

---

# Getting Started with Blossoc

An introduction to the Blossoc association mapping tool

---

Thomas Mailund  
mailund@birc.au.dk

Copyright © 2007 Thomas Mailund • Bioinformatics Research Center, University of Aarhus

Permission is granted to make and distribute verbatim copies of this manual provided the copyright notice and this permission notice are preserved in all copies.

## About Blossoc

Blossoc is a linkage disequilibrium association mapping tool that attempts to build (perfect) genealogies for each site in the input and score these according to non-random clustering of affected individuals, and judge high-scoring areas as likely candidates for containing disease affecting variation. Building the local genealogy trees is based on a number of heuristics that are not guaranteed to build true trees, but have the advantage of more sophisticated methods of being extremely fast. Blossoc can therefore handle much larger datasets than more sophisticated tools, but at the cost of sacrificing some accuracy. Simulation studies, however, show that the loss in accuracy is alleviated by the added power when having larger sample sizes, so by being able to process much larger datasets, accuracy can be gained by increasing sample size.

The method underlying Blossoc is described in the paper

Mailund, T., Besenbacher, S., and Schierup, M.H., *Whole genome association mapping by incompatibilities and local perfect phylogenies*. BMC Bioinformatics 2006 7(454). doi:[10.1186/1471-2105-7-454](https://doi.org/10.1186/1471-2105-7-454).

Blossoc provides two interfaces to users: a graphical user interface for easy use, and a command-line based version for efficient batch runs. As of version 1.1, the command-line version supports (experimental) QTL mapping and epistasis mapping for pairs of unlinked genes. Future versions will extend on these methods and incorporate them into the graphical user interface as well.

We are currently only able to provide pre-compiled binary versions to Linux, due to lack of manpower and access to other architectures. Both command-line and graphical user interface versions should compile on any UNIX, Windows or Mac OS X provided the necessary libraries are installed (popt and GSL for the command line version and GSL and Qt3.3 for the GUI version). In future releases we hope to provide pre-compiled binaries for Windows and OS X as well.

Blossoc is released under the [GNU General Public License \(GPL\)](https://www.gnu.org/licenses/gpl-3.0.html). There are no restrictions on the use of Blossoc, for commercial or academic use, but the use of the source code, in part or in whole, is restricted according to the GPL.

### *Installing Blossoc*

Blossoc is distributed as RPM files or as source code. For most users, we recommend installing from the RPM files, since building the tool from source requires setting up the right build environment and having access to the needed development tools. If you are not familiar with UNIX C++ development—using the Automake suite of tools—and with Qt development we do not recommend that you try building from source.

*Installing the RPM Files.* The RPM file, `blossoc-gui-x.y.z-r.i386.rpm`—where `x.y.z-r` is the version and release number—contains a binary version of both the command-line and the GUI program, compiled to an Intel

x86 Linux platform; the RPM file `blossoc-x.y.z-r.i386.rpm` contains the command-line tool only. To install Blossoc from the RPM package, run

```
> rpm -Uvh blossomoc-gui-x.y.z-r.i386.rpm
```

or

```
> rpm -Uvh blossomoc-x.y.z-r.i386.rpm
```

Since the RPM files installs in the directory `/usr/local/`, installing the RPM package requires root access.

*Installing from the Source Files.* The source code is distributed in a tar-file, `blossoc-x.y.z.tar.gz`. To build the source files, first uncompress and untar the file, then run ‘configure’ and finally ‘make’. To test that the build was successful, run ‘make check’. To install the program, run ‘make install’.

```
> tar xzf blossomoc-x.y.z.tar.gz
> cd blossomoc-x.y.z
> ./configure
> make
> make check
> make install
```

To also build the GUI, you must have Qt installed and then do

```
> cd Blossoc-Qt
> qmake
> make
```

## Running Blossoc

The command-line version of Blossoc is run by typing

```
> blossomoc
```

on the command line.

Installing the graphical user installing version should, on GNOME or KDE desktops, add an icon in the start menu for running Blossoc. If this is not the case, the tool can be started on the command-line with the command

```
> Blossoc-Qt
```

## Using Blossoc

For running Blossoc, you need to specify: 1) A list of marker positions, 2) A list of haplotypes—containing a SNP value (0 or 1) for each marker—together with case/control status, and 3) A choice of clustering score function. Optionally, you may also specify the minimal number of markers to consider for each point being scored, but we recommend using the default value for this option, unless compelling evidence suggests otherwise.

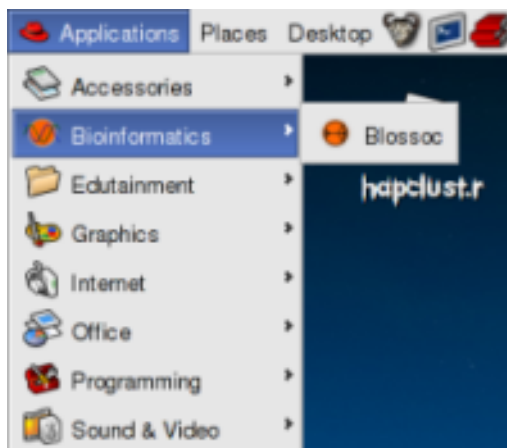
The graphical user interface for Blossoc lets you select these options through dialogues, while the command-line interface lets you specify them as arguments to the command.

### Running the GUI Blossoc

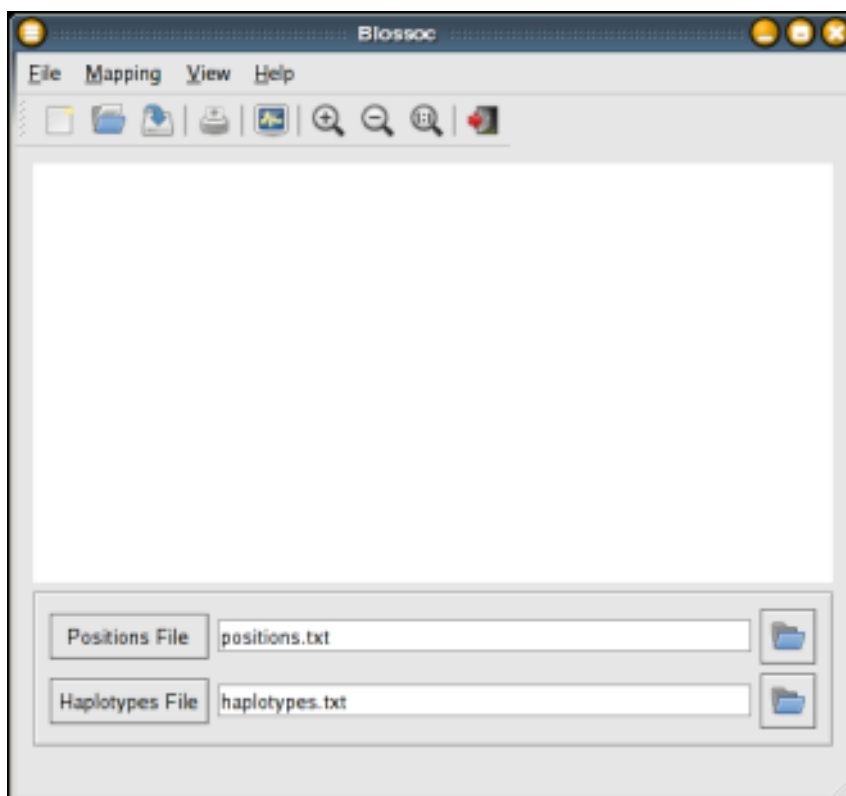
Once the GUI Blossoc has been installed, on most platforms it will appear in the menu, as shown in Fig. 1. If it does not, however, it can be started from the command line as

```
> Blossoc-Qt
```

When started, the main window, shown on Fig. 2, appears. Through the main window you can load and save ongoing projects plus specify the marker positions and haplotypes by specifying which files they should be read from.



**Figure 1:** The Blossoc icon in the menu.



**Figure 2:** The Blossoc main window before any mapping is done.

The positions file is selected by pressing the **Position File** button or by typing in the file name in the text entry next to the button. In the later case, the file is not immediately loaded, but will be loaded when the mapping algorithm is started. The positions file should contain a list of numbers, sorted ascending.

The haplotypes file is selected by pressing the **Haplotypes File** button or by typing in the file name in the text entry next to the button. As for the positions, in the later case, the file is not immediately loaded, but will be loaded when the mapping algorithm is started. The format of the haplotype file is: One line per haplotype, where a haplotype is represented as a list of space-separated alleles, and each allele represented as either a '0' or a '1'. The first column is a 'pseudo'-allele used for the case/control dichotomy: a '0' in the first column is taken to mean that the haplotype is a *control* haplotype and a '1' at the first column is taken to mean that the haplotype is a *case* haplotype.

The scoring function is selected in the **Mapping Parameters** dialogue, found in the `Mapping` menu (see Fig. 3). The scoring function determines how the clustering of a local phylogeny should be scored for significance. The choices are:

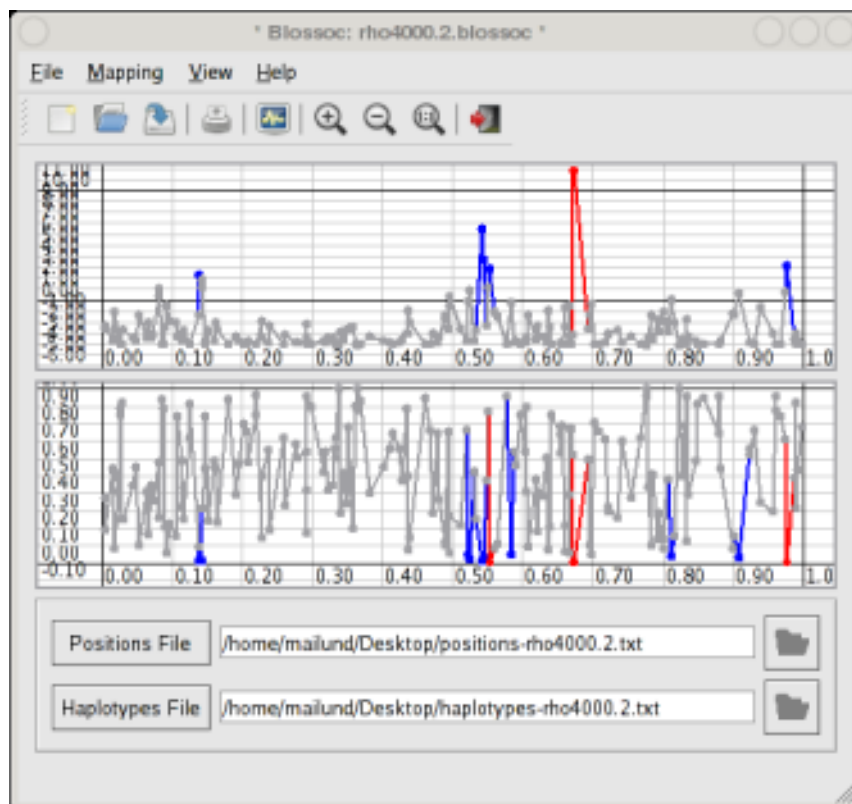
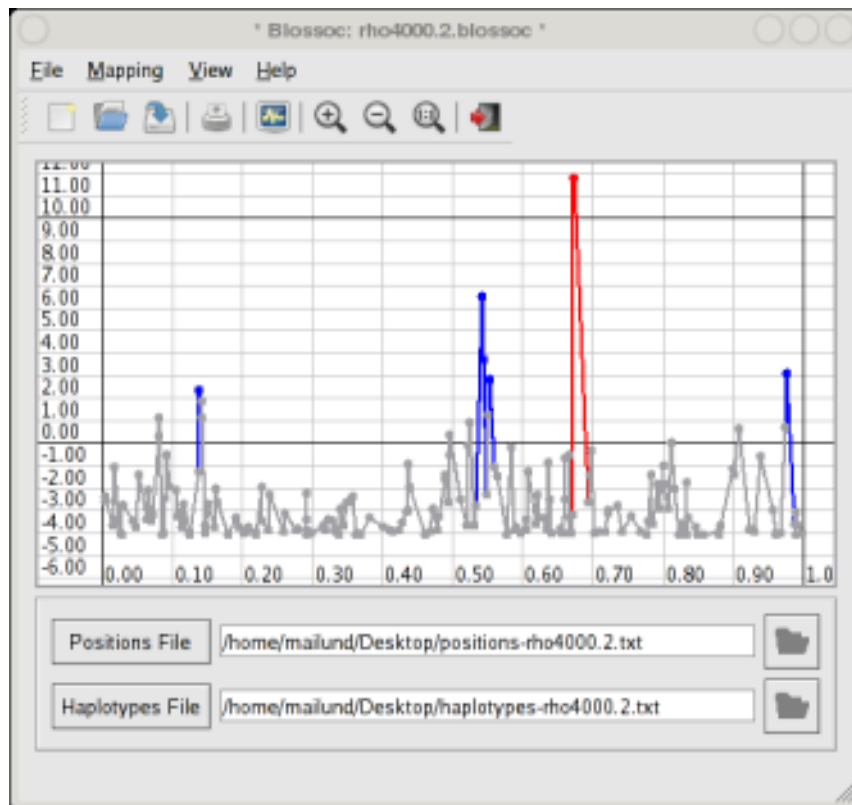
- **AIC** — Akaike's information criterion.
- **BIC** — Bayesian information criterion.
- **Gini** — Gini information score.
- **HQC** — Hannan and Quinn criterion.
- **Prob. score** — a multinomial probability based score.



**Figure 3:** Dialogue for setting mapping parameters.

For datasets with less than 200 individuals, we recommend using the Prob. score, and for datasets with more than 200 individuals we recommend HQC.

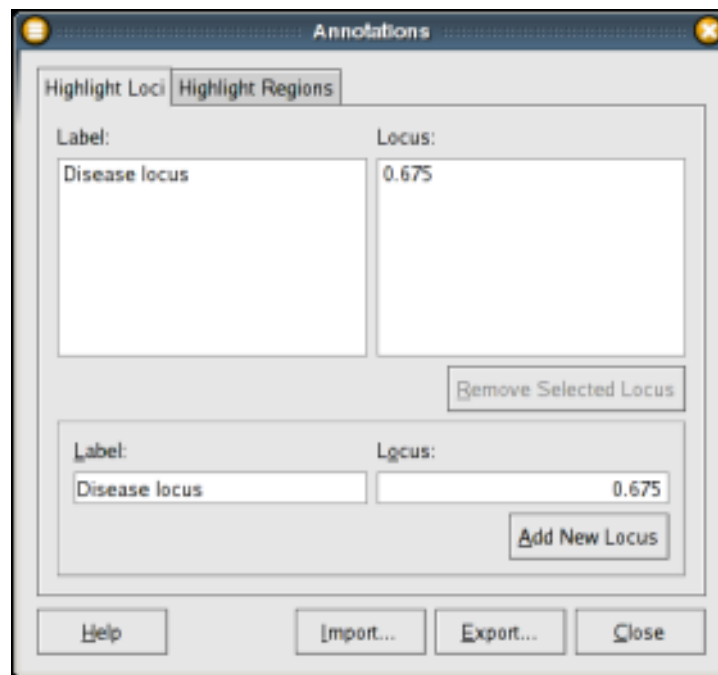
*Running the mapping algorithm.* Once the parameters have been set, the mapping algorithm can be run by selecting **Start Mapping** in the **Mapping** menu, or by pressing **Ctrl-Enter**. The result will be shown in the main window (as in Fig. 4 (top)) and can be printed or exported to a text file.



**Figure 4:** The Blossoc main window after running the mapping algorithm (top) and showing both the Blossoc mapping scores and the single marker ( $\chi^2$ ) association p-values (bottom).

From the **Mapping** menu, it is also possible to calculate a simple  $\chi^2$  single marker association score and a permutation test for the Blossoc score. Figure 4 (bottom) shows the Blossoc main window displaying both the Blossoc score and the single marker p-values.

*Annotating the mapping graphs.* It is possible to annotate the mapping graphs with interesting loci or regions (such as genes or loci previously known to be associated with the disease of interest). This is done by selecting **Annotations...** from the **View** menu. This opens the dialogue shown in Fig. 5.

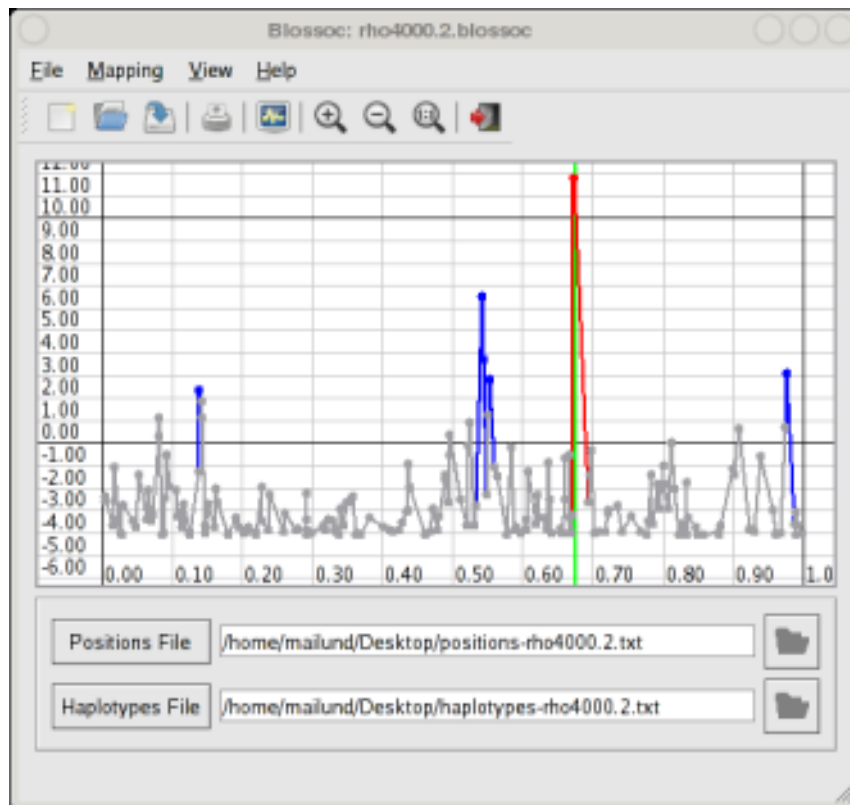


**Figure 5:** The dialogue for specifying annotations.

From this dialogue, you can specify loci (single points of interest) or regions (contiguous regions of loci) to highlight. Each locus or region is identified with a label and given a genomic coordinate (for loci) or a start and end coordinate (for regions). Figure 6 shows an example of annotations; it shows a green vertical line running through the position of the true causative SNP in a simulated dataset.

### *Running the command-line Blossoc*

When running the command-line version of Blossoc, the positions file is given as the first parameter after the command-line options (or specified with the `-p` or `--positions=` option) and the haplotypes file as the second parameter after the options (or with the `-h` or `--haplotypes=` option). The scoring function is selected with the `-f` or `--scorefunction=` parameter, that can be set to one of A, B, G, H, or P, for AIC, BIC, Gini, HQC, and Prob. scoring, respectively. The output is written to a file specified with the `-o` or `--scores=` option. To get a list of all supported options, use `-h` or `--help`.



**Figure 6:** The main window with a mapping and a locus highlighted (by the green vertical line at position 0.675).

```
> blossom -f H -o output.txt positions.txt haplotypes.txt
> blossom -p positions.txt -h haplotypes.txt -f H -o output.txt
```

The output file is simply a sequence of space-separated numbers, one for each position in the input, that contains the Blossoc score for that marker. Positive scores indicate association and negative scores no association; the higher the score the stronger the association (see [Mailund et al. 2006](#) for details).

**QTL mapping.** The command line version supports an experimental quantitative trait mapping method. By giving Blossoc the option `-q` the first column in the haplotypes file is treated as a quantitative trait rather than a case/control indicator, and the mapping uses the QT method.

The QTL method supports the same scoring functions as the case/control method, except for the prob. score.

**Epistasis.** The command line version supports an experimental method for mapping in the presence of interaction between two (unliked) genes. Currently, this method only supports case/control data, but future versions will support quantitative traits as well.

To map with interaction, use the program `iblossoc` and specify two input regions (in the form of a position file followed by a haplotype file). The out-

put will contain a row for each marker in the first region and a column for each marker in the second region, each cell containing the score for the joint association of the two markers.

```
> iblossoc -o output.txt \  
           positions.1.txt haplotypes.1.txt \  
           positions.2.txt haplotypes.2.txt
```

The interaction method only supports one scoring function (different from all of the methods used in the case/control and QTL method). Use option `-h` or `--help` to get a list of supported options for `iblossoc`.

Using `iblossoc` can be very memory consuming if the two regions are large. If this is a problem, the tool `low_mem_iblossoc` implements exactly the same functionality as `iblossoc`, but sacrifices runtime efficiency for space efficiency. It is thus slower than `iblossoc` but uses vastly less memory.

### *Using SNPfiles*

If Blossoc is compiled with support for the [SNPfile binary file format](#), you can analyse such files using the tool `snpfile_blossoc`

```
> snpfile_blossoc genotypes.snp output.txt
```

The output of `snpfile_blossoc` is currently different from the output from plain `blossoc` but is in a format that can easily be read into R using the `read.table` function with `header=TRUE`.

The input file is assumed to be in the format generated by the [text2snpfile](#) tool, part of the SNPfile library.

For interaction, the tool `snpfile_iblossoc` implements the same algorithm as `low_mem_iblossoc`.<sup>1</sup>

```
> snpfile_iblossoc region1.snp region2.snp
```

SNPfile data is currently not supported in the GUI.

## **Contact**

For comments or questions regarding Blossoc, please contact Thomas Mailund ([mailund@birc.au.dk](mailto:mailund@birc.au.dk)) or Søren Besenbacher ([besen@birc.au.dk](mailto:besen@birc.au.dk)).

---

<sup>1</sup>The SNPfile format is aimed at dealing with large datasets, and for those using `iblossoc` is usually not feasible, thus the low-memory but more CPU intensive version is used.