



Inverse inbreeding coefficient problems with an application to linkage analysis of recessive diseases in inbred populations

Richa Agarwala^{a, 1}, Leslie G. Biesecker^b, Alejandro A. Schäffer^{c, d, *}

^aNHGRI, NIH, National Center for Biotechnology Information, NIH, 8600 Rockville Pike, Bethesda, MD 20894, USA

^bGDRB/NHGRI, NIH, Bldg. 49, Bethesda, MD 20892, USA

^cIDRB/NHGRI, NIH, National Center for Biotechnology Information, NIH, 8600 Rockville Pike, Bethesda, MD 20894, USA

^dNational Institutes of Health, Baltimore, MD, USA

Abstract

Medical geneticists connect relatives having the same disease into a family structure called a pedigree. Genetic linkage analysis uses pedigrees to find the approximate chromosomal locations of disease-causing genes. The problem of choosing a pedigree is particularly interesting for diseases inherited in an autosomal recessive pattern in inbred populations because there are many possible paths of inheritance to choose from. A variety of shortcuts are taken to produce plausible pedigrees from inbred populations. We lay the mathematical foundations for a shortcut that was recently used in a pedigree-disease study of an inbred Mennonite population. Recessive disease genes can be localized using the shortcut of homozygosity mapping by finding regions of the genome where affected persons are homozygous. An important quantity in homozygosity mapping is the *inbreeding coefficient* of a person, which is the prior probability that the person inherited the same piece of DNA on both copies of the chromosome from a single ancestor. Software packages are ill-suited to handle large pedigrees with many inbreeding loops. Therefore, we consider the problem of generating small pedigrees that match the inbreeding coefficient of one or more affected persons in the larger pedigree. We call such a problem an *inverse inbreeding coefficient problem*. We focus on the case where there is one sibship with one or more affected persons, and consider the problem of constructing a pedigree so that it is “simpler” and gives the sibship a specified inbreeding coefficient. First, we give a construction that yields small pedigrees for any inbreeding coefficient. Second, we add the constraint that ancestor-descendant matings are not allowed, and we give another more complicated construction to match any inbreeding coefficient. Third, we show some examples of how to use the one-sibship construction to do pedigree replacement on real pedigrees with multiple affected sibships. Fourth, we give a different

* Correspondence address: National Center for Biotechnology Information, National Institutes of Health, Bldg. 38A/Room 8N-805; 8600 Rockville Pike; Bethesda, MD 20894, USA.

E-mail addresses: richa@helix.nih.gov (R. Agarwala), leslieb@helix.nih.gov (L.G. Biesecker), schaffer@helix.nih.gov (A.A. Schäffer).

¹ Contract to R.O.W. Sciences, Inc., Rockville.

construction to match the inbreeding coefficient of one sibship, while attempting to minimize a measure of the inbreeding loop complexity. © 2000 Elsevier Science B.V. All rights reserved.

Keywords: Pedigree; Linkage analysis; Inbreeding coefficient; Recessive diseases; Cycles; Graphs

1. Introduction

This paper addresses a family tree construction problem that arises in mapping genes that cause genetic diseases inherited in an autosomal recessive pattern. The mapping of disease-causing genes in the human genome is often carried out starting with the following major steps:

1. Find related persons affected with the same disease.
2. Connect the affected relatives into one or more family *pedigrees*.
3. Find the genotypes of affected and unaffected relatives at a large number of variable DNA *markers* that span the genome.
4. Use a collection of statistical and algorithmic tools called *genetic linkage analysis* to find those DNA markers that segregate with the disease [16].

Inbred populations are used in many human genetics studies partly because the inbreeding allows rare genetic diseases to appear. Existing linkage analysis software tools are not well suited to large inbred pedigrees, so users take a variety of shortcuts either in the pedigree construction or in the linkage analysis. We formulate and investigate mathematical and algorithmic questions underlying a pedigree construction shortcut taken by Chang et al. [3], in their recent location of the gene that causes glycogen storage disease type VI in an inbred Mennonite population.

Inbred populations have a high incidence of a number of diseases inherited in a recessive pattern because one “carrier” ancestor can pass a rare disease-causing mutation down several distinct paths of inheritance to affected descendants. We say that a person d is *homozygous by descent* at a gene or marker if he has two copies of an allele a of the gene or marker such that both copies were on the same chromosome copy in some ancestor p . Homozygosity mapping [18,9] is a shortcut in linkage analysis based on the idea that an affected person is likely to be homozygous by descent at the disease gene and nearby, but not throughout the genome. Homozygosity mapping starts by searching for DNA markers at which the affected persons are homozygous and then looks for the disease gene in the vicinity of those markers. Homozygosity mapping can yield false positive results when homozygosity occurs other than by descent (e.g., the same common allele reaches a person from two different ancestors). This can usually be detected because when we have a false positive, it is very likely that very close markers are heterozygous. An accurate measure of the prior probability of homozygosity by descent at a marker or region is important for successful homozygosity mapping (see [5] and the references therein).

The probability that a person is homozygous by descent at an arbitrary point (e.g. base pair or allele) in the genome is called the *inbreeding coefficient* and has been widely used in genetics for decades [24,23]. The inbreeding coefficient of a person can be defined with respect to a (small) linkage analysis pedigree or an entire (large) genealogy. For recessive diseases in inbred populations, the pedigree construction phase entails making some assumptions about which ancestors are most likely to have carried the disease mutation and what paths the mutation may have followed.

A first question is: how can one extract a pedigree of a few dozen or hundred individuals from a genealogy and preserve the likely paths of disease gene inheritance? We addressed this question with algorithms encoded in a software system called PedHunter [1]. A second question is: what can one do if the initial pedigree is too large and/or too inbred for the linkage analysis software? A common solution is to formally ignore some of the inbreeding loops (see [15] for an example). This is done because some of the existing linkage analysis software can handle very large pedigrees, if there is not too much inbreeding in a sense defined below. However, ignoring paths that reflect inbreeding can significantly change the inbreeding coefficient of the affected persons and can skew the results of homozygosity mapping. Another approach proposed by Thompson [21] is to find the most likely paths of inheritance. More generally, Kong, Lange, Thomas, Thompson, and co-workers [7,11,13,14,22,20] have proposed to compute approximate solutions to linkage analysis problems on complex pedigrees by a variety of randomization techniques. One software package that encodes such randomized methods is PANGAEA distributed by E. A. Thompson.

Chang et al. [3] proposed a different shortcut for pedigrees that have high inbreeding. They replaced the input pedigree by another pedigree in which the inbreeding coefficients of the affected sibships remained approximately the same. The replacement pedigree must include the affected sibships and their parents, and may replace the rest of the pedigree by fictitious persons and fictitious parent/child relationships. The fictitious persons typically replace real persons whose DNA is unavailable, serving only as placeholders to approximate the overall inbreeding coefficient for the affected persons. Chang et al. [3] did the replacement by a simple heuristic in which the inbreeding coefficients are not exactly preserved and the replacement pedigree size is not minimized. We present a more rigorous and exact algorithmic approach.

We address several variants of the fundamental question: given an inbreeding coefficient, W , can one construct a pedigree such that some person in it has inbreeding coefficient W ? Karigl [6] (see [21, p. 28]) provided such a construction for any rational W with denominator a power of 2. However, Karigl's construction "may involve repeated matings of many descendants to the same ancestor, and so will not be of practical relevance in human genetics [21, p. 28]". First, we extend the basic problem by adding various measures of pedigree complexity to optimize that correspond to the running time characteristics of widely used linkage analysis packages. Second, we prove that any rational W with denominator a power of 2 is achievable even when ancestor-descendant matings are not allowed and every individual is required to have two genetically distinct parents. Third, we consider a variation in which the pedigree

is allowed to have at most one person or couple at the top; this constraint is motivated by recessive diseases in which it may be desirable to assume that the disease allele entered the pedigree just once at the top. We call problems of this type, *inverse inbreeding coefficient* problems.

In the pedigree replacement problem, the input pedigree will often have multiple sibships with an affected person. We show two examples of how to use our construction for one sibship multiple times to make a combined pedigree that replaces the input pedigree and matches the inbreeding coefficient for each sibship with an affected person.

It may be useful to make the replacement pedigrees “optimal”. In this paper we seek to match the input inbreeding coefficient(s) exactly, and try to optimize other criteria. One could instead give optimization higher priority and seek only to approximate the inbreeding coefficient(s). To define useful optimality criteria it is necessary to present terms and concepts of pedigrees and existing linkage analysis software. We will formally define a graph representation for pedigrees and many related terms in the next section. For this introduction, it is sufficient that a pedigree can be represented as a graph with one vertex per person and an edge between parent and child. In this representation a *loop* is an undirected cycle. A *loop breaker* is a vertex whose removal from the graph breaks one or more cycles, with the additional constraint that the person at the bottom of the cycle cannot be used as a loop breaker where the person at the *bottom* of a cycle is the person in the cycle who does not have any descendants in the cycle. For example, when two first cousins mate and have a child, that child closes a loop, but only ancestors of the child in the loop can be used to break the loop. The number of distinct vertices (persons) whose removal breaks all loops is called the *number of loop breakers*. The loops that contribute to the inbreeding coefficient of a person p are called the *inbreeding loops* for p .

There are two classes of algorithms for linkage analysis most often used in practice, Elston–Stewart [4], and Lander–Green [10]. Elston–Stewart has the advantage that its time and space is polynomial in the number of persons in the pedigree, but the disadvantage that time and space is exponential in the number of markers and the number of loop breakers needed to break all the cycles. The Lander–Green algorithm is polynomial in the number of markers, but exponential in the number of persons. The most popular current software implementation of Elston–Stewart for looped pedigrees is called LINKAGE [12], and its improved variant called FASTLINK, particularly for looped pedigrees [17]. The most popular current implementation of Lander–Green for looped pedigrees and homozygosity mapping is MAPMAKER/HOMOZ [8], which has been included in the more comprehensive software package GENE-HUNTER.

The time/space performance of the two prevalent linkage analysis algorithms, suggests that the following optimality criteria are useful for the replacement pedigrees we construct:

- (C1) Minimum number of individuals in the pedigree.
- (C2) Minimum number of inbreeding loops needed for the pedigree.
- (C3) Minimum number of loop breakers needed for the pedigree.

Criterion (C1) is geared towards making the pedigrees small enough for the Lander–Green algorithm. Criteria (C2) and (C3) are geared to the Elston–Stewart algorithm. Criterion (C3) can differ from (C2) only when a person with multiple marriages is used to break loops which results in the number of loop breakers being less than the number of inbreeding loops [2]. Our first construction optimizes (C1) for all coefficients and optimizes (C3) if the coefficient is $< \frac{1}{2}$. Our second construction produces human pedigrees for all valid coefficients. It optimizes (C2) and provides an approximation guarantee on (C1), both if the coefficient is $\leq \frac{1}{8}$. We also present a third construction and some results on optimizing (C3). Inbreeding coefficients below $\frac{1}{8}$ are of more practical importance because that is more common in real populations. The offspring of a first cousin marriage have inbreeding coefficient $\frac{1}{8}$. One may see higher inbreeding coefficients when there are nested consanguineous marriages, but above $\frac{1}{4}$ is rare.

The rest of this paper is organized as follows. Section 2 presents definitions and basic theorems. Section 3 gives our first construction. Section 4 includes the second construction and examples with real pedigrees having multiple affected sibships. Section 5 contains our third construction. Section 6 contains a short summary, some open problems, and a discussion of another potential application.

2. Preliminary definitions and theorems

We first introduce the notion of a pedigree graph. A *pedigree graph* is a representation of a pedigree by a graph that has a vertex for each person and an edge for each child–parent pair among the persons in the pedigree. In this paper, we sometimes use undirected edges and sometimes use directed edges; when we use directed edges they are directed upward from the child vertex to the parent vertex. For example, the pedigree taken from [1] and shown in Fig. 1 can be represented as in Fig. 2. The graph traces the passage of genetic information over generations and provides a convenient mathematical representation. The individuals who do not have any of their ancestors in a pedigree are called *founders* of the pedigree.

A *path* from vertex u to vertex v in pedigree graph G is an alternating sequence of distinct vertices and edges of G , beginning with u and ending with v , such that each pair of consecutive vertices are joined by an edge. A path p connecting u to v is denoted $p : u, u_1, u_2, \dots, u_n, v$ where u_1, \dots, u_n is the sequence of other vertices on the path. We can also denote the same path as a sequence of edges $p : (u, u_1), (u_1, u_2), \dots, (u_n, v)$ or as $p : u \rightarrow u_1 \rightarrow u_2 \rightarrow \dots \rightarrow u_n \rightarrow v$. For example, $33 \rightarrow 10 \rightarrow 9 \rightarrow 8$ is a directed path in the graph of Fig. 2. A directed path p from u to v in a pedigree graph traces one way u receives genetic material from v . The vertices on p are also a (partial) list of ancestors of u . The absence of a directed path from u to v indicates that v is not an ancestor of u . Hence, there is no way that u can receive genetic material from v , e.g., in Fig. 2, person 28 does not receive genetic material directly from person 6, although they have a common ancestor. The *length* of a path is the number of edges in the path.

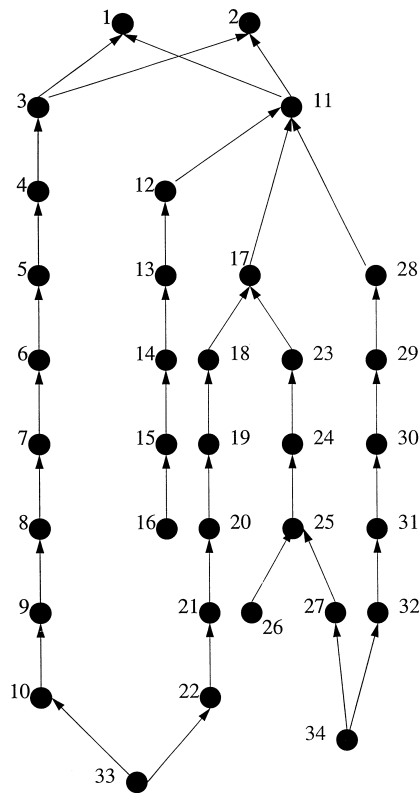


Fig. 2. Directed graph for the pedigree in the previous figure.

and

$$\text{kinship}(X, Y) = \begin{cases} 0 & \text{if } X \text{ or } Y \text{ is not known,} \\ (1 + \text{inbreeding}(X))/2 & \text{if } X = Y, \\ (\text{kinship}(X, \text{father}(Y)) + \text{kinship}(X, \text{mother}(Y)))/2 & \text{otherwise, where} \\ & \text{generation}(X) \leq \text{generation}(Y), \text{ without loss of generality.} \end{cases}$$

When computing inbreeding and kinship coefficients, we usually restrict the paths of inheritance being considered to the ones that are present in some pedigree P . When inheritance paths are ignored, the coefficients get underestimated since we ignore some paths through which the copied allele could descend. This is why extracting a (small) pedigree from a (large) genealogy typically introduces some heuristic approximation into linkage analysis of recessive diseases.

We use $\mathcal{I}(X, P)$ to denote the inbreeding coefficient of X in pedigree P and $\mathcal{K}(Y, Z, P)$ to denote the kinship coefficient of Y and Z in P . X can represent a person or a sibship since any set of full siblings have the same inbreeding coefficient.

Lemma 2.1 (Thompson [21]). *Every valid inbreeding and kinship coefficient can be written as $\sum_{i=1}^n 1/2^{L_i}$ for some n , integer $L_i > 0$, and $L_i \neq L_j$ for $i \neq j$.*

Definition 2.1. If $W = \sum_{i=1}^n 1/2^{L_i}$ for some n , integer $L_i > 0$, and $L_i \neq L_j$ for $i \neq j$, we call $1/2^{L_1}, \dots, 1/2^{L_n}$ the *factors* of W .

Definition 2.2. For a valid inbreeding coefficient W , we define $\text{BITS}(W) = \{l_1, \dots, l_J\}$ where

1. $\forall i, l_i > 1$,
2. $J = \min\{n: W = \sum_{i=1}^n 1/2^{l_i-1}, l_i > 1\}$, and
3. $W = \sum_{i=1}^J 1/2^{l_i-1}$.

For example, for $W = 21/2^5 = 1/2^1 + 1/2^3 + 1/2^5$, we get $\text{BITS}(21/2^5) = \{2, 4, 6\}$. Because of the minimality of J , for all $i \neq j, l_i \neq l_j$.

Definition 2.3. Let $\text{BITS}(W) = \{l_1, \dots, l_J\}$. We say that bit $1 \leq i \leq J$ is the *most [least] significant bit* if $\forall j \neq i, l_j > l_i [l_j < l_i]$ and the *value* of the most [least] significant bit is l_i . For example, in $\text{BITS}(\frac{21}{2^5}) = \{2, 4, 6\}$, the most significant bit is 1, the value of the most significant bit is 2, the least significant bit is 3, and the value of the least significant bit is 6.

Definition 2.4. A pair of edge-disjoint directed paths P_1, P_2 of the form

$$P_1 : X, \text{father}(X), \dots, Y \quad \text{and} \quad P_2 : X, \text{mother}(X), \dots, Y$$

in a pedigree P are said to form an *inbreeding loop* for X in (the undirected version) of P . We will use $\mathcal{JLS}(X, P)$ to denote *all* inbreeding loops of X in pedigree P .

An undirected cycle in a pedigree graph need not be an inbreeding loop, for example two brothers who marry two sisters and have children create a cycle that is not an inbreeding loop. On the other hand, an inbreeding loop can be the union of several edge-disjoint but not vertex-disjoint cycles, but it always contains an undirected cycle.

Lemma 2.2. *Let L_i be the length of loop C_i .*

$$\mathcal{JLS}(X, P) = \{C_1, \dots, C_n\} \Rightarrow \mathcal{J}(X, P) = \sum_{i=1}^n \frac{1}{2^{L_i-1}}.$$

Proof. Define $HBD(X, A)$ to mean that X is homozygous by descent due to an allele inherited twice from A , but not inherited twice (as a copy) from a descendant of A . Then

$$\mathcal{J}(X, P) = \sum_{A \in V(P)} \text{Prob}(HBD(X, A)).$$

Each contribution to the right-hand side is due to a pair of edge-disjoint paths from X to A . If the total length of the paths is L_i , then the contribution to the right-hand side is $1/2^{L_i-1}$. \square

Definition 2.5. A *loop breaker* b is a vertex whose removal breaks an undirected cycle in which b has a child. The number of *loop breakers* in a pedigree is the minimum number of vertices needed to break all cycles (whether they are inbreeding loops or not).

The problem of finding the minimum number of loop breakers in a graph arises in the study of Bayesian networks and has been proven NP-complete [19]. However, in some cases we can identify a minimum-size loop breaker set.

Lemma 2.3. If X has inbreeding coefficient W , then the minimum number of inbreeding loops in any pedigree P such that $\mathcal{I}(X, P) = W$ is $|\text{BITS}(W)|$.

Proof. From Lemma 2.1 we know that every valid inbreeding coefficient W can be written as $\sum_{i=1}^n 1/2^{L_i}$ for some n , integer $L_i > 0$, and $L_i \neq L_j$ for $i \neq j$. From the definition of BITS, it follows that the minimum n that can achieve $W = \sum_{i=1}^n 1/2^{L_i}$ is $|\text{BITS}(W)|$ and from Lemma 2.2 it follows that an inbreeding loop of length l contributes $1/2^{l-1}$ towards the inbreeding coefficient. Therefore, the minimum number of inbreeding loops required in the pedigree P such that it has an individual X with $\mathcal{I}(X, P) = W$ is $|\text{BITS}(W)|$. \square

Lemma 2.4. Let X be a sibship with inbreeding coefficient W . If $\text{BITS}(W) \neq \emptyset$, then the minimum number of individuals in any pedigree P such that $\mathcal{I}(X, P) = W$ is the value of the least significant bit in $\text{BITS}(W)$; otherwise, the minimum number is 1.

Proof. If $\text{BITS}(W) = \emptyset$, then $W = 0$ and a pedigree P with a single vertex X has the property that $\mathcal{I}(X, P) = W$. Otherwise, let l be the value of the least significant bit in $\text{BITS}(W)$. It follows from the definition of BITS that $1/2^{l-1}$ is a factor of W . From Lemma 2.2, it follows that the length of an inbreeding loop that contributes at most $1/2^{l-1}$ must be at least l . Hence, the number of individuals in P must be at least l . \square

Definition 2.6. A pedigree P for a given inbreeding coefficient W is called an *inbreeding pedigree* if it contains a vertex X such that $\mathcal{I}(X, P) = W$.

In some non-human populations, such as mice, there are strains that are so inbred that it is possible for the two parents of an individual to be effectively identical at the non-X chromosomes. We use identical parents in our first construction, but only for coefficients $\geq 1/2$, and this leads to the pedigree graph being a multi-graph. For the sake of clarity, we shall denote two copies of directed edge (X, Y) between child X and identical parents Y as $(X, Y)^1$ and $(X, Y)^2$. A second difference between human and non-human populations is that matings between ancestor and descendant are generally not allowed among humans. To reflect human populations, we say that a pedigree is a *human pedigree* if for every pair of parents X and Y of the same person, $X \neq Y$, and X is not an ancestor of Y .

3. Non-human pedigree construction

In this section we present algorithm **GENERAL** that constructs an inbreeding pedigree for every valid inbreeding coefficient such that the pedigree minimizes the number of individuals and minimizes the number of inbreeding loops for every inbreeding coefficient, and minimizes the number of loop breakers for every inbreeding coefficient $\leq 1/2$. However, the output pedigree is not necessarily a human pedigree.

Algorithm. **GENERAL**

Input: A valid inbreeding coefficient W

Output: An inbreeding pedigree P such that it has an individual X with $\mathcal{I}(X, P) = W$, satisfies (C1) for any W and satisfies (C3) for any $0 \leq W < 1/2$.

begin

$L \leftarrow \text{BITS}(W)$

if $L = \emptyset$ **return** $V(P) \leftarrow \{X\}$, $E(P) \leftarrow \emptyset$

$V(P) \leftarrow \{X = x_1, x_2, \dots, x_n\}$ where $n = \max\{l \mid l \in L\}$

$E(P) \leftarrow \emptyset, k \leftarrow 2$

while $k \in L$ **do begin**

$E(P) \leftarrow E(P) \cup \{(x_{k-1}, x_k)^1, (x_{k-1}, x_k)^2\}$

$L \leftarrow L - \{k\}$

$k \leftarrow k + 1$

end

if $L \neq \emptyset$ **then begin**

$E(P) \leftarrow E(P) \cup \{(x_{k-1}, x_k), (x_{k-1}, x_{k+1})\}$

for $i = k + 1$ **to** $n - 1$ **do** $E(P) \leftarrow E(P) \cup \{(x_i, x_{i+1})\}$

for each $i \in L$ **do** $E(P) \leftarrow E(P) \cup \{(x_i, x_k)\}$

end

return P

end

Fig. 3 shows an example pedigree graph constructed with **GENERAL**. Identical parents are used only if $W \geq 1/2$. The following theorems assert the correctness of **GENERAL** and its optimality properties.

Theorem 3.1. *For every valid inbreeding coefficient W , **GENERAL**(W) outputs a pedigree P that has a sibship X such that $\mathcal{I}(X, P) = W$.*

Proof. If $W = 0$, then $\text{BITS}(W) = \emptyset$ and **GENERAL**(0) produces a pedigree P with $V(P) = \{X\}$, $E(P) = \emptyset$. Hence, $\mathcal{I}(X, P) = 0 = W$. Otherwise, let $\text{BITS}(W) = \{l_1, \dots, l_n \mid l_i < l_{i+1}, 1 \leq i < n\}$. The output of **GENERAL** is a multigraph with at most two copies of each edge.

Following the strategy of the proofs of Lemmas 2.1 and 2.2, we seek to count how many inbreeding loops of each length have X at the bottom. Let K be the value of k

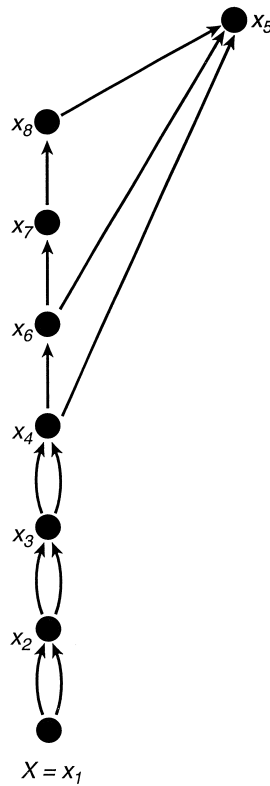


Fig. 3. Example of GENERAL where $\text{BITS}(W) = \{2, 3, 4, 6, 8\}$.

when the **while** loop finishes. We first consider the simpler case where $K = 2$ and the loop never executes. Then the only vertex with at least two incoming edges is x_2 , so it must be at the top of every inbreeding loop. The only way to get pairs of edge-disjoint paths from x_1 to x_2 is to use

$$P_1 = (X, x_2),$$

$$P_2 = (X, x_3), (X_3, x_4), \dots, (x_i, x_2).$$

There is one such pair of paths yielding an inbreeding loop of length i if and only if $i \in \text{BITS}(W)$. The correctness follows from Lemma 2.2.

If $K > 2$, define sets of loops C_1, C_2, \dots, C_n as follows. C_1 consists of a single inbreeding loop resulting from paths $P_1 = (X, x_2)^1, P_2 = (X, x_2)^2$. For $1 \leq p < K - 1$, C_{p+1} contains exactly two loops of length $2(p + 1)$ for each loop of length $2p$ in C_p . For each loop in C_p composed of the two paths

$$P_1 = (X, x_2)^a, \dots, (x_p, x_{p+1})^c,$$

$$P_2 = (X, x_2)^b, \dots, (x_p, x_{p+1})^d,$$

where $a \neq b$, $c \neq d$ and $a, b, c, d = 1$ or 2 , construct 2 pairs of longer paths (P_1^a, P_2^a) and (P_1^b, P_2^b) :

$$P_1^a = (X, x_2)^a, \dots, (x_p, x_{p+1})^c, (x_{p+1}, x_{p+2})^1,$$

$$P_2^a = (X, x_2)^b, \dots, (x_p, x_{p+1})^d, (x_{p+1}, x_{p+2})^2$$

and

$$P_1^b = (X, x_2)^a, \dots, (x_p, x_{p+1})^c, (x_{p+1}, x_{p+2})^2,$$

$$P_2^b = (X, x_2)^b, \dots, (x_p, x_{p+1})^d, (x_{p+1}, x_{p+2})^1.$$

These two pairs of longer paths make up two loops in C_{p+1} .

For $i \geq K$, let $p = l_i$. Construct C_i as follows: For each pair of paths

$$P_1 = (X, x_2)^a, \dots, (x_{K-2}, x_{K-1})^c,$$

$$P_2 = (X, x_2)^b, \dots, (x_{K-2}, x_{K-1})^d,$$

where $a \neq b$, $c \neq d$ and $a, b, c, d = 1$ or 2 , resulting in an inbreeding loop in C_{K-1} , construct 2 pairs of paths (P_1^a, P_2^a) and (P_1^b, P_2^b) resulting in 2 inbreeding loops in C_i as follows:

$$P_1^a = (X, x_2)^a, \dots, (x_{K-2}, x_{K-1})^c, (x_{K-1}, x_K),$$

$$P_2^a = (X, x_2)^b, \dots, (x_{K-2}, x_{K-1})^d, (x_{K-1}, x_{K+1}), \dots, (x_{p-1}, x_p), (x_p, x_K)$$

and

$$P_1^b = (X, x_2)^a, \dots, (x_{K-2}, x_{K-1})^c, (x_{K-1}, x_{K+1}), \dots, (x_{p-1}, x_p), (x_p, x_K),$$

$$P_2^b = (X, x_2)^b, \dots, (x_{K-2}, x_{K-1})^d, (x_{K-1}, x_K).$$

The loops in each set C_i are of the same length, and the loops in two different sets are of different lengths. One can prove by induction that for all $i < K$, C_i consists of 2^{i-1} loops of length $2i$ and for all $i \geq K$, C_i consists of 2^K loops of length $l_i + K$. Using Lemma 2.2 and definition of BITS, it suffices to show that $\mathcal{JLS}(X, P) = \bigcup_{i=1}^n C_i$. We prove set containment in each direction.

Two loops in C_i are always distinct either because they were formed from distinct smaller loops or because they were formed from the same loop by adding the last pair of edges in two different ways. It follows that $\bigcup_{i=1}^n C_i \subseteq \mathcal{JLS}(X, P)$.

Our other aim is to prove $\mathcal{JLS}(X, P) \subseteq \bigcup_{i=1}^n C_i$. Partition $\mathcal{JLS}(X, P)$ into two parts \mathcal{C}_1 and \mathcal{C}_2 such that if loop $C \in \mathcal{JLS}(X, P)$ and length of C is at most $2(K-1)$ then $C \in \mathcal{C}_1$, else $C \in \mathcal{C}_2$. Then we seek to prove that $\mathcal{C}_1 \subseteq \bigcup_{i < K} C_i$ and $\mathcal{C}_2 \subseteq \bigcup_{i \geq K} C_i$.

$\mathcal{C}_1 \subseteq \bigcup_{i < K} C_i$: Each iteration of the **while** loop introduces two copies of the same edge. One can prove by induction that the above construction of C_1, \dots, C_{K-1} accounts for all possible ways in which the two edges into x_i are the last edges on the two paths constituting the cycles. Moreover, any inbreeding loop of length $2i$, with X at the bottom, must be composed of two equal length paths from $X = x_1$ to x_{i+1} . Therefore, $\mathcal{C}_1 \subseteq \bigcup_{i < K} C_i$.

$\mathcal{C}_2 \subseteq \bigcup_{i \geq K} C_i$: Every iteration of the last **for** loop introduces one edge and creates loops of length $l_i + K$. The above construction accounts for all possible ways in which we can get a loop in which the new edge is the last edge on one of the two paths. Any loop of length $\geq 2K$ must include two edges into X_K because that is the only vertex far enough away from X with two incoming edges. Therefore, $\mathcal{C}_2 \subseteq \bigcup_{i \geq K} C_i$. \square

Theorem 3.2. *For every valid inbreeding coefficient W , $\text{GENERAL}(W)$ outputs pedigree P that satisfies (C1).*

Proof. If $\text{BITS}(W) = \emptyset$, then $V(P) = \{X\}$ else $|V(P)|$ equals the value of least significant bit of $\text{BITS}(W)$. The result follows from Lemma 2.4. \square

Theorem 3.3. *For every valid inbreeding coefficient $0 \leq W < 1/2$, $\text{GENERAL}(W)$ outputs a pedigree P that satisfies (C3).*

Proof. If $W = 0$, there are no loops in P . For $0 < W < 1/2$, the value of the most significant bit in $\text{BITS}(W)$ is at least 3. Hence, the **while** loop of $\text{GENERAL}(W)$ is never executed and all the loops can be broken by x_2 . As the minimum number of loop breakers needed to break a non-empty set of loops is 1, we get the result. \square

4. Human pedigree construction

In this section we present a construction for human pedigrees with one sibship to match any inbreeding coefficient W . The description is divided into 4 parts. The first part describes an algorithm MINLOOPS for $0 \leq W < 1/8$, and shows that it minimizes criterion (C2) and provides an approximation guarantee on the number of people (C1). The second part describes some practical examples using MINLOOPS . The third part extends the algorithm to $1/8 \leq W < 1/2$. The fourth part extends the construction to $\frac{1}{2} \leq W < 1$. This shows that any valid inbreeding coefficient can be matched with a human pedigree. We do not give any optimality guarantees for $W > 1/8$, but this range is less important for human populations.

4.1. Algorithm for $0 \leq \text{inbreeding}(X) < 1/8$

Algorithm MINLOOPS constructs an inbreeding pedigree for every valid inbreeding coefficient of at most $1/8$ such that the pedigree is a human pedigree and has minimum number of inbreeding loops.

Algorithm. MINLOOPS

Input: A valid inbreeding coefficient $W < 1/8$.

Output: A human inbreeding pedigree P such that it satisfies (C2) and has an individual X with $\mathcal{J}(X, P) = W$.

begin

$L \leftarrow \text{BITS}(W) = \{l_1, \dots, l_n\}$

Let $m^e = \max\{l \mid l \in \{0\} \cup \{l_i \mid l_i \in L, l_i \text{ is even}\}\}$,

$m^o = \max\{l \mid l \in \{0\} \cup \{l_i \mid l_i \in L, l_i \text{ is odd}\}\}$, and

$p = \max\{m^e, m^o - 1\}$

if $p \neq 0$

then $V(P) = \{X = x_1, x_2, x_3\}, E(P) = \{(x_1, x_2), (x_1, x_3)\}$

else return $V(P) = \{X\}, E(P) = \emptyset$

for $i = 2$ **to** $p/2$ **do** $V(P) \leftarrow V(P) \cup \{x_{2i}\}$

/ Create common backbone */*

for $i = 1$ **to** $p/2 - 1$ **do** $E(P) \leftarrow E(P) \cup \{(x_{2i}, x_{2(i+1)})\}$

for $i = 3$ **to** $m^e/2$ **do** $V(P) \leftarrow V(P) \cup \{x_{2i}^e\}$

if $m^e \neq 0, E(P) = E(P) \cup \{(x_3, x_6^e)\}$

/ Create even backbone */*

for $i = 3$ **to** $m^e/2 - 1$ **do** $E(P) \leftarrow E(P) \cup \{(x_{2i}^e, x_{2(i+1)}^e)\}$

for $i = 2$ **to** $(m^o - 1)/2$ **do** $V(P) \leftarrow V(P) \cup \{x_{2i+1}^o\}$

if $m^o \neq 0, E(P) = E(P) \cup \{(x_3, x_5^o)\}$

/ Create odd backbone */*

for $i = 2$ **to** $(m^o - 3)/2$ **do** $E(P) \leftarrow E(P) \cup \{(x_{2i+1}^o, x_{2i+3}^o)\}$

for $i = 1$ **to** n **do begin** */* Add loops */*

if l_i **is odd**

then $E(P) = E(P) \cup \{(x_{l_i}^o, x_{l_i-1})\}$

else $E(P) = E(P) \cup \{(x_{l_i}^e, x_{l_i})\}$

end

end

return P

end

For a non-zero W , we can partition $V(P)$ as follows:

- $V_r(P) = \{x_1, x_3\}$
- $V_b(P) = \{x_{2i} \mid 1 \leq i \leq p/2\}$
- $V_e(P) = \{x_{2i}^e \mid 3 \leq i \leq m^e/2\}$
- $V_o(P) = \{x_{2i+1}^o \mid 2 \leq i \leq (m^o - 1)/2\}$

The algorithm can be viewed as producing 3 “backbones” with no vertices in common: a *base backbone* consisting of vertices $V_b(P)$, an *odd backbone* consisting of vertices $V_o(P)$, and an *even backbone* consisting of vertices $V_e(P)$. The odd and even backbones are connected by edges (x_3, x_5^o) and (x_3, x_6^e) , which in turn is connected to the base backbone by edges (x_1, x_3) and (x_1, x_2) . For every odd bit, we add an edge from a vertex in odd backbone to a vertex in base backbone. Similarly, for every even bit, we add an edge from a vertex in even backbone to a vertex in base backbone.

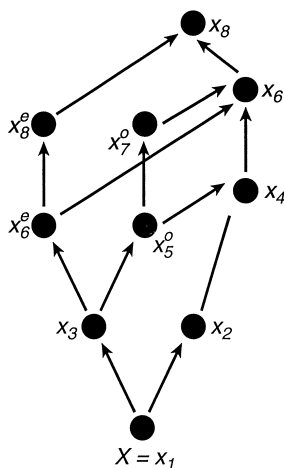


Fig. 4. Example of MINLOOPS where $\text{BITS}(W) = \{5, 6, 7, 8\}$.

Fig. 4 shows a pedigree produced by MINLOOPS. The following lemmas prove that the output is correct and satisfies some optimality guarantees.

Lemma 4.1. *For every valid inbreeding coefficient $0 \leq W < 1/8$, MINLOOPS(W) outputs a human pedigree.*

Proof. If $W = 0$, P consists of a single vertex and no edges and the result holds. If $W > 0$, to prove that the output of MINLOOPS(W) is a human pedigree P , we need to show that the outdegree of every vertex is at most 2 and if (X, Y) and (X, Z) are two edges in P , then $Y \neq Z$ and Y is not an ancestor of Z . Next, we show that the conditions hold for each set of vertices $V_r(P)$, $V_b(P)$, $V_e(P)$, and $V_o(P)$.

For vertices in $V_r(P)$, namely x_1 and x_3 , we have that the edges going out of them are always a subset of edges $\{(x_1, x_2), (x_1, x_3), (x_3, x_5^o), (x_3, x_6^e)\}$. Since $x_2 \neq x_3$, $x_5^o \neq x_6^e$, $x_2[x_5^o]$ is not an ancestor of $x_3[x_6^e]$, and vice versa, the conditions hold for vertices in $V_r(P)$.

Every vertex x_i in $V_b(P)$ has at most one outgoing edge (x_i, x_{i+2}) . Hence, the conditions hold for vertices in $V_b(P)$.

Every vertex x_i^e in $V_e(P)$ has at most one outgoing edge (x_i^e, x_{i+2}^e) to a vertex in $V_e(P)$, no outgoing edges from x_i^e to a vertex in $V_o(P)$ or $V_r(P)$, and at most one outgoing edge from x_i^e to a vertex in $V_b(P)$, namely (x_i^e, x_i) . Since $x_i \neq x_{i+2}^e$ and $x_i[x_{i+2}^e]$ is not an ancestor of $x_{i+2}^e[x_i]$, the conditions hold for vertices in $V_e(P)$.

Every vertex x_i^o in $V_o(P)$ has at most one outgoing edge (x_i^o, x_{i+2}^o) to a vertex in $V_o(P)$, no outgoing edges from x_i^o to a vertex in $V_e(P)$ or $V_r(P)$, and at most one outgoing edge from x_i^o to a vertex in $V_b(P)$, namely (x_i^o, x_{i-1}) . Since $x_{i-1} \neq x_{i+2}^o$ and $x_{i-1}[x_{i+2}^o]$ is not an ancestor of $x_{i+2}^o[x_{i-1}]$, the conditions hold for vertices in $V_o(P)$. \square

Theorem 4.1. Let $0 \leq W < 1/8$ be a valid inbreeding coefficient and let P be the pedigree output by $\text{MINLOOPS}(W)$. Then,

$$\text{BITS}(W) = \{l \mid l = \text{length}(C), C \in \mathcal{JLS}(X, P)\}.$$

Proof. We prove the set containment in each direction.

$[l \in \text{BITS}(W) \Rightarrow \exists C \in \mathcal{JLS}(X, P) \text{ such that } \text{length}(C) = l]$. Let $l \in \text{BITS}(W)$. If l is odd, then $\exists C \in \mathcal{JLS}(X, P)$ of length l comprised of the following edge-disjoint pair of paths:

$$P_1: x_1, x_2, x_4, \dots, x_{l-1}$$

and

$$P_2: x_1, x_3, x_5^0, \dots, x_l^0, x_{l-1}.$$

If l is even, then $\exists C \in \mathcal{JLS}(X, P)$ of length l comprised of the following pair of edge-disjoint paths:

$$P_1: x_1, x_2, x_4, \dots, x_l$$

and

$$P_2: x_1, x_3, x_6^e, \dots, x_l^e, x_l.$$

$[C \in \mathcal{JLS}(X, P), \text{length}(C) = l \Rightarrow l \in \text{BITS}(W)]$. By definition of an inbreeding loop, every inbreeding loop in $\mathcal{JLS}(X, P)$ must consist of an edge-disjoint pair of paths such that one path has edge (x_1, x_2) and the other path has edge (x_1, x_3) . The path containing edge (x_1, x_2) can only be comprised of x_1 followed by a portion of the base backbone because each vertex in $V_b(P)$ has at most one outgoing edge, and that edge goes to another vertex in $V_b(P)$. The path containing (x_1, x_3) can be comprised of x_1, x_3 followed by either a portion of the odd backbone or a portion of the even backbone because there is no edge between the odd and even backbones. Then the path that starts (x_1, x_3) must be followed by a vertex in base backbone to complete the loop. Since the only edges from odd/even backbone to the base backbone correspond to the bits in $\text{BITS}(W)$, we get the result. \square

Lemma 4.2. For every valid inbreeding coefficient $0 \leq W < 1/8$, $\text{MINLOOPS}(W)$ outputs a human pedigree P that has a sibship X such that $\mathcal{J}(X, P) = W$ and satisfies (C2).

Proof. Follows from Theorem 4.1 and Lemma 2.3. \square

Theorem 4.2. Let $0 \leq W < 1/8$ be the given valid inbreeding coefficient and P be the pedigree output by $\text{MINLOOPS}(W)$ with $|V(P)| = n$. If the number of individuals in any pedigree P' with a sibship with inbreeding coefficient W is l , then $n \leq 3l/2$.

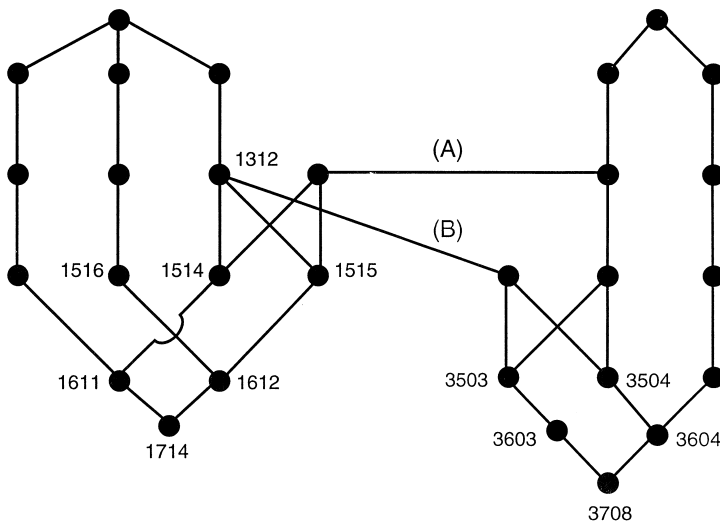


Fig. 5. Graph representation of “Israel-Kindred 3”.

Proof. Let l be the value of the least significant bit in $\text{BITS}(W)$. Then, by Lemma 2.4, any pedigree P' with a sibship with inbreeding coefficient W has at least l individuals. From construction, the number of vertices in pedigree P are $2 + p/2 + (m^e/2 - 2) + (m^o - 3)/2$. Since $m^e, m^o, p \leq l$, we get $|V(P)| < 3l/2$. \square

4.2. Application to pedigree replacement

In this subsection we illustrate by example how the previous construction can be used to do pedigree replacement in real disease pedigrees. The first example is taken from a recent paper that localized the gene for thiamine-responsive megaloblastic anemia syndrome to a region on chromosome 1 [15]. The paper indicates that the authors took the shortcut of ignoring some inbreeding loops to make their linkage analysis computations feasible. Fig. 5 shows our graph representation of one of their pedigrees, which they called “Israel-kindred 3” (kindred is another synonym of pedigree).

Israel-kindred 3 contains two affected sibships labeled as the single vertices 1714 and 3708. The edges labeled A and B are the only edges connecting the two pieces of the pedigree, and they do not participate in any inbreeding loops, so we treat this as two separate pedigrees with one sibship each. The left pedigree with sibship 1714 has inbreeding coefficient $35/512 = 1/16 + 1/256 + 1/512$, and it can be replaced by the pedigree shown in Fig. 6. In the replacement pedigree (Fig. 6), X corresponds to the affected person 1714, x_2 and x_3 correspond to the parents of 1714, namely 1611 and 1612, and the rest of the pedigree is fictitious. The right pedigree with sibship 3708 has inbreeding coefficient $129/2048 = 1/16 + 1/2048$, and it can be replaced by the pedigree in Fig. 7. As before, in the replacement pedigree (Fig. 7), X corresponds to the affected person 3708, x_2 and x_3 correspond to the parents of 3708, namely 3603

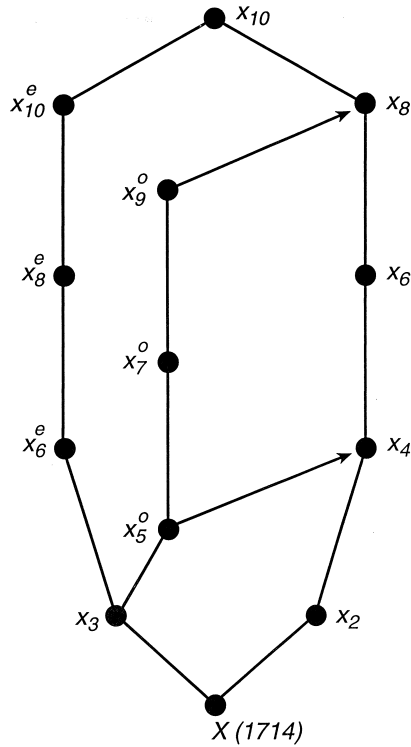


Fig. 6. Graph representation of replacement pedigree for sibship 1714 of “Israel-Kindred 3”.

and 3604, and the rest of the pedigree is fictitious. The two replacement pedigrees improve the total number of vertices from 29 to 25.

Our second example is taken from the paper by Chang et al. [3] that found the gene for glycogen storage disease type VI (GSD). In this case, the initial pedigree has 6 sibships with at least 1 affected person. The initial pedigree is shown in Fig. 9. The inbreeding coefficients for the affected sibships are $29/2^{14}(X_a)$, $1/2^{12}(X_b)$, $15/2^{14}(X_c)$, $73/2^{15}(X_d)$, $15/2^{13}(X_e)$, $11/2^{13}(X_f)$, which are all $< 1/8$.

If there is more than one sibship and each sibship has inbreeding coefficient at most $1/8$, we can adapt MINLOOPS to create one pedigree for all the sibships by creating a common base backbone and separate odd and even backbones for each sibship. When multiple sibships are affected, one still expects that the disease allele entered the pedigree once and passed down two paths from one founder to each affected person. Our construction preserves this possibility. In the case of Israel Kindred 3 our construction gives the related affected persons a common ancestor. The coalesced replacement pedigree for Israel Kindred 3, which has only 17 persons, is shown in Fig. 8. For the GSD pedigree, we can produce the replacement pedigree shown in Fig. 10 that preserves the 6 inbreeding coefficients. The input pedigree has 125 persons, while the replacement pedigree has only 70.

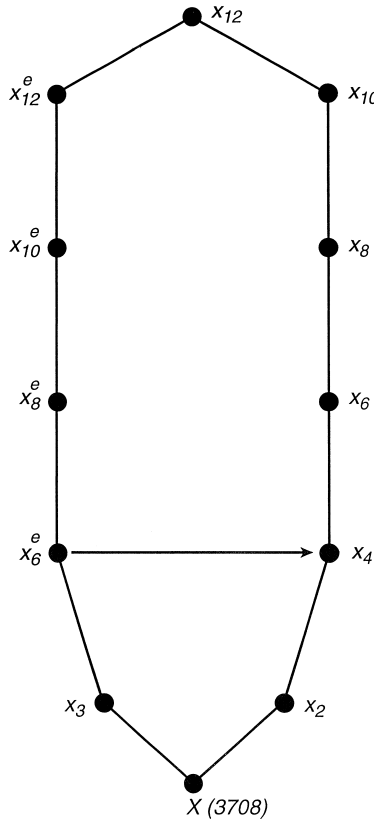


Fig. 7. Graph representation of replacement pedigree for sibship 3708 of “Israel-Kindred 3”.

In linkage analysis software it is conventional to include either both parents or zero parents for each person, rather than only those parents needed to show the possible paths of disease inheritance. This yields an inflated count of the number of persons needed in both original and replacement pedigrees. The count with spouses included sometimes gives a better performance ratio for MINLOOPS, and sometimes gives a worse performance ratio.

4.3. Algorithm for $1/8 \leq \text{inbreeding}(X) < 1/2$

In this subsection, we give an algorithm for constructing human pedigrees for an individual with inbreeding coefficient of at most $1/2$. The algorithm attempts to minimize the number of individuals but we do not claim that the pedigree produced has any of properties (C1), (C2) or (C3). Each component of the construction is described in symbols and in a figure, and the figures show that each mating we use is allowed in human pedigrees.

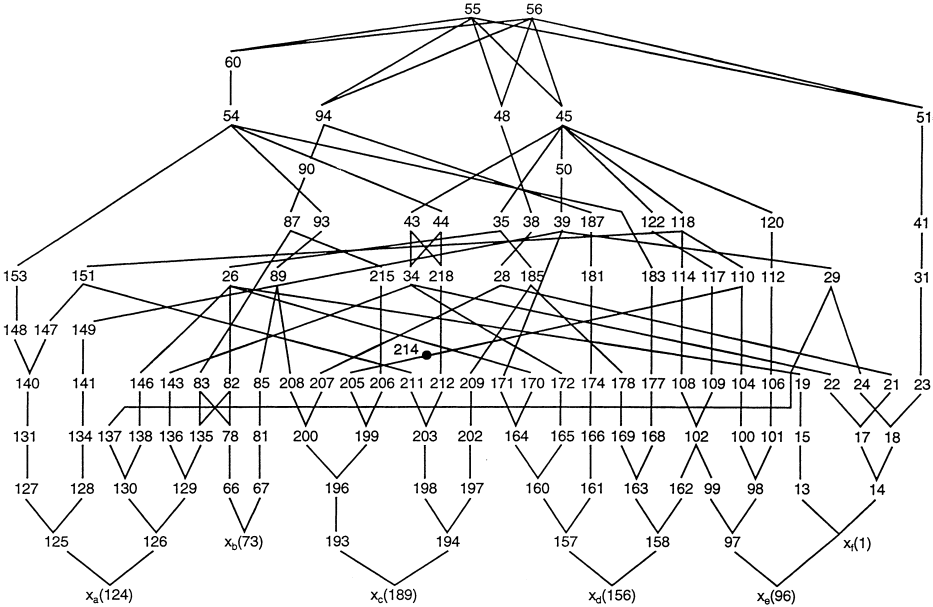


Fig. 9. Graph representation of GSD pedigree; number of persons is 125.

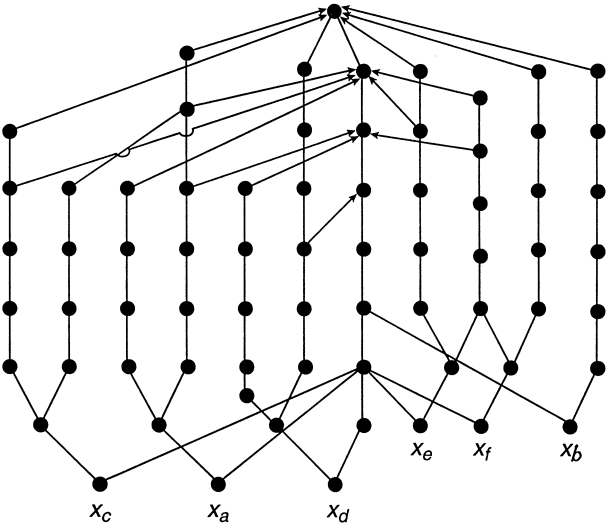


Fig. 10. Graph representation of GSD replacement pedigree; number of persons is 70.

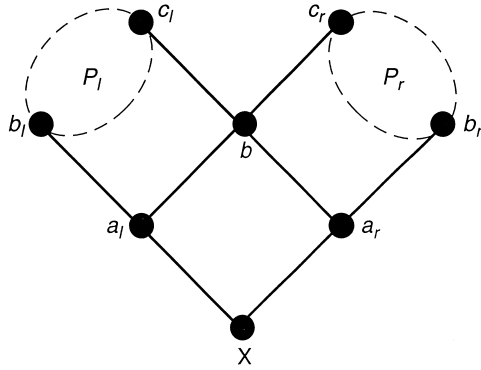


Fig. 11. GADGET1.

call returns a sibship X , for sake of clarity, we will denote a vertex $x \in V(P_l)[V(P_r)]$ as $x^l[x^r]$ where $P_l[P_r]$ distinguish the pedigrees returned by the calls. The descriptions of GADGET1, ..., GADGET8 and the proofs that they produce a pedigree P with a sibship X that has any given coefficient W in the corresponding ranges is given next.

GADGET1(W): If $W < 1/8$, **return** MINLOOPS(W). If $W = 1/8$, **return** $V(P) = \{X, a_l, a_r, b\}$, $E(P) = \{(X, a_l), (X, a_r), (a_l, b), (a_r, b)\}$. Otherwise, let $P_l, P_r = \text{MINLOOPS}(2W - 1/4)$. $V(P) = \{X, a_l, a_r, b, b_l, b_r, c_l, c_r\} \cup (V(P_l) \setminus \{X^l, x_2^l\}) \cup (V(P_r) \setminus \{X^r, x_2^r\})$, $E(P) = \{(X, a_l), (X, a_r), (a_l, b_l), (a_l, b), (a_r, b), (a_r, b_r), (b, c_l), (b, c_r)\} \cup E(P_l) \cup E(P_r)$. Collapse $x_3^l, x_4^l[x_3^r, x_4^r]$ with $b_l, c_l[b_r, c_r]$, respectively. GADGET1 is illustrated in Fig. 11.

Since MINLOOPS is called for $1/8 < \text{inbreeding}(X) < 3/16$, it is called for values in range $0 < 2W - 1/4 < 1/8$. In the pedigree produced, there are no inbreeding loops for B and no loop connecting $b_l[b_r]$ and $b_r, c_r[b_l, c_l]$. Hence, we get

$$\begin{aligned}
 \mathcal{I}(X, P) &= 1/8 + \mathcal{I}(B, P)/8 + \mathcal{K}(B, b_l, P)/4 + \mathcal{K}(B, b_r, P)/4 + \mathcal{K}(b_l, b_r, P)/4 \\
 &= 1/8 + \mathcal{K}(b_l, c_l, P)/8 + \mathcal{K}(b_r, c_r, P)/8 \\
 &= 1/8 + \mathcal{K}(b_l, c_l, P_l)/8 + \mathcal{K}(b_r, c_r, P_r)/8 \\
 &= 1/8 + 2(2W - 1/4)/8 + 2(2W - 1/4)/8 \\
 &= W.
 \end{aligned}$$

GADGET2(W): If $W \leq 3/16$, use GADGET2A(W), otherwise use GADGET2B(W). Fig. 12 shows GADGET2.

GADGET2A(W): If $W < 3/16$, **return** GADGET1(W). Otherwise, let

$$\begin{aligned}
 V(P) &= \{X, a_l, a_r, b_l, b_r, c\}, \\
 E(P) &= \{(X, a_l), (X, a_r), (a_l, b_l), (a_l, b_r), (a_r, b_l), (a_r, c), (c, b_r)\}.
 \end{aligned}$$

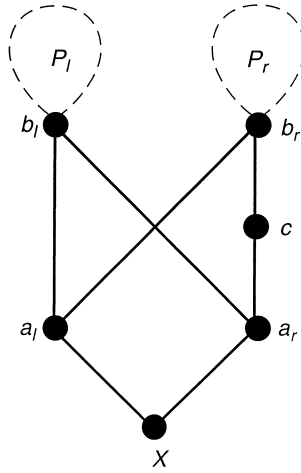


Fig. 12. GADGET2B; for GADGET2A the subpedigrees P_l and P_r are the single vertices b_l and b_r .

GADGET2B(W): If $3/16 < W \leq 27/128$, then let $P_l = \text{GADGET2A}(8W - 3/2)$ and $P_r = \text{NULL}$, else let

$$P_l = \text{GADGET2A}(3/16) \text{ and } P_r = \text{MinLoops}(16W - 27/8).$$

$$V(P) = \{X, a_l, a_r, b_l, b_r, c\} \cup V(P_l) \cup V(P_r),$$

$$E(P) = \{(X, a_l), (X, a_r), (a_l, b_l), (a_l, b_r), (a_r, b_l), (a_r, c), (c, b_r)\} \cup E(P_l) \cup E(P_r).$$

Collapse X^l with b_l and if P_r is not NULL , then collapse X^r with b_r . For $3/16 < W \leq 27/128$, we have that $0 < 8W - 3/2 \leq 3/16$ and for $27/128 < W < 7/32$, we have that $0 < 16W - 27/8 < 1/8$.

For $3/16 < W \leq 27/128$:

$$\begin{aligned} \mathcal{J}(X, P) &= 3/16 + \mathcal{J}(b_l, P)/8 + \mathcal{J}(b_r, P)/16 + 3\mathcal{K}(b_l, b_r, P)/8 \\ &= 3/16 + \mathcal{J}(b_l, P_l)/8 \\ &= 3/16 + (8W - 3/2)/8 \\ &= W. \end{aligned}$$

For $27/128 < W < 7/32$:

$$\begin{aligned} \mathcal{J}(X, P) &= 3/16 + \mathcal{J}(b_l, P)/8 + \mathcal{J}(b_r, P)/16 + 3\mathcal{K}(b_l, b_r, P)/8 \\ &= 3/16 + \mathcal{J}(b_l, P_l)/8 + \mathcal{J}(b_r, P_r)/16 \\ &= 3/16 + (3/16)/8 + (16W - 27/8)/16 \\ &= W. \end{aligned}$$

GADGET3(W): If $W \leq 7/32$, use GADGET3A(W), otherwise use GADGET3B(W). Fig. 13 shows GADGET3.

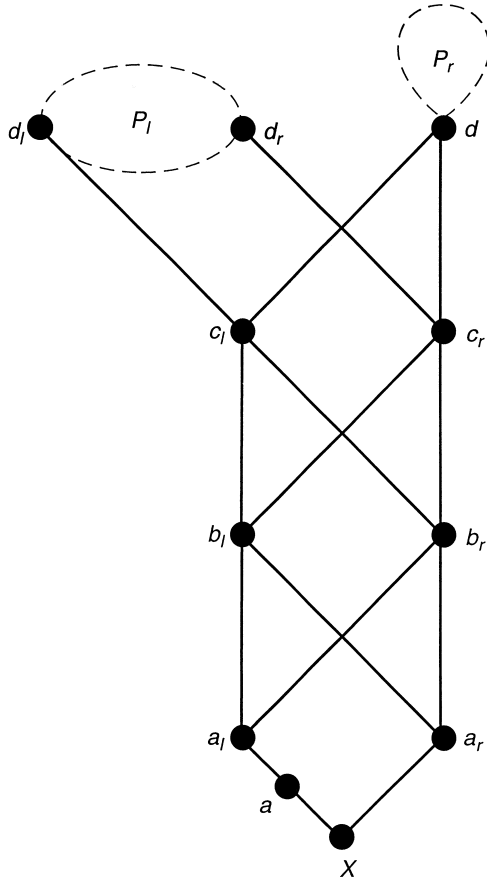


Fig. 13. GADGET3B; for GADGET3A delete P_l, P_r, d_l, d_r .

GADGET3A(W): If $W < 7/32$, **return** GADGET2(W). Otherwise,

$$V(P) = \{X, a, a_l, a_r, b_l, b_r, c_l, c_r, d\}.$$

$$E(P) = \{(X, a), (a, a_l), (X, a_r), (a_l, b_l), (a_l, b_r), (a_r, b_l), (a_r, b_r), (b_l, c_l), (b_l, c_r), (b_r, c_l), (b_r, c_r), (c_l, d), (c_r, d)\}.$$

GADGET3B(W): If $7/32 < W \leq 119/512$, then let $P_l = \text{GADGET3A}(16W - 7/2)$ and $P_r = \text{NULL}$, else let $P_l = \text{GADGET3A}(7/32)$ and $P_r = \text{MinLoops}(32W - 119/16)$.

$$V(P) = \{X, a, a_l, a_r, b_l, b_r, c_l, c_r, d, d_l, d_r\} \cup (V(P_l) \setminus \{X^l\}) \cup V(P_r),$$

$$E(P) = \{(X, A), (A, a_l), (X, a_r), (a_l, b_l), (a_l, b_r), (a_r, b_l), (a_r, b_r), (b_l, c_l), (b_l, c_r), (b_r, c_l), (b_r, c_r), (c_l, d_l), (c_l, D), (c_r, d_r), (c_r, d)\} \cup E(P_l) \cup E(P_r).$$

Collapse x_3^l, x_2^l with d_l, d_r , respectively, and if P_r is not NULL , then collapse X^r with D . For $7/32 < W \leq 119/512$, we have that $0 < 16W - 7/2 \leq 7/32$ and for $119/512 < W < 15/64$, we have that $0 < 32W - 119/16 < 1/16$.

For $7/32 < W \leq 119/512$:

$$\begin{aligned}\mathcal{J}(X, P) &= (7 + \mathcal{J}(d, P) + 3\mathcal{K}(d_l, d, P) + 2\mathcal{K}(d_l, d_r, P) + 3\mathcal{K}(d_r, d, P))/32 \\ &= 7/32 + \mathcal{K}(d_l, d_r, P_l)/16 \\ &= 7/32 + (16W - 7/2)/16 \\ &= W.\end{aligned}$$

For $119/512 < W < 15/64$:

$$\begin{aligned}\mathcal{J}(X, P) &= (7 + \mathcal{J}(d, P) + 3\mathcal{K}(d_l, d, P) + 2\mathcal{K}(d_l, d_r, P) + 3\mathcal{K}(d_r, D, P))/32 \\ &= 7/32 + \mathcal{J}(d, P_r)/32 + \mathcal{K}(d_l, d_r, P_l)/16 \\ &= 7/32 + (32W - 119/16)/32 + (7/32)/16 \\ &= W.\end{aligned}$$

GADGET4(W): If $W < 15/64$, **return** **GADGET3(W)**. Otherwise, Let $P_l, P_r = \text{MinLoops}(8W - 15/8)$.

$$\begin{aligned}V(P) &= \{X, a, a', b', c', a_l, a_r, b_l, b_r, c_l, c_r, d, d_l, d_r, e_l, e_r\} \cup (V(P_l) \setminus \{X^l, x_2^l\}) \\ &\quad \cup (V(P_r) \setminus \{X^r, x_2^r\}), \\ E(P) &= \{(X, a), (a, a_l), (X, a_r), (a_l, b_l), (a_l, b_r), (a_r, b_l), (a_r, b_r), (b_l, c_l), (b_l, c_r), \\ &\quad (b_r, c_l), (b_r, c_r), (c_l, d_l), (c_l, d), (c_r, d_r), (c_r, d), (a, a'), (a', b'), \\ &\quad (b', c'), (c', d_l), (d, e_l), (d, e_r)\} \cup E(P_l) \cup E(P_r).\end{aligned}$$

Collapse $x_3^l, x_4^l[x_3^r, x_4^r]$ with $d_l, e_l[d_r, e_r]$, respectively. Fig. 14 shows **GADGET4**. For $15/64 < W \leq 1/4$, we have that $0 < 8W - 15/8 \leq 1/8$ and

$$\begin{aligned}\mathcal{J}(X, P) &= (15 + 3\mathcal{J}(d, P) + 10\mathcal{K}(d_l, d, P) + 6\mathcal{K}(d_l, d_r, P) + 6\mathcal{K}(d, d_r))/64 \\ &= (15 + 5\mathcal{K}(d_l, e_l, P_l) + 3\mathcal{K}(d_r, e_r, P_r))/64 \\ &= (15 + 5(8W - 15/8) + 3(8W - 15/8))/64 \\ &= W.\end{aligned}$$

GADGET5(W): If $W \leq 1/4$, use **GADGET5A(W)**, otherwise use **GADGET5B(W)**. Fig. 15 shows **GADGET5**.

GADGET5A(W): If $W < 1/4$, **return** **GADGET4(W)**. Otherwise, $V(P) = \{X, a_l, a_r, b_l, b_r\}$.
 $E(P) = \{(X, a_l), (X, a_r), (a_l, b_l), (a_l, b_r), (a_r, b_l), (a_r, b_r)\}$

GADGET5B(W): Let $P_l, P_r = \text{GADGET4}(4W - 1)$.

$$\begin{aligned}V(P) &= \{X, a_l, a_r, b_l, b_r\} \cup V(P_l) \cup V(P_r), \\ E(P) &= \{(X, a_l), (X, a_r), (a_l, b_l), (a_l, b_r), (a_r, b_l), (a_r, b_r)\} \cup E(P_l) \cup E(P_r).\end{aligned}$$

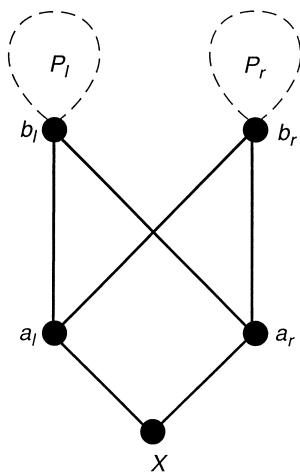


Fig. 15. GADGET5B; for GADGET5A the subpedigrees P_l and P_r are the single vertices b_l and b_r .

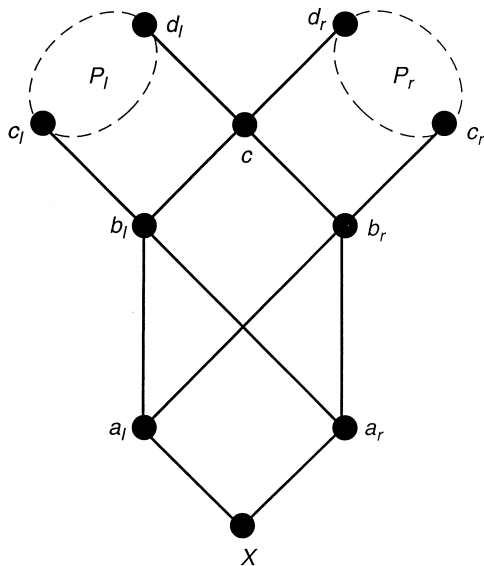


Fig. 16. GADGET6.

Collapse $x_3^l, x_2^l[x_3^r, x_2^r]$ with $c_l, d_l[c_r, d_r]$, respectively. For $5/16 \leq W < 3/8$, we have that $0 \leq 4W - 5/4 < 1/4$ and

$$\begin{aligned}
 \mathcal{J}(X, P) &= (5 + \mathcal{J}(C, P) + 2\mathcal{H}(c_l, d_l, P) + 2\mathcal{H}(c_l, d_r, P) + 2\mathcal{H}(c_r, d_l, P) \\
 &\quad + 2\mathcal{H}(c_r, d_r, P))/16 \\
 &= (5 + 2\mathcal{H}(c_l, d_l, P_l) + 2\mathcal{H}(c_r, d_r, P_r))/16 \\
 &= (5 + 2(4W - 5/4) + 2(4W - 5/4))/16 \\
 &= W.
 \end{aligned}$$

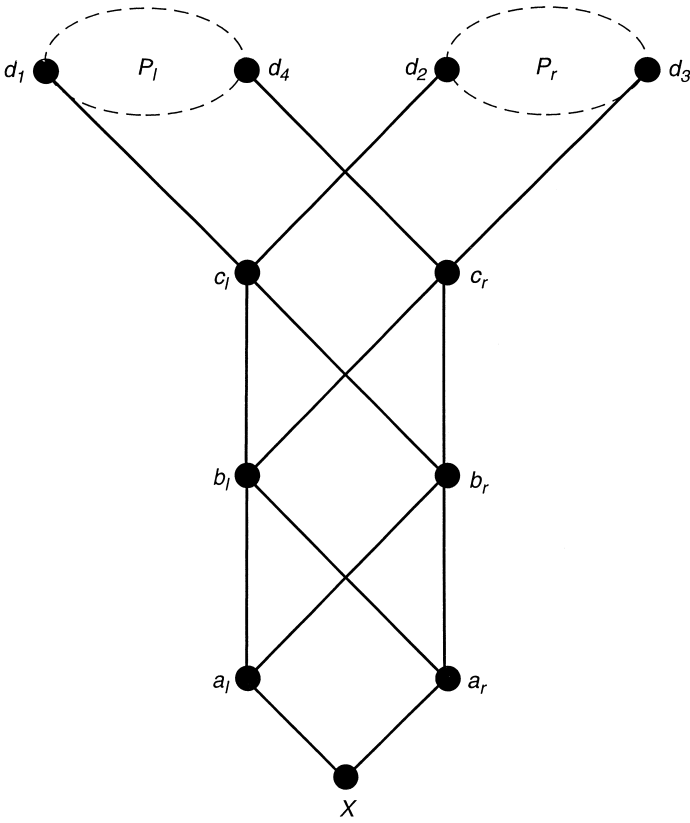


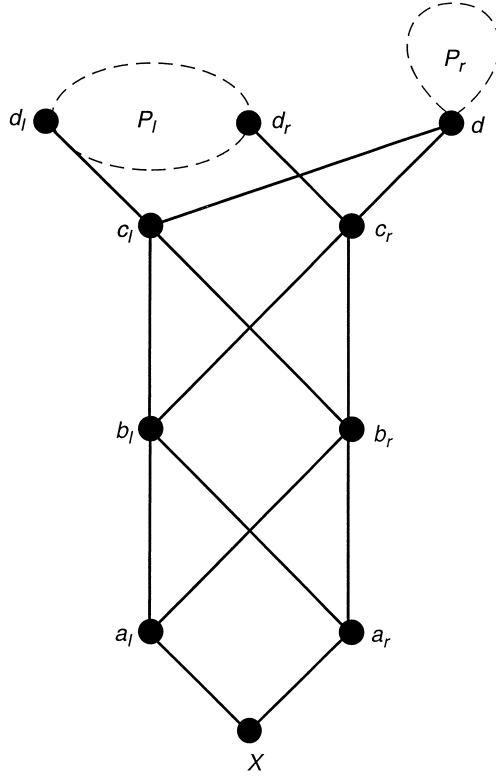
Fig. 17. GADGET7.

GADGET7(W): If $W < 3/8$, **return** **GADGET6(W)**. Otherwise, let $P_l, P_r = \text{GADGET5}(4W - 3/2)$. Fig. 17 shows **GADGET7**.

$$\begin{aligned}
 V(P) &= \{X, a_l, a_r, b_l, b_r, c_l, c_r, d_1, d_2, d_3, d_4\} \cup (V(P_l) \setminus \{X^l\}) \cup (V(P_r) \setminus \{X^r\}), \\
 E(P) &= \{(X, a_l), (X, a_r), (a_l, b_l), (a_l, b_r), (a_r, b_l), (a_r, b_r), (b_l, c_l), (b_l, c_r), (b_r, c_l), \\
 &\quad (b_r, c_r), (c_l, d_1), (c_l, d_2), (c_r, d_3), (c_r, d_4)\} \cup E(P_l) \cup E(P_r).
 \end{aligned}$$

Collapse $x_3^l, x_2^l[x_3^r, x_2^r]$ with $d_1, d_4[d_2, d_3]$, respectively. For $3/8 \leq W < 7/16$, we have that $0 \leq 4W - 3/2 < 1/4$ and

$$\begin{aligned}
 \mathcal{J}(X, P) &= (6 + \mathcal{K}(d_1, d_2, P) + \mathcal{K}(d_3, d_4, P) + 2\mathcal{K}(d_1, d_3, P) + 2\mathcal{K}(d_1, d_4, P) \\
 &\quad + 2\mathcal{K}(d_2, d_3, P) + 2\mathcal{K}(d_2, d_4, P))/16 \\
 &= (6 + 2\mathcal{K}(d_1, d_4, P_l) + 2\mathcal{K}(d_2, d_3, P_r))/16 \\
 &= (3 + (4W - 3/2) + (4W - 3/2))/8 \\
 &= W.
 \end{aligned}$$

Fig. 18. GADGET8B; for GADGET3A delete P_l, P_r, d_l, d_r .

GADGET8(W): If $W \leq 7/16$, use **GADGET8A(W)**, otherwise use **GADGET8B(W)**. Fig. 18 shows **GADGET8**.

GADGET8A(W): If $W < 7/16$, **return** **GADGET7(W)**. Otherwise,

$$V(P) = \{X, a_l, a_r, b_l, b_r, c_l, c_r, d\}.$$

$$E(P) = \{(X, a_l), (X, a_r), (a_l, b_l), (a_l, b_r), (a_r, b_l), (a_r, b_r), (b_l, c_l), (b_l, c_r), (b_r, c_l), (b_r, c_r), (c_l, d), (c_r, d)\}.$$

GADGET8B(W): If $7/16 < W \leq 63/128$, then let $P_l = \text{GADGET2A}(8W - 7/2)$ and $P_r = \text{NULL}$, else let $P_l = \text{GADGET2A}(7/16)$ and $P_r = \text{MinLoops}(16W - 63/8)$.

$$V(P) = \{X, a_l, a_r, b_l, b_r, c_l, c_r, d, d_l, d_r\} \cup (V(P_l) - \{X^l\}) \cup V(P_r),$$

$$E(P) = \{(X, a_l), (X, a_r), (a_l, b_l), (a_l, b_r), (a_r, b_l), (a_r, b_r), (b_l, c_l), (b_l, c_r), (b_r, c_l), (b_r, c_r), (c_l, d_l), (c_l, d), (c_r, d), (c_r, d_r)\} \cup E(P_l) \cup E(P_r).$$

Collapse x_3^l, x_2^l, X^r with d_l, d_r, d , respectively. For $7/16 \leq W < 63/128$, we have that $0 \leq 8W - 7/2 < 7/16$ and for $63/128 \leq W < 1/2$, we have that $0 \leq 16W - 63/8 < 1/8$

For $7/16 \leq W < 63/128$:

$$\begin{aligned}\mathcal{I}(X, P) &= (7 + \mathcal{I}(d, P) + 3\mathcal{K}(d_l, d, P) + 3\mathcal{K}(d_r, d, P) + 2\mathcal{K}(d_l, d_r, P))/16 \\ &= (7 + 2\mathcal{K}(d_l, d_r, P_l))/16 \\ &= (7 + 2(8W - 7/2))/16 \\ &= W.\end{aligned}$$

For $63/128 \leq W < 1/2$:

$$\begin{aligned}\mathcal{I}(X, P) &= (7 + \mathcal{I}(d, P) + 3\mathcal{K}(d_l, d, P) + 3\mathcal{K}(d_r, d, P) + 2\mathcal{K}(d_l, d_r, P))/16 \\ &= (7 + \mathcal{I}(d, P_r) + 2\mathcal{K}(d_l, d_r, P_l))/16 \\ &= (7 + (16W - 63/8) + 2(7/16))/16 \\ &= W.\end{aligned}$$

The above constructions and calculations prove that:

Theorem 4.3. *For every valid inbreeding coefficient W such that $1/8 < W < 1/2$, there is a human pedigree P with a sibship X , such that $I(X, P) = W$.*

4.4. Algorithm for $1/2 \leq \text{inbreeding}(X) < 1$

In this section, we give an incremental method of constructing human pedigrees with inbreeding coefficients $\geq 1/2$. The method works in two stages. In the first stage we construct a sequence of pedigrees P^1, P^2, \dots , such that for $i < j$, $I(X, P^i) < I(X, P^j)$ and $\lim_{k \rightarrow \infty} \mathcal{I}(X, P^k) \rightarrow 1$. This gives an infinite sequence of inbreeding coefficients spanning the interval $[1/2, 1)$ that are achievable. In the second stage we fill in all the gaps in the infinite sequence. Depending on the desired coefficient, we take a suitable pedigree P^i , and add one of the six specialized extender pedigrees on top of P^i . Some of the extender pedigrees are disconnected but when they are added to a P^i , the resulting pedigree is always connected.

Consider the family of pedigrees P^0, P^1, \dots defined as follows:

$$\begin{aligned}V(P^0) &= \{X, a_l^0, a_r^0, b_l^0, b_r^0, c_l^0, c_r^0, L^0, R^0\}, \\ E(P^0) &= \{(X, a_l^0), (X, a_r^0), (a_l^0, b_l^0), (a_l^0, b_r^0), (a_r^0, b_l^0), (a_r^0, b_r^0), \\ &\quad (b_l^0, c_l^0), (b_l^0, c_r^0), (b_r^0, c_l^0), (b_r^0, c_r^0), (c_l^0, L^0), (c_l^0, R^0), (c_r^0, L^0), (c_r^0, R^0)\}, \\ V(P^k) &= V(P^{k-1}) \cup \{a_1^k, a_2^k, a_3^k, a_4^k, b_l^k, b_r^k, c_l^k, c_r^k, L^k, R^k\}, \\ E(P^k) &= E(P^{k-1}) \cup \{(L^{k-1}, a_1^k), (L^{k-1}, a_2^k), (R^{k-1}, a_3^k), (R^{k-1}, a_4^k), \\ &\quad (a_1^k, b_l^k), (a_1^k, b_r^k), (a_2^k, b_l^k), (a_2^k, b_r^k), (a_3^k, b_l^k), (a_3^k, b_r^k), (a_4^k, b_l^k), (a_4^k, b_r^k), \\ &\quad (b_l^k, c_l^k), (b_l^k, c_r^k), (b_r^k, c_l^k), (b_r^k, c_r^k), (c_l^k, L^k), (c_l^k, R^k), (c_r^k, L^k), (c_r^k, R^k)\}.\end{aligned}$$

Each P^k is composed of the BASE P^0 shown in Fig. 19 which has been extended by adding k stacked copies of EXTENDER shown in Fig. 20.

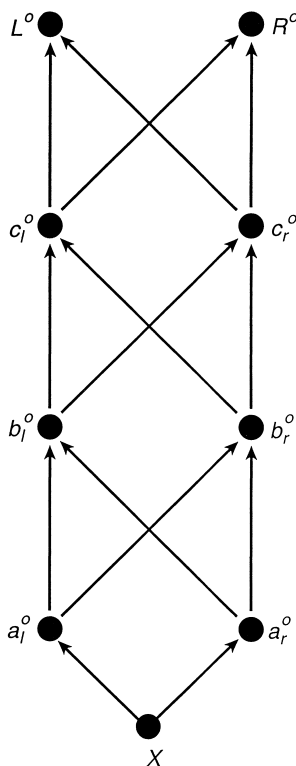


Fig. 19. BASE.

Lemma 4.3. *Let S^k be any pedigree that has P^k , as defined above, at the bottom and has an additional (possibly disconnected) pedigree Z stacked on top of L^k and R^k . We assume that Z has two vertices at the bottom and those vertices are named and coalesced with L^k , R^k . Unlike our other pedigrees, we allow Z to have two disconnected pieces, one on top of L^k and one on top of R^k . Then*

$$\mathcal{J}(X, S^k) = 1 - \frac{1}{2^{k+1}} + \frac{6\mathcal{K}(L^k, R^k, Z) + \mathcal{J}(L^k, Z) + \mathcal{J}(R^k, Z)}{2^{k+4}}.$$

When Z has just the two vertices and no edges,

$$\mathcal{J}(X, P^k) = 1 - \frac{1}{2^{k+1}}.$$

Proof. We prove the first, more general, statement by induction on k . If the specialized extender pedigree Z consists of two disconnected pieces, then the kinship term $6\mathcal{K}(L^k, R^k, Z) = 0$. If Z consists of just the two vertices, then the inbreeding terms are also 0, yielding the more specialized second formula from the first formula.

Base ($k = 0$): P^0 is as shown in Fig. 19. We consider $P^0 \cup Z$, where Z is any (arbitrary) specialized extender pedigree put on top of L^0 and R^0 . For the base case

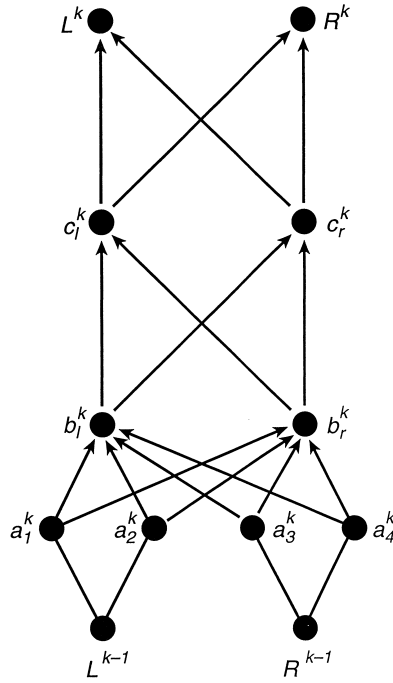


Fig. 20. EXTENDER.

we apply the recursive inbreeding coefficient calculation.

$$\begin{aligned}
 \mathcal{I}(X, P^0) &= \mathcal{K}(a_l^0, a_r^0, P^0 \cup Z) \\
 &= (2 + \mathcal{I}(b_l^0, P^0 \cup Z) + \mathcal{I}(b_r^0, P^0 \cup Z) + 4\mathcal{K}(b_l^0, b_r^0, P^0 \cup Z))/8 \\
 &= (6 + \mathcal{I}(c_l^0, P^0 \cup Z) + \mathcal{I}(c_r^0, P^0 \cup Z) + 8\mathcal{K}(c_l^0, c_r^0, P^0 \cup Z))/16 \\
 &= (8 + \mathcal{I}(L^0, P^0 \cup Z) + \mathcal{I}(R^0, P^0 \cup Z) + 6\mathcal{K}(L^0, R^0, P^0 \cup Z))/16 \\
 &= (8 + \mathcal{I}(L^0, Z) + \mathcal{I}(R^0, Z) + 6\mathcal{K}(L^0, R^0, Z))/16.
 \end{aligned}$$

Induction hypothesis: Suppose the claim hold true for all $k < n$ and for all specialized extender pedigrees Z .

Induction step ($k = n$): Using the induction hypothesis, we get

$$\begin{aligned}
 \mathcal{I}(X, P^{n-1}) &= 1 - 1/2^n + (6\mathcal{K}(L^{n-1}, R^{n-1}, Z) + \mathcal{I}(L^{n-1}, Z) + \mathcal{I}(R^{n-1}, Z))/2^{n+3} \\
 \Rightarrow \mathcal{I}(X, P^n) &= 1 - 1/2^n + (6\mathcal{K}(L^{n-1}, R^{n-1}, S^n) + \mathcal{I}(L^{n-1}, S^n) + \mathcal{I}(R^{n-1}, S^n))/2^{n+3} \\
 &= 1 - 1/2^n + (2\mathcal{K}(a_1^n, a_2^n, S^n) + 2\mathcal{K}(a_3^n, a_4^n, S^n) + 3\mathcal{K}(a_1^n, a_3^n, S^n) \\
 &\quad + 3\mathcal{K}(a_1^n, a_4^n, S^n) + 3\mathcal{K}(a_2^n, a_3^n, S^n) + 3\mathcal{K}(a_2^n, a_4^n, S^n))/2^{n+4} \\
 &= 1 - 1/2^n + (2 + \mathcal{I}(b_l^n, S^n) + \mathcal{I}(b_r^n, S^n) + 4\mathcal{K}(b_l^n, b_r^n, S^n))/2^{n+3} \\
 &= 1 - 1/2^n + (6 + \mathcal{I}(c_l^n, S^n) + \mathcal{I}(c_r^n, S^n) + 8\mathcal{K}(c_l^n, c_r^n, S^n))/2^{n+4} \\
 &= 1 - 1/2^n + (8 + \mathcal{I}(L^n, S^n) + \mathcal{I}(R^n, S^n) + 6\mathcal{K}(L^n, R^n, S^n))/2^{n+4}
 \end{aligned}$$

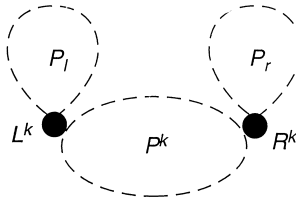


Fig. 21. EXTENDER1.

$$\begin{aligned}
 &= 1 - 1/2^n + (8 + \mathcal{J}(L^n, Z) + \mathcal{J}(R^n, Z) + 6\mathcal{H}(L^n, R^n, Z))/2^{n+4} \\
 &= 1 - 1/2^{n+1} + (\mathcal{J}(L^n, Z) + \mathcal{J}(R^n, Z) + 6\mathcal{H}(L^n, R^n, Z))/2^{n+4}. \quad \square
 \end{aligned}$$

Corollary 4.1. For $i < j$, $\mathcal{J}(X, P^i) < \mathcal{J}(X, P^j)$.

Corollary 4.2. $\lim_{k \rightarrow \infty} \mathcal{J}(X, P^k) \rightarrow 1$.

The above lemma and corollaries complete the first phase of the construction for $\text{inbreeding}(X) > 1/2$. For the second phase, we subdivide the range of inbreeding coefficients $1 - 1/2^{k+1} \leq W < 1 - 1/2^{k+2}$ as follows (for any $k \geq 0$):

1. $1 - 1/2^{k+1} \leq W < 1 - 7/2^{k+4}$,
2. $1 - 7/2^{k+4} \leq W < 1 - 13/2^{k+5}$,
3. $1 - 13/2^{k+5} \leq W < 1 - 3/2^{k+3}$,
4. $1 - 3/2^{k+3} \leq W < 1 - 5/2^{k+4}$,
5. $1 - 5/2^{k+4} \leq W < 1 - 9/2^{k+5}$,
6. $1 - 9/2^{k+5} \leq W < 1 - 1/2^{k+2}$.

For each range i , we define a function EXTENDER1 that constructs a pedigree Z such that $P^k \cup Z$ matches the inbreeding coefficient W in the corresponding range. The extender pedigree does not directly depend on k , but the constructions for EXTENDER4, EXTENDER5, and EXTENDER6 are nested. The primary role of k is to determine how large a stack P^k to put underneath the extender. For example, if $W = 925/1024$, then the largest k such that $1 - 1/2^k + 1 \leq W$ is $k = 2$. Given that $k = 2$, then $925/1024$ falls in the third range $920/1024 = (1 - 13/2^{k+5}) \leq W < (1 - 3/2^{k+3}) = 928/1024$. The output pedigree has EXTENDER3($925/1024$) stacked on top of P^2 .

EXTENDER1(W): Let $P_l, P_r = \text{GADGET8}(2^{k+3}(W - 1 + 1/2^{k+1}))$.

$V(P) = V(P^k) \cup V(P_l) \cup V(P_r)$,

$E(P) = E(P^k) \cup E(P_l) \cup E(P_r)$.

Coalesce vertices X^l, X^r with L^k, R^k , respectively. Fig. 21 shows EXTENDER1. Since the EXTENDER1 construction is for $1 - 1/2^{k+1} \leq W < 1 - 7/2^{k+4}$, we call GADGET8 with $0 \leq 2^{k+3}(W - 1 + 1/2^{k+1}) < 1/2$. In the pedigree produced,

$$\begin{aligned}
 \mathcal{J}(X, P) &= 1 - 1/2^{k+1} + (6\mathcal{H}(L^k, R^k, P) + \mathcal{J}(L^k, P) + \mathcal{J}(R^k, P))/2^{k+4} \\
 &= 1 - 1/2^{k+1} + 2^{k+3}(W - 1 + 1/2^{k+1})/2^{k+4} + 2^{k+3}(W - 1 + 1/2^{k+1})/2^{k+4} \\
 &= W.
 \end{aligned}$$

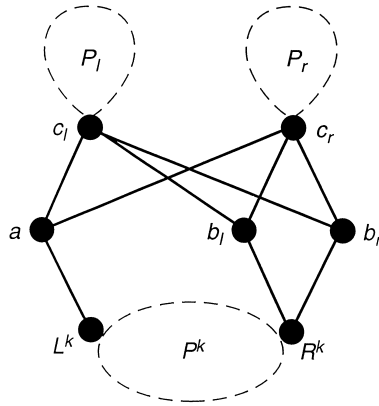


Fig. 22. EXTENDER2.

EXTENDER2(W): Let $P_l, P_r = \text{GADGET8}(2^{k+4}(W - 1 + 7/2^{k+4}))$.

$$V(P) = V(P^k) \cup \{a, b_l, b_r, c_l, c_r\} \cup V(P_l) \cup V(P_r),$$

$$E(P) = \{(L^k, a), (R^k, b_l), (R^k, b_r), (a, c_l), (a, c_r), (b_l, c_l), (b_l, c_r), (b_r, c_l), (b_r, c_r)\} \\ \cup E(P^k) \cup E(P_l) \cup E(P_r).$$

Coalesce X^l, X^r with c_l, c_r , respectively. Fig. 22 shows EXTENDER2. Since the EXTENDER2 construction is for $1 - 7/2^{k+4} \leq W < 1 - 13/2^{k+5}$, we call GADGET8 with $0 \leq 2^{k+4}(W - 1 + 7/2^{k+4}) < 1/2$. In the pedigree produced,

$$\begin{aligned} \mathcal{J}(X, P) &= 1 - 1/2^{k+1} + (6\mathcal{H}(L^k, R^k, P) + \mathcal{J}(L^k, P) + \mathcal{J}(R^k, P))/2^{k+4} \\ &= 1 - 1/2^{k+1} + (3\mathcal{H}(a, b_l, P) + 3\mathcal{H}(a, b_r, P) + 2\mathcal{H}(b_l, b_r, P))/2^{k+5} \\ &= 1 - 1/2^{k+1} + (2 + \mathcal{J}(c_l, P) + \mathcal{J}(c_r, P) + 4\mathcal{H}(c_l, c_r, P))/2^{k+5} \\ &= 1 - 7/2^{k+4} + 2^{k+4}(W - 1 + 7/2^{k+4})/2^{k+5} + 2^{k+4}(W - 1 + 7/2^{k+4})/2^{k+5} \\ &= W. \end{aligned}$$

EXTENDER3(W): Let $P_l = \text{GADGET8}(2^{k+4}(W - 1 + 13/2^{k+5}))$.

$$V(P) = V(P^k) \cup \{a_l, a_r, b_l, b_r, c_l, c_r\} \cup V(P_l),$$

$$E(P) = \{(L^k, a_l), (L^k, a_r), (R^k, b_l), (R^k, b_r), (a_l, c_l), (a_l, c_r), (b_l, c_l), (b_l, c_r), (b_r, c_l), \\ (b_r, c_r), (a_r, c_l)\} \cup E(P_l).$$

Coalesce X^l with c_l . Fig. 23 shows EXTENDER3. Since the EXTENDER3 construction is for $1 - 13/2^{k+5} \leq W < 1 - 3/2^{k+3}$, we call GADGET8 with $0 \leq 2^{k+4}(W - 1 + 13/2^{k+5}) < 1/2$. In the pedigree produced,

$$\begin{aligned} \mathcal{J}(X, P) &= 1 - 1/2^{k+1} + (6\mathcal{H}(L^k, R^k, P) + \mathcal{J}(L^k, P) + \mathcal{J}(R^k, P))/2^{k+4} \\ &= 1 - 1/2^{k+1} + (3\mathcal{H}(a_l, b_l, P) + 3\mathcal{H}(a_l, b_r, P) + 2\mathcal{H}(b_l, b_r, P) \\ &\quad + 3\mathcal{H}(a_r, b_l, P) + 3\mathcal{H}(a_r, b_r, P) + 2\mathcal{H}(a_l, a_r, P))/2^{k+5} \\ &= 1 - 1/2^{k+1} + (3 + 2\mathcal{J}(c_l, P) + \mathcal{J}(c_r, P) + 6\mathcal{H}(c_l, c_r, P))/2^{k+5} \end{aligned}$$

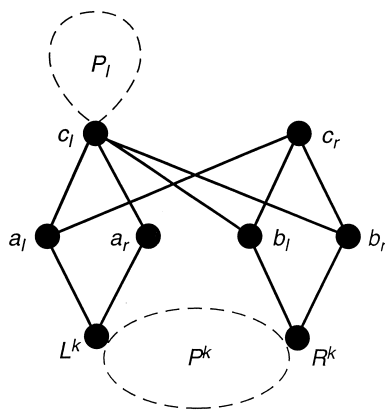


Fig. 23. EXTENDER3.

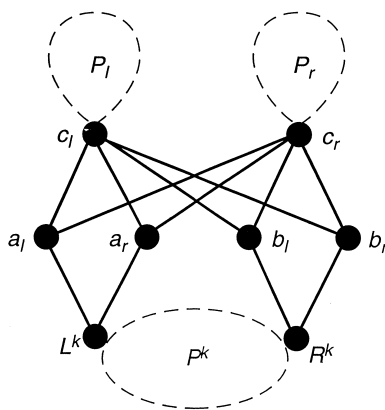


Fig. 24. EXTENDER4.

$$= 1 - 13/2^{k+5} + 2 \cdot 2^{k+4}(W - 1 + 13/2^{k+5})/2^{k+5} \\ = W.$$

EXTENDER4(W): Let $P_l, P_r = \text{GADGET8}(2^{k+3}(W - 1 + 3/2^{k+3}))$.

$$V(P) = V(P^k) \cup \{a_l, a_r, b_l, b_r, c_l, c_r\} \cup V(P_l) \cup V(P_r),$$

$$E(P) = \{(L^k, a_l), (L^k, a_r), (R^k, b_l), (R^k, b_r), (a_l, c_l), (a_l, c_r), (b_l, c_l), (b_l, c_r), (b_r, c_l), \\ (b_r, c_r), (a_r, c_l), (a_r, c_r)\} \cup E(P^k) \cup E(P_l) \cup E(P_r).$$

Coalesce X^l, X^r with c_l, c_r , respectively. Fig. 24 shows EXTENDER4. Since the EXTENDER4 construction is for $1 - 3/2^{k+3} \leq W < 1 - 5/2^{k+4}$, we call GADGET8 with $0 \leq 2^{k+3}(W - 1 + 3/2^{k+3}) < 1/2$. In the pedigree produced,

$$\mathcal{J}(X, P) = 1 - 1/2^{k+1} + (6\mathcal{H}(L^k, R^k, P) + \mathcal{J}(L^k, P) + \mathcal{J}(R^k, P))/2^{k+4} \\ = 1 - 1/2^{k+1} + (3\mathcal{H}(a_l, b_l, P) + 3\mathcal{H}(a_l, b_r, P) + 2\mathcal{H}(b_l, b_r, P))$$

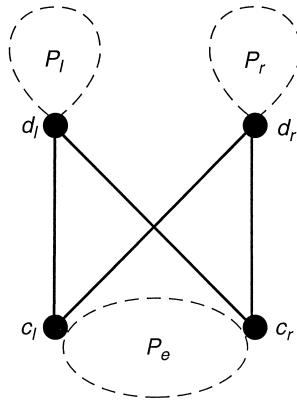


Fig. 25. EXTENDER5.

$$\begin{aligned}
 &= 3\mathcal{K}(a_r, b_l, P) + 3\mathcal{K}(a_r, b_r, P) + 2\mathcal{K}(a_l, a_r, P))/2^{k+5} \\
 &= 1 - 1/2^{k+1} + (2 + \mathcal{J}(c_l, P) + \mathcal{J}(c_r, P) + 4\mathcal{K}(c_l, c_r, P))/2^{k+4} \\
 &= 1 - 3/2^{k+3} + 2^{k+3}(W - 1 + 3/2^{k+3})/2^{k+4} + 2^{k+3}(W - 1 + 3/2^{k+3})/2^{k+4} \\
 &= W.
 \end{aligned}$$

EXTENDER5(W): Let $P_l, P_r = \text{GADGET8}(2^{k+4}(W - 1 + 5/2^{k+4}))$ and $P_e = \text{EXTENDER4}(1 - 3/2^{k+3})$.

$$V(P) = V(P_e) \cup \{d_l, d_r\} \cup V(P_l) \cup V(P_r),$$

$$E(P) = \{(c_l, d_l), (c_l, d_r), (c_r, d_l), (c_r, d_r)\} \cup E(P_e) \cup E(P_l) \cup E(P_r).$$

Collapse X^l, X^r with d_l, d_r , respectively. Fig. 25 shows EXTENDER5. Since the EXTENDER5 construction is for $1 - 5/2^{k+4} \leq W < 1 - 9/2^{k+5}$, we call GADGET8 with $0 \leq 2^{k+4}(W - 1 + 5/2^{k+4}) < 1/2$. To get the inbreeding coefficient of X . In the pedigree produced we start with one of the equations in the EXTENDER4 calculation,

$$\begin{aligned}
 \mathcal{J}(X, P) &= 1 - 1/2^{k+1} + (2 + \mathcal{J}(c_l, P) + \mathcal{J}(c_r, P) + 4\mathcal{K}(c_l, c_r, P))/2^{k+4} \\
 &= 1 - 3/2^{k+3} + (2 + \mathcal{J}(d_l, P) + \mathcal{J}(d_r, P) + 8\mathcal{K}(d_l, d_r, P))/2^{k+5} \\
 &= 1 - 5/2^{k+4} + 2^{k+4}(W - 1 + 5/2^{k+4})/2^{k+5} + 2^{k+4}(W - 1 + 5/2^{k+4})/2^{k+5} \\
 &= W.
 \end{aligned}$$

EXTENDER6(W): Let $P_l = \text{GADGET8}(2^{k+4}(W - 1 + 9/2^{k+5}))$ and $P_e = \text{EXTENDER5}(1 - 5/2^{k+4})$.

$$V(P) = V(P_e) \cup \{e_l, e_r\} \cup V(P_l),$$

$$E(P) = \{(d_l, e_l), (d_l, e), (d_r, e), (d_r, e_r)\} \cup E(P_e) \cup E(P_l).$$

Coalesce a_l^0, a_r^0 of P_l with e_l, e_r , respectively. Fig. 26 shows EXTENDER6. Since the EXTENDER6 construction is for $1 - 9/2^{k+5} \leq W < 1 - 1/2^{k+2}$, we call GADGET8 with $0 \leq 2^{k+4}(W - 1 + 9/2^{k+5}) < 1/2$. To get the inbreeding coefficient of X . In the pedigree produced we start with one of the equations in the EXTENDER5 calculation,

$$\mathcal{J}(X, P) = 1 - 3/2^{k+3} + (2 + \mathcal{J}(d_l, P) + \mathcal{J}(d_r, P) + 8\mathcal{K}(d_l, d_r, P))/2^{k+5}$$

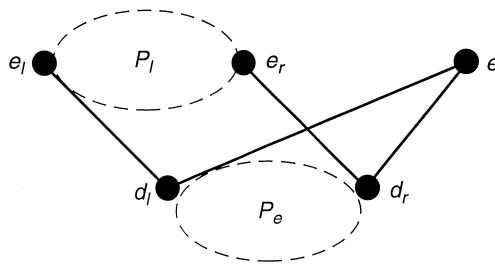


Fig. 26. EXTENDER6.

$$\begin{aligned}
 &= 1 - 5/2^{k+4} + (1 + \mathcal{I}(e, P) + 3\mathcal{K}(e, e_l, P) + 3\mathcal{K}(e, e_r, P) \\
 &\quad + 2\mathcal{K}(e_r, e_l, P))/2^{k+5} \\
 &= 1 - 9/2^{k+5} + 2 \cdot 2^{k+4}(W - 1 + 9/2^{k+5})/2^{k+5} \\
 &= W.
 \end{aligned}$$

The above constructions and calculations prove that:

Theorem 4.4. *For every valid inbreeding coefficient W such that $\frac{1}{2} < W < 1$, there is a human pedigree P with a sibship X , such that $I(X, P) = W$.*

5. Minimizing loop breakers

In this section we consider an inverse inbreeding coefficient problem in which we require all the loop breakers to be founders. We present an algorithm that given a coefficient of at most $\frac{1}{4}$, constructs a human pedigrees where all the loop breakers are founders and minimizes (C3).

When studying rare recessive diseases, geneticists sometimes look to minimize the number of times the disease allele must have entered the pedigree. In the extreme this means that the disease causing mutation occurred once, say in one founder individual F , and all the affecteds in the population have the mutation in both copies of DNA by inheriting a mutated copy from their father and mother who in turn inherited a mutated copy from F . Therefore, when studying a rare recessive disease, we may wish to construct a pedigree where all the mutations are explained by a single founder or founder couple and all the inbreeding loops are broken by the founder or founder pair. We call a pedigree with this property a *minimum founder human pedigree (MFHP)*.

We present an algorithm **Loops** that constructs a MFHP with one founder when the given coefficient $W \leq \frac{1}{8}$ and a MFHP with a founder couple when $\frac{1}{8} < W \leq \frac{1}{4}$. The pedigrees have only one loop breaker for $W \leq \frac{1}{8}$ and two loop breakers for $\frac{1}{8} < W \leq \frac{1}{4}$, where the loop breakers are required to be founders. We argue that this is the minimum number of loop breakers needed in any MFHP for these ranges of inbreeding coefficients. Fig. 27 shows a pedigree constructed by **Loops** for inbreeding coefficient

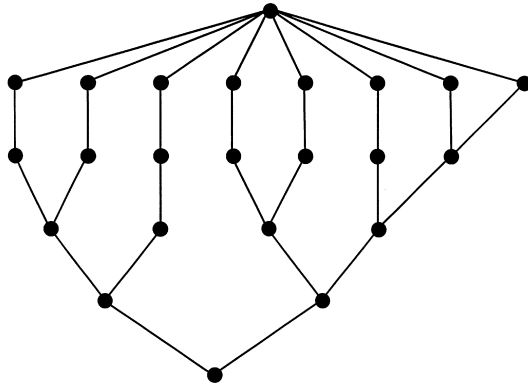


Fig. 27. Pedigree generated by LOOPS for coefficient 15/512.

15/512. Algorithm LOOPS uses two simple algorithms: BINARY and LEAVES. BINARY takes 2 input parameters n, d where $n \leq 2^d$, 3 output parameters X, V, E , and returns a binary tree with n branches of depth d such that X is the root, V is the set of vertices, and E is the set of edges of the binary tree. LEAVES takes 3 input parameters X, V, E defining a tree where X is the root, V is the set of vertices, and E is the set of edges of the tree, and returns the set of leaves in the tree.

Algorithm. LOOPS

Input: A valid inbreeding coefficient $W \leq \frac{1}{4}$

Output: A MFHP P such that it has an individual X with $\mathcal{I}(X, P) = W$.

begin

if $W > 1/8$

then $W \leftarrow W/2$, $double \leftarrow \text{true}$

else $double \leftarrow \text{false}$

find p_1, p_2 and minimum d such that $W = p_1 p_2 / 2^{2d+1}$ and $p_1, p_2 \leq 2^{d-1}$

 BINARY($p_1, d - 1, x_1, V_l, E_l$), BINARY($p_2, d - 1, x_2, V_r, E_r$)

$V(P) = \{X, F\} \cup V_l \cup V_r$, $E(P) = \{(X, x_1), (X, x_2)\} \cup E_l \cup E_r$

if $double = \text{true}$, $V(P) = V(P) \cup \{M\}$

if $double = \text{true}$

then $\forall l \in \text{LEAVES}(P)$, $E(P) = E(P) \cup \{(F, l), (M, l)\}$

else $\forall l \in \text{LEAVES}(P)$, $E(P) = E(P) \cup \{(F, l)\}$

return P

end

Lemma 5.1. For every valid inbreeding coefficient $W \leq \frac{1}{4}$, LOOPS(W) outputs MFHP P that has a sibship X such that $\mathcal{I}(X, P) = W$.

Proof. First, we show that LOOPS(W) always succeeds in finding p_1, p_2, d . Since $0 \leq W \leq \frac{1}{4}$ and we consider $W/2$ if $W > \frac{1}{8}$, we need to show that for $0 \leq W \leq \frac{1}{8}$ there

exist p_1, p_2, d such that $W = p_1 p_2 / 2^{2d+1}$ and $p_1, p_2 \leq 2^{d-1}$. Let $W = u/2^v$. Assign $p_1 = 4u$, $p_2 = 2^{v-1}$, $d = v$. Then $p_1 p_2 / 2^{2d+1} = 4u \cdot 2^{v-1} / 2^{2v+1} = u/2^v = W$, $p_2 = 2^{d-1}$, $p_1 = 4u = 4W2^v \leq 4(1/8)2^v = 2^{v-1} = 2^{d-1}$, and we have p_1, p_2, d with the required properties. For the proof, it suffices to show that when *double* is **false** [**true**], the number of inbreeding loops is $p_1 \cdot p_2 [2 \cdot p_1 \cdot p_2]$ and the length of each inbreeding loop is $2d + 2$. We show this next.

$\text{BINARY}(p_1, d, x_1, V_l, E_l)$ creates a binary tree with p_1 leaves and p_1 paths of length d . Similarly, $\text{BINARY}(p_2, d, x_2, V_r, E_r)$ creates a binary tree with p_2 leaves and p_2 paths of length d . Let $l \in V_l [r \in V_r]$ be a leaf and $P_l [P_r]$ be the path connecting $x_1 [x_2]$ to $l [r]$. Then the pair of paths $(X, x_1)P_l(l, F)$ and $(X, x_2)P_r(r, F)$ results in an inbreeding loop of length $2d + 2$ for X . If *double* is **true**, we get another inbreeding loop from the pair of paths $(X, x_1)P_l(l, M)$ and $(X, x_2)P_r(r, M)$. As there are p_1 choices for l and p_2 choices for r , the number of inbreeding loops is $p_1 \cdot p_2 [2 \cdot p_1 \cdot p_2]$ if *double* is **false** [**true**] and the length of each inbreeding loop is $2d + 2$. \square

For $0 \leq W \leq \frac{1}{8}$, the loop breaker in the pedigree produced by *LOOPS* is F and for $\frac{1}{8} < W \leq \frac{1}{4}$, the loop breakers in the pedigree produced by *LOOPS* are F and M . Next, we show that the maximum inbreeding coefficient that can be realized with one or two loop breakers in MFHP is indeed $\frac{1}{8}$ and $\frac{1}{4}$, respectively. We first need the following preliminary result.

Lemma 5.2. *If P is a pedigree with one loop breaker F , $\mathcal{I}(X, P) = W$, and the length of longest path from F to X is d , then there exists a pedigree P' with one loop breaker F , $\mathcal{I}(X, P') = W$ such that the length of every path from F to X is d .*

Proof. Let P, X, F, W be as stated. For every path $P_1 : X, \dots, y, F$ of length $k < d$, create a binary tree of depth $(d - k)$, coalesce the root of the binary tree with y , delete edge (y, F) and for every leaf x of the binary tree, add the edge (x, F) . For the proof, it suffices to show that the above modification maintains the inbreeding coefficient.

Let L be an inbreeding loop consisting of path P_1 and some other path P_2 , such that $k + z$ was the length of the loop. The reduction in inbreeding coefficient because of deleting edge (y, F) and breaking loop L is $1/2^{k+z-1}$. The gain because of creating 2^{d-k} loops of length $d + z$ with path P_2 is $2^{d-k}/2^{d+z-1} = 1/2^{k+z-1}$. \square

Lemma 5.3. *If the number of loop breakers in a MFHP P is one [two], then $\mathcal{I}(X, P)$ for any $X \in V(P)$ is at most $\frac{1}{8} [\frac{1}{4}]$.*

Proof. Let P be a pedigree with one loop breaker F and $\mathcal{I}(X, P) = W$. From the definition of MFHP and Lemma 5.2, we can assume that F does not have an ancestor in P and the lengths of all the paths from X to F are equal, say $d + 1$. As there can be at most $2^{(d-1)}$ paths from F to *father*(X) and *mother*(X), the maximum inbreeding coefficient we can get is $2^{(d-1)} \times 2^{(d-1)} / 2^{(2d+1)} = \frac{1}{8}$.

Let P be a pedigree with two loop breakers F and M and let $d + 1$ be the length of a longest path from X to F or M . From the proof of Lemma 5.2, it follows that we can construct a pedigree such that the lengths of all the paths from X to F and from X to M are equal, say $d + 1$, and the inbreeding coefficients are kept the same. The maximum number of paths from $father(X)$ and $mother(X)$ to F or M can be $2^{(d-1)}$ when the length of a longest path from X to F or M is $d + 1$. All of these paths can be present in the same pedigree if F and M is a married couple. Hence, the maximum inbreeding we can get is $2 \times 2^{(d-1)} \times 2^{(d-1)} / 2^{(2d+1)} = \frac{1}{4}$. \square

If we do not restrict ourselves to MFHP but are interested in human pedigree, then we can have pedigrees with one [two] loop breaker[s], such that $\mathcal{J}(X, P)$ for any $X \in V(P)$ is more than $\frac{1}{8}[\frac{1}{4}]$. For example, pedigree P with edges

$$E(P) = \{(x_1, x_2), (x_1, x_3), (x_2, x_4), (x_3, x_4), (x_4, x_5), (x_4, x_6), \\ (x_5, x_7), (x_5, x_8), (x_6, x_7), (x_6, x_8)\}$$

has only one loop breaker x_4 and $\mathcal{J}(x_1, P) = 5/32$ and pedigree P' with edges

$$E(P') = E(P) \cup \{(x_2, x_9), (x_3, x_9), (x_9, x_{10}), (x_9, x_{11}), \\ (x_{10}, x_7), (x_{10}, x_8), (x_{11}, x_7), (x_{11}, x_8)\}$$

has two loop breakers x_4 and x_9 and $\mathcal{J}(x_1, P) = 5/16$.

6. Discussion and open problems

We defined a collection of problems called inverse inbreeding coefficient problems. We presented three constructions for pedigrees in which one sibship has an inbreeding coefficient matching a specified probability. The constructions differ in whether they allow all matings or only (typical) human matings, and in which optimality criteria are satisfied. We wrote a program to construct human pedigrees for coefficients of at most $\frac{3}{16}$ using MINLOOPS and GADGET1. The software is available by sending e-mail to richa@helix.nih.gov.

Many open problems remain. The most important open problem for one sibship is to find a construction for human pedigrees that simultaneously minimizes the number of persons (C1) and either of the loop criteria (C2) or (C3). We also left open the possibly simpler problems of finding human pedigrees to minimize just the number of persons, or minimize just the number of inbreeding loops for coefficients $> \frac{1}{8}$, or minimize just the number of loop breakers for coefficients $> \frac{1}{4}$. For non-human applications we left open the problem of finding a construction that minimizes the number of loop breakers when the coefficient is $> \frac{1}{2}$. All variants of the inverse inbreeding coefficient problem are open for multiple sibships in the same pedigree. One could also add a constraint on how well a multi-sibship construction approximates the kinship coefficient of each non-sibling pair of affected persons.

Our work was primarily motivated by a problem of pedigree replacement in linkage analysis of recessive diseases [3], and we showed that our human pedigree construction yields smaller pedigrees for the motivating example, and another recently published example. Our work is also applicable to more large-scale epidemiology studies of inbred populations. In the epidemiology setting it is useful to construct simulated pedigrees matching the inbreeding coefficient pattern, and to trace the inheritance of traits under various models of inheritance, as a test of the models. Our constructions make it possible to systematically generate small pedigrees from any desired distribution of inbreeding coefficients.

References

- [1] R. Agarwala, L.G. Biesecker, K.A. Hopkins, C.A. Francomano, A.A. Schäffer, Software for constructing and verifying pedigrees within large genealogies and an application to the old order amish of Lancaster county, *Genome Res.* 8 (1998) 211–221.
- [2] A. Becker, D. Geiger, A.A. Schäffer, Automatic selection of loop breakers for genetic linkage analysis, *Hum. Heredity* 48 (1998) 49–60.
- [3] S. Chang, M.J. Rosenberg, H. Morton, C.A. Francomano, L.G. Biesecker, Identification of a mutation in liver glycogen phosphorylase in glycogen storage disease type VI, *Hum. Mol. Genet.* 7 (1998) 865–870.
- [4] R.C. Elston, J. Stewart, A general model for the analysis of pedigree data, *Hum. Heredity* 21 (1971) 523–542.
- [5] S.-W. Guo, Computation of multilocus prior probability of autozygosity for complex inbred pedigrees, *Genet. Epidemiol.* 14 (1997) 1–15.
- [6] G. Karigl, Genealogical relationship: its measurement and its use in human genetics, Ph.D. Thesis, Technische Universität Wien, 1982.
- [7] A. Kong, Efficient methods for computing linkage likelihoods of recessive diseases in inbred pedigrees, *Genet. Epidemiol.* 8 (1991) 81–103.
- [8] L. Kruglyak, M.J. Daly, E.S. Lander, Rapid multipoint linkage analysis of recessive traits in nuclear families, including homozygosity mapping, *Amer. J. Hum. Genet.* 56 (1995) 519–527.
- [9] E.S. Lander, D. Botstein, Homozygosity mapping: a way to map human recessive traits with the DNA of inbred children, *Science* 236 (1987) 1567–1570.
- [10] E.S. Lander, P. Green, Construction of multilocus genetic linkage maps in humans, *Proc. Natl. Acad. Sci. USA* 84 (1987) 2363–2367.
- [11] K. Lange, E. Sobel, A random walk method for computing genetic location scores, *Amer. J. Hum. Genet.* 49 (1991) 1320–1334.
- [12] G.M. Lathrop, J.M. Lalouel, C. Julier, J. Ott, Strategies for multilocus linkage analysis in humans, *Proc. Natl. Acad. Sci. USA* 81 (1984) 3443–3446.
- [13] S. Lin, E.A. Thompson, E.M. Wijsman, An algorithm for Monte Carlo estimation of genotype probabilities on complex pedigree, *Ann. Hum. Genet.* 58 (1994) 343–357.
- [14] S. Lin, E.A. Thompson, E.M. Wijsman, Finding non-communicating sets for Markov chain Monte Carlo estimations on pedigrees, *Amer. J. Hum. Genet.* 54 (1994) 695–704.
- [15] E.J. Neufeld, H. Mandel, T. Raz, R. Szargel, C.N. Yandava, A. Stagg, S. Fauré, T. Barrett, N. Buist, N. Cohen, Localization of the gene for thiamine-responsive megaloblastic anemia syndrome, on the long arm of chromosome 1, by homozygosity mapping, *Amer. J. Hum. Genet.* 61 (1997) 1335–1341.
- [16] J. Ott, Analysis of Human Genetic Linkage, Revised Edition, The Johns Hopkins University Press, Baltimore, 1991.
- [17] A.A. Schäffer, Faster linkage analysis computation for pedigrees with loops or unused alleles, *Hum. Heredity* 46 (1996) 226–235.
- [18] C.A.B. Smith, Detection of linkage in human genetics, *J. Roy. Statist. Soc. B* 15 (1953) 153–192.
- [19] H.J. Suermont, G.F. Cooper, Probabilistic inference in multiply connected belief networks using loop cutsets, *Int. J. Approx. Reason.* 4 (1990) 283–306.

- [20] A. Thomas, Linkage analysis on complex pedigrees by simulation, *IMA J. Math. Appl. Med. Biol.* 11 (1994) 79–93.
- [21] E.A. Thompson, *Pedigree Analysis in Human Genetics*, Johns Hopkins University Press, Baltimore, MD, 1973.
- [22] E.A. Thompson, Monte Carlo likelihood in genetic mapping, *Statist. Sci.* 9 (1994) 355–366.
- [23] B. Weir, *Genetic Data Analysis II*, Sinauer Associates, Sunderland, MA, 1996.
- [24] S. Wright, Mendelian analysis of the pure breeds of livestock. I. The measurement of inbreeding and relationship, *J. Heredity* 14 (1923) 339–348.