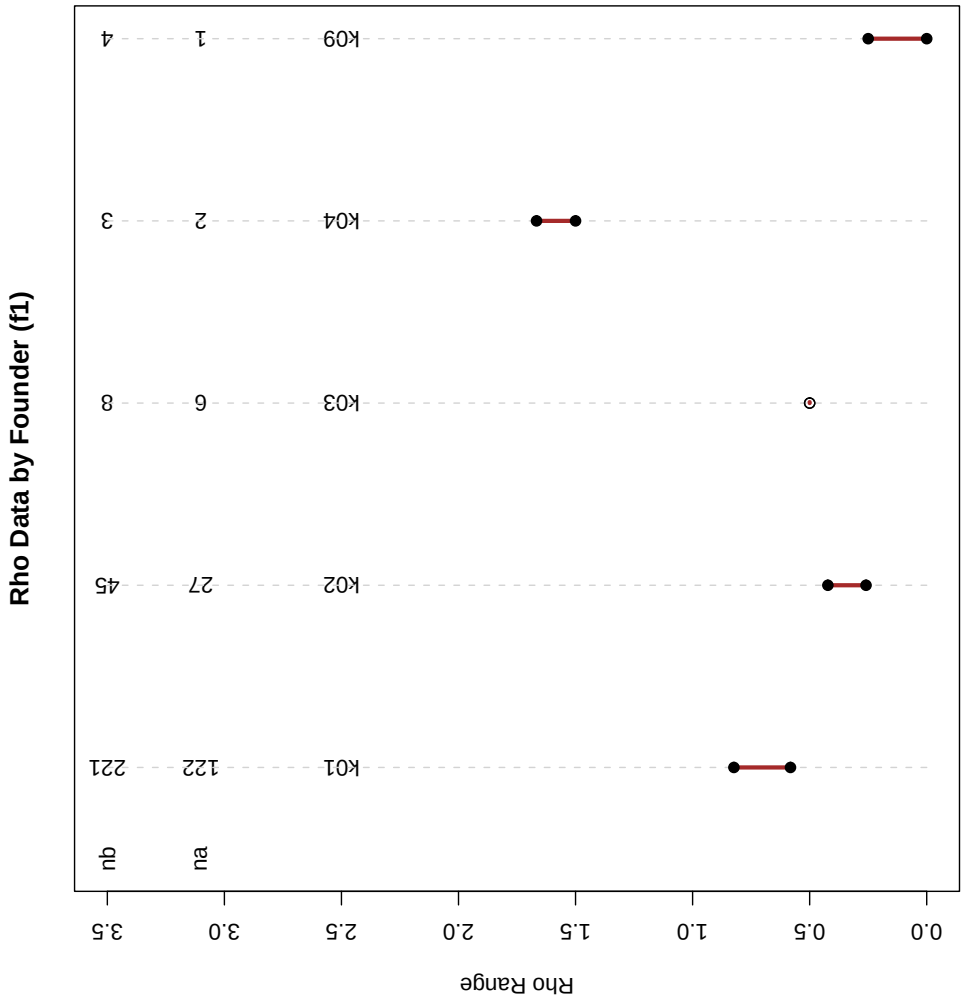

Figure A.5: Plot of those founders located in haplogroup K.

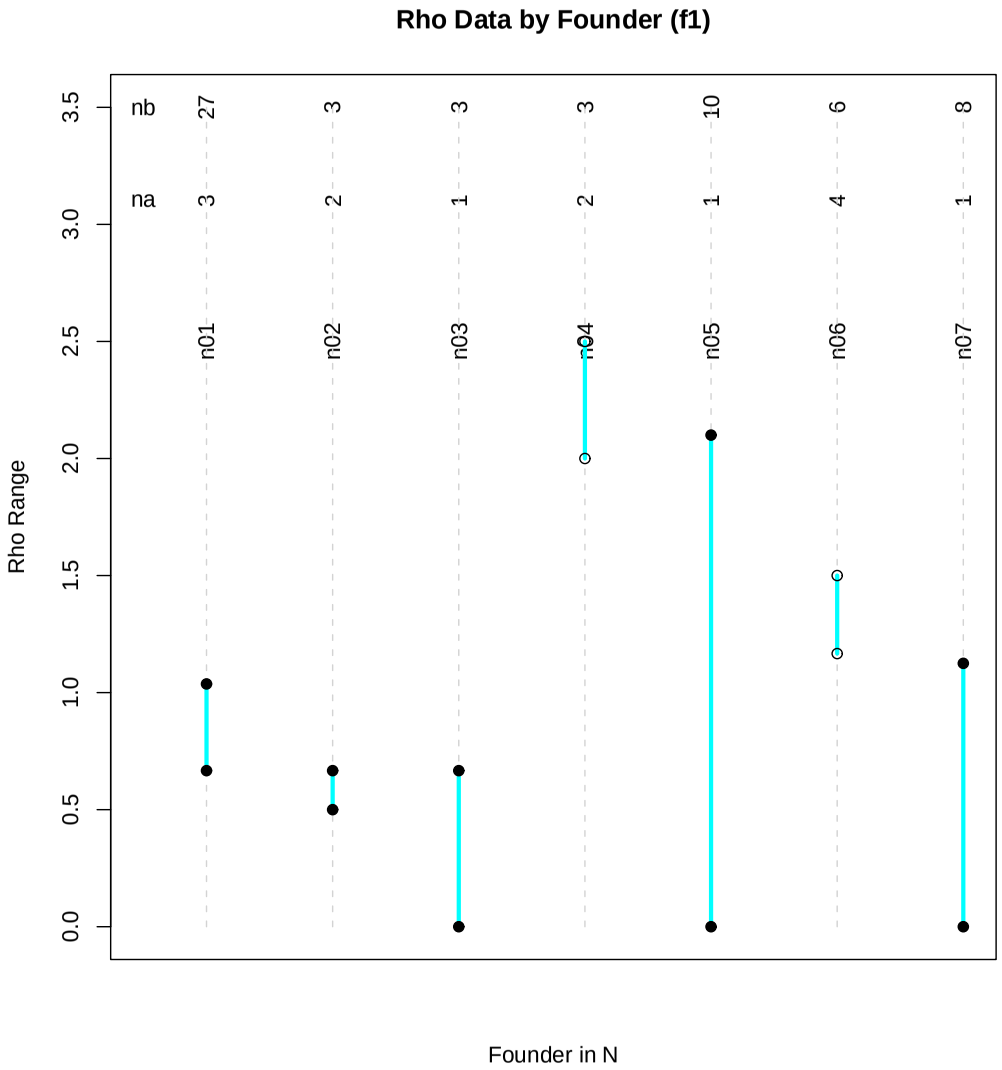

Figure A.6: Plot of those founders located in haplogroup N.

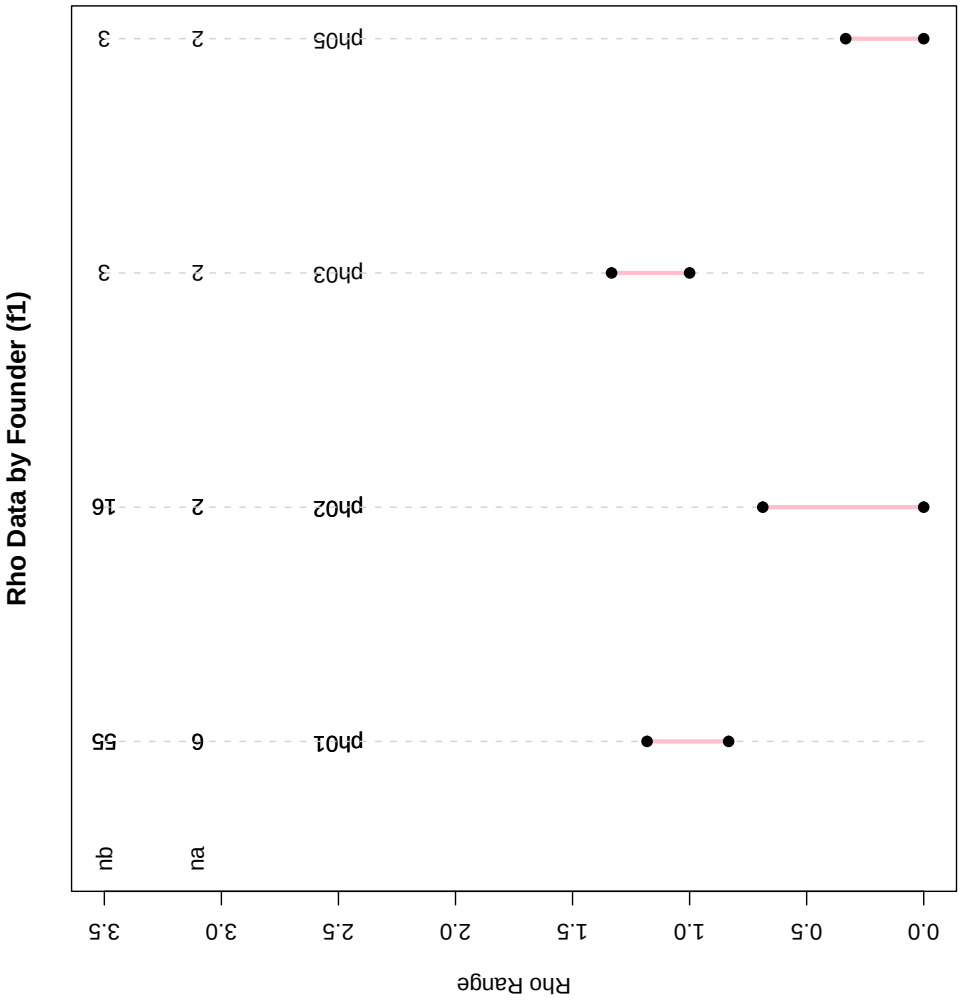

Figure A.7: Plot of those founders located in haplogroup PH.

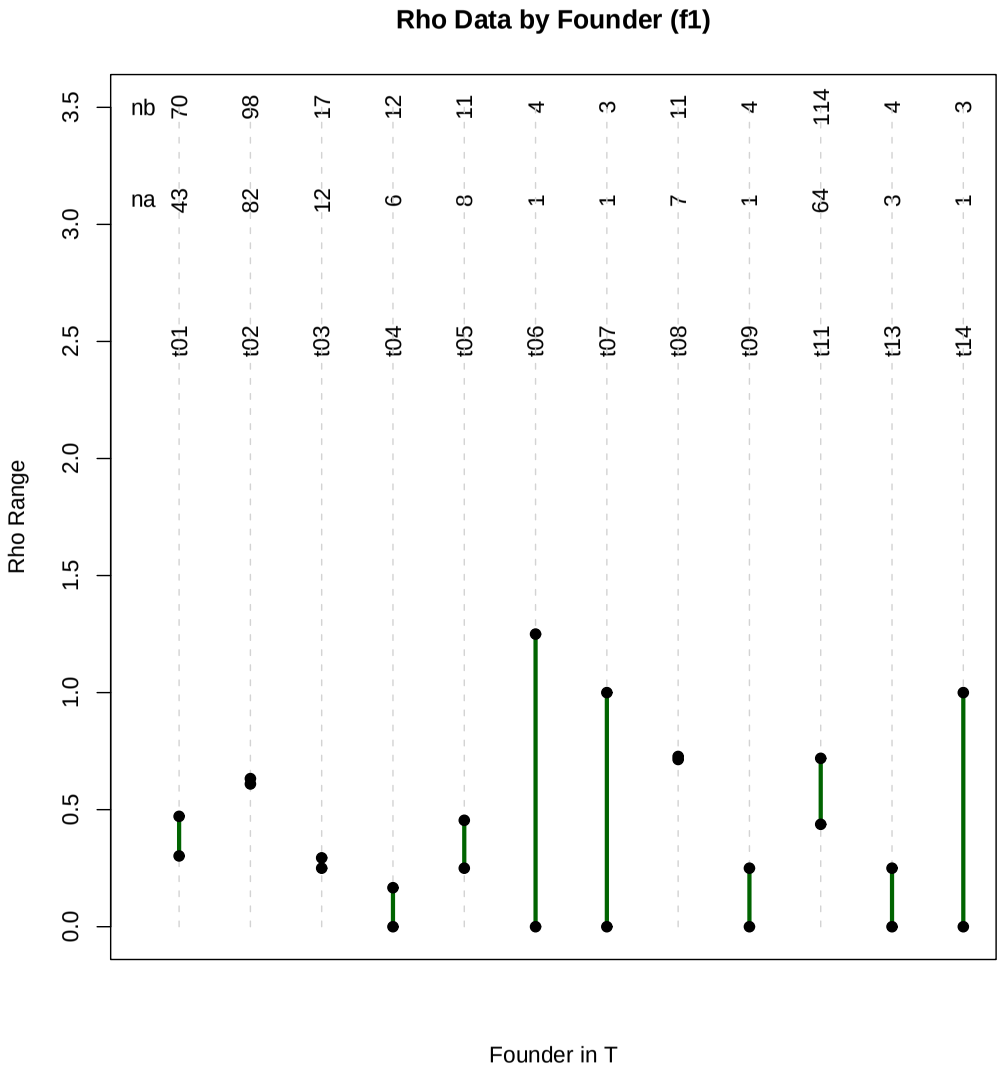

Figure A.8: Plot of those founders located in haplogroup T.

Figure A.9: Plot of those founders located in haplogroup U.

Figure A.10: Plot of those founders located in haplogroups V, W or X.

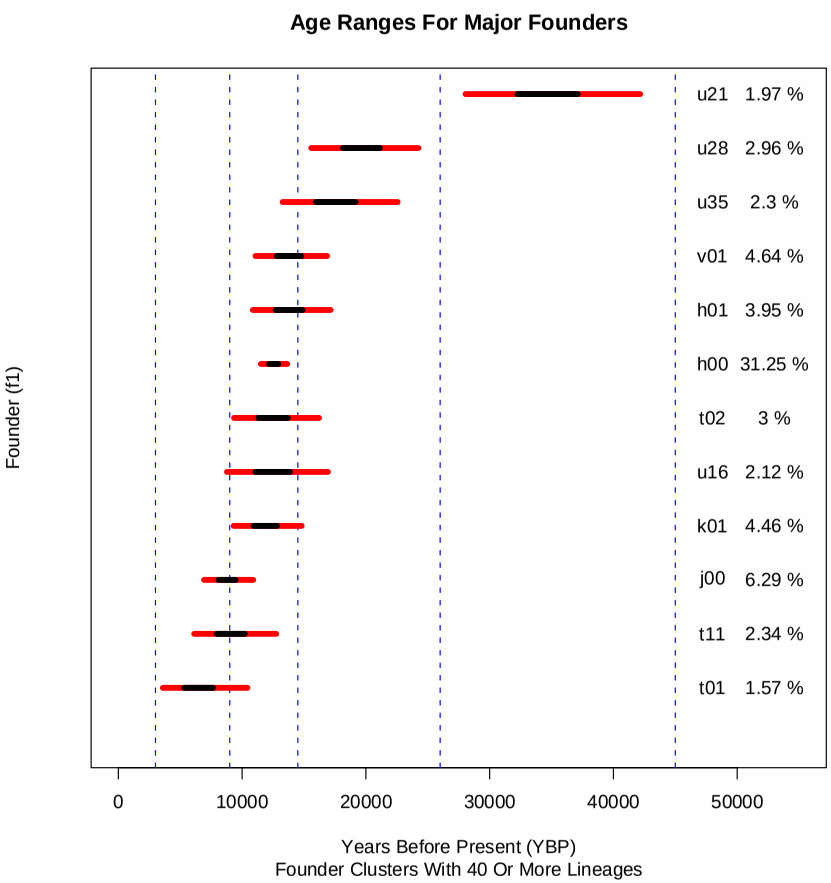

Figure A.11: Credible regions for ages of major founder clusters using the original method (gamma distribution on ages). The inner bars represent the 50% credible regions while the outer bars represent the 95% credible regions.

Credible regions by founder

The following table gives the $2.5\%, 25\%, 75\%, 97.5\%$ credible region values for every founder under the $f1$ criterion, together with the number of members in the founder cluster (n_a), with the percentage of the European sample contained in the cluster detailed in the final column.

Founder	2.5%	25%	75%	97.5%	n_a	Percentage
h00	11511.99	12185.10	12919.22	13645.61	855	31.25
h01	10857.20	12711.64	14874.61	17150.49	108	3.95
h02	2443.89	9699.30	27168.68	56217.88	2	0.07
h05	5460.33	11329.73	21103.00	34445.88	6	0.22
h06	16102.25	19877.02	24434.81	29375.77	39	1.43
h07	13176.16	17020.79	21801.39	27112.05	31	1.13
h08	5428.05	7806.13	10972.28	14684.05	34	1.24
h10	12096.37	19488.55	30052.68	43096.42	8	0.29
h12	510.91	5805.42	27975.42	74441.59	1	0.04
h13	9713.36	13968.87	19634.60	26276.72	19	0.69
h14	5647.30	8232.64	11704.31	15800.69	30	1.10
h15	4161.60	11618.97	26371.24	48598.07	3	0.11
h16	2978.36	6179.85	11510.73	18788.66	11	0.40
h18	1221.95	4849.65	13584.34	28108.94	4	0.15
h21	2080.80	5809.48	13185.62	24299.03	6	0.22
h23	977.56	3879.72	10867.47	22487.15	5	0.18
h25	72.99	829.35	3996.49	10634.51	7	0.26
h26	6516.11	9499.19	13504.98	18231.56	26	0.95
h28	1560.60	4357.11	9889.21	18224.27	8	0.29
h29	11616.39	20032.38	32571.97	48508.26	6	0.22

h30	2199.35	5116.28	10310.82	17692.36	10	0.37
h32	4095.24	8497.29	15827.25	25834.41	8	0.29
h35	4207.43	6778.63	10453.10	14990.06	23	0.84
h36	5498.37	12790.69	25777.06	44230.89	4	0.15
h37	11864.03	19711.95	31141.91	45443.02	7	0.26
h38	510.91	5805.42	27975.42	74441.59	1	0.04
h39	4937.14	9460.41	16643.35	26162.99	9	0.33
h40	510.91	5805.42	27975.42	74441.59	1	0.04
h41	1629.26	6466.20	18112.45	37478.59	3	0.11
h43	10996.74	25581.38	51554.12	88461.79	2	0.07
h44	510.91	5805.42	27975.42	74441.59	1	0.04
h45	6242.40	17428.45	39556.86	72897.10	2	0.07
h46	127.73	1451.36	6993.86	18610.40	4	0.15
h52	255.46	2902.71	13987.71	37220.79	2	0.07
h53	510.91	5805.42	27975.42	74441.59	1	0.04
h62	255.46	2902.71	13987.71	37220.79	2	0.07
h76	510.91	5805.42	27975.42	74441.59	1	0.04
h78	977.56	3879.72	10867.47	22487.15	5	0.18
h79	1629.26	6466.20	18112.45	37478.59	3	0.11
h81	16380.98	33989.18	63309.01	103337.63	2	0.07
hv01	19354.19	31181.68	48084.28	68954.27	5	0.18
hv02	4161.60	11618.97	26371.24	48598.07	3	0.11
hv03	8341.84	12805.73	18996.88	26478.90	15	0.55
hv04	510.91	5805.42	27975.42	74441.59	1	0.04
hv06	11358.77	20513.60	34541.97	52708.04	5	0.18
hv07	510.91	5805.42	27975.42	74441.59	1	0.04
hv08	102.18	1161.08	5595.08	14888.32	5	0.18

hv09	510.91	5805.42	27975.42	74441.59	1	0.04
i01	3774.92	6271.99	9908.79	14459.14	22	0.80
i02	13906.11	18141.82	23442.22	29360.84	27	0.99
i03	6242.40	17428.45	39556.86	72897.10	2	0.07
i05	6242.40	17428.45	39556.86	72897.10	2	0.07
i06	1629.26	6466.20	18112.45	37478.59	3	0.11
i07	2443.89	9699.30	27168.68	56217.88	2	0.07
i08	510.91	5805.42	27975.42	74441.59	1	0.04
j00	6921.30	8094.46	9461.79	10899.57	172	6.29
j01	2581.42	4451.64	7238.21	10779.61	27	0.99
j02	15430.97	21076.18	28350.67	36666.32	17	0.62
j03	999.70	2325.58	4686.74	8041.98	22	0.80
j04	19205.05	25487.39	33435.65	42389.99	17	0.62
j05	10920.65	22659.45	42206.00	68891.75	3	0.11
j13	510.91	5805.42	27975.42	74441.59	1	0.04
j18	10996.74	25581.38	51554.12	88461.79	2	0.07
k01	9318.46	10935.20	12823.78	14813.58	122	4.46
k02	2581.42	4451.64	7238.21	10779.61	27	0.99
k03	3665.58	8527.13	17184.71	29487.26	6	0.22
k04	10996.74	25581.38	51554.12	88461.79	2	0.07
k09	510.91	5805.42	27975.42	74441.59	1	0.04
n01	4161.60	11618.97	26371.24	48598.07	3	0.11
n02	2443.89	9699.30	27168.68	56217.88	2	0.07
n03	510.91	5805.42	27975.42	74441.59	1	0.04
n04	22217.11	42571.82	74895.06	117733.47	2	0.07
n05	510.91	5805.42	27975.42	74441.59	1	0.04
n06	14198.46	25642.00	43177.46	65885.05	4	0.15

n07	510.91	5805.42	27975.42	74441.59	1	0.04
ph01	7405.70	14190.61	24965.02	39244.49	6	0.22
ph02	255.46	2902.71	13987.71	37220.79	2	0.07
ph03	6242.40	17428.45	39556.86	72897.10	2	0.07
ph05	255.46	2902.71	13987.71	37220.79	2	0.07
t01	3592.01	5316.53	7654.44	10432.78	43	1.57
t02	9345.03	11325.16	13687.51	16222.45	82	3.00
t03	1832.79	4263.56	8592.35	14743.63	12	0.44
t04	85.15	967.57	4662.57	12406.93	6	0.22
t05	1560.60	4357.11	9889.21	18224.27	8	0.29
t06	510.91	5805.42	27975.42	74441.59	1	0.04
t07	510.91	5805.42	27975.42	74441.59	1	0.04
t08	6347.75	12163.38	21398.59	33638.13	7	0.26
t09	510.91	5805.42	27975.42	74441.59	1	0.04
t11	6123.92	7948.84	10225.04	12760.00	64	2.34
t13	170.30	1935.14	9325.14	24813.86	3	0.11
t14	510.91	5805.42	27975.42	74441.59	1	0.04
u01	15520.56	23367.80	34120.53	47000.53	9	0.33
u02b	1629.26	6466.20	18112.45	37478.59	3	0.11
u04	2443.89	9699.30	27168.68	56217.88	2	0.07
u06	17939.92	25202.33	34729.46	45772.69	12	0.44
u07	11616.39	20032.38	32571.97	48508.26	6	0.22
u09	5190.51	8623.98	13624.58	19881.32	16	0.58
u10	170.30	1935.14	9325.14	24813.86	3	0.11
u11	10996.74	25581.38	51554.12	88461.79	2	0.07
u12	170.30	1935.14	9325.14	24813.86	3	0.11
u13	510.91	5805.42	27975.42	74441.59	1	0.04

u14	510.91	5805.42	27975.42	74441.59	1	0.04
u16	8772.72	11061.51	13863.39	16936.07	58	2.12
u17	2080.80	5809.48	13185.62	24299.03	6	0.22
u18	13911.37	18849.70	25181.06	32389.64	20	0.73
u21	28051.32	32260.59	37112.63	42164.93	54	1.97
u22	13409.46	17493.90	22604.99	28312.24	28	1.02
u23	510.91	5805.42	27975.42	74441.59	1	0.04
u24	510.91	5805.42	27975.42	74441.59	1	0.04
u25	1629.26	6466.20	18112.45	37478.59	3	0.11
u26	7331.16	17054.25	34369.42	58974.52	3	0.11
u27	2581.42	4451.64	7238.21	10779.61	27	0.99
u28	15582.21	18146.38	21126.64	24252.76	81	2.96
u29	7549.84	12543.97	19817.58	28918.29	11	0.40
u31	4161.60	11618.97	26371.24	48598.07	3	0.11
u33	510.91	5805.42	27975.42	74441.59	1	0.04
u34	4161.60	11618.97	26371.24	48598.07	3	0.11
u35	13271.88	15962.38	19156.33	22569.02	63	2.30
u36	2047.62	4248.65	7913.63	12917.20	16	0.58
u38	2443.89	9699.30	27168.68	56217.88	2	0.07
u40	510.91	5805.42	27975.42	74441.59	1	0.04
u41	510.91	5805.42	27975.42	74441.59	1	0.04
u42	510.91	5805.42	27975.42	74441.59	1	0.04
u45	510.91	5805.42	27975.42	74441.59	1	0.04
v01	11072.53	12797.68	14792.82	16876.41	127	4.64
v04	510.91	5805.42	27975.42	74441.59	1	0.04
w01	11623.67	14882.75	18912.15	23367.39	38	1.39
w02	3276.20	6797.84	12661.80	20667.53	10	0.37

w04	2080.80	5809.48	13185.62	24299.03	6	0.22
x01	12013.55	15518.96	19877.73	24719.81	34	1.24
x02	6242.40	17428.45	39556.86	72897.10	2	0.07
x03	510.91	5805.42	27975.42	74441.59	1	0.04
x04	6242.40	17428.45	39556.86	72897.10	2	0.07
x09	14811.41	28381.22	49930.04	78488.98	3	0.11

Table A.2: 50% and 95% credible region end points for $f1$ founder clusters.

Appendix B

Some mathematical derivations

Proof of (3.33):

$$\begin{aligned}
\frac{d}{du}\frac{1}{\beta}\log\left[\gamma_{\epsilon_w}\delta + e^{\beta u}\right] &= \frac{1}{\beta}\frac{1}{\left[\gamma_{\epsilon_w}\delta + e^{\beta u}\right]}\left(\beta e^{\beta u}\right) \\
&= \frac{e^{\beta u}}{\left[\gamma_{\epsilon_w}\delta + e^{\beta u}\right]} \\
&= \frac{1}{1+\frac{\gamma_{\epsilon_w}\delta}{e^{\beta u}}} \\
&= \frac{1}{1+\gamma_{\epsilon_w}\delta e^{-\beta u}}.
\end{aligned}$$

By the Fundamental Theorem of Calculus [43], it is clear from the above that (3.33) is indeed the solution to (3.32).

Result (3.35)

$$
\begin{aligned}
X &= k_r N(0) \left[\frac{1}{\beta} \log \left(\gamma_\epsilon \delta + e^{\beta u} \right) \right]_{t_{\alpha-1}}^{t_\alpha} \\
&\Rightarrow \frac{X\beta}{k_r N(0)} = \log \left(\gamma_\epsilon \delta + e^{\beta t_\alpha} \right) - \log \left(\gamma_\epsilon \delta + e^{\beta t_{\alpha-1}} \right) \\
&\Rightarrow \log \left(\gamma_\epsilon \delta + e^{\beta t_\alpha} \right) = \frac{X\beta}{k_r N(0)} + \log \left(\gamma_\epsilon \delta + e^{\beta t_{\alpha-1}} \right) \\
&\Rightarrow \gamma_\epsilon \delta + e^{\beta t_\alpha} = \exp \left\{ \frac{X\beta}{k_r N(0)} + \log \left(\gamma_\epsilon \delta + e^{\beta t_{\alpha-1}} \right) \right\} \\
&\Rightarrow e^{\beta t_\alpha} = \exp \left\{ \frac{X\beta}{k_r N(0)} + \log \left(\gamma_\epsilon \delta + e^{\beta t_{\alpha-1}} \right) \right\} - \gamma_\epsilon \delta \\
&\Rightarrow \beta t_\alpha = \log \left\{ \exp \left[\frac{X\beta}{k_r N(0)} + \log \left(\gamma_\epsilon \delta + e^{\beta t_{\alpha-1}} \right) \right] - \gamma_\epsilon \delta \right\} \\
&\Rightarrow t_\alpha = \frac{1}{\beta} \log \left\{ \exp \left[\frac{X\beta}{k_r N(0)} + \log \left(\gamma_\epsilon \delta + e^{\beta t_{\alpha-1}} \right) \right] - \gamma_\epsilon \delta \right\} \\
&\Rightarrow t_\alpha = \frac{1}{\beta} \log \left\{ \exp \left[\frac{X\beta}{k_r N(0)} \right] \exp \left[\log \left(\gamma_\epsilon \delta + e^{\beta t_{\alpha-1}} \right) \right] - \gamma_\epsilon \delta \right\} \\
&\Rightarrow t_\alpha = \frac{1}{\beta} \log \left\{ \exp \left[\frac{X\beta}{k_r N(0)} \right] \left[\gamma_\epsilon \delta + e^{\beta t_{\alpha-1}} \right] - \gamma_\epsilon \delta \right\}. \qquad \text{(B.1)}
\end{aligned}
$$

Result (3.36)

$$
\begin{aligned}
X &= k_r N(0)\Bigg\{ \int_{t_{\alpha-1}}^{T_{l_{\alpha-1}+1}} \frac{du}{1+\gamma_{\epsilon_{l_{\alpha-1}+1}}\delta e^{-\beta u}} \\
&+ \int_{T_{l_\alpha}}^{t_\alpha} \frac{du}{1+\gamma_{\epsilon_{l_\alpha+1}}\delta e^{-\beta u}} \\
&+ \sum_{s=l_{\alpha-1}+1}^{l_\alpha-1} \int_{T_s}^{T_{s+1}} \frac{du}{1+\gamma_{\epsilon_{s+1}}\delta e^{-\beta u}} \Bigg\} \\
\Rightarrow \quad & \frac{X}{k_r N(0)} = \left[\frac{1}{\beta}\log\left(\gamma_{\epsilon_{l_{\alpha-1}+1}}\delta + e^{\beta u}\right)\right]_{t_{\alpha-1}}^{T_{l_{\alpha-1}+1}} \\
&+ \left[\frac{1}{\beta}\log\left(\gamma_{\epsilon_{l_\alpha+1}}\delta + e^{\beta u}\right)\right]_{T_{l_\alpha}}^{t_\alpha} \\
&+ \sum_{s=l_{\alpha-1}+1}^{l_\alpha-1} \left[\frac{1}{\beta}\log\left(\gamma_{\epsilon_{s+1}}\delta + e^{\beta u}\right)\right]_{T_s}^{T_{s+1}} \\
\Rightarrow \quad & \frac{X\beta}{k_r N(0)} = \left[\log\left(\gamma_{\epsilon_{l_{\alpha-1}+1}}\delta + e^{\beta T_{l_{\alpha-1}+1}}\right) - \log\left(\gamma_{\epsilon_{l_{\alpha-1}+1}}\delta + e^{\beta t_{\alpha-1}}\right)\right] \\
&+ \left[\log\left(\gamma_{\epsilon_{l_\alpha+1}}\delta + e^{\beta t_\alpha}\right) - \log\left(\gamma_{\epsilon_{l_\alpha+1}}\delta + e^{\beta T_{l_\alpha}}\right)\right] \\
&+ \sum_{s=l_{\alpha-1}+1}^{l_\alpha-1} \left[\log\left(\gamma_{\epsilon_{s+1}}\delta + e^{\beta T_{s+1}}\right) - \log\left(\gamma_{\epsilon_{s+1}}\delta + e^{\beta T_s}\right)\right] \\
\Rightarrow \quad & \log\left(\gamma_{\epsilon_{l_\alpha+1}}\delta + e^{\beta t_\alpha}\right) = \frac{X\beta}{k_r N(0)} - \log\left(\gamma_{\epsilon_{l_{\alpha-1}+1}}\delta + e^{\beta T_{l_{\alpha-1}+1}}\right) \\
&+ \log\left(\gamma_{\epsilon_{l_{\alpha-1}+1}}\delta + e^{\beta t_{\alpha-1}}\right) + \log\left(\gamma_{\epsilon_{l_\alpha+1}}\delta + e^{\beta T_{l_\alpha}}\right) \\
&- \sum_{s=l_{\alpha-1}+1}^{l_\alpha-1} \left[\log\left(\gamma_{\epsilon_{s+1}}\delta + e^{\beta T_{s+1}}\right) - \log\left(\gamma_{\epsilon_{s+1}}\delta + e^{\beta T_s}\right)\right] \\
\Rightarrow \quad & \gamma_{\epsilon_{l_\alpha+1}}\delta + e^{\beta t_\alpha} = \exp\left\{\frac{X\beta}{k_r N(0)}\right\} \exp\left\{-\log\left(\gamma_{\epsilon_{l_{\alpha-1}+1}}\delta + e^{\beta T_{l_{\alpha-1}+1}}\right)\right\} \\
&\cdot \left[\gamma_{\epsilon_{l_{\alpha-1}+1}}\delta + e^{\beta t_{\alpha-1}}\right]\left[\gamma_{\epsilon_{l_\alpha+1}}\delta + e^{\beta T_{l_\alpha}}\right] \\
&\cdot \exp\left\{-\sum_{s=l_{\alpha-1}+1}^{l_\alpha-1}\left[\log\left(\gamma_{\epsilon_{s+1}}\delta + e^{\beta T_{s+1}}\right) - \log\left(\gamma_{\epsilon_{s+1}}\delta + e^{\beta T_s}\right)\right]\right\} \\
\Rightarrow \quad & F + e^{\beta t_\alpha} = \exp\left\{A - E_1\right\} B^{-1} C D \\
\Rightarrow \quad & e^{\beta t_\alpha} = \exp\left\{A - E_1\right\} B^{-1} C D - F
\end{aligned}
$$

$$\begin{aligned}
&\Rightarrow \quad \beta t_\alpha = \log\left\{\exp\left[A - E_1\right] B^{-1}CD - F\right\} \\
&\Rightarrow \quad t_\alpha = \frac{1}{\beta}\log\left\{\exp\left[A - E_1\right] B^{-1}CD - F\right\} \\
&\Rightarrow \quad t_\alpha = \frac{1}{\beta}\log\left\{\exp\left[A - E_1\right] \frac{CD}{B} - F\right\}.
\end{aligned}$$

An equivalent result that is sometimes easier to work with is shown below.

First, note that E_1 can be re-expressed as:

$$E_1 = \log\left\{\prod_{s=l_{\alpha-1}+1}^{l_\alpha - 1}\left[\frac{\gamma_{\epsilon_{s+1}}\delta + e^{\beta T_{s+1}}}{\gamma_{\epsilon_{s+1}}\delta + e^{\beta T_s}}\right]\right\}. \tag{B.2}$$

In light of (B.2), $\exp\left[A - E_1\right]$ can be re-expressed as:

$$\begin{aligned}
\exp\left[A - E_1\right] &= \exp[A]\exp\left\{-\log\left\{\prod_{s=l_{\alpha-1}+1}^{l_\alpha - 1}\left[\frac{\gamma_{\epsilon_{s+1}}\delta + e^{\beta T_{s+1}}}{\gamma_{\epsilon_{s+1}}\delta + e^{\beta T_s}}\right]\right\}\right\} \\
&= \exp[A]\left\{\prod_{s=l_{\alpha-1}+1}^{l_\alpha - 1}\left[\frac{\gamma_{\epsilon_{s+1}}\delta + e^{\beta T_{s+1}}}{\gamma_{\epsilon_{s+1}}\delta + e^{\beta T_s}}\right]\right\}^{-1} \\
&= \exp[A]\left[E_2\right]^{-1},
\end{aligned}$$

where,

$$E_2 = \prod_{s=l_{\alpha-1}+1}^{l_\alpha - 1}\left[\frac{\gamma_{\epsilon_{s+1}}\delta + e^{\beta T_{s+1}}}{\gamma_{\epsilon_{s+1}}\delta + e^{\beta T_s}}\right]. \tag{B.3}$$

Thus, equation (3.36) becomes

$$t_\alpha = \frac{1}{\beta}\log\left\{\exp\left[A\right]\frac{CD}{BE_2} - F\right\}. \tag{B.4}$$

Result (3.41)

$$\begin{aligned} c_2 N &= N_e \frac{1}{T}\frac{1}{b}\left(e^{bT}-1\right) \\ \Rightarrow (1-c_1)\,N &= N_e \frac{1}{T}\frac{1}{b}\left(e^{bT}-1\right). \end{aligned}$$

Since $c_2 = (1 - c_1)$

$$\Rightarrow N - Nc_1 = N_e \frac{1}{T}\frac{1}{b}\left(e^{bT}-1\right).$$

But $Nc_1 = \Omega$, hence,

$$\begin{aligned} N - Nc_1 &= N_e \frac{1}{T}\frac{1}{b}\left(e^{bT}-1\right) \\ \Rightarrow N &= \Omega + N_e \frac{1}{T}\frac{1}{b}\left(e^{bT}-1\right). \end{aligned}$$

Covariance of $n_a\rho_A$ and $n_b\rho_B$

The calculation of the covariance between $n_a\rho_A$ and $n_b\rho_B$ leads to an interesting result that the covariance of these two random quantities is equal to the variance of $n_a\rho_A$.

Recall,

$$\begin{aligned} n_a\rho_A &\sim Po(n_a\tau_A), \\ V[n_a\rho_A] &= n_a\tau_A. \end{aligned} \tag{B.5}$$

Now,

$$Cov[n_a\rho_A, n_b\rho_B] = Cov[Y_1, Y_1 + n_aY_2 + Y_3]$$

Where $Y_1 \sim Po(n_a\tau_A), Y_2 \sim Po(\tau_B - \tau_A), Y_3 \sim Po((n_b - n_a)\tau_B)$
Y_1, Y_2, Y_3 are all independent, which gives,

$$\begin{aligned} Cov[n_a\rho_A, n_b\rho_B] &= Cov[Y_1, Y_1 + n_aY_2 + Y_3] \\ &= Cov[Y_1, Y_1] + Cov[Y_1, n_aY_2] + Cov[Y_1, Y_3] \\ &= Var[Y_1] + 0 \\ &= n_a\tau_A. \end{aligned}$$

Bibliography

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

[2] N. A. Rosenberg and M. Nordborg. Genealogical trees, coalescent theory and the analysis of genetic polymorphisms. *Nature Rev. Genet.*, 3:380–390, 2002.

[3] A. R. Templeton, E. Routman, and C. A. Phillips. Seperating population structure from population history: a cladistic analysis of the geographical distribution of mitochondrial DNA haplotypes in the Tiger salamander, *Ambystoma tigrinum*. *Genetics*, 140:767–782, 1995.

[4] L. L. Knowles and W. P. Maddison. Statistical phylogeography. *Mol. Ecol.*, 11:2623–2635, 2002.

[5] F. Tajima. Statistical method for testing the neutral mutation hypothesis by DNA polymorphism. *Genetics*, 123:585–595, 1989.

[6] J. Hein, M. H. Schierup, and C. Wiuf. *Gene Genealogies, Variation and Evolution: A Primer in Coalescent Theory*. Oxford University Press, 271pp, 2005.

[7] J. Wakeley and J. Hey. Estimating ancestral population parameters. *Genetics*, 145:847–855, 1997.

[8] M. A. Beaumont, W. Zhang, and D. J. Balding. Approximate Bayesian computation in population genetics. *Genetics*, 162:2025–2035, 2002.

[9] W. J. Ewens. The sampling theory of selectively neutral alleles. *Theor. Popul. Biol.*, 3:87–112, 1972.

[10] M. Slatkin and W. P. Maddison. A cladistic measure of gene flow inferred from the phylogenies of alleles. *Genetics*, 123:603–613, 1989.

[11] M. Bahlo and R. C. Griffiths. Inference from gene trees in a subdivided population. *Theor. Popul. Biol.*, 57:79–95, 2000.

[12] P. Beerli and J. Felsenstein. Maximum-likelihood estimation of migration rates and effective population numbers in two populations using a coalescent approach. *Genetics*, 152:763–773, 1999.

[13] J. Wakeley, R. Nielsen, S. N. Liu-Cordero, and K. Ardlie. The discovery of single-nucleotide polymorphisms — and inferences about human demographic history. *Am. J. Hum. Genet.*, 69:1332–1347, 2001.

[14] P. Beerli. Effect of unsampled populations on the estimation of population sizes and migration rates between sampled populations. *Mol. Ecol.*, 13:827–836, 2004.

[15] M. Slatkin. Seeing ghosts: the effect of unsampled populations on migration rates estimated for sampled populations. *Mol. Ecol.*, 14:67–73, 2005.

[16] R. C. Griffiths and S. Tavaré. Ancestral inference in population genetics. *Stat. Sci.*, 9:307–319, 1994.

[17] M. K. Kuhner, J. Yamato, and J. Felsenstein. Maximum likelihood estimation of population growth rates based on the coalescent. *Genetics*, 149:429–434, 1998.

[18] A. Drummond, A. Rambaut, B. Shapiro, and O. Pybus. Bayesian coalescent inference of past population dynamics from molecular sequences. *Mol. Biol. Evol.*, 22:1185–1192, 2005.

[19] M. K. Kuhner. LAMARC 2.0: maximum likelihood and Bayesian estimation of population parameters. *Bioinformatics*, 22:768–770, 2006.

[20] Z. Yang. *Computational Molecular Evolution*. Oxford University Press, 376pp, first edition, 2006.

[21] R. Nielsen and J. W. Wakeley. Distinguishing migration from isolation: an MCMC approach. *Genetics*, 158:885–896, 2001.

[22] M. Hasegawa, H. Kishino, and T-A. Yano. Dating of the human-ape splitting by a molecular clock of mitochondrial DNA. *J. Mol. Evol.*, 22:160–174, 1985.

[23] J. Hey and R. Nielsen. Integration within the Felsenstein equation for improved Markov chain Monte Carlo methods in population genetics. *Proc. Natl. Acad. Sci. USA.*, 104:2785–2790, 2007.

[24] I. J. Wilson, M. E. Weale, and D. J. Balding. Inferences from DNA data: population histories, evolutionary processes and forensic match probabilities. *J. R. Statist. Soc. A*, 166:155–188, 2003.

[25] A. Torroni, T. G. Schurr, C.-C. Yang, E. J. E. Szathmary, R. C. Williams, M. S. Schanfield, et al. Native American mitochondrial DNA

analysis indicates that the Amerind and the Nadene populations were founded by two independent migrations. *Genetics*, 130:153–162, 1992.

[26] M. Richards, V. Macaulay, E. Hickey, E. Vega, B. Sykes, V. Guida, et al. Tracing European founder lineages in the Near Eastern mtDNA pool. *Am. J. Hum. Genet.*, 67:1251–1276, 2000.

[27] H.-J. Bandelt, P. Forster, B. C. Sykes, and M. B. Richards. Mitochondrial portraits of human populations using median networks. *Genetics*, 141:743–753, 1995.

[28] G. Barbujani, G. Bertorelle, and L. Chikhi. Evidence for Paleolithic and Neolithic gene flow in Europe. *Am. J. Hum. Genet.*, 62:488–491, 1998.

[29] M. Richards, V. A. Macaulay, H.-J. Bandelt, and B. C. Sykes. Phylogeography of mitochondrial DNA in western Europe. *Ann. Hum. Genet.*, 62:241–260, 1998.

[30] V. Macaulay, M. Richards, E. Hickey, E. Vega, F. Cruciani, V. Guida, et al. The emerging tree of West Eurasian mtDNAs: a synthesis of control-region sequences and RFLPs. *Am. J. Hum. Genet.*, 64:232–249, 1999.

[31] A. Achilli, C. Rengo, V. Battaglia, M. Pala, A. Olivieri, S. Fornarino, et al. Saami and Berbers — an unexpected mitochondrial DNA link. *Am. J. Hum. Genet.*, 76:883–886, 2005.

[32] A. Achilli, C. Rengo, C. Magri, V. Battaglia, A. Olivieri, R. Scozzari, et al. The molecular dissection of mtDNA haplogroup H confirms that the Franco-Cantabrian glacial refuge was a major source for the European gene pool. *Am. J. Hum. Genet.*, 75:910–918, 2004.

[33] H.J. Bandelt, V. Macaulay, and M. Richards. *Human Mitochondrial DNA and the Evolution of Homo Sapiens*. Springer, 2006.

[34] U. Roostalu, I. Kutuev, E.-L. Loogväli, E. Metspalu, K. Tambets, M. Reidla, et al. Origin and expansion of haplogroup H, the dominant human mitochondrial DNA lineage in West Eurasia: the Near Eastern and Caucasian perspective. *Mol. Biol. Evol.*, 24:436–448, 2007.

[35] P. Forster, R. Harding, A. Torroni, and H.-J. Bandelt. Origin and evolution of Native American mtDNA variation: a reappraisal. *Am. J. Hum. Genet.*, 59:935–945, 1996.

[36] G. Barbujani and I. Dupanloup. *Examining the Farming/Language Dispersal Hypothesis (McDonald Institute Monographs)*. McDonald Institute for Archaeological Research, 520pp, 2002.

[37] M. Richards, H. Côrte-Real, P. Forster, V. Macaulay, H. Wilkinson-Herbots, A. Demaine, et al. Paleolithic and Neolithic lineages in the European mitochondrial gene pool. *Am. J. Hum. Genet.*, 59:185–203, 1996.

[38] T. Nagylaki. The strong-migration limit in geographically structured populations. *J. Math. Biol.*, 9:101–114, 1980.

[39] J. Wakeley. *Coalescent Theory*. Roberts and Company, 326pp, first edition, 2008.

[40] M. Slatkin and M. Veuille, editors. *Modern Developments in Theoretical Population Genetics: the Legacy of Gustave Malécote*. Oxford University Press., 280pp, 2002.

[41] S. M. Ross. *Introduction to Probability Models.* Academic Press Inc.,U.S., 800pp, 2006.

[42] R Development Core Team. *R: A language and Environment for Statistical Computing.* Vienna, Austria, 2004. ISBN 3-90051-07-0.

[43] R. Courant and J. Fritz. *Introduction to Calculus and Analysis I.* Springer-Verlag New York Inc., 558pp, 1989.

[44] J. S. Rosenthal. *A First Look at Rigorous Probability Theory.* World Scientific Publishing., 219pp, 2006.

[45] P. J. Donnelly and S. Tavaré, editors. *Progress in Population Genetics and Human Evolution.* Springer-Verlag New York Inc., 329pp, 1997.

[46] V. Macaulay, C. Hill, A. Achilli, C. Rengo, D. Clarke, W. Meehan, et al. Single, rapid coastal settlement of Asia revealed by analysis of complete mitochondrial genomes. *Science*, 308:1034–1036, 2005.

[47] R. Durret. *Probability Models for DNA Sequence Evolution.* Springer-Verlag New York Inc, 248pp, 2002.

[48] M. Nei and N. Takahata. Effective population size, genetic diversity, and coalescence time in subdivided populations. *J. Mol. Evol.*, 37:240–244, 1993.

[49] J. Wakeley and T. Takahashi. Gene genealogies when the sample size exceeds the effective size of the population. *Mol. Biol. Evol.*, 20:208–213, 2003.

[50] N. Morral, J. Bertranpetit, X. Estivill, V. Nunes, T. Casals, J. Giménez, et al. The origin of the major cystic fibrosis mutation ΔF508 in European populations. *Nat. Genet.*, 7:169–175, 1994.

[51] J. Saillard, P. Forster, N. Lynnerup, H.-J. Bandelt, and S. Norby. mtDNA variation among Greenland Eskimos: The edge of the Beringian expansion. *Am. J. Hum. Genet.*, 67:718–726, 2000.

[52] R. Thomson, J. K. Pritchard, P. Shen, P. J. Oefner, and M. W. Feldman. Recent common ancestry of human Y chromosomes: evidence from DNA sequence data. *Proc. Natl. Acad. Sci. USA*, 97:7360–7365, 2000.

[53] A. Torroni, H.-J. Bandelt, L. D'Urbano, P. Lahermo, P. Moral, D. Sellitto, et al. mtDNA analysis reveals a major late Paleolithic population expansion from southwestern to northeastern Europe. *Am. J. Hum. Genet.*, 62:1137–1152, 1998.

[54] J. F. C. Kingman. The coalescent. *Stoch. Process. Appl.*, 13:235–248, 1982.

[55] F. N. Neuts. *Matrix-Geometric Solutions in Stochastic Models: An Algorithmic Approach.* Dover, 332pp, 1981.

[56] M. P. Cox. Accuracy of molecular dating with the rho statistic: deviations from coalescent expectations under a range of demographic models. *Hum. Biol.*, 80:335–357, 2008.

[57] N. Metropolis, A. W. Rosenbluth, M. N. Rosenbluth, and A. H. Teller. Equation of state calculations by fast computing machines. *J. Chem. Phys.*, 21:1087–1092, 1953.

[58] W. K. Hastings. Monte Carlo sampling methods using Markov chains and their applications. *Biometrika*, 57:97–109, 1970.

[59] C. P. Robert. *The Bayesian Choice. From Decision-Theoretic Foundations to Computational Implementation.* Springer., 486pp, 2001.

[60] A. Gelman, J. B. Carlin, H. S. Stern, and D. B. Rubin. *Bayesian Data Analysis*. Chapman and Hall., 668pp, second edition, 2004.

[61] D. M. Titterington, A. F. M. Smith, and U. E. Makov. *Statistical Analysis of Finite Mixture Distributions*. Wiley, 243pp, 1985.

[62] Y. Pawitan. *In All Likelihood: Statistical Modelling and Inference Using Likelihood*. Oxford University Press, 528pp, 2001.

[63] A. Torroni, A. Achilli, V. Macaulay, M. Richards, and H.-J. Bandelt. Harvesting the fruit of the human mtDNA tree. *Trends Genet.*, 22:339–345, 2006.

[64] P. Soares, L. Ermini, N. Thomson, M. Mormina, T. Rito, A. Röhl, et al. Correcting for purifying selection: an improved human mitochondrial molecular clock. *Am. J. Hum. Genet.*, 84:740–759, 2009.

[65] G. O. Roberts and J. S. Rosenthal. Examples of adaptive MCMC. *Stat. and Comput.*, 6:269–275, 2006.

[66] P. J. Green. Reversible jump Markov chain Monte Carlo computation and Bayesian model determination. *Biometrika*, 82:711–732, 1995.

[67] S. Richardson and P. J. Green. On Bayesian analysis of mixtures with an unknown number of components. *J. R. Statist. Soc. B*, 59:731–792, 1997.

[68] M. A. Newton and A. E. Raftery. Approximate Bayesian inference with the weighted likelihood bootstrap. *J. R. Statist. Soc. B*, 56:3–48, 1994.

[69] D. B. Dahl et al. *xtable: export tables to LaTeX or HTML*, 2007. R package version 1.5-2.

[70] A. D. Martin, K. M. Quinn, and J. H. Park. *MCMCpack: Markov chain Monte Carlo (MCMC) Package*, 2008. R package version 0.9-4.