

Linkage Analysis Package  
User's Guide to Analysis Programs

Version 5.10 for IBM PC/compatibles  
10 Oct 1996, updated 2 November 2013

## Table of Contents

<b>CHAPTER 1: INTRODUCTION</b> .....	<b>1</b>
1.0 OVERVIEW.....	1
1.1 GENERAL FEATURES .....	2
1.2 HOW TO PROCEED .....	2
<b>CHAPTER 2: STRUCTURE OF INPUT DATA</b> .....	<b>3</b>
2.0 OVERVIEW.....	3
2.1 PHENOTYPES AND GENOTYPES .....	3
2.2 NUMBERED ALLELES .....	3
2.3 BINARY FACTORS.....	3
2.4 AFFECTION STATUS .....	5
2.5 QUANTITATIVE VARIABLES .....	5
2.6 DESCRIPTIONS OF LOCI (DATAFILE).....	5
2.7 PEDIGREE INFORMATION (PEDFILE).....	10
2.8 CONSANGUINITY AND MARRIAGE LOOPS .....	15
<b>CHAPTER 3: ANALYSIS PROGRAMS</b> .....	<b>17</b>
3.0 OVERVIEW.....	17
3.1 CONSTANTS .....	17
3.2 I LINK.....	19
3.3 M LINK.....	21
3.4 LINKMAP.....	23
3.5 C I LINK .....	24
3.6 C MAP .....	25
3.7 LOD SCORE.....	26
3.8 C LOD SCORE.....	26
<b>CHAPTER 4: AUXILIARY PROGRAMS</b> .....	<b>26</b>
4.0 OVERVIEW.....	26
4.1 UNKNOWN.....	26
4.2 C FACTOR .....	27

## Chapter 1: INTRODUCTION

### 1.0 OVERVIEW

The core of the LINKAGE package is a series of programs for maximum likelihood estimation of recombination rates, calculation of lod score tables, and analysis of genetic risks. The analysis programs are divided into two groups. The first group can be used for general pedigrees with marker and disease loci. Programs in the second group are for three-generation families and codominant marker loci, and are primarily intended for the construction of genetic maps from data on reference families.

The input to the LINKAGE programs is divided into pedigree and genotypic data, on the one hand; and locus description, recombination rates, and gene order, on the other. The pedigree and genotypic data must be processed prior to analysis by a series of preparatory programs that accompany the analytic programs in the LINKAGE package.

The LINKAGE package contains additional control programs that provide a “shell,” or interface, to facilitate the use of the analytic programs. The control programs are described in accompanying documents (see files at

<http://www.jurgott.org/linkage/LinkagePC.html>).

## ***1.1 GENERAL FEATURES***

The programs for general pedigrees allow linkage analysis with an arbitrary number of loci, either sex-linked or autosomal. In addition to marker loci, affection-status or quantitative phenotypes can be considered. Incomplete penetrance and liability (risk) classes can be specified for affection-status phenotypes, and several correlated quantitative measurements can be incorporated simultaneously. Program options allow for mutation at a single locus, with separate male and female mutation rates, and for linkage disequilibrium between different loci. Pedigrees can contain one or more inbreeding loops. Other programs in the analysis package are optimized for rapid likelihood calculations with codominant data, in reference three-generation families.

The following is a quick guide to the programs available for some common applications:

Estimation of recombination rates and calculation of the maximum lod score:  
ILINK for general pedigrees; CILINK for three-generation reference pedigrees.

Lod score tables and risk analysis: MLINK.

Location scores: LINKMAP for general pedigrees, CMAP for three-generation reference pedigrees.

## ***1.2 HOW TO PROCEED***

Running the LINKAGE programs requires the following steps:

### (1) Input pedigree and genotypic data

The pedigree and genotypic data must be entered into a single file. A text editor or database system that interfaces with the LINKAGE programs can be used for this. Neither of these programs is supplied with the LINKAGE package.

If you are entering data with a text editor, please consult section 2.8 of this manual for the format to use. The program MAKEPED, described in a separate document (see <http://www.jurgott.org/linkage/LinkagePC.html>), can assist in the construction of the necessary genealogical pointers.

### (2) Description of loci

A file describing the loci must be constructed with a text editor or with the program PREPLINK. Please consult section 2.6 for the format of this file.

### (3) Analysis

You must choose the LINKAGE program suitable for the analysis that you wish to undertake. The data and pedigree files constructed in (1) and (2) serve as input either directly to the LINKAGE programs or to the LINKAGE CONTROL PROGRAM (LCP), which is a shell for generating command files to control the analysis programs.

The LINKAGE programs contain constants that determine upper limits to the number of loci, alleles, etc. that can be considered simultaneously. Verify that your problem does not exceed the limits established when the program was compiled. If your needs exceed these limits, you can change the constants and recompile the programs. Please consult Chapters 3 and 4 for a list of the program constants and their meaning.

### (4) Modifying input files used with previous versions of LINKAGE

Because older versions of LINKAGE did not contain provisions for different mutation rates in males and females, you must modify any earlier files by adding a second mutation rate. Also, add the program identifiers required by the LCP

shell. See section 2.6 for a description of these modifications.

(5) Use with the CEPH data base

Data extracted from the CEPH data base with the program SETPED (supplied only in the CEPH program package) can be analyzed with LINKAGE if the file PAR.IN, containing the loci description, is modified as above. Pedigree data is contained in the file PED.IN.

## Chapter 2: STRUCTURE OF INPUT DATA

### 2.0 OVERVIEW

The input data consist of pedigree and genotypic information contained in one file, and locus descriptions, recombination rates and locus order contained in a second file. Internally (i.e. in the program code), these files are called PEDFILE and DATAFILE respectively.

### 2.1 PHENOTYPES AND GENOTYPES

To understand the format of the input files, you must know what kinds of phenotypic data can be interpreted by the LINKAGE programs. Phenotype data can be one of the following types:

- (a) Numbered alleles. These are codominant alleles at a single locus. The numbers run consecutively from 1 to the maximum number of alleles observed. The phenotype consists of two allele numbers corresponding to a genotype. An unknown genotype is coded as 0 0.
- (b) Binary factors. In this coding scheme a series of binary codes (1 or 0) indicates the presence or absence of a phenotype factor. This system is useful for describing either codominant or recessive/dominant systems. The phenotype is entered as a binary string.
- (c) Affection status. The presence or absence of disease (or other qualitative phenotype) is described by a numbered code. A risk or liability class can also be included as a separate numeric code.
- (d) Quantitative traits. One or more quantitative measurements can be used as a phenotype description.

The phenotypic codes for each of these types of data are described in more detail below.

NOTE: The present version of LINKAGE does not allow mixtures of affection status and quantitative variables, except in the case of sex-linked traits as described below. A future version will incorporate a modification allowing such mixtures for autosomal data.

### 2.2 NUMBERED ALLELES

Numbering alleles is the **simplest way** to code codominant marker data. A homozygote is indicated by repeating the allele number; thus 1 2 indicates that the alleles are 1 and 2 (a heterozygote) while 1 1 indicates the alleles are 1 and 1 (a homozygote). An unknown genotype is coded as 0 0. For sex-linked loci, males have a single allele. With the allele 1, for example, the phenotype can be coded 1 0 or 1 1.

### 2.3 BINARY FACTORS

Binary factors (sometimes called “factor-union” notation) can also represent phenotypes for codominant marker data, but this coding is most useful with recessive alleles or with complex systems such as Rh, ABO, and Gm. Each allele is assigned a set of properties, called factors, in such a way that all phenotypes can be specified as the union of two allele sets.

For codominant loci, each allele can be associated with one factor. If  $n$  alleles are present, the  $i$ th allele is represented by a series of  $n$  binary codes with a 0 in all locations, except in the  $i$ th position, which contains a 1. For example, in a two allele

system the allelic codes are:

1 0 (allele 1)  
0 1 (allele 2)

The three possible phenotypes are:

1 0 (union of alleles 1 and 1)  
1 1 (union of alleles 1 and 2)  
0 1 (union of alleles 2 and 2)

An *unknown* phenotype is coded as 0 0. Spaces between the codes are very important; they must be included when entering the phenotypes into the pedigree file as described below.

A locus with three codominant alleles is coded as:

1 0 0 (allele 1)  
0 1 0 (allele 2)  
0 0 1 (allele 3)

The six possible phenotypes are:

1 0 0 (union of alleles 1 and 1)  
1 1 0 (union of alleles 1 and 2)  
1 0 1 (union of alleles 1 and 3)  
0 1 0 (union of alleles 2 and 2)  
0 1 1 (union of alleles 2 and 3)  
0 0 1 (union of alleles 3 and 3)  
and an unknown phenotype is 0 0 0.

The advantage of the binary factor coding scheme is evident when a (fully penetrant) recessive disease gene is under study. To code such a system, we could indicate the normal gene by the presence of a single factor (1) and the disease gene by the absence of this factor (0). The phenotype 1 (unaffected) now corresponds to two possible genotypes, either the union of allele 1 and allele 1 (noncarrier) or the union of allele 1 and allele 0 (carrier).

This simple coding is usually not sufficient because both homozygote recessive and unknown phenotypes are coded as 0. To account for this, we introduce a second factor for which a 1 indicates that the phenotype is known, and a 0 that the phenotype is unknown. The allelic codes are:

1 1 (allele 1)  
0 1 (allele 2)

and the possible phenotypes are:

1 1 (union of alleles 1 and 1, or alleles 1 and 2)  
0 1 (union of alleles 2 and 2)  
0 0 (unknown)

## 2.4 AFFECTION STATUS

“Affection status” refers to the presence or absence of disease. The programs assume that an affected individual will have the phenotype code “2” and that an unknown individual will have the code “0.” By convention, “1” is used to designate unaffected status (in fact, this code can be any integer value other than 0 and 2). If necessary, the unknown and affected codes can be changed in the program code, and the programs recompiled.

For an “affection-status” locus, each genotype has an associated penetrance; this is the probability that an individual with a particular genotype will be affected. Penetrance can also be defined as a function of liability classes. In this case, one penetrance is given for each genotype in each liability class. The classes are numbered sequentially starting from 1. With two or more liability classes, the phenotype is the affection status plus the class number. When a single affection status class is defined, the class number is not included as part of the phenotype.

With sex-linked traits, different penetrances must be given for females and males. One penetrance in males is specified for each allele in each liability class.

## 2.5 QUANTITATIVE VARIABLES

Phenotypic information is sometimes presented in the form of quantitative measurements, e.g. creatine kinase for carrier detection in Duchenne muscular dystrophy. The phenotype is then the quantitative value. Unknown phenotypes are entered as 0.0. (The code for unknown quantitative values is a program constant that can be changed.) The genotypic means, the variance of the trait in homozygotes, and the ratio of the variances in heterozygotes and homozygotes must be specified.

If several traits are measured for the same locus, the phenotype is the list of all the variables. A single value of 0.0 in the list is interpreted as an unknown phenotype. The means must be given for each variable as a function of genotype, along with the variance-covariance matrix. The variance matrices for homozygotes and heterozygotes can differ by a constant factor.

For sex-linked traits it is assumed that males will have an affection-status variable rather than a quantitative value. If several variables have been measured ( $n_{\text{trait}}$ ), a male phenotype consists of affection status followed by  $n_{\text{trait}}-1$  arbitrary entries (for example, zeros). The present version of the programs supports only one affection status class, with full penetrance of the disease allele for sex-linked traits.

## 2.6 DESCRIPTIONS OF LOCI (DATAFILE)

Descriptions of loci and other information are contained in DATAFILE. The information in this file is divided into four parts: (1) general information on loci and locus order; (2) description of loci; (3) information on recombination; (4) program-specific information.

In explaining the structure of DATAFILE we will use two concepts of locus order. The first is the input order, or the order in which the phenotypes corresponding to the loci appear in PEDFILE (see section 2.7). The second is chromosome order, or the physical order assumed for the loci. The input order is fixed once PEDFILE is created, but the chromosome order can be changed to test various hypotheses.

Various parameters such as recombination rates, gene frequencies, penetrances, etc., are specified in the DATAFILE. These refer to the initial values of these parameters. The analysis programs can modify some of these values for specific purposes, e.g. maximum likelihood estimation. This feature is explained in Chapter 3.

The DATAFILE can be prepared with the program PREPLINK (see <http://www.jurgott.org/linkage/LinkagePC.html>).

### **Example**

Before we attempt to explain the format of various parts of the DATAFILE, it is useful to consider a complete file as an

example. The following is the DATAFILE for three sex-linked loci, one of which is Duchenne muscular dystrophy; creatine kinase measurements are available for heterozygote testing in women:

```

3 0 1 5          << no loci, risk locus, sexlinked (if 1), program code
3 0.001 0.001 0 << mut locus, mut mal, mut fem, hap freq (if 1)

1 3 2           << order of loci

2 2             <<< binary factors, # alleles
5.00000E-01 5.00000E-01 << gene freqs
2              << number of binary factors
1 0
0 1            << allelic codes

2 2             <<< binary factors, # alleles
5.00000E-01 5.00000E-01 << gene freqs
2              << number of binary factors
1 0
0 1            << allelic codes

0 2             <<< quan, # alleles
9.99800E-01 2.00000E-04 << gene freqs
1              << number of traits
1.57000E+00 2.10000E+00 2.10000E+00 << genotype means
5.90000E-02    << variance
2.90000E+00    << multiplier for variance in heterozygotes
0 0            << sex difference (if 1) and interference (if 1)
0.1 0.1        << recombination values
1 0.5 0.5

```

The last line contains information for the MLINK program; this is indicated by the program code 5 on the first line. Other parameters are specified as indicated in the comments following certain lines (indicated by <<). Comments are allowed on some lines for easy interpretation of the file.

### **Loci and Locus Order**

The first two lines of DATAFILE contain information on a variety of parameters, including the number of loci (*nlocus*), a risk locus (*risklocus*), sex-linked or autosomal data (*sexlink*), a mutation locus (*mutsys*) and mutation rates (*mutmale* and *mutfem*), linkage disequilibrium (*disequil*), and a program code (*nprogram*). The first two lines are followed by a third line giving the chromosome order for the loci. The format is:

```

nlocus risklocus      sexlink nprogram
mutsys mutmale      mutfem disequil
(chromosome order)

```

*mutsys* and the chromosome order of the loci must begin on new lines; comments can follow at the end of each line. *nprogram* is not used by the LINKAGE programs, but is required for interfacing with the shell program LCP. It is used to describe the program for which the file is constructed. LCP can use files constructed for one program as input for a different program. Therefore the datafile is not changed for different programs when using LCP.

Valid values for the variables are:

```

nlocus      =      1 to maxlocus (as specified by a constant in the programs)

risklocus   =      0 if risk is not to be calculated
                =      disease locus number (input order) if risk is to be calculated

sexlink     =      0 for autosomal data
                =      1 for sex-linked data

```

`nprogram` = 1 CILINK  
               2 CMAP  
               3 ILINK  
               4 LINKMAP  
               5 MLINK  
               6 LODSCORE  
               7 CLODScore

`mutsys` = 0 if mutation rates are zero  
           = mutation locus number (input order) for non-zero mutation rates

`mutmale` = male mutation rate

`mutfem` = female mutation rate

`disequil` = 0 if loci are assumed to be in linkage equilibrium  
           = 1 if loci are in linkage disequilibrium

When loci are in linkage equilibrium, allele frequencies must be given under each locus description; otherwise, haplotype frequencies are provided. When risk is calculated, a disease allele is provided in the locus description for the “risklocus.” As an example, consider the analysis of 3 autosomal loci in the chromosome order 1 3 2. The first three lines of the DATAFILE could be:

```

3 0 0 3    << no loci, risk locus, sexlinked (if 1), program code
3 0.1 0.1 0 << mut locus, mut mal, mut fem, haplotype freq (if 1)
1 3 2     << order of loci
  
```

The data are autosomal with mutation at the third locus.

### **Description of Loci**

The loci are described in the order in which they appear in the PEDFILE (see section 2.7). Assuming linkage equilibrium, the gene frequencies are specified as part of the locus description (linkage disequilibrium will be documented in a later version). The descriptions differ according to the type of locus. A numeric code distinguishes each of the types:

```

0 = Quantitative variable
1 = Affection status
2 = Binary factors
3 = Numbered alleles
  
```

The format for each locus type, assuming linkage equilibrium, is as follows:

### **Numbered alleles**

The locus description consists of two lines. The first gives the code for numbered alleles and the total number of alleles. The second gives the gene frequencies. For example:

```

3 2    << numbered alleles code, total number of alleles
0.5 0.5 << gene frequencies
  
```

specifies two alleles with equal gene frequencies.

## **Binary factors**

The first two lines are similar to those in the previous example. After this the number of factors is specified on a separate line, followed by one line for each allele specification. As an example, consider the case of a recessive trait:

```
2 2          << binary factor code, number of alleles
0.999 0.001 << gene frequencies
2           << number of factors
1 1
0 1         << alleles
```

## **Affection status**

The number of liability classes replaces the number of factors, and penetrances are given for each genotype in each class:

```
1 2          << affection status code, number of alleles
0.999 0.001 << gene frequencies
1           << number of liability classes
0.0 1.0 1.0 << penetrances
```

describes a fully penetrant, dominant disease locus. The genotypes are in the order 11, 12, 22 where 1 is the first allele and 2 is the second allele specified in the gene frequency list. For three alleles, the genotype order is 11, 12, 13, 22, 23, 33. The same pattern is followed for more alleles. To describe a similar locus, but with reduced penetrance and two liability classes, use the following:

```
1 2          << affection status code, number of alleles
0.999 0.001 << gene frequencies
2           << number of liability classes
0.0 0.5 0.5 << penetrances
0.0 0.9 0.9
```

With sex-linked data, male penetrances must also be defined for each allele. The following describes a sex-linked disease with 50% penetrance in males:

```
1 2          << affection status code, number of alleles
0.999 0.001 << gene frequencies
1           << number of liability classes
0.0 0.0 1.0 << female followed by male penetrances
0.0 0.5
```

## **Quantitative trait**

Quantitative traits are described by a first line containing the quantitative code (0) and the number of alleles, and a second line with gene frequencies, as in the previous examples. These are followed by lines indicating the number of quantitative variables, genotypic means for each variable, a variance-covariance matrix, and a constant that gives the ratio of variance-covariance in heterozygotes to homozygotes.

For a single quantitative variable, the format is:

```
0 2          << quantitative variable code, number of alleles
0.999 0.001 << gene frequencies
```

1 << number of quantitative variables  
 10.0 12.0 14.0 << genotypic means  
 1.5 << variance  
 1.0 << multiplier for heterozygote variance

The genotypes are 1/1, 1/2 and 2/2, respectively, where allele 1 has the frequency 0.999. For two quantitative variables, the description is:

0 2 << quantitative variable code, number of alleles  
 0.999 0.001 << gene frequencies  
 2 << number of liability classes  
 10.0 12.0 14.0  
 -10.0 0.0 10.0 << genotypic means  
 1.5 10.0 100.0 << variance-covariance  
 1.0 << multiplier for heterozygote variance-covariance

Only the upper triangle of the variance-covariance matrix is given; the order is  $V_{11}$ ,  $V_{12}$ ,  $V_{13}$  ...  $V_{22}$ ,  $V_{23}$  ... etc. Here, the variance of the first variable is 1.5, the covariance is 10.0, and the variance of the second variable is 100.0. When describing the "risk locus," the disease allele (risk allele) must be designated at the end of the locus description. For example:

1 2 << affection status code, number of alleles  
 0.999 0.001 << gene frequencies  
 1 << number of liability classes  
 0.0 1.0 1.0 << penetrances  
 2 << risk allele

### **Recombination Information**

In addition to recombination rates, sex-differences and interference must be specified in this section. Sex-difference options are indicated by an integer variable that takes the following values:

0 = no sex-difference  
 1 = constant sex-difference (the ratio of female/male genetic distance is the same in all intervals)  
 2 = variable sex-difference (the female/male distance ratio can be different in each interval)

The interference option can take the following values:

0 = no interference  
 1 = interference without a mapping function  
 2 = user-specified mapping function

Interference (i.e. options 1 or 2) is allowed only in some analysis programs with three loci. The programs, as distributed, contain Kosambi interference as the user-specified mapping function.

First, consider a case without interference. When the sex-difference is "0," one recombination rate is given for each of the  $n$ locus-1 segments (see the complete example above). If the sex-difference option is "1," the male recombination rates are given on one line, and the female/male genetic distance is specified on the next line, e.g.:

1 0 << sex difference, interference  
 0.1 0.2 0.1 << male recombination  
 2.0 << female/male ratio of genetic distance

When the sex-difference option is “2”, the male recombination rates are followed on the next line by female recombination rates:

```
2 0          << sex difference, interference
0.1 0.2 0.1  << male recombination
0.2 0.1 0.2  << female recombination
```

Interference can be specified for three loci. With the interference option 1, three recombination rates are given. These are the recombination rates between adjacent loci in the two segments and the recombination rate between the flanking loci. An example is:

```
1 1          << sex difference, interference
0.1 0.1 0.18 << male recombination
2.0         << female/male ratio of genetic distance
```

With the interference option 2, only the rates between the adjacent loci are provided:

```
1 2          << sex difference, interference
0.1 0.1      << male recombination
2.0         << female/male ratio of genetic distance
```

### **Program-specific information**

The program-specific information consists of a series of lines at the end of the DATAFILE describing which parameters should be varied iteratively by the analysis programs. The format for each program is described in Chapter 3.

### ***2.7 PEDIGREE INFORMATION (PEDFILE)***

In addition to phenotypes and description of loci, the LINKAGE programs require pedigree information in order to traverse the pedigree when calculating the likelihood. The input must contain the following information for each individual:

- a pedigree number (or name; MAKEPED will convert it to a number)
- an individual identification number, or id
- father’s id number
- mother’s id number
- first offspring id number \*
- next paternal sibling id number \*
- next maternal sibling id number \*
- sex
- “proband status” \*

\*) not required in original pedigree file. These items will be inserted by the MAKEPED program.

The first offspring can be any of an individual’s children, but the next sib id’s for the offspring will be constrained by this choice. The next-paternal-sibling and next-maternal-sibling numbers, along with first-offspring number, provide a set of pointers to pass from one child to the next. The first offspring of the father is any of his children; the next paternal sib of the first offspring is any other of his children, etc. The entry for the next paternal sib of the last child is “0”. Similar pointers are made for the mother’s children. For full-sibs, it is convenient to make id’s for the next maternal and next paternal siblings identical, but when one or both parents have children from different marriages, at least some will have different values.

Father and mother id’s are 0 for founders, or other members of the pedigree for whom information on parents is absent.

Otherwise, both parents must be present in the pedigree even if one is unknown. If one parent is unknown, an id number must still be created, and a record for the fictitious parent must appear in the pedigree file.

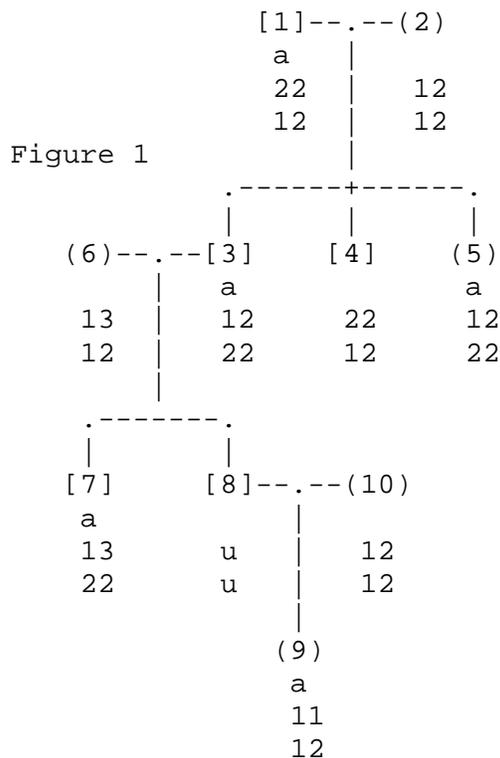
The “proband” refers to a starting individual for linkage calculations (indicated by a 1 in the proband field). The choice of the proband is not necessarily related to the ascertainment of the pedigree; indeed, it is usually more efficient to calculate from a founding ancestor rather than from the true proband. Risks are also calculated for the person designated as the proband. Other individuals should have a 0 in the proband field, except in pedigrees containing inbreeding or marriage loops as discussed in section 2.8. If no proband is designated, the first individual encountered for a pedigree will be used as a starting point for the calculation.

The sex field is coded 1 for males and 2 for females. These default values can be changed by modifying a program constant.

The creation of the first offspring, next maternal sib and next paternal sib pointers is done automatically by the program MAKEPED (see <http://www.jurgott.org/linkage/LinkagePC.html>). The input to the MAKEPED program is a file with individual records containing the pedigree number (or name), id number (or name), father id, mother id, sex and phenotypic data. The phenotypic data are coded as discussed above (sections 2.1 thru 2.5) and in the following examples. The MAKEPED program also allows automatic selection of probands for efficient likelihood calculations.

### **Example**

Consider the pedigree shown in Figure 1. Data on three loci are presented: one disease locus and two marker loci. An “a” below an individual stands for “affected,” and “u” stands for unknown. The first marker locus has three alleles present in the pedigree, while the second has two alleles present.



The input PEDFILE (after MAKEPED) can take the following form:

```

1  1  0  0  3  0  0  1  1  2  0  1  0  1  1  Ped: 1  Per:  1
1  2  0  0  3  0  0  2  0  1  1  1  0  1  1  Ped: 1  Per:  2
1  3  1  2  7  4  4  1  0  2  1  1  0  0  1  Ped: 1  Per:  3
1  4  1  2  0  5  5  1  0  1  0  1  0  1  1  Ped: 1  Per:  4
1  5  1  2  0  0  0  2  0  2  1  1  0  0  1  Ped: 1  Per:  5
1  6  0  0  7  0  0  2  0  1  1  0  1  1  1  Ped: 1  Per:  6
1  7  3  6  0  8  8  1  0  2  1  0  1  0  1  Ped: 1  Per:  7
1  8  3  6  9  0  0  1  0  1  0  0  0  0  0  Ped: 1  Per:  8
1  9  8 10  0  0  0  2  0  2  1  0  0  1  1  Ped: 1  Per:  9
1 10  0  0  9  0  0  2  0  1  1  1  0  1  1  Ped: 1  Per: 10
  
```

The PEDFILE has been produced from an input file by the MAKEPED program. Comments at the end of each record (not present above) indicate original pedigree and id codes (see documentation for MAKEPED, <http://www.jurgott.org/linkage/LinkagePC.html>). The first entry in each record is followed by the pedigree number, id number, five pedigree pointers (father id, mother id, first offspring id, next paternal sib id, next maternal sib id), sex, proband, disease status, and marker loci coded as binary factors. Comparison with the original pedigree will reveal the coding scheme. Individual 1 has been chosen as the “proband;” as this is the first individual of this pedigree encountered in the file, the entry in the proband field is optional.

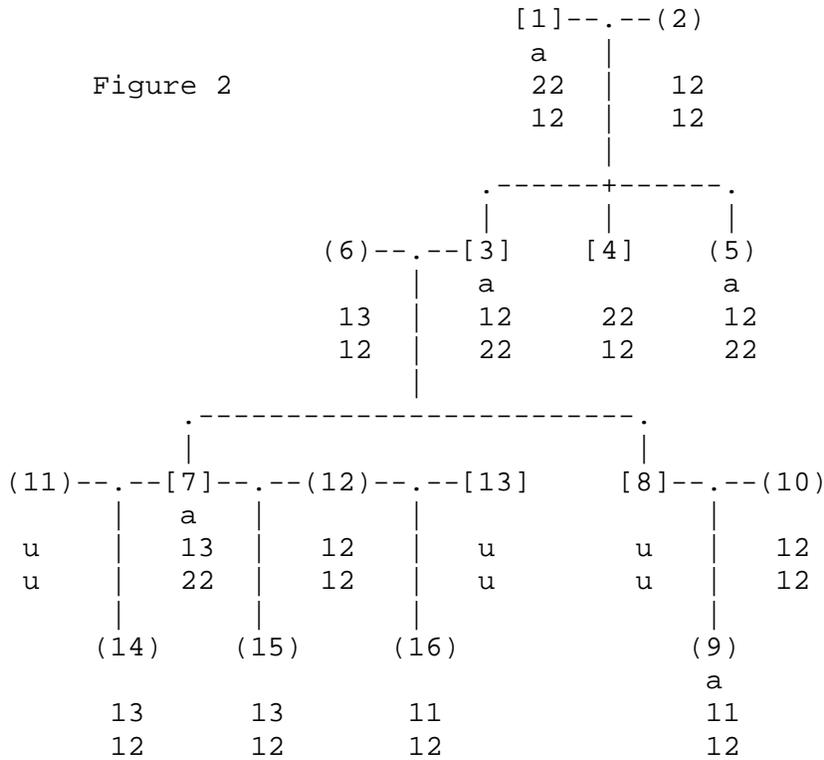
The original pedigree data, before processing by the MAKEPED program and with *Numbered Alleles* coding for the markers, may look as follows:

```

1 1 0 0 1 2 2 2 1 2
1 2 0 0 2 1 1 2 1 2
1 3 1 2 1 2 1 2 2 2
1 4 1 2 1 1 2 2 1 2
1 5 1 2 2 2 1 2 2 2
1 6 0 0 2 1 1 3 1 2
1 7 3 6 1 2 1 3 2 2
1 8 3 6 1 1 0 0 0 0
1 9 8 10 2 2 1 1 1 2
1 10 0 0 2 1 1 2 1 2

```

Now consider the same pedigree extended to include some half sibs (Figure 2).



The MAKEPED program produces the following file for this pedigree (figure 2):

```

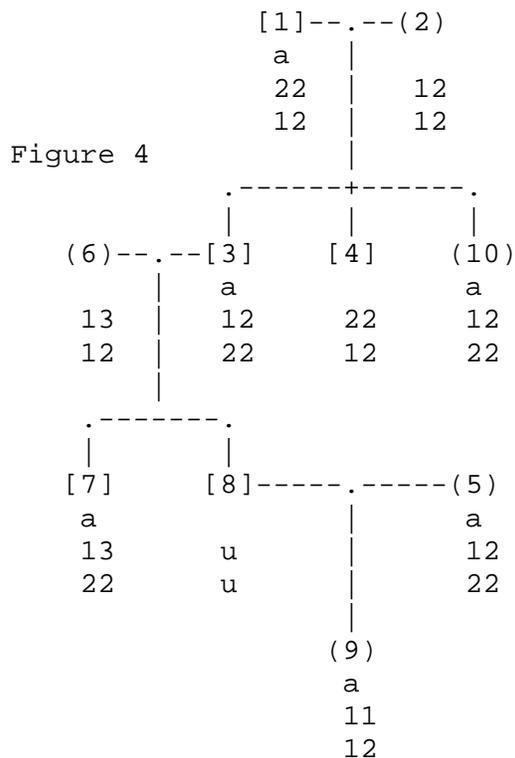
1 1 0 0 3 0 0 1 1 2 0 1 0 1 1 Ped: 1 Per: 1
1 2 0 0 3 0 0 2 0 1 1 1 0 1 1 Ped: 1 Per: 2
1 3 1 2 7 4 4 1 0 2 1 1 0 0 1 Ped: 1 Per: 3
1 4 1 2 0 5 5 1 0 1 0 1 0 1 1 Ped: 1 Per: 4
1 5 1 2 0 0 0 2 0 2 1 1 0 0 1 Ped: 1 Per: 5
1 6 0 0 7 0 0 2 0 1 1 0 1 1 1 Ped: 1 Per: 6
1 7 3 6 14 8 8 1 0 2 1 0 1 0 1 Ped: 1 Per: 7
1 8 3 6 9 0 0 1 0 1 0 0 0 0 0 Ped: 1 Per: 8
1 9 8 10 0 0 0 2 0 2 1 0 0 1 1 Ped: 1 Per: 9
1 10 0 0 9 0 0 2 0 1 1 1 0 1 1 Ped: 1 Per:10
1 11 0 0 14 0 0 2 0 1 0 0 0 0 0 Ped: 1 Per:11
1 12 0 0 15 0 0 2 0 1 1 1 0 1 1 Ped: 1 Per:12
1 13 0 0 16 0 0 1 0 1 0 0 0 0 0 Ped: 1 Per:13
1 14 7 11 0 15 0 2 0 1 1 0 1 1 1 Ped: 1 Per:14
1 15 7 12 0 0 16 2 0 1 1 0 1 1 1 Ped: 1 Per:15
1 16 13 12 0 0 0 2 0 1 1 0 0 1 1 Ped: 1 Per:16

```

The following data refer to a larger pedigree, taken from a coronary heart disease study, in PEDFILE form:

1	1	0	0	3	0	0	2	0	2	3	0	0	0.00	Ped:	1	Per:	1
1	2	0	0	3	0	0	1	1	2	3	0	0	0.00	Ped:	1	Per:	2
1	3	2	1	7	5	5	1	0	2	2	0	0	0.00	Ped:	1	Per:	3
1	4	0	0	7	0	0	2	0	1	2	0	0	0.00	Ped:	1	Per:	4
1	5	2	1	21	0	0	2	0	1	3	0	1	22.70	Ped:	1	Per:	5
1	6	0	0	21	0	0	1	0	2	3	0	0	0.00	Ped:	1	Per:	6
1	7	3	4	26	9	9	1	0	2	2	0	0	0.00	Ped:	1	Per:	7
1	8	0	0	26	0	0	2	0	1	2	0	1	9.20	Ped:	1	Per:	8
1	9	3	4	31	11	11	1	0	2	2	1	1	24.30	Ped:	1	Per:	9
1	10	0	0	31	0	0	2	0	1	2	1	0	9.30	Ped:	1	Per:	10
1	11	3	4	0	12	12	1	0	2	2	1	1	23.90	Ped:	1	Per:	11
1	12	3	4	34	14	14	1	0	2	2	1	1	20.70	Ped:	1	Per:	12
1	13	0	0	34	0	0	2	0	1	2	1	0	14.50	Ped:	1	Per:	13
1	14	3	4	0	15	15	2	0	1	2	1	0	2.10	Ped:	1	Per:	14
1	15	3	4	40	17	17	1	0	2	2	0	0	0.00	Ped:	1	Per:	15
1	16	0	0	40	0	0	2	0	1	2	1	1	9.80	Ped:	1	Per:	16
1	17	3	4	43	19	19	1	0	2	2	0	0	0.00	Ped:	1	Per:	17
1	18	0	0	43	0	0	2	0	1	2	1	0	11.50	Ped:	1	Per:	18
1	19	3	4	0	0	0	1	0	1	2	1	0	9.20	Ped:	1	Per:	19
1	20	0	0	47	0	0	2	0	0	1	0	0	0.00	Ped:	1	Per:	20
1	21	6	5	47	22	22	1	0	2	2	0	0	0.00	Ped:	1	Per:	21
1	22	6	5	48	24	24	1	0	2	2	0	0	0.00	Ped:	1	Per:	22
1	23	0	0	48	0	0	2	0	1	2	1	0	13.40	Ped:	1	Per:	23
1	24	6	5	0	25	25	2	0	1	2	1	1	10.40	Ped:	1	Per:	24
1	25	6	5	0	0	0	2	0	1	2	1	1	9.90	Ped:	1	Per:	25
1	26	7	8	0	27	27	2	0	1	2	1	1	16.80	Ped:	1	Per:	26
1	27	7	8	53	29	29	2	0	1	2	0	1	30.10	Ped:	1	Per:	27
1	28	0	0	53	0	0	1	0	1	2	1	0	6.90	Ped:	1	Per:	28
1	29	7	8	56	0	0	2	0	1	2	1	1	15.40	Ped:	1	Per:	29
1	30	0	0	56	0	0	1	0	1	2	1	0	14.30	Ped:	1	Per:	30
1	31	9	10	0	32	32	2	0	1	1	1	0	6.80	Ped:	1	Per:	31
1	32	9	10	0	33	33	1	0	1	1	1	0	5.60	Ped:	1	Per:	32
1	33	9	10	0	0	0	2	0	1	1	1	1	31.60	Ped:	1	Per:	33
1	34	12	13	0	35	35	1	0	1	1	1	0	19.40	Ped:	1	Per:	34
1	35	12	13	0	36	36	2	0	1	1	1	1	41.70	Ped:	1	Per:	35
1	36	12	13	0	37	37	1	0	1	1	1	0	20.50	Ped:	1	Per:	36
1	37	12	13	0	38	38	1	0	1	1	1	1	28.40	Ped:	1	Per:	37
1	38	12	13	0	39	39	2	0	1	1	1	0	11.50	Ped:	1	Per:	38
1	39	12	13	0	0	0	2	0	1	1	1	0	21.00	Ped:	1	Per:	39
1	40	15	16	0	41	41	2	0	1	1	1	0	10.50	Ped:	1	Per:	40
1	41	15	16	0	0	0	2	0	1	1	1	0	12.60	Ped:	1	Per:	41
1	42	0	0	52	0	0	1	0	1	1	1	0	11.20	Ped:	1	Per:	42
1	43	17	18	52	44	44	2	0	1	1	1	1	37.20	Ped:	1	Per:	43
1	44	17	18	0	45	45	2	0	1	1	1	0	10.10	Ped:	1	Per:	44
1	45	17	18	0	46	46	1	0	1	1	1	1	34.90	Ped:	1	Per:	45
1	46	17	18	0	0	0	1	0	1	1	1	1	25.30	Ped:	1	Per:	46
1	47	21	20	0	0	0	2	0	1	1	1	1	47.90	Ped:	1	Per:	47
1	48	22	23	0	50	50	2	0	1	1	1	0	14.00	Ped:	1	Per:	48
1	49	0	0	51	0	0	1	0	0	1	0	0	0.00	Ped:	1	Per:	49
1	50	22	23	51	0	0	2	0	1	2	1	1	55.30	Ped:	1	Per:	50
1	51	49	50	0	0	0	2	0	1	1	1	0	13.60	Ped:	1	Per:	51
1	52	42	43	0	0	0	2	0	1	1	1	0	12.50	Ped:	1	Per:	52
1	53	28	27	0	54	54	1	0	1	1	1	1	37.50	Ped:	1	Per:	53
1	54	28	27	0	55	55	1	0	1	1	1	1	14.70	Ped:	1	Per:	54
1	55	28	27	0	0	0	2	0	1	1	1	1	29.90	Ped:	1	Per:	55
1	56	30	29	0	57	57	1	0	1	1	1	0	5.70	Ped:	1	Per:	56
1	57	30	29	0	0	0	2	0	1	1	1	0	8.20	Ped:	1	Per:	57





If we choose to break the loop at individual 5 in figure 3, we replace that pedigree with the one shown in figure 4, where 10 is the duplicate of 5.

The PEDFILE then becomes:

```

1  1  0  0  3  0  0  1  1  2  0  1  0  1  1  Ped: 1  Per:  1
1  2  0  0  3  0  0  2  0  1  1  1  0  1  1  Ped: 1  Per:  2
1  3  1  2  7  4  4  1  0  2  1  1  0  0  1  Ped: 1  Per:  3
1  4  1  2  0  5  5  1  0  1  0  1  0  1  1  Ped: 1  Per:  4
1  5  0  0  9  0  0  2  2  2  1  1  0  0  1  Ped: 1  Per:  5
1 10  1  2  0  0  0  2  2  2  1  1  0  0  1  Ped: 1  Per:  5
1  6  0  0  7  0  0  2  0  1  1  0  1  1  1  Ped: 1  Per:  6
1  7  3  6  0  8  8  1  0  2  1  0  1  0  1  Ped: 1  Per:  7
1  8  3  6  9  0  0  1  0  1  0  0  0  0  0  Ped: 1  Per:  8
1  9  8  5  0  0  0  2  0  2  1  0  0  1  1  Ped: 1  Per:  9
  
```

where 2 is entered into the proband field for both 5 and 10. When the loop is broken at the “proband,” a 1 is entered into the proband field for one of the duplicates, e.g.:

```

1  1  0  0  3  0  0  1  0  2  0  1  0  1  1  Ped: 1  Per:  1
1  2  0  0  3  0  0  2  0  1  1  1  0  1  1  Ped: 1  Per:  2
1  3  1  2  7  4  4  1  0  2  1  1  0  0  1  Ped: 1  Per:  3
1  4  1  2  0  5  5  1  0  1  0  1  0  1  1  Ped: 1  Per:  4
1  5  0  0  9  0  0  2  1  2  1  1  0  0  1  Ped: 1  Per:  5
1 10  1  2  0  0  0  2  2  2  1  1  0  0  1  Ped: 1  Per:  5
1  6  0  0  7  0  0  2  0  1  1  0  1  1  1  Ped: 1  Per:  6
1  7  3  6  0  8  8  1  0  2  1  0  1  0  1  Ped: 1  Per:  7
1  8  3  6  9  0  0  1  0  1  0  0  0  0  0  Ped: 1  Per:  8
1  9  8  5  0  0  0  2  0  2  1  0  0  1  1  Ped: 1  Per:  9
  
```

## Chapter 3: Analysis Programs

### 3.0 OVERVIEW

The analysis programs fall into two groups. Programs in the first group (ILINK, MLINK, LINKMAP, LODSCORE) are designed for calculations in general pedigrees. Those in the second group (CILINK, CMAP) are optimized for three-generational pedigrees with codominant markers; they cannot be used for general pedigrees or for disease loci. CILINK and CMAP are the equivalent of ILINK and LINKMAP for three-generational pedigrees. (The "C" preface refers to the CEPH reference panel of families which has this three-generation structure.)

Each analysis program is described in a separate section in this chapter. Auxiliary programs for preprocessing of the input data are described in Chapter 4. The calling sequence, i.e. the order in which the auxiliary and analysis programs are invoked, is given under the program description.

The programs contain certain constants that establish limits on the number of loci, alleles, etc. that can be analyzed. These constants can be changed, and the programs recompiled if larger values are required. Important constants that recur in several programs are described in section 3.1.

### 3.1 CONSTANTS

In the program, constants are set for routine linkage problems. It may be necessary to increase some constants for specific problems, or to decrease others to minimize memory usage on some computers. All the constants used in a program are declared at the start of the code. The meaning of most constants is easy to interpret from information given with the declaration. Values for two constants, MAXNEED and MAXCENSOR, cannot be determined prior to running programs. When one of the analytic programs terminates in an error, one of these constants may be too small.

All the programs contain a boolean constant DOSTREAM. This should be set to TRUE for use with the control program shell.

#### Constants for ILINK, MLINK, and LINKMAP

Several of the constants that are common to the programs for calculations in general pedigrees are related to the maximum number of alleles and loci to be considered in multilocus runs. These are:

maxlocus	{ MAXIMUM NUMBER OF LOCI }
maxall	{ MAXIMUM NUMBER OF ALLELES AT A SINGLE LOCUS }
maxhap	{ MAXIMUM NUMBER OF HAPLOTYPES }

The minimum value of MAXHAP required is the product of the maximum number of alleles at each locus.

Pedigrees, families and individuals are constrained by the following constants:

maxind	{ MAXIMUM NUMBER OF INDIVIDUALS IN ALL PEDIGREES }
maxped	{ MAXIMUM NUMBER OF PEDIGREES }
maxchild	{ MAXIMUM NUMBER OF FULLSIBS IN A SIBSHIP }

The constants controlling default values for phenotype codes are:

affall=2;	{ DISEASE ALLELE FOR QUANTITATIVE TRAITS OR AFFECTION STATUS }
	{ QUANTITATIVE TRAIT }

```

maxtrait;      { MAXIMUM NUMBER OF QUANTITATIVE VARIABLES AT A SINGLE LOCUS }
missval=0.0;  { MISSING VALUES FOR QUANTITATIVE TRAITS }

{ AFFECTION STATUS }

missaff=0;    { MISSING VALUE FOR AFFECTION STATUS }
affval = 2;   { CODE FOR AFFECTED INDIVIDUAL }
maxliab;     { MAXIMUM NUMBER OF LIABILITY CLASSES }

{ BINARY (FACTOR UNION) SYSTEM }
maxfact;     { MAXIMUM NUMBER OF BINARY CODES AT A SINGLE LOCUS }

```

MAXTRAIT, MAXLIAB and MAXFACT can be varied to meet the requirements for various problems. The other default values should not be modified unless absolutely necessary. Modification of the default values may introduce problems of compatibility when data are transferred between program versions or installation sites.

MAXNEED and MAXCENSOR are variables that are difficult to determine prior to running the program. Probabilities of various recombination classes are stored in an array dimensioned by MAXNEED. If you compile the program with a large value of MAXNEED (e.g. 1000), a message giving the optimal value will be printed if all the probabilities are successfully stored within this limit. If MAXNEED is too small the program will terminate with an error message.

MAXCENSOR dimensions an array that increases the efficiency of calculations. Small values will not cause a runtime error, but may increase computation times. The program will give a message to help optimize the choice of this value. The constant MININT is used with MAXCENSOR; it should be assigned the minimum value supported by the compiler.

MINFREQ is another constant that can improve the efficiency of calculations when dominant or codominant loci are being analyzed. Rare homozygotes will not be considered in the calculations if the gene frequency is less than MINFREQ. Heterozygote × heterozygote matings will also be excluded from the calculations in this case. For analyzing recessive traits, or pedigrees in which heterozygote × heterozygote matings occur, you should declare MINFREQ = 0.0.

Likelihood values may underflow in large pedigrees. Scaling factors are used to avoid this:

```

scale          { SCALE FACTOR }
scalemult      { SCALE WEIGHT FOR EACH LOCUS }

```

The values of SCALE and SCALEMULT can be increased (decreased) if underflow (overflow) occurs. The suggested default values are SCALE = 2.0 and SCALEMULT = 3.0. If overflow occurs, SCALEMULT should be reduced to 2.0. To correct problems of underflow, try increasing SCALE to 3.0. Further modifications should be tried if these values do not correct the problems.

Underflow is often not detected, but may result in the likelihood becoming zero. The logarithm of the likelihood is then replaced by:

```

zerolike = -1.0E20; {FOR INCONSISTENT DATA OR RECOMBINATION}

```

resulting in the extreme negative values for the sum of the logarithms over all pedigrees. Such a result may also arise from errors in the data entry, so if the problem persists despite repeated modification of SCALE and SCALEMULT, please check your pedigree and genotypes carefully.

### **Constants for CILINK and CMAP**

Prior to running CMAP and CILINK, the data are transformed by the CFACTOR program (see description Section 4.2 CFACTOR). This program creates new “families” which contain fewer loci than the original but have the same total likelihood. MAXLOCUS is the maximum number of loci prior to data transformation. MAXSYSTEM, MAXRECTYPE,

MAXALL, MAXIND and MAXPED are maximum values after transformation:

{ SEE THE OUTPUT FROM CFACTOR TO DETERMINE THE FOLLOWING }

maxlocus        { MAXIMUM NUMBER OF LOCI IN MAPPING PROBLEM }

{ THE FOLLOWING REFER TO VALUES AFTER TRANSFORMATION }

maxsystem      {MAXIMUM NUMBER OF LOCI IN ONE FAMILY AFTER TRANSFORMATION}

maxrectype     { MAXIMUM NUMBER OF DIFFERENT RECOMBINATION PATTERNS }

maxall         { MAXIMUM NUMBER OF ALLELES AT A SINGLE LOCUS }

maxind         { MAXIMUM NUMBER OF INDIVIDUALS }

maxped         { MAXIMUM NUMBER OF PEDIGREES }

maxfact        { BINARY (FACTOR UNION) SYSTEM }

### 3.2 *ILINK*

#### Purpose

ILINK is a program for maximum likelihood estimation of recombination fractions for an arbitrary number of marker and disease loci. For two loci, the program determines the maximum lod score in addition to the recombination estimate. Sex-specific differences in the recombination rates can be incorporated as described in Chapter 2. ILINK can also estimate penetrance, gene frequencies and other parameters.

#### Using the Program

ILINK is used with the accompanying program UNKNOWN. The calling order is:

```
UNKNOWN
ILINK
```

The input files for this suite of programs are:

```
PEDFILE
DATAFILE
```

The output files are:

```
FINAL
OUTFILE
STREAM
```

UNKNOWN produces temporary files called IPEDFILE and SPEEDFILE; along with DATAFILE, these serve as input for ILINK.

#### Program Constants Specific to ILINK

The following constants can be modified for various purposes:

```
fitmodel = false;        { TRUE IF ESTIMATING PARAMETERS OTHER THAN RECOMBINATION }
dostream = true;        { STREAM FILE OUTPUT }
byfamily = false;       { GIVE LOD SCORES BY FAMILY IN FINAL }
```

```
{ GRADIENT APPROXIMATIONS }
```

```
approximate = true;
epsilon = 1.0E-3;
```

```
{ GEMINI }
maxn = 20;           { MAXIMUM NUMBER OF ITERATED PARAMETERS }
```

### **Datafile Structure**

The program-specific parts of DATAFILE consist of two lines. The first contains a number that indicates a locus (iterated locus) for which parameters, such as gene frequency or penetrance, can be estimated. The locus number is given in phenotype order.

The second line consists of a list of zeros and ones (binary list) to indicate parameters that are to be estimated or fixed. If a 1 is entered in a given location, the parameter corresponding to that location is estimated (iterated parameter); if the entry is zero, the corresponding parameter is fixed at the initial value specified in the DATAFILE (non-iterated parameter).

### **Specifications for Estimating Recombination Rates**

For  $n$  loci, the first  $n-1$  locations in the list correspond to the recombination rates between adjacent loci. In most applications, estimates of these recombination rates will be made with other parameters held fixed. In this case, any of the locus numbers can be used for the iterated locus (the value must be between 0 and  $n$ , where 0 indicates that none of the loci can have iterated parameters). For example, if 1 is chosen for the iterated locus and  $n$  is 3, the two lines to add to the DATAFILE are:

```
1      << iterated locus
1 1
```

The last line must end directly after the specification of the last iterated or non-iterated parameter, without trailing blanks. This end-of-line delimitator tells the program that only recombination fractions will be estimated.

Sometimes it is useful to fix the value of one of the recombination rates; for example, if the first two loci are known to be completely linked we might wish to fix the first recombination rate to 0.0 while estimating the second rate. In this case, the last two lines are:

```
1      << iterated locus
0 1
```

With sex-specific recombination rates, the number of parameters is increased by 1 (to  $n$ ) when assuming a constant ratio of female/male genetic distances, or by  $n-1$  (to  $2n-2$ ) when estimating different male and female recombination fractions in each interval. For the former, the last two lines are:

```
1      << iterated locus
1 1 1
```

and for the latter, they are:

```
1      << iterated locus
1 1 1 1
```

With three loci, ILINK supports interference. When a mapping function is used, two iterated parameters are specified for each sex. Without a mapping function, three recombination rates can be estimated for each sex, and the number of iterated

parameters should be adjusted accordingly.

### **Specifications for Estimating Other Parameters**

All locus types support the estimation of gene frequencies; these add an additional *n*allele-1 parameters to the list of iterated or non-iterated values. If the iterated locus is of the affection-status or quantitative type, other parameters can also be estimated.

The list of iterated parameters has the following orders for the four types of loci:

- Numbered alleles or binary factors: The order is recombination fractions (and female/male genetic distance under the sex-difference option 1) followed by *n*allele-1 gene frequencies.
- Affection status: The order is recombination fractions (and female/male genetic distance under the sex-difference option 1); *n*allele-1 gene frequencies; penetrance for each genotype in each of liability class [*n*liability × *n*allele × (*n*allele+1)/2]; an additional penetrance for each allele for sex-linked loci.
- Quantitative variables: The order is recombination fractions (and female/male genetic distance under the sex-difference option 1); *n*allele-1 gene frequencies, means for each of *n*trait quantitative traits (*n*trait × *n*allele); and the upper-triangle of the variance covariance matrix [*n*trait × (*n*trait-1)/2]. Within the program, quantitative variables are restricted to two alleles at a locus. The genotype means are transformed to the mean of first homozygote, displacement between the first and the second homozygote mean, and the dominance (ratio of the difference between heterozygotes and first homozygotes means to the displacement).

### **Gradient Approximation**

Approximations to the gradient are controlled by the boolean constant APPROXIMATE and the value of the constant EPSILON. The approximation applies only to the calculation of the gradient prior to a line search in the numerical estimation procedure. In a variety of examples, EPSILON = 0.00001 has been found satisfactory. To assure a maximum gain in efficiency a pedigree with little genotypic information should be selected for the proband.

### **3.3 MLINK**

#### **Purpose**

MLINK is a program for calculation of lod scores and risk with two or more loci. Typically, two loci will be used for lod score calculations. Sometimes, however, it is useful to consider several completely linked marker loci with a disease locus when calculating lod scores.

#### **Using the Program**

MLINK is used with the accompanying program UNKNOWN. The calling order is:

```
UNKNOWN  
MLINK
```

The input files for this suite of programs are:

```
PEDFILE  
DATAFILE
```

The output files are:

OUTFILE  
STREAM

UNKNOWN produces temporary files called IPEDFILE and SPEEDFILE; along with DATAFILE, these serve as input for MLINK.

### **Program Constants Specific to MLINK**

If the program constant SCORE is set to “true,” the program calculates lod scores; otherwise only the likelihood values are given. MLINK is distributed with SCORE set to “true.”

### **Datafile Structure**

The program-specific part of DATAFILE consists of a single line that contains the number of the recombination fraction to be varied, an increment for the recombination fraction, and a stopping value. The likelihood is evaluated for the initial recombination values, then the designated value is incremented and the likelihood recalculated if the incremented value is less than the final value. The incremental calculations are continued until the designated recombination is greater than the final value.

As an example, suppose that lod scores are calculated for two loci. If the following is the last line in DATAFILE:

```
1 0.1 0.5
```

the program will start with the initial recombination value, specified in DATAFILE, and calculate with increments of 0.1 until 0.5 is surpassed. To calculate for increments of 0.01 stopping at 0.2, this line should be:

```
1 0.01 0.2
```

For three loci, with the first two loci in the chromosome order completely linked, a lod score for linkage with the third locus could be calculated with MLINK at increments of 0.05 with the following line in DATAFILE:

```
1 0.05 0.5
```

Often different increments are desired in different regions of the lod score table. This refinement can be achieved in MLINK by adding additional lines at the end of DATAFILE. Each line specifies a new starting recombination fraction, increment and final value; the recombination to be varied is the same as previously designated. Thus, the following two lines:

```
1 0.05 0.2  
0.2 0.1 0.5
```

calculates the likelihood with steps of 0.05 until reaching 0.2, followed by steps of 0.1 until reaching 0.5.

### **Program Options**

If program constant SCORE is “true,” the initial evaluation is made with the recombination to be varied at 0.5. For two-locus analysis, lod scores are calculated thereafter at each evaluation of the likelihood. With three or more loci, the 2 ln likelihood difference is calculated in place of the lod score. When the constant BYFAMILY is “true,” the likelihood values are given for each family.

### **Comments**

MLINK does not support sex-differences in the recombination rates. Sex-differences can be included but only the male recombination rate will be incremented.

MLINK allows the interference option 2 (user defined mapping function) but not option 1 (interference without a mapping function).

### **3.4 LINKMAP**

#### **Purpose**

LINKMAP is a program for calculating location scores of one locus against a fixed map of other loci.

#### **Using the Program**

LINKMAP is used with the program UNKNOWN. The calling order is:

```
UNKNOWN  
LINKMAP
```

The input files for this suite of programs are:

```
PEDFILE  
DATAFILE
```

The output files are:

```
OUTFILE  
STREAM
```

UNKNOWN produces temporary files called IPEDFILE and SPEEDFILE; along with DATAFILE, these serve as input for LINKMAP.

#### **Program Constants Specific to LINKMAP**

These constants are the same as in MLINK.

#### **Datafile Structure**

The program-specific part of DATAFILE consists of a single line that contains the locus number (input order) to be varied (moving locus); a stopping value for the recombination fraction; and the number of points to be evaluated in an interval (gridsize). Evaluations are made by moving the "locus varied" within the interval between flanking markers, or to one or the other of the ending intervals in the map. The interval between the starting and ending recombination values is divided into equal portions so that the total number of evaluations is equal to gridsize. The program must be restarted with a different chromosome order to test other intervals.

For the outside interval on the left hand side of the map, the recombination will decrease automatically from the initial value to the final value; therefore, the final value should be smaller than the initial value. For other intervals, the recombination rate will increase from the starting value to the final value. In interior intervals, the recombination between the flanking markers is held constant (assuming Haldane's mapping function). If the final recombination rate specified in DATAFILE is greater than the total recombination in the interval, the latter figure determines the division.

As an example, consider a map of four fixed loci, with a fifth locus whose position is varied. The DATAFILE could have

the form:

```
5 0 0
0 0.0 0.0 0
5 3 2 4 1    << locus order
```

(locus description)

```
0 0          << sex-difference, interference
0.5 0.1 0.05 0.15
5 0.0 10
```

The program will make 10 likelihood evaluations at equally spaced intervals, with locus 5 moving from 0.5 to 0.0 recombination with locus 3. Similarly:

```
5 0 0
0 0.0 0.0 0
3 2 5 4 1    << locus order
```

(loci description)

```
0 0          << sex-difference, interference
0.1 0.0 0.05 0.15
5 0.05 10
```

will result in 10 likelihood evaluations starting with 0.0 recombination between loci 5 and 2, and ending with 0.0 recombination between loci 5 and 4. (As a consequence of rounding error, the final recombination may not be exactly 0.0.)

### **Program Options**

When the constant BYFAMILY is “true”, likelihoods are given for each family in OUTFILE.

### **Comments**

LINKMAP permits the sex-difference option 1 (constant ratio of female/male genetic distance). The option 2 (generalized sex-difference) cannot be used when calculating location scores.

Interference is not allowed with the LINKMAP program.

### ***3.5 CILINK***

#### **Purpose**

The program CILINK performs the same function as ILINK but is restricted to three-generation pedigrees and codominant marker loci. Only recombination rates, and the female/male genetic distance, can be iterated.

#### **Using the Program**

CILINK functions with the program CFACTOR. The calling order is:

```
CFACTOR
CILINK
```

The input files for this suite of programs are:

PEDFILE  
DATAFILE

The output files are:

FINAL  
OUTFILE  
STREAM

TEMPPED and TEMPDAT are temporary files produced by CFACTOR for input to CILINK.

### **Program Constants Specific to CILINK**

CILINK contains many of the program constants that ILINK does for controlling numerical maximization.

### **Datafile Structure**

The program code for CILINK is 1. Otherwise, the DATAFILE format is the same as that for the ILINK program except that all loci must be codominant. Either allele numbers or binary codes can be used to describe the loci. Only recombination rates, and the female/male genetic distance ratio can be estimated.

### **Comments**

Recoding reduces the number of alleles at each locus; hence CILINK uses arbitrary gene frequencies. This can affect the likelihood values slightly when some parent or grandparent genotypes are missing.

## ***3.6 CMAP***

### **Purpose**

The program CMAP performs the same function as LINKMAP but only in three-generation pedigrees with codominant marker loci.

### **Using the Program**

CMAP functions with the program CFACTOR. The calling order is:

CFACTOR  
CMAP

The input files for this suite of programs are:

PEDFILE  
DATAFILE

The output files are:

OUTFILE  
STREAM

TEMPPED and TEMPDAT are temporary files produced by CFACTOR for input to CMAP.

### **Program Constants Specific to CMAP**

See LINKMAP (section 3.4).

### **Datafile Structure**

Except for the program code, which is 2 for CMAP, DATAFILE has the LINKMAP format.

### **Program Options**

When the constant BYFAMILY is “true,” likelihoods are given for each family in OUTFILE.

### **Comments**

Recoding is used to reduce the number of alleles at each locus; see the comments in section 3.5.

### ***3.7 LODSCORE***

LODSCORE is a program for maximum likelihood estimation of the recombination rate and the maximum lodscore, from two-locus data. In the present implementation, it is almost identical to ILINK.

### **Datafile structure**

In contrast to ILINK, for which the program-specific parts of the DATAFILE consist of two lines, LODSCORE has four lines. The first two program specific lines are the same as for ILINK, whereas the last two list the two sets of loci (set 1 on line 3, set 2 on line 4) between which comparisons should be calculated.

### ***3.8 CLODScore***

CLODScore is analogous to LODSCORE except that it is restricted to three-generation pedigrees and codominant marker loci.

## **Chapter 4: Auxiliary Programs**

### ***4.0 OVERVIEW***

Several auxiliary programs are required for preprocessing data prior to invoking an analysis program. The calling order for the auxiliary and analysis programs are explained in Chapter 3. A brief description of each of the auxiliary programs follows.

### ***4.1 UNKNOWN***

UNKNOWN infers possible genotypes and mating combinations for parents with unknown genotypes for ILINK, MLINK and LINKMAP. The following constants must be set prior to compiling the program:

```
maxlocus      { MAXIMUM NUMBER OF LOCI }
maxfem        { MAXIMUM NUMBER OF SINGLE LOCUS GENOTYPES FOR A FEMALE }
maxmal        { MAXIMUM NUMBER OF SINGLE LOCUS GENOTYPES FOR A MALE }
maxall        { MAXIMUM NUMBER OF ALLELES AT A SINGLE LOCUS }
maxind        { MAXIMUM NUMBER OF INDIVIDUALS IN ONE PEDIGREE }
maxmarriage   { MAXIMUM NUMBER OF MARRIAGES FOR ONE MALE }
```

```
{ BINARY (FACTOR UNION) SYSTEM }
```

```
maxfact       { MAXIMUM NUMBER OF BINARY CODES AT A SINGLE LOCUS }
```

```
{ QUANTITATIVE TRAIT }
```

```

maxtrait      { MAXIMUM NUMBER OF QUANTITATIVE FACTORS AT A SINGLE LOCUS }
missval = 0.0; { MISSING VALUES FOR QUANTITATIVE TRAITS }
affal = 2;    { ALLELE GIVING AFFVAL FOR QUANTIT. TRAITS IN MALES, SEXLINK }

{ AFFECTION STATUS }

missaff = 0;  { MISSING VALUE FOR AFFECTION STATUS }
affval = 2;   { CODE FOR AFFECTED INDIVIDUAL }
maxliab      { MAXIMUM NUMBER OF LIABILITY CLASSES }

{ OTHERS }
maxn         { MAXIMUM NUMBER OF ITERATED PARAMETERS IN ILINK }

```

MAXFEM need be no larger than  $nallele \times (nallele+1)/2$  where *nallele* is the actual number of alleles at a locus. MAXMAL can be equal to MAXFEM, even for sex-linked traits without loss of efficiency.

The following constants should not be changed:

```

doeliminate = FALSE; { ELIMINATE SOME ALLELES IN UNKNOWNNS }
minfreq = 0.05;     { MINIMUM FREQUENCY FOR ELIMINATION }

```

UNKNOWN produces IPEDFILE and SPEEDFILE as output files. DATAFILE and PEDFILE are required as input.

## 4.2 CFACTOR

CFACTOR transforms data before CMAP or CILINK is invoked. Several program constants allow selection of undocumented features. The only constants subject to change by the user are:

```

maxlocus      { MAXIMUM NUMBER OF LOCI }
maxall        { MAXIMUM NUMBER OF ALLELES AT A SINGLE LOCUS }
maxind        { MAXIMUM NUMBER OF INDIVIDUALS }
maxped        { MAXIMUM NUMBER OF PEDIGREES AFTER FACTORIZATION }

{ BINARY (FACTOR UNION) SYSTEM }

Maxfact       { MAXIMUM NUMBER OF BINARY CODES AT A SINGLE LOCUS }
elimu=false;  { Eliminate locus when a parent cannot be inferred }
elimb=false;  { Eliminate locus when neither parent can be inferred }
esame=false;  { Eliminate locus when parents could be homozygote }

{ OTHERS }

maxn          { MAXIMUM NUMBER OF ITERATED PARAMETERS }

```

MAXPED cannot be determined before CFACTOR is applied to the data, but twice the actual number of pedigrees is probably a safe minimum value.

CFACTOR takes IPEDFILE and DATAFILE as input, and produces TEMPPED and TEMPDAT. When CFACTOR finishes it displays in a box on screen the (new) number of pedigrees and total number of individuals it wrote into the TEMPPED file. To read this information you may have to temporarily stop the PEDIN batch file (using Ctrl-S, and use Ctrl-Q to get it going again). Alternatively, insert a PAUSE statement after each occurrence of the CFACTOR statement such that the program stops after CFACTOR has executed.