UNPHASED

Version 3.0.7

User guide

by

Frank Dudbridge
MRC Biostatistics Unit
Cambridge, UK

March 2007

Contents

Introduction

UNPHASED is an application for performing genetic association analysis in nuclear families and unrelated subjects. It implements maximum-likelihood inference on haplotype and genotype effects while allowing for missing data such as uncertain phase and missing genotypes. Many of the commonly performed analyses are provided including transmission/disequilbrium tests, global and individual tests for haplotypes, tests that account for associations of nearby loci, tests of gene-gene interaction, adjustments for environmental covariates, genotype tests, comparison of risk between different haplotypes, and permutation tests.

If you use UNPHASED for a publication, please acknowledge my work. The reference for the last version of UNPHASED is

Dudbridge F (2003) Pedigree disequilibrium tests for multilocus haplotypes. *Genet Epidemiol* 25:115-21

However this paper is out-of-date regarding the current methodology. A manuscript describing the methods is in preparation; the methods are summarised later in this document, which is available as a technical report from the MRC Biostatistics Unit:

Dudbridge F (2006) UNPHASED user guide. *Technical Report 2006/5,* MRC Biostatistics Unit, Cambridge, UK

UNPHASED is distributed free-of-charge under the GNU public licence. The latest version of UNPHASED is available from

Frank Dudbridge
MRC Biostatistics Unit
Institute for Public Health
Robinson Way
Cambridge
CB2 2SR UK
frank.dudbridge@mrc-bsu.cam.ac.uk
http://www.mrc-bsu.cam.ac.uk/personal/frank/software/unphased/

I maintain an email list for announcing updates, but do not track all downloads. Please email me directly to be added to my update list.

Running UNPHASED

Installation

After downloading the distribution, extract the files to an appropriate directory. On Windows systems, used the **Extract All** facility for zipfiles. On unix systems, type

gunzip –c unphased-xxx.tar.gz | tar xf –

Your browser may have already gunzipped the tarfile, in which case simply type

tar xf unphased-xxx.tar

Three directories will be created: **bin/** containing executable files, **doc/** containing this documentation, and **src/** containing the source code. **unphased.sol** is a Solaris 10 binary, **unphased.lnx** a Linux binary, and **unphased.exe** a Windows binary. To use the graphical user interface, unix users should create a symbolic link called **unphased** pointing to the appropriate binary.

Binaries are provided for your convenience but use them at your own peril. You may find that the program runs faster if you build it on your own system. The easiest way to do so is with a **make** utility. The distributed executables were built using g++ for Solaris and Linux, and mingw for Windows. With mingw, the UnphasedAnalysis.cpp file must be edited to include "timeofday.h" instead of <sys/time.h>. The graphical user interface should work on Java Virtual Machine 1.2 or later.

Starting UNPHASED

There are two ways to run UNPHASED. The graphical user interface (GUI) allows you to select options from menus and hold the results of multiple analyses on screen at the same time. With the command line interface you have to type in all the options on the command line, but this method has greater flexibility, for example for writing shell scripts or piping the output.

To start the GUI from Windows, double click on the **unphased.bat** icon. To start the GUI from a unix system, type **unphased.sh** at the command line, after putting this file in your path, or click on its icon in a file manager. A blank window will then appear with various menus for setting options. After loading the input files using the **File** menu and selecting options, start an analysis by selecting **Run:Start**. A new tab will appear holding the output of this run. The run can be interrupted by pressing control-C. You can edit the output at

any time, save it to a file, and print it. You can have several analyses running at the same time, though this may mean each analysis takes longer.

Selection of options from the menus is intuitive. When you are prompted to enter a haplotype, enter the alleles separated by spaces. If genotype analysis is selected, enter the two alleles at each locus separated by a space, with the lower numbered allele first, and then a space to the next locus, and so on.

The GUI may not know where to find the main executable for UNPHASED, depending on your system configuration. In this case, select **Run:Locate executables** and then navigate to the directory containing the **unphased** or **unphased.exe** file. Double click on that file and the GUI will then remember where it is.

A few options are not directly available in the GUI. These can be invoked by selecting **Run:Command line arguments** and then typing in the command line syntax, as described below.

You can run UNPHASED from the command line in a unix shell or a Windows command prompt. Type **unphased** followed by a list of *options*. Each option consists of a dash followed by the option name. You do not need to type the option name in full, just enough letters to identify the option uniquely. Each option is followed by a list of *arguments*. Depending on the option, there may be no argument, one argument or several arguments. When several arguments are expected, each successive word on the command line is read as an argument until a word beginning with a dash is found, which is then parsed as the next option.

If a new option is expected but the next word does not begin with a dash, that word is interpreted as a pedigree file for input. You can list any number of pedigree files; they will be analysed one after the other in turn. To force UNPHASED to read a pedigree file, used the **–pedfile** option. This is useful if you use a variable argument option such as **–marker** and then want to add another pedigree file.

To send the output to a file instead of the screen, use the **–output** option followed by a file name. Alternatively, unix redirection will work as normal.

Here are a couple of examples of starting UNPHASED from the command line:

unphased mypeds.ped –marker 1 2 3 –missing –permutation 10

This will analyse the first three markers from the pedigree file **mypeds.ped**, estimate missing genotypes, and will run a permutation test with ten replicates.

unphased mypeds.ped –permutation 10 morepeds.ped

This will first analyse the file **mypeds.ped** and then the file **morepeds.ped**. Since the **–permutation** option has one argument, the word after that is interpreted as a pedigree file.

Some arguments take a haplotype as input. If more than one marker is being analysed, specify the haplotype as a list of alleles with a space between each one. If genotype analysis is selected, list the alleles as pairs with the lower value first. So for example

unphased mypeds.ped –window 2 –reference 1 2

will analyse two-locus haplotypes with the 1-2 haplotype as reference, and

unphased mypeds.ped –window 2 –genotype –reference 1 2 1 1

will analyse two-locus genotypes, using as reference the two-locus genotype with 1/2 at the first locus and 1/1 at the second.

Input files

The main input file for UNPHASED is a *pedigree file* containing all the subject trait and genotype data. Two other input files are optional: a *data file* that specifies the format of the pedigree file, and an *option file* that lets you load a predefined set of options.

Pedigree file

The pedigree file contains information on family relationships between study subjects, and all of the genotype and covariate data. UNPHASED uses the *LINKAGE pre-makeped* format, which in its simplest form consists of one line per subject, with columns organised as follows

PedID SubID PaID MaID Sex Trait M1A1 M1A2 M2A1 M2A2 …

where:

PedID is a pedigree identifier which can consist of letters and numbers
SubID is a subject identifier which can consist of letters and numbers
PaID is the identifier of this subject's father, 0 if the father is not in the file
MaID is the identifier of this subject's mother, 0 if the mother is not in the file
Sex is 1 for male subjects, 2 for females
Trait contains the trait information for this subject. For a disease affection status, it is 1 for unaffected, 2 for affected, and 0 for unknown. For quantitative and other traits, the trait value is entered; if the value is unknown, enter a dash -.
M1A1 and **M1A2** are the two alleles for the first marker. They must be numerical but can take any values. Multi-allelic markers are allowed. Missing alleles are coded as 0. For sex-linked markers in males, either enter the one allele followed by 0, or enter the allele as a homozygote genotype.
M2A1 and **M2A2** are the two alleles for the second marker, and so on.

For unrelated subjects, including case/control data, each subject is regarded as a pedigree with one member. The **PedID** must be unique, the **SubID** can have any value, and both **PaID** and **MaID** must be 0.

If the pedigree file contains extended pedigrees, they will be broken up into nuclear families which are then treated as independent.

It is sensible to group members of the same pedigree together in the file, but it is not compulsory. When combining data sets into a single pedigree file, be careful that pedigree identifiers are not duplicated.

As far as possible, UNPHASED is "robust" to errors in the pedigree structure. This means that if a Mendelian inconsistency is encountered, a warning is printed and the pedigree is discarded, but the program continues to run. If a parent is missing, which may seem to happen if there is typo in **PaID** or **MaID**, a dummy parent will be created and no warning will be given. Sometimes this is an advantage, sometimes not. Therefore I strongly recommend that you run your file through PEDCHECK before running UNPHASED. The error checking in UNPHASED is reasonable but not exhaustive.

Data file

The data file, which is optional, specifies the structure of the pedigree file and the names of the traits and markers. If no data file is given, the format is assumed to be as above, and UNPHASED automatically determines whether the trait is binary or quantitative by looking at the values seen in column 6. If a different format is used, for example to include covariates, a data file is needed to define the organisation of columns. UNPHASED recognises two formats: *QTDT* format and *LINKAGE* format. It will automatically recognise which format has been used. In all cases the first five columns of the pedigree file are the same as above.

QTDT format is easier to work with. Each line of the file refers to a column in the pedigree file. The first word defines the type of data and the second gives the name of the item. The type of data must be one of four codes:

M denotes marker genotype data. In this case the item refers to two columns of the pedigree file.
T denotes quantitative trait data.
C denotes covariate data. In UNPHASED this is interchangeable with quantitative trait data.
A denotes affection status data.

A full description of this format is available at
http://www.sph.umich.edu/csg/abecasis/QTDT/docs/data.html
Note that UNPHASED only recognises four codes and use of the **Z** and **S** codes will result in an error.

The following example is adapted from Gonçalo Abecasis' web page referenced above. If the data file is the following

```
T HEIGHT
M GH1_SNP132
M GH1_SNP146
C AGE
```

then the pedigree file might look like this:

```
1000 1 0 0 1  1.87  1 2 2 2  40
1000 2 0 0 2  1.65  1 2 1 1  38
1000 3 1 2 1  1.80  1 1 1 2  20
1000 4 1 2 1  1.75  2 2 1 2  17
```

LINKAGE format is rather more complex, being designed originally for parametric linkage analysis. UNPHASED will insist on all elements of the file being present, but will only use the locus types and locus names, which are specified in the same way as for GENEHUNTER, following a # symbol after the number of alleles. A full description of the format is at
http://linkage.rockefeller.edu/soft/linkage/sec2.6.html
and also at
http://linkage.rockefeller.edu/soft/gh/loadmark.html.

To code either quantitative or covariate data, UNPHASED recognises locus code 4 in the same way as GENEHUNTER, that is a single line containing "4 0" is sufficient. More detail on this is at
http://linkage.rockefeller.edu/soft/gh/vc.html

A LINKAGE datafile for the pedigree file above could then look like this:

```
4 0 0 5 << first header line must have total number of loci
0 0 0 0 << second header line is ignored
1 2 3 4 << third header line can just count the loci
4 0 # HEIGHT
3 2 # GH1_SNP132
0.5 0.5 << allele frequencies, ignored by UNPHASED
3 2 # GH1_SNP146
0.5 0.5
4 0 # AGE
0 0 << interference code, ignored by UNPHASED
0.5 0.5 0.5 << recombination fractions, must be one fewer
than number of loci, but ignored by UNPHASED
1 0.1 0.5 << MLINK parameters, ignored by UNPHASED
```

When trait and marker names are defined by the data file, they must be used as the arguments to options. If no names are defined, traits and markers are automatically named 1, 2, … from left to right in the pedigree file. Separate numerical lists are maintained for marker genotypes, affection statuses and quantitative traits.

Option file

Instead of selecting the program options from the menus, or typing them on the command line, UNPHASED can read its options from a previously created *option file*. This file contains one line per option, with the arguments for each option listed on the same line. The leading dash can be omitted from the option name. For example, the option file:

-pedfile mypeds.ped
-window 2
-genotype
reference 1 2 1 1

will set up the same options as the command line

unphased mypeds.ped –window 2 –genotype –reference 1 2 1 1

Only one option file should be specified per run. When using an option file, additional options can be specified on the command line or through the GUI.

Output

Screen output

The screen output is in three sections. The first section lists all the analysis options and selected covariates, if any. This section can be suppressed using the **–brief** option. The second section displays the progress of the analysis. The number of likelihood evaluations at each stage is shown. The third section gives the result of an overall test of association and then a table with results for each haplotype (or allele or genotype), for the currently selected markers.

The overall test of association is a likelihood ratio test. The log-likelihoods for the null and alternative hypotheses are displayed, the hypotheses themselves being determined by the analysis options. The **likelihood ratio** statistic is minus twice the difference in log-likelihoods, and is asymptotically distributed as χ^2 with degrees of freedom (**df**) equal to the difference in number of free parameters between the two hypotheses. The **p-value** is the probability of observing a likelihood ratio statistic at least as large as this one, if the null hypothesis were true.

The table of results contains the following information:

Allele/Haplotype/Genotype: the genetic unit of interest. Alleles at successive loci are separated by a dash –. For genotypes, alleles at the same loci are separated by a slash /. Thus, 1/2–3/4 means genotype 1/2 at the first locus and 3/4 at the second.

Case: the estimated count of this haplotype in cases. In nuclear families, cases are the affected offspring. For quantitative traits, this column is not displayed.

Control: the estimated count of the haplotype in controls. In nuclear families, controls refer both to unaffected offspring and the untransmitted alleles in parents. For quantitative traits, this column is headed **Count** and gives the estimated count over all subjects.

Ca-Freq: the marginal frequency of the haplotype in cases. This is the count in the **Case** column divided by the column total. For quantitative traits, this column is omitted.

Co-Freq: the marginal frequency of the haplotype in controls. This is the count in the **Control** column divided by the column total. For quantitative traits, this column is headed **MarFreq** and give the marginal frequency of the haplotype over all subjects.

Odds-R: the estimated odds ratio for this haplotype. For quantitative traits this column is headed **AddVal** and shows the estimated additive genetic value for this haplotype, assuming a normally distributed trait and small deviations from the mean. These effects are shown relative to one haplotype, termed the *reference* haplotype. By default this is the first haplotype in sorted numerical order, but it can be changed with the **–reference** option. Odds ratio has the usual meaning. Additive value gives the change in expected trait value due to this haplotype, relative to the reference haplotype. So if a subject has haplotypes with additive values x and y, its expected trait mean is $x+y$ higher than for a subject homozygous for the reference haplotype. The default model assumes a normal trait distribution with unit variance, with small additive values. For large additive values, the **–normal** option permits more accurate estimation, but see the documentation for this option before using it. Non-unit variance is specified by the **–variance** option. More details of quantitative trait analysis are in the "Methods" section.

When effect modifiers are included, the odds ratio and additive values refer to the baseline levels, or a value of zero if the modifier is continuous.

95%lo: lower bound on the 95% confidence interval for the odds ratio or additive value.

95%hi: upper bound on the 95% confidence interval for the odds ratio or additive value.

If individual haplotype tests are selected, two more columns will appear:

Chisq : a score statistic for the effect of this haplotype relative to all others pooled together.

P-value : the asymptotic significance of this statistic referring to χ^2 on 1df.

If covariates are included, more tables are displayed, one for each level of each covariate. For continuous covariates, one table is displayed corresponding to a covariate value of 1. The **Coeff** column gives the main covariate coefficient γ as described in "Methods". For effect modifiers, the **Odds-R** or **AddVal** columns give the odds ratio or additive value compared to that of the reference haplotype, after allowing for the baseline effects in the main part of the output. Confidence intervals are as above, and χ^2 tests for the covariate effects can be computed. These test whether the haplotype effects are equal for each

covariate level, in other words whether there is any gene-covariate interaction after allowing for effects are the baseline level.

If LD output is selected, a further table is displayed consisting of D' and r^2 values for each haplotype, together with a global D' statistic. These values are computed from the marginal control frequencies displayed in the **Co-Freq** or **MarFreq** columns above.

Haplotype dump

UNPHASED allows all the possible haplotypes consistent with each family to be written to a file together with the probability of each haplotype configuration. This *dump file* can be useful although it is not particularly easy to read.

For nuclear families, each line of the dump file corresponds to a family. If all offspring are considered jointly (the default option), the subject IDs of these siblings are first listed. Then each configuration is given as a list of phased haplotype pairs with loci separated by a dash –, haplotypes separated by a slash /, transmitted and untransmitted haplotypes separated by a backslash \, and siblings separated by a comma , . Then the probability of the configuration is given. Note that the dump file distinguishes the phase of each configuration, so that some unphased configurations seem to appear more than once. For example, the single locus heterozygote will appear both as 1/2 and 2/1.

If offspring are considered independently (the **–nolinkage** option), each sibling is listed in turn with the same syntax as above.

For unrelated subjects, the configurations are listed as phased haplotype pairs with their corresponding probabilities.

Options

UNPHASED provides options for performing several different types of analysis. Here the options are grouped according to the menus in the GUI, with the corresponding command line arguments. At the end there is a full list of command line options together with their minimal abbreviated forms.

Reading and writing files

The **File** menu is used for input of data files and output of results, and also for managing the appearance of the GUI.

Open pedigree file

–pedfile <filename1> <filename2> …

Reads in the pedigree file from disk. The format is described above in the section "Input files".

Open data file

-datafile <filename>

Reads in the data file from disk. The format is described above in the section "Input files".

Open option file

–optionfile <filename>

Reads in the option file from disk. The format is described above in the section "Input files".

Close pedigree file

Closes the currently loaded pedigree file in the GUI. No more analyses can be performed until another pedigree file has been opened.

Close data file

Closes the currently loaded data file in the GUI. This means that all currently loaded marker and trait names will be lost and the GUI will assume that the currently loaded pedigree file has the basic format.

Save output

–output <filename>

In the GUI this saves the currently displayed tab to a text file. On the command line, it sends the main output to a file instead of the screen.

Save options

This is only available in the GUI, and creates a new option file containing the current selection of options.

Rename tab

In the GUI, tabs are automatically named after the pedigree file, and also display the count of how many analyses have been performed so far. This action lets you change the same of the currently displayed tab.

Close tab

In the GUI, this closes the currently display tab.

Print

In the GUI, this attempts to print the currently selected tab. If this doesn't work, it is more likely to be a problem with your local Java printing setup than with the GUI itself. Please check this out before reporting a bug.

Selecting traits to analyse

The **Trait** menu selects which traits to analyse. If the pedigree file has the basic format, there is no need to use this menu. UNPHASED distinguishes between two types of trait: binary traits taking values 1 or 2 (also called affection status or disease), and quantitative traits that can take any numerical value. Several traits can be analysed in one run, with each trait being analysed in turn. If both binary and quantitative traits are selected, UNPHASED analyses all the binary traits first, and then all the quantitative traits. If no traits are selected, UNPHASED analyses just the first binary trait if one is present, otherwise the first quantitative trait.

Affection status

–disease <name1> <name2> …

Specifies which binary traits to analyse. Each trait will be analysed in turn. In the GUI they will be analysed in the lexical order, whereas on the command line they will be analysed in the order in which they are listed.

Quantitative trait

-trait <name1> <name2> …

Specifies which quantitative traits to analyse. Each trait will be analysed in turn. In the GUI they will be analysed in the lexical order, whereas on the command line they will be analysed in the order in which they are listed.

Selecting markers to analyse

The **Marker** menu selects markers for analysis. UNPHASED distinguishes between three types of marker. *Test markers* are the markers of interest, which will be tested for association. *Conditioning markers* may themselves be associated, and those associations are factored out of the association of the test markers. *Tag markers* can add information about missing genotypes, but any associations are not considered. For each haplotype, UNPHASED estimates a population frequency, which depends on the alleles at all three sets of markers, and an odds ratio, which depends only on the alleles at test and conditioning markers.

When displaying haplotypes in the output, UNPHASED places the conditioning markers first, then the test markers, then the tag markers. So if there is one of each type of marker, the haplotype 1-2-3 refers to allele 1 at the conditioning marker, allele 2 at the test markers and allele 3 at the tag marker.

When several analyses are specified by the **–window** option below, the conditioning and tag markers are held fixed while the test markers are varied.

Test markers

–marker <name1> <name2> …

Selects a subset of markers out of the pedigree file. Each marker will be tested in turn for association. Use the names in the data file, or if no data file is loaded, use numbers 1, 2, …

Each marker will be analysed in turn. In the GUI they will be analysed in the lexical order, whereas on the command line they will be analysed in the order in which they are listed. To analyse multi-locus haplotypes, set a window size as described below.

If no markers are selected, UNPHASED automatically selects all the markers in the file. This gives a quick way to specify that every marker should be analysed.

Conditioning markers

–condition <name1> <name2> …

Suppose the test markers are in linkage disequilibrium (LD) with other markers that are known to be associated. We want to know if the test markers are directly associated, beyond the indirect association occurring through LD. In order to perform such a test, the previously associated *conditioning markers* should be selected using this option.

The conditioning markers are held fixed over a sliding window analysis. In the output, the alleles of the conditioning markers are displayed to the *left* of the test markers. This convention also applies to options that have a haplotype as an argument (for example –reference).

Tag markers

–tag <name1> <name2> …

When some genotypes are missing, information from other markers can be used to estimate them, if the markers are in linkage disequilibrium. Here these additional markers are called *tag markers*.

The tag markers are held fixed over a sliding window analysis. In the output, the alleles of the tag markers are displayed to the *right* of the test markers. This convention also applies to options that have a haplotype as an argument (for example –reference).

Window size

–window <value>

Used to specify multi-locus analysis. The window size is the number of test markers included in each multi-locus analysis.

By default, UNPHASED runs a sliding window analysis over the selected markers. For example, if the selected test markers are 1, 2, 3, 5, 8, and the window size is 3, then the following 3-marker analyses will be performed:

Markers 1, 2, 3
Markers 2, 3, 5
Markers 3, 5, 8

Instead of running a sliding window, UNPHASED can analyse all combinations of markers with the given window size; see below.

Conditioning and tag markers are held fixed over multiple test markers selected by the sliding window.

If no window size is specified, the default value is 1.

All marker combinations

–allcombinations

Specifies that all combinations of markers with the given window size should be analysed, instead of running a sliding window over the test markers.

For example, if the selected test markers are 1, 2, 3, 5, 8, and the window size is 3 with this option selected, then the following 3-marker analyses will be performed:

Markers 1, 2, 3
Markers 1, 2, 5
Markers 1, 2, 8
Markers 1, 3, 5
Markers 1, 3, 8
Markers 1, 5, 8
Markers 2, 3, 5
Markers 2, 3, 8
Markers 3, 5, 8

All window sizes

–allwindows

Specifies that each window size will be run in turn, starting with 1 and ending with the number of test markers. This option overrides the **–window** option. It can be used in conjunction with the **–allcombinations** option.

Selecting covariates

The **Covariate** menu selects covariates to be included in the analysis. UNPHASED allows modeling of interactions between haplotypes and environmental or other covariates. There is a distinction between two types of covariate. *Confounders* are covariates whose values influence the population frequency of a haplotype and possibly the disease prevalence or trait mean. An example might be geographical region. *Modifiers* are covariates whose values influence the odds ratio of a haplotype, or its additive value on a quantitative trait: that is, they are effect modifiers. A common example is sex. UNPHASED allows for the possibility that an effect modifier also influence the population haplotype frequencies.

UNPHASED conditions on all trait and covariate values (see the "Methods" section). This means that marginal effects of covariates are conditioned out of the likelihood and they cannot be estimated. In other words, all trait values are automatically adjusted for the main effect of a covariate, and we do not, for example, need to worry about adjusting measurements for age unless there is a confounding effect or a gene-environment interaction.

A confounder or modifier can be either continuous (the default) or a *factor*. Factors are discrete variables taking values among a finite set of *levels*. Each factor has a *baseline* level, which is the level for which UNPHASED gives the main analysis. Once a covariate is defined as a factor, UNPHASED automatically reads its levels from the pedigree file. The levels can be any numerical values.

Confounders

–confounder <name1> <name2> …

Any number of confounders can be selected. They are assumed to have effects on haplotype frequency but not haplotype risk. See the "Methods" section for exactly how they are modeled by UNPHASED.

Parental and offspring sex can be used as confounders, although it is unlikely that haplotypes will have different population frequencies between males and females. To use parental sex as a confounder, use the name "parsex". To use offspring sex as a confounder, use the name "sibsex". For unrelated subjects, also use "sibsex".

Modifiers

–modifier <name1> <name2> …

Any number of modifiers can be selected. They are assumed to modify the effect of haplotypes, that is the odds ratio or mean. See the "Methods" section for exactly how they are modeled by UNPHASED.

Parental and offspring sex can be used as modifiers. To use parental sex as a modifier, use the name "parsex". This corresponds to a parent of origin effect in the offspring. To use

offspring sex as a modifier, use the name "sibsex". For unrelated subjects, also use "sibsex". This corresponds to differential risk between males and females.

Factors

–factor <name1> <name2> …

Defines which covariates should be regarded as factors with discrete levels. If parental or offspring sex are used as covariates, they are automatically set as factors with two levels and they should not be included in this option.

Baselines

–baseline <value1> <value2> …

The argument for this option should be a list of baseline levels with one for each selected covariate. List the confounders first, then the modifiers. For covariates that are not factors, the baseline level will be ignored and can be given as 0. For parental and offspring sex, a baseline level of 1 corresponds to males and 2 to females.

Setting up the type of analysis

The **Analysis** menu sets up the main analysis. The first part of the menu selects an analysis "model", which is one of four built-in designs for coding haplotype risks. The next part allows additional tests to be reported apart from the main likelihood ratio test. The other parts define particular haplotypes for various purposes.

Full model

–model full

UNPHASED has several different parameterisations for the haplotype risk, which can be used to perform various different tests. The full model is the default model and is used when the **–model** option is omitted. It corresponds to full haplotype coding, with a unique odds ratio parameter for each haplotype. Under the null, all the odds ratios are equal. With conditioning markers, the null defines odds ratios to be equal for all haplotypes sharing the same conditioning alleles, but haplotypes with different conditioning alleles can have different odds ratios. Among the conditioning tests, this has the most degrees of freedom, so will tend to be less powerful unless there are strong interactions between test and conditioning markers.

When specific test and conditioning haplotypes are given, the inference only applies to full haplotypes that match those sub-sections. Odds ratios for other haplotypes are considered as nuisance parameters. The constraints on odds ratios are as follows:

H_0: all haplotypes matching the specific conditioning haplotype have equal odds ratio. All other haplotypes have unique odds ratios.
H_1: all haplotypes matching the specific conditioning haplotype and not matching the specific test haplotype have equal odds ratio. All other haplotypes have unique odds ratios.

In the above, "matching" is defined as true if no specific haplotype is given. "Unique" applies only to the conditioning and test haplotypes: if there are tag markers, then haplotypes with the same conditioning and test haplotypes but different tag haplotypes will have the same odds ratio.

Haplotype main effects

–model haplomain

In this model, the odds ratio of a haplotype is modeled by multiplicative contributions from the conditioning haplotype and from the test haplotype. The null hypothesis is that none of the test haplotypes make any contribution, in other words they have no main effect on the overall odds ratio. This test has fewer degrees of freedom than the full model, but may be less likely to detect an effect when there is strong interaction between conditioning and test haplotypes.

If no conditioning markers are selected, this model will give the same results as the full model.

As for the full model, specific test and conditioning haplotypes restrict the inference to full haplotypes that match those sub-sections. Odds ratios for other haplotypes are considered as nuisance parameters. The constraints on odds ratios are as follows:

H_0: all haplotypes matching the specific conditioning haplotype have equal odds ratio. All other haplotypes have unique odds ratios.
H_1: all haplotypes matching the specific conditioning haplotype and not matching the specific test haplotype have equal odds ratio. Haplotypes matching the both the specific conditioning haplotype and the specific test haplotype have odds ratios estimated by the haplotype main effects model. All other haplotypes have unique odds ratios.

Allele main effects

–model allelemain

In this model, the odds ratio of a haplotype is modeled by multiplicative contributions from each test marker. The null hypothesis is that none of the test markers make any contribution to the overall odds ratio. This test has the fewest degrees of freedom of all the models, but may be less likely to detect an effect when there is strong interaction between markers.

Conditioning markers are modeled by full haplotype coding, so the main effects only apply to test markers. Because the test markers are modeled by main effects, there is no sensible interpretation of a specific test haplotype, and the **–specific** option is ignored by this model. However a specific conditioning haplotype may still be used to restrict inference to a subset of haplotypes. The constraints on odds ratios are as follows:

H_0: all haplotypes matching the specific conditioning haplotype have equal odds ratio. All other haplotypes have unique odds ratios.
H_1: all haplotypes matching the specific conditioning haplotype and not matching the specific test haplotype have equal odds ratio. Haplotypes matching both the specific conditioning haplotype and the specific test haplotype have odds ratios estimated by the allele main effects model. All other haplotypes have unique odds ratios.

Gene-gene interaction

–model gxg

This model tests for gene-gene interaction by considering whether the odds ratios of the conditioning haplotypes are independent of those of the test haplotypes. That is, it compares the odds ratio of the full haplotype to that expected if the conditioning and test haplotypes had independent risks. Note that this only tests for *cis*-phase interaction, unless genotype tests are selected, in which case either *cis*- or *trans*-phase interaction could be detected. Note also that this statistical definition of interaction does not necessarily correspond to the classical model of epistasis.

If no conditioning markers are selected, this model will compute identical likelihoods under both null and alternative.

Essentially, this model compares the likelihoods between the haplotype main effects model described above, and the full model. Inference can be restricted to haplotypes matching specific conditioning and test haplotypes. More precisely, the constraints on odds ratios are as follows:

H_0: all haplotypes have odds ratios estimated by the haplotype main effects model.
H_1: haplotypes matching both the specific conditioning haplotype and the specific test haplotype have unique odds ratios. All other haplotypes have odds ratios estimated by the haplotype main effects model.

Individual haplotype tests

–individual

This option will give individual tests for each haplotype in turn. The test is a score test for a difference in risk between a haplotype and all the others pooled together. This is in contrast to the odds ratios, which are shown relative to a single reference haplotype. It is possible for the individual test to be significant when the confidence interval for the odds ratio includes 1, and vice versa. To get an estimate of a haplotype risk relative to all others together, see the "Specific test haplotype" section.

This option has changed from the previous version of UNPHASED, 2.4, in that it now uses a score test whereas the old version used a likelihood ratio test. Results should be very similar but there may be differences in some cases. The likelihood ratio can still be used to double check results from the score test, again see the "Specific test haplotype" section.

The score test uses an outer product estimator for its variance, which is accurate in a large sample. Therefore the score test may give misleading results in small samples, and the likelihood ratio test should be used to for comparison.

Test confounder effects

–testconfounders

For each level of a confounder, this gives a global Wald test of whether any of the regression coefficients are non-zero.

Test modifier effects

–testmodifiers

For each level of a modifier, this gives a Wald test for each haplotype, indicating whether the modifier changes its odds ratio or mean when compared to the baseline level.

Compare haplotype risks

–compare <allele1> <allele2> …
-with <allele1> <allele2> …

This option allows the risks of two haplotypes to be compared. Under the null, their risks are set to be equal while all other haplotypes have freely estimated risk. Under the alternative all haplotypes can have different risks. The two haplotypes are specified with the **–compare** and **–with** options respectively.

Wildcards are allowed in the pairwise comparison, and are specified by 0. These have the effect of forcing the odds ratios to be equal for all haplotypes matching the wildcard. So for example

unphased mypeds.ped –window 2 –compare 1 0 –with 2 2

will test whether all haplotypes with a 1 at the first locus have the same odds ratio as the 2-2 haplotype. If there are wildcards in both haplotypes, a haplotype is considered to match the second haplotype only if it does not match the first.

The comparison options override the analysis specified by the **–model** option.

Reference haplotype

–reference <allele1> <allele2> …

This option specifies the reference haplotype, which is defined to have odds ratio 1. The default reference haplotype is the first one when the haplotypes are sorted into numerical order.

Specific test haplotype

–specific <allele1> <allele2> …

This option restricts the inference to a specific test haplotype. How this is done depends on which model is selected, and is described in the sections above. The specific haplotype should only be specified for the test markers. If conditioning or tag markers are selected, the number of arguments to this option must remain as the number of test markers.

Wildcards are allowed as arguments and are specified as 0.

Specific conditioning haplotype

–condspecific <allele1> <allele2> …

This option restricts the inference to a specific conditioning haplotype. How this is done depends on which model is selected, and is described in the sections above. The specific haplotype should only be specified for the conditioning markers. If test or tag markers are selected, the number of arguments to this option must remain as the number of conditioning markers.

Wildcards are allowed as arguments and are specified as 0.

Analysis options

The **Options** menu sets up various options for performing the analysis.

Missing data : Certain haplotypes only

–certain

Restricts analysis to subjects for whom the haplotypes are unambiguous. In nuclear families this corresponds to at most one heterozygous intercross. In unrelated subjects this selects only homozygous subjects, unless only one marker is selected. In all cases, subjects with missing genotypes are discarded.

Missing data : Uncertain haplotypes

This is the default setting and includes all subjects that have no missing genotype data.

Missing data : Uncertain haplotypes and missing genotypes

–missing

Selects all subjects that have some genotype data, and averages over all possible completions of the data. It is the slowest of the three missing data options but uses the most information.

Rare haplotypes : Rare frequency threshold

–rare <value>

Sets a frequency below which haplotypes are designated as rare. First, UNPHASED estimates all the haplotype odds ratios and estimates their marginal frequencies. Then it identifies the rare haplotypes, and from then on constrains them to have the same odds ratio, in order to reduce computation. The pooled odds ratio is treated as a nuisance parameter in the likelihood ratio tests, so that rare haplotypes are not tested for association.

By default the haplotype must be rare in both cases and controls. This can be changed with the **–rarfreq** option.

If the value is greater than 1, it is interpreted as a percentage, unless the **–cellcount** option is selected.

Rare haplotypes : Zero frequency threshold

–zero <value>

Sets a frequency below which haplotypes are assumed to be absent from the population. First, UNPHASED estimates all the haplotype frequencies and odds ratios. Frequencies below this threshold are rounded down to zero. From then on, configurations involving these haplotypes are ignored. This can significantly speed up the computation and declutter the output.

By default the haplotype must be absent from both cases and controls. This can be changed with the **–userare** option.

The default value of the zero frequency threshold is 10^{-8}.

Rare haplotypes : Threshold on cell counts

–cellcount

Identifies rare haplotypes according to the cell counts rather than the marginal frequencies. This option does not affect the zero frequency threshold.

Rare haplotypes : Use frequencies in : Both cases and controls

–userare both

Defines a haplotype to be rare if its marginal frequency is below the threshold in both cases and controls. This is the default setting.

Rare haplotypes : Use frequencies in : Either cases or controls

–userare either

Defines a haplotype to be rare if its marginal frequency is below the threshold in either cases or controls.

Rare haplotypes : Use frequencies in : Just cases

–userare case

Defines a haplotype to be rare if its marginal frequency is below the threshold in the cases.

Rare haplotypes : Use frequencies in : Just controls

–userare control

Defines a haplotype to be rare if its marginal frequency is below the threshold in the controls.

Nuclear families : Assume no linkage

–nolinkage

When there are multiple siblings in a nuclear family, transmissions are not independent if there is known to be linkage in the region. When selected, this option assumes that there is no linkage, so that the siblings are assumed to be independent. This results in a faster and more powerful analysis than the default.

Nuclear families : Model odds ratio in parents

–parentrisk

This option enables more valid inference when there are missing parents and multiple siblings in a nuclear family (see the "Methods" section). However it introduces more nuisance parameters and therefore reduces power and increases running time.

Genetic : Genotype tests

–genotype

Selects genotype tests instead of haplotype tests. Genotypes are unphased throughout.

Genetic : Condition on genotypes

–condgenotype

Models genotypes at the conditioning markers but not at the test and tag markers. This can be useful if there is a strongly dominant effect at the conditioning markers.

The output will show estimates for combinations of genotypes and haplotypes, for example 1/2–1, 1/2–2, etc. This is confusing from a genetics point of view, but statistically it corresponds to using the conditioning genotype as a covariate when making inferences on test haplotypes.

Genetic : Autosome

This default option assumes that the markers are autosomal.

Genetic : Chromosome X

–chrX

Indicates that all markers are on chromosome X. In family studies, only the maternal transmissions are counted. In male subjects, only one chromosome is considered. If males and females are combined in the same analysis, the odds ratio in males is assumed to equal the heterozygote odds ratio in females. Unless this is expected to be the case, it is advisable to include "sibsex" as a modifier, in order to give separate odds ratios in males and females.

Genetic : Chromosome Y

–chrY

Indicates that all markers are on chromosome Y. In nuclear families, the parents will be discarded and just the first male offspring will be used. It will be then be analysed as an unrelated subject.

Quantitative trait : Model normal distribution

–normal

Uses an explicit normal distribution to model quantitative traits, instead of the small-effects approximation used by default (see the "Methods" section). This option is more accurate at estimating the additive value of a haplotype, particularly when the effect is large, and can be more powerful when the trait is truly normal. However, the likelihood can be more difficult to maximize numerically, and this model is less robust to non-normality than the logistic model.

Quantitative trait : Trait variance

–variance

Specify the trait variance. When used without the **–normal** option, this does not change the p-value but does scale the estimated additive value. When used with the **–normal** option, this does change the inference as well as the additive value. Of course, correct inference occurs only when the correct variance is used. However the variance may be difficult to estimate numerically, and may even not be identifiable in the likelihood, so for these reasons the variance is left as an input option. It could be profiled across several values to choose the most likely value.

Permutation test

–permutation <value>

Specifies how many random permutations to run. In nuclear families, the permutations are generated by randomizing the transmission status of the parental haplotypes. In unrelated subjects, the trait values are randomly shuffled between subjects. The randomization is held constant over all analyses specified by the **–disease**, **–trait**, **–marker** and **–window** options, as well as any tests selected by the **–individual**, **–testconfounders** and **–testmodifiers** options. In each permutation, the minimum p-value is compared to the minimum p-value over all the analyses in the original data. This allows for multiple testing corrections over all tests performed in a run.

Show quantile from permutations

–quantile <value>

Outputs a percentile point from the permutation distribution of the minimum p-value. This allows us to say whether other results are also significant after correcting for multiple testing, apart from that with the minimum p-value. For example, if the second-most significant result from the original analysis is also less than the 5^{th} percentile of the permutation distribution, we could say that it is also significant over multiple tests.

Convergence threshold

–epsilon <value>

Sets the tolerance for convergence in the numerical maximisation of the likelihood. The default value is 10^{-8}.

Random restarts

–restarts <value>

Maximises the likelihood several times, using randomly chosen starting points. It is useful if the likelihood seems not be maximised at the first attempt, for example if there is an extraordinary difference between null and alternative likelihoods, or the alternative is less than the null.

Random number seed

–randomseed <value>

Sets the initial seed for the random number generator. The default value is 1.

Output options

The **Output** menu sets various options for controlling the output of UNPHASED.

Brief output

–brief

When not selected, UNPHASED prints a list of all options and their currently selected values. This option stops this list from printing, making the output briefer.

Show LD measures

–LD

Displays a table of D´ and r^2 measures for each haplotype, together with global D´ and correlation statistics. If more than two markers are selected, the LD is computed between the first marker and all the others grouped together. This can be used to assess how useful multiple markers could be when used with the **–tag** option.

Output permutation analyses

–permoutput

Displays the output from each permuted data set. It can be useful for observing the behaviour of the permutation distribution.

Output running time

–time

Displays the time in seconds taken to perform all the analysis.

Dump haplotypes to file

–dumpfile <filename>

Writes the probabilities of all possible haplotypes for each subject into a file. The format of the file is described in the section "Haplotype dump" above.

Just the most likely haplotypes

–mostlikely

Writes only the most likely haplotype solutions to the dump file, together with their probabilities. There may be more than one most likely solution.

Additional options

The following options are not available from the GUI, and are primarily used for development. They are included here for completeness.

–follow

Displays the log-likelihood after each evaluation in numerical maximisation.

–llhd

Gives a complete dump of parameter values, log-likelihood contributions and gradients, for each subject and each likelihood evaluation. Used for debugging.

–neldermead

Uses Nelder & Mead's downhill simplex method for likelihood maximisation, instead of the default Davidon/Fletcher/Powell. It is much slower, but can sometimes get an answer when DFP converges to the wrong solution.

–show

Displays the internal representation of haplotype solutions for all individuals, and the internal codes for haplotypes and genotypes. Mainly used for debugging.

–slow

Uses a slow algorithm for evaluating the denominator in the likelihood, fully enumerating all terms instead of factorising. Used for debugging. The **–genotype** and **–condgenotype** options automatically set this option.

Command line options

These are all the command line options with their minimal abbreviated forms and brief descriptions.

Option	Abbreviation	Description
-allcombinations	-allc	Test all combinations of a fixed number of markers
-allwindows	-allw	Test all window sizes
-baseline	-ba	Set baseline levels for factorial covariates
-brief	-br	Brief screen output
-cellcount	-cel	Identify rare haplotypes based on cell count
-certain	-cer	Only analyse certain haplotypes
-chrX	-chrX	Markers on chromosome X
-chrY	-chrY	Markers on chromosome Y
-compare	-com	First haplotype in comparison of two odds ratios
-condgenotype	-condg	Condition on genotypes
-condition	-condi	Select conditioning markers
-condspecific	-conds	Condition on a specific haplotype
-confounder	-conf	Select confounders
-datafile	-da	Open a data file
-disease	-di	Select affection status
-dumpfile	-du	Dump haplotypes to file
-epsilon	-e	Set convergence threshold
-factor	-fa	Designate covariates as factors
-follow	-fo	Display each evaluated log-likelihood
-genotype	-g	Analyse genotypes
-individual	-i	Test individual haplotypes
-llhd	-l	Output nearly all internal calculations
-marker	-ma	Select test markers
-missing	-mi	Include uncertain haplotypes and missing genotypes
-model full	-mode full	Full haplotype coding model
-model gxg	-mode gxg	Test for gene-gene interaction
-model haplomain	-mode haplomain	Haplotype main effects model
-model allelemain	-mode allelemain	Allele main effects model
-modifier	-modi	Select effect modifiers
-mostlikely	-mos	Only dump the most likely haplotypes
-neldermead	-ne	Maximise likelihood with Nelder-Mead method
-nolinkage	-nol	Assume no linkage in nuclear families
-normal	-nor	Model explicit normal distribution for quantitative trait
-optionfile	-op	Open option file
-output	-ou	Save output to a file
-parentrisk	-pa	Model odds ratio in parents
-pedfile	-ped	Open pedigree file
-permoutput	-permo	Output permutation analyses
-permutation	-permu	Set number of permutations
-quantile	-q	Show quantile from permutation distribution

-randomseed	-ran	Set random number seed
-rare	-rare	Set threshold for rare haplotypes
-reference	-ref	Reference haplotype
-restarts	-res	Number of random restarts in maximizing likelihood
-show	-sh	Show internal representation of possible haplotypes
-slow	-sl	Slower calculation of denominator in likelihood
-specific	-sp	Test a specific haplotype
-tag	-ta	Select tag markers
-testconfounders	-testc	Test confounder effects
-testmodifiers	-testm	Test modifier effects
-time	-ti	Output running time
-trait	-tr	Select quantitative trait
-userare both	-u both	Define rare haplotypes from both cases & controls
-userare either	-u either	Define rare haplotypes from either cases or controls
-userare case	-u case	Define rare haplotypes from cases
-userare control	-u control	Define rare haplotypes from controls
-window	-win	Sets number of markers for multi-locus analysis
-with	-wit	Second haplotype in comparison of odds ratio
-zero	-z	Set threshold for haplotypes with zero frequency

Methods

Likelihood

UNPHASED uses a retrospective likelihood, which is the probability of observing the parental and child genotypes, given the trait values of all the children in a nuclear family. This is appropriate when the data have been sampled based on the trait values of the children, which is usually true for case/parent trios, affected sib pairs, case/control samples and extreme sampling of quantitative traits. When other types of sampling occur, for example prospective sampling in a cohort study, the retrospective likelihood can still give valid results, but it is not the most efficient form of analysis.

In the following we will work with a quantitative trait *Y*, with specialization to a binary trait noted where appropriate. The retrospective likelihood for a nuclear family with *k* children is

$$\Pr(f,m,\mathbf{c}\mid\mathbf{y}) = \frac{\Pr(f,m)\Pr(\mathbf{c}\mid f,m)\Pr(\mathbf{y}\mid f,m,\mathbf{c})}{\sum_{f^*,m^*,c_1^*,\ldots,c_k^*\in G}\Pr(f^*,m^*)\Pr(\mathbf{c}^*\mid f^*,m^*)\Pr(\mathbf{y}\mid f^*,m^*,\mathbf{c}^*)}$$

where f and m are the genotypes of the father and mother, $\mathbf{c}=(c_1,\ldots c_k)^{\mathrm{T}}$ is the vector of child genotypes, $\mathbf{y}=(y_1,\ldots y_k)^{\mathrm{T}}$ is the vector of child traits, and G is the set of all genotypes. For the trait distribution we assume a simple multivariate normal model, with common mean μ and variance σ^2 for all children and no covariance between the traits of children. Although this model is over-simplified, the retrospective nature of the likelihood gives some robustness. Assuming random mating and Mendelian transmission we may write

$$\begin{aligned}\Pr(f,m,\mathbf{c}\mid\mathbf{y}) &= \frac{\exp(\alpha_f+\alpha_m-\tfrac{1}{2}\sigma^{-2}(\mathbf{y}-\mathbf{1}.\mu-\boldsymbol{\beta}[\mathbf{c}])'(\mathbf{y}-\mathbf{1}.\mu-\boldsymbol{\beta}[\mathbf{c}]))}{\sum_{f^*,m^*}\sum_{\mathbf{c}\in S(f^*,m^*)}\exp(\alpha_{f^*}+\alpha_{m^*}-\tfrac{1}{2}\sigma^{-2}(\mathbf{y}-\mathbf{1}.\mu-\boldsymbol{\beta}[\mathbf{c}^*])'(\mathbf{y}-\mathbf{1}.\mu-\boldsymbol{\beta}[\mathbf{c}^*]))} \\ &= \frac{\exp(\alpha_f+\alpha_m+\sigma^{-2}(\mathbf{y}'\boldsymbol{\beta}[\mathbf{c}]-\mu\boldsymbol{\beta}[\mathbf{c}]-\tfrac{1}{2}\boldsymbol{\beta}[\mathbf{c}]'\boldsymbol{\beta}[\mathbf{c}])'(\mathbf{y}'\boldsymbol{\beta}[\mathbf{c}]-\mu\boldsymbol{\beta}[\mathbf{c}]-\tfrac{1}{2}\boldsymbol{\beta}[\mathbf{c}]'\boldsymbol{\beta}[\mathbf{c}]))}{\sum_{f^*,m^*}\sum_{\mathbf{c}^*\in S(f^*,m^*)}\exp(\alpha_{f^*}+\alpha_{m^*}+\sigma^{-2}(\mathbf{y}'\boldsymbol{\beta}[\mathbf{c}^*]-\mu\boldsymbol{\beta}[\mathbf{c}^*]-\tfrac{1}{2}\boldsymbol{\beta}[\mathbf{c}^*]'\boldsymbol{\beta}[\mathbf{c}^*]))}\end{aligned}$$

where $\boldsymbol{\alpha}$ is a parameter vector for a multinomial distribution of genotypes, $\boldsymbol{\beta}$ is a parameter vector of additive genotype effects, $\boldsymbol{\beta}[\mathbf{c}]$ denotes the vector $(\beta_{c_1},\ldots,\beta_{c_k})$, and $S(f^*,m^*)$ is the set of possible child genotype vectors from parents with genotypes f^* and m^*.

We prefer to work with the conditional likelihood of children given parents, because this is more robust to population stratification. Writing

$$\Pr(f,m,\mathbf{c}\mid\mathbf{y}) = \Pr(\mathbf{c}\mid f,m,\mathbf{y})\cdot\Pr(f,m\mid\mathbf{y})$$

we have

$$\Pr(f,m,\mathbf{c}\mid\mathbf{y}) = \frac{\exp(\sigma^{-2}(\mathbf{y}'\boldsymbol{\beta}[\mathbf{c}]-\mu\boldsymbol{\beta}[\mathbf{c}]-\tfrac{1}{2}\boldsymbol{\beta}[\mathbf{c}]'\boldsymbol{\beta}[\mathbf{c}]))}{\sum_{\mathbf{c}^*\in S(f,m)}\exp(\sigma^{-2}(\mathbf{y}'\boldsymbol{\beta}[\mathbf{c}^*]-\mu\boldsymbol{\beta}[\mathbf{c}^*]-\tfrac{1}{2}\boldsymbol{\beta}[\mathbf{c}^*]'\boldsymbol{\beta}[\mathbf{c}^*]))}\cdot \frac{\sum_{\mathbf{c}^*\in S(f,m)}\exp(\alpha_f+\alpha_m+\sigma^{-2}(\mathbf{y}'\boldsymbol{\beta}[\mathbf{c}]-\mu\boldsymbol{\beta}[\mathbf{c}]-\tfrac{1}{2}\boldsymbol{\beta}[\mathbf{c}]'\boldsymbol{\beta}[\mathbf{c}]))}{\sum_{f^*,m^*}\sum_{\mathbf{c}^*\in S(f^*,m^*)}\exp(\alpha_{f^*}+\alpha_{m^*}+\sigma^{-2}(\mathbf{y}'\boldsymbol{\beta}[\mathbf{c}^*]-\mu\boldsymbol{\beta}[\mathbf{c}^*]-\tfrac{1}{2}\boldsymbol{\beta}[\mathbf{c}^*]'\boldsymbol{\beta}[\mathbf{c}^*]))}$$

The innovation in UNPHASED is to model the genotype effect $\boldsymbol{\beta}$ separately in the two components of the likelihood. This means that we can perform inference within the conditional likelihood while taking the parental distribution into account, which is necessary in order to deal with missing genotypes or uncertain haplotypes. We substitute new parameters into the parental probability and write

$$\Pr(f,m,\mathbf{c}\mid\mathbf{y}) = \frac{\exp(\sigma^{-2}(\mathbf{y}'\boldsymbol{\beta}[\mathbf{c}]-\mu\boldsymbol{\beta}[\mathbf{c}]-\tfrac{1}{2}\boldsymbol{\beta}[\mathbf{c}]'\boldsymbol{\beta}[\mathbf{c}]))}{\sum_{\mathbf{c}^*\in S(f,m)}\exp(\sigma^{-2}(\mathbf{y}'\boldsymbol{\beta}[\mathbf{c}^*]-\mu\boldsymbol{\beta}[\mathbf{c}^*]-\tfrac{1}{2}\boldsymbol{\beta}[\mathbf{c}^*]'\boldsymbol{\beta}[\mathbf{c}^*]))}\cdot \frac{\sum_{\mathbf{c}^*\in S(f,m)}\exp(\alpha_f+\alpha_m+\sigma^{-2}(\mathbf{y}'\widetilde{\boldsymbol{\beta}}[\mathbf{c}^*]-\mu\widetilde{\boldsymbol{\beta}}[\mathbf{c}^*]-\tfrac{1}{2}\widetilde{\boldsymbol{\beta}}[\mathbf{c}^*]'\widetilde{\boldsymbol{\beta}}[\mathbf{c}^*]))}{\sum_{f^*,m^*}\sum_{\mathbf{c}^*\in S(f^*,m^*)}\exp(\alpha_{f^*}+\alpha_{m^*}+\sigma^{-2}(\mathbf{y}'\widetilde{\boldsymbol{\beta}}[\mathbf{c}^*]-\mu\widetilde{\boldsymbol{\beta}}[\mathbf{c}^*]-\tfrac{1}{2}\widetilde{\boldsymbol{\beta}}[\mathbf{c}^*]'\widetilde{\boldsymbol{\beta}}[\mathbf{c}^*]))}$$

Inference is now performed on $\boldsymbol{\beta}$, with $\widetilde{\boldsymbol{\beta}}$ treated as nuisance parameters. In UNPHASED the default model sets $\widetilde{\boldsymbol{\beta}}=\mathbf{0}$. This is appropriate for the null hypothesis that all genotypes have the same effects, but for other tests such as the conditional or gene-gene interaction tests, it is more accurate to freely estimate $\widetilde{\boldsymbol{\beta}}$. It is also appropriate when testing association in the presence of linkage, as described below. Free estimation of $\widetilde{\boldsymbol{\beta}}$ is specified by the **–parentrisk** option.

Under the null hypothesis that $\boldsymbol{\beta}=\mathbf{0}$, the trait mean μ is not identifiable, which can lead to difficulties with the asymptotic theory; and when $\boldsymbol{\beta}$ is nonzero but small, μ is difficult to estimate numerically. We therefore replace terms involving μ by further nuisance parameters $\boldsymbol{\nu}$ and $\widetilde{\boldsymbol{\nu}}$ which are always estimable:

$$\Pr(f,m,\mathbf{c}\mid\mathbf{y})=\frac{\exp(\boldsymbol{\nu}[\mathbf{c}]+\sigma^{-2}(\mathbf{y}'\boldsymbol{\beta}[\mathbf{c}]-\frac{1}{2}\boldsymbol{\beta}[\mathbf{c}]'\boldsymbol{\beta}[\mathbf{c}]))}{\sum_{\mathbf{c}^*\in S(f,m)}\exp(\boldsymbol{\nu}[\mathbf{c}^*]+\sigma^{-2}(\mathbf{y}'\boldsymbol{\beta}[\mathbf{c}^*]-\frac{1}{2}\boldsymbol{\beta}[\mathbf{c}^*]'\boldsymbol{\beta}[\mathbf{c}^*]))}\cdot \frac{\sum_{\mathbf{c}^*\in S(f,m)}\exp(\alpha_f+\alpha_m+\tilde{\boldsymbol{\nu}}[\mathbf{c}]+\sigma^{-2}(\mathbf{y}'\tilde{\boldsymbol{\beta}}[\mathbf{c}^*]-\frac{1}{2}\tilde{\boldsymbol{\beta}}[\mathbf{c}^*]'\tilde{\boldsymbol{\beta}}[\mathbf{c}^*]))}{\sum_{f^*,m^*}\sum_{\mathbf{c}^*\in S(f^*,m^*)}\exp(\alpha_{f^*}+\alpha_{m^*}+\tilde{\boldsymbol{\nu}}[\mathbf{c}]+\sigma^{-2}(\mathbf{y}'\tilde{\boldsymbol{\beta}}[\mathbf{c}^*]-\frac{1}{2}\tilde{\boldsymbol{\beta}}[\mathbf{c}^*]'\tilde{\boldsymbol{\beta}}[\mathbf{c}^*]))}$$

The above model is the one implemented by UNPHASED under the **–normal** option. However, maximization of the likelihood can be time-consuming and numerically unstable, so this is not the default model. Some simplification is possible by assuming that the β's are sufficiently small that the second-order terms can be ignored. By further setting σ=1, the default model of UNPHASED is

$$\Pr(f,m,\mathbf{c}\mid\mathbf{y})=\frac{\exp(\boldsymbol{\nu}[\mathbf{c}]+\mathbf{y}'\boldsymbol{\beta}[\mathbf{c}])}{\sum_{\mathbf{c}^*\in S(f,m)}\exp(\boldsymbol{\nu}[\mathbf{c}^*]+\mathbf{y}'\boldsymbol{\beta}[\mathbf{c}^*])}\cdot\frac{\sum_{\mathbf{c}^*\in S(f,m)}\exp(\alpha_f+\alpha_m)}{\sum_{f^*,m^*}\sum_{\mathbf{c}^*\in S(f^*,m^*)}\exp(\alpha_{f^*}+\alpha_{m^*})}$$

For binary traits, it is convenient to code $y_i=1$ for affected, 0 for unaffected. In this case $\boldsymbol{\nu}$ represents transmission distortion to unaffecteds, but this is unlikely to deviate from expectation unless there are strong protective effects. We therefore set $\boldsymbol{\nu}$=**0** which gives the same conditional likelihood as used in standard models of the TDT, and $\boldsymbol{\beta}$ may be interpreted as log odds ratios.

When data are missing, we form the above probabilities for each possible completion and sum the probabilities to give the likelihood contribution for the family.

For unrelated subjects, we assume two missing parents and set $\tilde{\boldsymbol{\nu}}=\boldsymbol{\nu}$ and $\tilde{\boldsymbol{\beta}}=\boldsymbol{\beta}$, giving likelihood contribution

$$\Pr(c\mid y)=\frac{\exp(\alpha_c+\nu_c+\sigma^{-2}(y\beta_c-\frac{1}{2}\beta_c^2))}{\sum_{c^*\in G}\exp(\alpha_{c^*}+\nu_{c^*}+\sigma^{-2}(y\beta_{c^*}-\frac{1}{2}\beta_{c^*}^2))}$$

Again this may be simplified by omitting the second-order term and setting σ=1. For a binary trait this then becomes a standard retrospective likelihood for case-control data

$$\Pr(c\mid y)=\frac{\exp(\alpha_c+\nu_c+y\beta_c)}{\sum_{c^*\in G}\exp(\alpha_{c^*}+\nu_{c^*}+y\beta_{c^*})}$$

in which $\boldsymbol{\beta}$ may again be interpreted as log odds ratios and $\boldsymbol{\alpha}$ are multinomial logistic parameters for the genotype distribution in controls. When the sample consists only of unrelated subjects, $\boldsymbol{\alpha}$ and $\boldsymbol{\nu}$ become confounded so that it is sensible to set $\boldsymbol{\nu}=\mathbf{0}$.

So far we have assumed that transmissions to the children are independent. This actually corresponds to the **–nolinkage** option in UNPHASED, but the default behaviour allows for dependence amongst transmissions, such as when there is prior linkage. This is done by conditioning on the inheritance vector of the observed data, in the conditional part of the likelihood. For each child c_i of parents f and m, construct three *virtual genotypes* at follows:

u_i^m : haplotype transmitted by father + haplotype not transmitted by mother

u_i^f : haplotype not transmitted by father + haplotype transmitted by mother

u_i^{fm} : haplotype not transmitted by father + haplotype not transmitted by mother

Then the sets of child genotypes $\mathbf{u}^m$, $\mathbf{u}^f$ and $\mathbf{u}^{fm}$ have the same inheritance vector as the observed family, up to an equivalence class. Conditioning on this inheritance vector changes the summations over possible sibships. The normal likelihood becomes

$$\Pr(f,m,\mathbf{c}\mid\mathbf{y}) = \frac{\exp(\boldsymbol{\nu}[\mathbf{c}]+\sigma^{-2}(\mathbf{y}'\boldsymbol{\beta}[\mathbf{c}]-\tfrac{1}{2}\boldsymbol{\beta}[\mathbf{c}]'\boldsymbol{\beta}[\mathbf{c}]))}{\sum_{\mathbf{c}^*\in\{\mathbf{c},\mathbf{u}^m,\mathbf{u}^f,\mathbf{u}^{fm}\}}\exp(\boldsymbol{\nu}[\mathbf{c}^*]+\sigma^{-2}(\mathbf{y}'\boldsymbol{\beta}[\mathbf{c}^*]-\tfrac{1}{2}\boldsymbol{\beta}[\mathbf{c}^*]'\boldsymbol{\beta}[\mathbf{c}^*]))} \cdot$$

$$\frac{\sum_{\mathbf{c}^*\in\{\mathbf{c},\mathbf{u}^m,\mathbf{u}^f,\mathbf{u}^{fm}\}}\exp(\alpha_f+\alpha_m+\tilde{\boldsymbol{\nu}}[\mathbf{c}]+\sigma^{-2}(\mathbf{y}'\tilde{\boldsymbol{\beta}}[\mathbf{c}^*]-\tfrac{1}{2}\tilde{\boldsymbol{\beta}}[\mathbf{c}^*]'\tilde{\boldsymbol{\beta}}[\mathbf{c}^*]))}{\sum_{f^*,m^*}\sum_{\mathbf{c}^*\in S(f^*,m^*)}\exp(\alpha_{f^*}+\alpha_{m^*}+\tilde{\boldsymbol{\nu}}[\mathbf{c}]+\sigma^{-2}(\mathbf{y}'\tilde{\boldsymbol{\beta}}[\mathbf{c}^*]-\tfrac{1}{2}\tilde{\boldsymbol{\beta}}[\mathbf{c}^*]'\tilde{\boldsymbol{\beta}}[\mathbf{c}^*]))}$$

with the same simplifications as above for small effects and binary traits. Note however that if $\tilde{\boldsymbol{\beta}}=\mathbf{0}$, the parental part of the likelihood is proportional to that obtained when not conditioning on the inheritance vector, so the **–parentrisk** option should be used to maintain robustness. This approach has the effect of treating the whole family as one sampling unit, in which unmodelled correlation is absorbed into the error variance.

Covariates are included by adding terms to the linear model. UNPHASED only includes first-order terms for covariates. For a factorial confounder at level z_i in child i we add parameters $\gamma_{g;z_i}$ to terms involving α_g , and for a continuous confounder with value z_i we add terms $z_i\gamma_g$. The γ parameters reflect changes in haplotype frequency and population mean according to different covariate values, though they may not always be interpreted directly as such. For a factorial effect modifier at level z_i we add parameters $y_i\eta_{g;z_i}+\gamma_{g;z_i}$ to terms involving β_g , and for a continuous modifier with value z_i we add parameters $y_i z_i\eta_g + z_i\gamma_g$. Here the η parameters reflect changes in odds ratio or additive value, on a linear scale with trait value. These are the odds ratios or additive values displayed in the output. The γ parameters adjust the trait mean surrogates ν for the covariate, as well as

accounting for any potential confounding; these are the regression coefficients displayed in the output.

UNPHASED conditions on all the covariate values, so there is no "main effect" of a covariate. This means that all analyses are automatically adjusted for average effects of covariates. This is often more conditioning than we should need, but it results in simpler computations, particularly for continuous covariates.

To summarise, the default model assumes small genotype effects $\boldsymbol{\beta}$, unit variance (σ^2=1), no genotype effects in the parental likelihood ($\widetilde{\boldsymbol{\beta}} = \mathbf{0}$), and conditions on the inheritance vector. To use more complex likelihoods in UNPHASED, the **–normal**, **–variance**, **–parentrisk** and **–nolinkage** options may be applied to relax these respective restrictions. The **–normal** and **–parentrisk** options both increase the computation time, whereas **–nolinkage** decreases it, sometimes considerably. The **–normal** option may encounter problems of numerical convergence, which can be addressed with the **–restarts** and **–neldermead** options, both of which increase the running time again.

The variance is left as an input parameter, rather than being estimated by UNPHASED. This is because numerical maximization of the variance is somewhat unreliable, basically because the mean is not always identifiable. Also in the situation of $\boldsymbol{\beta}=\mathbf{0}$, the variance itself is not identifiable, leading to problems with the asymptotic theory. The variance could however be estimated by profiling the likelihood across a range of values, to give a rough idea of its true value.

The individual score tests are calculated from the first partial derivatives of the log-likelihood, evaluated at the null hypothesis that $\boldsymbol{\beta}=\mathbf{0}$. For simplicity assume that no additional covariates are included. Let $U_g^{(i)} = \left(\frac{\partial \log L^{(i)}}{\partial \beta_g}, \frac{\partial \log L^{(i)}}{\partial \alpha_g} \right)^{\mathrm{T}}$ be the contribution of family *i* to the score for genotype *g*. Then the total score vector is $\mathbf{U} = \sum_i \mathbf{U}^{(i)}$ and an estimate of its variance-covariance matrix ("outer product of gradients estimator") is $\mathbf{V} = \sum_i \mathbf{U}^{(i)\mathrm{T}} \mathbf{U}^{(i)}$. Let $\mathbf{U}_\beta$ be the sub-vector of $\mathbf{U}$ corresponding to the parameters of interest $\boldsymbol{\beta}$, with $\mathbf{V}_{\beta\beta}$ be the corresponding sub-matrix of $\mathbf{V}$. Let $\mathbf{V}_{\beta\alpha}$, $\mathbf{V}_{\alpha\alpha}$ and $\mathbf{V}_{\alpha\beta}$ be the appropriate sub-matrices of $\mathbf{V}$ corresponding to the nuisance parameters $\boldsymbol{\alpha}$. Then the variance-covariance matrix for $\mathbf{U}_\beta$ is $\mathbf{V}_\beta = \mathbf{V}_{\beta\beta} - \mathbf{V}_{\beta\alpha}\mathbf{V}_{\alpha\alpha}^{-1}\mathbf{V}_{\alpha\beta}$ and the χ^2 score statistic for genotype *g* is $(\mathbf{U}_\beta)_g^2 (\mathbf{V}_\beta)_{gg}^{-1}$.

When evaluated at maximum-likelihood estimates of $\boldsymbol{\beta}$, the above expression for $\mathbf{V}_\beta$ is used to obtain 95% confidence intervals according to $\beta_g \pm 1.96(\mathbf{V}_\beta)_{gg}^{½}$. Note that the use of the outer-product of gradients estimator is less accurate in small samples than the Fisher information, and for this reason confidence intervals from UNPHASED may be more conservative than those from standard software when applied to complete data in small samples.

Haplotype coding

The above description is in terms of genotypes, for clarity. In fact UNPHASED deals primarily with haplotypes, although genotype tests are also implemented. Basically this replaces each genotype parameter by a sum of two haplotype parameters. This imposes some assumptions of independence between haplotypes: Hardy-Weinberg equilibrium, random mating, multiplicative odds ratio, and independent risk between siblings. However, owing to the likelihood factorisation above, inference on β is relatively unaffected when these assumptions are violated.

There are two paradigms for parameterising the haplotype odds ratios. *Haplotype coding* simply assigns a odds ratio parameter to each observed haplotype. *Allele coding* represents haplotype risks in terms of main effects of each locus and interactions between them. Haplotype coding is more intuitive to geneticists, but allele coding is more natural to statisticians. UNPHASED uses both versions as convenient.

Consider three SNPs with eight haplotypes. The two schemes would parameterise the log odds ratios as follows:

Haplotype	Haplotype coding	Allele coding
1-1-1	β_{111}	β_0
1-1-2	β_{112}	$\beta_0+\beta_{\text{--}2}$
1-2-1	β_{121}	$\beta_0+\beta_{\text{-}2\text{-}}$
1-2-2	β_{122}	$\beta_0+\beta_{\text{-}2\text{-}}+\beta_{\text{--}2}+\beta_{\text{-}22}$
2-1-1	β_{211}	$\beta_0+\beta_{2\text{--}}$
2-1-2	β_{212}	$\beta_0+\beta_{2\text{--}}+\beta_{\text{--}2}+\beta_{2\text{-}2}$
2-2-1	β_{221}	$\beta_0+\beta_{2\text{-}}+\beta_{2\text{-}}+\beta_{22\text{-}}$
2-2-2	β_{222}	$\beta_0+\beta_{2\text{--}}+\beta_{\text{-}2\text{-}}+\beta_{\text{--}2}+\beta_{22\text{-}}+\beta_{2\text{-}2}+\beta_{\text{-}22}+\beta_{222}$

Both schemes involve eight parameters. A baseline parameter must be set to 0 to ensure identifiability. Haplotype coding is the default in UNPHASED, and is called the **full** model. By dropping the interaction terms, allele coding allows models to be fitted with fewer parameters, which is particularly useful for the conditioning tests. The **allelemain** model in UNPHASED would code the eight haplotypes with just four parameters, a baseline and three main effects:

Haplotype	Allele main effects
1-1-1	β_0
1-1-2	$\beta_0+\beta_{\text{--}2}$
1-2-1	$\beta_0+\beta_{\text{-}2\text{-}}$
1-2-2	$\beta_0+\beta_{\text{-}2\text{-}}+\beta_{\text{--}2}$
2-1-1	$\beta_0+\beta_{2\text{--}}$
2-1-2	$\beta_0+\beta_{2\text{--}}+\beta_{\text{--}2}$

2-2-1	$\beta_0+\beta_{2--}+\beta_{-2-}$
2-2-2	$\beta_0+\beta_{2--}+\beta_{-2-}+\beta_{--2}$

This can lead to a global test with fewer degrees of freedom that the full model. When this model is fitted, UNPHASED displays the haplotype odds ratios estimated by the above model, and does not display the actual parameters estimated.

When there are conditioning markers as well as test markers, UNPHASED can mix the two coding schemes. The **haplomain** model defines a full model for the conditioning markers, and another full model for the test markers, but has no interaction terms between the two sets of markers. Considering now two conditioning SNPs and two test SNPS, the 16 haplotypes are coded as follows:

Haplotype	Haplotype main effects
1-1-1-1	β_0
1-1-1-2	$\beta_0+\beta_{--12}$
1-1-2-1	$\beta_0+\beta_{--21}$
1-1-2-2	$\beta_0+\beta_{--22}$
1-2-1-1	$\beta_0+\beta_{12--}$
1-2-1-2	$\beta_0+\beta_{12--}+\beta_{--12}$
1-2-2-1	$\beta_0+\beta_{12--}+\beta_{--21}$
1-2-2-2	$\beta_0+\beta_{12--}+\beta_{--22}$
2-1-1-1	$\beta_0+\beta_{21--}$
2-1-1-2	$\beta_0+\beta_{21--}+\beta_{--12}$
2-1-2-1	$\beta_0+\beta_{21--}+\beta_{--21}$
2-1-2-2	$\beta_0+\beta_{21--}+\beta_{--22}$
2-2-1-1	$\beta_0+\beta_{22--}$
2-2-1-2	$\beta_0+\beta_{22--}+\beta_{--12}$
2-2-2-1	$\beta_0+\beta_{22--}+\beta_{--21}$
2-2-2-2	$\beta_0+\beta_{22--}+\beta_{--22}$

So this coding has 7 parameters instead of 16 in the full model. In this case the **haplomain** test would have 3df corresponding to the free parameters for the test haplotype. If the **allelemain** model is used, there would be 2df, reflecting the allele coding for the test markers. If the **full** model is used, there would be 3df for each of the four conditioning haplotypes, resulting in 12df in total. The **gxg** test compares the haplotype main effects model to the full model. Here the test would have 9df.

Differences from version 2.4

This version of UNPHASED is almost a complete rewrite of the previous version, 2.4. The major differences are as follows:

- Merging of TDTPHASE, COCAPHASE and QTPHASE into a single program that can analyse combined samples of trios and unrelateds.
- New likelihood used in TDTPHASE that is more robust to deviation from HWE. It reduces to the TDT when data are complete, whereas the old TDTPHASE reduced to the HHRR when the –EM option was used.
- Robustness to linkage in nuclear families with multiple siblings. Previously this was only available with the –robustperm option.
- Outputs confidence intervals for the odds ratio.
- Genotype tests now implemented.
- Basic support for covariates.
- Score tests for –individual option, instead of likelihood ratio test.
- Automated analysis of all combinations of a fixed number of markers, and automated analysis of all window sizes.
- QTDT format data file.
- Test for gene-gene interaction.
- New graphical user interface.
- -EM dropped and –certain introduced instead.
- -bygenotype renamed to –condgenotype.
- -cellcounts renamed to –cellcount.
- -conditiontype renamed to –condspecific.
- -hpt dropped.
- -maineffects and –nointeraction renamed to –model haplomain and –model allelemain.
- -observed dropped.
- -onesib dropped.
- -paraff and –sibaff dropped.
- -parsex and –sibsex dropped and now used as covariate names.
- -phenotype renamed to –trait.
- -robustperm dropped.