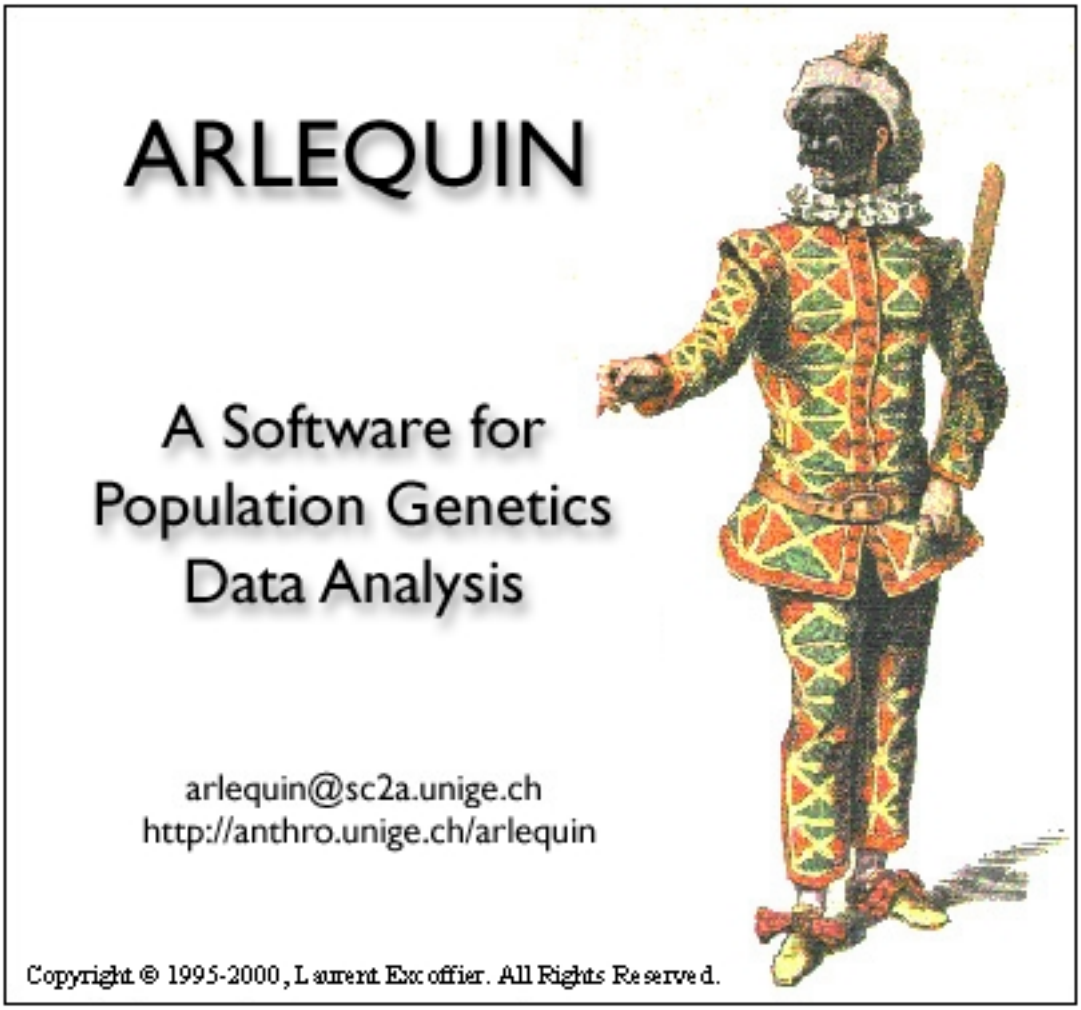

ARLEQUIN
A Software for
Population Genetics
Data Analysis
arlequin@sc2a.unige.ch
http://anthro.unige.ch/arlequin
Copyright © 1995-2000, Laurent Excoffier. All Rights Reserved.

Ver 2.000

ARLEQUIN ver 2.000

A software for population genetics data analysis

Authors:
Stefan Schneider, David Roessli, and Laurent Excoffier

Genetics and Biometry Laboratory
Dept. of Anthropology and Ecology
University of Geneva
CP 511
1211 Geneva 24
Switzerland

E-mail : arlequin@sc2a.unige.ch
URL: http://anthro.unige.ch/arlequin

March 2000

Table of contents:

1 INTRODUCTION

1.1 Why Arlequin?

Arlequin is the French translation of "Arlecchino", a famous character of the Italian "Commedia dell'Arte". As a character he has many aspects, but he has the ability to switch among them very easily according to its needs and to necessities. This polymorphic ability is symbolized by his colorful costume, from which the Arlequin icon was designed.

1.2 Arlequin philosophy

The goal of Arlequin is to provide the average user in population genetics with quite a large set of methods and statistical tests, in order to extract information on genetic and demographic features of a collection of population samples.

The Java graphical interface has been improved and designed to allow the user to rapidly select the different analyses he wants to perform on his data. We felt important to be able to explore the data, to analyze several times the same data set from different perspectives, with different selected options.

The core routines that make all computing intensive tasks are in the executable arlecore.exe. This core program is written in an "portable" C++ and recompiled for the different platforms. As a C++ compiler we used Borland C++ 5.01 on Win/Intel, Code Warrior 4 on Mac OS/Power PC and gcc 2.7.2 on Linux/Intel. The graphical interface and the core program exchange information through ASCII text files. The approach of separating the interface from the core program seemed to us the best way to combine the speed of C++ and the portability of Java.

The statistical tests implemented in Arlequin have been chosen such as to minimize hidden assumptions and to be as powerful as possible. Thus, they often take the form of either permutation tests or exact tests, with some exceptions.

Finally, we wanted Arlequin to be able to handle genetic data under many different forms, and to try to carry out the same types of analyses irrespective of the format of the data.

Because Arlequin has a rich set of features and many options, it means that the user has to spend some time in learning them. However, we hope that the learning curve will not be that steep.

Arlequin is made available free of charge, as long as we have enough local resources to support the development of the program.

1.3 About this manual

The main purpose of this manual is to allow you to use Arlequin on your own, **in order to limit as far as possible e-mail exchange with us**.

In this manual, we have tried to provide a description of

1. the data types handled by Arlequin
2. the way these data should be formatted before the analyses
3. the graphical interface
4. the impact of different options on the computations
5. methodological outlines describing which computations are actually performed by Arlequin.

Even though this manual contains the description of some theoretical aspects, it should not be considered as a textbook in basic population genetics. **We strongly recommend you to consult the original references provided with the description of a given method if you are in doubt with any aspect of the analysis**.

1.4 Data types handled by Arlequin

Arlequin can handle several types of data either in *haplotypic* or *genotypic* form. The basic data types are:

- DNA sequences
- RFLP data
- Microsatellite data
- Standard data
- Allele frequency data

By *haplotypic form* we mean that genetic data can be presented under the form of haplotypes (i.e. a combination of alleles at one or more loci). This haplotypic form can result from the analyses of haploid genomes (mtDNA, Y chromosome, prokaryotes), or from diploid genomes where the gametic phase could be inferred by one way or another. Note that allelic data are treated here as a single locus haplotype.

Ex 1: Haplotypic RFLP data : `1001101001010010 10`
Ex 2: Haplotypic standard HLA data : `DRB1*0101 DQB1*0102 DPB1*0201`

By *genotypic form*, we mean that genetic data is presented under the form of diploid genotypes (i.e. a combination of pairs of alleles at one or more loci). Each genotype is entered on two separate lines, with the two alleles of each locus being on a different line.

Ex1: Genotypic DNA sequence data:

```
ACGGCATTTAAGCATGACATACGGATTGACA
ACGGGATTTTAGCATGACATTCGGATAGACA
```

Ex 2: Genotypic Microsatellite data:

```
63    24    32
62    24    30
```

The gametic phase of a multi-locus genotype may be either known or unknown. If the gametic phase is known, the genotype can be considered as made up of two well-defined haplotypes. For genotypic data with unknown gametic phase, you can consider the two alleles present at each locus as codominant, or you can allow for the presence of a recessive allele. This gives finally four possible forms of genetic data:

- Haplotypic data,
- Genotypic data with known gametic phase,
- Genotypic data with unknown gametic phase (no recessive alleles)
- Genotypic data with unknown gametic phase (recessive alleles).

1.4.1 DNA sequences

Arlequin can accommodate DNA sequences of arbitrary length. Each nucleotide is considered as a distinct locus. The four nucleotides "C", "T", "A", "G" are considered as unambiguous alleles for each locus, and the "-" is used to indicate a deleted nucleotide. Usually the question mark "?" codes for an unknown nucleotide. The following notation for ambiguous nucleotides are also recognized:

R: A/G (purine)

Y: C/T (pyrimidine)

M: A/C

W: A/T

S: C/G

K: G/T

B: C/G/T

D: A/G/T

H: A/C/T

V: A/C/G

N: A/C/G/T

1.4.2 RFLP Data

Arlequin can handle RFLP haplotypes of arbitrary length. Each restriction site is considered as a distinct locus. The presence of a restriction site should be coded as a "1", and its absence as a "0". The "-" character should be used to denote the deletion of a site, not its absence due to a point mutation.

1.4.3 Microsatellite data

The raw data consist here of the allelic state of one or an arbitrary number of microsatellite loci. For each locus, one should **provide the number of repeats of the microsatellite motif** as the allelic definition, if one wants his data to be analyzed according to the step-wise mutation model (for the analysis of genetic structure). It may occur that the absolute number of repeats is unknown. If the difference in length between amplified products is the direct consequence of changes in repeat numbers, then the minimum length of the amplified product could serve as a reference, allowing to code the other alleles in terms of additional repeats as compared to this reference. If this strategy is impossible, then any other number could be used as an allelic code, but the stepwise mutation model could not be assumed for these data.

1.4.4 Standard data

Data for which the molecular basis of the polymorphism is not particularly defined, or when different alleles are considered as mutationally equidistant from each other. Standard data haplotypes are thus compared for their content at each locus, without taking special care about the nature of the alleles, which can be either similar or different. For instance, HLA data (human MHC) enters the category of standard data.

1.4.5 Allele frequency data

The raw data consist of only allele frequencies (mono-locus treatment), so that no haplotypic information is needed for such data. Population samples are then only compared for their allelic frequencies.

1.5 Methods implemented in Arlequin

The analyses Arlequin can perform on the data fall into two main categories: intra-population and inter-population methods. In the first category statistical information is extracted independently from each population, whereas in the second category, samples are compared to each other.

Intra-population methods:	***Short description:***
Standard indices	Some diversity measures like the number of polymorphic sites, gene diversity.
Molecular diversity	Calculates several diversity indices like nucleotide diversity, different estimators of the population parameter θ.
Mismatch distribution	The distribution of the number of pairwise differences between haplotypes based on computed inter-haplotypic distances.
Haplotype frequency estimation	Estimates the frequency of haplotypes present in the population either by gene counting or by the maximum likelihood method, depending on the type of data (haplotypic or genotypic).
Linkage disequilibrium	Test of non-random association of alleles at different loci.
Hardy-Weinberg equilibrium	Test of non-random association of alleles within diploid individuals.
Tajima's neutrality test (infinite site model)	Test of the selective neutrality of a random sample of DNA sequences or RFLP haplotypes under the infinite site model.
Fu's F_S neutrality test (infinite site model)	Test of the selective neutrality of a random sample of DNA sequences or RFLP haplotypes under the infinite site model.
Ewens-Watterson neutrality test (infinite allele model)	Tests of selective neutrality based on Ewens sampling theory under the infinite alleles model.
Chakraborty's amalgamation test (infinite allele model)	A test of selective neutrality and population homogeneity. This test can be used when sample heterogeneity is suspected.
Minimum Spanning Network (MSN)	Computes a Minimum Spanning Tree (MST) and Network (MSN) among haplotypes. This tree can also be computed for all the haplotypes found in different populations if activated under the AMOVA section.

Inter-population methods:	***Short description:***
Search for shared haplotypes between populations	Comparison of population samples for their haplotypic content. All the results are then summarized in a table.
AMOVA	Different hierarchical Analyses of MOlecular VAriance to evaluate the amount of population genetic structure.
Pairwise genetic distances	F_{ST} based genetic distances for short divergence time.
Exact test of population differentiation	Test of non-random distribution of haplotypes into population samples under the hypothesis of panmixia.
Assignment test of genotypes	Assignment of individual genotypes to particular populations according to estimated allele frequencies.
Mantel test:	***Short description:***
Correlations or partial correlations between a set of 2 or 3 matrices	Can be used to test for the presence of **isolation-by-distance**

1.6 System requirements

- Windows 95/98/NT/Intel, MacOS/(Power PC), or Linux/Intel.
- A minimum of 16 MB RAM, and more to avoid swapping.
- At least 10Mb free hard disk space.

1.7 Installing and uninstalling Arlequin

1.7.1 Win9X/NT installation

1.7.1.1 Archive file description

arlequin20jre_zip.exe	A self-extracting archive containing the two files below
arlequin20_zip.exe	A self-extracting archive containing all files necessary for Arlequin to work
jre117-win32_zip.exe	Setup file for installing the Java runtime environment (ver. 1.1.7.) *Only required if you have no Java runtime environment (ver. 1.1.7 or higher) already installed on your system*

1.7.1.2 Software installation

- **After the download of arlequin20jre_zip.exe, proceed as follows:**
  1. Execute the self-extracting archive *arlequin20jre_zip.exe*.
  2. Select a *temporary* folder where the two files *arlequin20_zip.exe* and *jre117-win32_zip.exe* will be extracted. To start the extraction, press the *Unzip* button.
3. The temporary folder will contain two files: arlequin20.exe, a self-extracting archive to install Arlequin ver. 2.000 and jre117-win32.exe the Java runtime environment installer.

- **After the download or extraction of jre117-win32_zip.exe, proceed as follows:**

1. Install the Java runtime environment (JRE) 1.1.7 on your system by executing the file jre117-win32.exe.
2. Follow the instructions on the screen. More information and more recent equivalent software may be found on the JavaSoft home page (http://www.javasoft.com/products/jdk/1.1/jre/download-jre-windows.html).

- **After the download or extraction of the self extractable archive arlequin2_.zip.exe proceed as follows:**

1. Execute the self-extracting archive *arlequin20_zip.exe*.
2. Select the folder where you wish to install Arlequin (for example "C:\ Program File\" using the Browse... button see upper figure). A folder called ArlequinFolder will be created containing all necessary files to make Arlequin work.
3. Start Arlequin by double-clicking on the arlequin.exe program, or on an alias you have created. You can also run the arlequin.bat batch file to start Arlequin. Note that the Java runtime environment has to be installed before you can run Arlequin. If it does not work, try to execute the batch file run_jre.bat in a Dos shell to see if something is wrong with the Java runtime environment.

- **Then start Arlequin in one of the following ways:**

1. Simply double-click on the *Arlequin.exe* icon or on its alias.
2. If you want to start Arlequin from a DOS shell, type *jre -cp arlequin.jar; -cp swingall.jar; arlequin.ArlequinApp* after the prompt.

3. Double click on the *run_jre.bat* file, which contains the command listed in point 2.

1.7.1.3 Win9X/NT uninstallation

Simply delete the directory where you installed Arlequin. The registries were not modified by the installation of Arlequin.

1.7.2 Mac OS installation

1.7.2.1 Archive file description

Arlequin2_Mrj2.1.sea.hqx	A bin-hex encoded self-extracting archive containing the two files below
Arlequin2.sea.hqx	A bin-hex encoded archive containing all files necessary for Arlequin to work
MRJ 2.1.4.smi.sit	Setup file for installing the Java runtime environment (ver. 1.1.7.) *Only required if you have no Java runtime environment (ver. 1.1.7 or higher) already installed on your system*

1.7.2.2 Software Installation.

Depending on which of the three files you did download, follow the procedures described below:

- **After the download of Arlequin2_Mrj2.1.sea.hqx, proceed as follows:**

1. Decode and unstuff the archive by dropping it into *Stuffit Expander* from Aladdin software (http://www.aladdinsys.com/).

2. A folder called Arlequin Installer will be created containing the two files *Arlequin2.sea.hqx* and *MRJ 2.1.4.smi.sit*. To continue the installation, proceed as described in the sections below.

- **After the download or extraction of MRJ 2.1.4.smi.sit, proceed as follows:**
  1. Drop *MRJ 2.1.4.smi.sit* into stuffit expander. An executable file *MRJ 2.1.4.smi* will be extracted. Execute it and follow the on screen instructions .
- **After the download or extraction of Arlequin2.sea.hqx proceed as follows:**
  1. Drop *Arlequin2.sea.hqx* onto *Stuffit expander*.
  2. A folder called *ArlequinFolder* will be created containing all necessary files for Arlequin to work. Drag this folder at the location where you wish to have Arlequin installed. After using Arlequin do not move this folder anymore .
  3. Start Arlequin by double-clicking on the arlequin program, or on an alias you have created. Normally a message like "*Starting Java*" should appear, and the Arlequin Java interface should show up.

1.7.2.3 Mac OS uninstallation:

Simply delete the directory where you installed Arlequin.

1.7.3 Linux istallation

1.7.3.1 Software installation

Step 1: Install the Java runtime environment (jre) (version 1.1.7 or higher) on your system by:

- If you have the Red Hat 6 Linux version, download the *jre_glibc.gz* file, otherwise go to *http://www.blackdown.org* for other ports and more documentation.
- Copy the file *jre_glibc.gz* or its equivalent into the directory /usr/local.
- Decompress the archive with the command : *tar -xvzf jre_glibc.gz*.
- A directory *jre_117_v3* containing all necessary files will be created.
- In the file "*.bshcr*" (initialization of the bash shell) add the following line:
  export PATH=/usr/local/jre_117_v3/bin:$PATH
  so that the shell knows where the jre file lies.

Step 2:

- After downloading the file *arlequin20b.tar* from our web site, *Untar* it with the command *tar -xvzf arlequin20b.tar*.
- A directory called *ArlequinFolder* will be created, containing *Arlequin.jar* , the executable version of Arlequin Ver. 2.0b and all-necessary files to make the software run properly.
- Move this folder at any location of your hard disk. Note that once you have chosen a location for the software, you should not change it, or it would become impossible to correctly browse the result files.
- To start Arlequin 2.0b, double-click on the file called *arlequin.bat*. If it does not work try either to launch *arlequin.bat* from a shell to see any error message, or in a shell type the full command :
  jre -cp arlequin.jar -cp swingall.jar arlequin.ArlequinApp

1.7.3.2 Linux uninstallation

Delete the directory where you installed Arlequin at step 2 of the installation process.

1.8 List of files included in the Arlequin package

Files	Description	Required by Arlequin to run properly
Arlequin files		
Arlequin.exe	Arlequin executable file. It launches the graphical interface written in Java.	✓
Arlequin.pdf	Arlequin user manual in pfd format	
Arlequin.jar	Java byte code archive containing the code for the graphical user interface.	✓
Arlecore.exe	The core executable that makes all computations. This collection of core routines is written in C++.	✓
Lax.jar	Java byte code used by the Java virtual machine.	✓
Swingall.jar	Java byte code archive for using the Java swing components.	✓
Arlequin.ini	A file containing the description of the last custom settings defined by the user	
Ftiens4.js and 12 gif files	ftiens4.js contains the Java script that allows the browsing of the result HTML files. This script needs gif files.	✓
Icones/	This folder contains all icons used by the graphical interface.	✓
Readme20.txt	A text file containing a description of the last release of Arlequin	

Example files in subdirectory datafiles

Batch\batch_ex.arb
Batch\amova1.arp
Batch\amova1.ars
Batch\amova2.arp
Batch\amova2.ars
Batch\amova1mat.dis
Batch\genotsta.arp
Batch\genotsta.ars
Batch\microsat.arp
Batch\microsat.ars
Batch\missdata.arp
Batch\missdata.ars
Batch\phenohla.arp
Batch\phenohla.ars
Batch\relfreq.arp
Batch\relfreq.ars
Batch\indlevel.arp
Batch\indlevel.ars

Microsat\2popmic.arp
Microsat\2popmic.ars
Microsat\micdipl.arp
Microsat\micdipl.ars
Microsat\micdipl2.arp
Microsat\micdipl2.ars

Dna\mtdna_hv1.arp
Dna\mtdna_hv1.ars
Dna\nucl_div.arp
Dna\nucl_div.ars

Neutrtst\chak_tst.arp
Neutrtst\chak_tst.ars
Neutrtst\ew_watt.arp
Neutrtst\ew_watt.ars
Neutrtst\Fu_s_test.arp
Neutrtst\Fu_s_test.ars

Haplfreq\hla_7pop.arp
Haplfreq\hla_7pop.ars

Amova\amovahap.arp
Amova\amovahap.ars
Amova\amovadis.arp
Amova\amovadis.ars
Amova\56hapdef.txt
Amova\amovadis.dis

Disequil\hwequil.arp
Disequil\hwequil.ars
Disequil\ld_gen0.arp
Disequil\ld_gen0.ars
Disequil\ld_gen1.arp
Disequil\ld_gen1.ars
Disequil\ld_hap.arp
Disequil\ ld_hap.ars

Mantel\custom_corr3mat.arp
Mantel\custom_corr3mat.ars
Mantel\fst_corr.arp
Mantel\fst_corr.ars
Mantelfst_partial_corr.arp
Mantel\fst_partial_corr.ars

Conversion\gene_pop1.gpp

Freqncy\cohen.arp
Freqncy\cohen.ars

1.9 Arlequin computing limitations

The amount of data that Arlequin can handle mostly depends on the memory available on your computer. However, a few parameters are limited to values within the range shown below.

Portions of Arlequin concerned by the limitations	Limited parameter	Maximum value
All	Number of population samples	1000
All	Number of groups of populations	1000
Ewens-Watterson and Chakraborty's neutrality tests	Sample size	2000
Ewens-Watterson and Chakraborty's neutrality tests	Number of haplotypes	1000

1.10 Arlequin platforms specificities

1.10.1 Windows 9X/NT

This is our standard development version. We first develop for that version and then port it to other versions so that it should be the version with most features.

1.10.2 Linux/PC

- Nothing special if you install it properly.

1.10.3 MacOS/PowerMac

- The console window messages do not appear in the message bar of the Arlequin Java interface. However, they correctly appear in the console window.
- The console window launched by the interface needs to be manually closed (-Q) at the end of each run.
- The browser is not automatically launched at the end of the computations. It has to be launched manually from the interface. Press the *Browse Result* button in the *Project* tab.

1.11 How to cite Arlequin

Stefan Schneider, David Roessli, and Laurent Excoffier (2000) Arlequin ver. 2.000: A software for population genetics data analysis. Genetics and Biometry Laboratory, University of Geneva, Switzerland.

1.12 Acknowledgements

This program has been made possible by Swiss NSF grants No. 32-37821-93 and No 32.047053.96

Many thanks to:

André Langaney, Jean-Marc Kuffer,Yannis Michalakis, Thierry Pun, Montgomery Slatkin, Peter Smouse, Alicia Sanchez-Mazas, Isabelle Dupanloup de Ceunink, Estella Poloni, Mathias Currat, Giorgio Bertorelle, Michele Belledi, Evelyne Heyer, Erika Bucheli, Alex Widmer, Philippe Jarne, Frédérique Viard, Peter de Knijff, Peter Beerli, Matthew Hurles, Rosalind Harding, Frank Struyf, A.J. Gharrett, Jennifer Ovenden, Steve Carr, Marc Allard, Omar Chassin, Alonso Santos, Oscar Gaggiotti, John Novembre, Nicolas Ray, Nelson Fagundes, Eric Minch, Pierre Darlu, Jérôme Goudet, François Balloux, Eric Petit, Ettore Randi, Natacha Mesquita, David Foltz, Guoqing Lu, Tomas Hrbek, Corinne Zeroual, Rod Norman, Chew-Kiat Heng, Russell Pfau, April Harlin, S Kark, Jenny Ovenden, Jill Shanahan, and all the other users or beta-testers of Arlequin that have sent us their comments.

1.13 Bug report and comments

Please report any bug through the bug report form available on

http://anthro.unige.ch/arlequin/bug-report.html

Other comments and suggestions will be also appreciated and can be communicated to us using the same web page.

1.14 How to get the last version of the Arlequin software?

Arlequin will be updated regularly and can be freely retrieved on

http://anthro.unige.ch/arlequin

1.15 What is new in version 2.000 compared to version 1.1

New features:

1. Several bug corrections (see the list on our web site).
2. An entirely redesigned user interface written in Java.
3. New versions for the MacOS on GX/Power PC processors and for the Linux OS on Intel platforms.
4. Output a different log file for each project instead of having a single log file for all projects. The log file is now put in the result directory that has the same name as the selected project, but with the [.res] extension.
5. Result files are now only accessible through a web browser. One can easily navigate between different sets of results and between portions of the result files through a document tree appearing in a separate pane. The web browser is automatically launched after each run of Arlequin.
6. Custom text editor and web browser have to be selected for having access to the project and result files, respectively. A text editor for the edition of Arlequin projects is not provided any more.
7. Several bug corrections (see the list on our web site).
8. Mismatch distributions are now fitted to the observations by a generalized-least square procedure instead of using a moment estimator. We now compute a test statistic of the validity of the estimated demographic expansion. We also provide confidence regions around the expected mismatch that contain a given percentage of the mismatch distributions simulated around the estimation.

9. Implementation of Fu's F_S test of selective neutrality.
10. AMOVA analyses can now be performed separately for each locus.
11. Computation of the mean number of pairwise differences within and between populations. Raw and corrected distances are available at the inter-population level.
12. Computation of genetic distances between populations that take into account potential differences in effective population sizes
13. Computation of a minimum spanning network from a matrix of distances between haplotypes.
14. Mantel tests. Computes the correlation or partial correlations between 2 or 3 matrices, and test their significance by permuting rows and columns in one or two matrices.
15. Assignment test of genotypes to populations. For each genotype in the sample, we compute its likelihood assuming that it belongs to different populations.
16. The references of each method used by Arlequin in a particular computation are now written directly in the result file.

1.16 Forthcoming developments

- Treatment of pure dominant data (RAPD, AFLP). Still on our To Do list...
- Incorporation of additional population genetics methods.

Suggestions are welcome, but we only have one life...

1.17 Remaining problems

- Some refreshment problem with Java interface
- Messages appear in console window for the Mac version. When the computations are finished, the console window remains open, and has to be closed by the user.

.

2 GETTING STARTED

The first thing to do before running Arlequin for the first time is certainly to read the manual or consult the help file. They will provide you with most of the information you are looking for. So, take some time to read them before you seriously start analyzing your data.

2.1 Preparing input files

The first step for the analysis of your data is to prepare an input data file for Arlequin. This input file is called here a *project file*. As Arlequin is quite a versatile program able to analyze several data types, you have to include some information about the properties of your data in the project file together with the raw data.

There are two ways to create Arlequin projects:

1) You can start from scratch and use a text editor to define your data using reserved keywords.
2) You can use Arlequin's "Project Outline Wizard" by selecting the tab panel *Project Wizard*. The controls on this tab panel allow you to specify the type of project outline that should be build. Once all settings done, the project outline is loaded into Arlequin by clicking on the button *open outline as project*. The name of the data file should have a "**.arp*" extension (for ARlequin Project).

2.2 Loading project files into Arlequin

Once the project file is built, you must load it into Arlequin. You can do this either by activating the menu *File | Open*, or by clicking on the Open project button on the toolbar. The Arlequin project files must have the **.arp* extension. If your project file is not valid, Arlequin will open the Arlequin Log file in the text editor you selected in the *Configuration* tab panel to help you pointing out the problems. For each project, Arlequin creates a log file called *Arlequin_log.txt*, where warnings and error messages are issued. The log file also keeps track of all the operations performed during an Arlequin session. This file is located in the result directory of the project file.

If your project file is valid, its main properties will be shown in the Project Tab.

At this point, you just have to choose which analyses to perform on your data by selecting the *Calculation Settings* Tab.

2.3 Selecting analyses to be performed on your data

The different settings can by tuned by first selecting the *Calculation Settings* Tab. Navigate the tree on the left upper part of the panel to select the group of controls you whish the set up. Depending on your selection the lower part of the panel is updated.

2.4 Creating and using Setting Files

By settings we mean any alternative choice that can be made when using Arlequin. As you can choose different types of analyses, as well as different options for each of these analyses, all these choices can be saved into setting files. These files generally take the same name as the project files, but with the extension **.ars*. Setting files can be created at any time of your work by clicking on the save button right to the tree. Alternatively, if you activate the *use*

associated settings checkbox, the last used settings used on this project will be automatically saved when you close the project and reloaded when you open it later again. The setting are stored in a file having same name as the project file, and the *.ars* extension. These setting files are convenient when you want to repeat some analyses done previously, or when you want to make different types of computations on several projects, as it is possible using batch files (see section 3.6) giving you considerable flexibility on the analyses you can perform, and avoiding tedious and repetitive mouse-clicks.

2.5 Performing the analyses

The selected analyses can be performed either by clicking on the Run. If an error occurs in the course of the execution, Arlequin will write diagnostic information in the log file. If the error is not too severe, Arlequin will open the log file for you. If there is a memory error, Arlequin will shut down itself. In the latter case, you should consult the Arlequin log file ***before*** launching a new analysis in order to get some information on where or at which stage of the execution the problem occurred. The file *Arlequin_log.txt* is located in the project results directory.

2.6 Stopping the computations

The computations can be stopped at any time by pressing the Stop button on the toolbar. However, note that the results may be incorrect if the computations did not terminate normally.

For very large project files, you may have to wait for a few seconds before the calculations are stopped.

2.7 Consulting the results

When the calculations are over, Arlequin will create a result directory, which has the same name as the project file, but with the **.res* extension. This directory contains all the result files, particularly the main result file with the same name as the project file, but with the **.htm* extension. The main result file can be viewed in any html browser, as specified in the *Configuration* tab panel. If a valid path to a web browser has been selected in the *Configuration* tab panel, the web browser, the result file *[project name]_main.html* is automatically loaded in the configured html browser after the end of the computations. You can also view your results at anytime by clicking on the *Browse results* button.

3 INPUT FILES

3.1 Format of Arlequin input files

Arlequin input files are also called project files. The project files contain both descriptions of the properties of the data, as well as the raw data themselves. The project file may also refer to one or more external data files.
Note that comments beginning by a "#" character can be put anywhere in the Arlequin project file. Everything that follows the "#"character will be ignored until an end of line character.

3.2 Project file structure

Input files are structured into two main sections with additional subsections that must appear in the following order:

1) Profile section (mandatory)
2) Data section (mandatory)
   - 2a) Haplotype list (optional)
   - 2b) Distance matrices (optional)
   - 2c) Samples (mandatory)
   - 2d) Genetic structure (optional)
   - 2e) Mantel tests (optional)

We now describe the content of each (sub-) section in more detail.

3.2.1 Profile section

The properties of the data must be described in this section. The beginning of the profile section is indicated by the keyword [Profile] (within brackets).

One must also specify

- *The title of the current project* (used to describe the current analysis)

  Notation: **Title**=

  Possible value: Any string of characters within double quotes

  Example: `Title="An analysis of haplotype frequencies in 2 populations"`

- *The number of samples or populations present in the current project*

  Notation: **NbSamples** =

  Possible values: Any integer number between 1 and 1000.

  Example: `NbSamples =3`

- *The type of data to be analyzed.* Only one type of data is allowed per project

  Notation: **DataType** =

  Possible values: DNA, RFLP, MICROSAT, STANDARD and FREQUENCY

  Example: `DataType = DNA`

- *If the current project deals with haplotypic or genotypic data*

  Notation: **GenotypicData** =

  Possible values: 0 (haplotypic data), 1 (genotypic data)

  Example: `GenotypicData = 0`

One can also optionally specify

- *The character used to separate the alleles at different loci* (the locus separator)

  Notation: **LocusSeparator** =

  Possible values: WHITESPACE, TAB, NONE, or any character other than "#", or the character specifying missing data.

  Example: `LocusSeparator = TAB`

  Default value: WHITESPACE

- *If the gametic phase of genotypes is known*

  Notation: **GameticPhase** =

  Possible values: 0 (gametic phase not known), 1 (known gametic phase)

  Example: `GameticPhase = 1`

  Default value: 1

- *If the genotypic data present a recessive allele*

  Notation: **RecessiveData** =

  Possible values: 0 (co-dominant data), 1 (recessive data)

  Example: `RecessiveData =1`

  Default value: 0

- *The code for the recessive allele*

  Notation: **RecessiveAllele** =

  Possible values: Any string of characters within double quotes. This string can be explicitly used in the input file to indicate the occurrence of a recessive homozygote at one or several loci.

  Example: `RecessiveAllele ="xxx"`

  Default value: "null"

- *The character used to code for missing data*

  Notation: **MissingData** =

  Possible values: A character used to specify the code for missing data, entered between single or double quotes.

  Example: `MissingData =’$’`

  Default value: `’?’`

- *If haplotype or phenotype frequencies are entered as absolute or relative values*

  Notation: **Frequency** =

Possible values: ABS (absolute values), REL (relative values: absolute values will be found by multiplying the relative frequencies by the sample sizes)

Example: `Frequency = ABS`

Default value: ABS

- *If a distance matrix needs to be computed from the original data, when calculating genetic structure indices*

  Notation: **CompDistMatrix** =

  Possible values: 0 (use the distance matrix specified in the DistanceMatrix sub-section), 1 (compute distance matrix from haplotypic information)

  Example: `CompDistMatrix = 1`

  Default value: 0

- *The number of significant digits for haplotype frequency outputs*

  Notation: **FrequencyThreshold** =

  Possible values: A real number between 1e-2 and 1e-7

  Example: `FrequencyThreshold = 0.00001`

  Default value: 1e-5

- *The convergence criterion for the EM algorithm used to estimate haplotype frequencies and linkage disequilibrium from genotypic data*

  Notation: **EpsilonValue** =

  Possible values: A real number between 1e-7 and 1e-12.

  Example: `EpsilonValue = 1e-10`

  Default value: 1e-7

3.2.2 Data section

This section contains the raw data to be analyzed. The beginning of the profile section is indicated by the keyword [Data] (within brackets).
It contains several sub-sections:

3.2.2.1 Haplotype list (optional)

In this sub-section, one can define a list of the haplotypes that are used for all samples. This section is most useful in order to avoid repeating the allelic content of the haplotypes present in the samples. For instance, it can be tedious to write a full sequence of several hundreds of nucleotides next to each haplotype in each sample. It is much easier to assign an identifier to a given DNA sequence in the haplotype list, and then use this identifier in the sample data section. This way Arlequin will know exactly the DNA sequences associated to each haplotype.

However, this section is optional. The haplotypes can be fully defined in the sample data section.
An identifier and a combination of alleles at different loci (one or more) describe a given haplotype. The locus separator defined in the profile section must separate each adjacent allele from each other.

It is also possible to have the definition of the haplotypes in an external file. Use the keyword EXTERN followed by the name of the file containing the definition of the haplotypes. Read Example 2 to see how to proceed. If the file "*hapl_file.hap*" contains exactly what is between the braces of Example 1, the two haplotype lists are equivalent.

Example 1:

```
[[HaplotypeDefinition]]  #start the section of Haplotype definition
  HaplListName="list1"  #give any name you whish to this list
  HaplList={
      h1  A T     #on each line, the name of the haplotype is
      h2  G C     # followed by its definition.
      h3  A G
      h4  A A
      h5  G G
  }
```

Example 2:

```
[[HaplotypeDefinition]] #start the section of Haplotype definition
  HaplListName="list1"  #give any name you whish to this list
  HaplList = EXTERN "hapl_file.hap"
```

3.2.2.2 Distance matrix (optional)

Here, a matrix of genetic distances between haplotypes can be specified. This section is here to provide some compatibility with earlier WINAMOVA files. The distance matrix must be a lower diagonal with zeroes on the diagonal. This distance matrix will be used to compute the genetic structure specified in the genetic structure section. As specified in AMOVA, the elements of the matrix should be squared Euclidean distances. In practice, they are an evaluation of the number of mutational steps between pairs of haplotypes.

One also has to provide the labels of the haplotypes for which the distances are computed. The order of these labels must correspond to the order of rows and columns of the distance matrix. If a haplotype list is also provided in the project, the labels and their order should be the same as those given for the haplotype list.

Usually, it will be much more convenient to let Arlequin compute the distance matrix by itself.

It is also possible to have the definition of the distance matrix given in an external file. Use the keyword EXTERN followed by the name of the file containing the definition of the matrix. Read Example 2 to see how to proceed.

Example 1:

```
[[DistanceMatrix]]   #start the distance matrix definition section
  MatrixName= "none"  # name of the distance matrix
  MatrixSize= 4       # size = number of lines of the distance matrix
  MatrixData={
       h1 h2 h3 h4  # labels of the distance matrix (identifier of the
         0.00000    # haplotypes)
         2.00000   0.00000
         1.00000   2.00000   0.00000
         1.00000   2.00000   1.00000   0.00000
  }
```

Example2:

```
[[DistanceMatrix]]     #start the distance matrix definition section
  MatrixName= "none"  # name of the distance matrix
```

```
MatrixSize= 4       # size = number of lines of the distance matrix
MatrixData= EXTERN "mat_file.dis"
```

3.2.2.3 Samples

In this obligatory sub-section, one defines the haplotypic or genotypic content of the different samples to be analyzed.

Each sample definition begins by the keyword SampleName and ends after a SampleData has been defined.

One must specify:

- *A name for each sample*

  Notation: **SampleName** =

  Possible values: Any string of characters within quotes.

  Example: `SampleName= "A first example of a sample name"`

  Note: This name will be used in the Structure sub-section to identify the different samples, which are part of a given genetic structure to test.

- *The size of the sample*

  Notation: **SampleSize** =

  Possible values: Any integer value.

  Example: `SampleSize=732`

  Note: For haplotypic data, the sample size is equal to the haploid sample size. For genotypic data, the sample size should be equal to the number of diploid individuals present in the sample. When absolute frequencies are entered, the size of each sample will be checked against the sum of all haplotypic frequencies will check. If a discrepancy is found, a *Warning message* is issued in the log file, and the sample size is set to the sum of haplotype frequencies. When relative frequencies are specified, no such check is possible, and the sample size is used to convert relative frequencies to absolute frequencies.

- *The data itself*

  Notation: **SampleData** =

  Possible values: A list of haplotypes or genotypes and their frequencies as found in the sample, entered within braces

  Example:

```
SampleData={
 id1 1  ACGGTGTCGA
 id2 2  ACGGTGTCAG
 id3 8  ACGGTGCCAA
 id4 10 ACAGTGTCAA
 id5 1  GCGGTGTCAA
}
```

  Note: The last closing brace marks the end of the sample definition. A new sample definition begins with another keyword SampleName.

  FREQUENCY data type:

If the data type is set to FREQUENCY, one must only specify for each haplotype its identifier (a string of characters without blanks) and its sample frequency (either relative or absolute). In this case the haplotype should not be defined.

Example:

```
SampleData={
   id1     1
   id2     2
   id3     8
   id4     10
   id5     1
}
```

Haplotypic data

For all data types except FREQUENCY, one must specify for each haplotype its identifier and its sample frequency. If no haplotype list has been defined earlier, one must also define here the allelic content of the haplotype. The haplotype identifier is used to establish a link between the haplotype and its allelic content maintained in a local database.

Once a haplotype has been defined, it needs not be defined again. However the allelic content of the same haplotype can also be defined several times. The different definitions of haplotypes with same identifier are checked for equality. If they are found identical, a warning is issued is the log file. If they are found to be different at some loci, an error is issued and the program stops, asking you to correct the error.

For complex haplotypes like very long DNA sequences, one can perfectly assign different identifiers to all sequences (each having thus an absolute frequency of 1), even if some sequences turn out to be similar to each other. If the option *Infer Haplotypes from Distance Matrix* is checked in the General Settings dialog box, Arlequin will check whether haplotypes are effectively different or not. This is a good precaution when one tests the selective neutrality of the sample using Ewens-Watterson or Chakraborty's tests, because these tests are based on the observed number of effectively different haplotypes.

Genotypic data

For each genotype, one must specify its identifier, its sample frequency, and its allelic content. Genotypic data can be entered either as a list of individuals, all having an absolute frequency of 1, or as a list of genotypes with different sample frequencies. During the computations, Arlequin will compare all genotypes to all others and recompute the genotype frequencies.

The allelic content of a genotype is entered on two separate lines in the form of two pseudo-haplotypes.

Examples:

1):

```
Id1 2   ACTCGGGTTCGCGCGC     # the first pseudo-haplotype
        ACTCGGGCTCACGCGC     # the second pseudo-haplotype
```

2)

```
my_id   4       0 0 1 1 0 1
                0 1 0 0 1 1
```

If the gametic phase is supposed to be known, the pseudo-haplotypes are treated as truly defined haplotypes.

If the gametic phase is not supposed to be known, only the allelic content of each locus is supposed to be known. In this case an equivalent definition of the upper phenotype would have been:

```
my_id  4       0 1 1 0 0 1
               0 0 0 1 1 1
```

3.2.2.4 Genetic structure

The hierarchical genetic structure of the samples is specified in this optional sub-section. It is possible to define groups of populations. This subsection starts with the keyword [[Structure]]. The definition of a genetic structure is only required for AMOVA analyses.

One must specify:

- *A name for the genetic structure*

  Notation: **StructureName** =

  Possible values: Any string of characters within quotes.

  Example: `StructureName= "A first example of a genetic structure"`

  Note: This name will be used to refer to the tested structure in the output files.

- *The number of groups defined in the structure*

  Notation: **NbGroups** =

  Possible values: Any integer value.

  Example: `NbGroups = 5`

  Note: If this value does not correspond to the number of defined groups, then calculations will not be possible, and an error message will be displayed.

- *If we add the individual level in the variance analysis*

  Notation: **IndividualLevel** =

  Possible values: 0 (no) or 1 (yes)

  Example: `IndividualLevel = 0`

  Note: Default value: 0. The value 1 is only possible with genotypic data.

- *The group definitions*

  Notation: **Group** =

  Possible values: A list containing the names of the samples belonging to the group, entered within braces.

  Repeat this for as many groups you have in your structure. It is of course not allowed to put the same population in different groups. Also note that a comment sign (#) is not allowed after the opening brace and would lead to an error message. Comments about the group should therefore be done *before* the definition of the group.

Example (NbGroups=2) :

```
Group ={
      population1
      population2
      population3
}
Group ={
      population4
      population5
}
```

3.2.2.4 Mantel test settings

This subsection allows to specify some distance matrices (*Ymatrix*, *X1* and *X2*). The goal is to compute a correlation between the *Ymatrix* and *X1* or a partial correlation between the *Ymatrix*, *X1* and *X2*. The *Ymatrix* can be either a pairwise population F_{ST} *matrix* or a custom matrix entered into the project by the user. X1 (and X2) have to be defined in the project.

This subsection starts with the keyword `[[Mantel]]`. The matrices, which are used to test correlation between genetic distances and one or two other distance matrices, are defined in this section.

One must specify:

- *The size of the matrices used for the Mantel test.*

  Notation: **MatrixSize**=

  Possible values: Any positive integer value.

  Example: `MatrixSize= 5`

- *The number of matrices among which we compute the correlations. If this number is 2 the correlation coefficient between the* ***YMatrix*** *(see next keyword) and the matrix defined after the* ***DistMatMantel*** *keyword. If this number is 3 the partial correlation between the* ***YMatrix*** *(see next keyword) and the two other matrices are computed. In this case the Mantel section should contain two* ***DistMatMantel*** *keywords followed by the definition of a distance matrix.*

  Notation: **MatrixNumber**=

  Example: `MatrixNumber= 2`

- *The matrix that is used as genetic distance. If the value is* "fst*" then the correlation between the population pairwise* F_{ST} *matrix other another matrix is computed. . If the value is* "custom*" then the correlation between a project defined matrix and other matrix is computed*

  Notation: **YMatrix**=

Possible values:	*Corresponding YMatrix*
"fst"	Y=Fst
"log_fst"	Y=log(Fst)
"slatkinlinearfst"	Y=Fst/(1-Fst)
"log_slatkinlinearfst"	Y=log(Fst/(1-Fst))
"nm"	Y=(1-Fst)/(2 Fst)
"custom"	Y= user-specified in the project

Example: `YMatrix = “fst”`

- *Labels that identify the columns of the **YMatrix**. In case of YMatrix* = “fst” *the labels should be names of population from witch we use the pairwise F_{ST} distances. In case of YMatrix* = “custom” *the labels can be chosen by the user. These labels will be used to select the sub-matrices on witch correlation (or partial correlation) is computed.*

  Notation: **YMatrixLabels** =

  Possible values: A list containing the names of the label name belonging to the group, entered within braces.

  Example:
```
YMatrixLabels =  {
                  "Population1 " "Population4" "Population2"
                  "Population8" "Population5"
        }
```

- *A keyword that allows to define a matrix with witch the correlation with the **YMatrix** is computed.*

  Notation: **DistMatMantel** =

  Example:
```
DistMatMantel={
            0.00
            3.20 0.00
            0.47 0.76 0.00
            0.00 1.23 0.37 0.00
            0.22 0.37 0.21 0.38 0.00
        }
```

- *Labels defining the sub-matrix on witch the correlation is computed.*

  Notation: **UsedYMatrixLabels**=

  Possible values: A list containing the names of the label name belonging to the group, entered within braces.

  Example:
```
UsedYMatrixLabels={
            "Population1 "
           "Population5"
            "Population8"
        }
```

Note: If you want to compute the correlation between entirely user-specified matrices, you need to list a dummy population sample in the `[[Sample]]`section, in order to allow for a proper reading of the Arlequin project. We hope to remove this weird limitation, but it is the way it works for now !

Two complete examples:

Example 1: We compute the partial correlation between the YMatrix and two other matrices X1 and X2. The YMatrix will be the pairwise F_{ST} matrix between the population listed after *YMatrixLabels* . The partial correlations will be based on the 3 by 3 matrix whose labels are listed after *UsedYMatrixLabels*.

```
[[Mantel]]
     #size of the distance matrix:
     MatrixSize= 5
     #number of declared matrixes:
     MatrixNumber=3
     #what to be taken as the YMatrix
     YMatrix="Fst"
     #Labels to identify matrix entry and Population
     YMatrixLabels ={
           "pop 1"
           "pop 2"
           "pop 3"
           "pop 4"
           "pop 5"
     }
     # distance matrix: X1
     DistMatMantel={
       0.00
       1.20 0.00
       0.17 0.84 0.00
       0.00 1.23 0.23 0.00
       0.12 0.44 0.21 0.12 0.00
     }

     # distance matrix: X2
     DistMatMantel={
       0.00
       3.20 0.00
       0.47 0.76 0.00
       0.00 1.23 0.37 0.00
       0.22 0.37 0.21 0.38 0.00
     }

     UsedYMatrixLabels ={
           "pop 1"
           "pop 3"
           "pop 4"
     }
```

Example 2: we compute the correlation between the YMatrix and another matrix X1. The YMatrix will be defined after the keyword **YMatrix**. The correlation will be based on the 3 by 3 matrix whose labels are listed after *UsedYMatrixLabels* .

```
[[Mantel]]
     #size of the distance matrix:
     MatrixSize= 5
     #number of declared matrixes: 1 or 2
     MatrixNumber=2
     #what to be taken as YMatrix
     YMatrix="Custom"
```

```
#Labels to identify matrix entry and Population
YMatrixLabels ={
    "1" "2" "3"
    "4" "5"
}
#This will be the Ymatrix
DistMatMantel={
  0.00
  1.20 0.00
  1.17 0.84 0.00
  1.00 1.23 0.23 0.00
  2.12 0.44 0.21 0.12 0.00
}
#This will be X1
DistMatMantel={
  0.00
  3.20 0.00
  2.23 1.73 0.00
  2.55 2.23 0.35 0.00
  2.23 1.62 1.54 2.32 0.00
}
UsedYMatrixLabels ={
 "1" "2"
 "3"
 "4" "5"
}
```

3.3 Example of an input file

The following small example is a project file containing four populations. The data type is STANDARD genotypic data with unknown gametic phase.

```
[Profile]
    Title="Fake HLA data"
    NbSamples=4
    GenotypicData=1
    GameticPhase=0
    DataType=STANDARD
    LocusSeparator=WHITESPACE
    MissingData='?'

[Data]

[[Samples]]
    SampleName="A sample of 6 Algerians"
    SampleSize=6
    SampleData={
           1  1 1104 0200
                0700 0301
           3  3 0302 0200
                1310 0402
           4  2 0402 0602
                1502 0602
    }
    SampleName="A sample of 11 Bulgarians"
    SampleSize=11
```

```
    SampleData={
           1  1   1103    0301
                  0301    0200
           2  4   1101    0301
                  0700    0200
           3  1   1500    0502
                  0301    0200
           4  1   1103    0301
                  1202    0301
           5  1   0301    0200
                  1500    0601
           6  3   1600    0502
                  1301    0603
    }
    SampleName="A sample of 12 Egyptians"
    SampleSize=12
    SampleData={
          1    2      1104   0301
                      1600   0502
          3    1      1303   0301
                      1101   0502
          4    3      1502   0601
                      1500   0602
          6    1      1101   0301
                      1101   0301
          8    4      1302   0502
                      1101   0609
          9    1      1500   0302
                      0402   0602
    }
    SampleName="A sample of 8 French"
    SampleSize=8
    SampleData={
          219    1    0301   0200
                      0101   0501
          239    2    0301   0200
                      0301   0200
          249    1    1302   0604
                      1500   0602
          250    3    1401   0503
                      1301   0603
          254    1    1302   0604
    }

[[Structure]]

    StructureName="My population structure"
    NbGroups=2
    Group={
          "A sample of 6 Algerians"
          "A sample of 12 Egyptians"
    }
    Group={
          "A sample of 11 Bulgarians"
          "A sample of 8 French"
    }
```

3.4 Automatically creating th e outline of a project file

In order to help you setting up quickly a project file, Arlequin can create the outline of a project file for you.
In order to do this, use the ***Project outline wizard*** dialog box by activating the *Project* | *Build Project Outline* menu. A special dialog box will appear, allowing to quickly define which type of data you have and some specificities of the data.

- **Data file**

  Specify the name of the target file (the new Arlequin project). It should have the extension ".arp".

- **Data type**

  Specify which **type of data** you want to analyze (DNA, RFLP, Microsat, Standard, or Frequency).

  Specify if the data is under **genotypic** or **haplotypic** form.

  Specify if the **gametic phase** is known (for genotypic data only).

  Specify if there are **recessive alleles** (for genotypic data only)

- **Controls**

  Specify the number of population samples defined in the project

  Choose a **locus separator**

  Specify the character coding for **missing data**

  Specify the **code for the recessive allele**

- **Optional sections**

  Specify if you want to include a global **list of haplotypes**

  Specify if you want to include a predefined **distance matrix**

  Specify if you want to include a **group structure**

By pressing the *Open outline as project* button, an empty outline of a project file will be created for you. It will be automatically loaded in Arlequin, and you can then paste your sample data by editing the project.

3.5 Conversion of data files

By selecting the Tab dialog *Import Data*, one activate a dialog box for the translation of data files from one format to the other. This might be useful for users already having data files set up for other data software packages. It is also possible to convert Arlequin data files into other formats.
The currently recognized data formats are:

- Arlequin ver. 1.1
- GenePop ver. 3.0,
- Biosys ver.1.0,
- Phylip ver. 3.5
- Mega ver. 1.0
- Win Amova ver. 1.55.

The translation procedure is fully described in section 6.3.4.
These conversion routines were done on the basis of the description of the input file format found in the user manuals of each of aforementioned programs. The tests done with the example files given with these programs worked fine. However, the original reading procedures of the other software packages may be more tolerant than our owns, and

some data may be impossible to convert. Thus, some small corrections will need to be done by hand, and we apologize for that.

3.6 Arlequin batch files

A large number of data files can be analyzed one after the other using batch files.
A batch file (having usually the *.arb* extension) is simply a text file having on each line the name of the project file that should be analyzed. The number of data files to be analyzed can be arbitrary large.
If the project type you open is of *Batch file type*, the *Batch file* tab panel allows you to tune the settings for your batch run.
You can either use the same options for all project files by selecting ***Use interface settings***, or use the setting file associated with each project file by selecting ***Use associated settings***. In the first case, the same analyses will be performed on all project files listed in the batch file. In the second case, you can perform different computations on each project file listed in the batch file, giving you much more flexibility on what should be done. However, it implies that setting files have been prepared previously, recording the analyses needing to be performed on the data, as well as the options of these analyses.
Some results can be collected from the analysis of each batch file, and put into summary files (see section 6.3.6).
If the associated project file does not exist, the current settings are used.
Note that the batch file, the project files, and the setting files should all be in the same folder.

4 OUTPUT FILES

The output files are now all located in a special sub-directory, having the same name as your project, but with the "*.htm*" extension. This has been done to structure your result files according to different projects. For instance, if your project file is called my_file.arp, then the result files will be in a sub-directory called [my_file.res]

4.1 Result file

The file containing all the results of the analyses just performed. By default, it has the same name than the Arlequin input file, with the extension (*.htm*. This file is opened in the right frame of the html browser at the end of each run. If the option *Append Results* of the configuration tab panel is checked, the results of the current computations are appended to the one of previous calculations, otherwise the results of previous analyses are erased, and only the last results are overwritten in the result file.

4.2 Arlequin log file

A file where run-time *WARNINGS* and *ERRORS* encountered during any phases of the current Arlequin session are issued. The file has the name *Arlequin_log.txt* and **is now located in the result directory of the opened project** You should consult this file if you observe any warning or error message in your result file. If Arlequin has crashed then consult *Arlequin_log.txt* **before** running Arlequin again. It will probably help you in finding where the problem was located. A reference to the log file is provided in the left pane of the html result file and can be activated in your web browser.

4.3 Linkage disequilibrium result file

This file contains the results of pairwise linkage disequilibrium tests between all pairs of loci. By default, it has the name LK_DIS.XL. As suggested by its extension, this file can be read with MS-Excel without modification. A tabulator separates the columns.

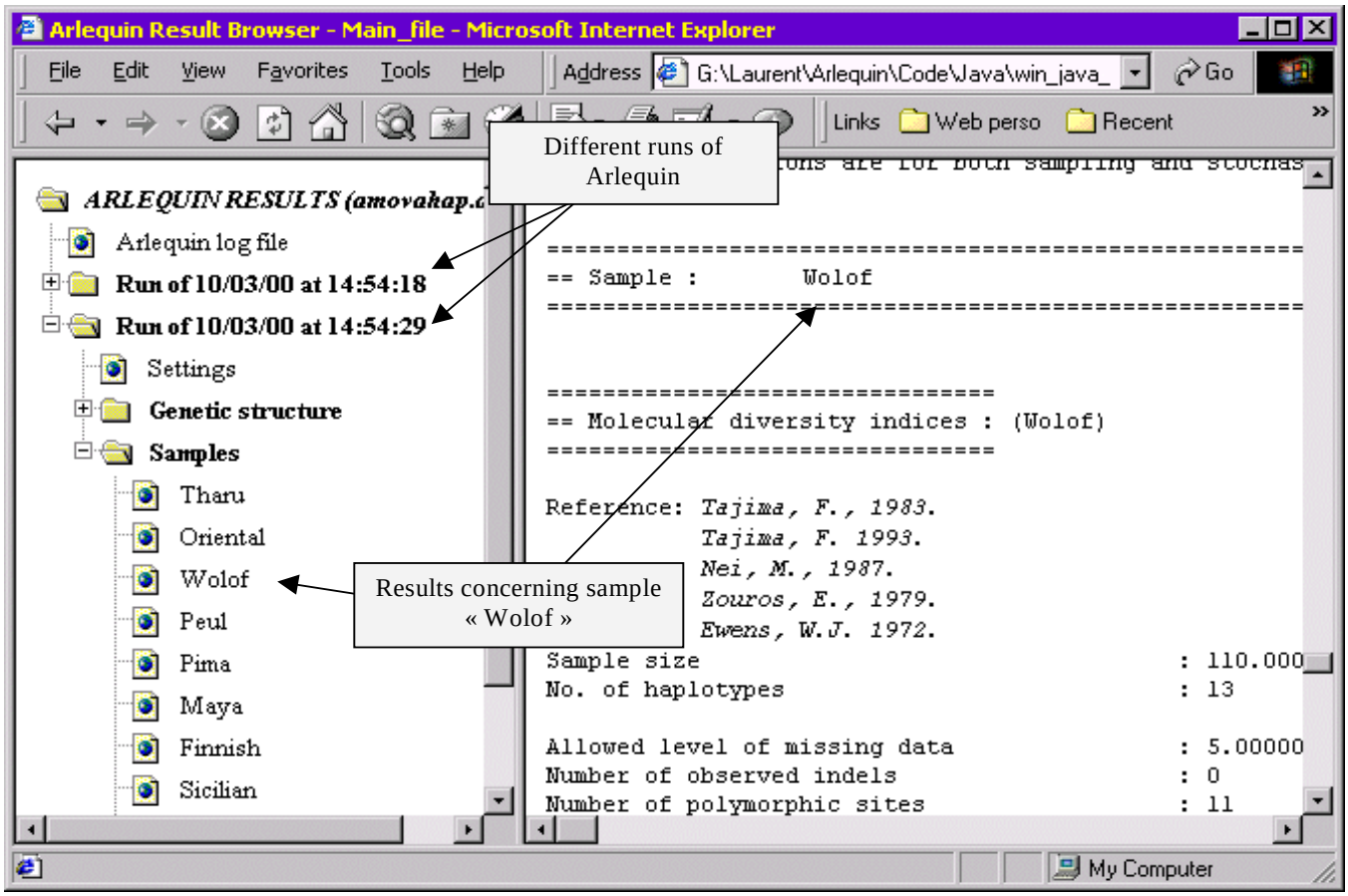
4.4 View your results in HTML browser

For very large result files or result files containing the product of several analyses, it may be of practical interest to view the results in an HTML browser. This can be simply done by activating the button *Browse results* of the project tab panel. The HTML browser can be selected through the *Configuration* tab panel. The location of your browser is then stored in *arlequin.ini*.
In the web browser the file *[project name]_main.html* is loaded and the main window is divided in two panes.

1. The **left pane** contains a tree where each first level branch corresponds to a run. For each run we have several entries corresponding to the used settings for the calculation, the inter-population analyses (Genetic structure, Shared haplotypes) and finally all intra-population analyses with one entry per population. The description of this tree is stored in *[project name]_tree.html*. At this point it is important to notice that this tree uses the java script file *ftiens4.js* located in Arlequin's installation directory. If you move Arlequin to an other location, or uninstall it, the left pane will not work anymore.

2. The **right pane**, shows the results concerning the selected item in the left pane. The HTML code of this pane is in the main result file. This file is located in result sub-directory of your project and is named *[project name].htm.*

The following figure illustrates way results are presented in a HTML browser.



4.5 Variance components null distribution histograms

Specifies the name of an output file where the histograms of the (co)variance component null distributions are output. By default, the name is set to AMO_HIST.XL. This tabulated text file can be read directly by MS-Excel, for a graphical output of the distributions.

All values of the permuted statistics are found in files, having the same name as the project file, with **.va*, **.vb*, **.vc* and **.vd* for σ_a^2, σ_b^2, σ_c^2, and σ_d^2, respectively.

5 EXAMPLES OF INPUT FILES

5.1 Example of allele frequency data

The following example is a file containing FREQUENCY data. The allelic composition of the individuals is not specified. The only informations we have are the frequencies of the alleles.

```
[Profile]
     Title="Frequency data"
     NbSamples=2
     GenotypicData=0
     DataType=FREQUENCY
[Data]
 [[Samples]]
     SampleName="Population 1"
     SampleSize=16
     SampleData= {
          000 1
          001 3
          002 1
          003 7
          004 4
     }
     SampleName="Population 2"
     SampleSize=23
     SampleData= {
          000 3
          001 6
          002 2
          003 8
          004 4
    }
```

5.2 Example of standard data (Genotypic data, unknown gametic phase, recessive alleles)

In this example, the individual genotypes for 5 HLA loci are output on two separate lines. We specify that the gametic phase between loci is unknown, and that the data has a recessive allele. We explicitly define it to be "xxx". Note that with recessive data, all single locus homozygotes are also considered as potential heterozygotes with a null allele. We also provide Arlequin with the minimum frequency for the estimated haplotypes to be listed (0.00001), and we define the minimum epsilon value (sum of haplotype frequency differences between two steps of the EM algorithm) to be reached for the EM algorithm to stop when estimating haplotype frequencies.

```
[Profile]
     Title="Genotypic Data, Phase Unknown, 5 HLA loci"
     NbSamples=1
     GenotypicData=1
     DataType=STANDARD
     LocusSeparator=WHITESPACE
     MissingData='?'
     GameticPhase=0
     RecessiveData=1
     RecessiveAllele="xxx"
     FrequencyThreshold=0.00001
     EpsilonValue=0.000000001
[Data]
 [[Samples]]
```

```
     SampleName="Population 1"
     SampleSize=63
     SampleData={
          MAN0102   12  A33   Cw10   B70     DR1304  DQ0301
                        A33   Cw10   B7801   DR1304  DQ0302
          MAN0103   22  A33   Cw10   B70     DR1301  DQ0301
                        A33   Cw10   B7801   DR1302  DQ0501
          MAN0108   23  A23   Cw6    B35     DR1102  DQ0301
                        A29   Cw7    B57     DR1104  DQ0602
          MAN0109   6   A30   Cw4    B35     DR0801  xxx
                        A68   Cw4    B35     DR0801  xxx
     }
```

5.3 Example of DNA sequence data (Haplotypic)

Here, we define 3 population samples of haplotypic DNA sequences. A simple genetic structure is defined that just incorporates the three population samples into a single group of populations.

```
[Profile]
     Title="An example of DNA sequence data"
     NbSamples=3
     GenotypicData=0
     DataType=DNA
     LocusSeparator=NONE
[Data]
 [[Samples]]
     SampleName="Population 1"
     SampleSize=6
     SampleData= {
          000    3   GACTCTCTACGTAGCATCCGATGACGATA
          001    1   GACTGTCTGCGTAGCATACGACGACGATA
          002    2   GCCTGTCTGCGTAGCATAGGATGACGATA
     }
     SampleName="Population 2"
     SampleSize=8
     SampleData= {
          000    1   GACTCTCTACGTAGCATCCGATGACGATA
          001    1   GACTGTCTGCGTAGCATACGACGACGATA
          002    1   GCCTGTCTGCGTAGCATAGGATGACGATA
          003    1   GCCTGTCTGCCTAGCATACGATCACGATA
          004    1   GCCTGTCTGCGTACCATACGATGACGATA
          005    1   GCCTGTCCGCGTAGCGTACGATGACGATA
          006    1   GCCCGTGTGCGTAGCATACGATGGCGATA
          007    1   GCCTGTCTGCGTAGCATGCGACGACGATA
     }
     SampleName="Population 3"
     SampleSize=6
     SampleData= {
          023    1   GCCTGTCTGCGTAGCATACGATGACGGTA
          024    1   GCCTGTCTGCGTAGCGTACGATGACGATA
          025    1   GCCTGTCTGCGTAGCATACGATGACGATA
          026    1   GCCTGTCCGCGTAGCATACGGTGACGGTA
          027    1   GCCTGTCTGCGTGGCATACGATGACGATG
          028    1   GCCTGTCTGCGTAGCATACGATGACGATA
     }
[[Structure]]
     StructureName="A group of 3 populations analyzed for DNA"
     NbGroups=1
     Group= {
          "Population 1"
          "Population 2"
          "Population 3"
```

```
     }
```

5.4 Example of microsatellite data (Genotypic)

In this example, we show how to prepare a project file consisting in microsatellite data. Four population samples are defined. Three microsatellite loci only have been analyzed in diploid individuals. The different genotypes are output on two separate lines. The frequencies of the different genotypes are listed in the second column of the first line of each genotype. Alternatively, one could just output the genotype of each individual, and simply set its frequency to 1. One should however be careful to use different identifiers for each individual. It does not matter if different genotype labels refer to the same genotype content. Here, only a few different genotypes have been found in each of the populations (which should not correspond to most real situations, but we wanted to save space). The genotypes consist in the number of repeats found at each locus. The genetic structure to be analyzed consists in 2 groups, each made up of 2 populations.

To make things clear, the genotype "Genot1" in the first population, has been observed 27 times. For the first locus, 12 and 13 repeats were observed, 22 and 23 repeats were observed for the second locus, and finally 16 and 17 repeats were found at the third locus.

```
[Profile]
     Title="A small example of microsatellite data"
     NbSamples=4
     GenotypicData=1
     #Unknown gametic phase between the 2 loci
     GameticPhase=0
     DataType=MICROSAT
     LocusSeparator=WHITESPACE
[Data]
  [[Samples]]
     SampleName="MICR1"
     SampleSize=28
     SampleData=      {
          Genot1      27    12 23 17
                            13 22 16
          Genot2      1     15 22 16
                            13 22 16
     }
     SampleName="MICR2"
     SampleSize=59
     SampleData=      {
          Genot3      37    12 24 18
                            12 22 16
          Genot4      1     15 20 18
                            13 22 18
          Genot5      21    14 22 16
                            14 23 16
     }
     SampleName="MICR3"
     SampleSize=30
     SampleData=      {
          Genot6      17    12 21 16
                            13 22 15
          Genot7      1     12 20 16
                            13 23 16
          Genot8      12    10 22 15
                            12 22 15
     }
```

```
     SampleName="MICR4"
     SampleSize=16
     SampleData=      {
          Genot9      15    13 24 16
                            13 23 17
          Genot10     1     12 24 16
                            13 23 16
     }
[[Structure]]
     StructureName="Test microsat structure"
     NbGroups=2
     #We explicitly exclude the individual level from the genetic
     #structure. FIS and FIT statistics won't be computed
     IndividualLevel=0
     #The first group is made up of the first 2 samples
     Group={
               "MICR1"
               "MICR2"
      }
     #The last 2 samples will be put into the second group
     Group={
               "MICR3"
               "MICR4"
      }
```

5.5 Example of RFLP data(Haplotypic)

In this example, we show how to use a definition list of RFLP haplotypes. Different RFLP haplotypes are first defined in the `[[HaplotypeDefinition]]` section. The allelic content of each haplotype is then defined after a given identifier. The identifier is then used at the population samples level. Note that the list of haplotypes can include haplotypes that are not listed in the population samples. The genetic diversity of the samples is then simply described as a list of haplotypes found in each population as well as their sample frequencies.

```
[Profile]
     Title="A small example of RFLP data: 3 populations"
     NbSamples=3
     GenotypicData=0
     DataType=RFLP
     LocusSeparator=WHITESPACE
     #We tell Arlequin to compute Euclidian square distances between
     #the haplotypes listed below
     CompDistMatrix=1
     MissingData='?'
[Data]
 [[HaplotypeDefinition]]
     HaplListName="A fictive list of RFLP haplotypes"
     HaplList= {
       1     00001110011101001101100100101100110111010010110 0
       2     10001110011101001101100100101100110111010010110 0100
       6     00001110011101001001100100101100110111010010110 1100
       7     10001110011101001101100100101100110111010010110 1100
       8     00001110011101001101100100100100110111010010110 1100
       11    00000110011101101101100100101100110111010010111 1100
       12    00001110011101001101100110101100110111010010110 1100
       17    00001110011101001101100100101100110011010010110 1100
       22    00001110011101101101100100101100110111010010110 0100
       36    00001110011101001101100100101000110011010010110 1100
       37    00001110011101101101100100111100110111010010110 0100
       38    00011110011101001101100100101100110111010010110 1100
```

```
        40    00001110011100001101100100101100110111010010110 1100
        47    00001110011101001101100100101100110111010010110 0100
        139   00001110011101001101100100101100111111010010100 1110
        140   00001110011101001101100100101100110111010010110 0101
        141   00001110011101001001100100001100110111010010110 0100
      }
   [[Samples]]
      #1
      SampleName="pop 1"
      SampleSize=28
      SampleData=      {
           1     27
           40    1
      }
      #2
      SampleName="pop 2"
      SampleSize=75
      SampleData=      {
           1     37
           17    1
           6     21
           7     1
           2     1
           22    5
           11    2
           36    1
           139   1
           47    1
           140   1
           141   1
           37    1
           38    1
      }
      #3
      SampleName="pop 3"
      SampleSize=48
      SampleData=      {
           1     46
           8     1
           12    1
      }
  [[Structure]]
      StructureName="A single group of 3 samples"
      NbGroups=1
      Group={
           "pop 1"
           "pop 2"
           "pop 3"
      }
```

5.6 Example of standard data (Genotypic data, known gametic phase)

In this example, we have defined 3 samples consisting of standard multi-locus data with known gametic phase. It means that the alleles listed on the same line constitute a haplotype on a given chromosome. For instance, the genotype G1 is made up of the two following haplotypes: AD on one chromosome and BC on the second, A and b being two alleles at the first locus, and C and D being two alleles at the second locus. Note that the same allele identifier can be used in different loci. This is obviously true for Dna sequences, but it also holds for all other data types.

```
  [Profile]
      Title="An example of genotypic data with known gametic phase"
      NbSamples=3
```

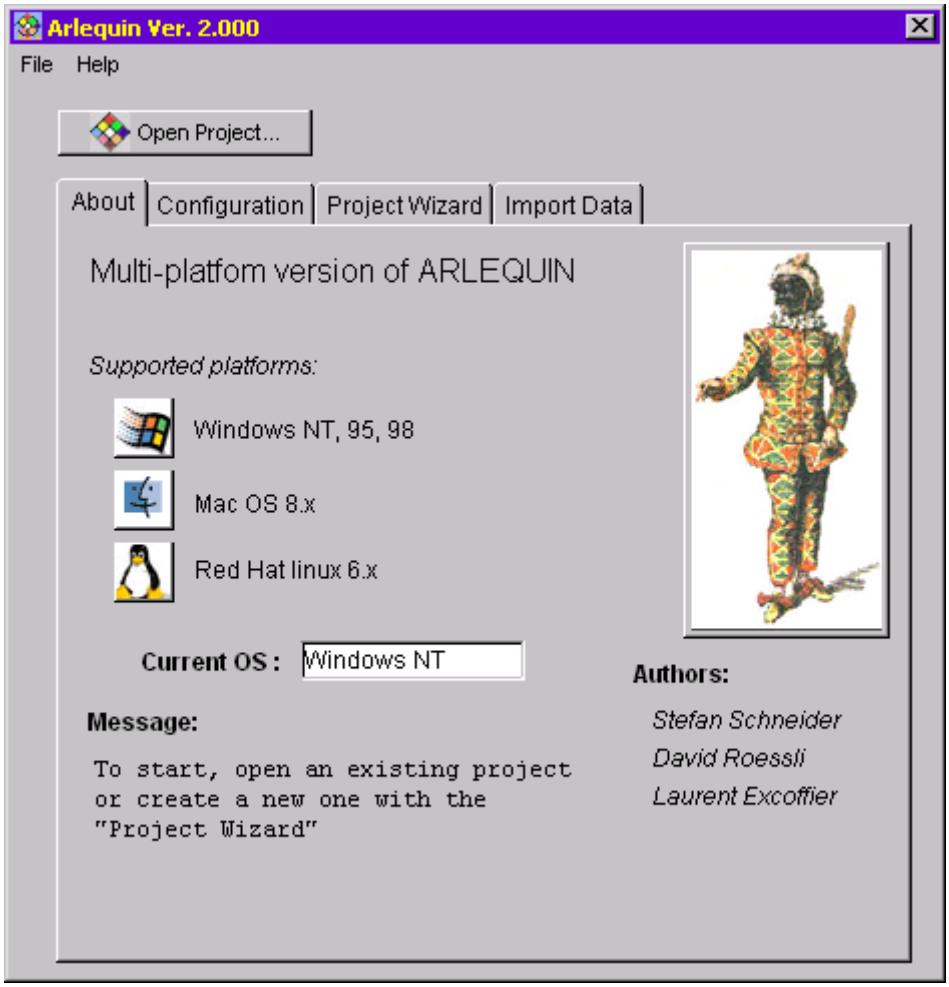
```
     GenotypicData=1
     GameticPhase=1
     #There is no recessive allele
     RecessiveData=0
     DataType=STANDARD
     LocusSeparator=WHITESPACE
[Data]
  [[Samples]]
     SampleName="standard_pop1"
     SampleSize=20
     SampleData=      {
          G1   4    A  D
                    B  C
          G2   5    A  B
                    A  A
          G3   3    B  B
                    B  A
          G4   8    D  C
                    D  C
     }
     SampleName="standard_pop2"
     SampleSize=10
     SampleData=      {
          G5   5    A  C
                    C  B
          G6   5    B  C
                    D  B
     }
     SampleName="standard_pop3"
     SampleSize=15
     SampleData=      {
          G7   3    A  D
                    C  A
          G8   12   A  C
                    B  B
     }

[[Structure]]
     StructureName="Two groups"
     NbGroups=2
     Group={
          "standard_pop1"
     }
     Group={
          "standard_pop2"
          "standard_pop3"
     }
```

6

6 ARLEQUIN INTERFACE

The Arlequin interface has been written in Java in order to present a consistent look and behavior across different platforms.



The graphical interface is made up of a series of tabbed dialog boxes, whose content varies dynamically depending on the type of data currently analyzed.

6.1 Menus

Only two menus will appear at the top of the program window.

6.1.1 File Menu

The menu by which you can open or close projects files.

- Open project... Call a dialog box listing the last 10 opened projects, and allowing you to browse files on your disk.
- Close project... Closes the active project file.
- Quit Exits Arlequin and closes all windows (Java graphic interface and console window).

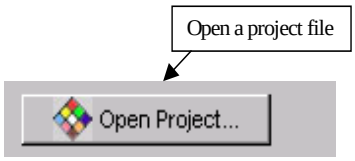
6.1.2 Help Menu

The menu to get access to the Help File System

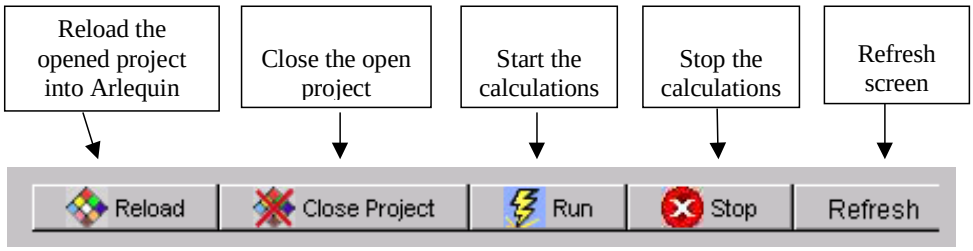
- About — Some information about Arlequin, its authors, contact address and the Swiss NSF grants that supported its development.
- Help... — Calls Arlequin help file. Actually it attempts at opening the file "arlequin.pdf" into the web browser. You thus need to have installed the Adobe Acrobat extensions in your web browser.

6.2 Toolbars

Arlequin's toolbar contains icons that are shortcuts to some commonly used menu items as shown below. Clicking on one of these icons is equivalent to activating the corresponding menu item.



Arlequin tool bar when no project is opened.



Arlequin tool bar when a project is opened.

6.3 Tab dialog boxes

Most of the tasks that Arlequin can perform are possible irrespective of the data type. Nevertheless, the testing procedure that might be used for performing a given task (e.g. testing linkage disequilibrium) may depend on the data type. The aim of this section is to give an overview of what happens in which situation and how to set up the numerous options in an optimal way.

The items that appear «grayed» in Arlequin's dialog boxes indicate that a given task is not possible in the current situation. For example, if you open a project containing haplotypic data, it is not possible to test Hardy-Weinberg equilibrium, and the task will appear as «grayed» in the dialog box. Or, for STANDARD data it is not possible to set up the transversion, transition, and deletion weights.

The way inter-haplotypic distances are calculated depends also on the data type. According to the situation, different lists of distance methods are presented in the dialog box.

Arlequin's interactive graphical user interface should prevent the user from selecting tasks impossible to perform, or from setting up parameters that are not taken into account in the analyses.

We do now describe the different dialog boxes accessible in Arlequin.

We have sometimes used the following symbols to specify which type of input was expected in the dialog boxes:

[f] : parameter to be set in the dialog box as a floating number.

[i] : parameter to be set in the dialog box as an integer.

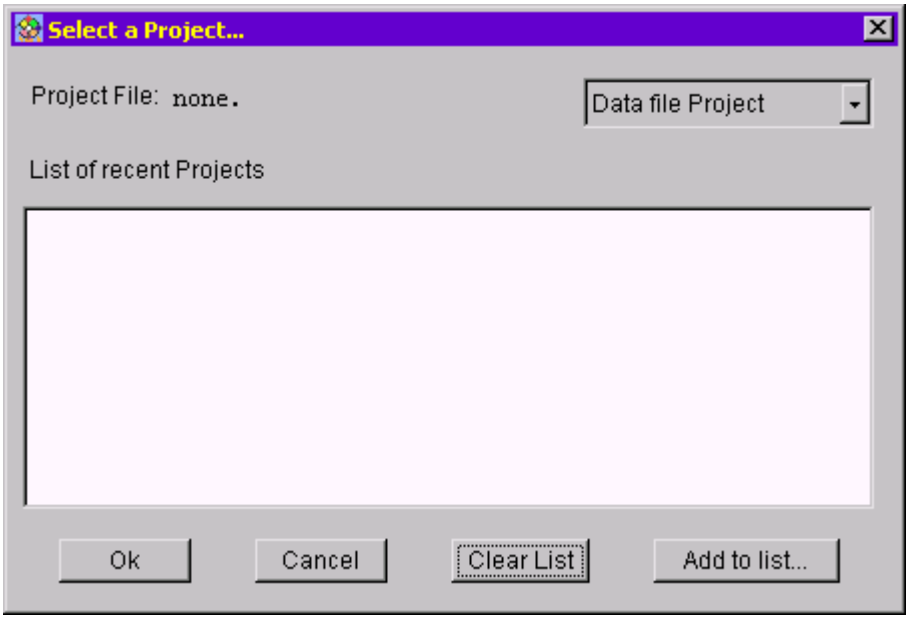
[b] : check box (two states: checked or unchecked).

[m] : multiple selection radio buttons.

[l] : List box, allowing the selection of an item in a downward scrolling list.

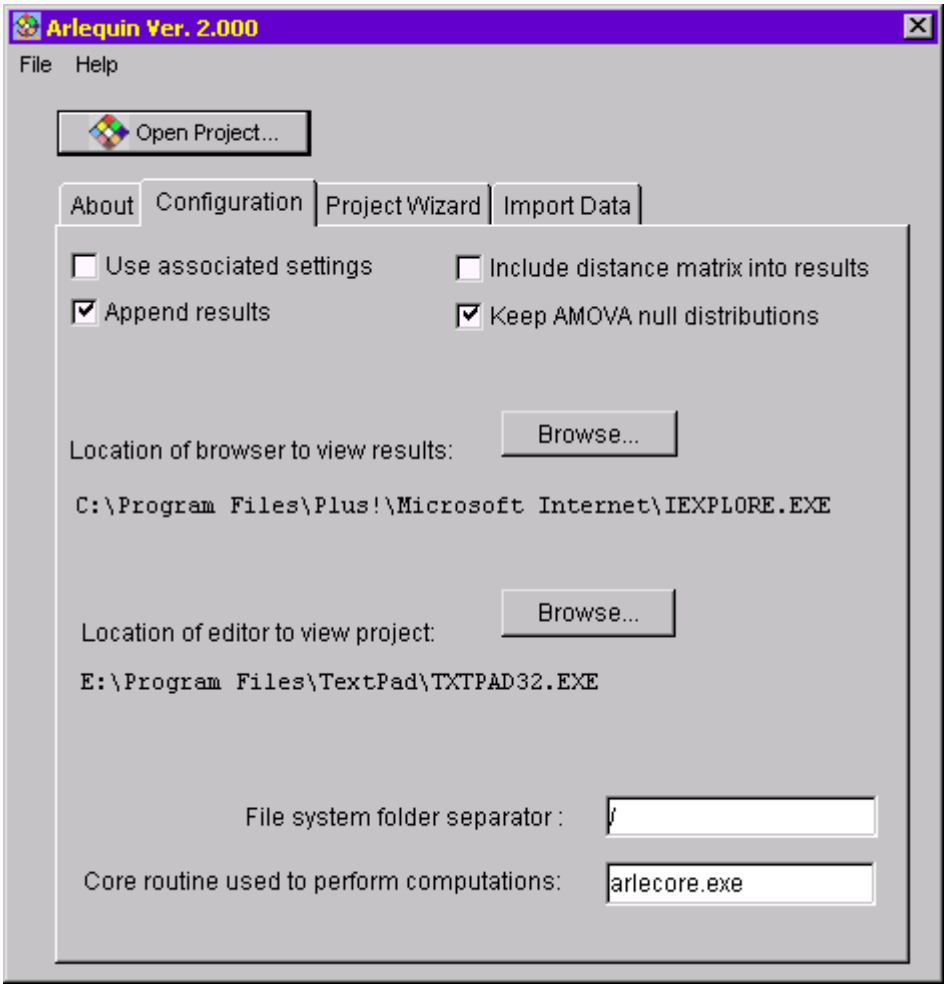
[r] : read only setting, cannot be changed by the user.

6.3.1 Open project



In this dialog box, one can select a new project to analyze: either from a list showing the last 10 opened projects, or by browsing the hard disk, using the *Browse new* button. One can also specify whether the chosen file is a project file or a batch file listing a series of project files to be analyzed.

6.3.2 Configuration

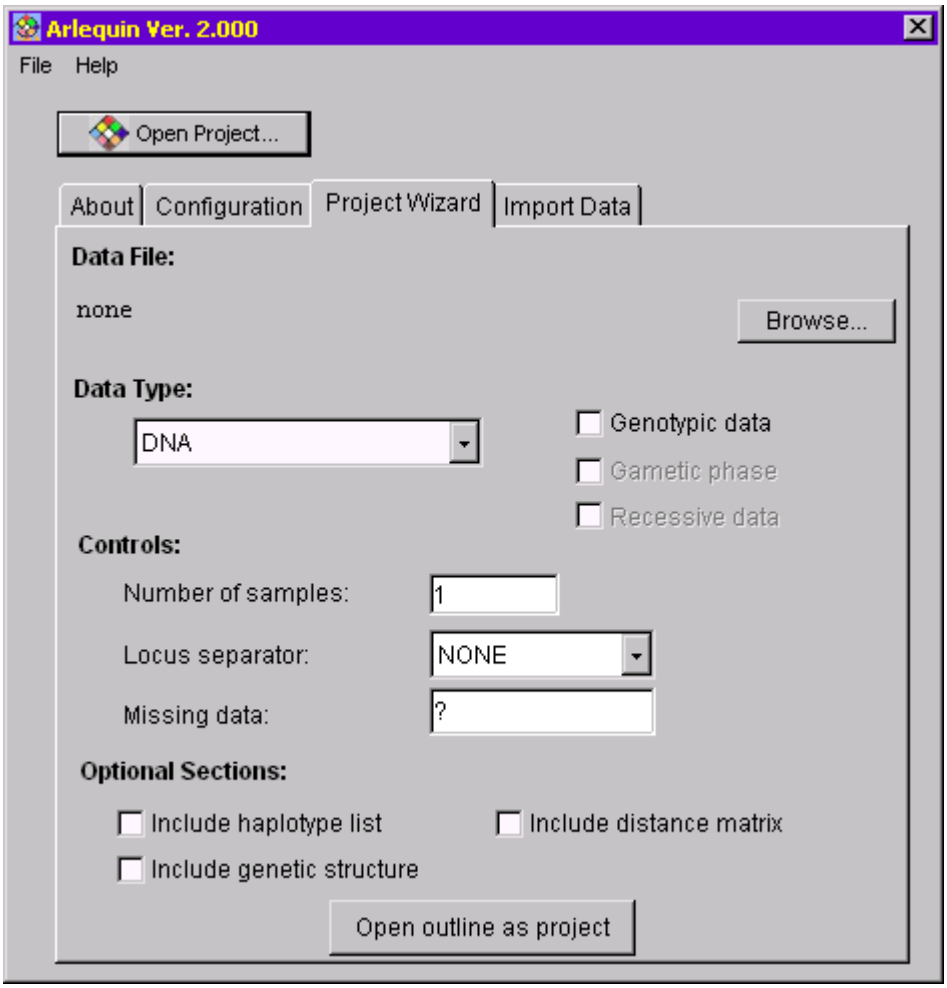


Different options can be specified in this tabbed dialog box.

- **Use associated settings**: By checking the *Use associated settings* checkbox, the settings and options last specified for your project will be used when opening a project file. When closing a project file, Arlequin automatically saves the current calculation settings for that particular project. Check this box if you want Arlequin to automatically load the settings associated to each project. If this box is unchecked, the same settings will be used for different projects.
- **Append Results**: If the option *Append Results* is checked, the results of the current computations are appended to the one of previous calculations, otherwise the results of previous analyses are erased, and only the last results are overwritten in the result file.
- **Include distance matrix into results**: If checked, the inter-haplotypic distance matrix used to evaluate the molecular diversity is printed in the result file.
- **Keep AMOVA null distributions**: If this option is checked, the null distributions of σ_a^2, σ_b^2, σ_c^2, and σ_d^2 generated by an AMOVA analysis are written in files having the same name as the project file, but with the extensions *.va, .vb, .vc,* and *.vd,* respectively.
- **Location of Browser to view results**: Specifies the path and filename of the chosen web browser, where html result files will be output.
- **Location of Editor to view project**: Specifies the path and filename for the chosen text editor one can use to edit the Arlequin projects and view the log files in case of errors during the computations

- **File system folder separator**: Separator used by the operating system to specify the directory hierarchy. It is usually a "\" for Windows, a":" for MacOS, and a "/" for Linux.
- **Core routine used to perform computations** [r]: The name of the compiled program called by the Java interface to make the computations specified by the interface.

6.3.3 Project Wizard



In order to help you setting up quickly a project file, Arlequin can create the outline of a project file for you. This dialog box should allow you to quickly define which type of data you have and some of its peculiarities.

- **Data file**

  Specify the name of the target file (the new Arlequin project) by browsing your hard drive. It should have the extension ".arp".

- **Data type**

  Specify which **type of data** you want to analyze (DNA, RFLP, Microsat, Standard, or Frequency).

  Specify if the data is under **genotypic** or **haplotypic** form.

  Specify if the **gametic phase** is known (for genotypic data only).

  Specify if there are **recessive alleles** (for genotypic data only)

- **Controls**

Specify the number of population samples defined in the project

Choose a **locus separator**

Specify the character coding for **missing data**

Specify the **code for the recessive allele**

- **Optional sections**

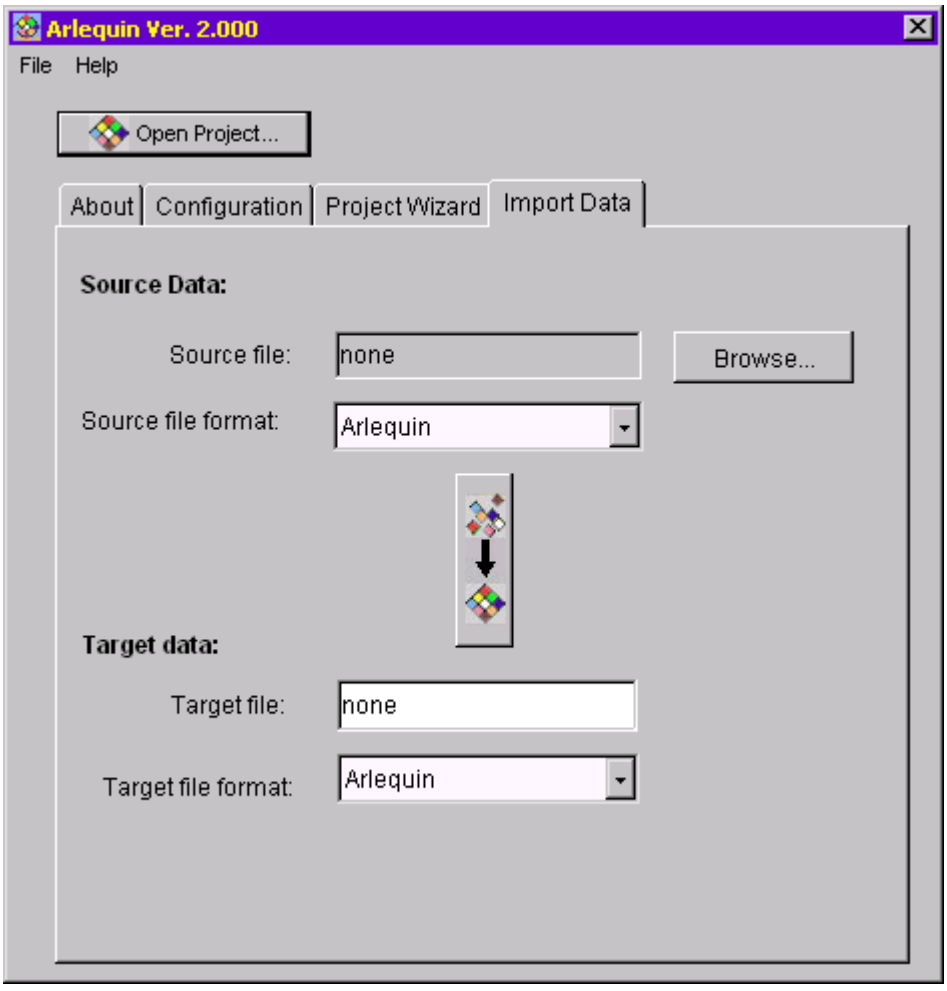
  Specify if you want to include a global **list of haplotypes**

  Specify if you want to include a predefined **distance matrix**

  Specify if you want to include a **group structure**

By pressing the *Open outline as project button*, an empty outline of a project file will be created for you and it will be automatically loaded into your text editor.

6.3.4 Import data



With this dialog box you can quickly translate data into several other file formats often us in population genetics analyses. The currently supported formats are:

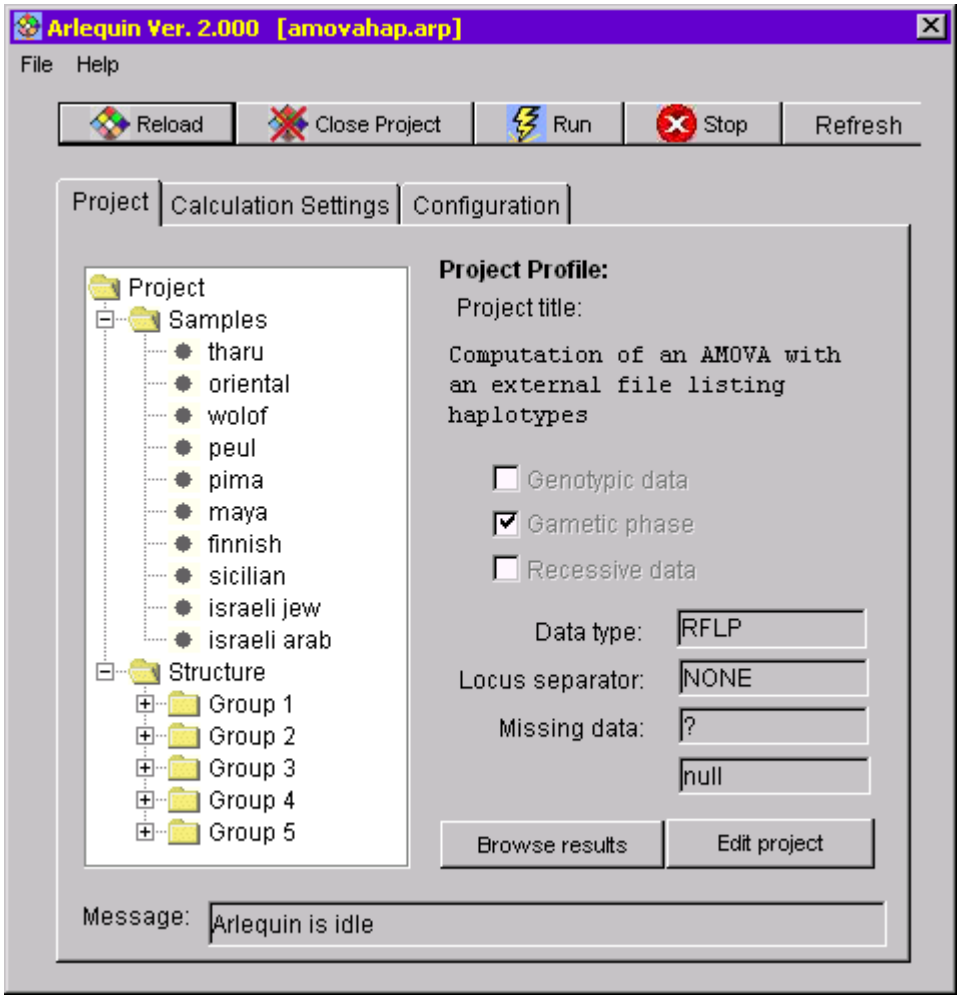
- Arlequin ver. 1.1
- GenePop ver. 1.0,
- Biosys ver.1.0,
- Phylip ver. 3.5

- Mega ver. 1.0
- Win Amova ver. 1.55.

The translation procedure is as follows:

1. Select the source file with the upper right *Browse* button.
2. Select the format of the source data file, as well as that of the target file.
3. A default extension is automatically given to the target file, but you can change the target file name and extension in the edit field.
4. The file conversion is started by pressing on the central button.
5. In some cases, you might be asked for some additional information, for instance if input data is fractionated in several input files (like in WinAmova).
6. If you have selected to translate a data file into the Arlequin file format, you'll have the option to load the newly created project file into the Arlequin Java Interface.

6.3.5 Loaded Project

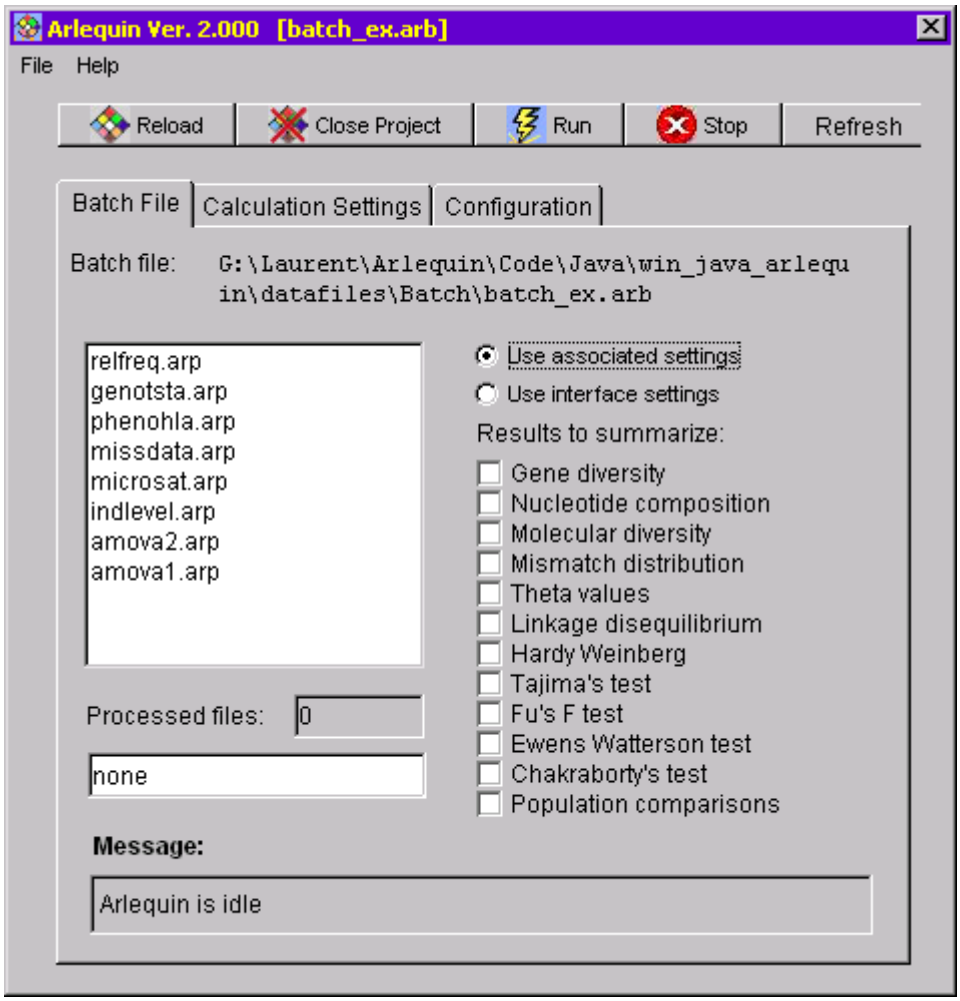


Once a project has been loaded, the *Project* tab dialogbecomes active. It shows a brief outline of the project in an explorable tree pane, and a few information on the data type. The project can be edited by pressing the *Edit Project* button that will launch the text editor currently specified in the *Configuration* tab dialog. All the information shown

under the project profile section is read only. In order to modify them, you need to edit the project file with your text editor and reload the project by pressing the *Reload project...* button at the top of the above window.

- **Project title**[r]: The title of the project as entered in the project.
- **Genotypic data** [r]: Specifies whether input data consist of diploid genotypic data or haplotypic data. For genotypic data, the diploid information of each genotype is entered on separate lines in the input file. The gametic phase of the genotype can be either assumed to be known or unknown. If the gametic phase is known, then the treatment of the data will be essentially similar to that of haplotypic data.
- **Gametic phase** [r]: Specifies whether the gametic phase is known or unknown when the input file is made up of genotypic data.
- **Recessive data** [r]: Specifies if the data contains a recessive allele or not.
- **Data type** [r]: Data type in the input file.
- **Locus separator**[r]: The character used to separate allelic information at adjacent loci.
- **Missing data**[r]: The character used to represent missing data at any locus. By default, a question mark (?) is used for unknown alleles.
- **Recessive allele** [r]: Specifies the identifier of the recessive allele.

6.3.6 Batch files



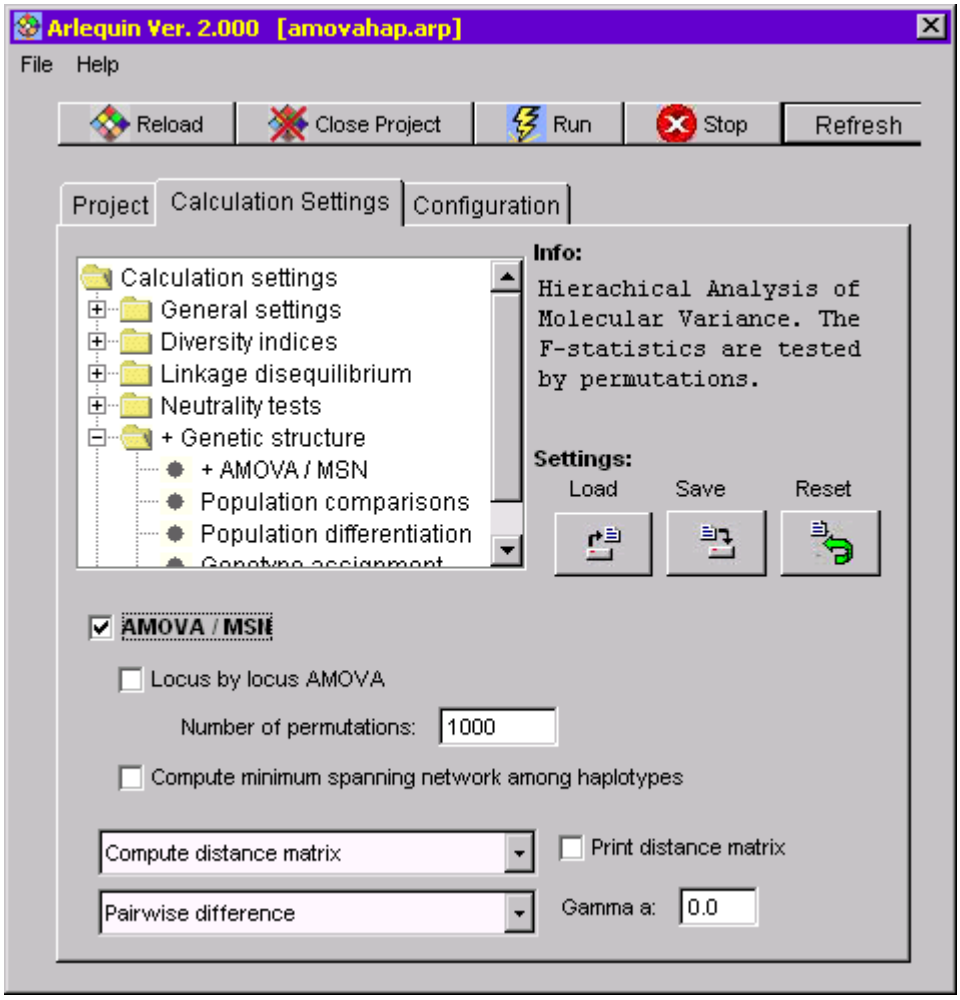
The project files found in the selected batch file appear listed in the left pane window.

- **Use associated settings** [b].: Use this button if you have prepared settings files associated to each project.
- **Use interface settings** [b] : Use this button if you want to use the same predefined calculation settings for all project files.
- **Results to summarize**: This option allows you to collect a summary of the results for each file found in the batch list. These results are written in different files, having the extension **.sum*. These *summary files* will be placed into the same directory as the batch file.

List of summary files that are created by activating different checkboxes.

Checkbox	***Summary file***	***Description***
Gene diversity	gen_div.sum	Gene diversity of each sample
Nucleotide composition	nucl_comp.sum	Nucleotide composition of each sample
Molecular diversity	mold_div.sum	Molecular diversity indexes of each sample
Mismatch distribution	mismatch.sum	Mismatch distribution for each sample
Theta values	theta.sum	Different theta values for each sample
Linkage disequilibrium	l_d_pro.sum	Significance level of linkage disequilibrium for each pair of loci
	link_dis.sum	Number of significantly linked loci per locus
Hardy Weinberg	hw.sum	Test of departure from Hardy-Weinberg equilibrium
Tajima's test	tajima.sum	Tajima's test of selective neutrality
Fu's Fs test	fu_fs.sum	Fu's F_S test of selective neutrality
Ewens Watterson	ewens.sum	Ewens-Watterson tests of selective neutrality
Chakraborty's test	chakra.sum	Chakraborty's test of population amalgamation
Population comparisons	coanst_c.sum	Matrix of Reynolds genetic distances (in linear form)
	NM_value.sum	Matrix of Nm values between pairs of populations (in linear form)
	slatkin.sum	Matrix of Slatkin's genetic distance (in linear form)
	tau_uneq.sum	Matrix of divergence times between populations, taking into account unequal population sizes (in linear form)
	pairdiff.sum	Matrix of mean number of pairwise differences between pairs of samples (in linear form)
	pairdist.sum	Different genetic distances for each pair of population (only clearly readable if 2 samples in the project)

6.3.7 Calculation Settings



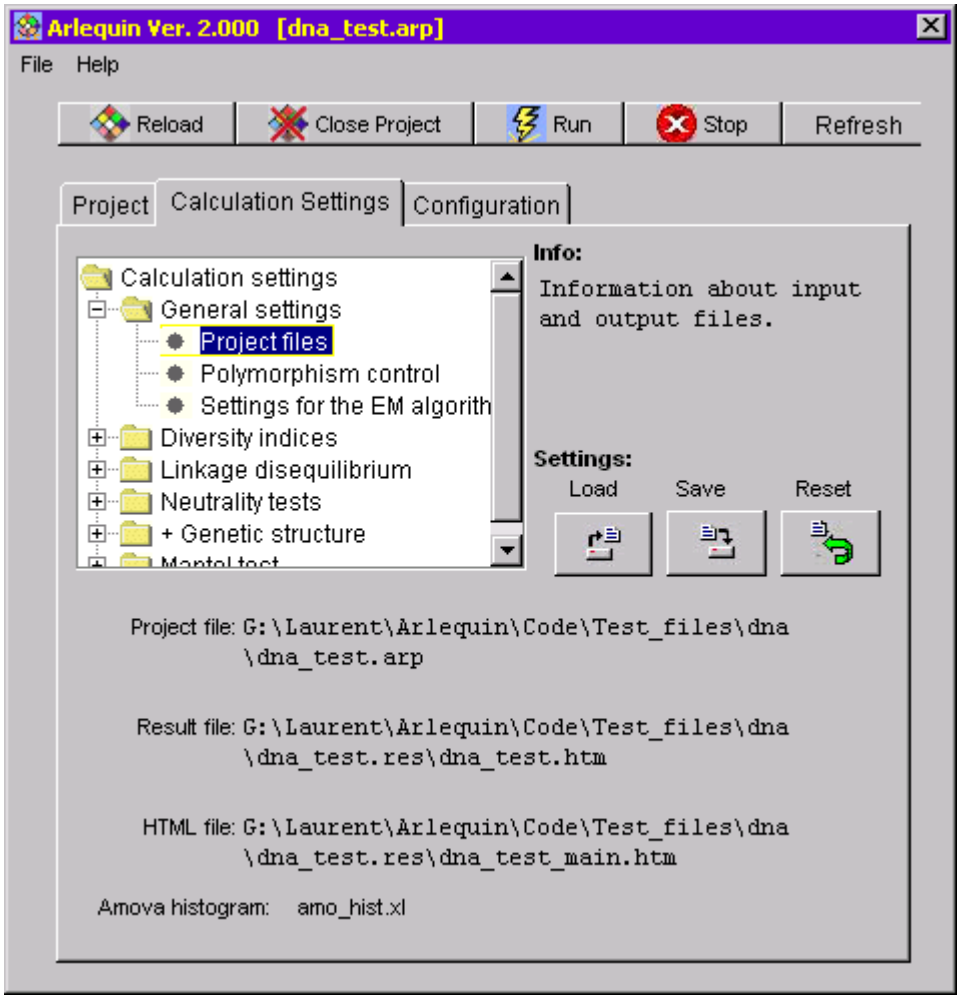
The *Calculation Settings* tabbed dialog is divided into three zones:
At the upper left, a **tree structure** allows the user to quickly select which task to perform. The options for those tasks (**settings**) will appear in the lower pane of the dialog. Finally, an upper right pane will show some **information** on the task selected in the lower pane.

If a particular computation has been selected, it will be reflected by a "+" sign in the tree structure.

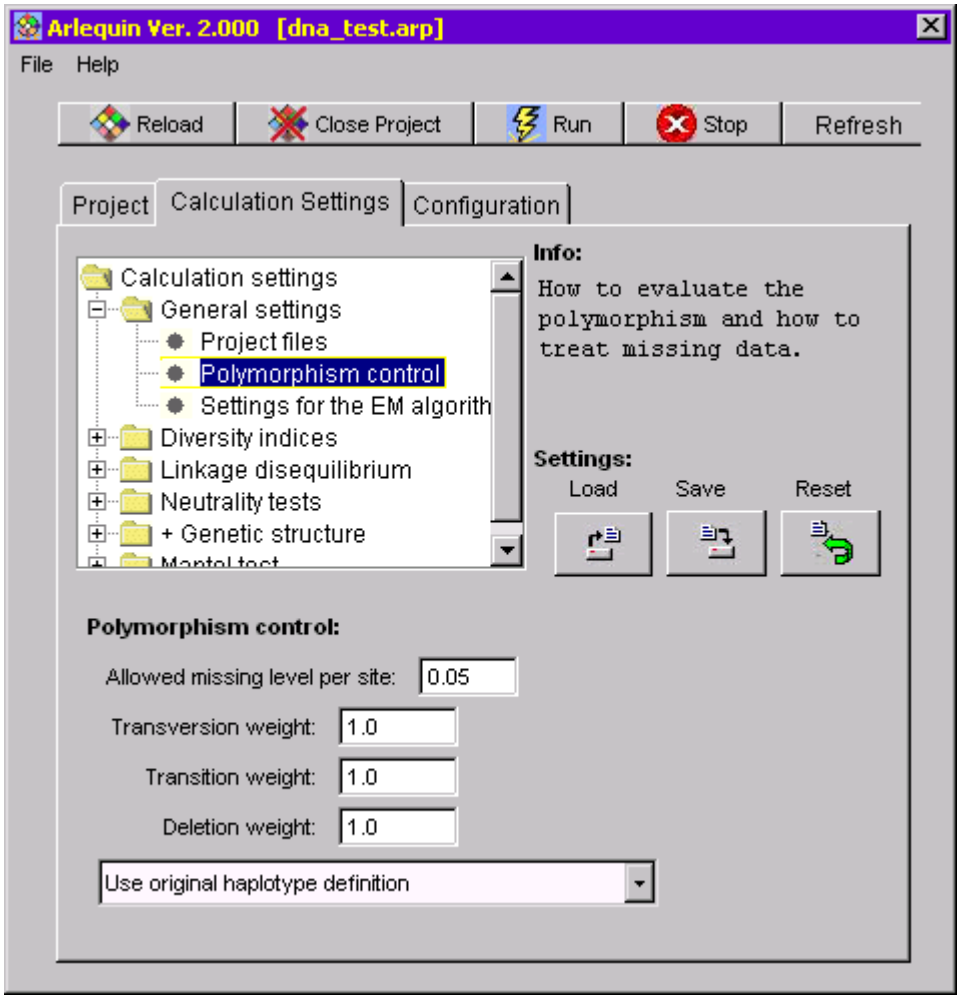
- **Settings**: Three buttons always allow to perform particular actions on the settings:

  - **Load**: Load a particular set of settings previously saved into a settings file (extension "*.ars*") .
  - **Save**: Saves the current settings into a given setting files (extension "*.ars*") .
  - **Reset**: Reset all settings to default values.

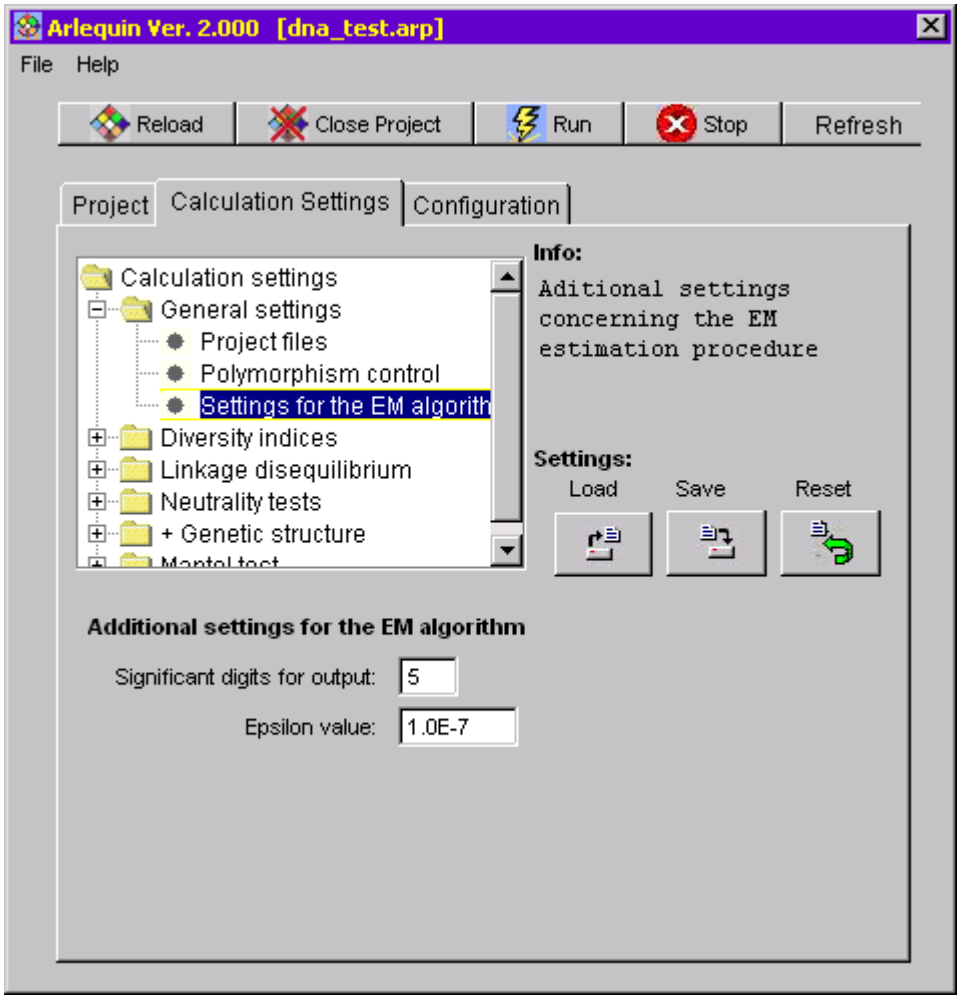
6.3.7.1 General Settings



- **Project file** [r]: The name of the project file containing the data to be analyzed (it usually has the "*.arp*" extension).
- **Result files**: The html file containing the results of the analyses generated by Arlequin (it has the same name as the project file, but the "*.htm"* extension).
- **HTML file** [r]: The main html file containing the structure of the result files.
- **Amova histograms** [r]: Specifies the name of an output file where the histograms of the covariance component null distributions are output. By default, the name is set to *amo_hist.xl*. It is a tabulated text file which can be read directly by MS-Excel, for a graphical output of the distributions.

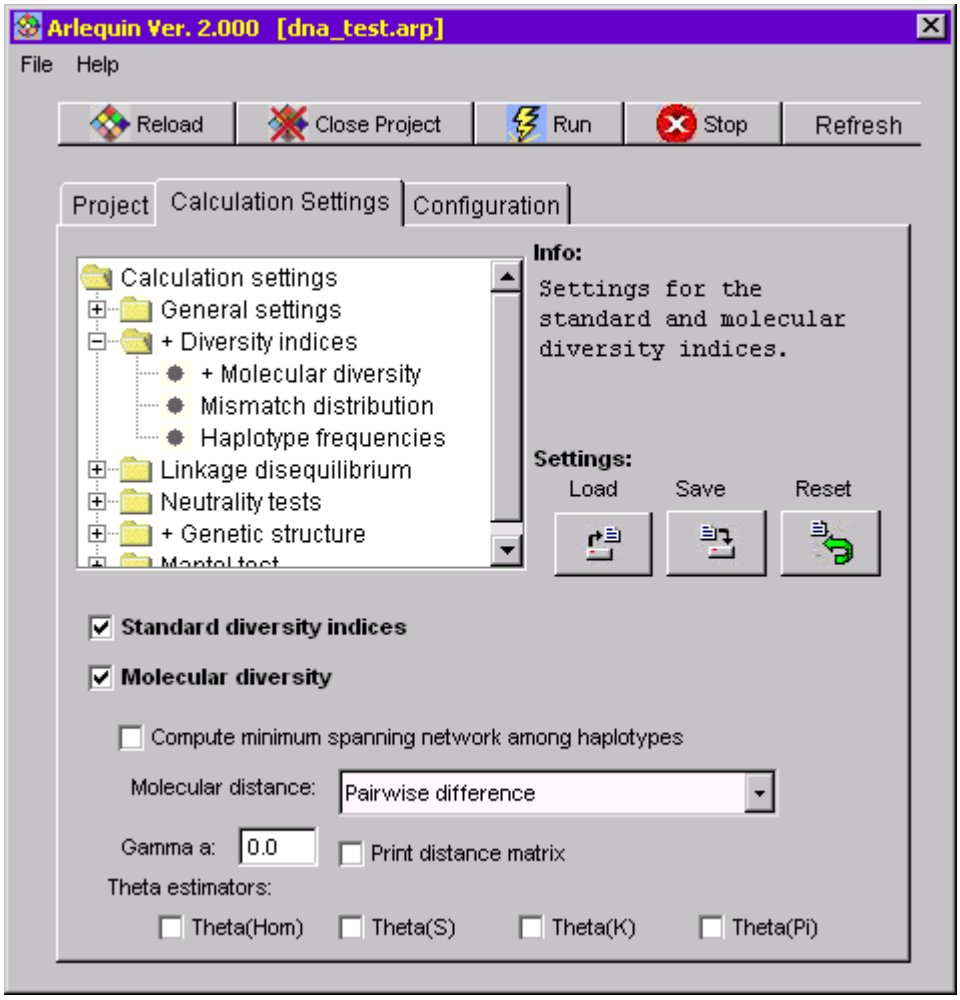


- **Polymorphism control:**
  - **Allowed missing level per site** [f]: Specify the fraction of missing data allowed for any locus to be taken into account in the analyses. For instance, a level of 0.05 means that a locus with more than 5% of missing data will not be considered in any analysis. This option is especially useful when dealing with DNA data where different individuals have been sequenced for slightly different fragments. Setting a level of zero will force the analysis to consider only those sites that have been sequenced in all individuals. Alternatively, choosing a level of one means that all sites will be considered in the analyses, even if they have not been sequenced in any individual (not a very smart choice, however).
  - **Transversion weight** [f]: The weight given to transversions when comparing DNA sequences.
  - **Transition weight** [f]: The weight given to transitions when comparing DNA sequences.
  - **Deletion weight** [f]: The weight given to deletions when comparing DNA or RFLP sequences.
  - **Infer haplotypes from distance matrix** [m] or **Use original haplotype definition** [m]: With the first option, similar haplotypes will be identified by computing a distance matrix based on the settings chosen above. Selecting the second option has the consequence that haplotypes are identified according to their original identifier.



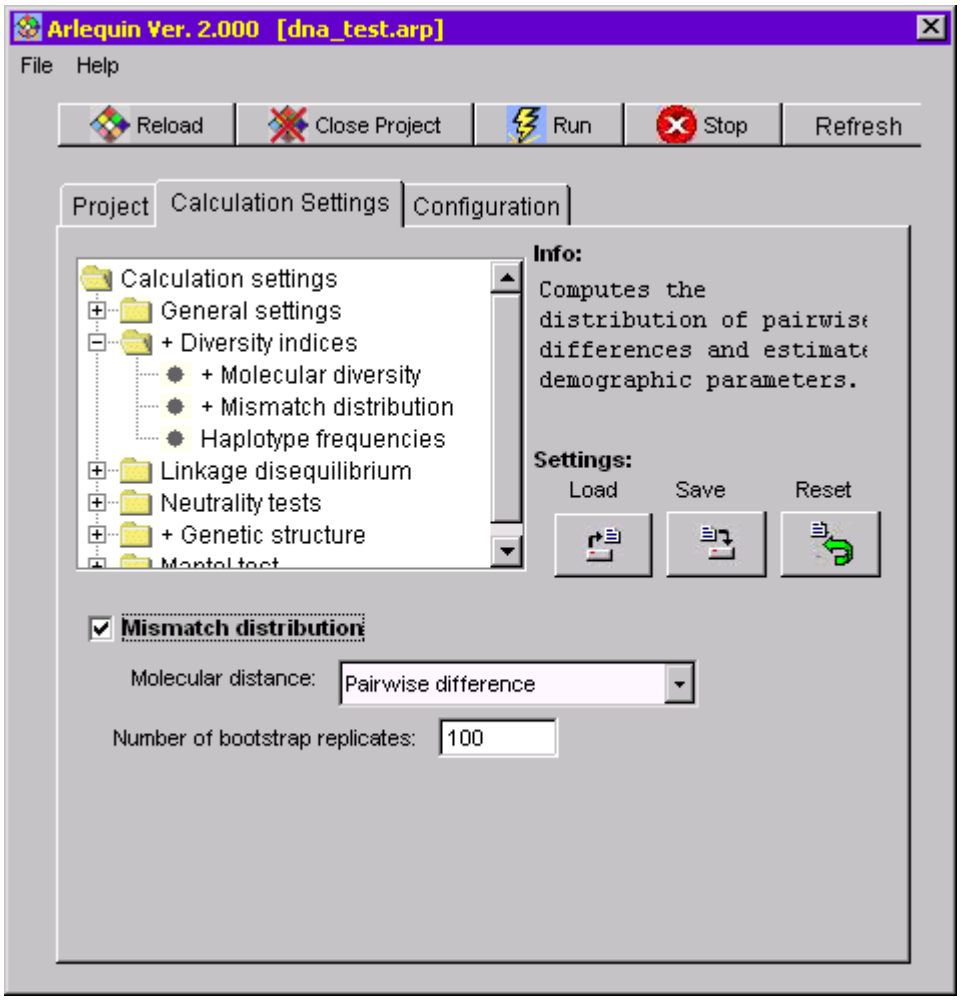
- **Settings for the EM algorithm**: Some settings directly related to haplotype frequency estimation by the EM algorithm and output.
  - **Significant digits for output** [i]: The number of significant digits shown for the estimated haplotype frequencies in the result files.
  - **Epsilon value** [f]: The criterion used to stop the EM algorithm when estimating haplotype frequencies or linkage disequilibrium from genotypic data with unknown gametic phase (see section 7.1.3.2). The criterion is the difference in the sum of haplotypic frequency change between two successive iterations. The default value is 1e-7.

6.3.7.2 Diversity indices



- **Standard diversity indices** [b]: Compute several common indices of diversity, like the number of alleles, the number of segregating loci, the heterozygosity level, etc. (see section 7.1.1).
- **Molecular diversity** [b]: Check box for computing several indices of diversity at the molecular level.
  - **Compute minimum spanning network among haplotypes** [b]: Computes a minimum spanning tree and a minimum spanning network among the haplotypes found in each population sample (see section 7.1.2.9).
  - **Molecular distance** [l]: Choose the type of distance used when comparing haplotypes (see section 7.1.2.5 and below).
  - **Gamma a value** [f]: Set the value for the shape parameter of the gamma function, when selecting a distance allowing for unequal mutation rates among sites. This option is only valid for some distances computed between DNA sequences. Note that a value of zero deactivates here the Gamma correction of these distances, whereas in reality, a value of infinity would deactivate the Gamma correction procedure.
  - **Print distance matrix** [b]: If checked, the inter-haplotypic distance matrix used to evaluate the molecular diversity is printed in the result file.
  - **Theta(Hom)** [b]: An estimation of θ obtained from the observed homozygosity H (see section 7.1.2.3.1).

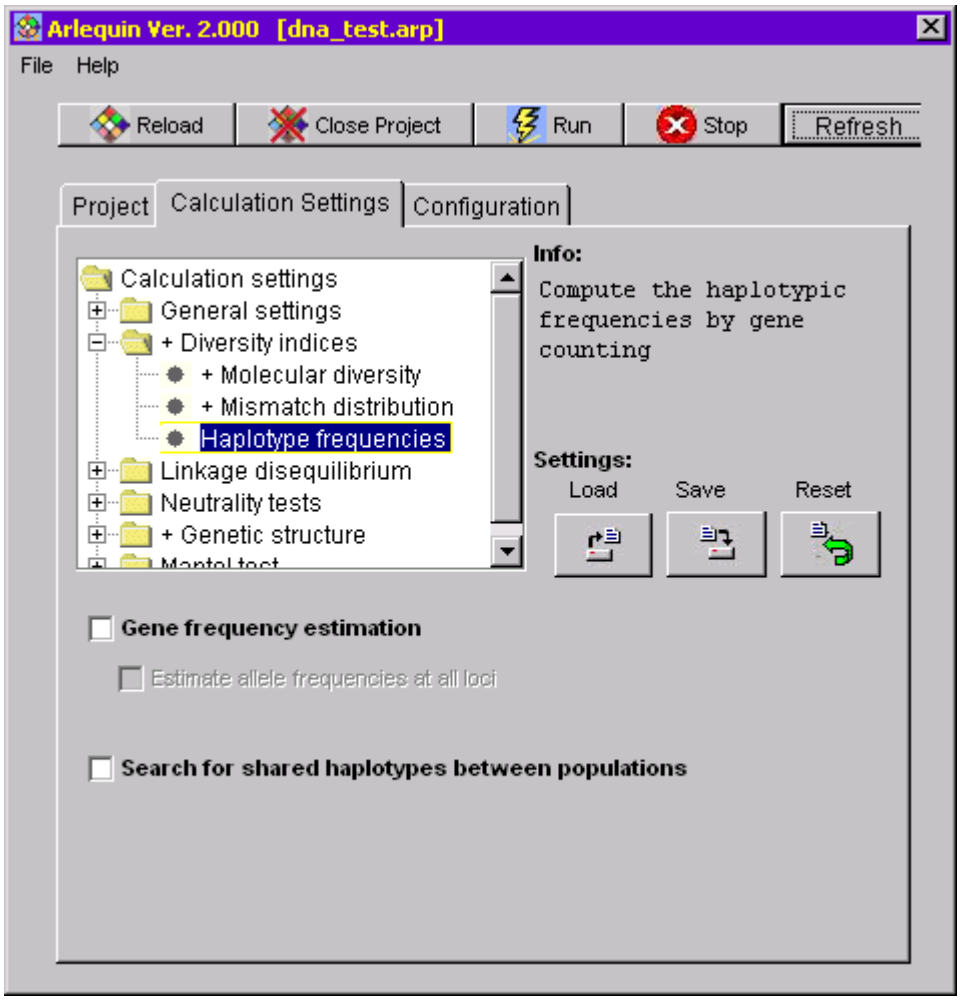
- **Theta(S)** [b]: An estimation of θ obtained from the observed number of segregating site S (see section 7.1.2.3.2).
- **Theta(k)** [b]: An estimation of θ obtained from the observed number of alleles k (see section 7.1.2.3.3).
- **Theta(π)** [b]: An estimation of θ obtained from the mean number of pairwise differences $\hat{\pi}$ (see section 7.1.2.3.4).



- **Mismatch distribution** [b]: Compute the distribution of the observed differences between all pairs of haplotypes in the sample (see section 7.1.2.4). It also estimates parameters of a sudden demographic expansion using a generalized least-square approach, as described in Schneider and Excoffier (1999) (see section 7.1.2.4).
  - **Molecular distance** [l]: Here we only allow one genetic distance: the mere number of observed differences between haplotypes.
  - **Number of bootstrap replicates** [l]: The number of coalescent simulations performed using the estimated parameters of the demographic expansion. The parameters of the stepwise expansion will be re-estimated for each simulation in order to obtain the empirical distribution of the output statistics such as the sum of squared deviations between the observed and the expected mismatch, the raggedness index, or percentile values for each point of the expected mismatch (see section 7.1.2.4) .
- **Haplotype frequencies:**

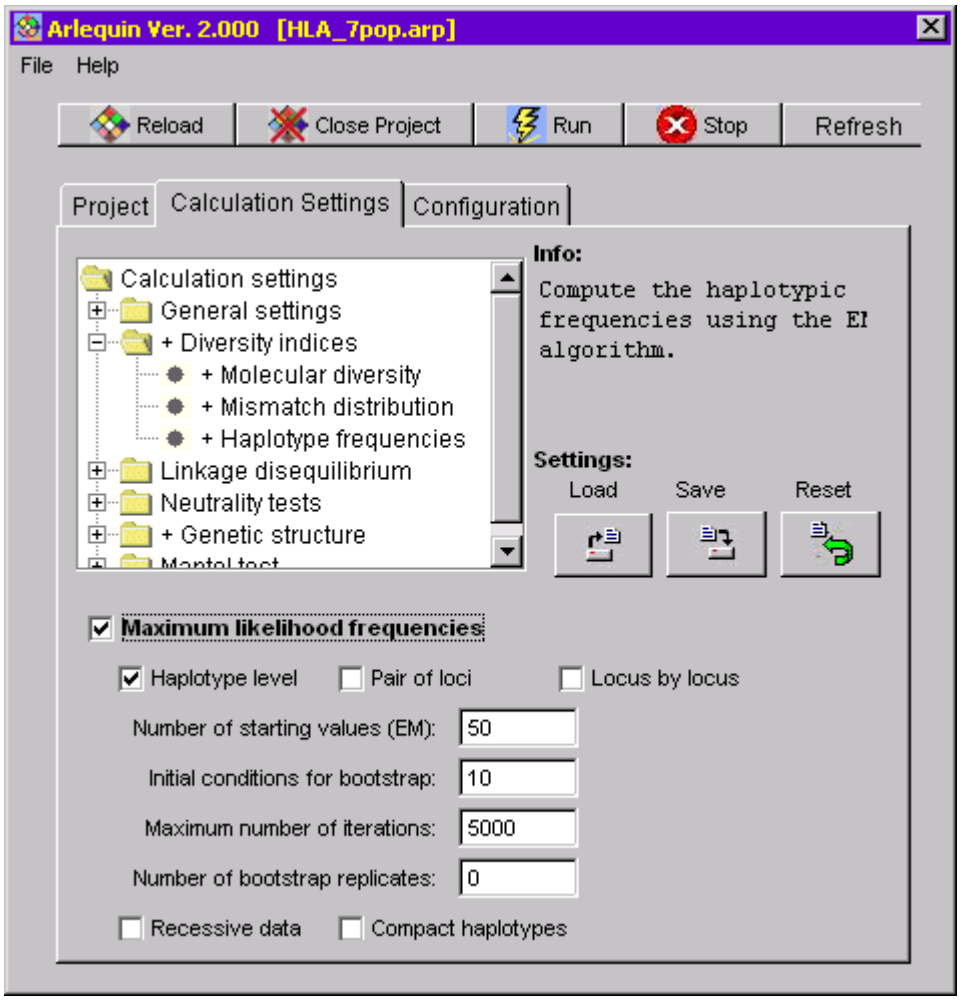
Depending on the data type, different methods are used to estimate the haplotypic frequencies.

Case a: Haplotypic data, or genotypic (diploid) data with known gametic phase



- **Gene frequency estimation** [b]: Estimate the maximum-likelihood haplotype frequencies from the observed data using a mere gene counting procedure
  - **Estimate allele frequencies at all loci**: Estimate allele frequencies at all loci separately.
- **Search for shared haplotypes within and between populations** [b]: Look for haplotypes that are effectively similar after computing pairwise genetic distances according to the distance calculation settings in the *Polymorphism control* section. For each pair of populations, the shared haplotypes will be printed out. Then will follow a table that contains, for every group of identified haplotypes, its absolute and relative frequency in each population. This task is only possible for haplotypic data.

Case b: Genotypic data with unknown gametic phase



- **Maximum likelihood frequencies** [b]: We estimate the maximum-likelihood (ML) haplotype frequencies from the observed data using an Expectation-Maximization (EM) algorithm for multi-locus genotypic data when the gametic phase is not known, or when recessive alleles are present (see section 7.1.3.2).
  - **Haplotype level** [b]: Estimate haplotype frequencies for haplotypes defined by alleles at all loci.
  - **Pair of loci** [b]: Estimate haplotype frequencies for all haplotypes defined for all pairs of loci, as well as for all loci taken separately. This option can be quite time-consuming when the number of loci is large. The EM procedure is done with the same settings as those used for the haplotypic frequency estimation.
  - **Locus by locus** [b]: Estimate allele frequencies for each locus.

  - **No. of starting values (EM)** [i]: Set the number of random initial conditions from which the EM algorithm is started to repeatedly estimate haplotype frequencies. The haplotype frequencies globally maximizing the likelihood of the sample will be kept eventually. Figures of 100 or more are usually in order.
  - **Initial conditions for bootstrap** [i]: Set the number of initial conditions for the bootstrap procedure. It may be smaller than the number of initial conditions set when estimating the haplotype frequencies, because the bootstrap replicates are quite time-consuming. Setting this number to small values is conservative, in the sense that it usually inflates the standard deviations.

- **Maximum no. of iterations** [i]: Set the maximum number of iterations allowed in the EM algorithm. The iterative process will have at most this number of iterations, but may stop before if convergence has been reached. Here, convergence is reached when the sum of the differences between haplotypes frequencies between two successive iterations is smaller than the epsilon value defined in the **Error! Reference source not found.** section 6.3.2.
- **No. of bootstrap replicates** [i]: Set the number of parametric bootstrap replicates of the EM estimation process on random samples generated from a fictive population having haplotype frequencies equal to previously estimated ML frequencies. This procedure is used to generate the standard deviation of haplotype frequencies. When set to zero, the standard deviations are not estimated.
- **Recessive data** [b]: Specify whether a recessive allele is present. This option applies to all loci. The code for the recessive allele can be specified in the project file (see 3.2.1).
- **Compact haplotypes** [b]: Specify whether haplotypes can be compacted to get rid of monomorphic loci. This option just saves up memory and has no effect on the estimation procedure outcome.

6.3.7.3 Gametic disequilibrium

- **Linkage disequilibrium** [b]: Test for the presence of significant association between pairs of loci.

This test can be done with all data types except FREQUENCY data type. The number of loci can be arbitrary, but if there are less than two polymorphic loci, there is no point performing this test.

Different approaches will be used depending on the data type:

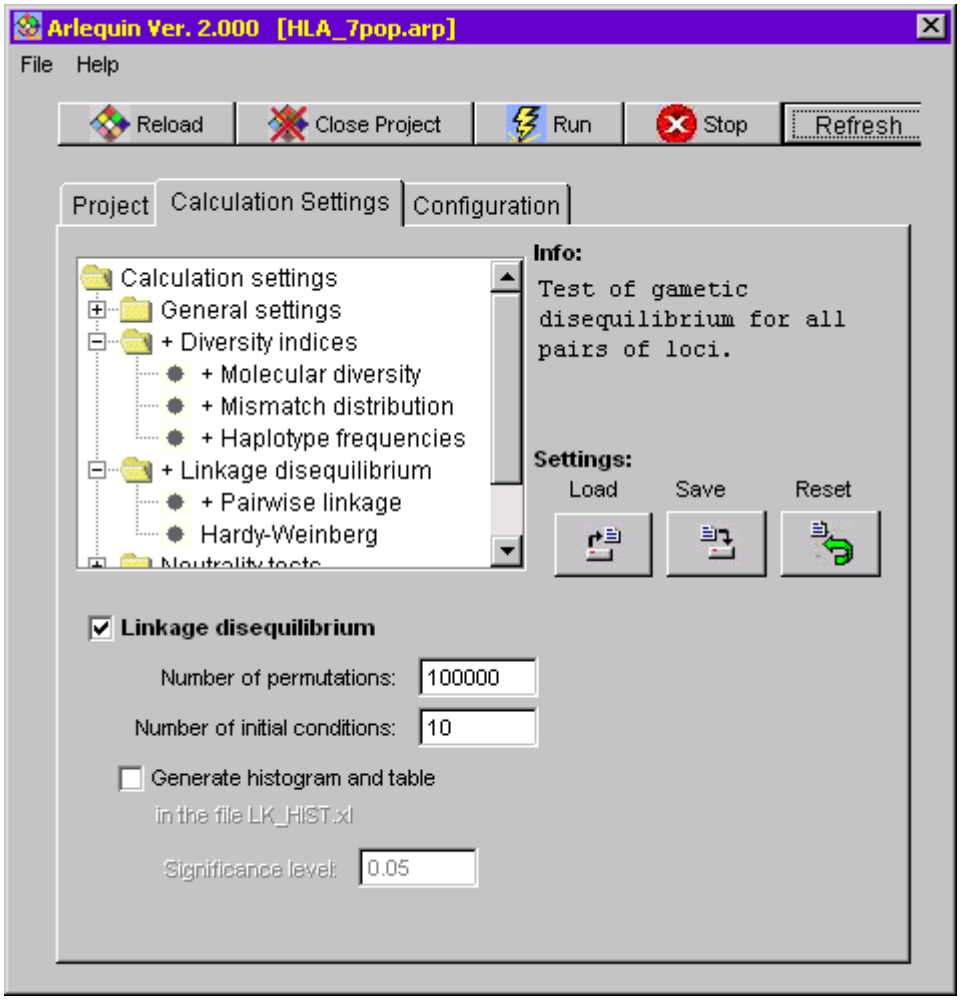
Case a): Genotypic data with unknown gametic phase

A procedure for testing the significance of the association between pairs of loci when the gametic phase is not known (see section 7.1.4.2). The likelihood of the sample under the hypothesis of no association between loci (linkage equilibrium) is compared to the likelihood of the sample when association is allowed (see Slatkin and Excoffier, 1996). The significance of the observed likelihood ratio is found by computing the null distribution of this ratio under the hypothesis of linkage equilibrium, using a permutation procedure.

- **No. of permutations** [i]: Number of random permuted samples to generate. Figures of several thousands are in order, and 16,000 permutations guarantee to have less than 1% difference with the exact probability in 99% of the cases (Guo and Thomson, 1992). A standard error for the estimated *P*-value is estimated using a system of batches (Guo and Thomson, 1992).
  - **No. of initial conditions** [i]: Sets the number of random initial conditions from which the EM is started to repeatedly estimate the sample likelihood. The haplotype frequencies globally maximizing the sample likelihood will be eventually kept. Figures of 100 or more are in order.
  - **Generate histogram and table** [b]: Generates an histogram of the number of loci with which each locus is in disequilibrium, and an s by s table (s being the number of polymorphic loci) summarizing the significant associations between pairs of loci. This table is generated for different levels of polymorphism, controlled by the value y: a locus is declared polymorphic if there are at least 2 alleles with y copies in the sample (Slatkin, 1994a). This is done because the

exact test is more powerful at detecting departure from equilibrium for higher values of y (Slatkin 1994a). The results are output in a file called "*lk_hist.xl*".

- **Significance level** [f]: The level at which the test of linkage disequilibrium is considered significant for the output table.:



Case b): Exact test of linkage disequilibrium

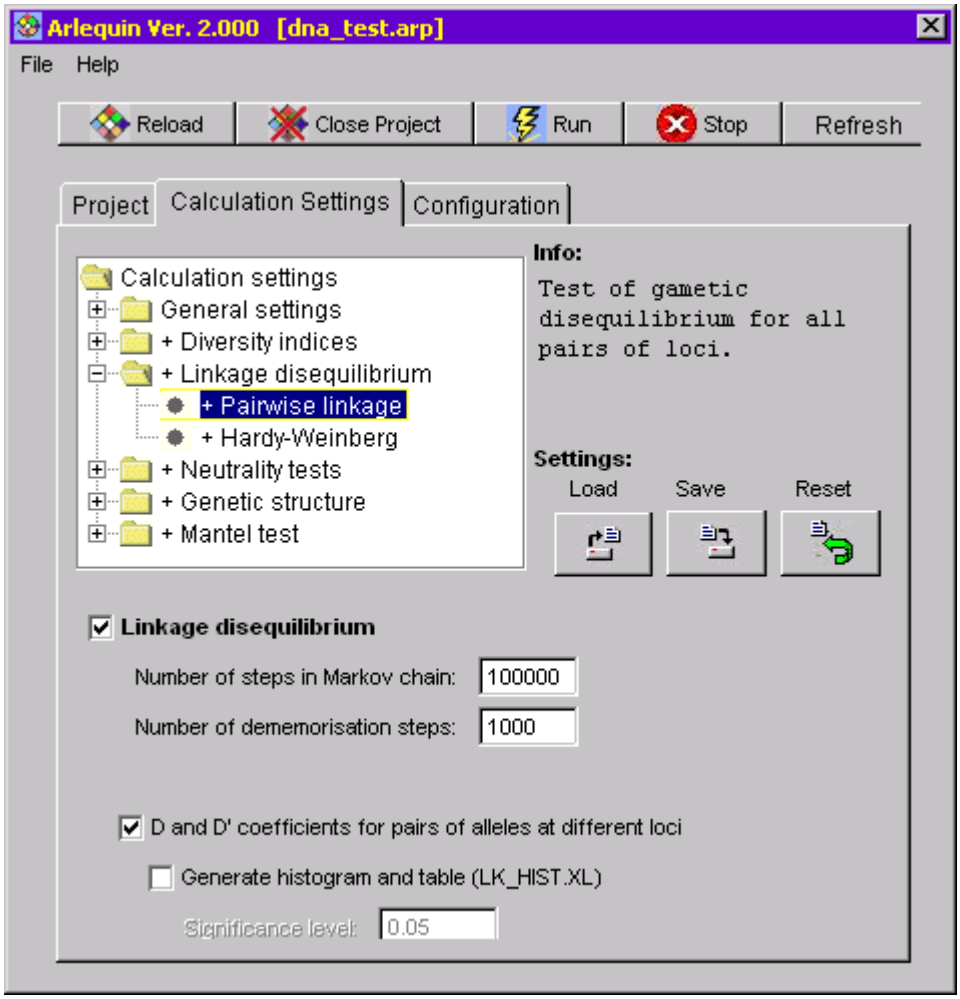
A test analogous to Fisher's exact test on a two-by-two contingency table but extended to a contingency table of arbitrary size (see section 7.1.4.1).

- **No. of steps in Markov chain** [i]: The maximum number of alternative tables to explore. Figures of 100,000 or more are in order. Larger values of the step number increases the precision of the *P*-value as well as its estimated standard deviation.
- **No. of dememorization steps** [i]: The number of steps to perform before beginning to compare the alternative table probabilities to that of the observed table. A few thousands steps are necessary to reach a random starting point corresponding to a table independent from the observed table.
- **Required precision on probability** [f]: The precision required on the inferred probability of linkage equilibrium. A system of batches (Guo and Thomson 1992) is used to constantly estimate the standard-deviation of the probability. The estimation process is stopped once the required precision has been reached, or once the maximal number of steps has been performed.

- **Generate histogram and table** [b]: Generates a histogram of the number of loci with which each locus is in disequilibrium, and an s by s table (s being the number of polymorphic loci) summarizing the significant associations between pairs of loci. This table is generated for different levels of polymorphism, controlled by the value y: a locus is declared polymorphic if there are at least 2 alleles with y copies in the sample (Slatkin, 1994a). This is done because the exact test is more powerful at detecting departure from equilibrium for higher values of y (Slatkin 1994a). The results are output in a file called "*lk_hist.xl*".
- **Significance level** [f]: The level at which the test of linkage disequilibrium is considered significant for the output table.
- ***D* and *D'* coefficients for all pairs of alleles at different loci** [b]:

  See section 7.1.4.3

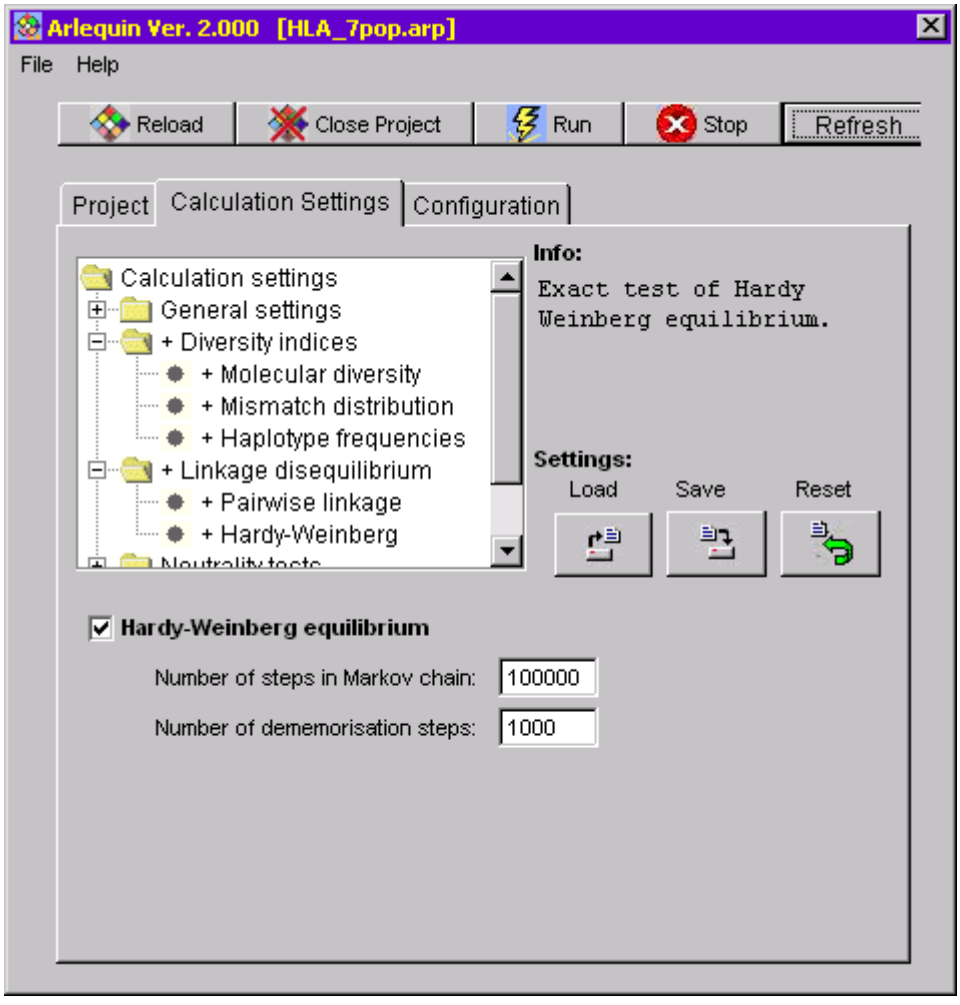
  1. D: The classical linkage disequilibrium coefficient measuring deviation from random association between alleles at different loci (Lewontin and Kojima, 1960) expressed as $D = p_{ij} - p_i p_j$.
  2. D': The linkage disequilibrium coefficient D standardized by the maximum value it can take ($D_{\max}$), given the allele frequencies (Lewontin 1964).



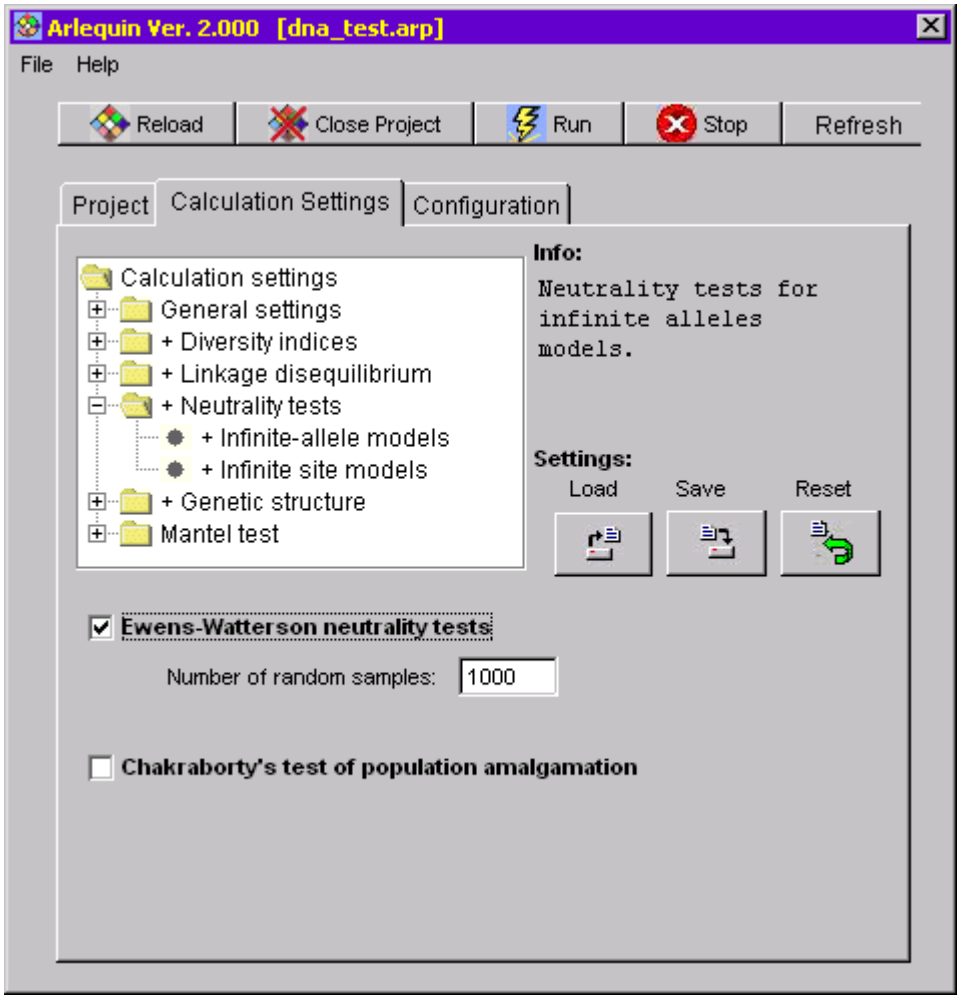
- **Hardy-Weinberg equilibrium** [b]: Test of the hypothesis that the observed diploid genotypes are the product of a random union of gametes. This test is only possible for genotypic data. Separate tests are carried out at each locus.

  This test is analogous to Fisher's exact test on a two-by-two contingency table but extended to a contingency table of arbitrary size (see section 7.1.5). If the gametic phase is unknown the test is only possible locus by locus. For data with known gametic phase, it is also possible to test the association at the haplotypic level within individuals.

  - **No. of steps in Markov chain** [i]: The maximum number of alternative tables to explore. Figures of 100,000 or more are in order.
  - **No. of dememorisation steps** [i]: The number of steps to perform before beginning to compare the alternative table probabilities to that of the observed table. A few thousands steps are necessary to reach a random starting point corresponding to a table independent from the observed table.



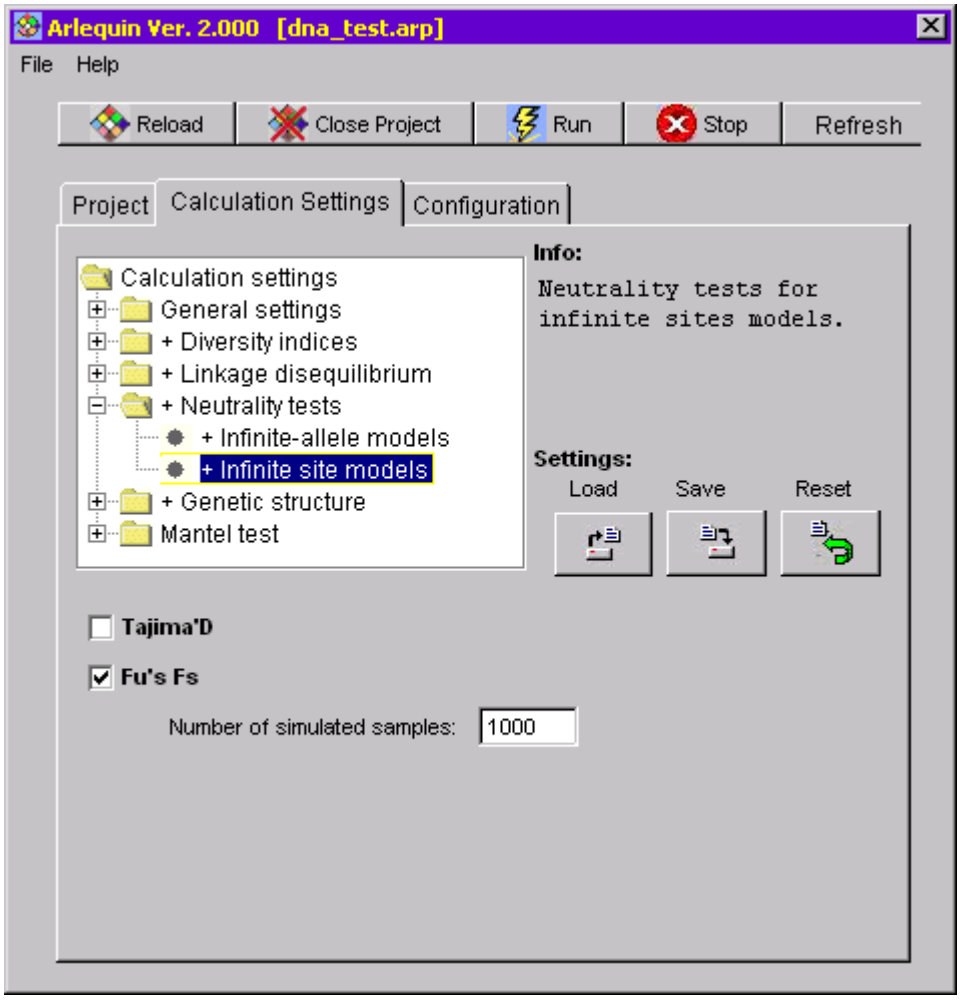
6.3.7.4 Neutrality tests



Tests of selective neutrality, based either on the infinite-allele model or on the infinite-site model (see section 7.1.6).

- **Ewens-Watterson neutrality tests** [b]: Performs tests of selective neutrality based on Ewens sampling theory in a population at equilibrium (Ewens 1972). These tests are currently limited to sample sizes of 2000 genes or less and 1000 different alleles (haplotypes) or less.
  - Ewens-Watterson homozygosity test: This test, devised by Watterson (1978, 1986), is based on Ewens' sampling theory, but uses as a statistic the quantity F equal to the sum of squared allele frequencies, equivalent to the sample homozygosity in diploids (see section 7.1.6.1).
  - Exact test based on Ewens' sampling theory: In this test, devised by Slatkin (1994b, 1996), the probability of the observed sample is compared to that of a random neutral sample with same number of alleles and identical size. The probability of the sample selective neutrality is obtained as the proportion of random samples, which are less or equally probable than the observed sample.
  - **No. of random samples** [i]: Number of random samples to be generated for the two neutrality tests mentioned above. Values of several thousands are in order, and 16,000 permutations guarantee to have less than 1% difference with the exact probability in 99% of the cases (see Guo and Thomson 1992).
- **Chakraborty's test of population amalgamation** [b]: A test of selective neutrality and population homogeneity and equilibrium (Chakraborty, 1990). This test can be used when sample heterogeneity is

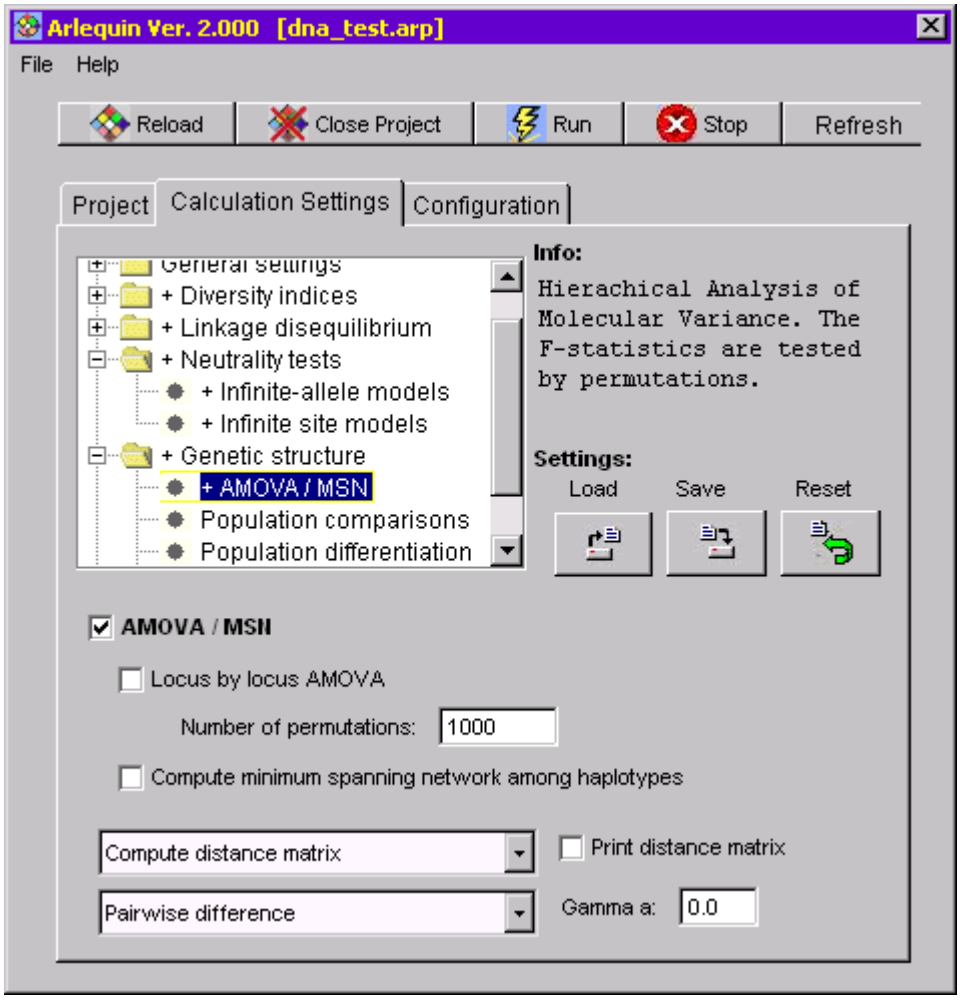
suspected. It uses the observed homozygosity to estimate the population mutation parameter θ_{Hom}. The estimated value of this parameter is then used to compute the probability of observing k alleles or more in a neutral sample drawn from a stationary population. This test is based on Chakraborty's observation that the observed homozygosity is not very sensitive to population amalgamation or sample heterogeneity, whereas the number of observed (low frequency) alleles is more affected by this phenomenon.



- **Tajima's test of selective neutrality** [b]: This test described by Tajima (1989a, 1989b, 1993) compares two estimators of the population parameter θ, one being based on the number of segregating sites in the sample, and the other being based on the mean number of pairwise differences between haplotypes. Under the infinite-site model, both estimators should estimate the same quantity, but differences can arise under selection, population non-stationarity, or heterogeneity of mutation rates among sites (see section 7.1.6.4).

- **Fu's F_S test of selective neutrality** [b]: This test described by Fu (1997) is based on the probability of observing *k* or more alleles in a sample of a given size, conditioned on the observed average number of pairwise differences. The distribution of the statistic is obtained by simulating samples according to a given θ value taken as the average number of pairwise differences. This test has been shown to be especially sensitive to departure from population equilibrium as in case of a population expansion (see section 7.1.6.4).

6.3.7.5 Genetic structure



A dialog box to set up the options for the analysis of population genetic structure, and genetic distances between populations. The genetic structure is analyzed using an analysis of variance framework (Weir and Cockerham, 1984; Excoffier et al. 1992; Weir, 1996).

- **AMOVA / MSN** [b]: Analysis of MOlecular VAriance framework and computation of a Minimum Spanning Network among haplotypes. Estimate genetic structure indices using information on the allelic content of haplotypes, as well as their frequencies (Excoffier et al. 1992). The information on the differences in allelic content between haplotypes is entered as a matrix of Euclidean squared distances. The significance of the covariance components associated with the different possible levels of genetic structure (within individuals, within populations, within groups of populations, among groups) is tested using non-parametric permutation procedures (Excoffier et al. 1992). The type of permutations is different for each covariance component (see section 7.1.7).

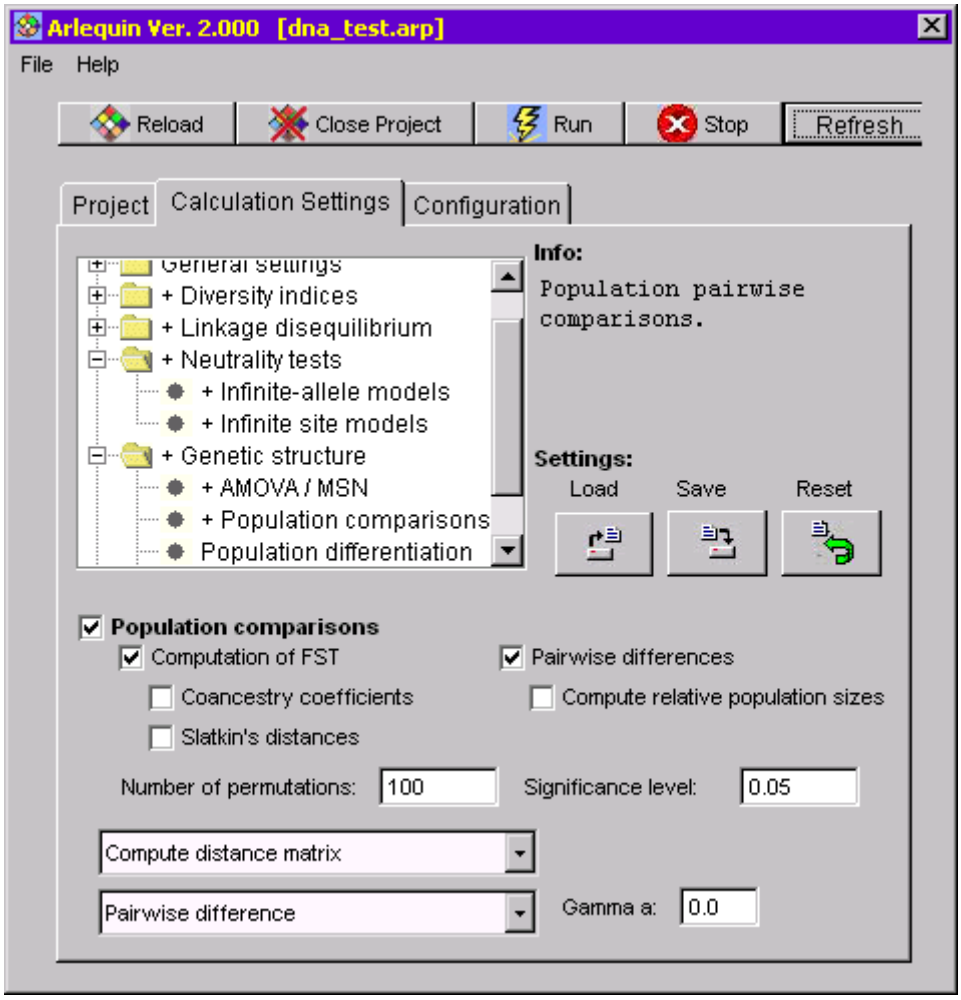
  The minimum spanning tree and network is computed among all haplotypes defined in the samples included in the genetic structure to test (see section 7.1.8).

  The number of hierarchical levels of the variance analysis and the kind of permutations that are done depend on the kind of data, the genetic structure that is tested, and the options the user might choose. All details will be given in section 7.1.7.

- **Locus by locus AMOVA** [b]: A separate AMOVA can be performed for each locus separately. For this purpose, we use the same number of permutations as in the global Amova.
  - **No. of permutations** [i]: Enter the number of permutations used to test the significance of covariance components and fixation indices. A value of zero will not lead to any testing procedure. Values of several thousands are in order for a proper testing scheme, and 16 000 permutations guarantee to have less than 1% difference with the exact probability in 99% of the cases (Guo and Thomson 1992).

    The number of permutations used by the program might be slightly larger. This is the consequence of subdivision of the total number of permutation in batches for estimating the standard error of the *P*-value.

    Note that if several covariance components need to be tested, the probability of each covariance component will be estimated with this number of permutation. The distribution of the covariance components is output into a tabulated text file called *amo_hist.xl*, which can be directly read into MS-EXCEL .
- **Compute minimum spanning network among haplotypes**. A Minimum Spanning Tree and a Minimum Spanning Network are computed from the distance matrix used to perform the AMOVA calculations.
  - **Include individual level for genotype data** [b]: Include the intra-individual covariance component of genetic diversity, and its associated fixation indices. It thus takes into account the differences between genes found within individuals. This is another way to test for global departure from Hardy-Weinberg equilibrium. The selection of this option is only possible for genotypic data with known gametic phase.
  - **Choice of Euclidian square distances** [m]:
    - **Use project distance matrix** [m]**:** Use the distance matrix defined in the project file (if available)
    - **Compute distance matrix** [m]**:** Compute a given distance matrix based on a method defined below. With this setting selected, the distance matrix potentially defined in the project file will be ignored. This matrix can be generated either for haplotypic data or genotypic data (Michalakis and Excoffier, 1996)
    - **Use conventional F-statistics** [m]**:** With this setting activated, we will use a lower diagonal distance matrix, with zeroes on the diagonal and ones as off-diagonal elements. It means that all distances between non-identical haplotypes will be considered as identical, implying that one will bas the analysis of genetic structure only on allele frequencies.
  - **Distance between haplotypes** [m]**:** Select a distance method to compute the distances between haplotypes. Different square Euclidean distances can be used depending on the type of data analyzed.
  - **Gamma *a* value** [f]: Set the value for the shape parameter a of the gamma function, when selecting a distance allowing for unequal mutation rates among sites. See the Molecular diversity section 7.1.2.5.

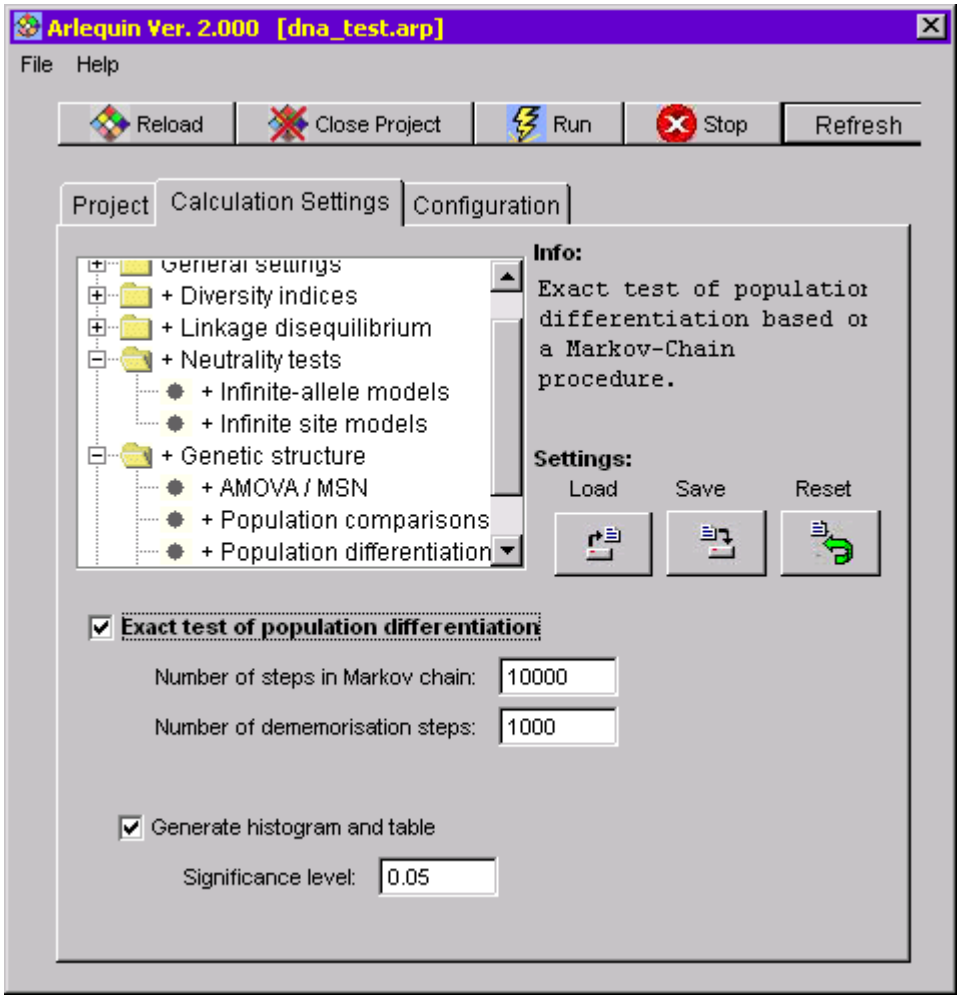


- **Population comparisons** [b]: Computes different indexes of dissimilarities (genetic distances) between pairs of populations, like F_{ST} statistics and transformed pairwise F_{ST} 's that can be used as short term genetic distances between populations (Reynolds et al. 1983; Slatkin, 1995), but also Nei's mean number of pairwise differences within and between pairs of populations.

  The significance of the genetic distances is tested by permuting the haplotypes or individuals between the populations. See section **7.1.9** for more details on the output results (genetic distances and migration rates estimates between populations).

- **Computation of F_{ST}** [b]: Computes pairwise F_{ST} 's for all pairs of populations
  - **Reynolds's distance** [b]: Computes Reynolds' et al. (1983) linearized F_{ST} for short divergence time (see section 7.1.10.1).
  - **Slatkin's distances** [b]: Computes Slatkin's (1995) genetic distance derived from pairwise F_{ST} (see section 7.1.10.2).
- **Pairwise differences** [b]: Computes Nei's average number of pairwise differences within and between populations (Nei and Li, 1979) (see section 7.1.10.4)
  - **Compute relative population sizes** [b]: Computes relative population sizes for al pairs of populations, as well as divergence times between populations taking into account these potential differences between population sizes (Gaggiotti and Excoffier 2000) (see section 7.1.10.5)

- **No. of permutations** [i]: Enter the required number of permutations to test the significance of the derived genetic distances.. If this number is set to zero, no testing procedure will be performed. Note that this procedure is quite time consuming when the number of populations is large.
- **Choice of Euclidian distance** [m]:
  - **Use project distance matrix** [m]**:** Use the distance matrix defined in the project file (if available)
  - **Compute distance matrix** [m]**:** Compute a given distance matrix based on a method defined below. With this setting selected, the distance matrix potentially defined in the project file will be ignored. This matrix can be generated either for haplotypic data or genotypic data (Michalakis and Excoffier, 1996)
  - **Use conventional F-statistics** [m]**:** With this setting activated, we will use a lower diagonal distance matrix, with zeroes on the diagonal and ones as off-diagonal elements. It means that all distances between non-identical haplotypes will be considered as identical, implying that one will bas the analysis of genetic structure only on allele frequencies.
- **Distance between haplotypes** [m]**:** Select a distance method to compute the distances between haplotypes. Different square Euclidean distances can be used depending on the type of data analyzed.
- **Gamma *a* value** [f]: Set the value for the shape parameter a of the gamma function, when selecting a distance allowing for unequal mutation rates among sites. See the Molecular diversity section 7.1.2.5

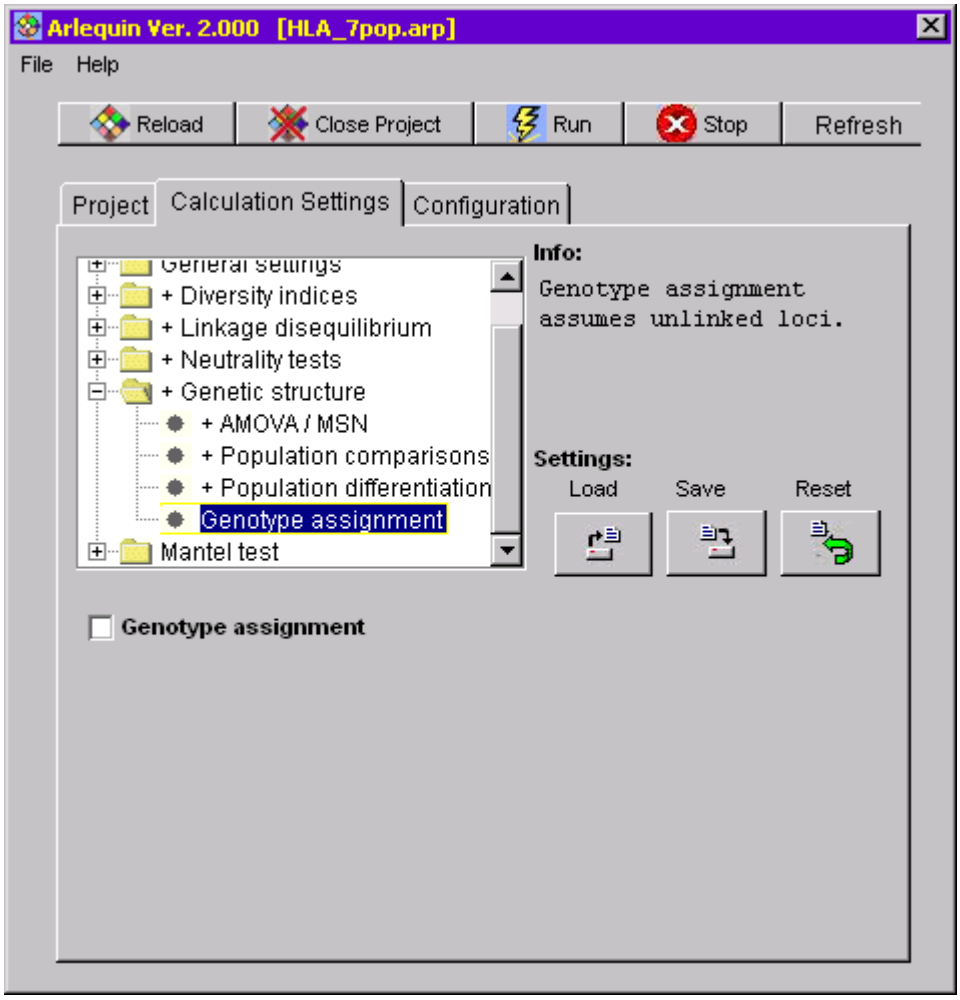


- **Exact test of population differentiation** [b]: We test the hypothesis of random distribution of the individuals between pairs of populations as described in Raymond and Rousset (1995) and Goudet et al.

(1996). This test is analogous to Fisher's exact test on a two-by-two contingency table, but extended to a contingency table of size two by (no. of haplotypes). We do also an exact differentiation test for all populations defined in the project by constructing a table of size (no. of populations) by (no. of haplotypes). (Raymond and Rousset, 1995).

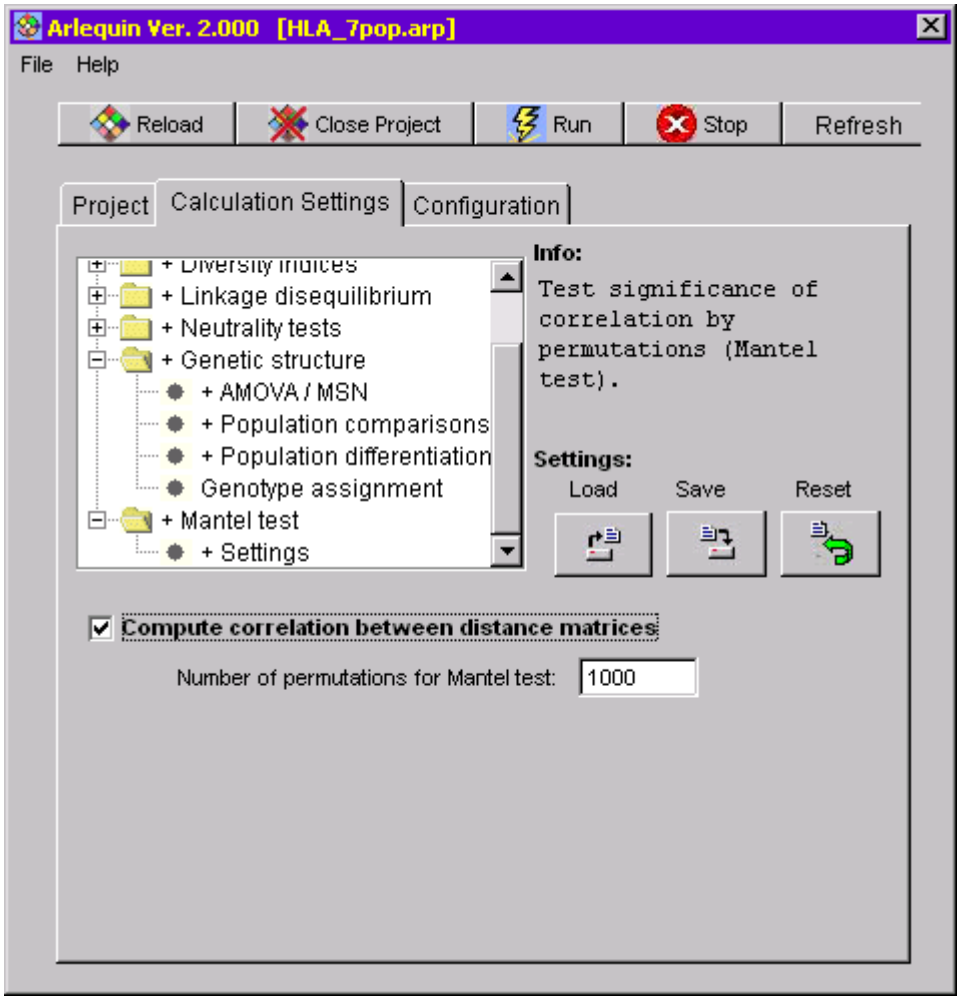
- **No. of steps in Markov chain** [i]: The maximum number of alternative tables to explore. Figures of 100,000 or more are in order. Larger values of the step number increases the precision of the *P*-value as well as its estimated standard deviation.
- **No. of dememorisation steps** [i]: The number of steps to perform before beginning to compare the alternative table probabilities to that of the observed table. A few thousands steps are necessary to reach a random starting point corresponding to a table independent from the observed table.
- **Generate histogram and table** [b]: Generates a histogram of the number of populations which are significantly different from a given population, and a *s* by *s* table (*s* being the number of populations) summarizing the significant associations between pairs of populations. An association between two populations is considered as significant or not depending on the significance level specified below.
- **Significance level** [f]: The level at which the test of differentiation is considered significant for the output table. If the *P*-value is smaller than the *Significance level*, then the two populations are considered as significantly different.

- **Genotype assignment**



- **Genotype assignment**: Computes the log likelihood of the genotype of each individual in every sample, as if it was drawn from a population sample having allele frequencies equal to those estimated for each sample (Paetkau et al. 1997; Waser and Strobeck, 1998). Multi-locus genotype likelihoods are computed as the product of each locus likelihood, thus assuming that the loci are independent. The output result file lists, for each population, a table of the log-likelihood of each individual genotype in all populations (see section 7.1.12).

6.3.7.6 Mantel test



- **Mantel test of correlation between distance matrices**: Test the correlation or the partial correlations between 2 or 3 matrices by a permutation procedure (Mantel, 1967; Smouse et al. 1986).
  - **Number of permutations**: Sets the number of permutations for the Mantel test.

7 METHODOLOGICAL OUTLINES

The following table gives a rapid overview of the methods implemented in Arlequin. A ✓ indicates that the task corresponding to the table entry is possible. Some tasks are only possible or meaningful if there is no recessive data, and those cases are marked with a ✕

	Data types									
	DNA and RFLP			**Microsat**			**Standard**			**Frequency**
Types of computations	**G+**	**G-**	**H**	**G+**	**G-**	**H**	**G+**	**G-**	**H**	
Standard indices ✕	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
Molecular diversity ✕	✓	✓	✓	✓	✓	✓	✓	✓	✓	
Mismatch distribution	✓		✓	✓		✓	✓		✓	
Haplotype frequency estimation	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
Linkage disequilibrium	✓	✓	✓	✓	✓	✓	✓	✓	✓	
Hardy-Weinberg equilibrium ✕	✓	✓		✓	✓		✓	✓		
Tajima's neutrality test	✓		✓							
Fu's neutrality test	✓		✓							
Ewens-Watterson neutrality tests	✓		✓				✓		✓	✓
Chakraborty's amalgamation test	✓		✓				✓		✓	✓
Search for shared haplotypes between samples			✓			✓			✓	✓
AMOVA ✕	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
Minimum Spanning Network[1]	✓		✓	✓		✓	✓		✓	
Pairwise genetic distances ✕	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
Exact test of population differentiation ✕	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
Individual assignment test ✕		✓			✓			✓		
Mantel test	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓

G+: Genotypic data, gametic phase known

G- : Genotypic data, gametic phase unknown

H : Haplotypic data

[1] Computation of minimum spanning network between haplotypes is only possible if a distance matrix is provided or if it can be computed from the data.

7.1 Intra-population level methods

7.1.1 Standard diversity indices

7.1.1.1 Gene diversity

Equivalent to the expected heterozygosity for diploid data. It is defined as the probability that two randomly chosen haplotypes are different in the sample. Gene diversity and its sampling variance are estimated as

$$\hat{H} = \frac{n}{n-1}(1-\sum_{i=1}^{k} p_i^2)$$

$$\mathrm{V}(\hat{H}) = \frac{2}{n(n-1)}\left\{2(n-2)\left[\sum_{i=1}^{k} p_i^3 - (\sum_{i=1}^{k} p_i^2)^2\right] + \sum_{i=1}^{k} p_i^2 - (\sum_{i=1}^{k} p_i^2)^2\right\},$$

where n is the number of gene copies in the sample, k is the number of haplotypes, and p_i is the sample frequency of the i-th haplotype.

Reference:

Nei, 1987, p.180.

7.1.1.2 Number of usable loci

Number of loci that present less than a specified amount of missing data. The maximum amount of missing data must be specified in the *General Settings* dialog box.

7.1.1.3 Number of polymorphic sites (S)

Number of usable loci that present more than one allele per locus.

7.1.2 Molecular indices

7.1.2.1 Mean number of pairwise differences (π)

Mean number of differences between all pairs of haplotypes in the sample. It is given by

$$\hat{\pi} = \sum_{i=1}^{k} \sum_{j<i} p_i p_j \hat{d}_{ij},$$

where $\hat{d}_{ij}$ is an estimate of the number of mutations having occurred since the divergence of haplotypes i and j, , k is the number of haplotypes, and p_i is the frequency of haplotype i. The total variance (over the stochastic and the sampling process), assuming no recombination between sites and selective neutrality, is obtained as

$$V(\hat{\pi}) = \frac{3n(n+1)\hat{\pi} + 2(n^2+n+3)\hat{\pi}^2}{11(n^2-7n+6)}.$$ (Tajima, 1993)

Note that similar formulas are also used for *Microsat* and *Standard* data, even though the underlying assumptions of the model may be violated.

References:

Tajima, 1983

Tajima, 1993

7.1.2.2 Nucleotide diversity or average gene diversity over L loci (RFLP and DNA data)

It is the probability that two randomly chosen homologous nucleotides are different. It is equivalent to the gene diversity at the nucleotide level.

$$\hat{\pi}_n = \frac{\sum_{i=1}^{k} \sum_{j<i} p_i p_j \hat{d}_{ij}}{L}$$

$$V(\hat{\pi}_n) = \frac{n+1}{3(n-1)L}\hat{\pi}_n + \frac{2(n^2+n+3)}{9n(n-1)}\hat{\pi}_n^2$$

Note that similar formulas are used for computing the average gene diversity over L loci for Microsat and Standard data, assuming no recombination and selective neutrality. As above, one should be aware that these assumption may not hold for these data types.

References:

Tajima, 1983

Nei, 1987, p. 257

7.1.2.3 Theta estimators

Several methods are used to estimate the population parameter $\theta = 2Mu$, where M is equal to $2N$ for diploid populations of size N, or equal to N for haploid populations, and u is the overall mutation rate at the haplotype level.

7.1.2.3.1 Theta(Hom)

The expected homozygosity in a population at equilibrium between drift and mutation is usually given by

$$H = \frac{1}{\theta+1}.$$

However, Zouros (1979) has shown that this estimator was an overestimate when estimated from a single or a few loci. Although he gave no closed form solution, Chakraborty and Weiss (1991) proposed to iteratively solve the following relationship between the expectation of $\hat{\theta}_H$ and the unknown parameter θ

$$E(\hat{\theta}_H) = \theta\left(1 + \frac{2(1+\theta)}{(2+\theta)(3+\theta)}\right)$$ (Zouros, 1979)

starting with a first estimate of $\hat{\theta}_H$ of $(1-H)/H$, and equating it to its expectation.

Chakraborty and Weiss (1991) give an approximate formula for the standard error of $\hat{\theta}_H$ as

$$\text{s.d.}(\hat{\theta}_H) \approx \frac{(2+\theta)^2(3+\theta)^2\,\text{s.d.}(H)}{H^2(1+\theta)\left[(2+\theta)(3+\theta)(4+\theta)+10(2+\theta)+4\right]},$$

where $\text{s.d.}(H)$ is the standard error of H given in section 7.1.1.1.

7.1.2.3.2 Theta(*S*)

$\hat{\theta}_S$ is estimated from the infinite-site equilibrium relationship (Watterson, 1975) between the number of segregating sites (S), the sample size (n) and θ for a sample of non-recombining DNA:

$$\theta = \frac{S}{a_1}$$

where

$$a_1 = \sum_{i=1}^{n-1} \frac{1}{i}.$$

The variance of $\hat{\theta}_S$ is obtained as

$$\text{V}(\hat{\theta}_S) = \frac{a_1^2 S + a_2 S^2}{a_1^2(a_1^2 + a_2)}, \qquad \text{(Tajima, 1989)}$$

where

$$a_2 = \sum_{i=1}^{n-1} \frac{1}{i^2}$$

7.1.2.3.3 Theta(*k*)

$\hat{\theta}_k$ is estimated from the infinite-allele equilibrium relationship (Ewens, 1972) between the expected number of alleles (k), the sample size (n) and θ :

$$\text{E}(k) = \theta \sum_{i=0}^{n-1} \frac{1}{\theta + i}$$

Instead of the variance of $\hat{\theta}_k$, we give the limits ($\hat{\theta}_0$ and $\hat{\theta}_1$) of a 95% confidence interval around $\hat{\theta}_k$, obtained from Ewens (1972)

$$\Pr(\text{less than } k \text{ alleles} \mid \theta = \theta_0) = 0.025$$

$$\Pr(\text{ more than } k \text{ alleles} \mid \theta = \theta_1) = 0.025,$$

These probabilities are obtained by summing up the probabilities of observing k' alleles (k'=0,...,k), obtained as (Ewens, 1972)

$$\Pr(K = k \mid \theta) = \frac{\left|S_n^k\right| \theta^k}{S_n(\theta)}$$

where $\left|S_n^k\right|$ is a Stirling number of the first kind (see Abramovitz and Stegun, 1970), and $S_n(\theta)$ is defined as $\theta(\theta+1)(\theta+2)\ldots(\theta+n-1)$.

7.1.2.3.4 Theta(π)

$\hat{\theta}_\pi$ is estimated from the infinite-site equilibrium relationship between the mean number of pairwise differences ($\hat{\pi}$) and theta (θ):

$$\mathrm{E}(\hat{\pi}) = \theta \ , \qquad \text{(Tajima, 1983)}$$

and its variance $\mathrm{V}(\hat{\pi})$ is given in section 7.1.1.1.

7.1.2.4 Mismatch distribution

It is the distribution of the observed number of differences between pairs of haplotypes. This distribution is usually multimodal in samples drawn from populations at demographic equilibrium, as it reflects the highly stochastic shape of gene trees, but it is usually unimodal in populations having passed through a recent demographic expansion (Rogers and Harpending, 1992; Hudson and Slatkin, 1991).

If one assumes that a stationary haploid population at equilibrium has suddenly passed τ generations ago from a population size of N_0 to N_1, then the probability of observing S differences between two randomly chosen non-recombining haplotypes is given by

$$F_S(\tau,\theta_0,\theta_1) = F_S(\theta_1) + exp(-\tau\frac{\theta_1+1}{\theta_1})\sum_{j=0}^{S}\frac{\tau^j}{j!}\left[F_{S-j}(\theta_0) - F_{S-j}(\theta_1)\right], \qquad \text{(Li, 1977)}$$

where $F_S(\theta) = \dfrac{\theta^S}{(\theta+1)^{S+1}}$ is the probability of observing two random haplotypes with S differences in a stationary population (Watterson, 1975), $\theta_0 = 2uN_0$, $\theta_1 = 2uN_1$, $\tau = 2ut$, and u is the mutation rate for the whole haplotype.

Rogers (1995) has simplified the above equation, by assuming that $\theta_1 \to \infty$, implying there are no coalescent events after the expansion, which is only reasonable if the expansion size is large. With this simplifying assumption, it is possible to derive the moment estimators of the time to the expansion (τ) and the mutation parameter θ_0, as

$$\begin{aligned}\hat{\theta}_0 &= \sqrt{v - m} \\ \hat{\tau} &= m - \hat{\theta}_0\end{aligned}, \qquad \text{(Rogers, 1995)}$$

where m and v are the mean and the variance of the observed mismatch distribution, respectively. These estimators can then be used to plot $F_S(\tau, \theta_0, \infty)$ values. Note, however, that this estimation cannot be done if the variance of the mismatch is smaller than the mean.

However, Schneider and Excoffier (1999) find that this moment estimator often leads to an underestimation of the age of the expansion (τ). They rather propose to estimate the parameters of the demographic expansion by a generalized non-linear least-square approach. This is the method we now use to estimate the parameters of the demographic expansion τ, θ_0, and θ_1.

Approximate confidence intervals for those parameters are obtained by a parametric bootstrap approach. The principle is the following: We computed approximate confidence intervals for the estimated parameters $\hat{\theta}_1, \hat{\theta}_0$ and $\hat{\tau}$ using a parametric bootstrap approach (Schneider and Excoffier, 1999) generating percentile confidence intervals (see e.g. Efron, 199, p. 53 and chap. 13).

- We generate a large number (*B*) of random samples according to the estimated demography, using a coalescent algorithm modified from Hudson (1990).
- For each of the *B* simulated data sets, we reestimate τ, θ_0, and θ_1 to get *B* bootstrapped values θ_0^*, θ_1^* and τ^*.
- For a given confidence level α, the approximate limits of the confidence interval were obtained as the $\alpha/2$ and $1-\alpha/2$ percentile values (Efron, 1993, p. 168).

It is important to underline that this form of parametric bootstrap assumes that the data are distributed according the sudden expansion model. In Schneider and Excoffier (1999), we showed by simulation that only the confidence interval (CI) for τ has a good coverage (i.e. that the true value of the parameter is included in a 100x(1-α)% CI with a probability very close to 1-α.). The CI of the other two parameters are overly large (the true value of the parameter was almost always included in the CI), and thus too conservative.

The validity of the estimated stepwise expansion model is tested using the same parametric bootstrap approach as described above. We used here the sum of square deviations (*SSD*) between the observed and the expected mismatch as a test statistic. We obtained its distribution under the hypothesis that the estimated parameters are the true ones, by simulating *B* samples around the estimated parameters. As before, we re-estimated each time new parameters θ_0^*, θ_1^* and τ^*, and computed their associated sums of squares SSD_{sim}. The P-value of the test is therefore approximated by

$$P = \frac{\text{number of } SSD_{sim} \text{ larger or equal to } SSD_{obs}}{B}.$$

For convenience, we also compute the raggedness index of the observed distribution defined by Harpending (1994) as

$$r = \sum_{i=1}^{d+1} (x_i - x_{i-1})^2 ,$$

where d is the maximum number of observed differences between haplotypes, and the x's are the observed relative frequencies of the mismatch classes. This index takes larger values for multimodal distributions commonly found in a stationary population than for unimodal and smoother distributions typical of expanding populations. Its significance is tested similarly to that of *SSD*.

7.1.2.5 Estimation of genetic distances between DNA sequences

Definitions:

L:	Number of loci
Gamma correction:	This correction is proposed when the mutation rates cannot be assumed as uniform for all sites. It had been originally proposed for mutation rates among amino acids (Uzell and Corbin, 1971), but it seems also to be the case of the control region of human mtDNA (Wakeley, 1993). In such a case, a Gamma distribution of mutation rates is often assumed. The shape of this distribution (the unevenness of the mutation rates) is mainly controlled by a parameter a, which is the inverse of the coefficient of variation of the mutation rate. The smaller the a coefficient , the more uneven the mutation rates. A uniform mutation rate corresponds to the case where a is equal to infinity.
n_d :	Number of observed substitutions between two DNA sequences
n_s :	Number of observed transitions between two DNA sequences
n_v :	Number of observed transversions between two DNA sequences
ω	G+C ratio, computed on all the DNA sequences of a given sample

7.1.2.5.1 Pairwise difference

Outputs the number of loci for which two haplotypes are different

$$\hat{d} = n_d$$

$$\mathrm{V}(\hat{d}) = \hat{d}(L - \hat{d}) / L$$

7.1.2.5.2 Percentage difference

Outputs the percentage of loci for which two haplotypes are different

$$\hat{d} = n_d / L$$

$$\mathrm{V}(\hat{d}) = \hat{d}(1-\hat{d}) / L$$

7.1.2.5.3 Jukes and Cantor

Outputs a corrected percentage of nucleotides for which two haplotypes are different.

The correction allows for multiple substitutions per site since the most recent common ancestor of the two DNA sequences. The correction also assumes that the rate of nucleotide substitution is identical for all 4 nucleotides A, C, G and T.

$$\hat{p} = n_d / L$$

$$\hat{d} = -\frac{3}{4}\log(1-\frac{4}{3}\hat{p})$$

$$\mathrm{V}(\hat{d}) = \frac{\hat{p}(1-\hat{p})}{(1-\frac{4}{3}\hat{p})^2 L}$$

Gamma correction:

$$\hat{d} = -\frac{3}{4}a\left[\ (1-\frac{4}{3}p)^{-1/a} - 1\ \right]$$

$$\mathrm{V}(\hat{d}) = p(1-p)\left[\ (1-\frac{4}{3}p)^{-2(1/a+1)}\ \right] / L$$

References:

Jukes and Cantor 1969
Jin and Nei 1990
Kumar et al. 1993

7.1.2.5.4 Kimura 2-parameters

Outputs a corrected percentage of nucleotides for which two haplotypes are different.

The correction also allows for multiple substitutions per site, but takes into account different substitution rates between transitions and transversions. The transition-transversion ratio is estimated from the data.

$$\hat{P} = \frac{n_s}{L}, \quad \hat{Q} = \frac{n_v}{L}$$

$$c_1 = 1/(1-2\hat{P}-\hat{Q}), c_2 = 1/(1-2\hat{Q}), c_3 = \frac{c_1 + c_2}{2}$$

$$\hat{d} = \frac{1}{2}\log(1-2\hat{P}-\hat{Q}) - \frac{1}{4}\log(1-2\hat{Q})$$

$$\mathrm{V}(\hat{d}) = \frac{c_1^2 P + c_3^2 \hat{Q} - (c_1\hat{P} + c_3\hat{Q})^2}{L}$$

Gamma correction:

$$c_1 = (1-2\hat{P}-\hat{Q})^{-(1/a+1)}, c_2 = (1-2\hat{Q})^{-(1/a+1)}, c_3 = \frac{c_1 + c_2}{2}$$

$$\hat{d} = \frac{a}{2}\left[\ (1-2\hat{P}-\hat{Q})^{-1/a} + \frac{1}{2}(1-2\hat{Q})^{-1/a} - \frac{3}{2}\ \right]$$

$$\mathrm{V}(\hat{d}) = \frac{c_1^2 \hat{P} + c_3^2 \hat{Q} - (c_1\hat{P} + c_3\hat{Q})^2}{L}$$

References:

Kimura (1980)
Jin and Nei (1990)

7.1.2.5.5 Tamura

Outputs a corrected percentage of nucleotides for which two haplotypes are different.

The correction is an extension of Kimura 2-parameters method, allowing for unequal nucleotide frequencies. The transition-transversion ratios, as well as the overall nucleotide frequencies are computed from the original data.

$$\hat{P} = \frac{n_s}{L}, \quad \hat{Q} = \frac{n_v}{L}$$

$$c_1 = \frac{1}{1 - \dfrac{\hat{P}}{2\omega(1-\omega)}}, \quad c_2 = \frac{1}{1-2\hat{Q}}, \quad c_3 = 2\omega(1-\omega)(c_1 - c_2) + c_2$$

$$\hat{d} = -2\omega(1-\omega)\log\left(1 - \frac{\hat{P}}{2\omega(1-\omega)} - \hat{Q}\right) - \frac{1}{2}(1-2\omega(1-\omega))\log(1-2\hat{Q})$$

$$V(\hat{d}) = \frac{c_1^2 \hat{P} + c_3^2 \hat{Q} - (c_1 \hat{P} + c_3 \hat{Q})^2}{L}$$

References:

Tamura, 1992,

Kumar et al. 1993

7.1.2.5.6 Tajima and Nei

Outputs a corrected percentage of nucleotides for which two haplotypes are different.

The correction is an extension of Jukes and Cantor method, allowing for unequal nucleotide frequencies. The overall nucleotide frequencies are computed from the data.

$$\hat{p} = \frac{n_d}{L}, \quad b = \frac{1}{2}(1 - \sum_{i=1}^{4} g_i^2 + \frac{\hat{p}^2}{c}, \quad c = \sum_{i=1}^{3} \sum_{j=i+1}^{4} \frac{x_{ij}^2}{2 g_i g_j},$$

where the g's are the four nucleotide frequencies, and x_{ij} is the relative frequency of the nucleotide pair i and j .

$$\hat{d} = -b \log(1 - \frac{\hat{p}}{b})$$

$$V(\hat{d}) = \frac{\hat{p}(1-\hat{p})}{(1 - \frac{\hat{p}}{b})^2 L}$$

References:

Tajima and Nei, 1984,

Kumar et al. 1993

7.1.2.5.7 Tamura and Nei

Outputs a corrected percentage of nucleotides for which two haplotypes are different.

Like Kimura 2-parameters, and Tajima and Nei distances, the correction allows for different transversion and transition rates, but a distinction is also made between transition rates between purines and between pyrimidines.

$$c_1 = \frac{2 g_A g_G}{g_R}, \quad c_2 = \frac{2 g_C g_T}{g_Y}, \quad c_3 = \frac{2 g_A g_G g_R}{2 g_A g_G g_R - g_R^2 \hat{P}_1 - g_A g_G \hat{Q}}$$

$$c_4 = \frac{2 g_T g_C g_Y}{2 g_T g_C g_Y - g_Y^2 \hat{P}_2 - g_T g_C \hat{Q}}$$

$$c_5 = \frac{2 g_A^2 g_G^2}{g_R (2 g_A g_G g_R - g_R^2 \hat{P}_1 - g_A g_G \hat{Q})}$$

$$+\frac{2g_T^2 g_C^2}{g_Y(2g_T g_C g_Y - g_Y^2 \hat{P}_2 - g_T g_C \hat{Q})}$$

$$+\frac{g_R^2(g_T^2+g_C^2)+g_Y^2(g_A^2+g_G^2)}{2g_R^2 g_Y^2 - g_R g_Y Q}$$

$$\hat{P}_1 = n_s(A \leftrightarrow G), \quad \hat{P}_2 = n_s(C \leftrightarrow T), \quad \hat{Q} = \frac{n_s}{n_d}$$

$$\hat{d} = \quad -c_1 \log(1 - \frac{\hat{P}_1}{c_1} - \frac{\hat{Q}}{2g_R}) - c_2 \log(1 - \frac{\hat{P}_2}{c_2} - \frac{\hat{Q}}{2g_Y})$$

$$-2(g_R g_Y - c_1 g_Y - c_2 g_R)\log(1 - \frac{Q}{2g_R g_Y})$$

$$\mathrm{V}(\hat{d}) = \frac{c_3^2 \hat{P}_1 + c_4^2 \hat{P}_2 + c_5^2 \hat{Q} - (c_3 \hat{P}_1 + c_4 \hat{P}_2 + c_5 \hat{Q})^2}{L}$$

Gamma correction:

$$\hat{d} = \quad 2a\Big[\ c_1(1 - \frac{\hat{P}_1}{c_1} - \frac{\hat{Q}}{2g_R})^{-1/a} + c_2(1 - \frac{\hat{P}_2}{c_2} - \frac{\hat{Q}}{2g_Y})^{-1/a}$$

$$+(g_R g_Y - \frac{g_Y}{c_1} - \frac{g_R}{c_2})(1 - \frac{\hat{Q}}{2g_R g_Y})^{-1/a} - 2g_A g_G - 2g_T g_C - 2g_R g_Y\ \Big]$$

$$\mathrm{V}(\hat{d}) = \frac{c_3^2 \hat{P}_1 + c_4^2 \hat{P}_2 + c_5^2 \hat{Q} - (c_3 \hat{P}_1 + c_4 \hat{P}_2 + c_5 \hat{Q})^2}{L}$$

References:

Tamura and Nei, 1994,
Kumar et al. 1993

7.1.2.6 Estimation of genetic distances between RFLP haplotypes

7.1.2.6.1 Number of pairwise difference

We simply count the number of different alleles between two RFLP haplotypes.

$$\hat{d}_{xy} = \sum_{i=1}^{L} \delta_{xy}(i)$$

where $\delta_{xy}(i)$ is the Kronecker function, equal to 1 if the alleles of the i-th locus are identical for both haplotypes, and equal to 0 otherwise.

When estimating genetic structure indices, this choice amounts at estimating weighted F_{ST} statistics over all loci (Weir and Cockerham, 1984; Michalakis and Excoffier, 1996).

7.1.2.6.2 Proportion of difference

We simply count the proportion of loci that are different between two RFLP haplotypes.

$$\hat{d}_{xy} = \frac{1}{L} \sum_{i=1}^{L} \delta_{xy}(i)$$

where $\delta_{xy}(i)$ is the Kronecker function, equal to 1 if the alleles of the i-th locus are identical for both haplotypes, and equal to 0 otherwise.

When estimating genetic structure indices, this choice will lead to exactly the same results as the number of pairwise differences.

7.1.2.7 Estimation of distances between Microsatellite haplotypes

7.1.2.7.1 No. of different alleles

We simply count the number of different alleles between two haplotypes.

$$\hat{d}_{xy} = \sum_{i=1}^{L} \delta_{xy}(i)$$

where $\delta_{xy}(i)$ is the Kronecker function, equal to 1 if the alleles of the i-th locus are identical for both haplotypes, and equal to 0 otherwise.

When estimating genetic structure indices, this choice amounts at estimating weighted F_{ST} statistics over all loci (Weir and Cockerham, 1984; Michalakis and Excoffier, 1996).

7.1.2.7.2 Sum of squared size difference

Counts the sum of the squared number of repeat difference between two haplotypes (Slatkin, 1995).

$$\hat{d}_{xy} = \sum_{i=1}^{L} (a_{xi} - a_{yi})^2 \quad ,$$

where a_{xi} is the number of repeats of the microsatellite for the i-th locus.

When estimating genetic structure indices, this choice amounts at estimating an analog of Slatkin's R_{ST} (1995) (see Michalakis and Excoffier, 1996, as well as Rousset, 1996 , for details on the relationship between F_{ST} and R_{ST}) .

7.1.2.8 Estimation of distances between Standard haplotypes

7.1.2.8.1 Number of pairwise differences

Simply counts the number of different alleles between two haplotypes.

$$\hat{d}_{xy} = \sum_{i=1}^{L} \delta_{xy}(i)$$

where $\delta_{xy}(i)$ is the Kronecker function, equal to 1 if the alleles of the i-th locus are identical for both haplotypes, and equal to 0 otherwise.

When estimating genetic structure indices, this choice amounts at estimating weighted F_{ST} statistics over all loci (Weir and Cockerham, 1984; Michalakis and Excoffier, 1996).

7.1.2.9 Minimum Spanning Network among haplotypes

We have implemented the computation of a Minimum Spanning Tree (MST) (Kruskal, 1956; Prim, 1957) between OTU's (Operational Taxonomic Units). The MST is computed from the matrix of pairwise distances calculated between all pairs of haplotypes using a modification of the algorithm described in Rohlf (1973). The Minimum Spanning Network embedding all MSTs (see Excoffier and Smouse 1994) is also provided. This implementation is the translation of a standalone program written in Pascal called MINSPNET.EXE running under DOS, formerly available on http://anthropologie.unige.ch/LGB/software/win/min-span-net/.

7.1.3 Haplotype frequency estimation

7.1.3.1 Haplotypic data or Genotypic data with known Gametic phase

If haplotype i is observed x_i times in a sample containing n gene copies, then its estimated frequency ($\hat{p}_i$) is given by

$$\hat{p}_i = \frac{x_i}{n} \quad ,$$

whereas an unbiased estimate of its sampling variance is given by

$$\mathrm{V}(p_i) = \frac{\hat{p}_i(1-\hat{p}_i)}{n-1} \, .$$

7.1.3.2 Genotypic data with unknown Gametic phase

Maximum-likelihood haplotype frequencies are computed using an Expectation-Maximization (EM) algorithm (see e.g. Dempster et al. 1977; Excoffier and Slatkin, 1995; Lange, 1997; Weir, 1996). This procedure is an iterative process aiming at obtaining maximum-likelihood estimates of haplotype frequencies from multi-locus genotype data when the gametic phase is unknown (phenotypic data). In this case, a simple gene counting is not possible because several genotypes are possible for individuals heterozygote at more than one locus. Therefore, a slightly more elaborate procedure is needed.

The likelihood of the sample (the probability of the observed data $\mathbf{D}$, given the haplotype frequencies - $\mathbf{p}$) is given by

$$L(\mathbf{D} \mid \mathbf{p}) = \sum_{i=1}^{n} \prod_{j=1}^{g_i} G_{ij} ,$$

where the sum is over all n individuals of the sample, and the product is over all possible genotypes of those individuals, and $G_{ij} = 2p_i p_j$, if $i \neq j$ or $G_{ij} = p_i^2$, if $i = j$.

The principle of the EM algorithm is the following:

1. Start with arbitrary (random) estimates of haplotype frequencies.
2. Use these estimates to compute expected genotype frequencies for each phenotype, assuming Hardy-Weinberg equilibrium (The E-step).
3. The relative genotype frequencies are used as weights for their two constituting haplotypes in a gene counting procedure leading to new estimates of haplotype frequencies (The M-step).
4. Repeat steps 2-3, until the haplotype frequencies reach equilibrium (do not change more than a predefined epsilon value).

Dempster et al (1977) have shown that the likelihood of the sample could only grow after each step of the EM algorithm. However, there is no guarantee that the resulting haplotype frequencies are maximum likelihood estimates. They can be just local optimal values. In fact, there is no obvious way to be sure that the resulting frequencies are those that globally maximize the likelihood of the data. This would need a complete evaluation of the likelihood for all possible genotype configurations of the sample. In order to check that the final frequencies are putative maximum likelihood estimates, one has generally to repeat the EM algorithm from many different starting points (many different initial haplotype frequencies). Several runs may give different final frequencies, suggesting the presence of several "peaks" in the likelihood surface, but one has to choose the solution that has the largest likelihood. It may also arise that several distinct peaks have the same likelihood, meaning that different haplotypic compositions explain equally well the observed data. At this point, there is no way to choose among the alternative solutions from a likelihood point of view. Some external information should be provided to make a decision.

Standard deviations of the haplotype frequencies are estimated by a parametric bootstrap procedure (see e.g. Rice, 1995), generating random samples from a population assumed to have haplotype frequencies equal to their maximum-likelihood values. For each bootstrap replicate, we apply the EM algorithm to get new maximum-likelihood haplotype frequencies. The standard deviation of each haplotype frequency is then estimated from the resulting distribution of haplotype frequencies. Note however that this procedure is quite computer intensive.

7.1.4 Linkage disequilibrium between pairs of loci

Depending on whether the haplotypic composition of the sample is known or not, we have implemented two different ways to test for the presence of pairwise linkage disequilibrium between loci.
We describe in detail below how the two tests are done.

7.1.4.1 Exact test of linkage diseq uilibrium (haplotypic data)

This test is an extension of Fisher exact probability test on contingency tables (*Slatkin*, 1994a). A contingency table is first built. The k_1xk_2 entries of the table are the observed haplotype frequencies (absolute values), with k_1 and k_2 being the number of alleles at locus 1 and 2, respectively. The test consists in obtaining the probability of finding a table with the same marginal totals and which has a probability equal or less than the observed table. Under the null-hypothesis of no association between the two tested loci, the probability of the observed table is

$$L_0 = \frac{n!}{\prod_{i,j} n_{ij}} \prod_i (n_{i*}/n)^{n_{i*}} \prod_i (n_{*i}/n)^{n_{*i}} ,$$

where the n_{ij}'s denote the count of the haplotypes that have the i-th allele at the first locus and the j-th allele at the second locus, n_{i*} is the overall frequency of the i-th allele at the first locus (i=1,... k_1) and n_{*i} is the count of the i-th allele at the second locus (i=1,... k_2).
Instead of enumerating all possible contingency tables, a Markov chain is used to efficiently explore the space of all possible tables. This Markov chain consists in a random walk in the space of all contingency tables. It is done is such a way that the probability to visit a particular table corresponds to its actual probability under the null hypothesis of linkage equilibrium. A particular table is modified according to the following rules (see also Guo and Thompson, 1992; or Raymond and Rousset, 1995) :

1. We select in the table two distinct lines i_1, i_2 and two distinct columns j_1, j_2 at random.
2. The new table is obtained by decreasing the counts of the cells (i_1, j_1) (i_2, j_2) and increasing the counts of the cells (i_1, j_2) (i_2, j_1) by one unit. This leaves the marginal allele counts n_i unchanged.
3. The switch to the new table is accepted with a probability equal to

$$R = \frac{L_1}{L_0} = \frac{(n_{i_1,j_2}+1)(n_{i_2,j_1}+1)}{n_{i_1,j_1} n_{i_2,j_2}},$$

where R is just the ratio of the probabilities of the two tables.

The steps 1-3 are done a large number of times to explore a large amount of the space of all possible contingency tables having identical marginal counts. In order to start from a random initial position in the Markov chain, the chain is explored for a pre-defined number of steps (the dememorization phase) before the probabilities of the switched tables are compared to that of the initial table. The number of dememorization steps should be enough (some thousands) such as to allow the Markov chain to "forget" its initial state, and make it independent from its starting point. The *P*-value of the test is then taken as the proportion of the visited tables having a probability smaller or equal to the observed contingency table.

A standard error on *P* is estimated by subdividing the total amount of required steps into B batches (see Guo and Thompson, 1992, p. 367). A *P*-value is calculated separately for each batch. Let us denote it by P_i (i=1,...,B). The estimated standard error is then calculated as

$$s.d.(P) = \sqrt{\frac{\sum_{i=1}^{B}(P - P_i)^2}{B(B-1)}}.$$

The process is stopped as soon as the estimated standard deviation is smaller than a pre-defined value specified by the user.

7.1.4.2 Likelihood ratio test of linkage disequilibrium (genotypic data, gametic phase unknown)

For genotypic data where the haplotypic phase is unknown, the test based on the Markov chain described above is not possible because the haplotypic composition of the sample is unknown, and is just estimated. Therefore, linkage disequilibrium between a pair of loci is tested for genotypic data using a likelihood-ratio test, whose empirical distribution is obtained by a permutation procedure (Slatkin and Excoffier, 1996). The likelihood of the data assuming linkage equilibrium (L_{H*}) is computed by using the fact that, under this hypothesis, the haplotype frequencies are obtained as the product of the allele frequencies. The likelihood of the data *not* assuming linkage equilibrium (L_H) is obtained by applying the EM algorithm to estimate haplotype frequencies. The likelihood-ratio statistic given by

$$S = -2\log(\frac{L_{H^*}}{L_H})$$

should in principle follow a Chi-square distribution, with (k_1-1) (k_2-1) degrees of freedom, but it is not always the case in small samples with large number of alleles per locus. In order to better approximate the underlying distribution of the likelihood-ratio statistic under the null hypothesis of linkage equilibrium, we use the following permutation procedure:

1. Permute the alleles between individuals at one locus only.
2. Re-estimate the likelihood of the data L_H' by the EM algorithm. Note that L_{H*} is unaffected by the permutation procedure.
3. Repeat steps 1-2 a large number of times to get the null distribution of L_H, and therefore the null distribution of S.

Note that this test of linkage disequilibrium assumes Hardy-Weinberg proportions of genotypes, and the rejection of the test could be also due to departure from Hardy-Weinberg equilibrium (see Excoffier and Slatkin, 1998)

7.1.4.3 Measures of gametic diseq uilibrium (haplotypic data)

- ***D* and *D'* coefficients**:

1. D: The classical linkage disequilibrium coefficient measuring deviation from random association between alleles at different loci (Lewontin and Kojima, 1960) is expressed as

$$D_{ij} = p_{ij} - p_i p_j ,$$

where p_{ij} is the frequency of the haplotype having allele i at the first locus and allele j at the second locus, and p_i and p_j are the frequencies of alleles i and j, respectively.

2. D'_{ij}: The linkage disequilibrium coefficient D_{ij} standardized by the maximum value it can take ($D_{ij,\max}$), given the allele frequencies (Lewontin 1964), as

$$D'_{ij} = \frac{D_{ij}}{D_{ij,\max}} ,$$

where $D_{ij,\max}$ takes one of the following values:

$$\min(p_i p_j , (1-p_i)(1-p_j)) \quad \text{if} \quad D_{ij} < 0$$

$$\min((1-p_i)p_j , p_i(1-p_j)) \quad \text{if} \quad D_{ij} > 0$$

7.1.5 Hardy-Weinberg equilib rium.

To detect significant departure from Hardy-Weinberg equilibrium, we follow the procedure described in Guo and Thompson (1992) using a test analogous to Fisher's exact test on a two-by-two contingency table, but extended to a triangular contingency table of arbitrary size. The test is done using a modified version of the Markov-chain random walk algorithm described Guo and Thomson (1992). The modified version gives the same results than the original one, but is more efficient from a computational point of view.

This test is obviously only possible for genotypic data. If the gametic phase is unknown, the test is only possible for each locus separately. For data with known gametic phase, it is also possible to test for the non random association of haplotypes into individuals. Note that this test assumes that the allele frequencies are given. Therefore, this test is not possible for data with recessive alleles, as in this case the allele frequencies need to be estimated.

A contingency table is first built. The kxk entries of the table are the observed allele frequencies and k is the number of alleles. Using the same notations as in section 8.2.2, the probability to observe the table under the null-hypothesis of no association is given by Levene (1949)

$$L_0 = \frac{n! \prod_{i=1}^{k} n_{i*}!}{(2n)! \prod_{i=1}^{k} \prod_{j=1}^{i} n_{ij}!} 2^H ,$$

where H is the number of heterozygote individuals.

Much like it was done for the test of linkage disequilibrium, we explore alternative contingency tables having same marginal counts. In order to create a new contingency table from an existing one, we select two distinct lines i_1, i_2 and two distinct columns j_1, j_2 at random. The new table is obtained by decreasing the counts of the cells (i_1, j_1) (i_2, j_2) and increasing the counts of the cells (i_1, j_2) (i_2, j_1) by one unit. This leaves the alleles counts n_i unchanged. The switch to the new table is accepted with a probability R equal to :

1. $$R = \frac{L_{n+1}}{L_n} = \frac{n_{i_1 j_1} n_{i_2 j_2}}{(n_{i_1 j_2} + 1)(n_{i_2 j_1} + 1)} \frac{(1 + \delta_{i_1 j_1})(1 + \delta_{i_2 j_2})}{(1 + \delta_{i_1 j_2})(1 + \delta_{i_2 j_1})}, \text{ if } i_1 \neq j_1 \text{ or } i_2 \neq j_2$$

2. $$R = \frac{L_{n+1}}{L_n} = \frac{n_{i_1 j_1} n_{i2 j2}}{(n_{i_1 j_2} + 1)(n_{i_2 j_1} + 2)} \frac{4}{1}, \text{ if } i_1 = j_1 \text{ and } i_2 = j_2$$

3. $$R = \frac{L_{n+1}}{L_n} = \frac{n_{i_1 j_1} (n_{i_2 j_2} - 1)}{(n_{i_1 j_2} + 1)(n_{i_2 j_1} + 1)} \frac{1}{4}, \text{ if } i_1 = j_2 \text{ and } i_2 = j_1$$

.

As usual δ denotes the Kronecker function. R is just the ratio of the probabilities of the two tables. The switch to the new table is accepted if R is larger than 1.

The P-value of the test is the proportion of the visited tables having a probability smaller or equal to the observed (initial) contingency table. The standard error on the P-value is estimated like in the case of linkage disequilibrium using a system of batches (see section 7.1.4.1).

7.1.6 Neutrality tests.

7.1.6.1 Ewens-Watterson homozygosity test

This test is based on *Ewens* (1972) sampling theory of neutral alleles. Watterson (1978) has shown that the distribution of selectively neutral haplotype frequencies could be conveniently summarized by the sum of haplotype (allele) frequencies (F), equivalent to the expected homozygosity for diploids. This test can be performed equally well on diploid or haploid data, as the test statistic is not used for its biological meaning, but just as a way to

summarize the allelic frequency distribution. The null distribution of F is generated by simulating random neutral samples having the same number of genes and the same number of haplotypes using the algorithm of Stewart (1977). The probability of observing random samples with F values identical or smaller than the original sample is recorded. This tests is currently limited to sample sizes of 2000 genes or less and 1000 different alleles (haplotypes) or less. It can be used to test the hypothesis of selective neutrality and population equilibrium against either balancing selection or the presence of advantageous alleles.

7.1.6.2 Ewens-Watterson-Slatkin exact test

This test is essentially similar to that of *Watterson* (1978) test, but instead of using F as a summary statistic, it compares the probabilities of the random samples to that of the observed sample (Slatkin 1994b, 1996). The probability of obtaining a random sample having a probability smaller or equal to the observed sample is recorded. The results are in general very close to those of Watterson's homozygosity test. Note that the random samples are generated as explained for the Ewens-Watterson homozygosity test.

7.1.6.3 Chakraborty's test of population amalgamation

This test is also based on the infinite-allele model, and on Ewens (1972) sampling theory of neutral alleles. By simulation, Chakraborty (1990) has noticed that the number of alleles in a heterogeneous sample (drawn from a population resulting from the amalgamation of previously isolated populations) was larger than the number of alleles expected in a homogeneous neutral sample. He also noticed that the homozygosity of the sample was less sensitive to the amalgamation and therefore proposed to use the mutation parameter inferred from the homozygosity (θ_{Hom}) (see section 7.1.2.3.1) to compute the probability of observing a random neutral sample with a number of alleles similar or larger than the observed value ($\Pr(K \geq k_{obs})$ (see section 7.1.2.3.3 to see how this probability can be computed). It is an approximation of the conditional probability of observing some number of alleles given the observed homozygosity.

7.1.6.4 Tajima's test of selective neutrality

Tajima's (1989a) test is based on the infinite-site model without recombination, appropriate for short DNA sequences or RFLP haplotypes. It compares two estimators of the mutation parameter theta ($\theta = 2Mu$, with M=2N in diploid populations or M=N in haploid populations of effective size N). The test statistic D is then defined as

$$D = \frac{\hat{\theta}_\pi - \hat{\theta}_S}{\sqrt{Var(\hat{\theta}_\pi - \hat{\theta}_S)}},$$

where $\hat{\theta}_\pi = \hat{\pi}$ and $\hat{\theta}_S = S / \sum_{i=0}^{n-1} (1/i)$, and S is the number of segregating sites in the sample. The limits of confidence intervals around D may be found in Table 2 of Tajima's paper (Tajima 1989a) for different sample sizes. The significance of the *D* statistic is tested by generating random samples under the hypothesis of selective neutrality and population equilibrium, using a coalescent simulation algorithm adapted from Hudson (1990). The P

value of the *D* statistic is then obtained as the proportion of random F_S statistics less or equal to the observation. We also provide a parametric approximation of the *P*-value assuming a beta-distribution limited by minimum and maximum possible D values (see Tajima 1989a, p.589). Note that significant *D* values can be due to factors other than selective effects, like population expansion, bottleneck, or heterogeneity of mutation rates (see Tajima, 1993; Aris-Brosou and Excoffier, 1996; or Tajima 1996, for further details).

7.1.6.5 Fu's F_S test of selective neutrality

Like *Tajima's* (1989a) test, Fu's test (Fu, 1997) is based on the infinite-site model without recombination, and thus appropriate for short DNA sequences or RFLP haplotypes. The principle of the test is very similar to that of Chakraborty described above. Here, we evaluate the probability of observing a random neutral sample with a number of alleles similar or smaller than the observed value (see section 7.1.2.3.3 to see how this probability can be computed) given the observed number of pairwise differences, taken as an estimator of θ. In more details, Fu first calls this probability $S' = \Pr(K \geq k_{obs} \mid \theta = \hat{\theta}_\pi)$ and defines the F_S statistic as the logit of S'

$$F_S = \ln\left(\frac{S'}{1-S'}\right) \qquad \text{(Fu, 1997)}$$

Fu (1997) has noticed that the F_S statistic was very sensitive to population demographic expansion, which generally lead to large negative F_S values.

The significance of the F_S statistic is tested by generating random samples under the hypothesis of selective neutrality and population equilibrium, using a coalescent simulation algorithm adapted from Hudson (1990). The P-value of the F_S statistic is then obtained as the proportion of random F_S statistics less or equal to the observation. Using simulations, Fu noticed that the 2% percentile of the distribution corresponded to the 5% cutoff value (i.e. the critical value of the test at the 5% significance level). We indeed confirmed this behavior by our own simulations. Even though this property is not fully understood, it means that a F_S statistic should be considered as significant at the 5% level, if its P-value is below 0.02, and not below 0.05.

7.1.7 Population genetic structure inferred by analysis of variance (AMOVA)

The genetic structure of population is investigated here by an analysis of variance framework, as initially defined by Cockerham (1969, 1973), and extended by others (see e.g. Weir and Cockerham, 1984; Long 1986). The Analysis of Molecular Variance approach used in Arlequin (AMOVA, Excoffier et al. 1992) is essentially similar to other approaches based on analyses of variance of gene frequencies, but it takes into account the number of mutations between molecular haplotypes (which first need to be evaluated).

By defining groups of populations, the user defines a particular genetic structure that will be tested (see the input file notations for more details). A hierarchical analysis of variance partitions the total variance into covariance components due to intra-individual differences, inter-individual differences, and/or inter-population differences. See also Weir (1996), for detailed treatments of hierarchical analyses, and Excoffier (2000) as well as Rousset (2000) for an explanation why these are *covariance* components rather than *variance* components. The covariance components

(σ_i^2's) are used to compute fixation indices, as originally defined by Wright (1951, 1965), in terms of inbreeding coefficients, or later in terms of coalescent times by Slatkin (1991).

Formally, in the haploid case, we assume that the *i*-th haplotype frequency vector from the *j*-th population in the *k*-th group is a linear equation of the form

$$\mathbf{x}_{ijk} = \mathbf{x} + \mathbf{a}_k + \mathbf{b}_{jk} + \mathbf{c}_{ijk}.$$

The vector **x** is the unknown expectation of $\mathbf{x}_{ijk}$, averaged over the whole study. The effects are **a** for group, **b** for population, and **c** for haplotypes within a population within a group, assumed to be additive, random, independent, and to have the associated covariance components σ_a^2, σ_b^2, and σ_c^2, respectively. The total molecular variance (σ^2) is the sum of the covariance component due to differences among haplotypes within a population (σ_c^2), the covariance component due to differences among haplotypes in different populations within a group (σ_b^2), and the covariance component due to differences among the G populations (σ_a^2). The same framework could be extended to additional hierarchical levels, such as to accommodate, for instance, the covariance component due to differences between haplotypes within diploid individuals.

Note that in the case of a simple hierarchical genetic structure consisting of haploid individuals in populations, the implemented form of the algorithm leads to a fixation index F_{ST} which is absolutely identical to the weighted average *F*-statistic over loci, $\hat{\theta}_w$, defined by Weir and Cockerham (1984) (see Michalakis and Excoffier 1996 for a formal proof). In terms of inbreeding coefficients and coalescence times, this F_{ST} can be expressed as

$$F_{ST} = \frac{f_0 - f_1}{1 - f_1} = \frac{\bar{t}_1 - \bar{t}_0}{\bar{t}_1}, \qquad \text{(Slatkin, 1991)}$$

where f_0 is the probability of identity by descent of two different genes drawn from the same population, f_1 is the probability of identity by descent of two genes drawn from two different populations, $\bar{t}_1$ is the mean coalescence times of two genes drawn from two different populations, and $\bar{t}_0$ is the mean coalescence time of two genes drawn from the same population.

The significance of the fixation indices is tested using a non-parametric permutation approach described in Excoffier et al. (1992), consisting in permuting haplotypes, individuals, or populations, among individuals, populations, or groups of populations. After each permutation round, we recompute all statistics to get their null distribution. Depending on the tested statistic and the given hierarchical design, different types of permutations are performed. Under this procedure, the normality assumption usual in analysis of variance tests is no longer necessary, nor is it necessary to assume equality of variance among populations or groups of populations. A large number of permutations (1,000 or more) is necessary to obtain some accuracy on the final probability. A system of batches

similar to those used in the exact test of linkage disequilibrium (see end of section 7.1.4.1) has been implemented to get an idea of the standard-deviation of the P values.

We have implemented here 6 different types of hierarchical AMOVA. The number of hierarchical levels varies from two to four. In each of the situations, we describe the way the total sum of squares is partitioned, how the covariance components and the associated *F*-statistics are obtained, and which permutation schemes are used for the significance test.

Before enumerating all the possible situations, we introduce some notations:

SSD(*T*)	:	Total sum of squared deviations.
SSD (*AG*)	:	Sum of squared deviations Among Groups of populations.
SSD (*AP*)	:	Sum of squared deviations Among Populations.
SSD (*AI*)	:	Sum of squared deviations Among Individuals.
SSD (*WP*)	:	Sum of squared deviations Within Populations.
SSD (*WI*)	:	Sum of squared deviations Within Individuals.
SSD (*AP*/*WG*)	:	Sum of squared deviations Among Populations, Within Groups.
SSD (*AI*/*WP*)	:	Sum of squared deviations Among Individuals, Within Populations.
G	:	Number of groups in the structure.
P	:	Total number of populations.
N	:	Total number of individuals for genotypic data or total number of gene copies for haplotypic data.
N_p	:	Number of individuals in population p for genotypic data or total number of gene copies in population p for haplotypic data.
N_g	:	Number of individuals in group g for genotypic data or total number of gene copies in group g for haplotypic data..

7.1.7.1 Haplotypic data, one group of populations

Source of variation	Degrees of freedom	Sum of squares (SSD)	Expected mean squares
Among Populations	*P* - 1	SSD(*AP*)	$n\sigma_a^2 + \sigma_b^2$
Within Populations	*N* - *P*	SSD(*WP*)	σ_b^2
Total	*N* - 1	SSD(*T*)	σ_T^2

Where n and F_{ST} are defined by

$$n = \frac{N - \sum_p \frac{N_p^2}{N}}{P-1},$$

$$F_{ST} = \frac{\sigma_a^2}{\sigma_T^2}.$$

- We test σ_a^2 and F_{ST} by permuting haplotypes among populations.

7.1.7.2 Haplotypic data, several groups of populations

Source of variation	Degrees of freedom	Sum of squares (*SSD*)	Expected mean squares
Among Groups	G - 1	SSD(*AG*)	$n''\sigma_a^2 + n'\sigma_b^2 + \sigma_c^2$
Among Populations / Within Groups	P - G	SSD(*AP*/*WG*)	$n\sigma_b^2 + \sigma_c^2$
Within Populations	N - P	SSD(*WP*)	σ_c^2
Total:	N - 1	SSD(*T*)	σ_T^2

Where the n's and the F-statistics are defined by:

$$S_G = \sum_{g \in G} \sum_{p \in g} \frac{N_p^2}{N_g}, \quad n = \frac{N - S_G}{P - G},$$

$$n' = \frac{S_G - \sum_{p \in P} \frac{N_p^2}{N}}{G - 1}, \quad n'' = \frac{N - \sum_{g \in G} \frac{N_g^2}{N}}{G - 1}$$

$$F_{CT} = \frac{\sigma_a^2}{\sigma_T^2}, \quad F_{SC} = \frac{\sigma_b^2}{\sigma_b^2 + \sigma_c^2} \quad \text{and} \quad F_{ST} = \frac{\sigma_a^2 + \sigma_b^2}{\sigma_T^2}$$

- We test σ_c^2 and F_{ST} by permuting haplotypes among populations among groups.
- We test σ_b^2 and F_{SC} by permuting haplotypes among populations within groups.
- We test σ_a^2 and F_{CT} by permuting populations among groups.

7.1.7.3 Genotypic data, one group of populations, no within- individual level

Source of variation	Degrees of freedom	Sum of squares (SSD)	Expected mean squares
Among Populations	P - 1	SSD(AP)	$n\sigma_a^2 + \sigma_b^2$
Within Populations	$2N$ - P	SSD(WP)	σ_b^2
Total:	$2N$ - 1	SSD(T)	σ_T^2

Where n and F_{ST} are defined by

$$n = \frac{2N - \sum_P \frac{2N_p^2}{N}}{P-1},$$

$$F_{ST} = \frac{\sigma_a^2}{\sigma_T^2}.$$

If the gametic phase is know:

- We test σ_a^2 and F_{ST} by permuting haplotypes among populations.

If the gametic phase is unknown:

- We test σ_a^2 and F_{ST} by permuting individual genotypes among populations.

7.1.7.4 Genotypic data, several groups of populations, no within- individual level

Source of Variation	Degrees of freedom	Sum of squares (SSD)	Expected mean squares
Among Groups	G - 1	SSD(AG)	$n''\sigma_a^2 + n'\sigma_b^2 + \sigma_c^2$
Among Populations / Within Groups	P - G	SSD(AP/WG)	$n\sigma_b^2 + \sigma_c^2$
Within Populations	$2N$ - P	SSD(WP)	σ_c^2
Total:	$2N$ - 1	SSD(T)	σ_T^2

Where the n's and the F-statistics are defined by:

$$S_G = \sum_{g \in G} \sum_{p \in g} \frac{2N_p^2}{N_g}, \quad n = \frac{2N - S_G}{P - G},$$

$$n' = \frac{S_G - \sum_{p \in P} \frac{2N_p^2}{N}}{G-1}, \quad n'' = \frac{2N - \sum_{g \in G} \frac{2N_g^2}{N}}{G-1},$$

$$F_{CT} = \frac{\sigma_a^2}{\sigma_T^2}, \quad F_{ST} = \frac{\sigma_a^2 + \sigma_b^2}{\sigma_T^2} \quad \text{and} \quad F_{SC} = \frac{\sigma_b^2}{\sigma_b^2 + \sigma_c^2}.$$

If the gametic phase is known:

- We test σ_c^2 and F_{ST} by permuting haplotypes among populations and among groups.
- We test σ_b^2 and F_{SC} by permuting haplotypes among populations but within groups.

If the gametic phase is not known:

- We test σ_c^2 and F_{ST} by permuting individual genotypes among populations and among groups.
- We test σ_b^2 and F_{SC} by permuting individual genotypes among populations but within groups.

In all cases:

- We test σ_a^2 and F_{CT} by permuting whole populations among groups.

7.1.7.5 Genotypic data, one population, within- individual level

Source of variation	Degrees of freedom	Sum of squares (SSD)	Expected mean squares
Among Individuals	N - 1	SSD(*AI*)	$2\sigma_a^2 + \sigma_b^2$
Within Individuals	N	SSD(*WI*)	σ_b^2
Total:	2N - 1	SSD(*T*)	σ_T^2

Where F_{IS} is defined as:

$$F_{IS} = \frac{\sigma_a^2}{\sigma_T^2}.$$

- We test σ_a^2 and F_{IS} by permuting haplotypes among individuals.

7.1.7.6 Genotypic data, one group of populations, within- individual level

Source of Variation	Degrees of freedom	Sum of squares (SSD)	Expected mean squares
Among Populations	P - 1	SSD(AP)	$n\sigma_a^2 + 2\sigma_b^2 + \sigma_c^2$
Among Individuals / Within Populations	$N - P$	SSD(AI/WP)	$2\sigma_b^2 + \sigma_c^2$
Within Individuals	N	SSD(WI)	σ_c^2
Total	$2N - 1$	SSD(T)	σ_T^2

Where n and the F-statistics are defined by:

$$n = \frac{2N - \sum_{p \in P} \frac{2N_p^2}{N}}{P - 1}$$

$$F_{ST} = \frac{\sigma_a^2}{\sigma_T^2}, \quad F_{IT} = \frac{\sigma_a^2 + \sigma_b^2}{\sigma_T^2} \quad \text{and} \quad F_{IS} = \frac{\sigma_b^2}{\sigma_b^2 + \sigma_c^2}.$$

- We test σ_c^2 and F_{IT} by permuting haplotypes among individuals among populations.
- We test σ_b^2 and F_{IS} by permuting haplotypes among individuals within populations.
- We test σ_a^2 and F_{ST} by permuting individual genotypes among populations.

7.1.7.7 Genotypic data, several groups of populations, within- individual level

Source of Variation:	Degrees of freedom	Sum of squares (SSD)	Expected mean squares
Among Groups	G - 1	SSD(AG)	$n''\sigma_a^2 + n'\sigma_b^2 + 2\sigma_c^2 + \sigma_d^2$
Among Populations / Within Groups	$P - G$	SSD(AP/WG)	$n\sigma_b^2 + 2\sigma_c^2 + \sigma_d^2$
Among Individuals / Within Populations	N - P	SSD(AI/WP)	$2\sigma_c^2 + \sigma_d^2$
Within Individuals	N	SSD(WI)	σ_d^2
Total:	$2N - 1$	SSD(T)	σ_T^2

Where the n's and the F-statistics are defined by:

$$n = \frac{2N - \sum_{g \in G} \sum_{p \in g} \frac{2N_p^2}{N_g}}{P - G} \quad , \quad n' = \frac{\sum_{g \in G} \frac{(N - N_g)}{N_g} \sum_{p \in g} 2N_p^2}{N(G-1)} \quad , \quad n'' = \frac{2N - \frac{\sum_{g \in G} 2N_g^2}{N}}{G-1}$$

$$F_{CT} = \frac{\sigma_a^2}{\sigma_T^2}, \quad F_{IT} = \frac{\sigma_a^2 + \sigma_b^2 + \sigma_c^2}{\sigma_T^2} \quad , \quad F_{IS} = \frac{\sigma_c^2}{\sigma_c^2 + \sigma_d^2} \quad \text{and} \quad F_{SC} = \frac{\sigma_b^2}{\sigma_b^2 + \sigma_c^2 + \sigma_d^2}.$$

- We test σ_d^2 and F_{IT} by permuting haplotypes among populations and among groups.
- We test σ_c^2 and F_{IS} by permuting haplotypes among individuals within populations.
- We test σ_b^2 and F_{SC} by permuting individual genotypes among populations but within groups.
- We test σ_a^2 and F_{CT} by permuting populations among groups.

7.1.8 Minimum Spanning Network (MSN) among haplotypes

It is possible to compute the Minimum Spanning Tree (MST) and Minimum Spanning Network (MSN) from the squared distance matrix among haplotypes used for the calculation of F-statistics in the AMOVA procedure. See section 7.1.8 for a brief description of the method and references.

7.1.9 Locus-by-locus AMOVA

AMOVA analyses can now be performed for each locus separately in the same way it was performed at the haplotype level. Variance components and F-statistics are estimated for each locus separately and listed into a global table. The different variance components from different levels are combined to produce synthetic estimators of F-statistics, by summing variance components estimated at a given level in the hierarchy in the numerator and denominator to produce F-statistics as variance component ratios. Therefore the global F-statistics are not obtained as an arithmetic average of each locus F-statistics (see e.g. Weir and Cockerham 1984, or Weir 1996).

If there is no missing data, the locus-by-locus and the haplotype analyses should lead to identical sums of squares, variance components, and F-statistics. If there are missing data, the global variance components should be different, because the degrees of freedom will vary from locus to locus, and therefore the estimators of F-statistics will also vary.

7.1.10 Population pairwise genetic distances

The pairwise F_{ST} s can be used as short-term genetic distances between populations, with the application of a slight transformation to linearize the distance with population divergence time (Reynolds et al. 1983; Slatkin, 1995).

The pairwise F_{ST} values are given in the form of a matrix.

The null distribution of pairwise F_{ST} values under the hypothesis of no difference between the populations is obtained by permuting haplotypes between populations. The *P*-value of the test is the proportion of permutations leading to a F_{ST} value larger or equal to the observed one. The *P*-values are also given in matrix form.

Three other matrices are computed from the F_{ST} values:

7.1.10.1 Reynolds' distance (Reyno1ds et al. 1983):

Since F_{ST} between pairs of stationary haploid populations of size N having diverged *t* generations ago varies approximately as

$$F_{ST} = 1-(1-\frac{1}{N})^t \approx 1-e^{-t/N}$$

The genetic distance $D = -\log(1-F_{ST})$ is thus approximately proportional to *t/N* for short divergence times.

7.1.10.2 Slatkin's linearized F_{ST}*'s (Slatkin 1995):*

Slatkin considers a simple demographic model where two haploid populations of size N have diverged τ generations ago from a population of identical size. These two populations have remained isolated ever since, without exchanging any migrants. Under such conditions, F_{ST} can be expressed in terms of the coalescence times $\bar{t}_1$, which is the mean coalescence time of two genes drawn from two different populations, and $\bar{t}_0$ which is the mean coalescence time of two genes drawn from the same population. Using the analysis of variance approach, the F_{ST}'s are expressed as

$$F_{ST} = \frac{\bar{t}_1 - \bar{t}_0}{\bar{t}_1}$$ (Slatkin, 1991, 1995)

Because, $\bar{t}_0$ is equal to N generations (see e.g. Hudson, 1990), and $\bar{t}_1$ is equal to τ + N generations, the above expression reduces to

$$F_{ST} = \frac{\tau}{\tau + N}.$$

Therefore, the ratio $D = F_{ST}/(1-F_{ST})$ is equal to τ / N, and is therefore proportional to the divergence time between the two populations.

7.1.10.3 M values (M = Nm for haploid populations, M = 2Nm for diploid populations).

This matrix is computed under very different assumptions than the two previous matrices. Assume that two populations of size N drawn from a large pool of populations exchange a fraction m of migrants each generation, and that the mutation rate u is negligible as compared to the migration rate m. In this case, we have the following simple relationship at equilibrium between migration and drift,

$$F_{ST} = \frac{1}{2M+1}$$

Therefore, M, which is the absolute number of migrants exchanged between the two populations, can be estimated by

$$M = \frac{1 - F_{ST}}{2F_{ST}}.$$

If one was to consider that the two populations only exchange with each other and with no other populations, then one should divide the quantity M by a factor 2 to obtain an estimator $M' = Nm$ for haploid populations, or $M' = 2Nm$ for diploid populations. This is because the expectation of F_{ST} is indeed given by

$$F_{ST} = \frac{1}{\frac{4Nmd}{(d-1)} + 1}$$ (e.g. Slatkin 1991)

where d is the number of demes exchanging genes. When d is large this tends towards the classical value $1/(4Nm+1)$, but when d=2, then the expectation of F_{ST} is $1/(8Nm+1)$.

7.1.10.4 Nei's average number of differences between populations

As additional genetic distance between populations, we also provide Nei's raw (D) and net (D_A) number of nucleotide differences between population (Nei and Li, 1979). D and net D_A are respectively computed between populations 1 and 2 as

$$D = \hat{\pi}_{12} = \sum_{i=1}^{k} \sum_{j=1}^{k'} x_{1i} x_{2j} \delta_{ij}, \text{ and}$$

$$D_A = \hat{\pi}_{12} - \frac{\hat{\pi}_1 + \hat{\pi}_2}{2},$$

where k and k' are the number of distinct haplotypes in populations 1 and 2 respectively, x_{1i} is the frequency of the i-th haplotype in population 1, and δ_{ij} is the number of differences between haplotype i and haplotype j.

Under the same notation concerning coalescence times as described above, the expectation of D_A is

$$D_A = 2\mu(\bar{t}_1 - \bar{t}_0) = 2\mu\tau + \theta,$$

where μ is the average mutation rate per nucleotide, τ is the divergence time between the two populations, and θ is either $2N\mu$ for haploid populations or $4N\mu$ for diploid populations. Thus D_A is also expected to increase linearly with divergence times between the populations.

7.1.10.5 Relative population sizes - Divergence between populations of unequal sizes

We have implemented a method to estimate divergence time between populations of unequal sizes (Gaggiotti and Excoffier 2000). The model assumes that two populations have diverged from an ancestral population of size N_0 some T generations in the past, and have remained isolated from each other ever since. The sizes of the two daughter populations can be different, but their sum adds up to the size of the ancestral population.

From the average number of pairwise differences between and within populations, we try to estimate the divergence time scaled by the mutation rate ($\tau = 2Tu$), the size of the ancestral population size scaled by the mutation rate

($\theta_0 = 2N_0u$ for haploid populations and $\theta_0 = 4N_0u$ for diploid populations), as well as the relative sizes (*k* and [1-*k*]) of the two daughter populations.

The estimated parameters result from the numerical resolution of a system of three non-linear equations with three unknowns, based on the Broyden method (Press et al. 1992, p.389).

The significance of the parameters is tested by a permutation procedure similar tot that used in AMOVA. Under the hypothesis that the two populations are undifferentiated, we permute individuals between samples, and re-estimate the three parameters, in order to obtain their empirical null distribution. The percentile value of the three statistics is obtained by the proportion of permuted cases that produce statistics larger or equal to those observed. It thus provides a percentile value of the three statistics under the null hypothesis of no differentiation.

The values of the estimated parameters should be *interpreted with caution*. The procedure we have implemented is based on the comparison of intra and inter-population diversities (π's) which have a large variance, which means that for short divergence times, the average diversity found within population could be larger than that observed between populations. This situation could lead to negative divergence times and to daughter population relative size larger than one or smaller than zero (negative values). Also large departures from the assumed pure-fission model could also lead to observed diversities that would lead to aberrant estimators of divergence time and relative population sizes. One should thus make those computations if the assumptions of a pure fission model are met and if the divergence time is relatively old. Simulation results have shown that this procedure leads to better results than other methods that do not take unequal population sizes into account when the relative sizes of daughter populations are indeed unequal. According to our simulations (Table 4 in Gaggiotti and Excoffier 2000) conventional methods such as described above lead to better results for equal population size (k=0.5) and short divergence times (T/N_0<0.5). However, the fact that the present method leads to clearly aberrant results in some cases is not necessarily a drawback. It has the advantage to draw the user attention to the fact that some care has to be taken with the interpretations of the results. Some other estimators that would be grossly biased but whose values would be kept within reasonable bounds would often lead to misinterpretations.

Note that the numerical method we have used to resolve the system of equation may sometimes fail to converge. An asterisk will indicate those cases in the result file that should be discarded because of convergence failure.

7.1.11 Exact tests of population differentiation

We test the hypothesis of a random distribution of k different haplotypes or genotypes among r populations as described in Raymond and Rousset (1995). This test is analogous to Fisher's exact test on a 2x2 contingency table extended to a $r \times k$ contingency table. All potential states of the contingency table are explored with a Markov chain similar to that described for the case of the linkage disequilibrium test (section 7.1.4.1). During this random walk between the states of the Markov chain, we estimate the probability of observing a table less or equally likely than the observed sample configuration under the null hypothesis of panmixia.

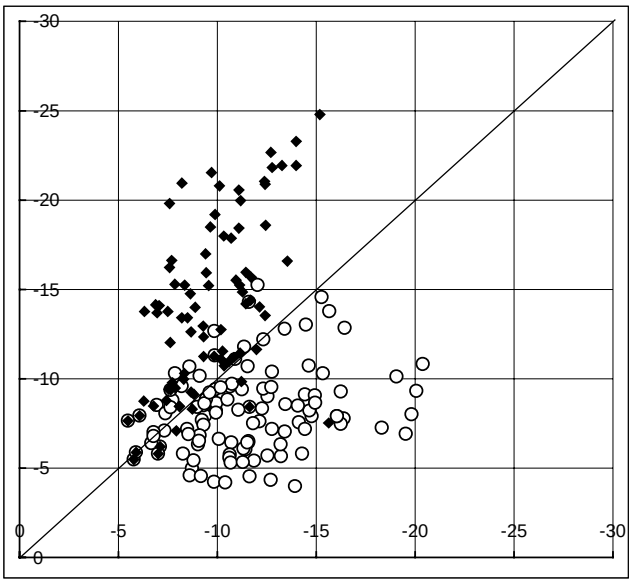
For haplotypic data, the table is built using sample haplotype frequencies (Raymond and Rousset 1995).

For genotypic data with unknown gametic phase, the contingency table is built from sample genotype frequencies (Goudet et al. 1996).

As it was done previously, an estimation of the error on the *P*-value is done by partitioning the total number of steps into a given number of batches (see section 7.1.4.1).

7.1.12 Assignment of individual genotypes to populations

It can be of interest to try to determine the origin of particular individuals, knowing a list of potential source populations (e.g. Rannala and Montain, 1997; Waser and Strobeck, 1998; Davies et al. 1999). The method we have implemented here is the most simplest one, as it consists in determining the log-likelihood of each individual multi-locus genotype in each population sample, assuming that the individual comes from that population. For computing the likelihood, we simply use the allele frequencies estimated in each sample from the original constitution of the samples. We also assume that all loci are independent, such that the global individual likelihood is obtained as the product of the likelihood at each locus. The method we have implemented is inspired from that described in Paetkau et al. (1995, 1997) and Waser and Strobeck (1998). The resulting output tables can be used to represent log-log plots of genotypes for pairs of populations likelihood (see Paetkau et al. 1997 and Waser and Strobeck 1998), to identify those genotypes that seem better explained by belonging to another population from that they were sampled.



For instance we have plotted on this graph the log-likelihood of individuals sampled in Algeria (white circles) for two HLA class II loci versus those of Senegalese Mandenka individuals (black diamonds). The overlap of the two distribution suggests that two loci are not enough to provide a clear cut separation between these two populations. One also sees that there is at least one Mandenka individual whose genotype would be much better explained if it came from the Algerian population than if it came from Eastern Senegal. Note that interpreting these results in terms of gene flow is difficult and hazardous.

7.1.13 Mantel test

The Mantel test consists in testing the significance of the correlation between two or more matrices by a permutation procedure allowing getting the empirical null distribution of the correlation coefficient taking into account the auto-correlations of the elements of the matrix. In more details, the testing procedure proceeds as follows:

Let's first define two square matrices $\mathbf{X} = \{x_{ij}\}$ and $\mathbf{Y}=\{y_{ij}\}$ of dimension N. The N^2 elements of these matrix are not all independent as there are only N-1 independent contrasts in the data. This is why the permutation procedure does not permute the elements of the matrices independently. The correlation of the two matrices is classically defined as

$$r_{XY} = \frac{SP(\mathbf{X},\mathbf{Y})}{\sqrt{SS(\mathbf{X}) \,.\, SS(\mathbf{Y})}},$$

the ratio of the cross product of **X** and **Y** over the square root of the product of sums of squares. We note that the denominator of the above equation is insensitive to permutation, such that only the numerator will change upon

permutation of rows and columns. Upon closer examination, it can be shown that the only quantity that will actually change between permutations is the Hadamard product of the two matrices noted as

$$Z_{XY} = \mathbf{X} * \mathbf{Y} = \sum_{i=1}^{N} \sum_{j=1}^{i} x_{ij}\, y_{ij}$$

which is the only variable term involved in the computation of the cross-product.

The Mantel testing procedure applied to two matrices will then consist in computing the quantity Z_{XY} from the original matrices, permute the rows and column of one matrix while keeping the other constant, and each time recompute the quantity Z^{*}_{XY}, and compare it to the original Z_{XY} value (Smouse et al. 1986).

In the case of three matrices, say **Y**, $\mathbf{X}_1$ and $\mathbf{X}_2$, the procedure is very similar. The partial correlation coefficients are obtained from the pairwise correlations as,

$$r_{Y X_1 . X_2} = \frac{r_{YX_1} - r_{X_1 X_2} r_{YX_2}}{\sqrt{(1 - r^2_{X_1 X_2})(1 - r^2_{YX_2})}} .$$

The other relevant partial correlations can be obtained similarly (see e.g. Sokal and Rohlf 1981). The significance of the partial correlations are tested by keeping one matrix constant and permuting the rows and columns of the other two matrices, recomputing each time the new partial correlations and comparing it to the observation (Smouse et al. 1986). Applications of the Mantel test in anthropology and genetics can be found in Smouse and Long (1992).

8 APPENDIX

8.1 Overview of input file key words

Keywords	Description	Possible values
[Profile]		
Title	A title describing the present analysis	A string of alphanumeric characters within double quotes
NbSamples	The number of different samples listed in the data file	A positive integer larger than zero
DataType	The type of data to be analyzed (only one type of data per project file is allowed)	STANDARD, DNA, RFLP, MICROSAT, FREQUENCY
GenotypicData	Specifies if genotypic or gametic data is available	0 (haplotypic data), 1 (genotypic data)
LocusSeparator	The character used to separate adjacent loci	WHITESPACE, TAB, NONE, or any character other than "#", or the character specifying missing data Default: WHITESPACE
GameticPhase	Specifies if the gametic phase is known (for genotypic data only)	0 (gametic phase not known), 1 (known gametic phase) Default: 1
RecessiveData	Specifies whether recessive alleles are present at all loci (for genotypic data)	0 (co-dominant data), 1 (recessive data) Default: 0
RecessiveAllele	Specifies the code for the recessive allele	Any string within quotation marks This string can be explicitly used in the input file to indicate the occurrence of a recessive homozygote at one or several loci. Default: "null"
MissingData	A character used to specify the code for missing data	"?" or any character within quotes, other than those previously used Default: "?"
Frequency	Specifies the format of haplotype frequencies	ABS (absolute values), REL (relative values: absolute values will be found by multiplying the relative frequencies by the sample sizes) Default: ABS
CompDistMatrix	Specifies if the distance matrix has to be computed from the data	0 (use any specified distance matrix), 1 (compute distance matrix from haplotypic information) Default: 0

FrequencyThreshold	The minimum frequency a haplotype has to reach for being listed in any output file	A real number between 1e-2 and 1e-7. Default: 1e-5
EpsilonValue	The EM algorithm convergence criterion. (For advanced users only)	A real number between 1e-7 and 1e-12. Default: 1e-7

Keywords	**Description**	**Possible values**
[Data]		
[[HaplotypeDefinition]]	(facultative section)	
HaplListName	The name of a haplotype definition list	A string within quotation marks
HaplList	The list of haplotypes listed within braces ({...})	A series of haplotype definitions given on separate lines for each haplotype. Each haplotype is defined by a haplotype label and a combination of alleles at different loci. The Keyword EXTERN followed by a string within quotation marks may be used to specify that a given haplotype list is in a different file

Keywords	**Description**	**Possible values**
[Data]		
[[DistanceMatrix]]	(facultative section)	
MatrixName	The name of the distance matrix	A string within quotation marks
MatrixSize	The size of the matrix	A positive integer larger than zero (corresponding to the number of haplotypes listed in the haplotype list)
LabelPosition	Specifies whether haplotypes labels are entered by row or by column	ROW (the haplotype labels will be entered consecutively on one or several lines, within the MatrixData segment, before the distance matrix elements), COLUMN (the haplotype labels will be entered as the first column of each row of the distance matrix itself)
MatrixData	The matrix data itself listed within braces ({...})	The matrix data will be entered as a format-free lower-diagonal matrix. The haplotype labels can be either entered consecutively on one or several lines (if LabelPosition=ROW), or entered at the first column of each row (if labelPosition=COLUMN). The special keyword EXTERN may be used followed by a file name within quotation marks, stating that the data must be read in an another file

Keywords	Description	Possible values
[Data]		
[[Samples]]		
SampleName	The name of the sample. This keyword is used to mark the beginning of a sample definition	A string within quotation marks
SampleSize	Specifies the sample size	An integer larger than zero. For haplotypic data, it must specify the number of gene copies in the sample. For genotypic data, it must specify the number of individuals in the sample.
SampleData	The sample data listed within braces ({...})	The keyword EXTERN may be used followed by a file name within quotation marks, stating that the data must be read in a separate file. The SampleData keyword ends a sample definition

Keywords	Description	Possible values
[Data]		
[[Structure]]	(facultative section)	
StructureName	The name of a given genetic structure to test	A string of characters within quotation marks
NbGroups	The number of groups of populations	An integer larger than zero
IndividualLevel	Specifies whether the level of genetic variability within individuals has to be taken into account (for genotypic data only)	0 :the component of variance due to differences between haplotypes within individuals will be ignored 1 :the component of variance due to differences between haplotypes within individuals, and its associated statistics will be computed
Group	The definition of a group of samples, identified by their SampleName listed within braces ({...})	A series of strings within quotation marks all enclosed within braces, and, if desired, on separate lines

Keywords	Description	Possible values
[Data] **[[Mantel]]**	(facultative section) Allows computing the (partial) correlation between *YMatrix* and *X1* (*X2*).	
MatrixSize	The size of the matrix entered into the project	An integer larger than zero
YMatrix	Specifies which matrix is used	"fst", "log_fst", "slatkinlinearfst", "log_slatkinlinearfst",

	as *YMatrix*.	"nm", "custom"
MatrixNumber	Number of matrices to be compared with the *YMatrix*.	1 :we compute the correlation between *YMatrix* and *X1* 2 :we compute the partial correlation between *YMatrix, X1*and *X2*
YMatrixLabels	Labels to identify the entries of the *YMatrix*. In case of *YMatrix*=”fst”, these labels should correspond to population names in the sample.	A series of strings within quotation marks all enclosed within braces, and, if desired, on separate lines
DistMatMantel	A keyword used to define a matrix, which can be either the Ymatrix, or another matrix that will be compared with the Ymatrix.	The matrix data will be entered as a format-free lower-diagonal matrix.
UsedYMatrixLabels	Labels defining the sub-matrix of the YMatrix on which the correlation is computed.	A series of strings within quotation marks all enclosed within braces, and, if desired, on separate lines

9 REFERENCES

Abramovitz, M., and I. A. Stegun, 1970 Handbook of Mathematical Functions. Dover, New York.

Aris-Brosou, S., and L. Excoffier, 1996 The impact of population expansion and mutation rate heterogeneity on DNA sequence polymorphism. Mol. Biol. Evol. 13: 494-504.

Cavalli-Sforza, L. L., and W. F. Bodmer, 1971 The Genetics of Human Populations. W.H. Freeman and Co., San Francisco, CA.

Chakraborty, R. 1990 Mitochondrial DNA polymorphism reveals hidden heterogeneity within some Asian populations. Am. J. Hum. Genet. 47:87-94.

Chakraborty, R., and K. M. Weiss, 1991 Genetic variation of the mitochondrial DNA genome in American Indians is at mutation-drift equilibrium. Am. J. Hum. Genet. 86: 497-506.

Cockerham, C. C., 1969 Variance of gene frequencies. Evolution 23: 72-83.

Cockerham, C. C., 1973 Analysis of gene frequencies. Genetics 74: 679-700.

Davies N, Villablanca FX and Roderick GK, 1999. Determining the source of individuals: multilocus genotyping in nonequilibrium population genetics. TREE 14:17-21.

Dempster, A., N. Laird and D. Rubin, 1977 Maximum likelihood estimation from incomplete data via the EM algorithm. J Roy Statist Soc 39: 1-38.

Efron, B. 1982 The Jacknife, the Bootstrap and other Resampling Plans. Regional Conference Series in Applied Mathematics, Philadelphia:.

Efron, B., and R. J. Tibshirani. 1993. An Introduction to the Bootstrap. Chapman and Hall, London.

Ewens, W.J. 1972 The sampling theory of selectively neutral alleles. Theor. Popul. Biol. 3:87-112.

Ewens, W.J. 1977 Population genetics theory in relation to the neutralist-selectionist controversy. In: Advances in human genetics, edited by Harris, H. and Hirschhorn, K.New York:Plenum Press,p. 67-134.

Excoffier, L., 2000. Analysis of Population Subdivision, in *Handbook of Statistical Genetics* D. Balding, M. Bishop and C. Cannings. (eds.) Wiley & Sons, Ltd.,

Excoffier, L., Smouse, P., and Quattro, J. 1992 Analysis of molecular variance inferred from metric distances among DNA haplotypes: Application to human mitochondrial DNA restriction data. Genetics 131:479-491.

Excoffier, L., and P. Smouse, 1994. Using allele frequencies and geographic subdivision to reconstruct gene genealogies within a species. Molecular variance parsimony. Genetics 136, 343-59.

Excoffier, L. and M. Slatkin. 1995 Maximum-likelihood estimation of molecular haplotype frequencies in a diploid population. Mol. Biol. Evol. 12:921-927

Excoffier, L., and M. Slatkin, 1998 Incorporating genotypes of relatives into a test of linkage disequilibrium. Am. J. Hum. Genet. (January issue)

Fu, Y.-X. (1997) Statistical tests of neutrality of mutations against population growth, hitchhiking and backgroud selection. Genetics 147:915-925.

Gaggiotti, O., and L. Excoffier, 2000. A simple method of removing the effect of a bottleneck and unequal population sizes on pairwise genetic distances. Proceedings of the Royal Society London B 267: 81-87.

Goudet, J., M. Raymond, T. de Meeüs and F. Rousset, 1996 Testing differentiation in diploid populations. Genetics 144: 1933-1940.

Guo, S. and Thompson, E. 1992 Performing the exact test of Hardy-Weinberg proportion for multiple alleles. Biometrics 48:361-372.

Harpending, R. C., 1994 Signature of ancient population growth in a low-resolution mitochondrial DNA mismatch distribution. Hum. Biol. 66: 591-600.

Hudson, R. R., 1990 Gene genealogies and the coalescent proces, pp. 1-44 in Oxford Surveys in Evolutionary Biology, edited by Futuyama, and J. D. Antonovics. Oxford University Press, New York.

Jin, L., and Nei M. (1990) Limitations of the evolutionary parsimony method of phylogenetic analysis. Mol. Biol. Evol. 7:82-102.

Jukes, T. and Cantor, C. 1969 Evolution of protein molecules. In: Mammalian Protein Metabolism, edited by Munro HN, New York:Academic press, p. 21-132.

Kimura, M. 1980 A simple method for estimating evolutionary rate of base substitution through comparative studies of nucleotide sequences. J. Mol. Evol. 16:111-120.

Kruskal, J. B., 1956. On the shortest spanning subtree of a graph and the travelling salesman problem. Proc. Amer. Math. Soc. 7:48-50.

Kumar, S., Tamura, K., and M. Nei. 1993 MEGA, Molecular Evolutionary Genetic Analysis ver 1.0.The Pennsylvania State University, University Park, PA 16802.

Lange, K., 1997 Mathematical and Statistical Methods for Genetic Analysis. Springer, New York.

Levene H. (1949). On a matching problem arising in genetics. Annals of Mathematical Statistics 20, 91-94.

Lewontin, R. C. (1964) The interaction of selection and linkage. I. General considerations; heterotic models. Genetics 49: 49-67.

Lewontin, R. C., and K. Kojima. (1960) The evolutionary dynamics of complex polymorphisms. Evolution 14: 450-472.

Li, W.H. (1977) Distribution of nucleotide differences between two randomly chosen cistrons in a finite population. Genetics 85:331-337.

Long, J. C., 1986 The allelic correlation structure of Gainj and Kalam speaking people. I. The estimation and interpretation of Wright's F-statistics. Genetics 112: 629-647.

Mantel, N. 1967. The detection of disease clustering and a generalized regression approach. Cancer Res 27:209-220.

Michalakis, Y. and Excoffier, L. , 1996 A generic estimation of population subdivision using distances between alleles with special reference to microsatellite loci. Genetics 142:1061-1064.

Nei, M., 1987 Molecular Evolutionary Genetics. Columbia University Press, New York, NY, USA.

Nei, M., and W. H. Li. 1979. Mathematical model for studying genetic variation in terms of restriction endonucleases. Proc.Natl.Acad.Sci.USA 76:5269-5273.

Paetkau D, Calvert W, Stirling I and Strobeck C, 1995. Microsatellite analysis of population structure in Canadian polar bears. Mol Ecol 4:347-54.

Paetkau D, Waits LP, Clarkson PL, Craighead L and Strobeck C, 1997. An empirical evaluation of genetic distance statistics using microsatellite data from bear (Ursidae) populations. Genetics 147:1943-1957.

Prim, R. C., 1957. Shortest connection networks and some generalizations. Bell Syst. Tech. J. 36:1389-1401.

Press, W. H., S. A. Teukolsky, W. T. Vetterling and B. P. Flannery, 1992. Numerical Recipes in C: The Art of Scientific Computing. Cambridge: Cambridge University Press.

Rannala B, and Mountain JL, 1997. Detecting immigration by using multilocus genotypes. Proc.Natl.Acad.Sci.USA 94:9197-9201.

Raymond M. and F. Rousset. 1994 GenePop. ver 3.0. Institut des Sciences de l'Evolution. Université de Montpellier, France.

Raymond M. and F. Rousset. 1995 An exact tes for population differentiation. Evolution 49:1280-1283.

Reynolds, J., Weir, B.S., and Cockerham, C.C. 1983 Estimation for the coancestry coefficient: basis for a short-term genetic distance. Genetics 105:767-779.

Rice, J.A. 1995 Mathematical Statistics and Data Analysis. 2nd ed. Duxburry Press: Belmont, CA

Rogers, A., 1995 Genetic evidence for a Pleistocene population explosion. Evolution 49: 608-615.

Rogers, A. R., and H. Harpending, 1992 Population growth makes waves in the distribution of pairwise genetic differences. Mol. Biol. Evol. 9: 552-569.

Rohlf, F. J., 1973. Algorithm 76. Hierarchical clustering using the minimum spanning tree. The Computer Journal 16:93-95.

Rousset, F., 1996 Equilibrium values of measures of population subdivision for stepwise mutation processes. Genetics 142: 1357-1362.

Rousset, F., 2000. Inferences from spatial population genetics, in *Handbook of Statistical Genetics*, D. Balding, M. Bishop and C. Cannings. (eds.) Wiley & Sons, Ltd.,

Schneider, S., and L. Excoffier. 1999. Estimation of demographic parameters from the distribution of pairwise differences when the mutation rates vary among sites: Application to human mitochondrial DNA. Genetics 152:1079-1089.

Slatkin, M., 1991 Inbreeding coefficients and coalescence times. Genet. Res. Camb. 58: 167-175.

Slatkin, M. 1994a Linkage disequilibrium in growing and stable populations. Genetics 137:331-336.

Slatkin, M. 1994b An exact test for neutrality based on the Ewens sampling distribution. Genet. Res. 64(1):71-74.

Slatkin, M. 1995 A measure of population subdivision based on microsatellite allele frequencies. Genetics 139: 457-462.

Slatkin , M. 1996 A correction to the exact test based on the Ewens sampling distribution. Genet. Res. 68: 259-260.

Slatkin, M. and Excoffier, L. 1996 Testing for linkage disequilibrium in genotypic data using the EM algorithm. Heredity 76:377-383.

Smouse, P. E., and J. C. Long. 1992. Matrix correlation analysis in Anthropology and Genetics. Y. Phys. Anthop. 35:187-213.

Smouse, P. E., J. C. Long and R. R. Sokal. 1986. Multiple regression and correlation extensions of the Mantel Test of matrix correspondence. Systematic Zoology 35:627-632.

Sokal, R. R., and F. J. Rohlf. 1981. Biometry. 2nd edition. W.H. Freeman and Co., San Francisco, CA.

Stewart, F. M. 1977 Computer algorithm for obtaining a random set of allele frequencies for a locus in an equilibrium population. Genetics 86:482-483.

Strobeck, K. 1987 Average number of nucleotide differences in a sample from a single subpopulation: A test for population subdivision. Genetics 117: 149-153.

Tajima, F. 1983 Evolutionary relationship of DNA sequences in finite populations. Genetics 105: 437-460.

Tajima, F. 1989a. Statistical method for testing the neutral mutation hypothesis by DNA polymorphism. Genetics 123:585-595,.

Tajima, F. 1989b. The effect of change in population size on DNA polymorphism. Genetics 123:597-601,.

Tajima, F. 1993. Measurement of DNA polymorphism. In: Mechanisms of Molecular Evolution. Introduction to Molecular Paleopopulation Biology, edited by Takahata, N. and Clark, A.G., Tokyo, Sunderland, MA:Japan Scientific Societies Press, Sinauer Associates, Inc., p. 37-59.

Tajima, F. and Nei, M. 1984. Estimation of evolutionary distance between nucleotide sequences. Mol. Biol. Evol. 1:269-285.

Tajima, F., 1996 The amount of DNA polymorphism maintained in a finite population when the neutral mutation rate varies among sites. Genetics 143: 1457-1465.

Tamura, K., 1992 Estimation of the number of nucleotide substitutions when there are strong transition-transversion and G+C content biases. Mol. Biol. Evol. 9: 678-687.

Tamura, K., and M. Nei, 1993 Estimation of the number of nucleotide substitutions in the control region of mitochondrial DNA in humans and chimpanzees. Mol. Biol. Evol. 10: 512-526.

Uzell, T., and K. W. Corbin, 1971 Fitting discrete probability distribution to evolutionary events. Science 172: 1089-1096.

Waser PM, and Strobeck C, 1998. Genetic signatures of interpopulation dispersal. TREE 43-44.

Watterson, G., 1975 On the number of segregating sites in genetical models without recombination. Theor.Popul.Biol. 7: 256-276.

Watterson, G. 1978. The homozygosity test of neutrality. Genetics 88:405-417

Watterson, G. A., 1986 The homozygosity test after a change in population size. genetics 112: 899-907.

Weir, B. S., 1996 Genetic Data Analysis II: Methods for Discrete Population Genetic Data. Sinauer Assoc., Inc., Sunderland, MA, USA.

Weir, B.S. and Cockerham, C.C. 1984 Estimating F-statistics for the analysis of population structure. Evolution 38:1358-1370.

Wright, S., 1951 The genetical structure of populations. Ann.Eugen. 15: 323-354.

Wright, S., 1965 The interpretation of population structure by F-statistics with special regard to systems of mating. Evol 19: 395-420.

Zouros, E., 1979 Mutation rates, population sizes and amounts of electrophoretic variation of enzyme loci in natural populations. Genetics 92: 623-646.