Master thesis Artificial Intelligence

Identifying kinship relations using incomplete DNA: A Bayesian approach to determine the maximum likelihood pedigree using MCMC

Jonas Ahrendt
j.ahrendt@student.ru.nl
Student number: s4085256

Department of Artificial Intelligence
Radboud University Nijmegen
The Netherlands

November 2013

Supervisors:

M.A.J. van Gerven
Department of Artificial Intelligence
Radboud University Nijmegen

W.A.J.J. Wiegerinck
Donders Institute for Brain Cognition and Behaviour, Department of Biophysics
Radboud University Nijmegen

W.G. Burgers
Donders Institute for Brain Cognition and Behaviour, Department of Biophysics
Radboud University Nijmegen

Abstract

A method for pedigree reconstruction is proposed using Markov Chain Monte Carlo and Bayesian inference, which can reconstruct the family relations of several individuals in question, based on DNA profiles. Kinship relations are reconstructed using genetic microsatellite (STR) data from samples of related individuals. In particular, this research extends methods for pedigree reconstruction to incorporate mutations and to handle incomplete genotype samples, in which genetic profiles were either not observed for all individuals in the pedigree, or genetic profiles contain missing observations on some genetic markers. This extends pedigree reconstruction to take account for distant family relations. The algorithm is demonstrated using generated datasets and a single human dataset. The proposed method can be applied in forensic science, criminology, legal decisions, archeology, and medicine.

Contents

1 Introduction

1.1 Motivation

A pedigree represents the line of ancestors[1] and is usually depicted as a graph, also called a family tree or pedigree chart. It includes all kinship relations, e.g. the relations to one's ancestors, descendants as well as other kinship relations like cousins, nephews, great-aunts, etc. An example for a pedigree graph - as used in the context of this thesis - is depicted in figure 1.1.

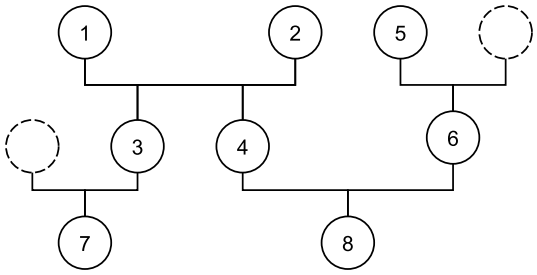


Figure 1.1: Example for a pedigree graph. Numbered circles denote individuals and edges denote the kinship relations, and dashed circles indicate the existence of a second parent that is not part of the pedigree.

Advances in forensic science enable the extraction of human DNA to identify individuals based on their genetic fingerprints. These techniques are commonly employed by forensic scientists in criminology and victim identification cases.

A more general problem is to reconstruct the full pedigree graph given the genetic profiles of a set of individuals. More specifically, pedigree reconstruction aims to determine the most likely pedigree. Problematic is the vast number of possible pedigrees, which is huge and grows exponentially in the number of individuals. Therefore, this problem is hard to solve and techniques from artificial intelligence such as Bayesian decision models and Monte Carlo methods may help.

Previous researches reported several efficient methods for pedigree reconstruction, which were limited to complete genotype samples, and many of those neglect the possibility of mutations [1, 2, 3, 4, 5, 6]. Both components are crucial for a complete solution to the problem of pedigree reconstruction, and thus are the two current challenges to be solved. Therefore, an efficient method, which can handle missing genetic data as well as mutation, is demanded.

This thesis proposes a method to reconstruct pedigrees given sets of genetic profiles, and in particular incomplete samples of genetic profiles. The method uses a Bayesian network inference algorithm to handle mutations and incomplete genotype samples, and a stochastic search method using a Markov Chain Monte Carlo (MCMC) approach inspired by the Metropolis-Hastings sampling in order to cope with the large search space.

Being able to solve pedigree reconstruction, the applicability of genetic methods extends to new fields [7]: (1) scenarios following disasters like mass graves, in which family relations are of interest [8], (2) resolving the family relations in cases, in which incest is suspected [7], (3) immigrations cases, in which a person applying for immigration claims to have relatives and authorities require trustful facts rather than the imprecise statements of forensic experts [7, 9] (4) in medical research to detect genetic risk factors of diseases [6].

1.2 Outline

The thesis is organized as follows. Relevant background information about pedigree reconstruction, genetics, and fingerprinting technologies, Bayesian methods, and those sampling methods that inspired this approach, and previous studies regarding pedigree reconstruction are presented in section 2. The proposed method is presented in section 3, which covers the developed pedigree representation, a search algorithm that traverses the exponentially large search space in a "random walk", as well as information regarding the complexity of this method. The results of the performance assessment in terms reconstruction quality and the required computational resources are presented in section 4. Finally, the conclusion of this thesis is presented in section 6, followed by a discussion in section 5.

[1] Source: http://www.thefreedictionary.com/pedigree

2 Background

2.1 Genetic fingerprinting

2.1.1 Introduction

Genetic fingerprinting (also called DNA profiling or DNA typing) allows to identify people based on their DNA profiles. These techniques are most commonly employed by forensic scientists in criminology and parental testing.

A genetic fingerprint consists of only a small portion of a person's DNA, rather than his or her complete genotype information. Genotype information is extracted from different known and selected locations on the chromosome, so-called loci, and together they combine to a DNA profile. Loci of a genetic fingerprint are carefully selected, and they intend to highlight inter-individual differences in the genotype data because about 99.7% of the human genotype information is identical [10].

In forensic DNA analysis, several technologies can be used for genetic fingerprinting. The best solution has been achieved using short tandem repeat (STR) DNA markers in terms of a high power of discrimination and a rapid analysis speed [10, pp. 4-5]. Short Tandem Repeats (STR), also known as microsatellites, are repeating sequences in DNA profiles. STR DNA markers are length polymorphisms [10, p. 26]. STR DNA became popular for human identity testing [10, p. 30] and has several advantages over other technologies. The repeat size is small and the number of repeats in STR marker is highly variable among individuals [10, p. 85].

STR marker can be distinguished into autosomal[2] markers, which are gender-independent, and lineage markers, which are gender-dependent [10, p. 201], of which only the former is used in this thesis. STR from the Y-Chromosome are only present in males and can be used to track the paternal lineage, and mitochondrial DNA (mtDNA) are only present in females (and thus can only be transferred from mother to child) and can be used to track maternal lineage [10, pp. 201-202]. This research ignores information about the gender of the individuals. Consequently, genotype data is solely based on autosomal STR marker data, which is available independent of the gender, and gender specific genotype data such as Y-STR or mtDNA are ignored.

Sets of polymorphic markers are used in genetic profiles to distinguish unrelated individuals from one another reliably or to match related individuals, so that the chance of a false match is low [10, p. 491]. For effective use of DNA typing markers (across a wide number of jurisdictions) standardized DNA typing markers are used, as for example the European SGM Plus kit, which uses 10 loci, or the CODIS (Combined DNA Index System), which consists of 13 loci and has been utilized in the United States.

An alternative to STRs are single nucleotide polymorphisms (SNPs), which are variations of single base sequences at a particular point in the genome [10, p. 182]. SNPs are more common but less polymorphic in the human genome, and thus more SNPs (compared to STRs) are required to obtain a similarly high discriminatory power [10, p. 182].

Genetic fingerprinting methods can be distinguished between direct and indirect matching cases: In the direct matching case, a person's DNA profile is matched to a reference DNA from the same person. Indirect identification uses the DNA profiles of relatives, i.e. consanguineous or kindred, which are genetically similar but not identical.

2.1.2 Applications

Forensic DNA tests can be used in criminal investigations, in particular to convict criminals or to protect innocents from wrongful convictions [10, p. 8]. In kinship analysis, indirect matching is used to review the biological relationship between mother, father and child (cf. paternity test respectively maternity test). These are simple cases, which quantify evidence for or against two alternative hypotheses, e.g. "F is the father of C" vs. "F is not the father of C" [7]. The likelihood ratio (LR) expresses the ratio between these two (mutually exclusive) hypotheses:

$$LR = \frac{H_1}{H_0} \tag{2.1}$$

where H_0 the null hypothesis and H_1 is the alternative hypothesis (cf. [10, p. 459]). This approach can be extended to choose between more than just two hypotheses, or it can be further generalized to find the kinship relations between several family members (e.g. [7]), i.e. the search for the best suitable hypothesis.

[2] Autosomal: referring to any chromosome except the sex chromosomes.

Kinship identification based on genetic marker data is relevant in diverse fields, such as conservation research, epidemiological and genealogical research, and forensic science identification problems [6]. Pedigree reconstruction can be applied in many different fields, e.g. in scenarios following disasters, to resolve family relations, in immigration cases and in medical research, and is not restricted to human DNA.

In scenarios following disasters, which also include found graves and mass graves, the family relations might be of interest and need to be reconstructed, because all knowledge about them is lost. An example is the Romanov case, in which the last Tsar of Russia and his family were shot in 1918, and investigators tried to identify the remains of the found bodies [8]. Another example is the Srebrenica massacre in Bosnia and Herzegovina in 1995[3], a genocide in which thousands of Bosnian Muslims were killed during the Bosnian war.

Disaster Victim Identification (DVI) extends the simple identification cases to a more complex problem, in which one decides between more than just two alternative hypotheses. Here the task is to identify victims by matching them with a set of missing persons. In order to perform a direct check, a DNA profile needs to be present for both, the victim and the missing person. If both match, the person is identified. For that, the DNA of the victim can usually be recorded from small samples of body remains [11].

Reliable reference material of the missing person might be unavailable. In this case, the victim can also be indirectly matched by using DNA profiles from relatives [11]. Here, the matching is performed between (1) a set of genetic profiles of the victims and (2) a set of pedigrees, which include one or multiple missing persons and their relatives. If there is only one victim and one missing person, then this problem is quite easy to solve, but in case of a mass disaster, where several or many missing persons need to be identified, the task becomes more complex. In this case, the set of proposed pedigrees to be tested may become very large, and then the task is to select the most likely hypothesis out of a large number of possible hypotheses.

The Bonaparte Disaster Victim Identification System[4] implements efficient methods to identify victims in a larger scale, e.g. after a mass disaster with many victims and missing persons. Bonaparte DVI can effectively identify victims using indirect matching, i.e. using reference DNA profiles from relatives of the missing person. In particular, given a set of pedigrees, in which each contains at least one missing person, and another set, which contains the victims, several candidate pedigrees in question are generated and the likelihood for each combination of those can be computed effectively using Bayesian Network inference methods. The Bonaparte DVI system and the hereby used Bayesian approach are described in more detail in Wiegerinck et al. (2010) [11] and van Dongen et al. (2012) [12]. Currently, this system does not enable the reconstruction of kinship relations (see pedigree reconstruction in section 2.5) based only on genetic data, a more challenging problem that is investigated in this thesis. However, the proposed method in this thesis might be employed as a component in such a system, and may provide forensic scientists with an additional tool for their investigations.

Another possible application of pedigree reconstruction is in legal cases, in which the authorities require trustful facts and may prefer to relate to a numerical quantification rather than the imprecise statements of forensic experts [7]. Jeffreys et al. (1985) for example demonstrated the applicability of genotyping technology in an immigration case, in which a person applying for immigration claimed to have particular relatives [9]. Pedigree reconstructed method can be employed to resolve family relations in cases, in which incest is suspected [7].

In medical research, large population bio-bank studies are performed in order to detect the effects of rare genes, which are of interest as they may cause the diseases of major public health concern [6]. These population studies typically lack in statistical power to detect such gene effects and usually involve sets of undeclared relatives [6]. Knowing the kinship relations may help to improve statistical power and enable the detection of gene-effects as well as the genetic risk factor causing diseases [6]. This underlines the importance of efficient methods, which use sample genetic data from large population studies, in order to reconstruct pedigrees [6].

Finally, pedigree reconstruction is not limited to the human species, also natural populations can be investigated in, e.g. the estimation of heritabilities in the wild [4].

2.2 Pedigree and graphs

2.2.1 Definition of a pedigree

A pedigree consists of individuals that are interconnected by edges, which indicate their kinship relations. In biology, every individual has parents, which would result in an infinitive chain of ancestors, so that a pedigree cannot cover the complete ancestral line for any of its individuals. Therefore, a pedigree is by definition limited to a small subset of individuals and only kinship relations between those individuals are covered by the pedigree.

Formally, a pedigree is defined as $P = (I, R)$ in which $I = i_1, \ldots, i_N$ denotes a set of individuals and R a set of kinship relations among the individuals. An individual $i \in I$ represent a single person in a pedigree for that genotype information may be available. N is the total number of individuals in the pedigree.

[3]Srebrenica massacre: see www.ic-mp.org

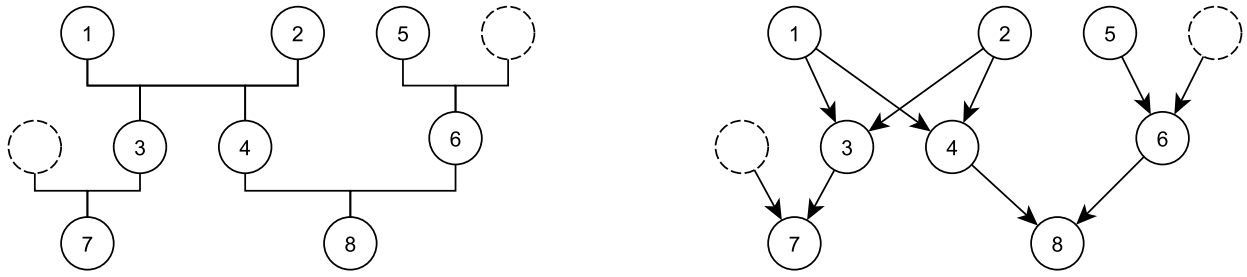
[4]Bonaparte Disaster Victim Identification System: http://www.bonaparte-dvi.com

The kinship relations R are the structural component in the pedigree graph and their collectivity defines the connectivity of the individuals in the pedigree. R denotes the binary relations between two individuals in I.

$$R \subseteq I \times I \tag{2.2}$$

A single kinship relation $r = (i_p, i_c)$ with $r \in R$ is a directed relation between two individuals $i_p, i_c \in I$ with one individual i_p being the parent and the other individual i_c being the child. This parent-child relation is directed from the parent $i_p \in I$ to the child $i_c \in I$. Further this implies that parent i_p inherited its DNA to child i_c using the principles of genetic transmission, which are introduced in the later section 2.3.3.

A pedigree can be represented as a directed acyclic graph (DAG) in which the nodes represent individuals and arcs represent the kinship relations between those individuals, and the direction each arc represents the direction of the kinship relation from the parent to the child [6, 5], see figure 2.1b.



(a) Pedigree graph

(b) Corresponding Directed acyclic graph

Figure 2.1: The same pedigree presented in two forms

Pedigrees used in this thesis do not assign a gender to individuals and both, males and females, are simply denoted as circles, cf. figure 2.1, in contrast to conventional pedigree graphs in which male individuals are represented as squares.

2.2.2 Pedigree representation based on parent sets

A representation of the pedigree structure based on parent sets was commonly used in previous studies (as for example in [5]). The parent set of an individual denotes the set which consisting of all nodes having outgoing edges to the node of the individual. Let i be an individual, $m(i)$ its mother and $f(i)$ its father then π_i denotes the parent set of individual i and $|\pi_i|$ the number of i's parents in the pedigree.

$$\pi_i = \{m(i), f(i)\} \tag{2.3}$$

In a pedigree graph, nodes with two incoming edges represent that the individual has two parents in the pedigree, i.e. $|\pi_i| = 2$. There are also individuals which only have one parent in the pedigree, and thus only one incoming arc, i.e. $|\pi_i| = 1$. For those the other parent is depicted as a dotted circle in the pedigree graph, as in figure 2.1. A special type of individual in a pedigree is the founder. Biologically, every individual has two parents but founders are those individuals, of which the parents are not included in the pedigree, i.e. $|\pi_i| = 0$. Thus, their corresponding nodes do not have incoming arcs and their parents are not part of the pedigree.

2.2.3 Requirements for biological valid pedigrees

A pedigree needs to be biologically valid. To ensure biological validity a few requirements need to be met: (1) parent-compliance, (2) age-consistency, and (3) gender-consistency.

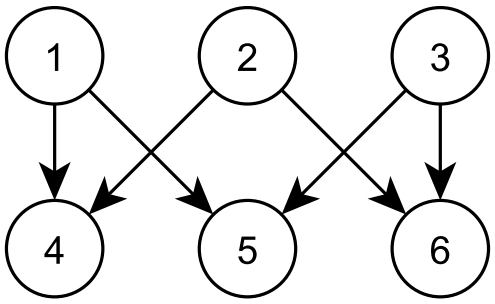
A pedigree needs to be compliant to the number of parents. Since every individual has two biological parents, which are not required to be in the pedigree as well, an individual may also only have none or just one parent in the pedigree. Therefore, every node can at most have two incoming edges representing that an individual can have 0, 1 or 2 parents, i.e. $|\pi_i| \in [0, 1, 2]$. This requirement is denoted as parent-compliance.

Biologically valid pedigrees are age-consistent, i.e. individuals cannot be their own ancestor. To take account for this, the pedigree graph is required to exclude directed cycles. Age-consistency can be assured by a applying an age ordering in among the individuals, in which parents must be older than their children [13, 1]. Cussens

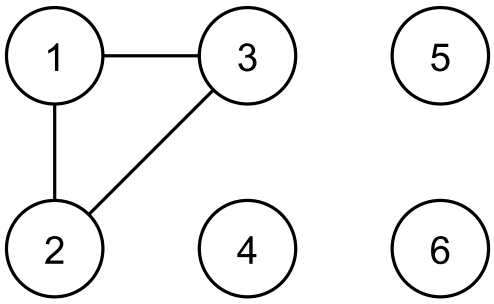
et al. (2013) employed a generation number for the same purpose [6]. Age-consistency is implied by using a directed acyclic graph (DAG) as a representation for a pedigree.

To assure gender-consistency (respectively sex-consistency as denoted by [5]) both parents need to be of opposite genders. This research is solely based on autosomal STR data and thus gender information about individuals is not incorporated. In the research of Cussens (2013), a *female*-attribute was used and parents were constrained to contain at least and at most one female [6].

Even though gender information is missing, pedigrees can still be gender-inconsistent. To detect gender inconsistent pedigrees, the so-called marriage graph can be constructed for a pedigree [14]. The marriage graph contains edges between those individuals who have a common child and thus edges represent "opposite gender relations". If the corresponding marriage graph contains cycles of an odd length, i.e. $n = 3, 5, 7, etc.$, then the pedigree is not admissible due to containing kinship relations, which would require at least two parents to be of the same gender, which is biologically not possible [14].



(a) Gender-inconsistent pedigree [6]: Even though gender information is not available, there exists no single assignment of genders to these individuals, for which this DAG also represents a gender-consistent pedigree.



(b) Corresponding marriage graph. The marriage graph contains edges between those individuals who have a common child. Individuals i_1, i_2, and i_3 form a cycle of an odd length $n = 3$ and thus the pedigree is not gender-consistent.

Figure 2.2: Assuring gender-consistency without prior information about genders. (Please note that different positions for the individuals were used in both graphs.)

2.2.4 Undirected loops in incestuous pedigrees

Pedigrees may contain incest relations, so called consanguineous marriages. They occur if two individuals in the pedigree graph share a (at least one) common ancestor and a (at least one) common child. An example for a pedigree graph containing incestuous relations is shown in figure B.12 on page 64. Such pedigree constellations are biologically valid and are represented by undirected loops in the pedigree graph, respectively inbreeding loops as termed by [14]. Hence, the associated undirected graph after removing the directions of the edges in the DAG does not contain loops. In human pedigrees, incestuous relations are relatively uncommon, but biologically possible, and variations between different cultures exist. In some other species inbreeding is more common, e.g. in many domestically bred animals [14]. To conclude, undirected loops in the pedigree graph are admissible whereas directed loops are not.

The number of ancestors doubles from a generation to the previous generation. On a large scale, *Pedigree loss* describes the phenomenon of "missing ancestors", which arises when considering the number of ancestors for an individual many generations ago, e.g. 40, which would produce a large number of ancestors, e.g. $2^{40} = 1,099,511,627,776$, which is more than the world's population. This phenomenon can be explained by the occasional mating between two "distant relatives", which is not considered as inbreeding. This reduces the number of ancestors many generations ago by reducing the breadth of the whole family tree by involving large scale directed cycles.

2.3 Genetics

Pedigrees and in particular the involved kinship relations among the individuals are responsible for the nature of the genotypes of the individuals and are governed by the laws of Mendelian inheritance. This section covers the biological foundations and introduces DNA typing and further the mathematical notations used in the course of this thesis.

2.3.1 Biological foundations and DNA typing

Deoxyribonucleic acid (DNA), also referred to as the genetic blueprint, stores the genetic information of living organisms, and provides information that determines the organisms physical attributes and can be passed to next generations through inheritance events [10, p. 17]. In human, DNA is found in the nucleus of the cells and is divided into 46 chromosomes, arranged as 23 pairs, which are further distinguished into 22 autosomal pairs and one sex determining pair of chromosomes [10, p. 20].

A *marker* refers to a known genetic location specified by a short DNA sequence. Human identity testing is usually performed using autosomal markers whereas gender determination uses the sex chromosomes [10, p. 21]. A *locus* (pl. *loci)* refers to the position or location of a gene, or a marker, on the chromosome [10, p. 22-23]. Genetic information as used in forensic investigations is observed for a number of loci. A set of observed loci is denoted as $K = \{k_1, k_2, \dots k_{|K|}\}$, in which $|K|$ is the total number of observed loci, and $k \in K$ may refer to any single non-specified locus.

Allele denotes the alternative possibilities for a gene or a genetic locus [10, p. 23], i.e. one allele is a state a particular gene can take. In the course of this thesis, a single allele is a particular value z that codes for the observed short tandem repeat (STR) count. Missing allele values are represented as zero values, i.e. $z = 0$, whereas observed alleles are represented as non-zero values, i.e. $z \neq 0$.

Chromosomes are diploid, i.e. they contain two sets of each chromosome, in contrast to gametes (sperm or egg) which are haploid until both cells combine to form a zygote, which is diploid again [10, p. 21]. Pairs of chromosomes are homologous, containing the same genetic structure and contain a copy of each gene on both chromosomes, one inherited from the mother (maternal origin) and the other one inherited from the father (paternal origin) [10, p. 23].

Thus, for every genetic marker, an individual owns two alleles, one of maternal origin, and one of paternal origin. A pair of alleles is denoted as (z_1, z_2), representing the maternal and the paternal allele. Information about the parental origin, indicating whether an allele was inherited from one's mother or father, is not available from a single genetic profile. Therefore, pairs of alleles can be ordered by value, i.e. $(z_1, z_2) : z_1 \leq z_2$.

On two homologous chromosomes, two alleles at the same locus are termed (1) *heterozygous*, if they are different, or (2) *homozygous,* if they are identical [10, p. 23]. In the used notation, this corresponds to the values of STR counts, which are either (1) heterozygous $z_1 \neq z_2$, or (2) homozygous $z_1 = z_2$. In forensic DNA typing, typing of homozygous alleles is more difficult than typing heterozygote alleles (cf. [10]).

2.3.2 Genetic data

A genotype (i.e. a single genetic information) for an individual $i \in I$ at a locus $k \in K$ is formally denoted as g_i^k, which is a pair of alleles, i.e. $g_i^k = (z_1, z_2)$. A DNA profile, also called a genetic profile, is a combination of genotypes obtained at multiple loci [10]. A genetic profile denoted as g_i specifies all allele-pair observations for an individual i and for a set of loci K.

$$g_i = \bigcup_{\forall k \in K} g_i^k \tag{2.4}$$

Genotypes may be missing. In particular, an individual i is said to be (1) *untyped,* if all the allele values in its genotype g_i are missing, i.e. $\forall k \in K : g_i^k = (0, 0)$, (2) *typed,* if all alleles in its genotype g_i were observed, i.e. $\forall k \in K, z_{1,2} \neq 0 : g_i^k = (z_1, z_2)$, or (3) *partially typed,* if alleles were observed for some but not all loci. Similarly, genetic information about a locus g^k refers to the observations of allele pairs on a single locus k for several individuals I.

$$g^k = \bigcup_{\forall i \in I} g_i^k \tag{2.5}$$

Notational remarks: As common in Bayesian theory, capital variables denote random variables, e.g. G_i, and lower case variables denote the states the variable can take, such as particular genetic profile g_i. The probability for a particular event, e.g. $G_i = g_i$, to occur is denoted with a lower case letter, e.g. $P(g_i)$ instead of $P(G_i = g_i)$. Hence $P(G_i = g_i)$ and $P(g_i)$ have the same meaning. Evidence, i.e. certainty that variable G_i is in state g_i is expressed with the probability $P(g_i) = 1$. $P(G_i)$ refers to the probability that any of its possible instantiations occurs, and thus $P(G_i) = 1$.

Genotype information g for a whole pedigree is defined for a set of individuals I and a set of loci K

$$g = \bigcup_{\forall k \in K} \bigcup_{\forall i \in I} g_i^k \tag{2.6}$$

and thus g can be represented using a two dimensional matrix of allele pairs, which consists of the genetic profiles g_i of all individuals I, i.e.

$$g = \bigcup_{\forall i \in I} g_i. \tag{2.7}$$

The corresponding set of random variables G can take particular genotype information g as states in which each G_i^k is a random variable which can take particular allele values $g_i^k = (z_1, z_2)$, see table B.2 on page 59 for an example.

2.3.3 Mendelian inheritance and Hardy-Weinberg equilibrium

The basic laws or principles of genetics were first described by Gregory Mendel (1822-1884). He described (1) the law of segregation in which gene pairs separate in their parts during sex-cell formation (meiosis) and become haploid, and (2) the law of independent assortment, which states that genes are passed independently from parent to offspring, i.e. genes are unlinked [10, p. 466-470]. Derived from that, the probability to observe a particular genotype of an individual i given the genotypes of mother $m(i)$ and the father $f(i)$ can be expressed with

$$P(g_i | g_{f(i)}, g_{m(i)}) \tag{2.8}$$

which computation is further specified in section 2.4.3 on page 11 as part of a probabilistic model.

These laws form the basis for the linkage equilibrium and the Hardy-Weinberg equilibrium [10, p. 466-470]. Under the assumption of the Hardy-Weinberg equilibrium, genotype frequencies can be computed based on allele frequencies [10, p. 466-470]. Two alleles for a gene in a population, with frequencies p and q, sum up to one [10, p. 466-470].

$$p + q = 1 \tag{2.9}$$

Combining these two alleles, the following genotype frequencies can be expected [10, p. 466-470].

$$(p + q)^2 = p^2 + 2pq + q^2 = 1 \tag{2.10}$$

The population is said to be in Hardy-Weinberg equilibrium, if the expected genotype frequencies are close to the observed genotype frequencies, and all allele combinations are assumed independent of each other [10, p. 466-470].

Allele frequencies are typically derived from population statistics. Unknown allele frequencies are reasonable to be estimated by using all genotypes, if the sample contains many individuals and small families [4].

2.3.4 Genotyping errors and mutations

Genotype datasets may contain errors, i.e. alleles between parents and offspring do not match according to the Mendelian laws [4]. These mismatches can be rooted in mutations [4]. In biology, mutations occasionally occur some at genetic loci during meioses, and cause a change of the corresponding STR repeat count. Such mutational events can be estimated by comparing the offspring DNA marker with the parents' DNA marker *[10, pp. 138-139]*. In biology, the mutation rate is rather low for most typically used STR markers, in average below 0.1% per allele transfer [10, pp. 139-141].

Another source for such errors may be genotyping errors, which can occur during the observation of an allele, i.e. during the molecular analysis [4]. This research ignores genotyping errors and assumes that genotypes are observed correctly.

2.3.5 Incomplete genotype data

Similar to individuals that are referred as typed, untyped or partially typed, genotype samples can be distinguished in complete and incomplete samples [1]. In complete samples, the genetic data is observed for all individuals in the pedigree, and each unobserved parent is assumed to be unrelated all other individuals in the sample [1, 6]. In incomplete samples, typically not all individuals have genetic data in the sample, i.e. additional individuals and their genotypes are required to explain the pedigree including distant family relations, e.g. grandparent-child relationship [1]. Another way in which samples can be incomplete is on a per-allele base, i.e. single alleles in the genetic profiles are missing. Previous research has been conducted to reconstruct pedigrees given complete samples, see section 2.6.1 on page 12. This research proposes a method that extends the applicability to incomplete samples. The experiments conducted were limited to a single untyped individual among a set of typed individuals.

2.4 Bayesian approach

Bayesian networks are well suited to be applied in kinship analysis [11, 15, 16], and represent a core component of the proposed method. Using a Bayesian network, all relevant factors in kinship analysis can be incorporated in a transparent and flexible way, in particular any knowledge about genetic information, such as evidence (e.g. in form of Short Tandem Repeat (STR) counts) and uncertain knowledge from population statistics (about allele occurrences and mutation rates). The statistical relations between the genetic profiles in a pedigree can easily be modeled using a Bayesian network [16]. Furthermore, mutations and genotyping errors, as well as missing genotype observations can be taken into account. Bayesian networks are accepted models of dealing with uncertainty [11, 17]. Hence, the handling of missing genotype data, i.e. incomplete genotype samples, becomes viable. Using Bayesian inference, the likelihood for observing a pedigree given the genotypes can be computed, and be used to search for the maximum likelihood pedigree, i.e. the pedigree that can explain the observed genotype data best. This sub-section aims to introduce Bayesian networks in general as well as its application in the context of kinship analysis.

2.4.1 Bayesian networks

Bayesian networks are probabilistic models, which model the relations between several random variables. In particular, the conditional dependencies between random variables are represented in a directed acyclic graph. Formally, a Bayesian network is a probabilistic model $\mathcal{B} = (x, D)$ using a graph x and a joint probability distribution D. The graph $x = (I, R)$ is a directed acyclic graph consisting of vertices I and directed edges R. In total, there are N vertices. Every node $i \in I$ corresponds to a random variable G_i which can take finite a number of states $g_i \in G_i$.

This is suitable to be applied in kinship analysis: The genotypes of the individuals in the pedigree can be modeled using the random variables, and the kinship relations among the individuals can be modeled by conditional dependency relations, i.e. the arcs in a corresponding graph.

In a Bayesian network, the random variables $G = \{G_1, G_2, \ldots, G_N\}$ can be distinguished in observed an unobserved variables. For observed variables, there exist knowledge about the exact state g_i of a random variable G_i, which is termed evidence, i.e. $G_i = g_i$. In contrast, knowledge about the states of unobserved variables is missing, but it can be estimated with a prior probability distribution about possible states.

In kinship analysis, these random variables are very suitable to code for the genetic information, such as observations of genetic profiles and a prior allele distribution for the population. The genotype data g represents the observed data, which can be used as evidence in the Bayesian network. This is done by assigning values to their random variable $G_i = g_i$, which gives evidence about their state. Missing genetic profiles of unrelated individuals are assumed to be distributed according to a prior distribution $P(G)$, which models the probabilities of all possible alleles to occur and it is derived from statistical observations in the population.

As a genotype profile consists of multiple markers, only the genotypes of a single locus at a time are considered in a Bayesian network. As genetic markers are chosen in a way that their inheritance is approximately independent, especially if markers are distant and lie on different chromosomes, a dependency is rather unlikely. Therefore, statistical relations between the observations of the same genetic locus across several individuals can be assumed independent. Hence, they can also be computed independently:

$$P(g) = \prod_{\forall k \in K} P(g^k) \tag{2.11}$$

Please note, that in the following description, only one locus is considered, but the same technique can be applied for multiple loci (see section 2.4.2).

The principles of genetic inheritance can be defined in a notation common in Bayesian probability theory. In Bayesian networks, the conditional dependencies between random variables are given by the graph x, respectively its edges. The parent-set $\pi(i)$ describes the set of vertices that have an edge $r \in R$ pointing towards the vertex i.

In kinship analysis, the biological parents can be modeled analogous to the definition of the parent-set in a Bayesian network, so that the dependencies between the genetic markers of related individuals, as provided by the Mendelian laws of inheritance, can be accounted for.

The joint distribution to observe the genotypes $P(g)$, which depends on the structure of the pedigree x as well as the mode of inheritance for the locus, can be factorized into the product of conditional distributions, which is defined for every individuals in the pedigree given its parents [6]. The joint probability distribution D consists of several conditional probability distributions $P(g_i|g_{\pi(i)})$ as factors.

$$P(g) = P(g_1, \ldots, g_N) = \prod_{\forall i \in I} P(g_i | g_{\pi(i)}) \tag{2.12}$$

Respectively in logarithmic representation:

$$\log P(g) = \sum_{\forall i \in I} \log P(g_i | g_{\pi(i)}) \tag{2.13}$$

In some applications (but not in pedigree reconstruction), it may be of interest to estimate the state of the unobserved variables using knowledge about the observed variables, which is possible using Bayesian inference. The problem of computing the posterior probabilities for unobserved model variables given evidence, i.e. observations of model variables, is termed probabilistic inference [11]. As evidence about model variables is acquired, the probabilities of unobserved random variables can be updated using Bayesian inference, which computes the posterior probabilities. For that, Bayes rule can be applied to inverse the direction of computation:

$$P(h|e) = \frac{P(e|h)\, P(h)}{P(e)} \tag{2.14}$$

in which h is a hypothesis and e the evidence. For all other unobserved variables, which are not of interest, it can be summed over all possible instantiations of those, to acquire the updated posterior distribution.

The Junction Tree algorithm can perform exact inference on a network of reasonable size [2]. Hence, one can efficiently handle uncertainty, i.e. unobserved variables, in a flexible way. Applied to kinship analysis, the Junction Tree algorithm enables the handling of unobserved genetic profiles [18], so that incomplete genotype samples can be dealt with.

Using Bayesian networks, the genotype information g of any arbitrary number of individuals of a pedigree can be taken into account, rather than analyzing the kinship relations between just two or three individuals [16]. A manual computation of such probabilities becomes difficult using simple formulas, if many individuals are involved, so that the scalability of an automated computation method using a probabilistic network is preferable. Moreover, if not all DNA profiles were observed, so that computation involves uncertainty, the benefits of using a Bayesian network outweigh.

2.4.2 Likelihood computation

In pedigree reconstruction, the probability distribution $P(X|g)$ is actually subject to question, which indicates the probabilities distribution of all possible pedigrees $x \in X$ given the observed genotype data g. Due to the vast amount of possible pedigree structures, computing $P(X|g)$ becomes computationally infeasible (see section 2.5.2), and only a single pedigree x can practically be considered at a time.

Instead, the likelihood of the observed genotype data g given a pedigree structure x, respectively its kinship relations, can be computed:

$$L(x) = P(g|x) \tag{2.15}$$

In the course of this thesis, $L(x)$ is simply termed the likelihood of a pedigree. The likelihood of a pedigree is a single value, which is particularly useful for the reconstruction of kinship relations, because whole sets of many pedigrees X can be tested for likelihood [7], ordered by likelihood, as well as the maximum likelihood pedigree can be determined.

Finally, in order to compute a likelihood involving multiple genetic loci, the likelihoods per each locus can be computed separately, due to the assumption of independent markers, and then be integrated afterwards [16]. Let g^k denote the genotype observations over several individuals for a single locus k then

$$P(g|x) = \prod_{k \in K} P(g^k|x) \tag{2.16}$$

specifies the likelihood for all observed loci K. To compute the likelihood $P(g^k|x)$, an inference algorithm can be used, such as the Junction tree algorithm.

Furthermore, using Bayesian networks, mutations can be taken into account, which enables changing alleles from generation to generation to be explained stochastically. The mutation model takes place in the computation of the conditional probabilities and integrated into the likelihood of a pedigree. Thus, those pedigrees, which require mutations to explain its genotype data, are characterized with a lower likelihood.

2.4.3 Conditional probabilities

The likelihood of a pedigree structure $L(x)$ can be factorized into the product of conditional probability distributions, which is defined for every individual in the pedigree given its parents [6].

$$P(g|x) = \prod_{\forall i \in I} P(g_i|g_{\pi_x(i)}) \tag{2.17}$$

respectively in log10-likelihood notation

$$\log P(g|x) = \sum_{\forall i \in I} \log P(g_i|g_{\pi_x(i)}) \tag{2.18}$$

in which $g_{\pi_x(i)}$ denotes the set genotypes of the parents. Please note, that the particular selection of parent-set π derives from the given pedigree structure x. Such a likelihoods of observing a single genotype given the parents' genotypes $P(g_i|g_{\pi(i)})$ is also termed a local likelihood, i.e. the local likelihood of i.

In the following, the computation of the local likelihood is demonstrated given that pedigree contains none, only one or both parents of i. Please note that these computations assume, that the genotypes of individuals, which are not part of the pedigree, are independent. These formulas are adapted from [6] to the here used notation.

Let i be a founder, i.e. an individual without parents in the pedigree, then $P(g_i)$ denotes the marginal probability distribution that individual i has genotype g_i. The probability distribution $P(G_i) = 1$ is distributed according to the prior allele distribution of the population.

Let i be an individual, $m(i)$ its mother, and $f(i)$ its father, then $P(g_i|g_{f(i)}, g_{m(i)})$ denotes the probability that individual i has genotype g_i given the genotypes of both of i's parents, i.e. $g_{f(i)}$ and $g_{m(i)}$.

If only one parent of i is part of the pedigree (in this case i's mother), then $P(g_i|g_{m(i)})$ denotes the probability that individual i has genotype g_i given only the genotypes of i's mother $g_{f(i)}$. In this case, the other parent $f(i)$ is assumed a founder. The probability $P(g_i|g_{m(i)})$ can be computed by summing over all possible joint observations of $G_{f(i)}$ and $G_{m(i)} = g_{m(i)}$:

$$P(g_i|g_{m(i)}) = \sum_{g_{f(i)}} P(g_i|g_{f(i)}, g_{m(i)}) P(g_{f(i)}) \tag{2.19}$$

Here, the marginal probability distribution $P(G_{f(i)})$ effectively act as weights and prefers common occurring alleles to rare alleles.

Finally, to obtain the likelihood of the whole pedigree, the conditional probabilities are integrated over all individuals and all genetic loci. It is the likelihood that the particular genotypes are observed given the kinship relations of a pedigree x.

$$P(g|x) = \prod_{\forall i \in I} \prod_{\forall k \in K} P(g_i^k|g_{\pi(i)}^k) \tag{2.20}$$

2.4.4 Alternative approaches

In this thesis, all kinship relations are modeled by parent-offspring relations (1st degree relations) which can either present or absence, and those can combine to form more complex relations (between more distant relatives).

Alternative ways to compute the likelihoods between pairs of individuals to have a particular relationship (e.g. unrelated, parent-offspring, full-sib, half-sib, first cousins etc.) can be expressed with match probabilities (cf. [10, p. 510]) or with IBD (*identity-by-descent)* coefficients [4]. Different hypotheses about the relatedness between a triplet of individuals and their genotypes can also be compared and expressed by LOD-scores [4].

2.5 Pedigree reconstruction

A more challenging problem, which is investigated in this thesis, is the problem of pedigree reconstruction. It generalizes the traditional methodology of genetic fingerprinting beyond the traditional applications, such as parental testing and direct identification, and Disaster Victim Identification. In pedigree reconstruction, information about the kinship relations is missing and thus needs to be reconstructed.

2.5.1 Problem description and definition

The pedigree identification problem considers determining the most likely pedigree among a set of possible alternatives [6]. In theory, the maximum likelihood pedigree for a set observed genetic marker data from individuals can be simply determined by considering all possible pedigree structures and computing the likelihood of observing the genotype data given each of those, and finally selecting the pedigree with the maximum likelihood.

Formally, the likelihood $P(g|x)$ is of interest. The aim is to find maximum likelihood to observe the genetic profiles g given a pedigree $x \in X$.

$$x^* = \arg\max_x P(g|x) \tag{2.21}$$

Hence, the problem of pedigree reconstruction is a graph optimization problem, in which one searches for a valid pedigree graph x^* that maximizes the likelihood $P(g|x)$. Due to the high number of possible pedigrees enumeration of all possible pedigrees becomes impractical [6, 7], even for a small number of individuals, e.g. $N = 10$.

It cannot be guaranteed that the maximum likelihood pedigree is also the true pedigree [6][5]. Marker data of unrelated individuals may falsely indicate relatedness even though they are not [5]. In such a pedigree, a higher likelihood is produced compared to a similar pedigree without this relation. Therefore, high likelihood pedigrees are also interest rather than just finding the most likely one [5]. However, the maximum likelihood pedigree can be expected to represent also the true pedigree if a sufficient amount of markers is used.

Riester et al. (2009) distinguishes between one-, two-, and multi-generation pedigree reconstruction [4]. Sibship algorithms can be used to infer full-sibling and half-sibling relationships from the genotype data, if the (partial) pedigree consists of only one generation [4]. In two generation-pedigrees, pedigrees can be classified as possible parents and offspring, if generation data is available, in order to constraint the search space [4]. Moreover, multi-generation pedigrees are harder to reconstruct as the sets of parents and offspring may overlap [4].

2.5.2 Computational complexity

The problem of pedigree reconstruction is NP-hard. It involves the search over all possible acyclic directed graphs (DAG) with at most two incoming arcs per node, which represent the kinship relations to the parents.

The exact number of such graphs is unknown to the author but the number of possible directed graphs can act as an upper boundary. Any node can possibly be connected to any two other nodes in the pedigree. Thus, the complexity of the problem was estimated as $2^{N(N-1)}$ with N denoting the number of individuals in the pedigree, respectively nodes in the corresponding graph. Hence, the number of admissible pedigrees is exponential in the number of individuals N. Interested in a lower boundary, the number of possible directed acyclic graph was expressed with the recurrence relation by R.W. Robinson (1977) [19]:

$$a_n = \sum_{k=1}^{n} (-1)^{k-1} \binom{n}{k} 2^{k(n-k)} a_{n-k} \tag{2.22}$$

Further researches were done by A. Piccolboni and D. Gusfield (2003) who focused on the computational complexity in pedigree analysis [14]. The root of the pedigree reconstruction problem is considered as NP-hard, i.e. a solution is non-determinable in polynomial time. No such algorithm in known, which can solve the problem in polynomial time, and thus the problem is considered as computationally intractable.

2.6 Related researches

In the scope of this research, a literature study was conducted to get aware of all relevant researches regarding the problem of pedigree reconstruction. A lot of research has already been done to solve pedigree reconstruction for completely observed genotype data, i.e. the genotype profiles of all individuals in the pedigree are available prior the reconstruction.

2.6.1 Methods for complete data

The first maximum likelihood approaches in pedigree analysis were developed E.A. Thompson (1976, 1986) [20, 21]. T. Egeland (2000) determines the most probable pedigree given a certain set of data, which includes the genotypes and may incorporate other data such as age and gender information about the individuals [7].

[5] Original citation from Thompson (1976) A paradox of genealogical inference.

Their method by involves (1) selecting the set of relevant pedigrees using (1a) distinction between possible parents and non-parents and (1b) gender information, (2) using prior probabilities, and (3) obtaining the posterior probability distribution using genotype data and a mutation model [7] (which uses the likelihood computation is described in [22]). Finally, their method was employed in the software *familias* [7].

A. Almudevar (2003) presented a method for pedigree reconstruction for completely observed genetic data with simulated annealing [1]. His method can estimate the maximum likelihood pedigree with minimal error given fully observed DNA profiles data and further the possibility to incorporate additional information such as age or gender but are not required[1]. Effectively this method searches only in the space of all possible age-orders between the individuals to reduce the search space [1]. In particular, given an age-order the pedigree is divided into smaller subsets, for which the maximum likelihood can effectively be enumerated, and combines them to a maximum likelihood pedigree [1]. This latter idea was adapted in this thesis for enumeration using complete samples (see 3.8).

Simulated annealing is known to be an effective technique for problems like the Traveling Salesman Problem (TSP), in which the domain of optimization is a permutation space [1]. Such approaches were used in the before-mentioned research by A. Almudevar (2003), as well as by Riester et al. (2009). Both used simulated annealing to find the maximum likelihood pedigree [1, 4].

T. Lin et al. (2006) enhanced the method of indirectly matching victim DNA to family DNA in a few aspects, particularly in robustness [2]. It (1) clusters samples, i.e. identification of identical genotype data taken from different body remains, which originate from the same person, (2) conservatively eliminates implausible sample-pedigree pairings, so that only forensically satisfactory conclusions can remain, i.e. having a low likelihood of being wrong, (3) handles degraded samples, i.e. missing values in DNA profiles, and (4) errors during genetic fingerprinting, i.e. during production of genetic material [2].

Another search using complete samples of STR genotype data was developed by R.G. Cowell (2009), whose reconstruction algorithm is based on Bayesian network learning using dynamic programming [3]. The method is highly efficient but it does not consider mutations (and other genotyping errors), making the problem easier to solve [3]. Cowell's methods [3] has complexity $O(n^3 2^n)$ in the number of individuals n, and finding the maximum likelihood pedigree is feasible for up to 30 individuals. On the one hand, it was claimed that the algorithm could guarantee that the maximum likelihood will be found, but on the other hand, it the algorithm occasionally finds biologically invalid pedigrees, which was reported as a problem [3]. This approach was extended by Tian et al. (2010) to find the k-highest likelihood pedigrees using a structures learning Bayesian network [23]. Similar to other studies, this approach can optionally incorporate age and gender information.

Riester et al. (2009) developed a software implementation (*FRANz*) for pedigree reconstruction, which uses local probabilities about parent-offspring relations as well as sibship, and takes into account genotyping errors [4]. The method uses the simulated annealing approach, as described in A. Almudevar (2003)[1], and the method described by R.G. Cowell (2009) [4, 3]. Their method [4] cannot guarantee the maximum likelihood pedigree to be found [6]. However, it incorporates changing beliefs about the allele distribution during the search process and allows missing genotype data for unobserved parents, whose allele values were estimated using Gibbs sampling, a variant of MCMC [4].

2.6.2 Greedy search

Greedy algorithms reconstruct the pedigree sequentially. Starting from the assumption that all individuals are unrelated, such an algorithm gradually accepts kinship relations, which increase the overall observed likelihood of the pedigree, finally resulting in a single high likelihood pedigree. Greedy algorithms lack in "getting trapped in local maxima" and cannot guarantee to find the global maximum.

One such greedy algorithm was developed by R.G. Cowell (2013) [5], which is limited to complete genotype samples of related individuals, and does not consider mutations and other genotyping errors. As typical for a greedy method, it can find high likelihood pedigrees but cannot guarantee the maximum to be found [5]. Similar to this research, his algorithm uses STR data. Age and gender information is not required but can be used to constrain the search space [5]. In particular, his algorithm uses a partition procedure to create new candidate pedigrees from pedigrees (Kruskal's algorithm to find the maximum weight spanning tree in an undirected graph) and a local likelihood score, which expresses the conditional probabilities of individuals to have particular sets of parents [5]. His algorithm was demonstrated using a human, non-human and simulated datasets.

2.6.3 Constraints

Several constraint-based methods could successfully be applied to pedigree reconstruction assuming complete samples. Pedigree reconstruction method can use constraints, such as known relationships, age and gender to constrain the search space [4, 6, 5].

Constraint-based approaches can be distinguished into hard- and soft-constraints. Hard constraints include all knowledge with is known, with certainty (i.e. evidence), such as known relationships, age or gender information. In contrast to that, soft constraints also contain vague knowledge, i.e. uncertainty, such as the number of generations, cultural preferences about promiscuity or inbreeding. This section solely covers hard-constraints. The use of soft-constraints is covered in the subsequent section.

Structural prior information such as known relations in a pedigree can limit the search space, so that all proposed pedigrees, which do not contain any of the known kinship relations, are excluded from the search space.

Available information about the age of individuals in the sample can constrain the search space. Admissible pedigrees, as described above, are limited to a valid age-order in between the individuals, in which parents necessarily must be older than their children are. Evidence about the age of individuals constrains the set of valid age-orders, which effectively removes the possibility of individuals to have younger parents, and thus reduces the search space.

For age information to be useful, the knowledge about the exact age is not required. Instead, the relative age among the individuals is sufficient to infer age-constraint between any pair of two individuals. R.G. Cowell (2013) suggests that more refined age constraints, such as a minimum age gap between parents and offspring, can constrain the set of possible parents per individual [5]. T. Egeland et al. (2000) further suggests excluding young individuals, such as children, as possible candidates for being a parent based on age, which constraint is incorporated in the *familias* software [7].

Besides age information, also information about the gender of the individuals can constrain the search space. It restricts the search space in a way that both parents are required to be of opposite genders.

Cussens et al. (2013) uses a constraint-based integer linear programming (ILP) approach for the pedigree reconstruction problem, which is guaranteed to find the maximum likelihood pedigree. The approach incorporated constraints involving age and gender. The method is highly efficient and can find pedigrees for more than 30 individuals, and in the paper example for up to 65 individuals are presented [6]. Further, it can find the k-th highest likelihood pedigrees after adding further constraints (which exclude previously found pedigrees) [6]. The method guarantees maximum likelihood pedigree but does not take into account mutations and incomplete genotype samples.

2.6.4 Prior probabilities

Besides constraints, also knowledge, which is not known with certainty, can be used as structural prior information about the pedigree. In contrast to the before-mentioned constraints, prior probabilities do not exclude pedigrees entirely from the search space. Instead, they assign a lower likelihood some pedigrees, which are considered as unlikely.

In their method, Egeland et al. [7] incorporated non-DNA evidence, such as prior information about inbreeding[6], promiscuity[7] and the number generations[8] in the computation of the posterior probabilities. Both, Cussens et al. (2013) and Cowell et al. (2013), also suggested the use of additional information, such as an average sibship size, a typical generation gap and the age disparity amongst the parents, to be used as prior information [6, 5].

Incorporation of structural prior into a search is straightforward, if those priors can decompose into simple local factors, i.e. prior information per-individuals [5].

2.6.5 Demand for methods to handle incomplete samples

Current solutions only consider complete genotype samples, and thus limiting the reconstructed pedigrees to involve only close family relations between two typed individuals. The incorporation of distant-family relations is demanded to provide a complete solution to the problem of pedigree reconstruction. For a set of individuals, for which genotype data was observed, additional individuals may be required to explain distant family relations, i.e. all kinds of family relations beyond the elementary kinship relations between a parent and its offspring, such as sibship, grandparent-grandchild, cousins, aunt-nephew, etc. To the knowledge of the author, no such methods were reported yet, which face pedigree reconstruction using incomplete genotype samples, and thus this research intends to fill this gap.

A common approach in pedigree reconstruction for completely observed genotype data is to determine the most likely parents of each individual in the sample, as e.g. [1, 4]. For incomplete genotype samples, this approach is not feasible. In order to account for the added uncertainty due to missing observations, the likelihood computations need to consider all individuals and their kinship relations as whole, rather than just the kinship relations of a single individual of the pedigree at a time.

[6]Inbreeding refers to *"the number of children where both parents have a common ancestor in the pedigree"*[7]
[7]Promiscuity refers to *"the number of pairs having precisely on parent in common"* [7]
[8]Generations refers to *"the number of different generation in the pedigree"* [7]

2.7 Sampling methods

In order to reconstruct pedigree structures, Monte Carlo methods can be used, in particular the Metropolis-Hastings algorithm. Therefore, an introduction to the underlying sampling methods is provided in this section.

2.7.1 Monte Carlo methods

Monte Carlo methods are widely used for sampling, but in this research, they were adapted and used for search. D. MacKay (2003) presented a solid introduction on both techniques, which is summarized here [24]: Monte Carlo methods allow generating samples from a high-dimensional probability distribution $P(x)$ based on random numbers. The probability distribution $P(x)$ is also called the *target density*. If x is a high-dimensional vector with N dimensions, direct sampling from $P(x)$ is difficult. The probability distribution $P(x)$ can be complex and may contain various regions with different densities, so that the expectation cannot be evaluated by exact methods.

Monte Carlo sampling assumes that a density $P(x)$ can be computed at least within a multiplicative constant Z. For that another function $P^*(x)$ can be evaluated at a discrete points x.

$$P(x) = P^*(x)/Z \tag{2.23}$$

However, the normalizing constant Z remains unknown.

$$Z = \int d^N x \, P^*(x) \tag{2.24}$$

Determining Z is difficult for many dimensions N and requires enumeration over all states of x. Even if $P^*(x)$ is easy to evaluate, determining $P(x)$ remains hard as sampling from $P(x)$ requires knowledge about the densities of the regions, which can usually only be acquired by visiting all possible states x.

Using a proposal density, e.g. $Q(x)$, from which samples can be generated, can help to guide the sampler to regions of interest, i.e. to regions of high density. There are several algorithms, which intend to cope with this problem, such as the Metropolis-Hastings algorithm, which is explained in section 2.7.3. Other algorithms include importance sampling, or Gibbs sampling.

2.7.2 Markov Chain

The Metropolis-Hastings algorithm uses a Markov Chain, which is introduced in the following. A Markov chain consists of states, which can undergo transitions to traverse within the chain. In a Markov process a sequence of states x is generated in which each sample $x^{(t+1)}$ is dependent on the previous state $x^{(t)}$. This property also holds for the probability distributions for each sample, i.e. each sample $x^{(t)}$ has a probability distribution which depends on the previous value $x^{(t-1)}$. Therefore, it is required to run the Markov chain for a considerable time, i.e. until it has converged, in order to generate samples from $P(X)$ that are effectively independent. Assessing when the MCMC method has converged is another difficult problem.

Markov Chains can have several properties such as aperiodicity, irreducibility, recurrence and ergodicity. Irreducibility refers to the property, that there exists a path from any configuration x to any other configuration y with a non-zero probability. In contrast, a reducible Markov Chain contains two or more subsets of states that cannot be reached from one another [24, p. 385]. A Markov Chain is termed (positive) recurrent if it is possible to return to a state $p \in P$ with a non-zero probability after $n \geq 1$ transitions. The length of the path is n where n can take any value. Aperiodicity refers to the fact, that there exists a path from any configuration x to any other configuration y with a length greater than a number $n > n(x, y)$.

In a reversible Markov chain, the detailed balance criterion is satisfied [24, p. 386], i.e. the probability to reach a state x' from another state x is equal to the probability of the reverse transition.

$$P(x)P(x|x') = P(x')P(x'|x) \tag{2.25}$$

A reversible Markov chain implies that the distribution $P(X)$ is invariant. To design a Markov Chain for Monte Carlo methods, the distribution $P(X)$ is required to be invariant and the Markov Chain is required to be ergodic, which effectively means that it is irreducible and aperiodic [24, p. 385]. The path along the states is based on randomness and is described as a "random walk".

2.7.3 Metropolis Hastings algorithm

One Monte Carlo Markov Chain method is the Metropolis Hastings algorithm and is described by D. MacKay (2003) as follows [24]: In the Metropolis-Hastings algorithm the proposal density $Q(x'; x^{(t)})$ depends on the current state $x^{(t)}$. Knowledge about $P(x)$ is not required and the proposal density $Q(x'; x^{(t)})$ is not required be similar to $P(x)$ in order that the algorithm useful in practice, in contrast to other MCMC methods, such as e.g. importance or rejection sampling. Assuming that $P^*(x)$ can be evaluated for any x, new proposal states x' are generated from the proposal density $Q(x'; x^{(t)})$, and then either accepted or rejected depending on the quantity a

$$a = \frac{P^*(x')}{P^*(x^{(t)})} \frac{Q(x^{(t)}; x')}{Q(x'; x^{(t)})} \tag{2.26}$$

The new state is accepted directly, if $a \geq 1$. Otherwise x' is accepted with probability a. Accepting the state sets $x^{(t+1)} = x'$ and rejecting a proposed state sets $x^{(t+1)} = x^{(t)}$. As all generated samples $x^{(t+1)}$ are dependent on the previous state $x^{(t)}$, the algorithm cannot draw independent samples from the target distribution P and successive samples are correlated.

By running the Metropolis-Hastings algorithm for a considerable time, the generated samples become effectively independent to one another. With an increasing number of sampling steps, the created distribution, i.e. a histogram of all generated states, converges towards the desired distribution $P(X)$. Hence, the algorithm can eventually be used to approximate the desired probability distribution.

2.8 Research goal

Similar to previous studies, this research intends to find an answer to the central question:

1. How can a pedigree be reconstructed, if only information about the DNA profiles of several individuals is available?

This question is targeted by the development of an algorithm, which is proposed, to solve pedigree reconstruction, in particular because the problem of pedigree reconstruction is not completely solved yet. Previous studies addressed the problem of pedigree reconstruction, but did not take into account the possibility of incomplete genotype samples, and some did not consider mutations. Considering only complete genotype samples limited previous approaches of pedigree reconstruction to identify only the family relations between close or direct relatives. Thus, those distant family relations that require additional unobserved individuals were ignored. Another current challenge is the handling of genetic mutations, which were not addressed by many previous studies. Without considering mutations, the search space can be constrained efficiently, making the problem easier to solve. In contrast, considering mutations enables to explain any possible pedigree constellation with a non-zero probability, so that it gets harder to find the maximum likelihood pedigree efficiently.

Striving forward to find a complete solution for pedigree reconstruction, it aimed to develop a method, which extends its applicability to deal with incomplete genotype data as well as mutations. The method uses a Bayesian approach to compute the likelihood and a MCMC approach to search the space of possible pedigrees.

2. How suitable is the proposed search algorithm, which uses a Bayesian approach and Markov Chain Monte Carlo (MCMC) methods, and takes account for mutations, to reconstruct pedigrees based on incomplete genotype samples?

Regarding the computational complexity, the problem is considered NP-hard, and thus is non-determinable in polynomial time. Therefore, it is of general interest to find a method, which can effectively reconstruct pedigrees. It is subject to question how effectively this method can find the maximum likelihood pedigree:

3. How efficient can the proposed method, using a Bayesian approach and MCMC methods, reconstruct pedigrees based on incomplete genotype samples?

The proposed approach intends to incorporate distant family relations in pedigree reconstruction, which is not restricted to a particular kind of distant family relations (such as sibship, or grandparent-child). Given a set of genotype samples, additional untyped individuals can explain distant family relations beyond the direct kinship relations. Moreover, the proposed method can make use of all available genetic information, even if some genotype profiles are incomplete and contain missing allele values, e.g. in the case of degraded DNA, if removing the affected genetic profiles or genetic markers may not be an appropriate alternative.

3 Proposed method

This section describes the developed method to reconstruct the maximum likelihood pedigree. The input of the pedigree reconstruction problem is the genotype information for a set of individuals. The output is a maximum likelihood pedigree or a list of the most likely pedigrees, which best explains the observed genotypes in a stochastic way, respectively.

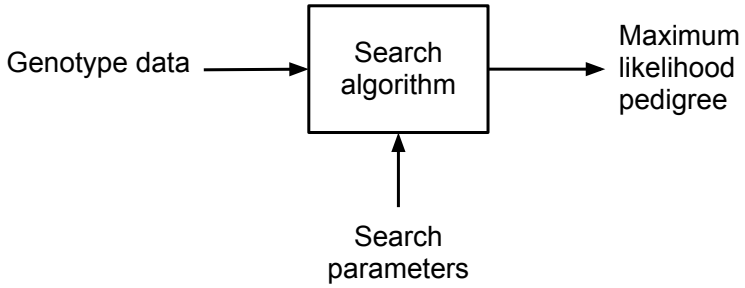


Figure 3.1: Overview of the proposed method

This section is organized as follows. First, assumptions are presented which need to be made prior to applying the here proposed method. Second, the search algorithm is described and subsequent sections cover more in-depth information about the several subroutines of the search, such as the likelihood computation and pedigree generation.

3.1 Assumptions

A couple of assumptions were made about the statistical relations between the genetic profiles. These are very similar to those made in previous researches regarding pedigree reconstruction (cf. section 2.6.1 on page 12).

The principles of genetic transmission according to the laws of Mendelian inheritance are assumed, in particular the probabilities of gene transmission from parents to offspring. All genotype frequencies are assumed to be distributed according to the Hardy-Weinberg equilibrium. A uniform allele distribution derived from population statistics is assumed which does not change between generations. Further, this method assumes that genetic markers are independent, i.e. unlinked genetic loci.

Parents of individuals, who are not in the pedigree, are assumed unrelated to one another. In particular, this affects founders that are assumed unrelated, but also for individuals who only have one parent in the pedigree the other parent is assumed unrelated to all other founding individuals in the pedigree.

Genotype samples, which are subject to pedigree reconstruction, are assumed to be observed correctly. This only concerns genotyping errors, i.e. errors made in the molecular analysis of the DNA, and should not be confused with mutations. The possibility that mutations occur is incorporated in the approach in order to obtain a realistic model with can cope with all relevant biological aspects.

As the literature study showed up, many other researches relevant in pedigree reconstruction assumed completely observed marker data (such as [1, 13, 3, 5, 6, 23]). In reality, completely observed marker data cannot be taken as granted. Hence, this method drops this assumption and the here proposed method can also be applied if genetic data was not completely observed across all related individuals. This includes both cases, in which complete genetic profiles are missing, or cases in which only some of the genetic markers were not observed. Experiments in this research were limited to only a single untyped individual at a time but the proposed method can also be applied if more than one genetic fingerprint is missing.

3.2 Search algorithm

The algorithm starts at a random pedigree x, the current pedigree. Iteratively, the algorithm proposes a new pedigree x' based on the current pedigree x. If a proposed pedigree seems promising regarding a high likelihood, then the algorithms replaces its current pedigree x with the proposed pedigree and continues proposing pedigrees, and so forth. In that way, the search algorithm traverses within the search space. The pseudo-code is provided in figure 3.2.

```
Initialize current pedigree x randomly
Initialize an empty list
loop
        x' = propose_pedigree(x)
        if likelihood(x') > likelihood(x)
                x = x'
                add x to list
        else
                probability p = likelihood(x') / likelihood(x)
                r = random number between 0 and 1
                if r > p
                        x = x'
                        add x to list
                end if
        end if
end loop
sort list by likelihood (descending)
max = list(1)
```

Figure 3.2: Pseudo-Code for the proposed method

The trajectory of the used Metropolis-Hastings algorithm is termed a *random walk*. It can be argued that this behavior is not random, because the *random walk* prefers to walk in directions of a higher likelihood. However, the proposals are based on randomness. The likelihood at each step can be printed in a trace plot as in figure 3.3 and is indicated by the red line.

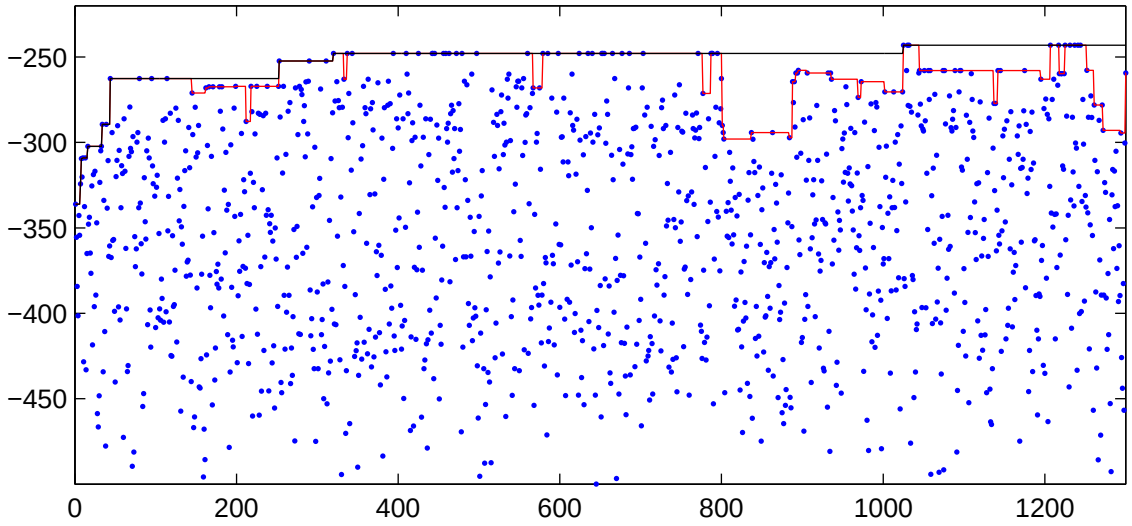


Figure 3.3: A trace plot depicting a typical search for the maximum likelihood pedigree using MCMC. The blue dots represent the proposals whereas the red line indicates the likelihood of the *current pedigree*, i.e. last accepted proposal. The solid black line denotes the most likely found pedigree at each step.

3.2.1 Markov chain

Markov Chain was constructed by defining the pedigree representation (in section 3.3), which models the states $x \in X$ of the Markov chain, and the generation of proposals in the neighborhood of a current pedigree (see section 3.5), which models the transitions of the Markov chain. The Markov Chain has the properties of aperiodicity, irreducibility, and positive recurrence.

Every state $x \in X$ has a non-zero probability of transitioning into another state $x \in X$. By using multiple transitions, there is a path so that every state $y \in X$ can be reached from any other state $x \in X$. The search space cannot be separated into several classes and t iconsists of a single class of all states X between which transitions occur. Thus, the Markov Chain is not reducible. This property is of the Markov Chain is termed irreducibility.

Further, any state $x \in X$ has a non-zero probability of transitioning into the same state $x \in X$, i.e. remaining in the state, so that there exists a path from any state $x \in X$ to any other state $y \in X$ with any positive length $n \geq 1$ with $n \in \mathbb{N}$. A return to the same state may occur at irregular times with the path length n being any multiple of $k = 1$. This property of the states in the Markov Chain is called aperiodicity. Since every state in the chain is aperiodic, the whole Markov Chain is termed aperiodic.

Finally, every state $x \in X$ can be reached from any other state $y \in X$, and the Markov chain is said to be ergodic, which requires reducibility and aperiodicity.

Furthermore, the constructed Markov chain is positive recurrent because it is possible to return to state $p \in P$ with a non-zero probability after $n \geq 1$ transitions.

The reversibility of the Markov Chain depends on the particular transition probabilities. One of the search strategies, which is described in 3.6.1, represents also a reversible Markov chain and fulfills the detailed balance criterion. In that case, the transition probabilities of the Markov Chain are invariant.

3.2.2 Acceptance ratio

The acceptance ratio decides about the probability to which a generated proposal step is accepted, respectively rejected. To compute the acceptance ratio a, the formula as presented in equation (2.26) can be used, in which the latter part is unity, i.e.

$$Q(x^{(t)}; x') = Q(x'; x^{(t)}) \tag{3.1}$$

because the chosen transitions are symmetrical (detailed balance) and the two probabilities (1) $Q(x^{(t)}; x')$, i.e. the probability to reach state x' from $x^{(t)}$, and (2) $Q(x'; x^{(t)}$, i.e. the probability to reach state $x^{(t)}$ from x', are equally likely. Therefore, the remainder of the formulas is

$$a = \frac{P^*(x')}{P^*(x^{(t)})} \tag{3.2}$$

For appliance in pedigree reconstruction, the acceptance ratio is scaled exponentially by a factor c to compensate for magnitude changes when using different inputs. This leads to an acceptance ratio of

$$a = \left(\frac{P^*(x')}{P^*(x^{(t)})} \right)^c \tag{3.3}$$

respectively when using log-likelihoods

$$a = exp \left[\left(\log P(x') - \log P^*(x^{(t)}) \right) \times c \right] \tag{3.4}$$

where c is the scaling factor

$$c = \frac{f}{|K| \times N} \tag{3.5}$$

in dependence of a parameter f, the acceptance factor, which is normalized by the number of loci $|K|$ and the number of individuals N in order to compensate for different magnitudes of the likelihood.

For explanation: Changes of the input in the number of individuals N or the number of loci $|K|$ yield to different magnitudes of the likelihood, e.g. a small pedigree ($N = 3, |K| = 20$) yields to a maximum likelihood of around $L = 10^{-110}$, whereas a larger pedigree ($N = 9, |K| = 20$) yields to a maximum likelihood of around $L = 10^{-310}$. In order to develop a general method, which is independent of the input size, the factor c compensates for changing N and $|K|$. The choice of an acceptance factor of $f = 50$ showed robust performance across different experimental settings (see section 4.8.4).

3.2.3 Termination

The search can find high likelihood pedigrees, but does not have a well-defined goal state, as knowledge about the solution is not available, e.g. maximum likelihood pedigree respectively the k-highest likely pedigrees. Therefore, alternative termination criteria are required to halt the search.

First, the most simple stop criterion is to stop the search after a specific number of steps, e.g. 1000. For that, one requires a good estimate about how many steps are required to find the maximum likelihood pedigree with

confidence. Estimates about the confidence can be made using simulation studies, which use generated genetic data.

Second, the search can be halted if converged after not finding any higher likelihood pedigree for a specific number of steps t. This happens if the likelihood of the best found pedigree, i.e. the one with the highest likelihood so far, remains constant and did not improve further. This stop criterion is adapted from the Metropolis-Hastings sampling, which is guaranteed to converge to the stationary distribution. Similarly, the maximum likelihood pedigree is guaranteed not to improve further once it converged.

Third, a specific target likelihood can be given. In that case, the search stops once a pedigree with a likelihood greater or equal than the target likelihood is found. Forth, a specific target pedigree can be given so that the search stops once this pedigree is found. This can be extended to a set of pedigrees, so that the search stops once all of the pedigrees in a specific set are found. The latter two stop criteria are rather artificial, as they require knowledge about the solution to be found. Therefore, these criteria are not applicable in real scenarios, but they were applied in some experiments in this research (see section4).

3.3 Pedigree representation

The set of admissible pedigrees X is defined which also represents the search space, i.e. the pedigree space containing $x \in X$. For a pedigree $x = (I, R)$ in which R represents the kinship relations of an admissible pedigree.

In the following, the used representation to model the kinship relations in a pedigree is presented.

3.3.1 Graph representation

In the most simple form, the relations R of a pedigree $P = (I, R)$ can be represented in a (0,1) matrix m in which each element represents a possible single kinship relation $m_{c,p} \in \bar{x}$ with $c, p \in I$, which can be either present (1) or absent (0).

$$m_{p,c} = \begin{cases} 1 & (i_p, i_c) \in R \\ 0 & (i_p, i_c) \notin R \end{cases} \tag{3.6}$$

The matrix m can represent the structure of any directed graph. The axes of the matrix, columns and rows, correspond to the set of individuals and thus the matrix has size $N \times N$. An example for such a matrix is depicted in figure 3.4 and a corresponding graph in figure 3.5.

$$\bar{x} = \begin{Bmatrix} 1 & 0 & 1 & 1 & 0 \\ 0 & 0 & 1 & 0 & 1 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 1 & 1 & 0 & 0 \\ 1 & 1 & 0 & 0 & 0 \end{Bmatrix}$$

Figure 3.4: The matrix $\bar{x}$ represents the connectivity of a directed graph.

The matrix m may contain invalid kinship relations, such as relations to oneself, i.e. $m_{i,i} = 1$, which are represented with a 1 along the diagonal of the matrix, or directed cycles (e.g. a directed cycle of length 1: $m_{i,j} = 1 \wedge m_{j,i} = 1$), or matrices, which represent directed graphs that have three or more incoming arcs in one node, and thus violent the requirement for parent-compliance. In order to draw samples from X, an appropriate representation of a pedigree is necessary which will be introduced in section 3.3.2.

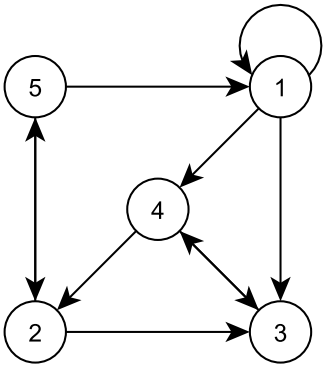


Figure 3.5: The corresponding graph for the matrix m from figure 3.4. This graph is not a directed acyclic graph and thus cannot represent a pedigree, which is required to be biologically admissible.

3.3.2 Admissible pedigree representation

This representation of pedigrees was developed with respect to several properties. In particular, the chosen representation facilitates (1) completeness, i.e. it can represent all possible kinship relations, (2) biologically validity, i.e. it only represent biologically admissible pedigrees, (3) transitions in the search space, i.e. the space of all admissible pedigrees. For the latter a notion of vicinity between pedigrees puts similar pedigrees closer together in the search space

The matrix m as introduced in section 3.3.1 on the previous page is not an appropriate representation for pedigrees, as it is not limited to biologically admissible pedigrees. Therefore, a pedigree matrix m_a is defined, which only represents valid kinship relations R among the individuals I of a pedigree $P = (I, R)$. This notation of a pedigree matrix is similar to the one used in by A. Almudevar (2003) [1]. Reconsidering the requirements for a biologically admissible pedigree there are parent-compliance, age-consistency, and gender-consistency.

The representation of the pedigree matrix m_a is initially based on the previously described (0,1) matrix m but extended by constraints which are explained subsequently. These are (1) assuring parent-compliance, (2) introducing an age ordering, (3) making the matrix dependent on the order, and (4) assuring age-consistency, respectively acyclicity when considering the corresponding directed graph.

First, parent-compliance requires that every individual in the pedigree can have at most two parents in the pedigree. To assure this, the pedigree matrix m_a can contain at most two 1's per column. This assures that all kinship relations represented by the pedigree matrix m_a are biologically valid concerning the number of parents. Now every column of the matrix m_a corresponds to the parent sets of an individual.

Second, to yield age-consistency, an age ordering $a \in A$ is defined on the set of individuals I, e.g.:

$$a = (i_3, i_1, i_4, i_5, i_2) : age(i_3) > age(i_1) > age(i_4) > age(i_5) > age(i_2) \tag{3.7}$$

Elements occurring earlier in the order represent the older individuals, and later occurring elements the younger individuals. The age ordering constrains the set of admissible kinship relations such that in every kinship relation the parents are required to be older than their children are.

$$\forall r \in R \Rightarrow age(i_p) > age(i_c) \tag{3.8}$$

with $r = (i_p, i_c)$ denoting a kinship relation, $i_p, i_c \in I$ denoting the involved individuals, and $age(\cdot)$ denoting the age of the individuals. Transitivity implies acyclicity, i.e. one cannot be its own ancestor. This also excludes the possibility that individuals can be their own parent. In total there are $N!$ possible age orderings.

Third, the pedigree matrix is aligned using an age ordering. The age ordering a defines how individuals are arranged along the axes of the pedigree matrix m_a. This is the reason why a pedigree matrix m_a is denoted with the subscript a. At this stage, a pedigree matrix is termed the extended pedigree matrix $\hat{m}_a$ which will be explained in subsequent section 3.3.3.

$$\hat{m}_a = \begin{Bmatrix} 1 & 0 & 1 & 1 & 0 \\ 0 & 0 & 1 & 0 & 1 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 & 0 \end{Bmatrix}$$

(a) Example for an extended pedigree matrix $\hat{m}_a$, using age ordering $a = (i_1, i_2, i_3, i_4, i_5)$. The matrix represents the connectivity of a directed graph with at most two incoming arcs.

$$\hat{m}_a = \begin{Bmatrix} 0 & 0 & 1 & 0 & 0 \\ 1 & 1 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 \\ 0 & 1 & 0 & 0 & 1 \\ 1 & 0 & 0 & 1 & 0 \end{Bmatrix}$$

(b) Another example for an extended pedigree matrix $\hat{m}_a$ which uses a different age ordering $a = (i_3, i_1, i_4, i_5, i_2)$.

Figure 3.6: Two examples for extended pedigree matrices. Both matrices represent the same kinship relations but using different age-orderings.

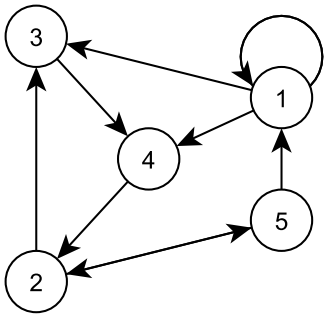


Figure 3.7: The corresponding directed graph for the extended pedigree matrices from figure 3.6a and figure 3.6b. It represents the connectivity of a directed graph with at most two incoming arcs. Furthermore, the graph incorporates the age ordering $a = (i_3, i_1, i_4, i_5, i_2)$ by depicting nodes corresponding older individuals more towards the top than the nodes representing younger individuals.

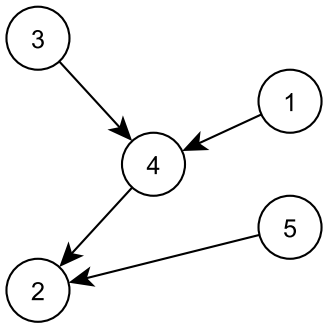


Figure 3.8: Example for imposing an age ordering on a directed graph in order to achieve acyclicity. The same graph as in figure 3.7 after removing non-admissible arcs which violent the age-ordering $a = (i_3, i_1, i_4, i_5, i_2)$.

Forth and last, to obtain an admissible pedigree matrix, an age order a is enforced in the pedigree matrix m_a. The resulting matrix m_a consists of only upper triangular elements (without the diagonal). For that, all elements along the diagonal are set to 0 because one cannot be its own parent, and all elements in the lower triangular matrix are also set to 0 because individuals cannot be their own ancestors.

In a corresponding directed graph this operation is equivalent to removing all arcs which point up, i.e. from a younger individual towards an older individual so that the younger individual, e.g. as in figure 3.8, as well as removing all arcs which point from and to the same node (self-cycles).

Finally, only the upper triangular part of the matrix (excluding the diagonal) remains, and may contain 0's or 1's as elements. Thus, the lower triangular part of the matrix can be omitted in the denotation. An example for an admissible pedigree matrix m_a is shown in figure 3.9a. This constraint implies that the resulting graph is acyclic, i.e. it does not contain cycles. In that progress all invalid arcs are removed from a corresponding graph, e.g. compare matrices figure 3.6b and figure 3.9a, or matrices figure 3.6b and figure 3.9b, respective their graphs.

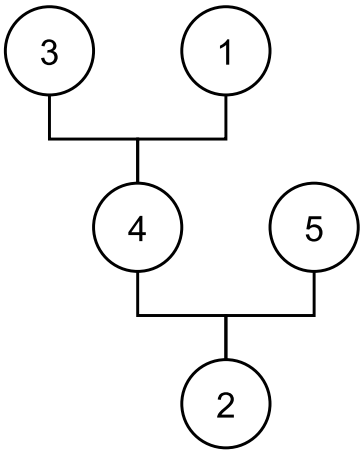
$$m_a = \begin{Bmatrix} \cdot & 0 & 1 & 0 & 0 \\ \cdot & \cdot & 1 & 0 & 0 \\ \cdot & \cdot & \cdot & 0 & 1 \\ \cdot & \cdot & \cdot & \cdot & 1 \\ \cdot & \cdot & \cdot & \cdot & \cdot \end{Bmatrix}$$

(a) A pedigree matrix m_a using age ordering $a = (i_3, i_1, i_4, i_5, i_2)$. Elements in the bottom triangular part (including the diagonal) are all zero, and thus those elements were omitted.

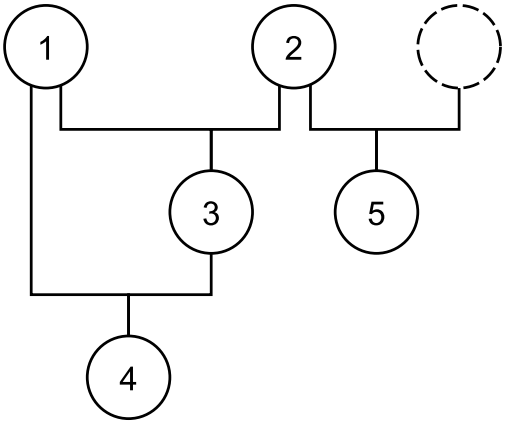
$$m_a = \begin{Bmatrix} \cdot & 0 & 1 & 1 & 0 \\ \cdot & \cdot & 1 & 0 & 1 \\ \cdot & \cdot & \cdot & 1 & 0 \\ \cdot & \cdot & \cdot & \cdot & 0 \\ \cdot & \cdot & \cdot & \cdot & \cdot \end{Bmatrix}$$

(b) An example for a pedigree matrix m_a, using age ordering $a = (i_1, i_2, i_3, i_4, i_5)$. Elements in the bottom triangular part (including the diagonal) are zero, and thus those elements were omitted.

Figure 3.9



(a) The corresponding pedigree graph to the pedigree matrix m_a as shown in figure 3.9a.



(b) The corresponding pedigree graph to the pedigree matrix m_a from figure 3.9b.

Figure 3.10

Now the pedigree space X - the space of all admissible pedigrees - can be defined as

$$X = M_A = \bigcup_{\forall a \in A} M_a = \bigcup_{\forall a \in A, \forall m_a \in M_a} m_a \tag{3.9}$$

where M_a is the space containing all possible admissible (upper triangular) pedigree matrices $m_a \in M_a$ using age order a, and A the space of possible age orderings. The chosen pedigree representation facilitates completeness and validity, as it can represent all possible pedigrees that are also biologically admissible.

3.3.3 Extended pedigree representation

Along the way the admissible pedigree representation was constructed (section 3.3.2), the extended pedigree matrix $\hat{m}_a$ was already mentioned. In contrast to the (admissible) pedigree matrix m_a, which is strictly upper triangular (without the diagonal), the extended pedigree matrix also contains elements in the lower triangular

part (and the diagonal). The extended pedigree matrix cannot represent biologically admissible pedigrees but it has favorable properties, which helps to traverse in the space of pedigrees X.

All elements of the matrix may contain ones and zeros. This representation facilitates some matrix operations to take place, which swap two rows (or columns respectively) or change the order of the rows (or column respectively). After employing these matrix operations, elements cannot be guaranteed to remain upper triangular and thus these operations can be applied on the extended pedigree matrix $\hat{m}_a$ but not for the (admissible) pedigree matrix m_a. These operations are explained in detail in the later section 3.5 on page 26. The applicability of such modification operations on the chosen representation is essential for traversing the search space effectively.

The advantage of the extended pedigree matrix $\hat{m}_a$ is that the integrity of the representation can be maintained easily. The operations move and swap (as explained in later section 3.5) can be applied in a straightforward way. In contrast to that, the admissible pedigree representation using the matrix m_a, which is required to be upper triangular only, does not facilitate these operations. A move or swap to both, rows and columns, may result in a matrix, which is not strictly upper-triangular anymore, i.e. it contains a (one or more) '1' elements in the lower triangular part.

Furthermore, by using the extended pedigree matrix $\hat{m}_a$, the modification operations (as explained in later section 3.5) fulfill the properties required by the Metropolis-Hastings algorithm, which are ergodicity and reversibility of the Markov Chain. Hence, this representation may also be used for sampling rather than search. In contrast, the admissible pedigree representation using the matrix m_a does not satisfy the reversibility criterion, and it cannot be implied that the resulting distribution $P(X)$ is invariant. Concluding, the extended pedigree representation has some properties, which make it preferable to be used to search to the space of pedigrees.

3.4 Complexity

As described in section 2.5.2, pedigree reconstruction is a NP-hard problem. This can be demonstrated using the described pedigree representations. All those create a search space that is exponential in the number of individuals N.

Using a simple matrix m as in the graph representation (see section 3.3.1) to represent pedigrees, there are in total 2^{N^2} possible matrices, which represent directed graphs with also contain self-loops, i.e. arcs from a node to itself. Removing those, there remain $2^{N(N-1)}$ possible directed graphs. As every instance of this matrix m does not necessarily also represent an admissible pedigree, the created search space is too large and includes many invalid pedigrees.

3.4.1 Extended pedigree representation

Using the proposed representation for pedigrees, applying only the constraint for compliant parents, the size of the search space is reduced. In particular the parent set $\pi_i \in \Pi_i$ for a single individual i can have

$$|\Pi_i| = 1 + N + \frac{N \times (N-1)}{2} \tag{3.10}$$

possible instances. Here Π_i denotes the space of all possible parent sets. Please note that self-loops are allowed in the extended pedigree representation, so that one can be its own parent.[9] It

s size $|\Pi_i|$ is smaller than N^2 for all $N \geq 2$, so that its size is upper bounded by a polynomial expression. As there are N parent sets in a pedigree, one for each individual, the total search space has size

$$|\hat{M}_a| = |\Pi_i|^N \tag{3.11}$$

Hence the total search space remains of exponential size compared to X' but with a lower exponent, i.e. N instead of N^2. Table 3.1 demonstrates that the search space got smaller for large N compared to the graph representation.

[9]For example in a trivial pedigree with N=1, the only individual i_1 may have two possible instantiations of its parent set, i.e. $\emptyset$ and $\{i_1\}$.

N	2^{N^2}	$2^{N(N-1)}$	$\lvert\hat{M}_a\rvert = \lvert\Pi_i\rvert^N$	N	2^{N^2}	$2^{N(N-1)}$	$\lvert\hat{M}_a\rvert = \lvert\Pi_i\rvert^N$
1	2	1	2	8	1.8E+19	7.2E+16	3.5E+12
2	16	4	16	9	2.4E+24	4.7E+21	9.2E+14
3	512	64	343	10	1.3E+30	1.2E+27	3.0E+17
4	65,536	4,096	14,641	20	2.6E+120	2.5E+114	3.1E+46
5	33,554,432	1,048,576	1,048,576	30	8.5E+270	7.9E+261	1.1E+80
6	68,719,476,736	1,073,741,824	113,379,904	40			3.7E+116
7	5.6E+14	4.4E+12	17,249,876,309	50			2.0E+155

Table 3.1: The complexity was reduced by constraining the search space with the requirement for parent-compliance, i.e. an individual can have at most two parents.

In the chosen pedigree representation, the search takes places in the space of all extended pedigree matrices, each of which consisting of an age order a and an extended matrix $\hat{m}_a$. Therefore, the search takes place in the space of both, the set of possible age orderings A and the set of possible extended pedigree matrices $\hat{M}_a$. The set of possible age orderings has size $|A| = N!$ and can be any permutation of the ordering among the individuals. For each age ordering $a \in A$, the set of possible extended pedigree matrices has size $|\hat{M}_a| = |\Pi_i|^N$. Together, both components combine and create a search space with size

$$|\hat{M}_A| = |A| \times |\hat{M}_a| = N! \times |\Pi_i|^N \tag{3.12}$$

Individuals N	Age orderings $\lvert A\rvert = N!$	Total search space $\lvert\hat{M}_A\rvert$	Individuals N	Age orderings $\lvert A\rvert = N!$	Total search space $\lvert\hat{M}_A\rvert$
1	1	2	8	40,320	1.4E+17
2	2	32	9	362,880	3.3E+20
3	6	2,058	10	3628,800	1.1E+24
4	24	351,384	20	2.4E+18	7.4E+64
5	120	125,829,120	30	2.7E+32	3.0E+112
6	720	81,633,530,880	40	8.2E+47	3.1E+164
7	5,040	8.7E+13	50	3.0E+64	6.0E+219

Table 3.2: Size of the search space using the proposed representation - the extended pedigree representation

The search space created by this representation became smaller for $N \geq 7$ individuals compared to the graph representation using matrix m which has complexity $|X'| = 2^{N \times (N-1)}$. This is demonstrated in table 3.2.

3.4.2 Admissible pedigree representation

The complexity of the search space can be further reduced by removing the lower triangular matrix, including the diagonal, of the extended pedigree matrix $\hat{m}_a$. The result is the admissible pedigree matrix m_a of which there are in total $|M_a|$ possibilities if the age ordering a is fixed. Let k be the number of older individuals for every individual in the age ordering then

$$|M_a| = \prod_{k=0}^{N-1} \left(1 + k + \frac{k \times (k-1)}{2}\right) \tag{3.13}$$

$|M_a|$ represents the number of possible parent sets integrated over all individuals. Given an age ordering, every individual has a different number of possible parents sets, which are all required to be older in the age ordering, e.g. the oldest individual has 1 possible parent set (the empty set), the second oldest has 2, the third oldest has 4, the forth oldest 7, and so on. Under this constraint, the total search space is reduced to

$$|M_A| = |A| \times |M_a| = N! \times |M_a| \tag{3.14}$$

Individuals	Parent sets	Total search space	Individuals	Parent sets	Total search space
N	$\|M_a\|$	$\|M_A\|$	N	$\|M_a\|$	$\|M_A\|$
1	1	1	8	6,288,128	2.5e+11
2	2	4	9	232,660,736	8.4e+13
3	8	48	10	1.1e+10	3.9e+16
4	56	1,334	20	2.6e+30	6.3e+48
5	616	73,920	30	2.1e+55	5.5e+87
6	9,856	7,096,320	40	1.5e+83	1.2e+131
7	216,832	1.1e+09	50	1.6e+113	4.9e+177

Table 3.3: Size of the search space using the admissible pedigree representation

However, it still contains redundant pedigrees, which can be represented with several different age orderings, for example a trivial pedigree with three unrelated individuals. The real search space of all biologically valid pedigrees is smaller, but due to its difficulty to count them efficiently (no method besides an exhaustive enumeration is known to the author), no such values are presented here.

3.5 Proposal generation

As part of the search algorithm, proposal pedigrees x' are generated in the neighborhood of the current pedigree x. To generate a proposal, a copy of the current pedigree undergoes multiple transitions and its result is a new proposed pedigree. The transition step size s defines the number of transitions that are sequentially performed during the generation of the proposals. As a result the generated pedigree x' will be similar to x and lie in its neighborhood $N(x, s)$ of x using s transitions. This neighborhood expresses the similarity between pedigrees. The similarity between two pedigrees x and x' is defined by a distance function, which is the number of transitions which are required to change a pedigree x into a pedigree x'.

$$d(x, x') \tag{3.15}$$

The neighborhood $N(x, s)$ of a pedigree $x \in X$ consists of all pedigrees which can be reached from x using s transitions. The parameter s will be introduced in section 3.5.

To move in pedigree space, the extended pedigree matrix $\hat{m}_a$ is used. In order to obtain an admissible pedigree, the connectivity represented by the extended pedigree matrix $\hat{m}_a$ can be mapped back the (admissible) pedigree matrix m_a, which procedure is explained in section 3.5.5. The transitions are chosen by random using a random number generator.

As there are two data structures that represent a pedigree, there are also two basic types of transitions that modify pedigrees. Either the age ordering a is changed or connectivity, which is represented by the extended pedigree matrix $\hat{m}_a$, is changed. The age ordering is changed with probability p and the connectivity is changed with probability $1-p$. Generally a value of $p = 0.5$ was used so that both types of transition were equally likely to occur.

3.5.1 Age ordering transitions

In order to change the age order a, two individuals a can simply be swapped which is termed a swap-transition, or a single individual in a can be moved to a different position in a which is termed a move-transition[10].

Changing the age ordering using a move-transition corresponds to changing the age of a randomly selected single individual, i.e. making a single individual older or younger compared to the other individuals. Once an individual i_a was made older than another individual i_b through a move-transition, i_a becomes a new candidate to be a parent of individual i_b while at the same time i_b is no longer a possible candidate to be a parent of i_a. Similarly this applies to the opposite scenario, i.e. if i_a would "be made younger".

The individual to change the age is chosen randomly from all possible individuals. For that two non-equal random numbers r_1 and r_2 are drawn from the interval $[1, \ldots, N]$, and then the individual which is at position r_1 in the age ordering a is moved to position r_2 in the age ordering a. Since r_1 and r_2 are different, an age order transition is guaranteed to change the age ordering and therefore it cannot result in the same age ordering.

As an example for a move-transition with $r_1 = 4$ and $r_2 = 2$, the age ordering $a = (i_1, i_2, i_3, i_4, i_5)$ of a pedigree is changed to $a = (i_1, i_4, i_2, i_3, i_5)$ by moving i_4 from the 4th position to the 2nd position within the order a. This implies that i_2 moves to position 3 and i_3 moves to position 4.

[10]Move-transition: A. Almudevar [1] denoted this transition as a "step" but this notation is not used in this thesis as it might be confused with other notations used in this thesis.

Similar to results obtained by A. Almudevar (2003) [1], the move-transition was observed to achieve better performance than the swap-transition. Therefore, only the move-transition was considered throughout the remainder of this research. It can be argued that the move-transition seems favorable as it only changes the age of a single individual relative to all other individuals, whereas the swap-transition changes the age of two individuals relative to all other individuals in the pedigree. The change obtained by using a swap-transitions may be too large so that already correctly identified family relations may "get lost again" during search.

To make the extended pedigree matrix $\hat{m}_a$ consistent with a changed age ordering a, the (extended) pedigree matrix $\hat{m}_a$ is transformed by changing the rows and columns of the matrix respectively.

$$\hat{m}_a = \begin{Bmatrix} 1 & 0 & 1 & 1 & 0 \\ 0 & 0 & 1 & 0 & 1 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 & 0 \end{Bmatrix}$$

(a) The pedigree matrix $\hat{m}_a$ using $a = (i_1, i_2, i_3, i_4, i_5)$ prior to the transition

$$\begin{Bmatrix} 1 & 1 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 & 1 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 1 & 0 & 1 & 0 & 0 \end{Bmatrix}$$

(b) and the pedigree matrix during the transition after columns were changed, i.e. moving the 4th column to the 2nd column,

$$\hat{m}_a = \begin{Bmatrix} 1 & 1 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 1 \\ 0 & 1 & 0 & 0 & 0 \\ 1 & 0 & 1 & 0 & 0 \end{Bmatrix}$$

(c) and final pedigree matrix $\hat{m}_a$ using $a = (i_1, i_4, i_2, i_3, i_5)$ after completing the transition, i.e. after moving the 4th row to the 2nd row.

Figure 3.11: Example of an age order transition using the extended pedigree matrix $\hat{m}_a$.

Continuing the example from above, in which the age ordering $a = (i_1, i_2, i_3, i_4, i_5)$ of a pedigree is changed to $a = (i_1, i_4, i_2, i_3, i_5)$, the corresponding pedigree matrix is changed accordingly, as shown in figure 3.11. In particular, the corresponding extended pedigree matrix $\hat{m}_a$ is transformed by moving the 4th column in front of the 2nd column (previous columns 2 and 3 then move to columns 3 and 4 respectively) and moving the 4th row in front of the 2nd row (previous rows 2 and 3 then move to rows 3 and 4 respectively).

3.5.2 Connectivity transitions

The second type of transition modifies the connectivity of the pedigree. This transition changes the parents of a randomly selected individual i of the pedigree by reassigning the elements (individuals) in i's parent set $\pi(i)$.

To perform this transition using the proposed pedigree representation, the pedigree matrix $\hat{m}_a$ is modified in a single column. First, a random variable r is drawn from the interval $[1, \ldots, N]$ in which r corresponds to the column number to be replaced in the pedigree matrix $\hat{m}_a$, respectively to the individual i which is at position r in the age ordering a. Second, a new column vector c is created. Finally, the column r of the pedigree matrix $\hat{m}_a$ is replaced by the new column c.

In order to create a new column vector c, a distribution of all possible parent sets per individual is used. The particular distribution is dependent on the chosen parent selection strategy. This distribution is either uniform in which all possible parents sets have the same probability to be selected, or non-uniform so that more likely parent sets for an individual have a higher probability to be selected. These are two variants, of which the new column vector c can be created. Both are described subsequently in sections 3.5.3 and 3.5.4.

3.5.3 Random parent selection

This search variant decides about the selection of new proposed parent sets using the column vector c. Using this variant, the distribution of all admissible column vectors was created prior to the search, e.g. as depicted in figure 3.12. Every column in that matrix has the same likelihood to be selected and contains at most two ones in order to comply with the requirement for parent-compliance.

$$\begin{array}{cccccccccccccccc} 0 & 1 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 1 & 0 & 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 1 & 0 & 1 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 1 & 0 & 1 & 1 \end{array}$$

Figure 3.12: Set of admissible column vectors for $N = 5$ individuals

3.5.4 Guided parent selection

The second variant for parent selection (variant 2) exploits knowledge about likely parents for individuals to increase the quality of the proposed pedigree samples. Instead of drawing a transition randomly from a uniform distribution, transitions are drawn from a distribution that gives the more likely parents a higher probability of being drawn. Therefore, this variant is termed, guided parent selection.

For that, the column vector c is determined by choosing a random parent set from the distribution of all possible parent sets for that individual. In that distribution, more likely parent sets have a higher probability of being selected. The probabilities in that distribution are derived from their local likelihoods

$$P(g_i|g_{\pi(i)}) \tag{3.16}$$

which is the conditional probability of observing the genotype g_i of an individual i given the genotypes of his or her parents $g_{\pi(i)}$. The local likelihoods are computed for all possible parent sets prior to the search and each local likelihood is conditionally dependent on the genotypes $g_{\pi(i)}$ of the corresponding parent set $\pi(i)$. Hence, each $g_{\pi(i)}$ contains two, one or no genotypes. Using these local likelihoods directly for the distribution of possible parent sets is impractical because parent sets with a high likelihoods are overrated which produces a very skew probability distribution. To produce a more reasonable distribution the following score

$$f(P(g_i|g_{\pi(i)})) = \left(\frac{1}{-\log_{10}\left[P(g_i|g_{\pi(i)})\right] + 0.1} \right)^2 \tag{3.17}$$

was used, which first projects the original likelihood on a $\log_{10}$ scale, producing a negative number in the range $[0, -\infty]$, which sign is inverted and consecutively modified by a small value 0.1 to prevent a division by zero in the following step in which the reciprocal is taken. The extent to which these local scores contribute was raised by taking the square of that value. Finally, the probability for an individual i that particular parents $\pi(i)$ are drawn is given by the guided function

$$g(i, \pi(i)) = \frac{f(P(g_i|g_{\pi(i)}))}{Z_i} \tag{3.18}$$

in which Z_i is the sum of all scores

$$Z_i = \sum_{\forall \pi(i) \in \Pi_i} f(P(g_i|g_{\pi(i)})) \tag{3.19}$$

and Π_i being the set of all possible parents sets of i, which consists of all combinations of individuals (except the individual i itself) with a set size of one or two elements plus the empty set.

Eventually the produced probability distribution has preferences towards parent sets that have a higher local probability, i.e. considering only three individuals at a time rather than the complete pedigree. Finally, the column vector c is created based on the chosen parent set, i.e. a vector of length N filled with zeros, except at the positions that correspond to the selected parents, where its elements are one.

Using this variant, the 'random walk' still uses randomness and generally prefers to walk in directions of higher likelihood but its guidance originates in a modified proposal generation rather than in the acceptance of proposals.

3.5.5 Resolving invalid kinship relations

Employing transitions changes the age ordering a, the pedigree matrix $\hat{m}_a$, none of them or both of them. After a transition, the pedigree matrix $\hat{m}_a$ may contain elements in the lower triangular part of the matrix (including the diagonal) and thus does no longer represent a biologically admissible pedigree. Therefore, the extended pedigree matrix $\hat{m}_a$ is mapped back to an admissible pedigree matrix m_a. For resolving these invalid kinship relations, there are three different strategies: (1) dropping, (2) masking, and (3) correcting.

In the most simple strategy, the lower triangular (including the diagonal) of the extended pedigree matrix $\hat{m}_a$ is simply dropped after every transition so that both matrices $\hat{m}_a$ and m_a become identical. This strategy is identical to not using the extended pedigree matrix, and any kinship relations that turn invalid due to a change of the age ordering are removed. This strategy to resolve invalid kinship relations is termed dropping.

Using another strategy, the extended pedigree matrix $\hat{m}_a$ is kept after every transition and used for the next transition. The admissible pedigree matrix m_a is just derived from it by removing the lower triangular part

(including the diagonal) and is the matrix which represents the pedigree. Using this strategy, those kinship relations in which the child is older than the parent are masked and become temporarily inactive. Inactive kinship relations do not contribute to the current connectivity of the pedigree and consequently do not contribute to the likelihood computation but they may become active again in subsequent transitions if the age ordering changes again. This resolving strategy is termed masking.

In another strategy, the pedigree representation can be 'corrected' to prevent that information loss caused by dropping the lower triangular matrix. The pedigree matrix $\hat{m}_a$ may have elements in the lower triangular part but still represent a valid directed acyclic graph. In this case, the matrix $\hat{m}_a$ which has age ordering a can be traversed to a different matrix $\hat{m}_b$ using a 'corrected' age order b. Both matrices represent the same connectivity but the corrected matrix $\hat{m}_b$ is also upper triangular. Therefore, if such a corrected matrix exists, then the pedigree matrix $\hat{m}_b$ is used as the extended pedigree matrix . For that, the age ordering a and the pedigree matrix $\hat{m}_a$ are replaced (overwritten) with b and $\hat{m}_b$ respectively. Eventually, not ones exists in the lower triangular part and thus $\hat{m}_a$ and m_a are identical.

3.5.6 Additional structural constraints

During search involving incomplete genotype samples, those individuals without genotype observation may be 'insufficiently connected'. Forensic scientists may not be interested in the likelihood of untyped individuals at specific locations in the pedigree, e.g. as a leaf (i.e. a child without children), or as founders with only a single child. Optionally those pedigrees can be excluded from the sampling space. In case an excluded pedigree was generated, the pedigree generation is simply repeated, until a valid pedigree is returned. Eventually, the effects of this constraint to the search were considered as doubtful, as it may distort the search space by hindering a proposed state x' to be reached from another state x, so eventually this option was not used as part of the proposed algorithm.

3.5.7 Summary

To summarize the generation of proposals, pedigrees are created in the neighborhood of a 'current pedigree' using multiple transitions. These transitions change the connectivity of the pedigree in terms of an age ordering and kinship relations (represented using the pedigree matrix). Changing the age ordering always occurs at random whereas for changing the parent sets two strategies were proposed: (1) Random changes based on a uniform distribution and (2) guided changes which exploits knowledge about local probabilities represented by a non-uniform distribution. As kinship relations may become biologically invalid during the transitions, such relations are resolved by removing (drop), hiding (mask) or correcting strategies to complete the transitions.

3.6 Search strategies

The different variants of changing the age-order or the parent sets, and the used pedigree matrix and their method to resolve invalid kinship relations after a transition, combine and make up several search strategies, which are summarized in table 3.4.

Search strategy	Age ordering	Parent set selection	Resolving variant	Pedigree matrix
(1a) Metropolis-Hastings	random	random	mask	extended $\hat{m}_a$
(1b) Random walk			drop	admissible m_a
(2a) Guided search		guided	mask	extended $\hat{m}_a$
(2b) Guided search			drop	admissible m_a
(3) Guided search	derived automatically	guided	correct	admissible m_a

Table 3.4: Combinations of different transition variants, regarding age-order selection, parent-set selection, and representation

3.6.1 Metropolis-Hastings (strategy 1a)

The search strategy (1a) employs completely random transition to generate proposal pedigrees. In particular, it uses random transitions of the age ordering and a random parent selection. Furthermore, it uses the masking strategy to resolve invalid kinship relations to complete the transition.

This strategy fulfills the detailed balance requirement of the Metropolis-Hastings method, in which the Markov chain is reversible, the transitions in between any two states x and x' have the same probabilities to be selected, i.e. in the probability to reach a state x' from another state x is equal to the probability to reach x from x'.

$$P(x)P(x|x') = P(x')P(x'|x) \tag{3.20}$$

This property implies, that the produced distribution $P(X)$ is invariant. In particular, given that an age order transition selected, the probability to choose a particular transition of the age ordering is

$$\frac{1}{N(N-1)} \tag{3.21}$$

and given that a parent set transition was selected, the probability to choose particular parent set transition is

$$\frac{1}{N \times a(N)} \tag{3.22}$$

in which $a(N)$ is the number of possible parent sets per individual. Therefore, with a constant N, the probability to choose a particular transition remains invariant from transition to transition.

Having the two properties, an invariant probability distribution as well as ergodic transitions, the constructed Markov Chain fulfills the requirements to be used as a Metropolis-Hastings algorithm. Hence, the search strategy is termed 'search based on the Metropolis-Hastings algorithm' and could be used for sampling as well, i.e. to create samples from the distribution $P(X)$.

3.6.2 Random walk (strategy 1b)

In contrast to the Metropolis-Hastings approach, this search variant (1b) uses "dropping" to resolve invalid kinship relations to complete the transitions and thus does not employ the proposed representation of a pedigree using the extended pedigree matrix. The author could not proof that the Markov Chain remains reversible using the (only upper-triangular) admissible pedigree matrix, particularly if the transitions to generate the proposal contain both, changes to the age ordering and changes to the parent sets. Hence, this search strategy is simply termed a "random walk".

3.6.3 Guided search (strategies 2a and 2b)

Guided search strategies are proposed, which employ local likelihoods that aim to direct the search algorithm to areas of high likelihood. This guidance part takes place the parent set selection of a transition. On the one hand, this can improve the search performance in terms of steps to find the maximum likelihood pedigree. On the other hand, guided search strategies 'reshape' the landscape of the search space in a way, that the Markov chain is not reversible, and the transitions between two states x and x' may have different probabilities per direction. Thus, these search strategies are not called Metropolis-Hastings, and the random walk is still based on randomness but is biased by using a non-uniform distribution.

Similarly to the previously described Metropolis-Hastings and the Random walk, the guided search strategy can be divided in a strategy (2a) which uses 'masking' to resolve non-admissible pedigrees after each transition and another strategy (2b) which does not use the extended pedigree matrix.

3.6.4 Search only in the space of parent sets (strategy 3)

Another search strategy is proposed, which only employs transitions that change the parent sets. That is, the probability (as termed in 3.5) to change the age ordering is $p = 0$. After such a transition, the variant 'correction' is used to resolve potentially invalid kinship relations. Thus, an appropriate age ordering is selected automatically that enables the transition without 'dropping' or 'masking' of invalid kinship relations. As not every transition, i.e. parent set change, produces an admissible pedigree, transitions need to be repeated until an admissible pedigree is proposed. The shortage of this approach is the computationally more demanding reordering of the individuals to obtain a valid age ordering as well as the before-mentioned occasional repetition of transitions.

In the scope of this thesis, only a single experiment was conducted involving this strategy (B.18). This strategy seems promising, as one may not require searching through the space of age orderings (which has a factorial complexity) as a suiting correct age ordering can be derived automatically after applying an transition of the connectivity.

3.7 Likelihood computation

One of the core components of the proposed method is the likelihood computation. It is the likelihood of observing the genotype sample g given the pedigree x

$$L = P(g|x) \tag{3.23}$$

The likelihood is computed after every proposal generation, first to determine the acceptance ratio, which decides about if the proposal step is accepted or rejected, and second to keep track of the pedigree to determine the supposed maximum likelihood pedigree.

In the scope of this research, two ways of computing the likelihood for a pedigree given the genetic data were used, which are explained subsequently (in sections 3.7.1 and 3.7.2). Both approaches were implemented in Matlab and were automatically selected during run-time depending on missing genetic data. Further, a hybrid likelihood computation was implemented, which decided on a per-locus base, which method was used (see section 3.7.3).

3.7.1 Bayesian network inference algorithm for incomplete samples

The first method, the Bayesian inference algorithm has the advantage that it can deal with unobserved genotypes, but it lacks in computational performance. The Bayesian network computes the likelihood on a per-locus base. It first computes the likelihood for all individuals and for a single locus at once, and then multiplies these per-loci-likelihoods for all loci to acquire the total likelihood of the pedigree.

$$P(g^k|x) = \prod_{\forall i \in I} P(g_i^k|x) \tag{3.24}$$

$$P(g|x) = \prod_{\forall k \in K} P(g^k|x) \tag{3.25}$$

respectively in log-likelihood notation, in which the likelihoods per locus contribute to a sum

$$\log P(g^k|x) = \sum_{\forall i \in I} \log P(g_i^k|x) \tag{3.26}$$

$$\log P(g|x) = \sum_{\forall k \in K} \log P(g^k|x) \tag{3.27}$$

In particular, this research employed the component to compute the likelihood of a pedigree as used in the Bonaparte Disaster Victim Identification System. It computes $P(g^k|x)$, i.e. the likelihood that the genotypes g^k are observed on the single locus k for all individuals given the pedigree connectivity x, which represents the relations between the genotypes.

The used Bayesian network inference algorithm provides a robust and flexible solution for computing the likelihood, but for being used as part of a Monte Carlo simulation, which entails many calls of the inference algorithm, the overall performance was observed to be poor. Therefore, it can be considered as the bottleneck in computation speed, especially if many likelihood values have to be computed as in this proposed method. Therefore, several optimizations were applied to increase its performance, in particular by avoiding unnecessary calls of the Bayesian inference algorithms using caching, as well as parallel computing.

An optimization was used, which stores the computed likelihood values on a per locus basis with the intention to re-use them, so that the Bayesian inference algorithm only had to run once for the same input. The likelihood $P(g^k|x)$ can be seen as a function $f(g^k, x)$ which is dependent on the genotype data for a single locus g^k and the pedigree connectivity x. Therefore, a combined value of both function arguments, i.e. g^k and x, was used as an index in a lookup-table. Further, hash-values were used to speed up retrieval of stored likelihoods. Using this technique, unnecessary calls of the Bayesian inference algorithm could effectively be avoided, and eventually a noteworthy performance increase was observed.

Another improvement uses parallel computing to reduce the time for a single likelihood computation. This exploits that the Bayesian network inference takes place on a per locus base, so that the computation can be factorized and distributed over several machines. Here, the computation of a single worker computes the likelihood for a single locus:

$$P(g^k|x) = \prod_{\forall i \in I} P(g_i^k|x) \tag{3.28}$$

Using parallel computation, the performance could was increased.

3.7.2 Local likelihood computation for complete samples

If genotype information was completely observed for all individuals, then the Bayesian Network computation can be replaced by a faster alternative. This alternative method is based on simple formulas, which compute the local likelihood $P(g_i|g_{\pi(i)})$, i.e. the likelihood of observing the genotype of a single individuals given its parents genotypes, as described previously in section 2.4.3. Simplified, it is a likelihood that i has parents $\pi(i)$.

One the one hand, computing these local likelihoods for all individuals is computationally cheaper than computing the likelihood for all individuals at once using the Bayesian network inference with the Junction Tree algorithm. On the other hand, the computation of the local likelihoods is limited to complete samples.

The Bayesian inference algorithm is required to compute the likelihood of observing the genotypes on a single locus for the whole pedigree at once, and then integrates it over all loci. Using local likelihoods instead, this order of operation can be switched, i.e. the likelihood of observing the genotypes of each individual given its particular parent set can be computed first, and then the product over all individuals, which eventually leads to the same likelihood.

$$P(g|x) = \prod_{\forall i \in I} P(g_i|x) = \prod_{\forall k \in K} P(g^k|x) = \prod_{\forall i \in I} \prod_{\forall k \in K} P(g_i^k|x) \tag{3.29}$$

On its lowest level, the likelihood of an individual to have a particular pair of alleles g_i^k at the genetic locus k, i.e. $P(g_i^k|x)$, is computed. After these local likelihoods are computed, they act as factors and enable simple multiplication to take place to obtain the likelihood for the whole pedigree. This way of computing the likelihood was found to be significantly faster than using the Bayesian network inference algorithm.

Using local likelihoods, the approach enables factorization of the likelihoods, which is not possible while using the Bayesian network inference algorithm. Besides the initial speed gain, this property can be exploited to speed up the sampling process even further. An improvement was used, in which the likelihood computation is carried out only once before the actual search. All local likelihood values can be pre-computed and stored before the sampling process and retrieved during the sampling. This limits the required computational effort of calculating the likelihood during the actual search to merely a few multiplications of retrieved factors.

3.7.3 Hybrid

Genotype data may also be missing on a per marker base, so that for different individuals different markers were observed, which only partially overlap, e.g. some individual may only have genotype data for markers k_1, k_2, k_3, k_4, k_5 available, whereas for another individual observations are only available for markers k_1, k_2, k_4, k_6, k_7. This may occur (1) if different genotyping standards are used, as for example CODIS and SGM Plus, (2) if genotype data was retrieved using outdated technologies regarding the choice of genetic markers, as for example in the Romanov data (see experiment B.8), or (3) in any other case the data is incomplete on some markers, as for example in the shrimp dataset used by R.G. Cowell (2012) [5].

For partially observed genetic profiles, in which alleles were observed completely for some but not all loci, i.e. $\forall i \exists k : g_i^k = (x_1, x_2)$ for that $x_{1,2} \neq 0$, a hybrid method was used. For completely observed loci $K_1 \subset K$, the faster alternative way of computing the likelihoods was used (as in section 3.7.2), whereas for the remaining loci $K_2 = K \setminus K_1$, which contain missing values, the Bayesian Network inference algorithm (as in section 3.7.1) was used. For that, the likelihood function is divided into two parts:

$$LH(\cdot) = LH(\cdot)_{K_1} \times LH(\cdot)_{K_2} \tag{3.30}$$

$$P(g|x) = \left(\prod_{\forall k \in K_1} P(g^k|x) \right) \times \left(\prod_{\forall k \in K_2} P(g^k|x) \right) = \prod_{\forall k \in K} P(g^k|x) \tag{3.31}$$

Using this factorization, the likelihood computation of genotype data, which was only observed for some markers, was improved.

3.7.4 Possible improvements

A number of additional performance improvements regarding the likelihood computation are possible, which were not implemented in the scope of this thesis, but are presented for the sake of completeness.

Instead of computing the likelihood of observing a genotype g_i of individuals i given its parents' genotypes, one may reduce the computational demands by only specifying the number of required allele mutations and the number of homozygote alleles, rather than employing the actual allele values for the computations.

Some genotypes of the pedigree may be conditionally independent from other genotypes, and should be computed as such. For example, if the pedigree can be divided into multiple connected sub-pedigrees, then the likelihood computations can be factorized per sub-pedigree. In another example, the likelihood computation among a group of typed individuals, which only have typed ancestors as well as typed children, does not require a fully featured Bayesian network inference like the Junction Tree algorithm and can be replaced by more simple local likelihood computations.

In the proposed method, the likelihood computation takes place to decide about the probability, with which the new proposal is accepted. Instead of computing the likelihood first, and then decide about the acceptance involving a random number, this procedure could be inverted. For that, a threshold likelihood is selected first, which corresponds to the probability that the proposal is accepted. This benefits in that the likelihood computation can be halted earlier, as soon as the proposal exceeds the threshold likelihood, so that the likelihood does not need to be computed completely for pedigrees, which are about to be rejected anyway.

3.8 Enumeration

To assess the correctness of the proposed method, it was useful to have knowledge about the correct solution, i.e. the maximum likelihood pedigree based on the genotype observation. Knowing the correct solution, a statement about the quality of the found solution can be made, such as "the maximum likelihood pedigree" was found. If such a check was not feasible, the pedigree, for which genotype data was created beforehand, was simply also assumed to represent the maximum likelihood pedigree.

To determine the maximum likelihood pedigree, a computationally expensive search was occasionally performed. The enumeration used the method as described in Almudevar (2003), which is limited to a small number of individuals as well as to complete genotype samples. The complexity of this method is $O(N!)$, so that this approach was practically feasible for about up to $N = 7$ individuals, which involved $7! = 5040$ likelihood computations.

The approach used by Almudevar (2003) assumed completely observed genotype information and was able to reduce the search space drastically[1]. The approach is based on the idea that the maximizing pedigree connectivity, represented by the pedigree matrix m_a, can be determined easily when genotype samples are complete and an age ordering a is given [1]. For that, the approach iterates over all possible age-orders $a \in A$, for which the maximum likelihood pedigree connectivity is determined [1]. Thus, the search only takes place in the space of all possible age orderings A, instead of in both, the age orderings A and the pedigree matrices M_a [1].

To determine the maximum pedigree matrix m_a given the age-order a, the maximum likelihood parent set is selected for each individual. For that, only older parents represent candidates to be in the parent set. Then, the probabilities of these local maxima are multiplied to obtain to maximum likelihood for the whole pedigree. This method can make use of the pre-computed local likelihoods (as in section 3.7.2), so that the maximum likelihood pedigree for any age ordering can be computed quickly by merely using a few multiplications of pre-computed values.

This method only returns a single maximum pedigree matrix m_a for each age-order a, even though multiple maxima are actually possible, because any individual might have more than one equally likely maximum parent set. This is a limiting factor, so that the method cannot guarantee all maxima to be found. Therefore, this method was only used to assure that the constructed pedigree, for which genotype data was generated, also represents a maximum likelihood pedigree.

4 Results

In order to assess the performance of the method, several experiments were performed, which can be found in appendix B. In this section, the results using the proposed method are presented.

4.1 Sample data generation

In order to test the method, sample genotype data was required. With exception of the data of the Romanov case, which was taken from Gill et al. (1994) [8], solely synthetic genotype data was used in the tests. This process involved the manual construction of several samples pedigrees, of which a list can be found in appendix A. The pedigrees used for testing of the method were mainly constructed manually. For these pedigrees, genotype data was generated according to the statistical assumptions made. Either the generated DNA data was based on a real or a constructed origin pedigree, which differs in connectivity (may involve incest, the number of generations).

4.1.1 Generating DNA

Using genuine human DNA is problematic as genetic profiles are usually treated confidentially and are typically not easily accessible. Therefore, most genotype samples used in this research were synthetic using a genotype generator.

Given a pedigree without genotype data, i.e. individuals and a pedigree structure, the genotype generator creates statistically valid genetic fingerprint data for any number of loci. Therefore, besides the individuals and their connectivity, and the number of loci, the statistical assumptions, i.e. the allele distribution and a mutation rate, need to be specified as the parameters for the genotype generator.

This method can handle incomplete genotype samples in which the genotype data of individuals can be missing. For that, the search is informed about the existence of an untyped individual by providing an additional individual without genotype data. To simulate a missing genetic profile, all allele observations of the genetic profile of that individual i were set to zero, i.e.

$$i \in I, \forall k \in K : g_i^k = (0, 0) \tag{4.1}$$

In this scope of the thesis and in the experiments, only a single individual at a time was untyped. This represents a limitation of this approach to be investigated further.

4.2 Computation time

In a typical search using the Bayesian network inference algorithm, using genotype data consisting of 20 genetic loci, and using 12 parallel worker threads, about 3000 to 4000 search steps could be performed per hour. This is roughly one search step per second. Due to the long run-time of a single search trial, which takes about 3 to 6 hours per trial (as for example in B.10), the number and the size of experiments that were conducted in the scope of this thesis was seriously limited. Thus, the number of tests performed using incomplete samples was reduced.

By using complete samples, the computation time of the likelihood computation could be reduced by replacing the Bayesian network inference algorithm with the faster alternative (as described in section 3.7.2). This allowed performing many more trials per experimental settings, so that results in such experiments have an increased significance. On the other hand, this improvement was made on the cost of a limited transferability towards conditions using incomplete samples.

4.3 Number of individuals

Given the limited computational resources available, the accuracy of the output of the proposed method is trade-off between the computation time given to the algorithm, and an estimated confidence about the correctness of result.

The maybe most important aspect is the number of individuals for which the proposed algorithm can reconstruct the pedigree. In several experiments, pedigrees of different sizes with a varying number of individuals were reconstructed successfully.

Unfortunately increasing the number of individuals also entails a modified pedigree structure, which also requires specifying the kinship relations for the added individuals. Thus, a comparison of different pedigree sizes with N

being the only varying variable is cumbersome. Therefore, different pedigrees with varying connectivity besides varying the number of individuals are presented. A list of all pedigrees reported in the scope of this thesis can be found in the appendix in A.

The smallest possible pedigree consists just of a single individual, $N = 1$. The computation of the maximum likelihood pedigree based on the genotypic data of only a single individual is trivial, i.e. the unrelated case. For $N = 2$ individuals, there exist three possible pedigrees. For $N = 3$ individuals there are already 25 admissible pedigrees[11] and computation is not trivial anymore.

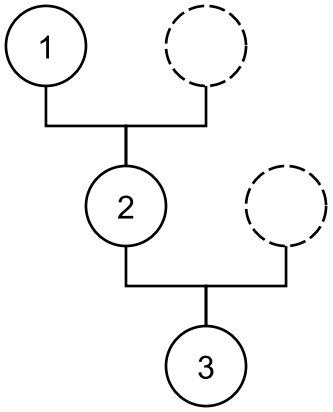


Figure 4.1: A small non-trivial pedigree of three observed individuals over three generations.

One such pedigree investigated in is the pedigree depicted in figure 4.1, which was aligned as a grandparent, parent, and a child. All three maximum likelihood pedigrees were found confidently after in average 16 search steps, which solution was be found within seconds of computation (cf. experiment B.1 and B.2). For such a small pedigree, the same results could also have been acquired using exhaustive search, but this results shows that the proposed method than reconstruct the pedigree for such small pedigrees at as fast as an exhaustive search.

As the number of individuals N increases, the search space expands exponentially. Hence, it is of interest, how well the proposed method can cope with the increasing search space. For $N = 5$ the search space is believed to consists of at least 1,000,000 possible pedigrees (cf. 2.5.2 on page 12 and 3.4.1 on page 24), and an exhaustive search is expected to require about 11.5 days (considering that the time to compute the likelihood of a pedigree takes one second).

Using the proposed method, in particular the Metropolis-Hastings based search variant, a pedigree of size $N = 5$ was reconstructed with high confidence. The search found the original pedigree already within the first 1000 search steps in 10 out of 25 trials (40%). It can be taken as granted, that any higher number of search steps would have lead to better performance per trial, but it may be more appropriate to repeat the search simply using a different initialization to gain a higher overall search performance.

Assuming that this reflects the probability to find any pedigree consisting of 5 individuals, only 6000 search steps in total (6 trials with each 1000 steps) would be required to reconstruct the pedigree with a confidence of 0.95, respectively 10,000 search steps (10 trials with each 1000 steps) to reach a confidence of 0.99. The time taken to compute a single trial was about 15 minutes using 12 parallel processing CPU's. Finally, a search to obtain this confidence would require about 1.5 hours, or 2.5 hours respectively, to complete. Depending on the demands in possible applications, such as in forensic science, the estimated confidence can be further increased to a desired level on the cost of higher computational demands.

Increasing the number of individuals further, the algorithm reached its limits in experiments involving up to 8 individuals, e.g. experiments B.10 or B.15. In an experiment involving $N = 8$ individuals (cf. experiment B.15), the maximum likelihood pedigree was found in two times, respectively three times in a similar condition, out of 25 search trials within the first 1000 iterations a guided search strategy (2a). Hence, the pedigree reconstruction was successful in about 8 percent given a single trial. Taking this probability as granted for all pedigrees of size $N = 8$, at least 36 search trials would be required to reach a confidence of 0.95, respectively 56 search trials to reach a confidence of 0.99. A search of this size requires about 8.5 hours, respectively 13.5 hours, to complete.

[11] 25 admissible pedigrees: 1 unrelated individuals, 6 single parent-child, 6 grandparent-parent-child, 3 two-parent, 3 two-child, 6 incestuous pedigrees

number of individuals N	0.95 confidence	0.99 confidence
$N = 3$ individuals	< 1 minute	< 1 minute
$N = 5$ individuals	1.5 hours	2.5 hours
$N = 8$ individuals	8.5 hours	13.5 hours

Table 4.1: Estimated computation times to reconstruct a pedigree with a desired confidence.

Table 4.1 summarizes the estimated computation times using 12 parallel processing CPU's to reconstruct a pedigree with an estimated confidence. Results were obtained using incomplete as well as complete genotype samples, but the values in the table refer to incomplete samples, i.e. when using the Bayesian network inference algorithm.

To conclude, with regard to its computational demands the proposed method is expected to find the maximum likelihood pedigree confidently, i.e. with a small chance of being wrong, for small pedigrees up to eight individuals $N \leq 8$, given that several trails were performed.

4.4 Complete vs. incomplete genetic data

Experiments investigated in the performance differences of the algorithm between complete and incomplete samples of genotype data. If genotype data was completely observed for all individuals, then the Bayesian network inference algorithm to compute the likelihood was replaced by a more efficient algorithm (cf. 3.7.2 on page 32).This resulted in a faster computation for complete samples. This helped to speed up the experiments without changing its results with exception of the measured computation time. Therefore, differences between complete and incomplete samples were measured in the number of search steps instead, rather than in the consumed computation time.

For a very small pedigree ($N = 3$) no differences in performance were observed. The same performance, i.e. a similar amount of required search steps, was achieved independent of whether the genotype profile of an individual was observed (cf. B.1) or unobserved (cf. B.2).

For a larger pedigree involving $N = 7$ individuals, in one experiment (cf. B.10), the search converged (which happened when the likelihood did not further improve for successive 10,000 steps) in average after 14,142.64 steps in an observed condition compared to 19,476.4 search steps in an unobserved condition. Hence, the search converged faster, i.e. found its maximum earlier in the trials, if samples were completely observed. Results were considered not significant enough to enable generalization of this finding but they may indicate a higher difficulty of pedigree reconstruction if genotype data is missing.

In an experiment (c.f. experiment B.10), the acceptance rate for incomplete samples was observed to be higher in comparison to a similar scenario with complete genotype samples. Searching for an explanation for this result, the acceptance rate generally expresses the mean acceptance, which is computed based on the comparison of the likelihood of the current pedigree x and a proposal pedigree x' (see section 3.2.2). Thus, a low acceptance rate indicates that the proposal had in average a low likelihood compared to the "current samples". Possible explanations for this may be a relatively low quality of the generated proposal pedigrees or a high quality of the average "current" pedigree, or the combination of both. However, it remains subject to question, why more proposal were accepted when using incomplete samples.

The two observations, the faster convergence as well as the higher acceptance rate for incomplete samples are suspected to be rooted in a change of the landscape of the search space. For incomplete genotype samples, the likelihoods $P(g|x)$ of all possible pedigrees $x \in X$ may produce fewer and less high peaks compared to complete samples. This may lead to a more uniformly distributed search space using incomplete samples. This facilitates the acceptance of proposals, which were generated in the neighborhood of the "current pedigree", so that the search lingers around in high peaks for a shorter amount of time compared to a search space with fewer and less high peaks. Eventually, this may cause a higher acceptance rate for incomplete samples compared to complete samples.

Some experiments were conducted to investigate in the applicability of the method using genotypes, which contained missing observations only for some markers. Using a Bayesian network, the method can make use of all available marker data, and can handle the uncertainty due to missing genotype observations in a transparent way. This extends the applicability of pedigree reconstruction, and increases the robustness of the method.

Furthermore, the hybrid way of computing the likelihood (as described in 3.7.3) was successfully employed Compared to using the Bayesian network inference algorithm for all markers, a performance gain was achieved, because this method automatically selected per marker the faster way of computing the likelihood. In particular, using genotype data, which was completely observed for all individuals on a single markers, the faster alternative method to compute the likelihood could be used, which reduced the computational effort, whereas for the remaining genotype data the Bayesian network inference algorithm was used.

4.5 Applicability to human genotype samples in the Romanov case

In experiment B.8, the applicability of the proposed method to on real human pedigrees was demonstrated (cf. experiment B.8 on page 59). In contrast to most other pedigrees in question covered by this research, which were manually constructed pedigree structures with generated genotype data, an experiment using real human DNA was conducted.

Gill et al. (1994) investigated in the identification of the remains of the Romanov family, the former Russian royal family, consisting of the last Tsar, Tsarina, three of their children, a doctor and three servants [8]. In their analyses, they used matching methods based on STR and mtDNA data as well as sex testing [8]. The found family relations between the nine found bodies, as found by their investigations, are depicted in figure 4.2, consisting of the five Romanovs, as well as the doctor and the servants which are all assumed unrelated to each other. It is believed that these family relations also represent the correct pedigree.

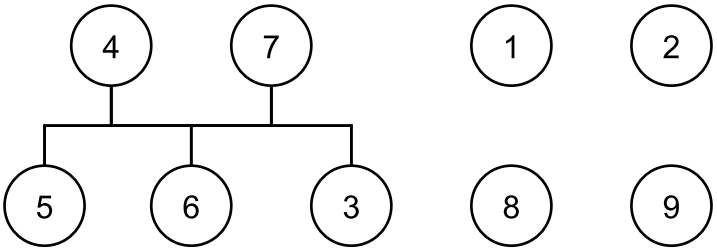


Figure 4.2: The Romanov family: Tsar (4), Tsarina (7) and three of their children (5, 6 and 3). The other found skeletons: The doctor (2) and the three servants (1, 8 and 9) are assumed to be unrelated.

As this research is restricted to STR genotype data, not all of the indications as used by Gill et al. could be incorporated in the experiment. Using the here proposed method, only genotype data for four STR marker was used, so that the algorithm found several different and more likely pedigree than the one found by Gill et al. (1994) [8]. The different result can be explained by the amount of markers used as well as the choice of STR markers, which can be considered as outdated and of low practical use. In a follow-up experiment using generated data for the same pedigree, the family relations were reconstructed correctly (cf. experiment B.9) due to a higher quality of the genotype samples. An additional test with missing genetic profiles using genotype samples from the Romanov case was not performed, as all individuals were aligned in at most two generations, so that no single genotype sample could be removed reasonably to simulate a reconstruction scenario with incomplete samples.

Due to the confidentiality of genotype data, the author had no access to perform more experiments involving real human genotype data, as for example in a case like the one reported by Egeland et al. (2000) [7]. However, the results demonstrate that the proposed method can take real human DNA data as an input, which was obtained from a real investigation case (cf. experiment B.8 on page 59).

4.6 Undirected loops in pedigrees containing incest relations

Incestuous pedigree structures were estimated to be more difficult to solve, because inbreeding results in a reduced variety between genetic profiles of the individuals in the pedigree. Hence, less informative alleles remain available to indicate the correct kinship relations among the individuals. In that case, individuals are genetically more similar and it is expected that the determination of the correct kinship relations is more difficult.

The effect of incest on the genetic profiles may be comparable to the effect of observing less genetic marker data in a non-incestuous pedigree since both effects lower the informational content of the data and hence making pedigree reconstruction using genetic data more difficult.

An experiment (B.11) was performed to investigate if pedigrees containing incestuous family relations can be reconstructed with the proposed method.

Using complete genotype samples, the incestuous pedigree was reconstructed successfully in 27 out of 100 trials (compared to 74 out of 100 successful reconstructions using a non-incestuous pedigree; as in B.10). Using incomplete samples, the pedigree was not reconstructed successfully in any out of five trials (compared to four out of five successful search trials for a non-incestuous pedigree; as in B.10). It is expected that the original pedigree would have been found, if more trials would be performed. However, this result confirms the

increased difficulty to reconstruct pedigrees involving inbreeding. The reconstruction of the used incestuous pedigree can be estimated as hard, as it contains two inbreeding loops involving each three individuals, of which two individuals are involved in both of inbreeding loops. Moreover, in the incomplete condition, the untyped individual was one of those two individuals, which may have increased the difficulty even further.

4.7 Number of loci

The accuracy of pedigree reconstruction can be analyzed as a function of the number of available loci in the genetic profiles, as in [4]. An increasing number of loci was observed to improve the reconstruction performance. In most experiments, the genotype data consisted of 10, 15 or 20 loci per fingerprint.

In the scope of this research, some experiments were performed which a low number of loci[12]. In particular, if only a single genetic locus ($K = 1$) was used, then the search algorithm usually found different pedigrees besides the original pedigree. On the other hand, these found pedigrees also represented the stochastically best explanation, i.e. the maximum likelihood pedigrees, so that the solution found was considered as correct. This is supported by the results of another experiment (cf. B.4), which lacked in the quality of the (generated) genotype samples (only 10 markers with only 8 different possible alleles).

In some experiments, which only used genotype data from a single STR marker, sometimes gender-inconsistent pedigrees were found. Increasing the number of used loci generally led to a higher quality of the sampled pedigree, and this effect is expected also in terms of gender-consistency. Using sufficiently many markers, as in any realistic scenario the algorithm could be applied in, gender-inconsistent pedigrees were not found during the experiments, and they are not expected to be found using the proposed method.

The overall reconstruction performance can be said to decrease with a lower number of loci used, which effect can be explained by a lack in quality of the genotype data rather than by a deficit of the proposed search algorithm. In the scope of this research, no single experiment was dedicated to the reconstruction performance of the algorithm in dependence on the number of used genetic loci. However, such results are expected to be consistent to those obtained by Riester et al. (2009) and would require more sophisticated methods to measure the performance, respectively the accuracy (than the ones used in this research) [4].

4.8 Search parameters

4.8.1 Number of search steps

It can be taken as granted that with an increasing number of search steps, the probability that the maximum likelihood pedigree be found increases. This is consistent with results obtained in all experiments. On the one hand, with an increasing number of search steps it becomes more likely that the maximum likelihood pedigree is found, while on the other hand the computation time increases. This limited the extent to which experiments could be conducted in the scope of this thesis.

One is interested in finding the number of required search steps required to reconstruct the pedigree. Due to the random number generator involved, and the disparity between the search results from trial to trial, one rather aims to find the number of search steps required to reconstruct the pedigree with a desired confidence. Therefore, the author choose to made a trade-off in the choice of the number of search steps in order to minimize computational resources while maintaining a given confidence of not being wrong, i.e. the confidence about that the best found pedigree is also the maximum likelihood pedigree.

4.8.2 Stop-criterion based on convergence

Instead of specifying a fixed number of search steps to perform, a convergence criterion was successfully applied in the experiments. This allowed a more precise estimation of the required number of search steps, which adapts to the current performance of the search. In particular, the search stopped if no higher likelihood pedigree was found for t successive steps and continued if the likelihood was still climbing.

The effect of changing t was investigated in an experiment (cf. B.5). A lower value for t employs a stricter stop criterion whereas a value for t weakens the stop criterion based on convergence. The results are presented in figure 4.3.

[12] These experiments are not reported in this thesis

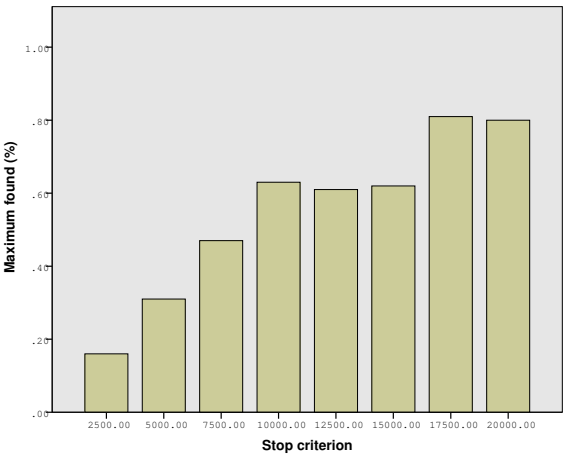


Figure 4.3: The effect of the stop criterion t on the reconstruction success rate: With a less strict stop criterion, i.e. a larger t, the pedigree was reconstruction more often.

In other experiments different values for t were used: $t = 2000$ (cf. B.18), $t = 5000$ (cf. B.4), $t = 10000$ (cf. experiments B.3, B.6, B.7, B.8, B.9, B.10 and B.11).

On the one hand, a less strict stop-criterion based on convergence improved the performance of the algorithm in terms of a higher number of times the pedigree was reconstructed successfully. On the other hand, using a less strict stop-criterion, the number of search steps increased, which entailed additional computational demands.

4.8.3 Step size

Some experiments were performed to investigate in the performance of the algorithm using different step sizes s. Initially a step size of $s_{max} = 10$ was used, so that the step size s was randomly chosen from the range of $1 \leq s \leq 10$ during the generation of a proposal. This settings result in poor performance regarding the quality of the proposed pedigree as indicated by a low acceptance rate (cf. experiment B.3). Using a lower value for s_{max} slightly better reconstruction performance was achieved (cf. experiment B.4).

Reducing the step size, the generated pedigrees are more correlated to each other. In contrary to the original idea of the Metropolis-Hastings algorithm, which aims to propose independent samples from a distribution, correlated proposals are acceptable in a search algorithm, which merely aims to find the maximum, or the k-maxima respectively.

Two experiments (cf. B.6 and B.7) were dedicated to investigate in the effect of the step size s, respectively s_{max}, on the performance. With a lower choice of s_{max} the acceptance rate increased and the pedigree could be reconstructed successfully more often (cf. B.6). In particular, the maximum likelihood was found more often and earlier during the search, if smaller steps were performed. In a follow up experiment, the step size s was controlled so that the factor of randomness during the search was removed. The results were similar and they highlight the importance of the step size parameter on the performance of the algorithm.

Finally, the parameter choice for the maximum step size was lowered to $s_{max} = 3$, which uses a random number of transitions s from the interval $[1, 3]$. This choice was made as a trade-off between the experimental results which suggest a value of $s_{max} = 1$, and the belief that larger transitions steps sizes might be beneficial in some situations, e.g. if a individual needs to be relocated in the pedigree which requires multiple transitions, such as switching the age-ordering as well as the parent sets. Another reason against choosing the minimum value of $s_{max} = 1$ was that the algorithm may behave like a greedy-algorithm. In that case the search is believed to linger in local maxima for a considerable amount of time and only being able to leave this state with a very low probability, i.e. the combined probability of accepting three proposed steps of step size $s = 1$ subsequently.

4.8.4 Acceptance-rejection behavior

In an early version of the search algorithm, poor search performance was observed across experiments with changing number of loci $|K|$ and number of individuals N. The algorithm accepted either too many proposals and thus could not find the maximum likelihood pedigree, or accepted too few proposed steps which led to a greedy-like behavior in which the likelihood quickly improved during the search towards a state which did not necessarily also represent the maximum (a suboptimal local maximum). This problem was caused by different likelihood magnitudes dependent on the size of the genotype sample (see section 3.2.2 on page 19). Introducing a scaling factor helped to normalize these values and to achieve the same acceptance-rejection behavior of the

algorithm independent of the number of involved individuals and the number of loci. Using a simple try-and-error approach (which is not documented in this thesis), an acceptance factor of $f = 50$ was chosen which showed robust performance across various experiments. Lower values increased the acceptance ratio and thus promoted acceptance of proposed pedigrees whereas higher values decreased the acceptance ratio and thus demoted acceptance. In other words, with larger f the algorithm behaved increasingly greedy-like whereas with smaller f the algorithm accepted pedigree increasingly independent of the likelihood of the underlying pedigrees.

4.9 Search strategies

In the thesis several strategies, variants of the search were used: A Metropolis-Hastings search, based on random transitions, and a guided search using stochastic transitions, as well as another guided search strategy with a more sophisticated pedigree proposal generation.

The Metropolis-Hastings search algorithm (strategy 1a) could also be used for sampling, in particular to generate samples x according to the distribution of pedigrees given the genotype data. This application was not studied intensively enough in order to report results here. However, by using the Metropolis-Hastings algorithm, it can be guaranteed that it will to converge to the stationary distribution. From the property can be derived that also the maximum likelihood pedigree will be found given that the algorithm can run for long enough. From a theoretical viewpoint, this represents the main advantage of using the Metropolis-Hastings approach.

In contrast to that, the guided search variants (strategy 2a and 2b) cannot be used for sampling due to the score function (as described in 3.5.4), which changes the underlying distribution. Similarly, the random walk (strategy 1b) cannot be used for sampling with the Metropolis-Hastings algorithm, because its transitions do not satisfy the detailed balance criterion.

The search based on the Metropolis-Hastings algorithm (search strategy 1a), was successfully applied to reconstruct pedigrees of up to 8 individuals. One such experiment is reported in B.10, in which a pedigree consisting of $N = 7$ individuals, as depicted in figure B.9a on page 61, was reconstructed successfully in 4 out of 5 performed trials using incomplete samples. Using a higher number of individuals increased the number of steps required to find the maximum likelihood pedigree up to a level, which would no longer permit to obtain significant results. Therefore, such experiments were spared out using the Metropolis-Hastings algorithm.

The guided search (variant 2a) successfully reconstructed a pedigree with 8 individuals using incomplete samples (see experiment B.15 on page 70). The search found the maximum likelihood pedigree two times, respectively three times, out of 25 search trials (in average 10%).

Assuming that this result can be generalized, there is a chance of 10% to find the maximum likelihood pedigree, so that the guided search would be required to run for 29 trials to reach 95% confidence, respectively 44 trials to reach 99% confidence, to reconstruct the pedigree. The time to perform such a search is estimated to take about 7, respectively 10.5 hours, using parallel processing using 12 CPU's.

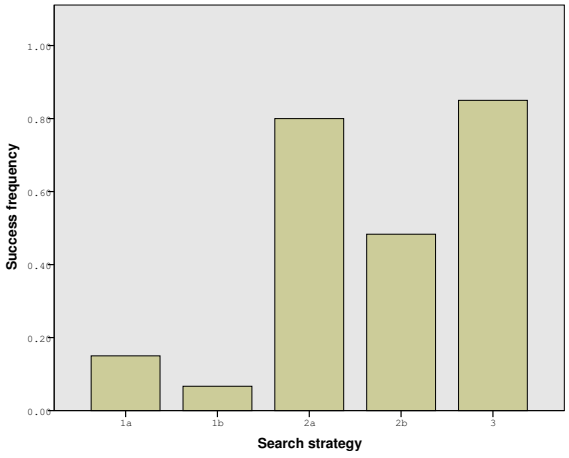


Figure 4.4: Performance comparison between different search strategies (cf. experiment B.18) using complete samples: the guided search strategies (2a, 2b, 3) found the maximum likelihood pedigree more often than the non-guided search strategies (1a and 1b).

Different search strategies were proposed. In an experiment using complete samples, the performance was compared across using six different pedigrees, five different search strategies, and ten different search trials per setting (cf. B.18). In most experiments, guided variants of the search outperformed the original Metropolis-

Hastings algorithm, and found the maximum likelihood earlier in the search (cf. experiments B.16,B.17, and B.18), but not in a single experiment using a small pedigree (with $N = 5$; cf. B.14).

To resolve invalid kinship relations after every transition, 'masking' (strategies 1a and 2a) using the extended pedigree matrix were more successful than 'dropping' strategies (1b and 2b) which only used the (upper-triangular) pedigree matrix to represent admissible pedigrees (cf. experiment B.18).

4.10 Pedigree representation

A novel representation for a pedigree was proposed that uses the extended pedigree matrix $\hat{m}_a$, which was used in combination with a 'masking' strategy for transitions to resolve invalid kinship relations. The benefit of this representation was investigated for complete samples (see experiment B.16 on page 71) as well as in a follow up experiment using incomplete genotype samples (see experiment B.17 on page 74). Especially, the guided search variant in combination with the extended pedigree matrix $\hat{m}_a$ (i.e. search variant 2a) showed an increased performance compared to the other search variants (cf. experiment B.18), particularly in a multi-generation pedigree (see experiments B.16 and B.17).

4.11 Simulated annealing

Simulated annealing could not successfully be applied to improve the search. It was investigated if simulated annealing methods can improve the proposed search algorithm. In two experiments an annealing schedule was applied to control the transition step size s_{max} (cf. experiment B.12), respectively the transition step size s (cf. experiment B.13). Both experiments did not improve the performance. Therefore, without optimizing the annealing schedule, no further investigations regarding the applicability of simulated annealing methods were conducted in the scope of the thesis.

4.12 Reconstruction difficulty

Finally, a notion of the reconstruction difficulty of a pedigree using the proposed algorithm can be expressed with the mean number of steps required to find the maximum likelihood pedigree. The number of required search steps to reconstruct pedigrees successfully was subjectively[13] observed to correlate mainly with the number of individuals N, but also other factors contributed to an increase of the reconstruction difficulty, such as a higher number of generations, a higher number of kinship relations among the individuals (i.e. the connectedness), or less informative alleles per genetic profile.

Multi-generation pedigrees were investigated in two experiments, using complete samples (cf. experiment B.16) and using incomplete samples (cf. experiment B.17). These pedigrees included many generations, 7 (respectively 5), relative to the number of total individuals in the pedigree, i.e. 8 (respectively 6). These multi-generation pedigrees required relatively many search steps using the proposed method to find the maximum likelihood pedigree, and thus, they are considered as harder to reconstruct.

Furthermore, sparsely connected pedigrees were observed to be easier to be reconstructed, e.g. the pedigree of the Romanov case (cf. experiments B.9 or B.18), which contained four completely unrelated individuals. In an experiment (cf. B.18), the pedigree of the Romanov's could be reconstructed most successfully (see figure B.26b on page 76), even though it contains many individuals ($N = 9$) relative to the other pedigrees considered in that experiment.

The latter can be caused by observing too few genetic loci, or markers with too little variety between individuals, or due to incestuous family relations.

[13] A subjective observation of the author due to the difficulty to compare across a changing number of individuals N, as already explained in section 4.3.

5 Discussion

5.1 Performance

The proposed method, particularly the search based on the Metropolis-Hastings algorithm (strategy 1a), can guarantee that the maximum likelihood pedigree will be found given enough time. Under limited computational resources, this is claimed to be possible up to an estimated confidence. By performing many trials, the error of being wrong can be minimized and the confidence can be estimated using simulation studies. Performing such simulations may be cumbersome.

In the conducted experiments, guided search strategies generally performed better, and were able to reconstruct the pedigree using fewer search steps, but in the worst case guided variants might case take considerably longer to find the maximum. The local probabilities used to guide the search might be deceptive and not be able to guide the search towards the area of the maximum likelihood solution.

5.2 Incomplete samples

It is claimed that the proposed method can be used for pedigree reconstruction using incomplete samples. The method is capable of being applied to any number of unobserved individuals but experiments conducted in the scope of this thesis only involved pedigrees of which the genetic profiles were observed for all individuals except for one. Thus, the experimental results lack in demonstrating the approach to a wider range of problems, which involve a higher number of incomplete samples. Follow-up studies should be conducted to prove the claimed applicability of this approach.

In particular, it was assumed that the number of unobserved individuals N_0, which is necessary to explain higher-level kinship relations, e.g. grandparent-child relations, was provided. This was implicitly done by providing N_0 extra individuals with "empty" genetic profiles, which consists of missing allele observations on all genetic markers. In realistic scenarios, the number of missing individuals to explain the pedigree may not be available.

The method can simply be extended by considering a number N_0, which is at least a high as the true number of additional required individual. Using this method, the algorithm may - depending on the pedigree to be reconstructed - find pedigrees, in which unnecessary additional individuals are identified as unrelated. This is not considered as problematic, but it is rather the fact that all extra individuals have identical genotype information, which is based on the prior allele distribution rather than actual observation. This would cause multiple similar solution in which the extra unobserved individuals are merely interchanged among each other, e.g. in a pedigree that merely requires only two additional individuals for the reconstruction, but four extra unobserved were provided in the reconstruction, there exists already twelve similar solutions. Future studies are required to extend pedigree reconstruction to be used in scenarios with multiple missing profiles.

Regarding the computational complexity, sophisticated methods are required, to keep this problem computationally tractable. This may involve limiting the maximum distance between to typed individuals, i.e. the maximum distant of distant relatives to consider, so that larger distant relatives are considered as unrelated. Other methods may constrain positions in a pedigree, in which additional untyped individuals are legal at all.

Genotype datasets may also be incomplete only for a few loci rather than entirely missing genetic profiles, as for example with degraded DNA. These missing loci can be handled in a transparent way, as the Bayesian network can deal with incomplete information, so that all available genotype information can be used for the reconstruction. To give a counter-example, Cowell (2012) had to remove some genetic profiles from a genotype dataset (for shrimps), because the method was not able to handle genotype data, which is incomplete on some markers for just some genetic profiles [5]. Thus, the proposed method extends pedigree reconstruction to cases involving incomplete genetic profiles, and does not require additional measures such as removing data, in order to make the reconstructed work.

5.3 Correctness

In the proposed method (and in other studies, e.g. [5]), gender-consistency was not taken into account and hence the algorithm might come up with gender-inconsistent pedigrees (which is actually not a pedigree in that sense), which have higher likelihoods than the actual maximum likelihood pedigrees. In the conducted experiments using multiple genetic loci, gender inconsistent solutions were never found, but in theory, in specific cases, they might occur as well.

To ensure correctness of the method and to be applied in practice, every pedigree is required to be checked for gender-inconsistency. It is possible to include a check for gender-consistency (e.g. based on marriage-graphs as in 2.2b), which might be part of future researches or developments.

Besides that, all proposal pedigrees fulfill the other requirements for biologically valid pedigrees. Age consistency excludes directed cycles from the pedigree, i.e. individuals cannot be their own ancestors, and parent compliance guarantees that every individual has two parents, of which are at most two in the pedigree.

5.4 Algorithm and implementation

The reconstruction algorithm was implemented in Matlab. The implementation should be optimized for runtime performance, which may require the use of a more efficient programming language than Matlab, especially to be employed in a software application.

5.5 Testing

5.5.1 Sample pedigree structures

In several experiments, the pedigrees that were subject to be reconstructed were constructed manually by the author prior to the experiment. These pedigree structures may contain subliminal factors. These pedigrees may be biased towards specific pedigree structures, and it may not represent average pedigrees that allow results of the pedigree reconstruction to be generalized. Another way to create sample data is generating them automatically (as in [4]).

Further, only a few manually constructed pedigrees were employed to obtain results, but the number of different pedigrees is considered as too low, and thus being not representative enough for generalization of the obtained results. Hence, the experimental results may merely provide an estimate rather than representing accurate results.

5.5.2 Sample genotypes

The use of realistic genotype data is desirable to demonstrate the applicability of the proposed method to pedigree reconstruction. The experiments conducted during the research mostly used generated genotype data with exception of the data from the Romanov case (as reported by Gill et al. (1994) [8]; see experiment B.8), which lacked in quality of the observed samples due do outdated genotyping technology. Thus, experimental results of the method using contemporary genotype data are demanded. The use of genuine human data is complicated, and the author had no access to proper datasets due to the confidential treatment of human genetic data. Animal-based data, as used by other studies, may have been easier accessible for this research.

Instead, this research employed a genotype generator, which created ideal genotype data according to the assumptions about the prior distribution and the mutate rate per allele μ. Both, the prior allele distribution and the mutation rates, were additionally assumed identical for each allele. Further, the allele distribution was assumed to be distributed uniformly and remained unaltered after inheritance events for all generations in the pedigree. In the experiments during this research, such data was generated for usually ten to twenty different genetic markers, and eight to twenty different allele types per loci, and all generated allele were distributed accordingly. In realistic scenarios, statistical data about the allele distribution and the mutation rates are derived from large population studies, and do no used generated data. Thus, the used datasets may represent idealized datasets, and it need to be demonstrated that realistic datasets can be coped with as well.

Furthermore, the effect of mutations on the likelihood was increased ($\mu = 0.01$) during the experiments, compared to a biologically more realistic value ($\mu \approx 0.001$), so that mutations contributed to a larger extend to the results, which was intentional. Besides the changed likelihoods of the pedigrees, this may also have changed the way the search space was traversed, so that results may not be as representative as assuming a more realistic mutation rate.

5.5.3 Enumeration

For complete samples, i.e. involving genotypes of all individuals, a simple enumeration method is used, as in [1], in order to verify if the true pedigree is also the maximum likelihood pedigree. This approach is feasible for up to $N = 8$ individuals, and for larger pedigrees knowledge about the maximum likelihood pedigree was merely assumed than proven. Here, a faster and more efficient method could have been used, such as the one from Cussens et al. (2013) [6], which can be used for much larger pedigrees (at least for $N \leq 64$ individuals) for this verification. However, for incomplete samples none such method exists to the knowledge of the author.

5.6 Limitation and assumptions

The proposed method, particularly the search based on the Metropolis-Hastings algorithm (strategy 1a), can guarantee that the maximum likelihood pedigree will be found given enough time. Using limited computational resources, this is claimed to be possible only up to an estimated confidence. By performing many trials, the error of being wrong can be minimized and the confidence can be estimated using simulation studies, but performing such simulations may be cumbersome. An exhaustive enumeration across all pedigrees is infeasible, so that an appropriate subset of pedigrees is required to be selected for such simulation studies, which allows generalizations to all kinds of pedigrees. Hence, estimates about the confidence using simulation studies may be inaccurate.

It was assumed that all genetic markers are independent, which is a reasonable assumption in many realistic scenarios. Contemporary DNA typing technologies employ genetic markers from different chromosomes. Due to the spatial distance between the markers, linkage between those markers is not expected.

As A. Almudevar (2007) argued, a likelihood function could also incorporate linkage among loci but on the cost of losing the product form to compute the likelihood for multiple loci, so that more sophisticated methods are demanded *[13]*.

The reconstruction algorithm was limited to be used with STR data. It may also be useful to be able to reconstruct pedigrees using SNP data. However, STR are available for many species. Hence, this method is not limited to reconstruct human pedigrees and is widely applicable.

It was assumed, that all founding individuals of the pedigree are unrelated to each other, which seems to be widely an accepted assumption as it was made by many other studies regarding pedigree reconstruction (cf. section 2.6.1).

This research was limited to an unconstrained search, which is solely based on STR genotype data, and thus, any additional data, as for example in form of age or gender information was not used. These can be used to constrain the search space.

Kinship relations may be known. In a simple example reconstruction, adapted from a parental test, the kinship relation between mother and child is known and out of question, but a third individual's relation to both is subject to be reconstructed. Known kinship relations may easily be incorporated using the proposed method by enforcing these kinship relations in the representation. This may be done by assigning and fixing a certain element of the pedigree matrix m_a, which represents a particular kinship relation, to either 1 (present) or 0 (absent). This implies that transitions need to be modified as well.

In realistic settings, age information may be available, which could be taken into account into the reconstruction algorithm, and was previously successfully incorporated in several approaches for pedigree reconstruction based on complete genotype samples (see section 2.6.1). Age information can be used to employ age constraints on the search space. These constraints take place between a pair of individuals. The current algorithm does not incorporate such information, but it could be extended to incorporate age information. In particular, it constraints the space of possible age orderings, so that the set of possible kinship relations is automatically reduced to kinship relations between parent and child, in which the parent is required to be older than the child.

Similar to age information, also information about gender can be used to constrain the space of possible pedigrees. In particular, knowing the gender of a single person, the other parent any of its children must be of the opposite gender.

Some kinship relations are unlikely to occur in practice, indicated by a very low local likelihood of an individual's genotype given its parents' genotypes. Such kinship relations can possibly only be explained with a high number of mutations taking place. In this research, all kinship relations were considered, in order to obtain a statistically correct solution, even if genotype pairs seem to be incompatible. In practice, such genotype pairs could be considered incompatible, and according kinship relations may be regarded as impossible, and hence could be excluded from the search space, making the search problem smaller.

5.7 Prospects

With advancing genotyping techniques and (perhaps changes in society regarding the use of genetic data), genotyping data may become available for whole genomes and complete populations [4]. With such an amount of data, kinship relations can be determined accurately, and it may become a question of reconstructing the past generations without available genetic data in order to explain the genetic relations between present generations.

6 Conclusion

Pedigree reconstruction generalizes contemporary applications, such as parental tests, in which only the kinship relations between the parents and the child are subject to question, to more complex family relations involving more than just three individuals. In this thesis, a method for pedigree reconstruction based on genetic profiles was proposed. It is the problem of finding the stochastically best explanation of family relations given only the genetic profiles of several individuals.

Current challenges like the handling of incomplete genotype samples and mutations were addressed and can be handled using the proposed method. Mutations are fully supported by this approach, whereas in the pedigree reconstruction using incomplete genotype samples the number of additional extra individuals (for whose genotype information was not observed) is required. Previous studies already developed efficient methods for a simplified problem of pedigree reconstruction, but they did not address the handling of incomplete genotype samples, and some of them also neglected the possibility of mutations.

The problem of pedigree reconstruction is an optimization problem, in which one searches for the graph that maximizes the likelihood of the observed genotypes. The search takes place is the space of all possible pedigrees, which size is large and high dimensional and depends on the number of individuals, i.e. nodes in the graph. The problem of pedigree reconstruction has exponential complexity dependent of the number of individuals considered. Exploring the space of all possible pedigrees in an exhaustive manner is computationally intractable, and a solution to this problem is non-determinable in polynomial time.

Therefore, a search based on Markov Chain Monte Carlo methods was proposed in this thesis. Originally those algorithms are used to draw samples from an unknown distribution, but can be similarly well be used to find the maximum in a distribution. In particular, a search based on the Metropolis-Hastings algorithm is proposed, which creates a "random walk" in the pedigree space based on small steps. This algorithm proposes steps in the neighborhood of the "current pedigree", which are made with random probability based on the likelihood ratio between the current and the proposed pedigree.

For that, a novel pedigree representation was proposed - the extended pedigree representation - which facilitates genuine MCMC simulations to be performed. In particular, it enables transitions to occur with invariant probabilities, a requirement when using the Metropolis-Hastings algorithm. It was demonstrated how this representation allows to search in the space of pedigrees. Furthermore, this novel representation also enables sampling approaches, in which effectively independent samples from the desired distribution of pedigrees can be drawn, which approach was not studied in the scope of this thesis. Eventually, using the novel pedigree representation, the search space was successfully constrained so that almost exclusively biologically valid pedigrees are generated; the only exception represent gender-inconsistent pedigrees (as e.g. in figure 2.2a), which can be generated in theory, but in practice the search only halts at those inconsistent pedigrees, if the given genotype information consists of insufficient loci.

The stochastic model used in this method builds up on the principles of genetic inheritance, which were initially discovered by Gregor Mendel in the 19th century, and transmission probabilities of the genotypes from parent to offspring including mutational events. Besides the principles of Mendelian inheritance, this approach assumes a Hardy-Weinberg equilibrium, a uniform allele distribution with independent markers (i.e. unlinked markers), and a mutation model. The mutation model assumes a certain fixed mutation rate per allele, which can vary between genetic markers. Moreover, the stochastic model assumes that parents, which are not part of the pedigree, are unrelated and are considered as founding individuals, which have independent alleles.

Intending to develop a general method with wide applicability, the proposed method does not require any additional data, such as age or gender information, besides the genotype samples itself. As genetic information Short Tandem Repeat (STR) marker data from the autosomal chromosomes are used, which is available independent of the gender. The accuracy of the approach can be improved by increasing the number of genetic loci, for which STR marker data was observed. Furthermore, the method can be extended, similar to previous studies, to incorporate age and gender information, which can be used to constrain the search space.

To model the genetic relations among a set of genotypes, a Bayesian network, i.e. a probabilistic network, was employed. The advantages of this method is that incomplete data in form of missing alleles or whole missing genetic profiles can be handled uniformly, as well as mutations. Biologically admissible pedigrees were modeled using directed acyclic graphs, constrained to at most two incoming arcs per node, which represent the kinship relations to the parents. An inference algorithm computed the likelihood of observing the genotypes given the family relations, which were subject to be reconstructed. To make use of multiple markers, the likelihood was integrated over multiple available genetic loci. Finally, this notion of the likelihood was used to determine the maximum likelihood pedigree, i.e. the pedigree structure that can explain the observed genotypes best.

The Bayesian network inference algorithm, i.e. the Junction Tree algorithm, showed a poor time performance (at least in the used implementation in Matlab). The used inference algorithm represented the bottleneck of the algorithm and consumed most computation time. Regarding the amount of times the inference algorithm

was executed within a single search, and the high demands of the Metropolis-Hastings algorithm to run for a considerable time, a poor overall performance of the search algorithm was achieved, and thus its applicability is limited to small pedigrees. Future investigations are recommended to use an optimized Bayesian inference algorithm to increase the performance.

Using a modified search, a guided search, which exploits local knowledge about likely parents in the proposal generation, the search performance could be improved and pedigrees could be reconstructed more efficiently. It remains subject for future investigations, if the guided search variants scale well with larger pedigrees.

Finally, the search can only reconstruct pedigrees up to an estimated confidence, which can be estimated using simulation studies. In particular, given that a real world scenario, in which the computation may be required to be finished after 24 hours, and the availability of 12 CPUs, pedigrees consisting of up to eight individuals could be reconstructed confidently, which was shown experimentally.

References

[1] Anthony Almudevar. A simulated annealing algorithm for maximum likelihood pedigree reconstruction. *Theoretical Population Biology*, 63(2):63–75, 2003.

[2] Tien-ho Lin, Eugene W. Myers, and Eric P. Xing. Interpreting anonymous DNA samples from mass disasters – probabilistic forensic inference using genetic markers. *Bioinformatics*, 22(14):298–306, 2006.

[3] Robert G. Cowell. Efficient maximum likelihood pedigree reconstruction. *Theoretical Population Biology*, 76(4):285 – 291, 2009.

[4] Markus Riester, Peter F. Stadler, and Konstantin Klemm. Franz: reconstruction of wild multi-generation pedigrees. *Bioinformatics*, 25(16):2134–2139, 2009.

[5] Robert G. Cowell. A simple greedy algorithm for reconstructing pedigrees. *Theoretical Population Biology*, 83(0):55–63, 2013.

[6] James Cussens, Mark Bartlett, Elinor M. Jones, and Nuala A. Sheehan. Maximum Likelihood Pedigree Reconstruction Using Integer Linear Programming. *Genetic Epidemiology*, 37(1):69–83, 2013.

[7] T. Egeland, P.F. Mostad, B. Mevag, and M. Stenersen. Beyond traditional paternity and identification cases: Selecting the most probable pedigree. *Forensic science international*, 110:47–59, 2000.

[8] P. Gill, P.L. Ivanov, C. Kimpton, R. Piercy, N. Benson, G. Tully, I. Evett, E. Hagelberg, and K. Sullivan. Identification of the remains of the romanov family by dna analysis. *Nature Genetics*, 6(2):130–135, 1994.

[9] A.J. Jeffreys, J.F.Y. Brookfield, and R. Semeonoff. Positive identification of an immigration test-case using human DNA fingerprints. *Nature*, 317(6040):818–819, 1985.

[10] J.M. Butler. *Forensic DNA typing: biology, technology, and genetics of STR markers*. Elsevier Academic Press, 2005.

[11] Wim Wiegerinck, Bert Kappen, and Willem Burgers. Bayesian networks for expert systems: Theory and practical applications. In Robert Babuska and Frans C.A. Groen, editors, *Interactive Collaborative Information Systems*, volume 281 of *Studies in Computational Intelligence*, pages 547–578. Springer Berlin Heidelberg, 2010.

[12] van Dongen C.J., Slooten K., Slagter M., Burgers W.G., and Wiegerinck W.A.J.J. Bonaparte: Application of new software for missing persons program. *Forensic Science International: Genetics Supplement Series*, 3(1):119–120, 2011.

[13] Anthony Almudevar. A graphical approach to relatedness inference. *Theoretical population biology*, 71(2):213–229, 2007.

[14] A. Piccolboni and D. Gusfield. On the complexity of fundamental computational problems in pedigree analysis. *Journal of Computational Biology*, 10(5):763–773, 2003.

[15] M. Fishelson and D. Geiger. Exact genetic linkage computations for general pedigrees. *Bioinformatics*, 18(Suppl. 1):S189–S198, 2002.

[16] K. Bruijning-van Dongen, C.J.and Slooten, W. Burgers, and W. Wiegerinck. Bayesian networks for victim identification on the basis of DNA profiles. *Forensic Science International: Genetics Supplement Series*, 2(1):466 – 468, 2009.

[17] Stuart Russell and Peter Norvig. *Artificial intelligence: a modern approach (2nd edition)*. Prentice Hall, 2003.

[18] Steffen L. Lauritzen and Nuala A. Sheehan. Graphical models for genetic analyses. *Statistical Science*, 18(4):489–514, 2003.

[19] R.W. Robinson. Counting unlabeled acyclic digraphs. In Charles H.C. Little, editor, *Combinatorial Mathematics V*, volume 622 of *Lecture Notes in Mathematics*, pages 28–43. Springer Berlin Heidelberg, 1977.

[20] Elizabeth A Thompson. Inference of genealogical structure. *Social Science Information*, 15(2-3):477–526, 1976.

[21] Elizabeth Alison Thompson. *Pedigree analysis in human genetics*. Johns Hopkins University Press Baltimore, 1986.

[22] T. Egeland, P.F. Mostad, and B. Olaisen. A computerised method for calculating the probability of pedigrees from genetic data. *Science & Justice*, 37(4):269–274, 1997.

[23] Jin Tian, Ru He, and Lavanya Ram. Bayesian Model Averaging Using the k-best Bayesian Network Structures. *CoRR*, abs/1203.3520, 2012.

[24] D.J.C. MacKay. *Information theory, inference and learning algorithms*. Cambridge university press, 2003.

Appendix

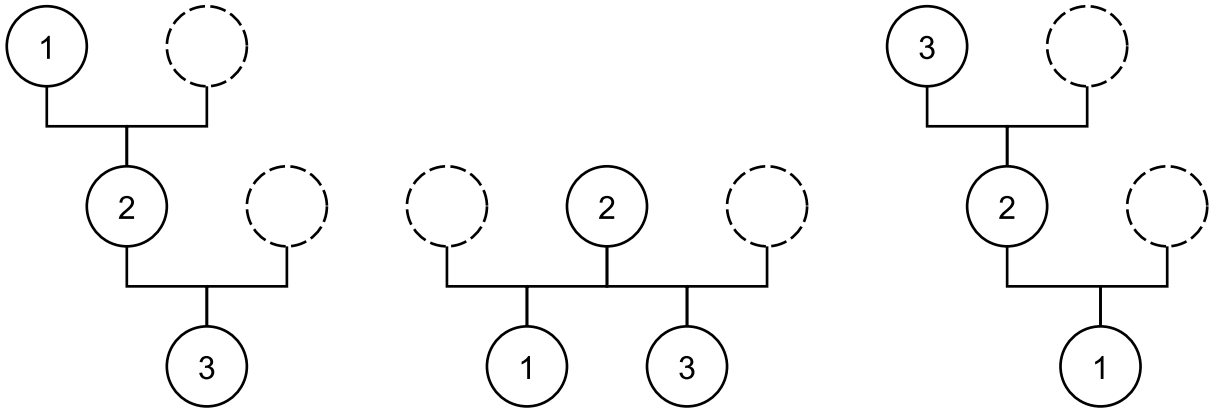
A List of sample pedigrees

1. A pedigree consisting of grandparent, parent and child, as depicted in figure 4.1 on page 36, was used in experiments B.1 and B.2.
2. A pedigree adapted from A. Almudevar (2003) [1, Fig. 1], as depicted in figure B.4 on page 54, was used in experiments B.3, B.4, B.6, B.7, B.12, and B.13.
3. A constructed pedigree with $N = 7$ individuals, as depicted in figure B.9a on page 61, was used in experiments B.10 and B.18.
4. A modified version of pedigree 3 with extra incestuous family relations, as depicted in figure B.12 on page 64, was used in experiments B.11 and B.18.
5. A constructed pedigree with $N = 12$ individuals, as depicted in figure B.25 on page 75, was used in experiment B.18.
6. The pedigree of the Romanov family [8], as depicted in figure 4.2 on page 38, was used in experiments B.8, B.9 and B.18.
7. A constructed pedigree with $N = 8$ individuals over three generations, as depicted in figure 1.1 on page 1, was used in experiments B.15 and B.18.
8. A constructed pedigree with 8 individuals align over 7 generations, as depicted in figure B.22 on page 72, was used in experiment B.16.
9. A constructed pedigree with 6 individuals align over 5 generations, as depicted in figure B.24 on page 74, was used in experiments B.17 and B.18.

B Experiments

B.1 A very small pedigree with three individuals (Grandparent-parent-child)

This pedigree contains of $N = 3$ individuals, arranged as grandparent, parent and child, as depicted in figure 4.1. An uniform allele distribution was assumed consisting of 20 different allele types, ranging from 10 to 29. The mutation rate was assumed to be $\mu = 0.01$. Using the genotype generator, genetic profiles were generated for all three individuals. Mutations were generated according to the mutation rate $\mu = 0.01$. Genotype data for individuals was completely observed. In total, 10 datasets of genetic data were generated. With a pedigree of this size, the maximum likelihood pedigree can be enumerated quickly. Enumeration results show that there are in total three maximum likelihood pedigrees for each of the 10 DNA datasets that can account for the generated genotype data, see figure B.1. Thus, it was aimed to find all three maxima rather than just the original pedigree as in figure 4.1 (also depicted in figure B.1a below).



(a) Original pedigree

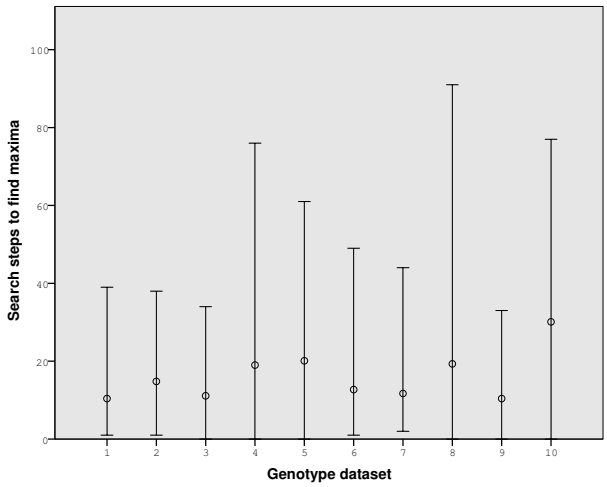
(b) Grandparent I_1 falsely identified as a child of I_2

(c) Inversed order of generations

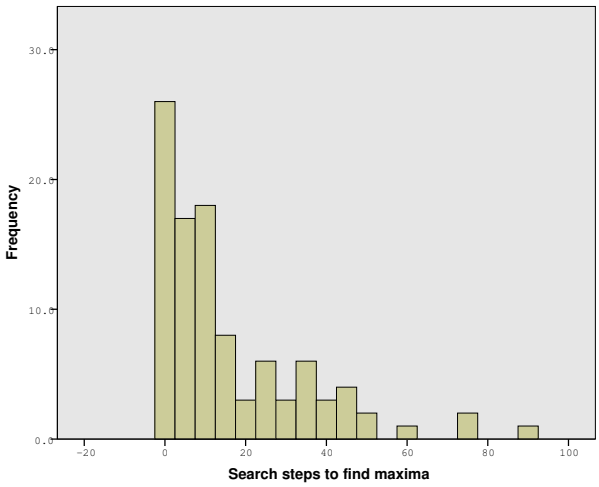
Figure B.1: Given the genetic profiles of individuals I_1, I_2 and I_3, the original pedigree (a) is non-determinable. Three competing pedigrees (a, b, and c), including the original pedigree (a), give equally likely explanations for the correct kinship relations.

The Metropolis-Hastings based search was used with a transition step size of $1 \leq s \leq 10$ and an acceptance factor $f = 50$, which was stopped after all maxima were found. For each DNA-dataset 10 sampling trials were performed.

The three maximum likelihood pedigrees were found confidentially in every single trial for all 10 generated DNA datasets. The number of sampling steps taken to find all three maxima is depicted in figure B.2. The mean acceptance rate for all trials was 16.7%. It took about 1 minute to compute all 10 trials using a standard personal computer.



(a) Maximum, minimum and average number of sampling steps taken per dataset to find all three maximum likelihood pedigrees



(b) Frequency of how many sampling steps were necessary to find all three maxima. The mean number of required sampling steps was 15.96

Figure B.2: Number of iterations to find the maxima

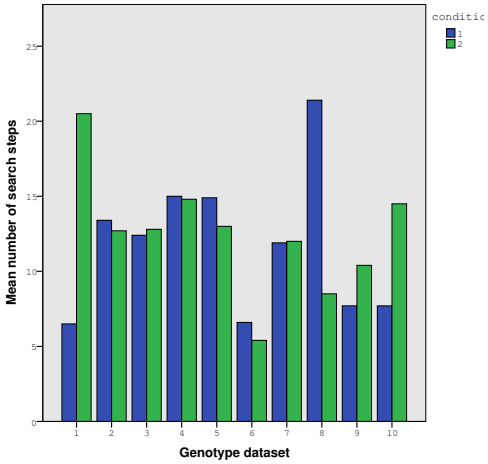
B.2 Incomplete genotype samples from a very small pedigree

A similar experiment as experiment B.1 on the previous page was performed to investigate if missing genetic profiles for a single individual have an effect on the performance of the search for such a small pedigree sample. The same original pedigree was used and the same statistical assumptions were made as in experiment B.1.

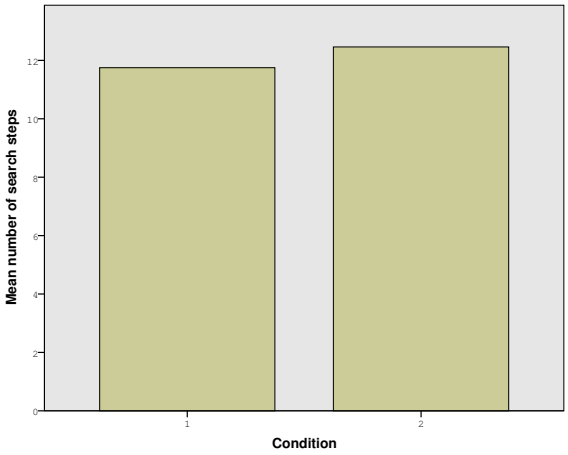
In contrast, enumeration was not performed because the author did not implement an exhaustive search to find multiple maxima for incomplete genotype samples. Instead, it was assumed that the original pedigree (see figure 4.1) is also the maximum likelihood pedigree. The search stopped if a pedigree was found that is as least as likely as the original pedigree.

10 datasets of genotype data were generated. The experiment consists of two conditions. In the first condition all individuals were typed, as in experiment B.1. In the second condition, the genetic profiles of individual i_1 and i_3 were identical to condition 1 but the genetic profile for individual i_2 was removed from the input of all datasets, to simulate a condition in which the genetic profile was not observed.

The original pedigree was found in all trials and datasets in both conditions. No significant differences in the quality of the sampling or the number of taken sampling steps were observed between the conditions.



(a) The mean (over all trials) number of sampling steps taken per dataset to find the original pedigree.



(b) The mean number of taken sampling steps to find the original pedigree for both conditions in comparison. In condition 1 in which individuals were completely observed the mean was 11.75, and in condition 2 in which the genetic profile for individual i_2 was not observed the mean was 12.46.

Figure B.3: Results: Number of iterations to find the maxima

Compared to experiment B.1, the mean number of steps to was slightly lower. Besides the factor of randomness, this is likely to be caused by the slightly different experimental setup, in which the search only had to find a single maximum instead of three maxima.

The total computational time in condition 1 (about 1 min) was faster than in condition 2 (about 5 min). This is due to using the optimized algorithm to compute the likelihood which can only be applied if genotype samples were completely observed. In condition 2 the slower Bayesian network inference algorithm was used.

B.3 A pedigree with eight individuals (1)

This pedigree structure was adapted from the pedigree structure from A. Almudevar (2003) [1, Fig. 1]. It consists of $N = 8$ individuals in three generations, as depicted in figure B.4.

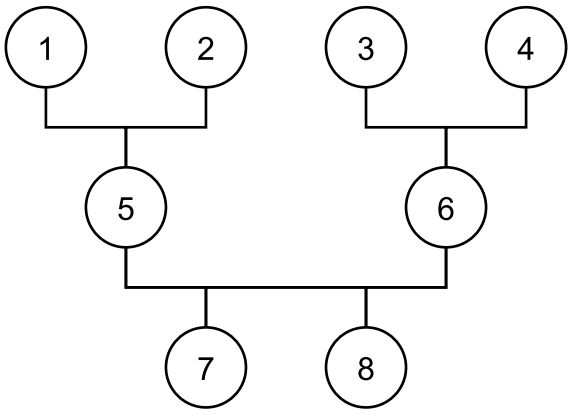


Figure B.4: Sample pedigree structure adapted from A. Almudevar (2003) [1, Fig. 1].

The statistical assumptions were adapted from A. Almudevar (2003) [1], i.e. a uniform allele distribution for all loci with eight different allele types (ranging from STR count values of 10 to 17), and a mutation rate of $\mu = 0.01$. Genotype data was generated for 10 loci, using a mutation rate of $\mu = 0.01$. Genotype data was completely observed and it is depicted in table B.1.

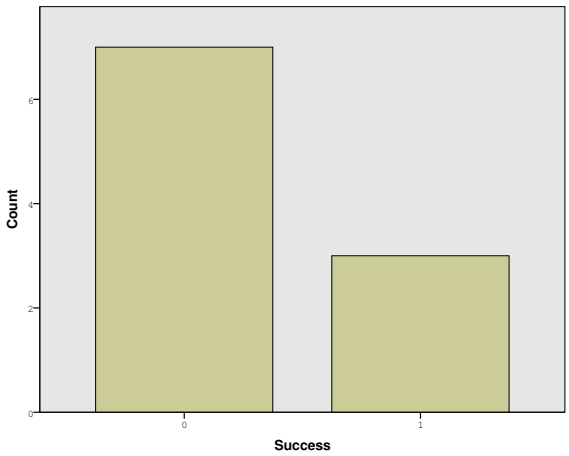
g	i_1	i_2	i_3	i_4	i_5	i_6	i_7	i_8
k_1	16 17	10 15	13 13	12 16	15 16	12 13	12 16	13 16
k_2	11 17	16 17	16 16	14 15	17 17	14 16	16 17	16 17
k_3	10 15	15 16	11 13	17 17	10 15	11 17	11 15	11 15
k_4	12 14	13 15	13 15	11 14	14 15	11 13	11 15	13 14
k_5	17 17	11 15	15 16	11 12	11 17	12 16	16 17	11 12
k_6	11 17	10 15	12 15	12 16	10 11	12 15	11 15	10 12
k_7	13 17	10 12	11 15	11 16	10 13	11 11	11 13	11 13
k_8	11 16	10 16	10 13	12 17	10 16	10 12	12 16	10 12
k_9	13 17	12 15	12 17	11 12	12 13	11 17	11 12	11 13
k_{10}	16 17	10 17	11 14	13 14	10 17	11 13	13 17	10 11

Table B.1: Generated genotype data g for $|K|$ =10 loci for the pedigree structure from A. Almudevar (2003) [1, fig. 1], with 8 different alleles with STR counts in the range from 10 to 17.

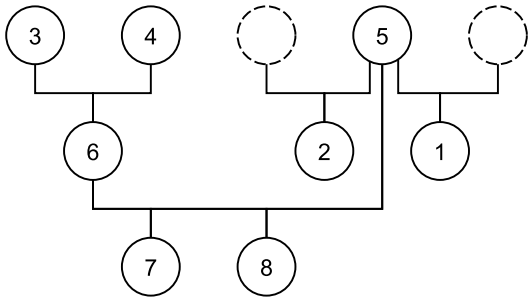
An enumeration was performed for all $|A| = 40320$ different age orderings, which took about 25 minutes using parallel computing. Enumerations discovered that the original pedigree is also the maximum likelihood pedigree with a log10-likelihood of -83.3820.

Search variant 1 (based on the Metropolis-Hastings algorithm) was used, using a maximum transition step size of $s_{max} = 10$ (i.e. a transition step size s randomly picked from the interval $1 \leq s \leq 10$), and an acceptance factor of $f = 50$. The search halted if the maximum likelihood pedigree was found, or no higher likelihood pedigree was found for 10,000 search steps. To increase the significance of the experimental results, the experiment was repeated for 10 trials.

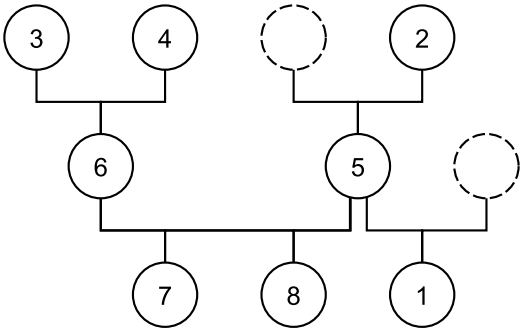
The maximum likelihood pedigree was found in three out of the ten trials (30%), see figure B.5a, after 4,403, 12,941 and 20,271 steps respectively. In the other seven trials, the algorithm converged and stopped before the maximum likelihood could be found. In five of those trials (in trials 1, 3, 5, 6 and 10) a different pedigree was found, which had a lower $\log_{10}$-likelihood of -85.173, see figure B.5.



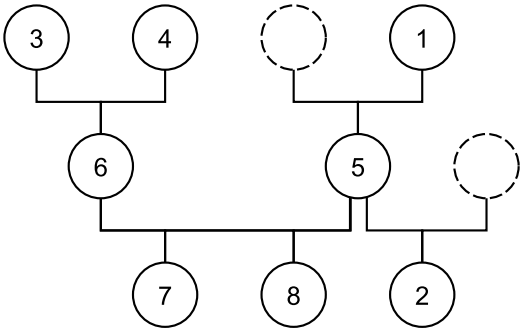
(a) The algorithm found the maximum likelihood pedigree in three out of ten trials.



(b) The most likely pedigree as found by trials 1, 3, 5, 6, and 10. Individuals i_1 and i_2 were falsely identified as children of i_5 who was falsely identified as founder.



(c) A different solution found by trials 1, 3 and 6 in which individual i_1 who is actually a parent of i_5 was falsely identified as a child of i_5.



(d) A different solution found by trials 1, 3, 6, and 10 in which individual i_2 who is actually the parent of individual i_5 was falsely identified as a child of i_5.

Figure B.5: Results of experiment B.3

The computation for all 10 trials took 12 minutes using a standard personal computer. Because of the low success rate in this experiment, another scenario with incomplete genotype observations was eventually skipped due to the high time consumption to perform that experiment and the low expectations of success.

The mean acceptance rate (= .038145) was low, only accepting every 26th proposed sample. The low acceptance rate might indicate a low quality of the proposed pedigree which were hence rejected, finally causing the algorithm to fail. The low acceptance rate might be caused by the parameter choice of the transition step size $1 \leq s \leq 10$. Using a smaller transition step size, may increase the performance of the sampling in terms of the acceptance rate. This was done in the next experiment for the same pedigree (as well as in experiments B.6 and B.7).

B.4 A pedigree with eight individuals (2)

This experimental setup is similar to the experiment B.3 but different sampler parameters were used, in particular a different choice of the number of transition s, as well as a different choice of stopping criteria. The pedigree as in figure B.4 is intended to be found, involving $N = 8$ individuals. For that, the same statistical assumptions were made and the same genotype data is used. This original pedigree was considered to be also the maximum likelihood pedigree.

Search variant 1 (based on the Metropolis-Hastings sampling) was performed using a smaller maximum transition step size of $s_{max} = 3$ (as suggested by results from experiments B.6 and B.7), an acceptance factor $f = 50$. As stop-criteria, the algorithm halted after not improving the likelihood of the best-found solution for successive 5000 search steps, which criterion was combined with a halt as soon as a pedigree with the likelihood of the original pedigree was found. The latter is a rather artificial stop-criterion with the intention to speed up the required computational time to perform this experiment. Similarly, the number of trials was reduced to 5

(instead of 10 as in experiment B.3). This experiment was performed in (1) a completely observed and (2) an unobserved condition in which the genetic profile of individual i_5 was not observed.

In the completely observed condition, the original pedigree was found in three out of five trials (60%).

In the unobserved condition, the original pedigree was not found in any of the trials (0%). Instead, the algorithm came up with solutions, which falsely identified individual i_6 as the parent of either i_3 or i_4, or both, similar as in experiment C, in which the family relation between i_1, i_2 and i_5 were incorrectly identified. Interestingly, the found solutions correctly identified the kinship relations which involve the untyped individual i_5.

In the completely condition, the computation of all five trials took about 5 minutes (1 minute per trial). In contrast, the computation in the unobserved condition took about 8,5 hours in total, respectively 1 - 2.5 hours per trial using parallel computing with 12 CPUs, which is considerably slower. This result is due to the slow performance of Bayesian network inference algorithm, which was only used in the second condition.

Comparing the completely observed condition with the one from experiment B.3, the change of the transition step size parameter from $s_{max} = 10$ to $s_{max} = 3$ seems to have a positive influence on the results, even though the search algorithm was set up to halt earlier after converging (after 5,000 instead of 10,000 steps). This time the original pedigree was found in 60% of the trials compared to only 30% in preceding experiment B.3.

B.5 Determining a parameter for the stop-criterion after convergence

In order to investigate in the number of steps needed to find the maximum confidently, an experiment was performed. For this experiment, the pedigree adapted from Almudevar (2003) [1, Fig. 1] was used, and shown in B.4. Statistical assumptions were made about the allele distribution to be uniform with 20 different allele types and a mutation rate of $\mu = 0.01$. Genotype data was generated for 20 markers, which were completely observed in order to reduce the computational time to perform this experiment. The original pedigree was assumed to represent also the maximum likelihood pedigree. The search was performed using search variant 1 (based on the Metropolis-Hastings algorithm), using a maximum transition step size of $s_{max} = 3$, and acceptance factor of $f = 50$.

As a stop-criterion the number of steps, that the best found pedigree remained constant at a certain likelihood without further improvement, was changed ranging from 2500 to 20000 steps. This created eight different scenarios for this experiment with changing strictness of the stop-criterion. To decrease the time required to conduct this experiment, the search was aborted earlier if all maxima were found before reaching convergence. In order to obtain statistical significance, for each condition 100 trials were computed.

Results are depicted in figure 4.3 on page 40. The pedigree could be reconstructed increasingly confident with a less strict confidence interval, i.e. a larger number of steps without improvement of the likelihood. The total computation time for this experiment was about 20 hours.

B.6 Investigating the effect of the maximum transition step size s_{max}

This experiment investigated in the effect of the parameter of the maximum transition step size on the sampling process, in order to estimate a good value for this parameter. For this purpose, the used experimental setup is adapted from experiment B.3, for which experiment a 'bad' parameter choice was suspected.

The pedigree from A. Almudevar (2003) [1, Fig. 1], see figure B.4, with $N = 8$ individuals is subject to be found. The same statistical assumptions are made, i.e. a uniform allele distribution, 8 different allele types, and a mutation rate of $\mu = 0.01$, and the same genetic dataset as in experiment was used (see B.1), which included genotype data for 10 loci. Genotype data was completely observed.

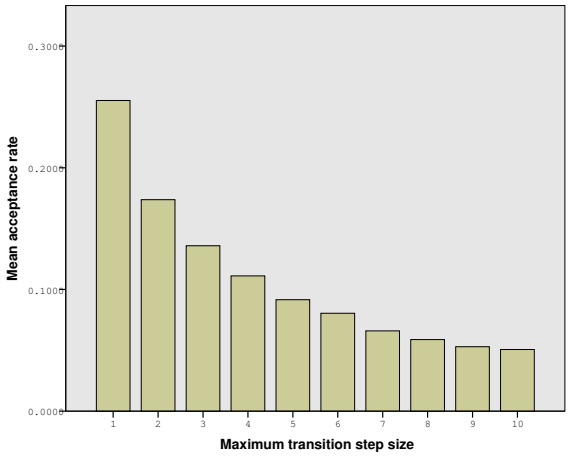
The enumeration confirmed that the original pedigree is also the maximum pedigree (cf. B.3).

Search variant 1 was used, which is based on the Metropolis-Hastings algorithm. The search was performed with an acceptance factor of $f = 50$, and stopped after the highest likelihood did not increase for 10000 search steps, or if the original pedigree was found. To increase significance the experiment was performed for 10 trials.

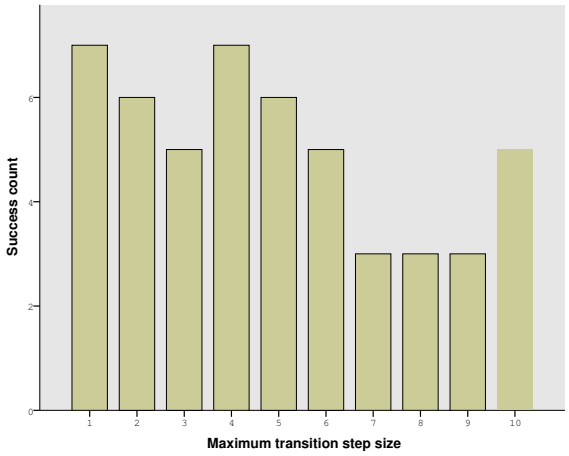
For the maximum transition step size s_{max} different values were used in the range from 1 to 10, so that transition step size s during sampling was drawn at random from the integer values in the interval $[1, s_{max}]$. This created 10 different conditions for this experiment.

Results are depicted in figure B.6. Throughout all runs, the mean acceptance rate was observed to increase with smaller step sizes s_{max} (cf. figure B.6a), and might indicate a higher quality of the generated proposal pedigrees, as more of them got accepted as the new 'current pedigree'. Regarding the overall performance, the algorithm tends to perform slightly better with a lower choice of the maximum step size s_{max}. The maximum was found more often and earlier, if smaller steps were performed. This finding is supported in terms of the

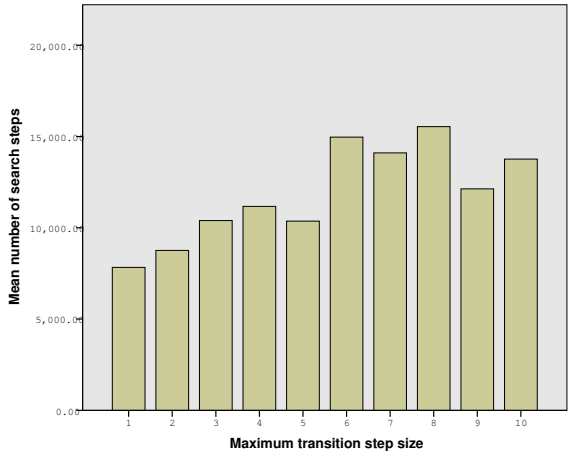
number of times the maximum likelihood pedigree was found (see figure B.6b) and in terms of the number of search steps taken to find the maximum (see figure B.6c). The computation time for this experiment was about 6.25 h.



(a) With increasing maximum transition step size s_{max} the mean acceptance rate decreased. During sampling the transition step size was chosen at random from the interval $[1, s_{max}]$



(b) With an increasing maximum transition step size s_{max}, the algorithm tends to find the maximum likelihood pedigree less often



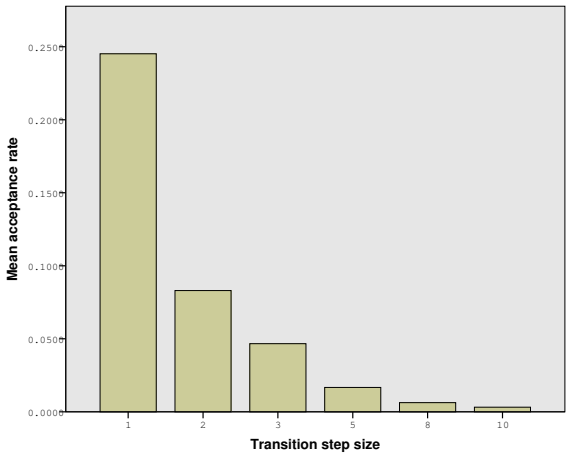
(c) With a decreasing maximum transition step size s_{max}, the algorithm tended to find the maximum earlier, i.e. with less search steps. This graph only considers those trials which found the original pedigree.

Figure B.6: Effect of the maximum transition step size s_{max}. During sampling the transition step size was chosen at random from the interval $[1, s_{max}]$.

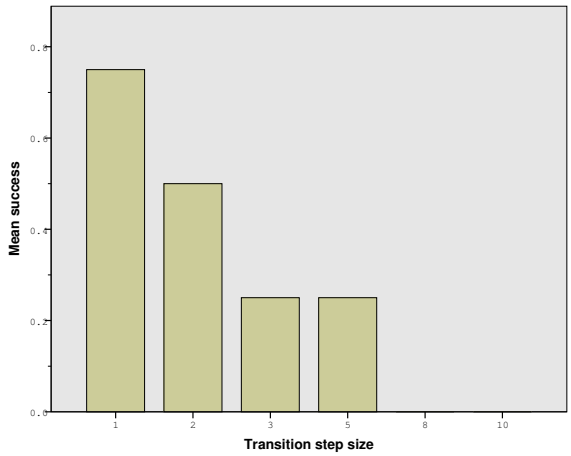
B.7 Investigating the effect of the transition step size s

This experiment investigated in the effect of the transition step size s on the performance of the search algorithm. In the proposal generation, the transition step size s is chosen randomly from an interval $[1, s_{max}]$. In this experiment the random component was removed, an all proposals were generated using the same fixed number of transitions s. For that the same experimental setup as in experiment B.6 was used, with exception of the parameter for the transition step size s which was used instead of the maximum transition step size s_{max}, and a reduced number of trials per condition.

For the transition step size s difference parameter values were used, $s \in [1, 2, 3, 5, 8, 10]$, creating six different conditions. The search was performed for in total 20 trials, four trials in each condition. Results are depicted in figure B.7. Similar as in experiment B.6, the reconstruction performance was higher with smaller step sizes s, in terms of that the maximum likelihood pedigree was found. Controlling the step size s directly, instead of indirectly through s_{max}, the importance of the effect of the transition steps size during the proposal generation was exposed.



(a) With increasing transition step size s the mean acceptance rate decreased.



(b) With an increasing transition step size s, the algorithm tends to find the maximum likelihood pedigree less often.

Figure B.7: Effect of the transition step size s on the sampling process. During sampling the transition step size was chosen at random from 1 to s. Please note that only four trials were performed for each setting of the transition step size, so that results might not be significant.

Skeleton \ Loci		1		2		3		4	
1	Servant	14	20	9	10	6	16	10	11
2	Doctor	17	17	6	10	5	7	10	11
3	Child	15	16	8	10	5	7	12	13
4	Tsar	15	16	7	10	7	7	12	12
5	Child	15	16	7	8	3	7	12	13
6	Child	15	16	8	10	3	7	12	13
7	Tsarina	15	16	8	8	3	5	12	13
8	Servant	15	17	6	9	5	7	8	10
9	Servant	16	17	6	6	6	7	11	12

Table B.2: STR genotypes for the skeletal remains in the Romanov case. Loci: (1) HUMVWA/31; (2) MUMTH01; (3) HUMF13A1; (4) HUMFES/FES.

B.8 The Romanov case: Reconstruction of a real human pedigree

This experiment applied the proposed method to a real human pedigree. The family relations of the Romanov family which are subject to be found are depicted in 4.2 on page 38, consisting of the five Romanov's, as well as the doctor and the servants which are unrelated to each other.

The STR data is available for only four loci (see table B.2) which can contemporary be considered as outdated in the choice of markers[14] as well as in the low number of loci. Still this dataset is consists of real human DNA and is well suited for illustrating the here proposed pedigree reconstruction method since this DNA dataset is small and effects of using different pedigree structures on the resulting likelihood remain comprehensible for the reader. The assumed correct pedigree (as in figure 4.2) of the found bodies can be explained without mutations.

As prior information the allele distributions were assumed to be uniform and consisting of only the occurring allele types for each locus, see table B.3, and the mutation rate was assumed to be 0.001. In contrast to this, in a similar experiment conducted by R.G. Cowell (2013) who used the same genotype data, eight (instead of five) uniformly distributed alleles were assumed.

The search was performed using variant 1, i.e. a search based on the Metropolis-Hastings algorithm, with a maximum transition step size of $s_{max} = 3$, an acceptance factor of $f = 50$, and stopped after the highest found likelihood did not further improve for $10,000$ sampling steps. In total 50 sampling trials were performed.

In all search trials, high likelihood pedigrees were found which all have a higher likelihood than the original pedigree. Six pedigrees are depicted in figure B.8, which all have the highest found likelihood and they showed a clustering of the nine individuals into three distinct groups. Similar results were found by R. Cowell (2009) who used different assumptions and a different mutation model [3].

[14] Two of the four used markers are no longer used [W. Burgers, from personal conversation, 2013].

Allele	Frequency
14	0.2
15	0.2
16	0.2
17	0.2
20	0.2

(a) HUMVWA/31

Allele	Frequency
6	0.2
7	0.2
8	0.2
9	0.2
10	0.2

(b) MUMTH01

Allele	Frequency
3	0.2
5	0.2
6	0.2
7	0.2
16	0.2

(c) HUMF13A1

Allele	Frequency
8	0.2
10	0.2
11	0.2
12	0.2
13	0.2

(d) HUMFES/FES

Table B.3: The prior allele distribution is uniform and consists of only the occurring alleles for each locus.

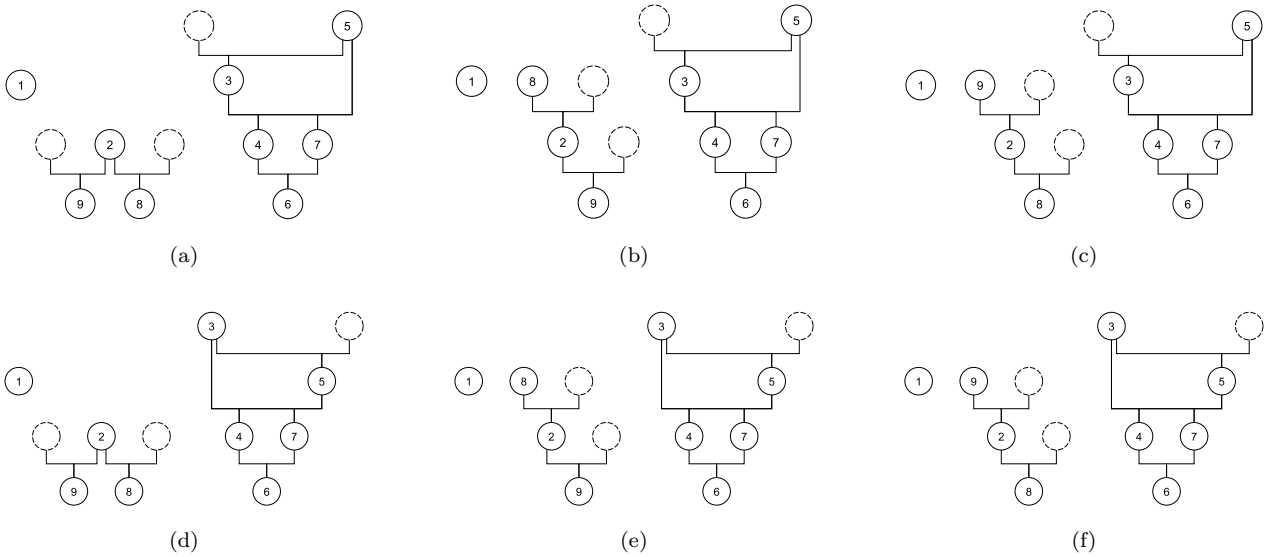


Figure B.8: Maximum found pedigrees using the Romanov data. All those pedigrees have $LH = 10^{-29.0870}$.

This implies that the original pedigree does not represent the best stochastic explanation of the family relations, and that the found pedigrees are candidates for the maximum likelihood pedigree. Reconsider that the results found by Gill et al. (1994) [8] are not subject to question, as in their research was not restricted to the use of only four loci of STR genotype data.

Analysing the found solutions, several kinship relations involving the doctor and the three servants were identified. These are likely to have caused the increase in likelihood compared to the original constellation in which servants and doctor are mutually unrelated. In some trials different family relations between the five members of the Romanov family were identified. All those different found constellations are reasonable when inspecting the given DNA dataset (cf. table B.2), as none of those used mutations in order to explain the pedigree connectivity.

B.9 The Romanov case: Reconstruction using generated samples

The reason for not finding the original pedigree of the Romanov family in the previous experiment (B.8 on the preceding page) was be suspected to be rooted in the lack of the used genotype data, which includes both, the amount of used loci as well as the choice of markers. The selected markers do not provide sufficient material in order to express the genetic differences between individuals. Aiming to resolve this, this follow-up experiment was conducted using higher quality generated genotype samples.

For the Romanov pedigree B.8, as identified by Gill et al. [8], DNA profiles were generated with each consisting of 20 genetic marker. Each allele could be one out of 20 possible allele types, which were assumed to be uniformly distributed. Mutations were assumed to occur with a rate of $\mu = 0.01$ and were generated as those.

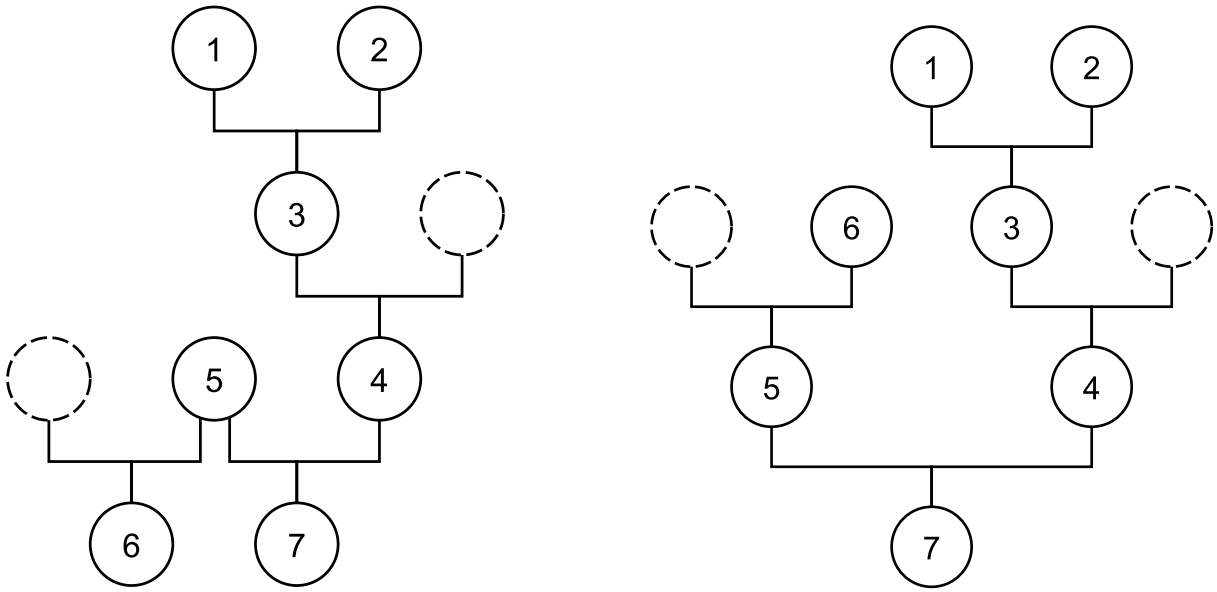
All parameters of the search algorithm were identical to those used in the previous experiment (B.8).

Using generated data, which was of higher quality for the pedigree structure of the Romanov family, the true pedigree (as in figure 4.2) could be reconstructed in most cases, in 43 out of 50 trials (86%). This result

confirms the previous claim (as in B.8) that the STR marker data of the four loci was insufficient to reconstruct the pedigree.

B.10 A constructed pedigree with seven individuals

A pedigree with $N = 7$ individuals and four different generations was manually constructed, see figure B.9a.



(a) A manually constructed pedigree structure

(b) Another pedigree as the original pedigree was found with has the same likelihood. In contrast to the original pedigree (cf. figure B.9a), individual i_6 is a parent instead of a child of i_5.

Figure B.9

A uniform allele distribution with 20 different allele types, and a mutation rate of $\mu = 0.01$ was assumed.

Genetic data was generated for 20 different loci for all individuals, with a mutation probability of $\mu = 0.01$ per allele. In a first condition, all genotype data was used as the input for the algorithm. In a second condition, genotype data was not observed for a single individual.

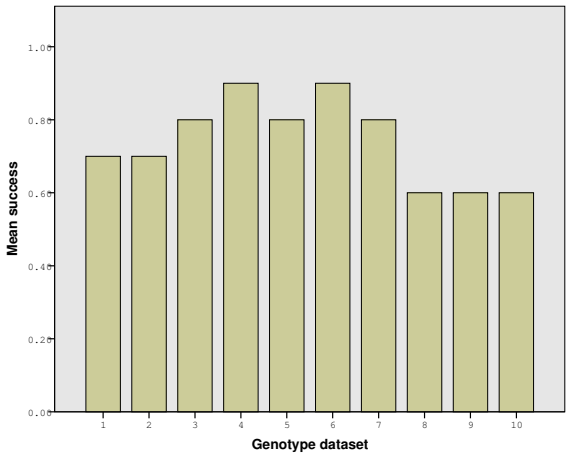
Search variant 1, i.e. the Metropolis-Hastings like search, was performed for 10 trials using a maximum transition step size of $s_{max} = 3$, and an acceptance factor of $f = 50$. Sampling stopped after no higher likelihood was found for 10000 steps.

Completely observed condition

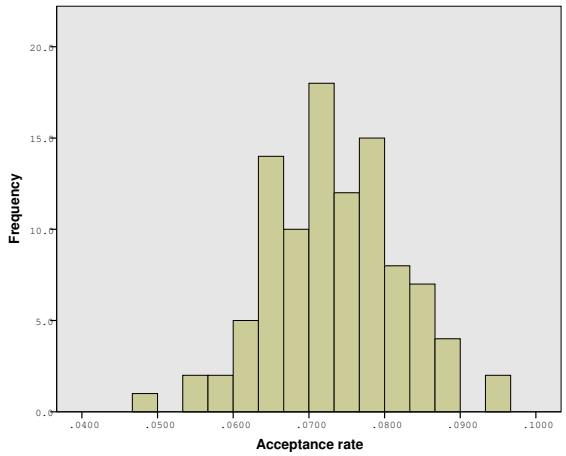
In the first condition, genetic data was completely observed for all individuals. For that 10 datasets of genetic data were generated. Results were acquired by running 10 sampling trials for each of the generated test case.

Prior to the search using an enumeration (as described in section 3.8) was performed, which confirmed that the original pedigree, as in B.9a, is also the maximum likelihood pedigree. Further, a second maximum likelihood pedigree was found which is shown in B.9b, and hence has same likelihood than the original pedigree.

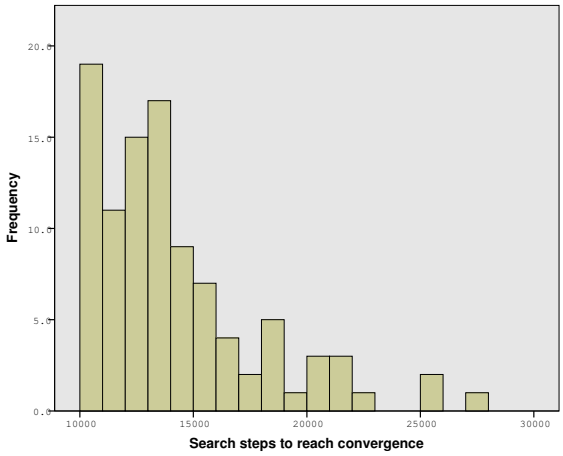
Given 10 trials, both maximum likelihood pedigrees (cf. figure B.9) were found confidently for every dataset. Further, no significant differences in the reconstruction difficulty between the generated genotype datasets were found, as the maximum likelihood pedigrees were reconstructed successfully for all datasets.



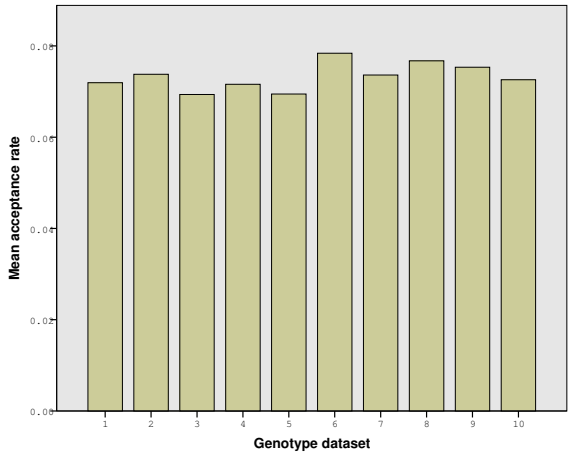
(a) For every dataset, at least 6 out of 10 trials were successful to find at least one of both the maximum pedigrees. In average, the reconstruction was successful in 74 out of 100 trials.



(b) Over all trials, a mean acceptance rate of 0.073 was observed.



(c) Most trials converged and stopped before 14,000 search steps, with a mean of 14,142.64 steps. Thus those trials which found the maximum, found it in average within 4142.64 steps, i.e. 10,000 steps earlier.



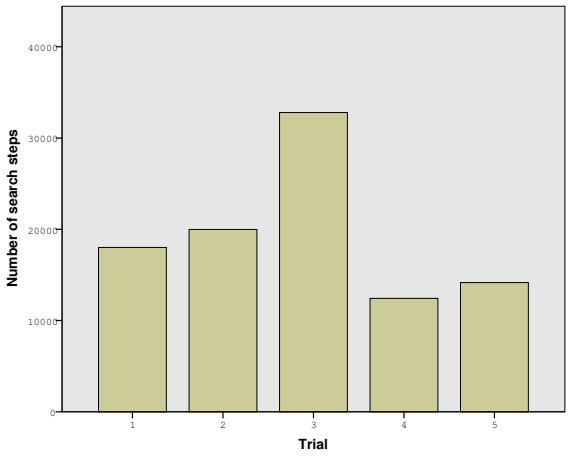
(d) The mean acceptance rate per dataset over all trials was between 0.06 and 0.08.

Figure B.10: Results for condition 1 in which samples were completely observed.

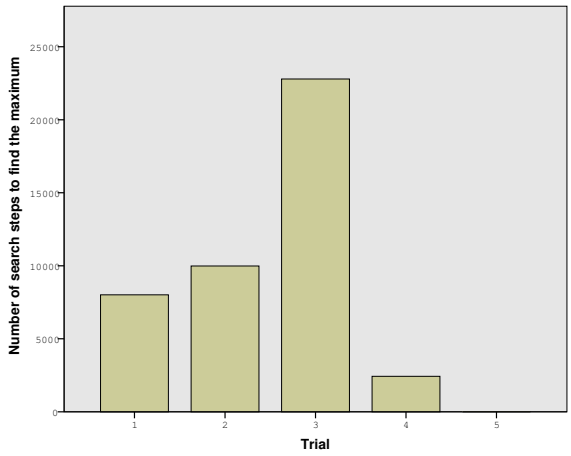
Untyped condition

In the second condition, the genetic profile data for individual i_4 is missing. This was achieved by generating a single set of genetic data for the whole pedigree and removing the allele observations from individual i_4. The generated pedigree is assumed to be the maximum likelihood pedigree with likelihood of $10^{-210.7741}$. Due to the longer computation time, only 5 search trials were performed.

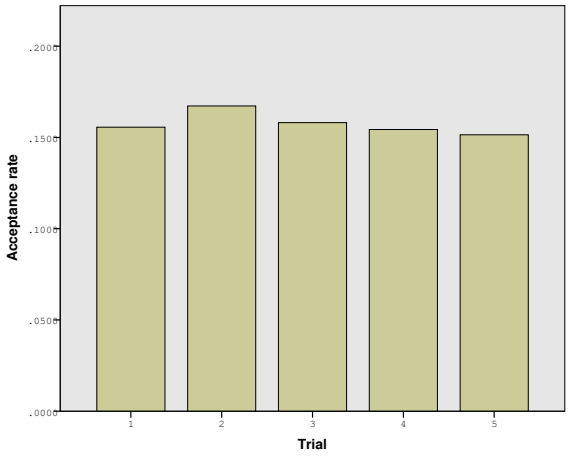
The results are depicted in figure B.11. The original pedigree structure was found in 4 out of 5 trials. The mean acceptance rate (0.157) was about twice as high as in the completely observed condition (0.073), as shown in figure B.11c. One possible explanation for this is suggested in section 4.4 on page 37.



(a) Trials converged and stopped in average after 19,476.4 steps. In trial 3 the search took considerably longer (32,792 search steps) and was successful, compared to trial 5 which was shorter (14,158 steps) and did not find the original pedigree.



(b) The number of sampling steps per trial taken to find the original pedigree. In trial 5 the maximum was not found at all.



(c) Over all trials, a mean acceptance rate of 0.157 was observed.

Figure B.11: Results for condition 2 in which samples were not completely observed.

For the condition with incomplete genotype data, the total time to perform this experiment was 20.5 hours, and about 3 to 6 hours per trial.

B.11 A pedigree with seven individuals and inbreeding

In this experiment, the previous sample pedigree was modified to also include incestuous relationships, see figure B.12, in order to investigate in the applicability of the proposed method on incestuous pedigree structures.

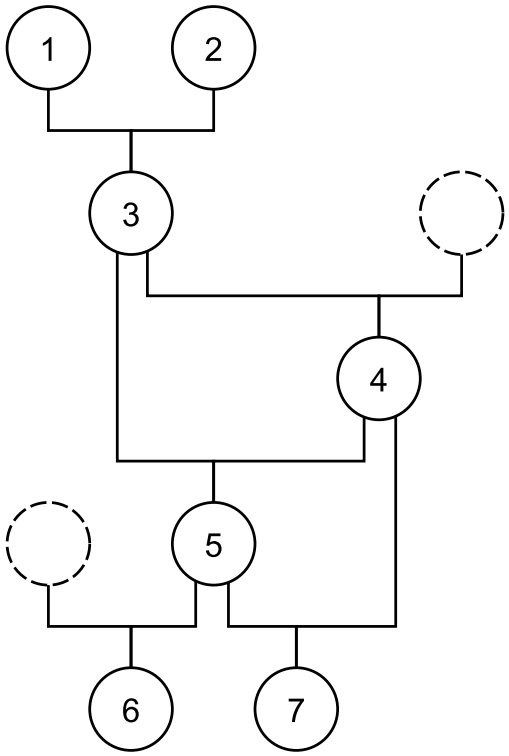


Figure B.12: A pedigree with incestuous relationships: Compared to the pedigree from experiment B.10, individual i_5 became the child of individuals i_3 and i_4.

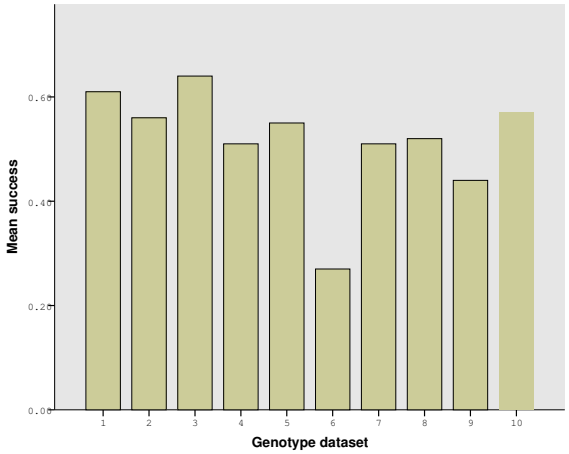
The allele distribution was assumed to be uniform with 20 different allele types and a mutation rate of $\mu = 0.01$. Genetic profiles consists of 20 loci and may include mutations which were generated according to the mutation rate μ.

The search using strategy 1a, i.e. the Metropolis-Hastings like search, was performed using a maximum transition step size of $s_{max} = 3$, and an acceptance factor of $f = 50$. The search stopped after convergence, i.e. after no higher likelihood was found for 10,000 steps. The experiment was performed in two conditions with complete and incomplete samples.

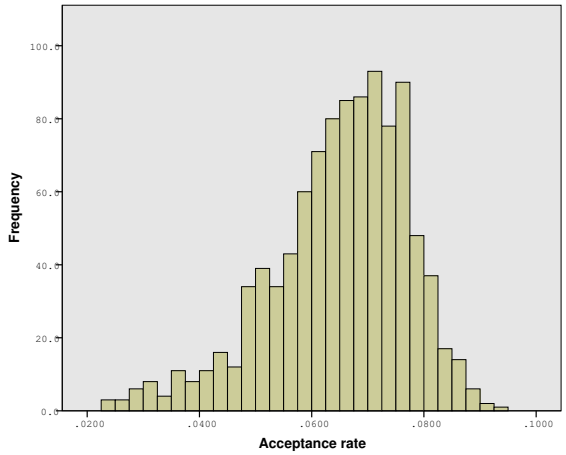
Condition 1: Complete samples

Similar to the previous pedigree, 10 datasets of genetic data were generated for this pedigree in the completely observed condition. Results were acquired by running 100 trials for each of the generated genotype datasets, and are presented in figure B.13.

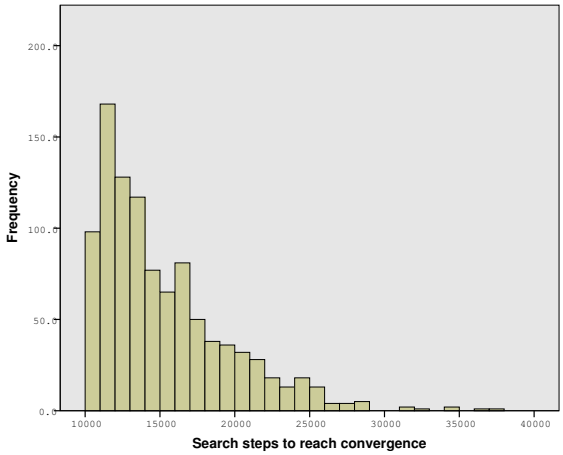
The results are presented in figure B.13. The original pedigree structure was reconstructed successfully for all sets of genotype data, and but for one of the datasets the original pedigree was only found in 27 out of 100 trials (30%), as shown in figure B.13a. This result is consistent with the expectation that inbreeding pedigrees can be more difficult to reconstruct because they have a lower genetic variance, making the correct kinship relations more difficult to find.



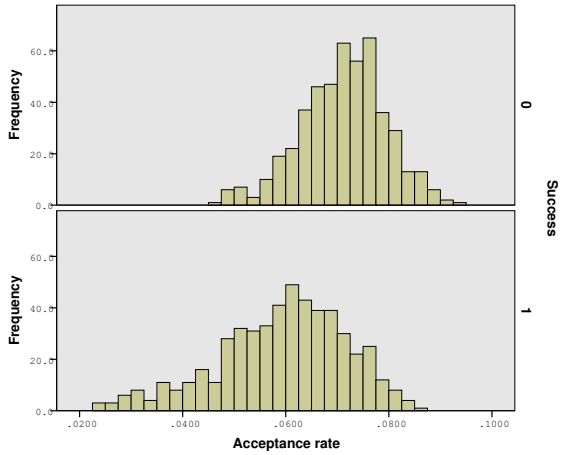
(a) The reconstruction of an incestuous pedigree was successful for all generated genotype datasets, using complete samples.



(b) Over all trials, a mean acceptance rate of 0.065 was observed.



(c) Most trials converged early with a mean of 15,209.01 steps.



(d) The mean acceptance rate for successful trials was lower than for trials which did not successfully reconstructed the pedigree.

Figure B.13: Results for the completely observed condition.

Condition 2: Incomplete samples

Results for the condition using incomplete genotype samples were acquired running 5 trials for a single generated set of genotypes in which the genetic profile data for individual i_4 was missing. The reduced number of performed tests (only 1 dataset and only 5 trials) in the untyped condition is due to the long runtime of a single sampling trial which took about 5h per trial.

In this condition, the pedigree could not be reconstructed successfully. All trials failed to find the original pedigree. In two trials (4 and 5), a different pedigree was found, which is depicted in figure B.14. Individuals i_1 and i_2 were correctly identified as founders, as well as their common child i_3. Individual i_4, for which no genetic data was available for reconstruction, was falsely identified to be younger than individual i_5. This entailed that the kinship relation between i_4 and i_5 was reversed, so that individual i_5 falsely was identified as the second parent of i_4 (besides i_3), and i_4 was not identified as the second parent of i_5 (besides i_3). However, the parents of individuals i_6 and i_7 were correctly identified.

The algorithm was close to a successful reconstruction, which requires at least an age ordering transition, which swaps i_4 and i_5 and which occurs in about 4.76% of the time. Additional parent set changes may be required as well, depending on the lower triangular part of the extended pedigree matrix $\hat{m}_a$.

Over all five trials, a mean acceptance rate of 0.109 was observed, which is an increase compared to the completely observed condition (0.065). The trials converged in average after 13929.6 search steps with a standard deviation of 3194 search steps, which is similar to the completely observed condition.

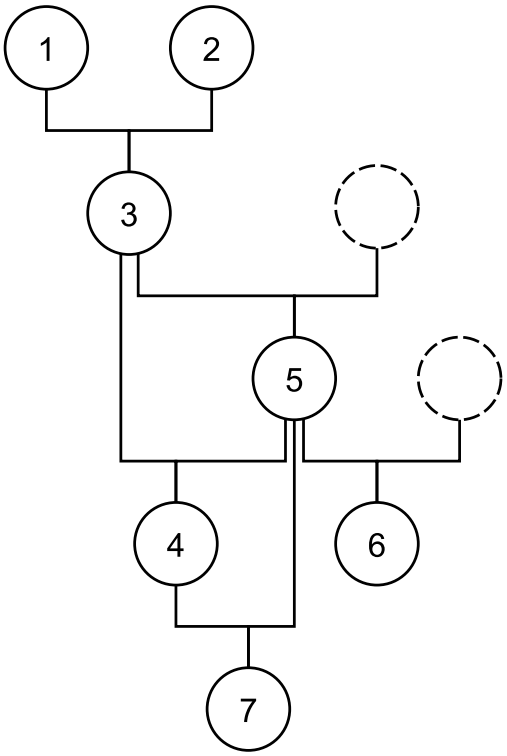


Figure B.14: Results for the completely observed condition.

B.12 Simulated annealing applied to the maximum transition step size s_{max}

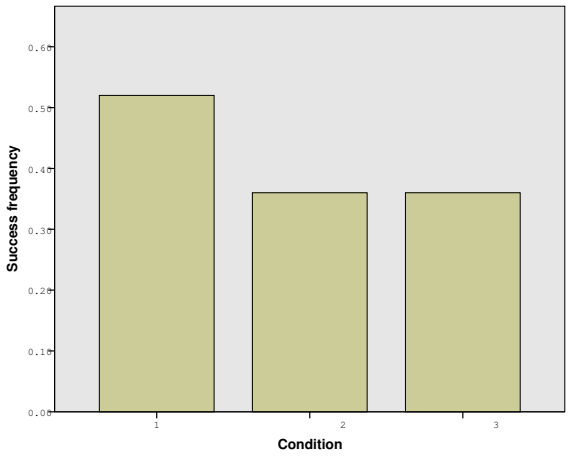
This experiment investigated if simulated annealing can improve the search performance of the proposed algorithm. For that, the pedigree adapted from A. Almudevar (2003) [1, Fig. 1], as shown in figure B.4, was subject to be found which contains eight individuals. The allele distribution was assumed to be uniformly distributed and consisting of 20 allele types, and the a mutation rate of $\mu = 0.01$ was assumed. Generated genetic data was completely observed for all individiual for 15 loci. The original pedigree was assumed to be also the maximum likelihood pedigree for the generated genotype dataset. For the search variant 1 was used, i.e. using the Metropolis-Hastings algorithm, using an acceptance factor $f = 50$, and as the stop criterion the search was performed for a fixed number of 10,000 steps.

This experiment was performed in three conditions, which employed different transition step sizes. Condition 1 is the control condition, which uses a fixed maximum transition step size of $s_{max} = 3$. In condition 2 to 4, the maximum transition step size s_{max} depends on the used annealing schedule. In particular, in condition 2, the search started with a value $s_{max} = 10$ which was lowered by 1 every 1000 steps, so that it had value $s_{max} = 1$ for the last 1000 search steps (steps 9001-10000). In condition 3, the search started with $s_{max} = 5$ which was lowered every 2000 steps. To obtain some statistical significance, the search was performed for 25 trial per condition.

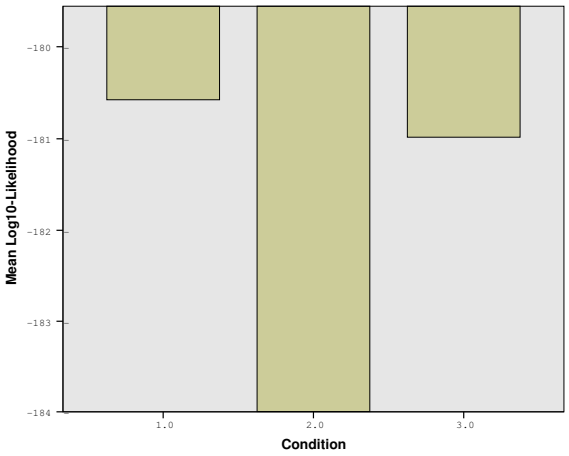
The results are presented in table B.5 and figures B.17 and B.18. Both simulated annealing conditions performed worse than the control condition. To confirm this finding, a similar experiment for simulated annealing was conducted in which the transition step size s was controlled directly.

	Successful reconstruction	Frequency
Condition 1	13 out of 25 trials	0.52
Condition 2	9 out of 25 trials	0.36
Condition 3	9 out of 25 trials	0.36

Table B.4: Results



(a) Using simulated annealing the pedigree was reconstructed less often (0.36) than not using it (0.52).



(b) Using simulated annealing the best found solution had in average a lower likelihood than not using simulated annealing.

Figure B.15: Both simulated annealing conditions (2 and 3) performed worse than the control condition in which simulated annealing was not used.

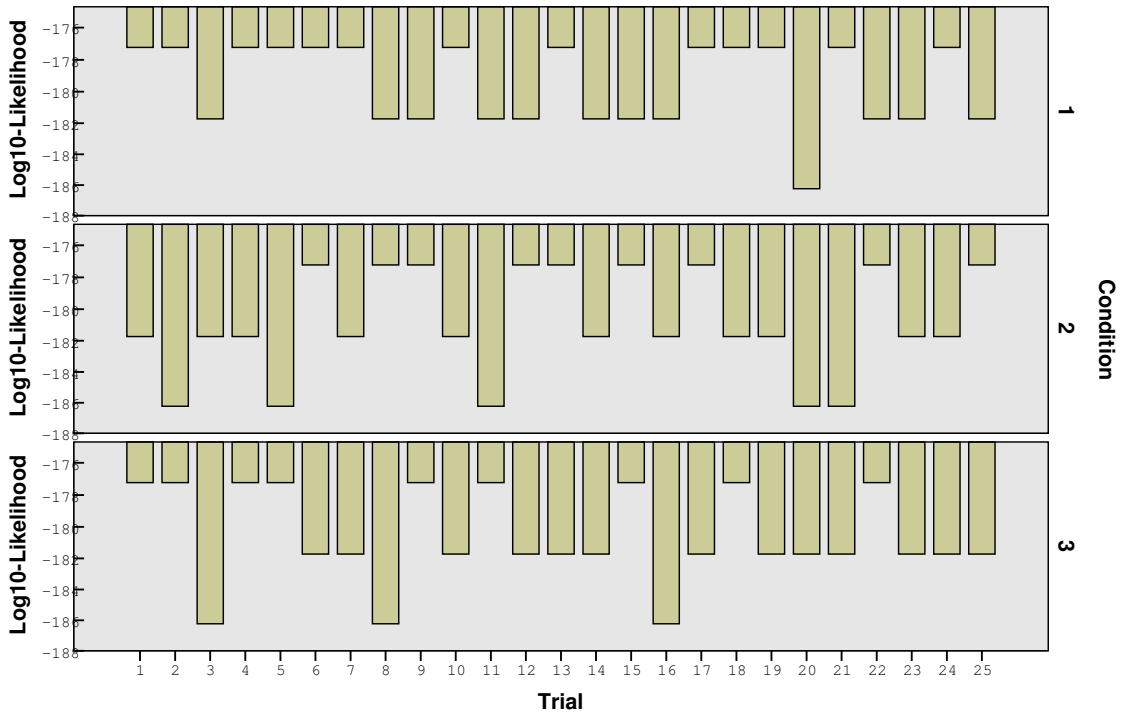


Figure B.16: All trials and the likelihood of their best found solution in comparison.

B.13 Simulated annealing applied to the transition step size s

Similar to experiment B.12, this experiment investigated in the performance of the algorithm if simulated is applied. This experiment has the same experimental setup as B.12, with exception to the variable which is controlled with the annealing schedule. Instead of the maximum transition step size s_{max}, the transition step size s is used, so that the random component which chooses s from the interval $[1, s_{max}]$ is removed and s is controlled directly.

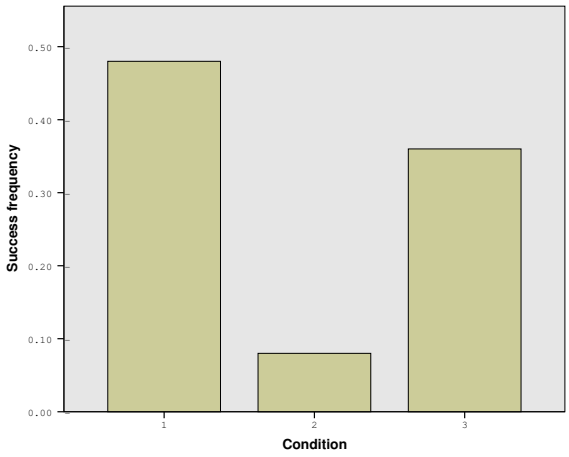
Similar than the previous experiment, this experiment was performed for the three condition, a control condition without simulated annealing, and two condition which used simulated annealing with varying annealing schedules. In particular condition 2 employed transition step sizes decreasing from $s_{max} = 10$ to

$s_{max} = 1$ every 1000 steps, whereas in condition 2 transition steps sizes were decreasing from $s_{max} = 5$ to $s_{max} = 1$ every 2000 steps. Similarly, the search was performed for 25 trial per condition.

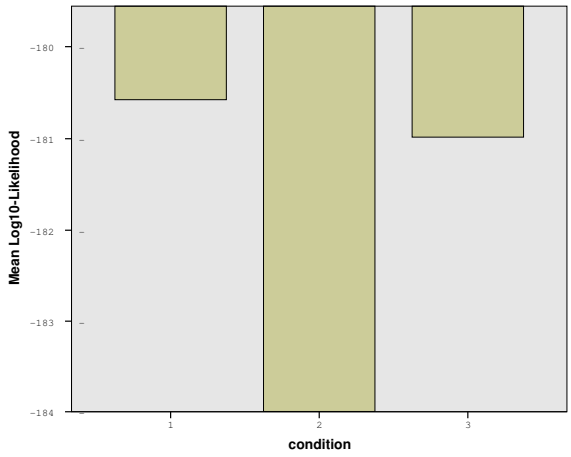
The experimental results are shown in table B.5 and figures B.17 and B.18. Similar to experiment B.12, both simulated annealing condition performed worse than the control condition. Therefore, no additional experiments using simulated annealing were conducted. However, these results might lack of significance since only 25 trials on a single genotype dataset were performed.

	Successful reconstruction	Frequency
Condition 1	12 out of 25 trials	0.48
Condition 2	2 out of 25 trials	0.08
Condition 3	9 out of 25 trials	0.36

Table B.5: Results



(a) Using simulated annealing (condition 2 and 3) the pedigree was reconstructed successfully less often than without using it (condition 1).



(b) Using simulated annealing the best found solution had in average a lower likelihood than not using simulated annealing.

Figure B.17: Both simulated annealing conditions (2 and 3) performed worse than the control condition in which simulated annealing was not used.

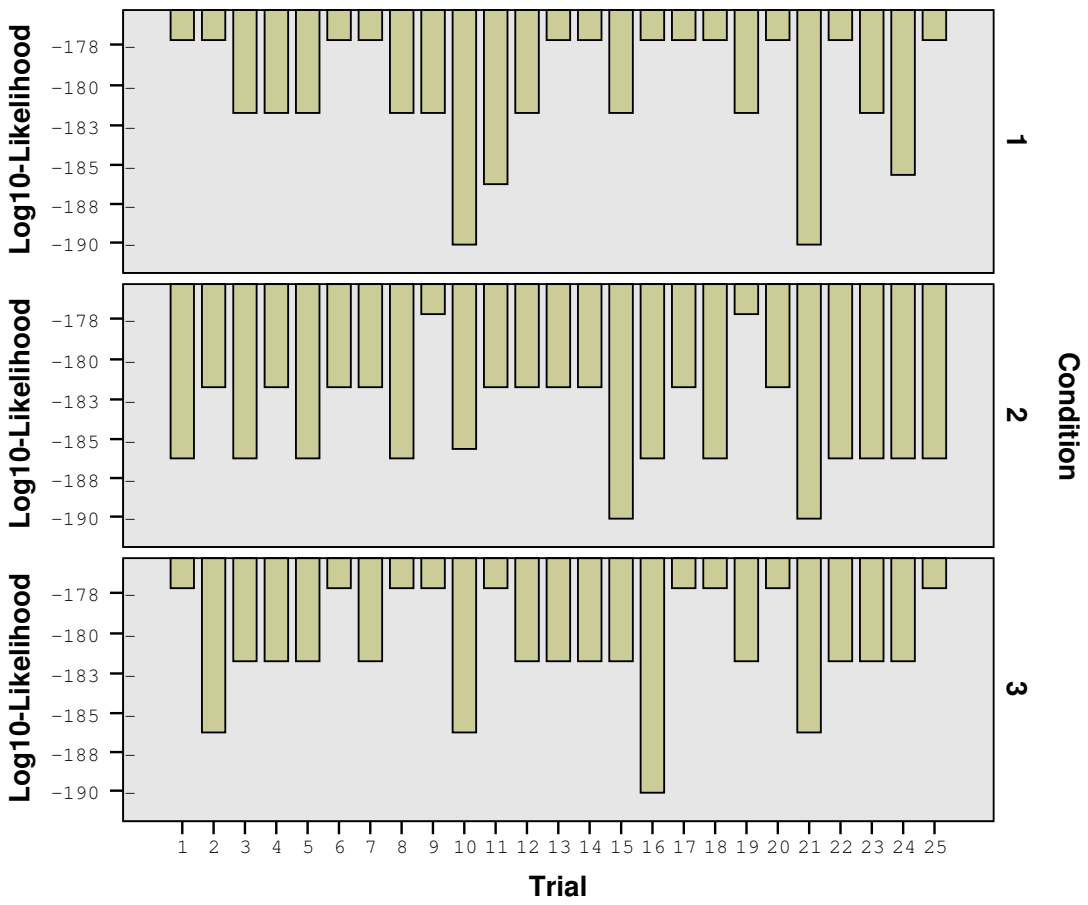


Figure B.18: All trials and the likelihoods of their best found solution in comparison.

B.14 Comparison of different search strategies in a small pedigree

This experiment assessed the performance of the different search variants in a small pedigree including a single untyped individual. The pedigree which is subject to be found contains $N = 5$ individuals with the structure as depicted in figure B.19.

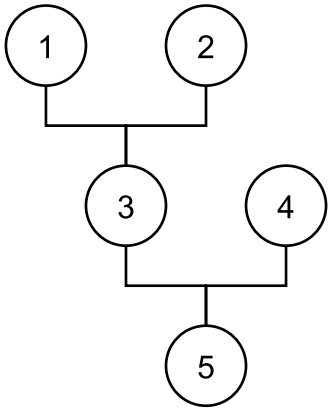


Figure B.19: : A small individual consisting of $N = 5$ individuals.

Alleles were assumed to be distributed uniformly with 20 different allele types per marker with a mutation rate of $\mu = 0.01$. A single dataset of genotype data was generated for 20 different loci, and genotype data g_3 was removed, to simulate that the genetic fingerprint of individual i_3 was not observed. The original pedigree is assumed to be also the maximum likelihood pedigree. In the generation of proposals, the number of transition steps was randomly chosen from the interval $s = [1, 3]$, and accepted using an acceptance factor of $f = 50$.

The experiment was performed with different transition strategies. In experimental conditions 1a and 1b, the random parent selection was employed, which uses random transitions to generate proposals, and in conditions 2a and 2b, the guided parent selection was used, which uses knowledge about likely parents of every individual to increase the quality of the proposals.

In experimental conditions 1a and 2a, 'masking' was used, so that the lower triangular part was kept after each transition, whereas in experiments 1b and 2b 'dropping' was used, in which the lower triangular part (including the diagonal) of the matrix were dropped after each transition. This results in four different experimental conditions.

Transition variants	(a) masking	(b) dropping
1. random parent selection	method 1a	method 1b
2. guided parent selection	method 2a	method 2b

Table B.6: Four different search strategies were compared, each using different transitions to generate proposals

All experiments were performed for 25 trials and stopped after 1000 iterations, respectively earlier if the maximum likelihood pedigree was found earlier.

The results are depicted in table B.7 and in figure B.20. All variants of the search found the original pedigree (as in figure B.19). Search variants using random parent selection, i.e. in condition 1a and 1b, found the maximum likelihood more often (in average in 10.5 of the 25 trial) than the guided parents selection (in average in 4.5 of the 25 trials).

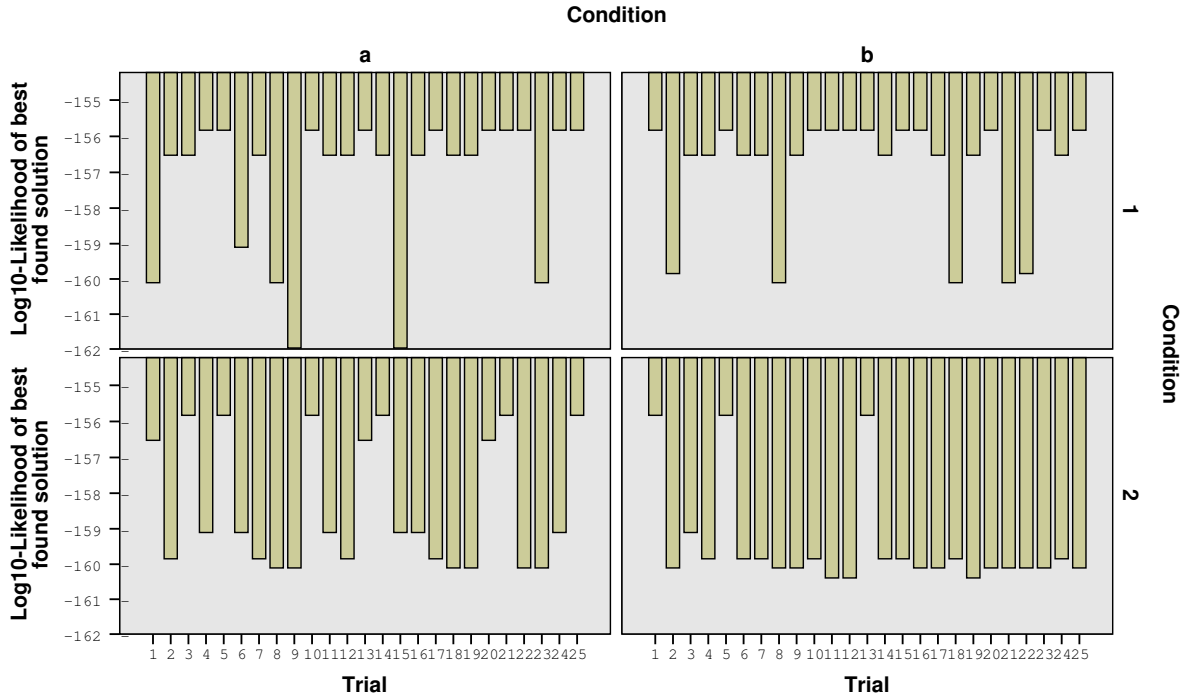


Figure B.20

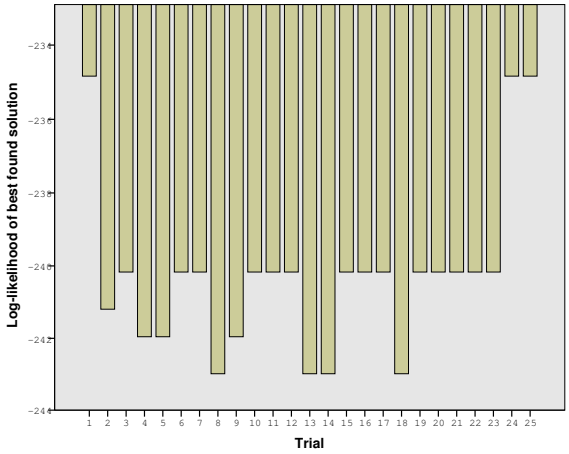
Transition	(a) keep lower triangular matrix	(b) drop lower triangular matrix
1. random selection of parents	10	11
2. guided selection of parents	6	3

Table B.7: The number of times the search found the maximum likelihood pedigree out of 25 trials

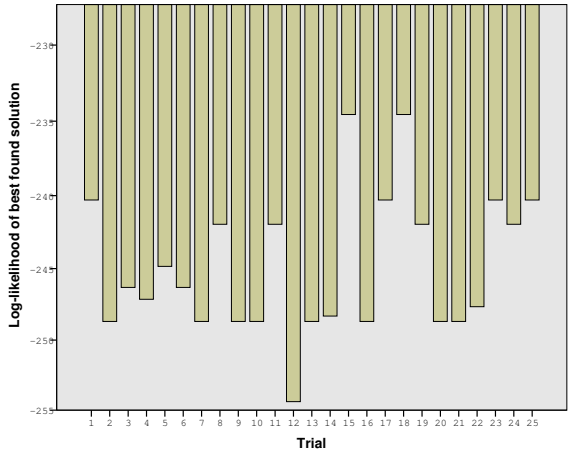
B.15 Guided search applied on an incompletely observed sample

In this experiment the pedigree as depicted in figure 1.1 was subject to be reconstructed. It contained of $N = 8$ individuals which were arranged over three generations and did not involve incestuous family relations. The

alleles were assumed to be distributed uniformly with 20 different allele types per genetic locus (in the range from 10 to 29). A mutation rate of $\mu = 0.01$ was assumed. A single set of genotype data was generated for 20 different loci, using the assumed mutation rate μ. The experiment was performed in two conditions in which not all genotypes were observed. In the first condition, the DNA profile of individual i_3 was missing (g_3) and, in the second condition, the DNA of individual i_4 was missing (g_4). The original pedigree structure (as in figure 1.1) was assumed to represent also the maximum likelihood pedigree, with maximum log-likelihood of around -234 (in particular -234.83 in condition 1, respectively -234.55 in condition 2). The search parameters were set up as follows: A guided search (variant 2a) was performed with a maximum transition step size of $s_{max} = 3$, an acceptance factor of $f = 50$, and the search halted after 1000 steps, respectively earlier, if a pedigree with the maximum likelihood was reached earlier. In total 50 trials were performed, 25 in each condition.



(a) Log-likelihood for the best found solution per trial if the genetic fingerprint of individual i_3 was untyped



(b) Log-likelihood for the best found solution per trial if the genetic fingerprint of individual i_4 was untyped

Figure B.21: Results

The results for both conditions are presented in figure B.21. The maximum likelihood pedigree was found three times in the first condition, respectively two times in the second condition. In total 5 times in 50 trials. The algorithm also found an alternative maximum likelihood pedigree in which individual i_5 was identifed as the child of i_6 instead of the parent as in the original pedigree. In the second condition only the second maximum was found because the search also stopped as soon as the maximum likelihood was found. In a better experimental setup this stop-criterion should not have been used since the assumption made, that the original pedigree would represent the only maximum likelihood pedigree, turned out to be wrong. However, it can be expected that the original pedigree would be found shorty after finding the alternative maximum likelihood pedigree. The computation of 25 trials took about 6 hours using parallel processing using 12 CPUs. In particular, 5.5 h for condition 1, and 6.5 h for condition 2.

B.16 Comparing search variants in a multi-generation pedigree with completely observed genotypes

This experiment compares the performance of the different search variants. The pedigree which is subject to be found contains $N = 8$ individuals over 7 generations with the structure as depicted in figure B.24. Alleles were assumed to be distributed uniformly with 20 different allele types per marker with a mutation rate of $\mu = 0.01$. Genotype data was generated for 20 different loci. Enumeration results confirm that the original pedigree is also the maximum likelihood pedigree. In the generation of proposals, the number of transition steps was randomly chosen from the interval $s = [1, 3]$, and accepted using an acceptance factor of $f = 50$.

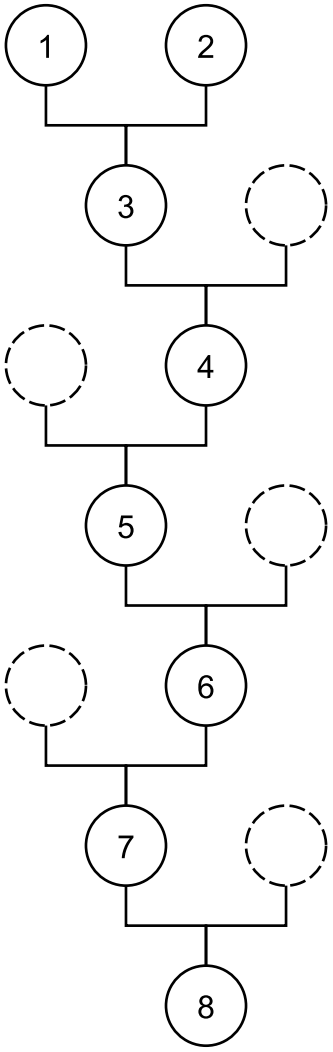


Figure B.22: A constructed pedigree containing eight individuals which are aligned over seven generations.

The experiment was performed with different transition strategies. In experimental conditions 1a and 1b, search variant 1 was used, which uses random transitions to generate proposals, and in conditions 2a and 2b, search variant 2 is used, which uses knowledge about likely parents of every individual to increase the quality of the proposals. In experimental conditions 1a and 2a, the lower triangular part was kept after each transition, whereas in experiments 1b and 2b slightly modified transitions were used, in which the lower triangular part (including the diagonal) of the matrix was dropped after each transition. This applied for both types of transitions, i.e. changes of the age-ordering and changes to the parent-sets of individuals. This results in four different experimental conditions which are summarized in table B.8.

The experiment was performed for 100 trials in each condition, resulting in 400 trials in total. Every single trials consisted of 5000 iterations. In order to compare the performance across different search strategies, the random number generator was chosen to use the same seed values (the seed was the trial number) for the different search strategies. Thus each search trial started at the same pedigree, i.e. the same position in the search space.

Transition	(a) keep lower triangular matrix	(b) drop lower triangular matrix
1. random selection of parents	method 1a	method 1b
2. guided selection of parents	method 2a	method 2b

Table B.8: Four different search strategies were compared, each using different transitions to generate proposals

The results are depicted in figure B.23 and in table B.9. The guided search approach using the extended pedigree (variant 2a) was considerably more successful in finding the maximum likelihood pedigree and found

the maximum likelihood pedigree in 28 out of 100 trials. The other search variants, 1a, 1b and 2b, found the maximum likelihood only in 2, 0 or 1 times respectively.

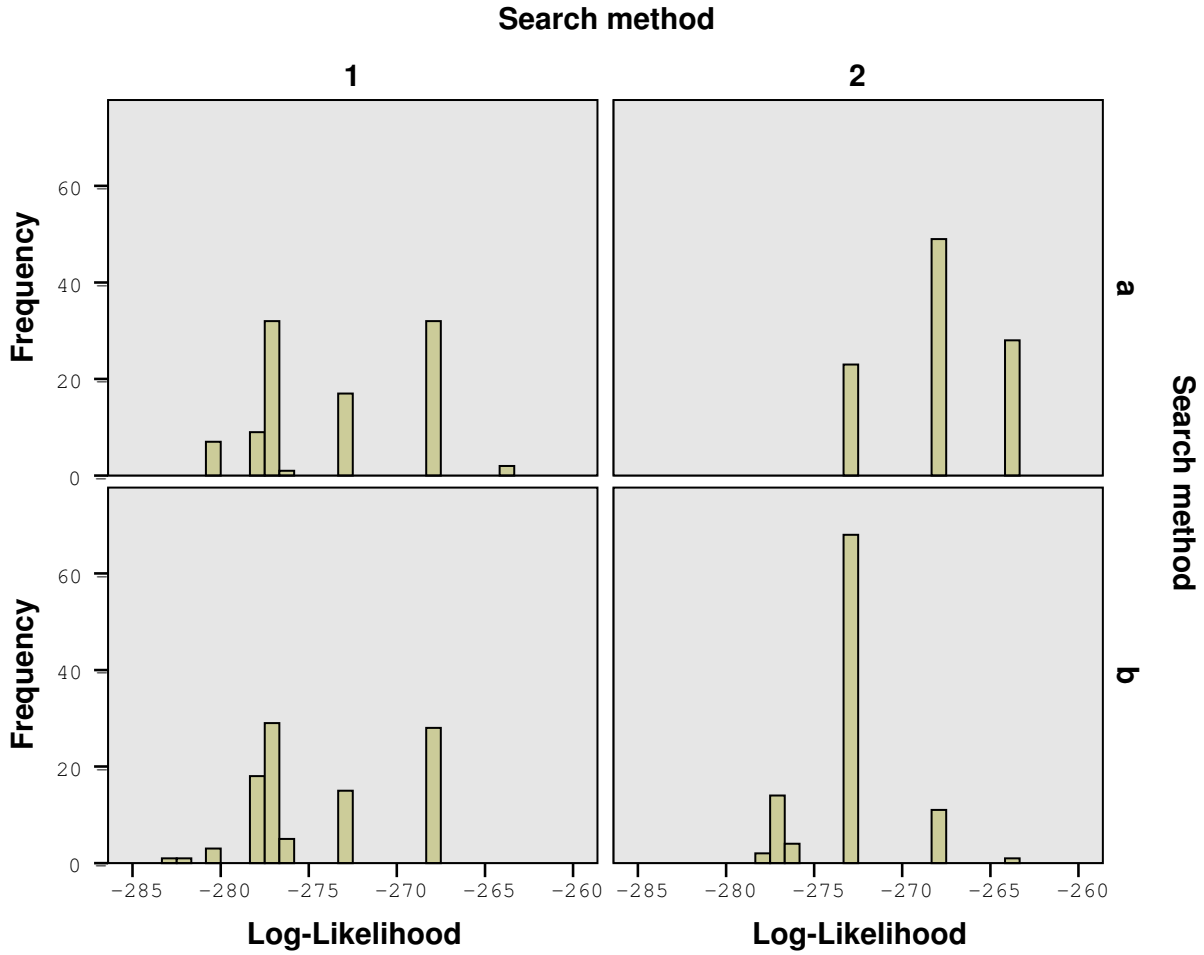


Figure B.23: The performance of the search methods in comparison. Search method 2a (guided search using the extended pedigree matrix) was most successful and found the maximum likelihood 28 times.

Transition	1. random selection of parents	2. guided selection of parents
(a) keep lower triangular matrix	2	28
(b) drop lower triangular matrix	0	1

Table B.9: The number of times the search found the maximum likelihood pedigree.

Regarding the computational effort, the computation times for each condition are presented in table B.11. The guided search was slightly slower which because it first needed to compute local likelihood values prior to the search and second drawing parent sets from the distribution was not implemented as efficiently as choosing parent sets by random.

Transition	1. random selection of parents	2. guided selection of parents
(a) keep lower triangular matrix	21 min	21 min
(b) drop lower triangular matrix	29 min	26 min

Table B.10: Total computation times in each condition to perform 100 trials

B.17 Comparing search variants in a multi-generation pedigree with incompletely observed genotypes

Similarly to experiment B.16, this experiment compares the performances in between the different proposed search variants but this time involving a single untyped individual. For that, the same experimental setup as in experiment B.16 was used with exception of the pedigree which was replaced with a smaller pedigree with has a similar connectivity pattern, as shown in B.24, for which genotype data for individual i_4 was not part of the sample. This change was made because experimental results using the larger pedigree were not able to show any significant differences between the search variants. All other details regarding the genotype generation and the statistical assumption are identical with the experiment B.16.

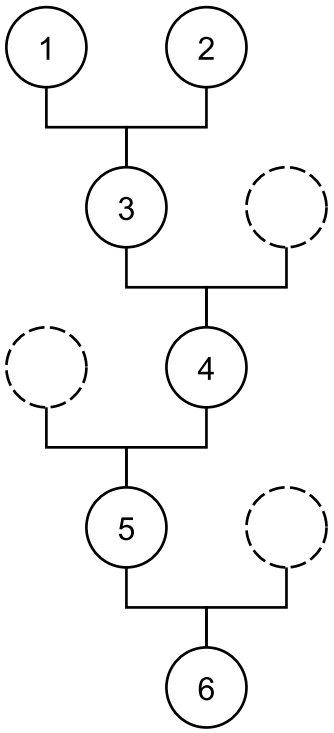


Figure B.24: A constructed pedigree containing eight individuals which are aligned over seven generations.

Similarly, the experiment was performed for the four conditions using different search variants, i.e. 1a, 1b, 2a, and 2b. The experiment was performed for 5 trials in each condition, resulting in 20 trials in total of which each consisted of 5000 search steps. Similarly to experiment B.16, the random number generator used the same seed values the five trials in each condition, in order to facilitate comparability between the conditions..

The search variant using guided transitions and the extended pedigree matrix was able to reconstruct the pedigree in 1 out of 5 trials (20%) whereas the other search strategies failed completely to reconstruct the pedigree. This result may not be significant but it shows the tendency towards that search variant 2a is superior in those multi-generation pedigrees, and thus confirms the finding from experiment B.16.

Regarding the computational effort, the computation times for each condition are presented in table B.11. In contrast to experiment B.16, results are expected to be not significant. The computationally more expensive Bayesian inference algorithm consumed most of the CPU time. Thus, differences in computation times between the search variants were outweighed by magnitudes and thus they remain concealed.

Transition	1. random selection of parents	2. guided selection of parents
(a) keep lower triangular matrix	3.5 h	2.2 h
(b) drop lower triangular matrix	2.8 h	1.5 h

Table B.11: Total computation times in each condition to perform 100 trials

B.18 Comparing search variants across multiple pedigrees with complete samples

An experiment was conducted to investigate in the performance of the different search variants. For that, several of the pedigrees presented earlier were subject to be reconstructed, in particular

1. a pedigree with $N = 7$ individuals, as shown in figure B.9a on page 61,
2. a modifed version of that with extra incestuous family relations, as depicted in figure B.12 on page 64,
3. a larger pedigree with $N = 12$ individuals, as depicted in figure B.25,
4. a pedigree with $N = 8$ individuals, as depicted in figure 1.1 on page 1,
5. a multi-generation pedigree with 6 individuals, as depicted in figure B.24 on the previous page,
6. the pedigree of the Romanov case [8], as depicted in figure 4.2 on page 38.

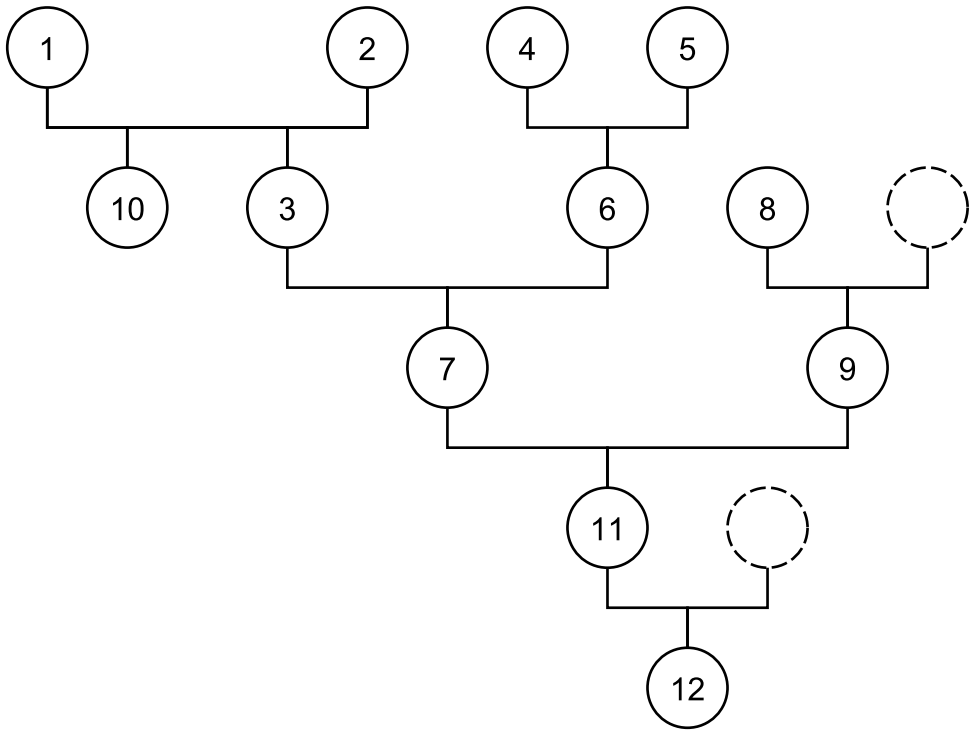
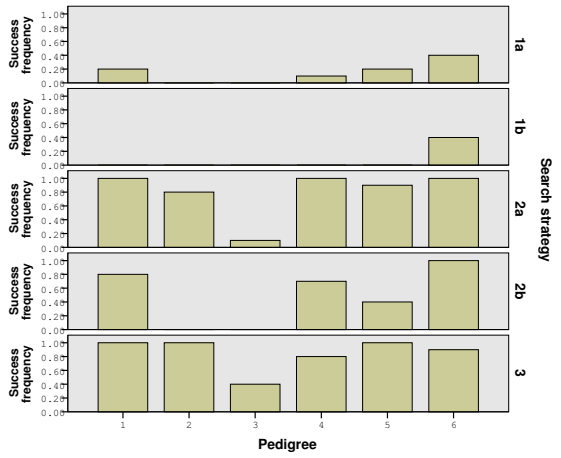


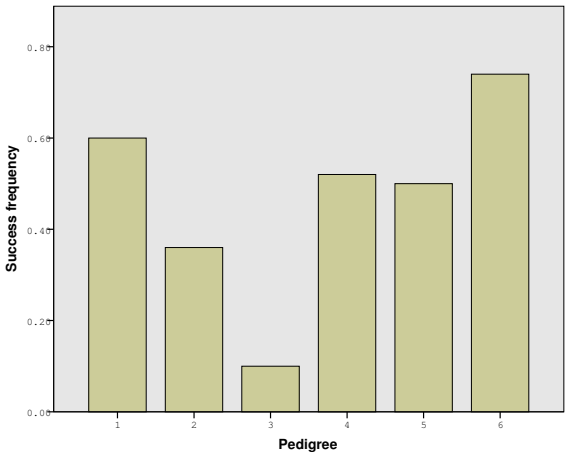
Figure B.25: A constructed pedigree consisting of $N = 12$ individuals

The allele distribution was assumed to be uniform with in total 20 different allele types per genetic locus, and the mutation rate was assumed to be $\mu = 0.01$ per allele. For all individuals of all six pedigrees, genetic profiles was generated for 20 loci. To reduce computation time for the experiment, genetic data was completely observed without missing values, so that findings might only be partially transferable to scenarios with missing genotype data. Each those pedigrees was assumed to represent also the maximum likelihood pedigree for the generated genotype data. The search was performed for ten trials for all five search strategies, (1a) search using the Metropolis-Hastings algorithm, (2a) a guided search, two search strategies (1b and 2b) which do not use the extended pedigree matrix using the 'drop' strategy, as well as (3) another search strategy in which enforces age-consistent transitions with deriving a valid age ordering automatically. Search was set to stop after no higher likelihood was found for 2000 steps. To save computation time, the search also stopped when the maximum likelihood pedigree was found.

Results are presented in figure 4.4 on page 41 and figure B.26. The search strategies (1a and 1b) which search through the space in non-guided way reconstructed the original pedigree less often than the guided search strategies (cf. figure 4.4). To resolve invalid kinship relations after every transition, 'masking' (strategies 1a and 2a) using the extended pedigree matrix was the more successful than 'dropping' (strategies (1b and 2b). Search strategy 3 showed the best accuracy over the tested pedigrees per trial, but the computation for each trial using that strategy was significantly slower (cf. B.12), because each steps is computationally more demanding.



(a)



(b)

Figure B.26

Search strategy	Computation time
1a	20 min
1b	16 min
2a	17 min
2b	15 min
3	6.25 hours

Table B.12: The time to compute 10 trials for 6 pedigrees compared across the different search strategies (without using the Bayesian network inference algorithm).