

MapQTL[®] 5

Software for the mapping of quantitative trait loci
in experimental populations

J.W. van Ooijen

Wageningen, February 2004

MapQTL is developed in collaboration with statistical geneticists of Biometris of Wageningen UR (www.biometris.nl). The sales and support are taken care of by Kyazma B.V..

Copyright © 1996-2004 Plant Research International B.V. and Kyazma B.V.
All rights reserved. Unauthorized reproduction and distribution prohibited.

MapQTL and JoinMap are a trademarks of Plant Research International B.V. and Kyazma B.V. registered in the Benelux and the U.S.A.. Other brand and product names are registered trademarks of their respective holders.

Kyazma B.V.
P.O. Box 182
6700 AD Wageningen
Netherlands

support@kyazma.nl
www.kyazma.nl

Contents

Introduction	1
Installation	1
Overview	2
Final remarks	4
How to cite MapQTL	5 ? 5
Acknowledgement	5
Using MapQTL	7
Controlling the program	7
The MapQTL project	8
Navigation panel	9
Contents-and-results panel	10
Starting an analysis	11
Nonparametric mapping (Kruskal-Wallis analysis)	12
Nonparametric mapping output	13
Interval mapping	14
Interval mapping output	15
MQM mapping	16
MQM mapping output	17
Automatic selection of cofactors	18
Permutation test	19
Permutation test output	20
Tutorial	21
Mapping theory	31
Interval mapping	31
Genotypic information coefficient	33
Selective genotyping	35
MQM mapping	35
LOD significance threshold	38
Data files	39
General	39
Data file characteristics	39
Locus genotype file	40
Map file	46
Quantitative data file	48
Cofactors file	50
Default file name extensions	50
Lists and references	51
List of figures	51
List of tables	51
List of examples	51
References	52
Index	55

Introduction

MapQTL[®] is a computer program for the calculation of QTL positions on genetic linkage maps in experimental populations of diploid species. The present version 5 is based on its predecessor, version 4.0 (Van Ooijen et al, 2002), of which the user interface is completely revised, giving more ease of use, better QTL charts and improved exportability. Multiple populations and maps can now be loaded into a project, thereby allowing an easier comparison of results across related populations or using different maps. Results of analyses are stored in so-called sessions within the project; these sessions can be inspected immediately after the computations and stay available for later re-inspection. A very important enhancement is the creation of QTL charts by the program, in which charts of all or a selection of linkage groups can be combined on a single page and many options are easily controlled. The results and their charts can be exported to files, copied to other MS-Windows[®] programs like MS-Word[®] or MS-Excel[®], and printed, and there is also a preview prior to printing.

Installation

MapQTL is a program for the MS-Windows platform on the PC. It was tested to run under the Windows versions ME, NT 4.0, and XP, and is further expected to run flawlessly under all current PC Windows platforms starting from 95 and above. It comes with an InstallShield[®] installation program that does most of the installation work. Start the SETUP.EXE program from the set of installation files, e.g. by double-clicking on it from within *Windows Explorer* or *My Computer*. Choose the settings you are prompted for and let SETUP.EXE finish. After this process the license file MAPQTL.LIC will be present in the program directory (typically: C:\Program Files\MapQTL5). This is the *evaluation* license file which allows you to use the software with your own and demonstration data under certain limitations: there are maxima of two populations, two numerical traits per population and two linkage groups per map, while printing, copying to the clipboard and exporting to file are not available. A purchased copy of MapQTL comes with your individual license file, which usually resides in the *Licenses* directory of the product CD. Replace the evaluation license file with your individual license file, and make sure it gets the name MAPQTL.LIC; in the MapQTL *Help* menu there is an *Install License* function that can assist you with this. Successful installation of the individual

license removes all above mentioned limitations and gives unrestricted access to the program; the About-box will show the name of the licensed organisation.

MapQTL 5 stores its various program settings in the directory *MapQTL5* which is created in the *My Documents* directory when running the program.

Apart from the length of names (maximum of twenty characters for population, locus, trait and linkage group names) there are no limits built into the software, memory for storing data is allocated dynamically only for the amount needed. Thus, project size is limited only by the amount of RAM memory in the PC, for which a size of 256 MB is recommended for reasonably sized projects.

Overview

Start the program by using the Windows *Start* menu. When the program runs you will see a window that is divided into several main parts: on the top the *menu* and the *tool bar* with buttons and a selector, on the left-hand side the *navigation* panel, on the right-hand side the *contents-and-results* panel, and on the bottom the *status bar* ([Figure 1](#)).

The navigation panel contains three tabbed pages (tabsheets), which will show bookmarks for the data loaded into a mapping project (the traits and the marker genotypes of populations, the maps) and for the analysis sessions performed. These bookmarks are given as nodes in treeviews, like the *Folders* panel in the Windows Explorer. The contents-and-results panel also contains several tabbed pages which (as the name suggests) display the contents or results of the bookmark selected in the navigation panel.

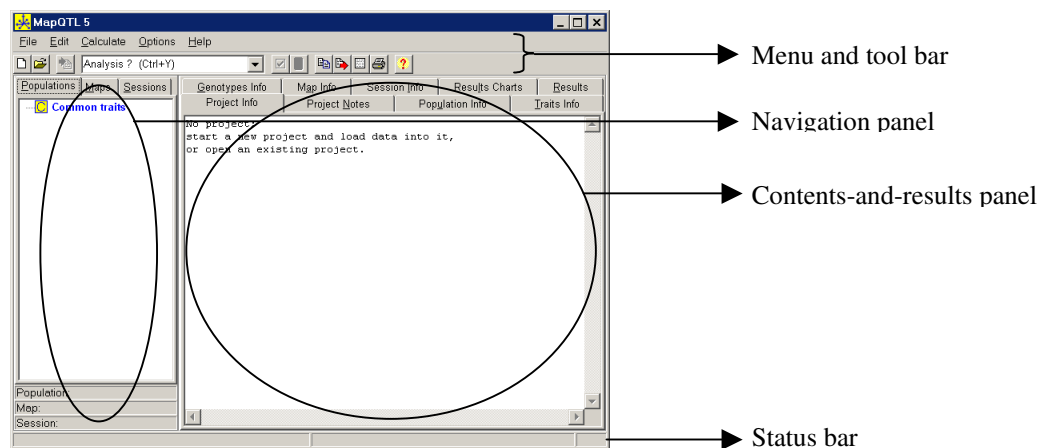


Figure 1. User interface

The QTL analysis is organised into so-called *MapQTL projects*. A project consists of a project file and a project directory, both are (and must be) in the same directory. The project directory will contain all files used internally by the program. You can view these plain text files, but it is strongly advised not to edit, remove or rename them, because that may damage the project so that it cannot be handled by MapQTL anymore (copying is fine). When creating a new project, which is done using *New Project* function of the *File* menu or the *New Project* button , you are prompted for a project file name (with a standard save-file dialog-window); a project file with the extension *.mqp* and the corresponding project data directory of the same name with the extension *.mqd* will be created. Once a new project is opened, you load data into the project. This must be done with the *Load Data* function of the *File* menu (or with a tool bar button ) . Data must be loaded from three separate files: (1) the set of locus genotypes of a population, (2) the set of quantitative trait data of a population, and (3) the map data. The formats of data files used by MapQTL are described thoroughly in the [Data files chapter](#) (p. 39). Data for demonstration are available in the DemoData subdirectory of the program directory (typically: C:\Program Files\MapQTL5). There may be more than just one population and more than just a single map loaded into a project.

After successful loading of data into a project, the *Populations* tabsheet on the navigation panel will show the populations with their traits and genotypes as nodes in a treeview; the *Maps* tabsheet will also show the maps and their linkage groups as nodes in a treeview. In addition, the *Populations* tabsheet has a node called *Common traits* with as child-nodes the traits available in all loaded populations. (NB: Traits within the quantitative trait data set that contain (some) non-numerical data will show up as nodes with a green font and icon; these cannot be used for analysis.)

The selection of a node within the navigation panel enables the inspection of its data in the corresponding tabsheet of the contents-and-results panel. The names of the selected nodes are given in the three stacked *status bars* at the bottom of the navigation panel. In addition to the tabsheets for the loaded data and the results of analyses, the contents-and-results panel also has a *Project Info* tabsheet with an overview of all actions done within the project, and a *Project Notes* tabsheet on which you can make your own notes about the project and which will be stored with the project.

In order to perform an analysis, the trait or traits must be selected from the Populations tabsheet by *right-clicking* their nodes (or by pressing the space-bar when the node is selected, i.e. usually blue); as a result the node will show up magenta or red. This selection is a toggle, i.e. right-clicking again will deselect the node. Selection of child nodes under the Common traits node automatically selects the trait within all populations. In a similar fashion, the linkage groups that must be analysed for the selected trait(s) must be selected by right-clicking their nodes on the Maps tabsheet. Once this is done and an analysis is selected on the tool bar, the *Calculate* function will be enabled (available both

as a menu option and as a tool bar button ) and can be chosen or clicked, respectively. This will start a so-called *calculation session*, with a corresponding node in the *Sessions* tabsheet on the navigation panel. Nodes will be created in the Sessions tabsheet for each analysed trait and linkage group, with the appropriate hierarchy of the various nodes in the sessions treeview. If the analysis requires marker cofactors, they can be selected by checking their *Cofactor* checkboxes on the *Map Info* tabsheet (of course the appropriate map must be selected in the Maps tabsheet). The *Cofactors Tool* can be very helpful with this; it is available from the *Edit* menu and from a tool bar button .

The results of a calculation session can be inspected by selecting the requested trait node in the sessions treeview; the results will be shown on the *Results* tabsheet as a table and as a chart or set of charts on the *Results Charts* tabsheet. An exception here are the results of automatic cofactor selection which are shown as plain text on the Results tabsheet and are not shown graphically. The Results Charts tabsheet contains a set of two subordinate tabsheets, one for the control of the charts and one for the actual charts. There are many features of the charts that can be handled using this subordinate *Control* tabsheet.

The current view of the contents-and-results panel (except the chart control tabsheet) can be *printed*, *exported* to file and *copied* to the MS-Windows clipboard to enable the pasting into for instance an MS-Word document. This can be done using the *Print* option of the *File* menu and the *Export To File* and *Copy To Clipboard* options of the *Edit* menu. The tool bar has buttons to perform these functions: , , , respectively. When one or more rows in a table are selected, or when there is a text selection in a plain text view, the print, export and copy functions are performed on the selection only; pressing ctrl+A will select all of the current view. Charts are exported in the Enhanced Windows Meta File format, which as an MS-Windows standard can be used in many other applications. Prior to printing, a preview of the print-out can be obtained through the *Print Preview* option of the *File* menu or the tool bar button . From within the Print Preview and from the *File* menu the *Page Setup* and the *Print Setup* can be modified.

This user manual is accessible as an Adobe® pdf-document through the *Help* menu.

Final remarks

With MapQTL 5 you have quite a powerful tool to analyse the data that you have obtained from your experiments. It is important to realize that the quality of your data is crucial to the possibility to discover real QTLs: many missing marker observations reduce the power of the analyses, erroneous marker scores and an incorrect linkage map (usually the product of missing and erroneous marker observations) both may generate inconsistent results. It does not lie within the power of a software tool to compensate for

missed out quality of its input data. But even with good quality data, the detection of QTLs is only possible if QTLs do segregate in the population under study and if their genetic effects are sufficiently large in relation to the residual variance and the size of the experiment. And above all, you have to keep in mind that MapQTL is a statistical tool and that the results point you to statistical conclusions with a definite amount of uncertainty.

How to cite MapQTL 5 ?

Van Ooijen, J.W., 2004. MapQTL ® 5, Software for the mapping of quantitative trait loci in experimental populations. Kyazma B.V., Wageningen, Netherlands.

Acknowledgement

All new versions of software programs build on their predecessors, MapQTL 5 is no exception; the main contributors to version 4.0 are gratefully acknowledged for their input: Martin Boer, Ritsert Jansen and Chris Maliepaard. To the present version several people of Wageningen University and Research Centre contributed with wish lists, remarks, positive criticism, software testing and alike: Richard Finkers, Sjaak van Heusden, Hans Jansen, Piet Stam, Roeland Voorrips; their assistance is greatly appreciated!

Using MapQTL

The program can be started in the various ways of MS-Windows, by using the *Start* menu, by double-clicking on the *MapQTL5.exe* file from within *Windows Explorer* or *My Computer*, or by double-clicking on a project file. The latter way is established only after running the program a first time. When the program runs you will see a window that is divided into several main parts: on the top the *menu* and the *tool bar* with buttons and a selector, on the left-hand side the *navigation* panel, on the right-hand side the *contents-and-results* panel, and on the bottom the *status bar* ([Figure 1](#)). Once a project is created and data are loaded, the navigation panel will show the populations with their traits and genotypes, and the maps with their linkage groups. The contents-and-results panel contains a set of tabbed pages (tabsheets), in which contents of data sets and results of analyses are displayed for the population, map and analysis session *selected* (i.e. blue) in the navigation panel.

Controlling the program

Because MapQTL is an MS-Windows program, you can expect the many features to be controlled in the normal MS-Windows fashion with the mouse and the keyboard. Below is a summary of some normal and special keys and key combinations:

alt+ <u>key</u>	<u>key</u> being any underlined character shown in the program: as usual, go to the associated part of the window or perform the associated action
ctrl+A	select all in selected tabsheet of the contents-and-results panel
ctrl+C	copy the selected tabsheet of the contents-and-results panel to clipboard (or its selection)
ctrl+F	open the <i>Find</i> dialog
ctrl+H	show the <i>Charts</i> page within the <i>Results Charts</i> page
ctrl+O	show the <i>Control</i> page within the <i>Results Charts</i> page
ctrl+P	print the selected tabsheet of the contents-and-results panel (or its selection)
ctrl+Y	go to the <i>Analysis</i> selector
shift+Del	delete the selected node in the visible tree view
Tab	rotate focus through all visual elements
Esc	close the <i>Cofactors</i> tool; close the <i>Print Preview</i> tool; cancel the options

	dialogs; cancel the calculations
Break	cancel the calculations
F1	view the manual as Adobe pdf-document
F4	load data into the project
F9	execute the selected analysis on the selected traits and selected linkage groups
alt+F4	exit the program

The *Environment Options* of the *Options* menu allow the setting of the fonts for the various elements of the user interface and the various chart options. The *Analysis Options* allow the setting of the various calculation parameters. Clicking on the *Preset default* button on these options dialogs, changes all values to the internal program values of all parameters. Clicking on the *Save as default* button stores all current values to the program settings directory (*My Documents\MapQTL5*), which will be used as starting values for each new project and can be loaded into an opened project by clicking on the *Reset to default* button. Clicking on the *OK* button applies all fonts settings immediately to the current project; all chart and analysis options will be applied to all new calculation sessions. The new chart options can be applied to existing charts by clicking on the *Reset* button on the charts Control page after selecting the particular chart.

The MapQTL project

In MapQTL your work is organised into a *project*. You create a new project  or open an existing project  using the *File* menu or tool bar buttons; the location and name of the project can be chosen with a standard save-file dialog-window. The whole of a MapQTL project consists physically of (a) the project file with extension *.mqp*, and (b) the project data directory with the same name as the project file, but with the extension *.mqd*. The project data directory resides in the same directory as the project file; it will contain all (many) internal data files. After the data files (i.e. the locus genotype file, the map file, and the quantitative data file) are loaded into the project, the original files are not needed by MapQTL. When backing up a MapQTL project, always take the project file as well as the project directory with all its files.

Once a project is opened, you can load data into the project. This must be done with the *Load Data* function of the *File* menu (or with a tool bar button ). Data must be loaded from three separate files: (1) the set of locus genotypes of a population, (2) the set of quantitative trait data of a population, and (3) the map data; the location and name of the files can be chosen with a standard save-file dialog-window. The formats of data files used by MapQTL are described thoroughly in the [Data files chapter](#) (p. 39). Some example data files are present in the DemoData subdirectory of the program directory

(typically: C:\Program Files\MapQTL5). More than just one population and more than a single map can be loaded into a project.

Navigation panel

The navigation panel has three tabsheets and at the bottom three small *status bars*. The *Populations* tabsheet will show the populations with their traits and genotypes loaded into the project. The *Maps* tabsheet will display the maps with their linkage groups. The *Sessions* tabsheet will show the calculation sessions that have been performed. These tabsheets show their contents as so-called *treeviews*, like the *Folders* panel in the Windows Explorer. Populations, their traits and genotypes, and maps with their linkage groups are shown hierarchically as nodes in a tree. The Populations tabsheet also has a *Common traits* node, which will show all traits (as its child nodes) that are common to all loaded populations. NB: Traits within the quantitative trait data set that contain (some) non-numerical data will show up as nodes with a green font and icon, i.e. different from completely numerical traits, as they cannot be used for analysis.

The nodes in the treeviews can be selected by clicking on them. The names of the population, map and session currently selected are shown in the three small status bars at the bottom of the navigation panel. The selection of a node enables the inspection of its data in the corresponding tabsheet of the contents-and-results panel. Selecting a linkage group child node under a map or session node puts the focus on the corresponding table position in the Map Info or Results tabsheets of the contents-and-results panel, respectively.

The trait nodes and the linkage group nodes can also be *specially* selected by *right-clicking* (or by pressing the space bar when the node is selected, i.e. usually blue). This is a special type of selection, which applies only to the analysis that is going to be performed on these traits and linkage groups. As a result the nodes will show up in *red* (or *magenta* for the selected node that should be blue as well). Right-clicking any trait node under the Common traits node will apply this special selection to this trait under all populations. Right-clicking a population node specially selects all numerical traits of that population. Right-clicking a map node specially selects all linkage groups of that map. This special selection is a toggle, i.e. when right-clicking again the nodes are deselected.

Only when (1) one or more traits and (2) one or more linkage groups are specially selected, (3) the corresponding population(s) has the genotypes loaded, and (4) an analysis is selected on the tool bar, only then the Calculate function is enabled and it becomes possible to activate the Calculate button or Calculate function of the Calculate menu. Once calculations have been performed, a corresponding session node is created in

the session treeview, with subordinate nodes for each trait and linkage group, all shown hierarchically.

It is possible to remove complete maps (i.e. all linkage groups), complete populations (i.e. all traits and genotypes), complete sets of traits of populations, the set of genotypes of populations and complete sessions from the project. This can be done by selecting (blue) the relevant node and pressing shift+Del or applying the *Delete Node* function from the *File* menu. There is a *splitter* between the navigation panel and the contents-and-results panel, allowing you to reassign the space available to the panels.

Contents-and-results panel

The contents-and-results panel contains a set of tabsheets that will display the data of the population, map and session nodes. In addition to the tabsheets for loaded data (i.e. *Population Info*, *Traits Info*, *Genotypes Info* and *Map Info* tabsheets) and the results of analyses, the contents-and-results panel also has a *Project Info* tabsheet with an overview of all actions done within the project, and a *Project Notes* tabsheet on which you can make your own notes about the project and which will be stored with the project.

The *Map Info* tabsheet does not only show the linkage groups and positions of loci, it also has a column with checkboxes for indicating whether loci should be used as cofactors in the analysis to be performed. The *Cofactors Tool* can be very helpful with this; it is available from the *Edit* menu and from a tool bar button . It is a floating tool allowing you to change the tabsheets while the tool remains available. The *Map Info* tabsheet will indicate for each locus its presence within each loaded population by the character "X" in the column for that population. Any locus present within a loaded population, but not present in the loaded map file, will be added to the map on the *Map Info* tabsheet as an unmapped locus without a map position. A node named *Unmapped* will be created in the *Maps* navigation treeview, that will correspond to all unmapped loci.

Depending on the type of data the contents and results are shown as plain text or as a table on the *Results* tabsheet. Results shown as a table can also be viewed as a chart (see below). Of any table the view can temporarily be changed: columns can be moved by dragging the header with the mouse, column widths can be resized by dragging the *splitter* between column headers, rows can be sorted by clicking on the column header to use as sorting key (click twice for sorting in the opposite direction). You can revert to the original row order by sorting on the first column, always labelled *Nr*. Some functions within the program will automatically revert corresponding tables to the original row order. The changes in the view are not stored, so closing and reopening the project results in the original views of the tables.

The *Results Charts* tabsheet contains a set of two subordinate tabsheets, one for the control of the charts and one for the actual charts. There are many features of the charts that can be handled using this subordinate *Control* tabsheet. On this tabsheet there are *splitters* that can be dragged with the mouse to divide the space between the checklists for *Groups* and *Left* and *Right Y-axis* data, and also between the upper part with its checklists and the lower part with checkboxes and fields for various chart options. Most features are self-explaining, just two need some description: when plotting cofactors these will always be plotted as symbols on the X-axis; when plotting the results of an unmapped group of loci these will always show symbols and there will be no line connecting the points. Clicking the *Reset* button on the charts *Control* page restores all chart options to the project default values.

The tabsheet on display (or a selection of it) can be exported to file , printed  and copied to the clipboard  using the corresponding *File* or *Edit* menu options or tool bar buttons. File export and copying to clipboard are useful for taking the data or charts to, for instance, MS-Excel or MS-PowerPoint®. Charts are exported in the Enhanced Windows Meta File (.emf) format, which as an MS-Windows standard can be used in many other applications. When one or more rows in a table are selected (not necessarily a contiguous set of rows), or when there is a text selection in a plain text view (selection is done in the regular MS-Windows fashion), the print, export and copy functions are performed on the selection only. Selections can also be dragged with the mouse and subsequently dropped into other (accepting) applications, such as MS-Excel or MS-Word. Prior to printing, a preview of the print-out can be obtained through the *Print Preview* option of the *File* menu or the tool bar button . From within the Print Preview and from the *File* menu the *Page Setup* and the *Print Setup* can be modified. The Print Preview also allows the selection of pages for printing.

As a nice navigation feature the selection of a linkage group node in the Maps tabsheet will select the first locus of that group in the Map Info tabsheet. Similarly, the selection of a linkage group node in the Sessions tabsheet will select the first locus of that group in the Results tabsheet.

Finally, when you are looking for certain text on a plain text or table tabsheet in the contents-and-results panel, you can make use of the *Find* tool, available from the *Edit* menu. It is a floating tool allowing you to change the tabsheets while the tool remains available.

Starting an analysis

On the tool bar there is a selector for the analysis. In order to start an analysis you must

first choose the analysis itself from this selector. Secondly, the traits that must be analysed must be *specially* selected by right-clicking (see the [Navigation panel section](#), p. 9). Even when traits are to be analysed separately multiple traits can be selected to be analysed in one go. Traits that have non-numerical data cannot be selected for analysis; they are shown in green. When a population has no subordinate *Genotypes* node, i.e. its locus genotypes are not loaded into the project, the traits for that population cannot be analysed. Thirdly, the linkage groups that must be analysed must be (specially) selected on the Maps tabsheet. The group of unmapped loci, represented by the node *Unmapped*, can only be used for separate nonparametric and interval mapping analyses. Only if all above conditions are satisfied will the Calculate function be enabled. When all set, the analysis can be started by pressing the *Calculate* button , by pressing the F9 function key, or by using the *Calculate* menu. When everything is OK the progress of the calculations is shown on the right-hand side of the status bar, a set of nodes is created on the Sessions tabsheet, and when done the results are shown on the Results tabsheet. The parent of the set of nodes is the main session node, child nodes represent the individual populations from which the analysed traits are selected, the grandchild nodes represent the traits themselves, and the great-grandchild nodes represent the linkage groups that are analysed. For the permutation test there will also be a *Genome wide* node (as a sibling to the linkage group nodes) representing the genome wide results of the test. The main session node has a serial number and between brackets the abbreviation of the analysis ([Table 1](#)). All session details like parameters and population summary data are listed in the *Session Info* tabsheet.

Nonparametric mapping (Kruskal-Wallis analysis)

Nonparametric means that no assumptions are being made for the probability distribution(s) of the quantitative trait (after fitting the QTL genotype). For the nonparametric mapping method MapQTL uses the rank sum test of Kruskal-Wallis (see e.g. Lehmann, 1975, ch. 5); when a locus segregates in only two genotype classes, such as in a backcross, this test is equivalent to the two-sided Wilcoxon rank sum test. The test is performed on each locus separately, no use is being made of the linkage map other than for sorting the loci. An application is described by Van Ooijen et al. (1993).

The Kruskal-Wallis test can be regarded as the nonparametric equivalent of the one-way analysis of variance. The test ranks all individuals according to the quantitative trait, while it classifies them according to their marker genotype. A segregating QTL (with big effect) linked closely to the tested marker will result in large differences in average rank of the marker genotype classes. A test statistic based on the ranks in the genotype classes is calculated. For individuals in ties, i.e. several individuals have equal values of the quantitative trait, the average rank (midrank) is used, while for the test the statistic

Table 1. Analysis abbreviations

Code	Description
KW	nonparametric mapping, or Kruskal-Wallis analysis
IM	interval mapping
MQM	MQM mapping
rMQM	restricted MQM mapping
ACS	automatic cofactor selection
PT	permutation test

adjusted for ties is used (indicated by K^* , Lehmann, 1975, eqn. 5.11). For the genotype classification the usual genotype classes are used; when you wish to classify in another way, i.e. when there is dominance or for recombinant inbreds, it is possible to indicate another classification by adding a code to the locus in the loc-file (see [Table 9](#), p. 47).

Under the null-hypothesis, i.e. there is no segregating QTL (or perhaps better, the segregating QTL has no effect), the Kruskal-Wallis statistic is distributed approximately as a chi-square distribution with the number of genotype classes minus one as degrees of freedom (e.g. 1 degree for a backcross, 2 degrees for an F2). Since the test will generally be performed on many linked and unlinked loci, it is prudent to use a stringent significance level (P-value) for the individual tests in order to obtain an overall significance level of about 0.05; we suggest a level of at least 0.005.

The linkage group with a segregating QTL must reveal a gradient in the test statistic towards the locus with the closest linkage to the QTL. The power of the test depends on the degrees of freedom. So for instance, when codominant loci are combined with dominant loci, the latter may show a smaller significance level even if they are more closely linked. The power also depends on the number of individuals in the test. Because the analysis can only be done on individuals for which both marker genotype and quantitative trait value are known, differences between markers in numbers of individuals in the test will affect the gradient in the test statistic over the linkage group.

Nonparametric mapping output

A summary of the parameters input and the data read from the files are given on the Session Info tabsheet. The Results tabsheet lists the results under column headers with the following meaning (in alphabetical order):

<class>	the genotype class for which the details are given
Df	the degrees of freedom

Group	the linkage group of the locus
K*	the Kruskal-Wallis test statistic K*
Locus	the name of the locus at the current position
Mean-<class>	the arithmetic mean of the class
Meanrank-<class>	the mean rank of the class
Nr	sequential number of the row
Nr-<class>	the number of individuals in the class
Nr inf.	the number of informative individuals, i.e. the individuals with a genotype within the current classification and with a known quantitative trait value
Position	the current position on the map
Signif.	the significance level in asterisks (details are given on Session Info tabsheet)

Sporadically it may occur that there is just a single tie in the quantitative data (of course you can conclude here that there is no genetic effect); in this case it is impossible to calculate the statistic and the relevant cells of the table will stay empty. When a genotype class is empty, its mean and mean rank cannot be calculated, relevant cells will stay blank. When one or more genotypes are detected outside the current classification a warning is printed in the *Signif.* column in the form of "(?)".

Interval mapping

The interval mapping method was developed by Lander & Botstein (1989). The method is more extensively described in a paper by Van Ooijen (1992). In interval mapping a so-called *QTL likelihood map* is calculated, i.e. for each position on the genome (say every centiMorgan) the likelihood for the presence of a segregating QTL is determined (the *likelihood under the alternative hypothesis*, H1). At the same time the genetic effects of the QTL and the residual variance are calculated. This likelihood under H1 is compared to the likelihood for the situation when a locus with zero genetic effect would segregate, i.e. there is no segregating QTL (the *likelihood under the null-hypothesis*, H0). This comparison is done with a likelihood ratio statistic called the *LOD* (or *LOD score*), which is the 10-base logarithm of the quotient of the two respective likelihoods. When the LOD score exceeds the (predefined) significance threshold somewhere on a linkage group, a segregating QTL is detected; the position with the largest LOD on the linkage group is the estimated position of the QTL on the map. To obtain a (roughly) 95% confidence interval around this point estimate, a so-called *two-LOD support interval* must be constructed by taking the two positions left and right of the point estimate, that have a LOD value of two less than the maximum. In the [Mapping theory chapter](#) (p. 31) the choice of the significance threshold and other more technical details are discussed.

For the F2 and RIX population types MapQTL allows the fit of *dominance* of the QTL, but this may also be restricted so that the heterozygous QTL genotype is strictly intermediate. In advanced RIX generations the fit of dominance can be impossible due to

a complete lack of heterozygous marker genotypes. For an F2 the default is to fit dominance, for an RIX population it is no dominance.

The analysis of a [selectively genotyped](#) population (see the Mapping theory chapter, p. 35) is easily arranged by putting the data of the genotyped individuals in the top of the quantitative data file and those of the not-genotyped ones below. The top individuals should correspond to the individuals in the loc-file.

The calculation of the maximum likelihood is implemented in MapQTL as an iterative EM procedure. The iterations stop when the relative change in the logarithm of the likelihood is less than the so-called *functional tolerance value*, or when the *maximum number of iterations* is reached. These and some other options may be set in MapQTL. It is possible to print the test statistic as a [deviance](#) instead of a LOD score (see the Mapping theory chapter (p. 32) for a definition of deviance). Further, MapQTL uses the so-called *mapping step size* parameter to go from one position on the map to the next in between loci for the positions for which the LOD (or deviance) must be calculated; choose a large value if you only want computations on locus positions and not in between. For population type RIX it is possible to speed up computations using a QTL genotype probability approximation instead of using correct three point genotype probabilities; details are described in the [Mapping theory chapter](#) (p. 33). Finally, for population types F2, RIX and CP, the parameter called *maximum number of neighbouring markers used* can be modified. This parameter is important in the calculation of the QTL genotype probabilities based on the marker genotypes, when the markers have a dominant genotype or a not fully informative segregation type; details are described in the [Mapping theory chapter](#) (p. 31). These are all parameters that can be set with the *Analysis Options* of the *Options* menu.

Interval mapping output

A summary of the parameters input and the data read from the files are given on the Session Info tabsheet. The presented population variance is the usual ML estimate; to get the unbiased variance this is multiplied by $n/(n-1)$, with n being the number of individuals. The population skewness and kurtosis are the coefficients of skewness and kurtosis, denoted as g_1 and g_2 , respectively, by Snedecor & Cochran (1980, sec. 5.13, 5.14). The Results tabsheet lists the results for all fitted QTL positions under the following headers (in alphabetical order; see the [Data files chapter](#) (p. 39) for the genotype codes):

Additive	the estimated additive effect: BC1: $\mu_A - \mu_H$ or: $\mu_H - \mu_B$ F2, RIX, HAP1 or DH1: $(\mu_A - \mu_B)/2$ HAP or DH: $(\mu_{A\{0\}} - \mu_{B\{0\}})/2$
Deviance	the deviance
Dominance	the estimated dominance effect (F2 or RIX):

```

mu_H = (mu_A + mu_B)/2
when dominance could not be fitted it is given as "0.0"
% Expl. the percentage of the variance explained for by the QTL:
100*(H0_var - var)/population variance),
in which H0_var is the residual variance under the current null
hypothesis (depends on cofactors used in MQM mapping)
GIC genotypic information coefficient (see Genotypic information
coefficient section of Mapping theory chapter, p. 33)
GIC_1 genotypic information coefficient for the first parent
GIC_2 genotypic information coefficient for the second parent
GIC_m mean of GIC_1 and GIC_2
Group the linkage group of the locus
# Iter. the number of iterations needed to reach the tolerance criterium; when
this number is followed by an asterisk, the maximum number of
iterations was reached without satisfying the tolerance
Locus the name of the locus at the current position
LOD the LOD score
Nr sequential number of the row
Position the current position on the map
mu_A the estimated mean of the distribution of the quantitative trait
associated with the "a"-genotype
mu_B idem for the "b"-genotype
mu_H idem for the "h"-genotype
when no dominance was fitted:
mu_H = (mu_A + mu_B)/2
mu_A{0} the mean associated with the "a"-genotype with phase type {0} or with
the "b"-genotype with phase type {1}
mu_B{0} idem for the "b"-genotype with phase type {0} or for the
"a"-genotype with phase type {1}
mu_ac{00} idem for the "ac"-genotype with phase type {00}
mu_ad{00} idem for the "ad"-genotype with phase type {00}
mu_bc{00} idem for the "bc"-genotype with phase type {00}
mu_bd{00} idem for the "bd"-genotype with phase type {00}
Var the residual variance after fitting the QTL

```

MQM mapping

The MQM mapping method, based on multiple-QTL models, was developed by Jansen (1993, 1994) and Jansen & Stam (1994). Although the definition of MQM mapping is very wide, the current implementation in MapQTL is limited to using markers as cofactors in an approximate multiple-QTL model with additive and dominant gene actions only. Other uses of marker cofactors, such as with gene-by-environment (Jansen et al, 1995) or gene-by-gene (Fijneman et al, 1996) interactions, and the inclusion of the experimental design are quite difficult to implement in an easy, user friendly manner in a general purpose mapping program.

Using a true multiple-QTL model to detect and map QTLs would mean a multi-dimensional search over the linkage groups. At present, this is computationally not really feasible. The suggested approach is to first look for putative QTLs, either by multiple regression (preferably using backward elimination) or by using interval mapping (i.e. a single-QTL model). Care must be taken not to pick up so-called *ghost QTLs* (Martínez & Curnow, 1992). Next, close to detected QTLs markers are selected as cofactors to take over the role of the nearby QTLs in the approximate multiple-QTL models used in the

subsequent MQM mapping. With this MQM mapping a one-dimensional search over the genome is done by testing for a single segregating QTL as in interval mapping, while simultaneously fitting the selected cofactors, both under H0 and under H1. Thus, the cofactors will reduce the residual variance. If a QTL explains a large proportion of the total variance, then the use of a linked marker as cofactor in subsequent MQM mapping will importantly enhance the power in the search for other segregating QTLs. After the first attempt of MQM mapping it is possible that the most likely positions of some QTLs are different from those in the cofactor selection phase, after all, the power is enhanced. In such cases one should adjust the selection of cofactors and redo the MQM mapping. Sometimes even more of these rounds will be necessary to obtain the best possible final solution.

MapQTL offers two options in MQM mapping. The first, called *restricted MQM mapping*, is to use all cofactor markers except the ones on the linkage group the QTL is fitted on. The second option, just called *MQM mapping*, is to use all indicated cofactor markers; in this method a cofactor is temporarily excluded (from the H0 and H1 models) when it is one of the flanking markers of the interval on which the QTL is fitted. This means that in moving through the map the set of cofactors included in the model will change, and hence the H0 needs to be recalculated on change of the set of cofactors (because it has become a different H0 model). At the start of each linkage group the H0 model is always (re)calculated. The set of cofactors for the selected traits must be chosen using the Map Info tabsheet (see the [Contents-and-results panel section](#), p. 10).

In contrast to interval mapping, in MQM mapping dominance is always fitted (also for cofactors) for an F2 population, whereas it is never fitted for an RIx population (an F2 may be analysed as an RI2 when no dominance is required). Unmapped loci cannot be used in the analysis. Selective genotyping is not possible.

MQM mapping output

The output is similar to that of interval mapping, so please see the [Interval mapping output section](#) (p. 15). Of course, some additional information is given. Loci used as cofactor are indicated with an "X" in the additional *Cofactor* column on the Results tabsheet, while they are listed with group and position information on the Session Info tabsheet. This tabsheet also gives the name of the so-called *cofactor monitor* output file, this plain text file resides in the project directory. The file lists at each calculated map position the estimated values of the regressors for each cofactor and of the means associated with the QTL genotypes; at the map positions where the H0 is calculated, the output lists the estimates of the cofactor regressors and the overall mean (see the [MQM mapping section](#) (p. 35) of the Mapping theory chapter for details). The locus names are printed at corresponding map positions at the end of the line, and cofactors are indicated with an exclamation point. On the Session Info tabsheet the information is given for the

successive H0 models that were fitted: (a) the locus name at the position for which the H0 applies, (b) the ln-likelihood ($\ln = \log$), (c) the number of iterations, (d) the residual variance (after fitting the overall mean plus the cofactors), and (e) the variance explained with this H0 model.

Automatic selection of cofactors

MapQTL offers the possibility for automatic selection of cofactors. The analysis is based on backward elimination. Starting with the set of cofactor loci that are selected by the user, for instance four on each linkage group, a standard MQM model is fitted that includes these cofactors and excludes the QTL (the starting set of cofactors for the selected traits must be chosen from the Map Info tabsheet). As such it is equal to a null-hypothesis model in MQM mapping. Subsequently, by leaving out one cofactor locus at a time subsets of loci are created, for which the corresponding models are fitted. The likelihoods of each of these subset models (there are as many as there are loci in the starting set) are compared to the likelihood of the full model with all cofactor loci. The subset of which the model caused the smallest change in likelihood, is chosen as the starting set for a subsequent round of elimination, in which new subsets containing all but one locus are modelled and tested. The process stops when the change in likelihood is significant according to the P-value for the test (can be modified with the *Analysis Options* of the *Options* menu), or when there is no more cofactor locus remaining in the set.

The test statistic used for the comparison of subset models with the full model is the [deviance](#) (see the Mapping theory chapter, p. 32). The deviance is assumed to follow a chi-square distribution. The degrees of freedom is the number of regressors per cofactor, which is one for population types BC1, HAP1, HAP, DH1, DH, and RIX, two for F2, and three for CP. However, when the number of parameters in the model (one for the overall mean plus one (BC1, etc.), two (F2) or three (CP) for each cofactor locus) is large, the estimate of the residual variance will be biased, and as a result the assumption of the deviance following a chi-square distribution will be violated. Therefore, the number of parameters in the model should not be too large, preferably less than twice the square root of the number of individuals in the population (Jansen & Stam, 1994). A warning is issued by the software when this is not the case. Often, though, this warning can be ignored, because in such situations the difference in the likelihood of the full model with that of the subset model can be so small, that even with a violation of the chi-square assumption the test will most probably be not significant. When the number of parameters is so large that it leaves twenty or less degrees of freedom for the estimation of the residual variance, then the automatic selection algorithm is not executed.

The Results tabsheet presents the entire procedure of backward elimination. The final selection of cofactors is saved in a cofactors file in the project data directory under a name given at the end of the Results tabsheet. This final set of cofactors can be used in subsequent MQM analysis and can be loaded from this cofactors file using the Cofactors tool. In MQM analysis the QTL likelihood map is studied, in which it may be seen that sometimes QTLs are fitted somewhat distant to the closest cofactor in the set. Then it can be a good idea to modify the set of cofactor loci, replace the cofactor in the set with one that is closer to the maximum in the QTL likelihood map, and redo the MQM analysis. Possibly this needs to be done a few times. In the end you would like to finish with a set of cofactor loci that are the loci closest the significant maxima in the QTL likelihood map.

In many cases there are insufficient degrees of freedom and/or RAM memory to accommodate all or many loci in the starting set of the automatic selection procedure; this is generally due to a larger number of missing observations, dominance or less informative segregation types in CP populations. One can think of several approaches where the procedure can still be used in an adapted way. For instance, one could start with interval mapping to find areas with higher LOD scores, and use a subset of loci in those regions as starting set. Another option, possibly combined with the previous, is to start of with one linkage group, select a locus every 10 or 20 cM for the starting set, perform the automatic cofactor selection, and then go forward to the next linkage group while keeping the resulting final selection of the previous linkage group (easily done with the cofactors file and the Cofactors tool), and so on until all linkage groups have been done.

Permutation test

In order to determine the significance threshold of the LOD score or the deviance, it is possible to use the permutation test. This is a resampling method to obtain empirical significance threshold values (Churchill & Doerge, 1994). MapQTL offers this method for interval mapping. Over a set of iterations (in this case: permutations) the frequency distribution of the maximum LOD score (or deviance) is determined. In each iteration the quantitative trait data are permuted (i.e. sampled without replacement) over the individuals while the marker data remain fixed. Subsequently, interval mapping is done on the thus obtained data set. The maximum LOD score (or deviance) over each linkage group as well as over all linkage groups (the genome) is observed in each iteration. After a large set of iterations (at least 1,000 but preferably 10,000 or more) an estimate of the frequency distribution of the maximum test statistic (LOD or deviance) under the null-hypothesis (no QTL) is obtained. The number of permutations can be set with the *Analysis Options* of the *Options* menu. The results are presented as frequency tables

(absolute, relative and cumulative) in the Results tabsheet; the frequency tables are given per linkage group and genome wide, genome wide meaning over the set of analysed (i.e. selected) groups as a whole.

To determine the significance threshold one first has to decide upon what P-value to use, and whether or not to use individual thresholds per linkage group or to use the genome wide threshold. Next do the permutation test for the required linkage groups. In general, for standard applications the genome wide (including all groups) threshold with a P-value 0.05 (or 5%) is required. This means that we have to find the interval in the results of the permutation test where the relative cumulative count is $1-0.05=0.95$ and take its upper boundary (which is given as the value under the header *Interval*) as the significance threshold value to use. As the exact relative cumulative count 0.95 is not always present, the first higher value that is realised must be taken, merely to be on the safe side.

An alternative method of getting the significance threshold is described by Van Ooijen (1999); often this method gives very similar answers as to which threshold value to use.

Permutation test output

A summary of the parameters input and the data read from the files are given on the Session Info tabsheet. The Results tabsheet lists the frequency distributions of the LOD score (or deviance) for all analysed (selected) linkage groups as well as the genome wide frequency distribution. The distributions are given for intervals of 0.1 LOD units or 0.5 deviance units in size, starting from 0.0 upto the value to accommodate for largest value that came about. The total number of permutations is corrected for the numbers of cases where singularity or perfect fit occurred. The frequencies are presented as interval counts, cumulative counts, relative counts and relative cumulative counts under the following headers (in logical order):

Group	the linkage group; if this shows "GW" these are the genome wide results
Interval	the upper (exclusive) boundary of the interval into which a single permutation result can be classified; the lower (inclusive) boundary is given by the previous value in the table (e.g. the interval labelled "1.1" ranges from 1.0 (inclusive) to 1.1 (exclusive))
Count	the absolute count: the number of permutations that had a result (i.e. a maximum LOD score or deviance) in the interval as defined above
Cum.count	the cumulative count: the number of permutations that had a result in the interval or in any lower value interval
Rel.count	the relative count: the absolute count divided by the total number of permutations
Rel.cum.count	the relative cumulative count: the cumulative count divided by the total number of permutations

Tutorial

In this tutorial you will be taken through the most important steps of a QTL mapping project using a simulated data set of an F2 population that has several segregating QTLs, including two linked QTLs.

The first thing to do after starting MapQTL is to create a new project:

- Use the *New Project* function from the *File* menu.
- You will get a dialog in which you are prompted for a file name under which to save the project; this file name is also used for the project subdirectory name; if necessary change the directory where the dialog is pointing to, and enter *tutorial* in the dialog's *File name* field.
- Click on the *Save* button; this will create your project file *tutorial.mqp*, and in addition the project directory *tutorial.mqd*, which will contain all internal files of MapQTL for this project; a new project is just a new workspace to store results.

Now you have a new project, you can load the basic data: the marker scores, the map and the quantitative data:

- Make sure the *Populations* tabsheet is shown in the *navigation* panel and use the *Load Data* function from the *File* menu.
- A dialog will prompt you for a quantitative data or locus genotype file; go to the DemoData directory, which is a subdirectory of the program directory (typically C:\Program Files\MapQTL5) and find the locus genotype file *DemoF2.loc* and click on the *Open* button.
- Next you will be prompted for the name of the population that it should be stored under; just click *OK* for the default *DemoF2*.
- Notice that on the *Populations* tabsheet a *DemoF2* population node is created, with a child node *Genotypes*.
- To load the quantitative data click on the *Load Data* tool bar button  and select the file *DemoF2.qua* (in the DemoData directory).
- Next you will be prompted for the name of the population that it should be stored under; just click *OK* for the default *DemoF2*; the navigation panel will resemble [Figure 2](#); notice the population DemoF2 with two numerical traits, *nr* and *qtrait*, and with *Genotypes*; also notice the common traits node, which is only useful when there are more populations in a project.

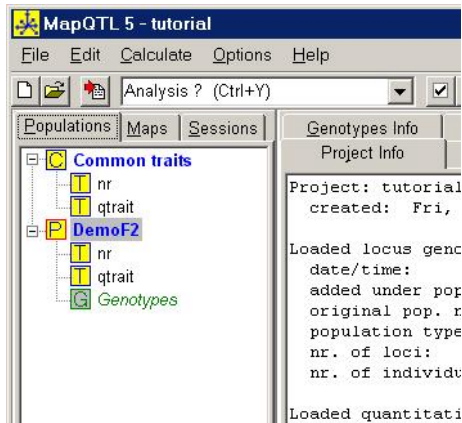


Figure 2. *Populations* tabsheet with the DemoF2 population

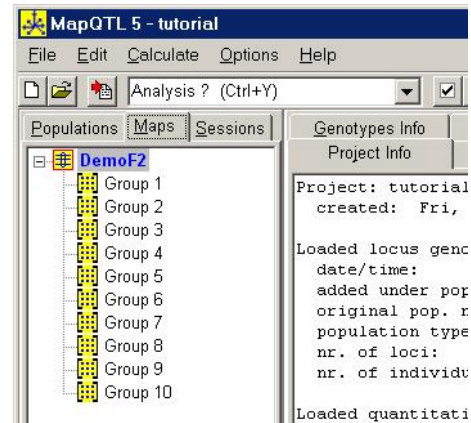


Figure 3. *Maps* tabsheet with the DemoF2 map

- Select the *Maps* tabsheet, click on the *Load Data* button  and select the map file *DemoF2.map*; the navigation panel will resemble [Figure 3](#), notice there are ten linkage groups.

The project *tutorial* now has the basic data loaded. Let's have a look at the *contents-and-results* panel and see what the contents of the data are.

- First, click on the *DemoF2* nodes in the *Populations* panel and the *Maps* panel, just to make sure the population node and map node are selected in the treeviews.
- Select the *Project Info* tabsheet. Here you can see when the project was created and what data sets were loaded, including a summary of the data sets.
- Select the *Project Notes* tabsheet. This tabsheet is empty; when you click in it, you can start entering notes; these will be stored with the project.
- Select the *Population Info* tabsheet. This tabsheet shows a summary of the population currently selected (i.e. DemoF2).
- Select the *Traits Info* tabsheet. This tabsheet has a table with all numerical traits data (of the currently selected population), each trait in its own column. The table includes a first column *Nr* for the original row number in the table. (There is also a numerical trait with the name *nr*; this was entered in the original quantitative data file as the individual number, of which the data happen to coincide here with the row number *Nr*.)
- Any table in MapQTL can be sorted by clicking on the column header. Try this by clicking on the *qtrait* column header; click again and notice that the rows are sorted in the opposite direction; notice that the largest *qtrait* value is 5.744531. To return to the original order, click on the *Nr* column header.

- Select the *Genotypes Info* tabsheet. This tabsheet shows a copy of the loaded loc-file with the marker names and genotype scores for all individuals and markers.
- Select the *Map Info* tabsheet. This tabsheet displays the details of the loaded map-file: the locus (marker) name, its linkage group name (usually a number) and its map position. As in any table in MapQTL, there is a column *Nr* with the original row number. Notice there is a column with check boxes; this will be used later for indicating which markers are to be used as cofactors. Also notice a column filled with X's under the header of the population name DemoF2; an X indicates if a marker of the map is present in the genotypes of the population; when loading more populations this is especially useful to see in which populations each marker is determined.
- Select the *Maps* tabsheet on the navigation panel. Click on the *Group 4* node in the treeview and observe that on the *Map Info* tabsheet the row pointer moves to the beginning of this group in the map table. This navigation feature works for each group node of the treeview.

Select one by one the remaining three tabsheets and observe that they are empty; these are for showing the results of analyses, which you haven't done yet. After having looked for a little while at the user interface of MapQTL you may wish to modify the fonts your copy of MapQTL is using. This is possible by selecting the *Environments Options* of the *Options* menu. Do this and pick the fonts and font sizes of your preference that will be used for the various elements of the user interface. By clicking the *OK* button the current choice is saved with this project. If you wish to use this choice for any future MapQTL project, press the *Save as default* button.

You are now ready to start doing analyses. Let's do interval mapping for the trait *qtrait* on all linkage groups.

- Click in the *Analysis* selector and pick *Interval Mapping*.
- Select the *Maps* tabsheet on the navigation panel, and right-click on the map node *DemoF2*. The result will be that all linkage group nodes will be highlighted with a red background. This is a toggle: right-clicking again removes the red highlighting. This *special* selection may also be done by pressing the keyboard space bar; try this. The specially selected linkage groups will be used in the analysis.
- Select the *Populations* tabsheet on the navigation panel, and right-click on the trait node *qtrait*. You may also click in the treeview area, use the keyboard arrow keys to navigate to the *qtrait* node, and then press the space bar, to *specially* select this node; try this.
- If the trait and the linkage groups are *specially* selected (i.e. highlighted in red), and interval mapping is selected in the analysis selector, the calculate function is enabled; notice that the *Calculate* button  and the *Calculate* option in the *Calculate* menu are enabled (i.e. not greyed out).
- Click on the *Calculate* button.

- Observe that the *Sessions* tabsheet is automatically selected and gets filled with various session nodes, and that the progress bar on the status bar gradually proceeds while the calculations are being performed.
- Once the calculations are finished, the *Results* tabsheet is automatically selected and filled with the outcomes of the analysis. Inspect these results. Similar to the *Map Info* tabsheet you can navigate through the results using the linkage group nodes, but here using those in the *Sessions* treeview. If the group nodes are not visible, click on the + symbol before the *qtrait* node in the *Sessions* treeview. Try this.
- Take a look at the *Session Info* tabsheet, to see what parameter settings and data were used in the current calculation session.
- Studying the results as charts is possible with the *Results Charts* tabsheet; select this tabsheet ([Figure 4](#)). Notice that it consists of two subordinate tabsheets, one for the control of the charts and another for the charts themselves (they can be selected with keyboard combinations ctrl+O and ctrl+H, respectively; try this).
- Select the subordinate *Charts* tabsheet. Each linkage group has a separate chart in which the LOD score is plotted against the map. By default the axes are rounded upward to "natural" values (depends on font and screen size and resolution), but the data are plotted up to their largest values.
- Select the subordinate *Control* tabsheet. There are many options to control what is plotted and how it is plotted. In the upper part there are three checklists, one labelled *Groups* to choose which linkage groups must be plotted, two labelled *Left Y-axis* and *Right Y-axis*, respectively, to control which data should be plotted against the corresponding Y-axis. In the lower part there are three tabsheets full with options that allow you to set the charts to your preference.

Before you go on, you will need to know what LOD value is significant. You can use the formula and tables in the paper by Van Ooijen (1999). For this you need the following: the current map (DemoF2) has ten linkage groups ($n=10$) and an average chromosome map length of 98.6 cM; the population type is an F2; you used the MapQTL option to have unrestricted dominance (see the *Analysis Options* under the *Options* menu), therefore you will need Table 2 in the above paper. Verify that for the standard genome wide significance of 5% ($\alpha_g = 0.05$) the formula and tables of the paper give you the value of 3.8 as the LOD significance threshold. This means, more formally, the probability that the LOD score is above this threshold value just by chance (rather than by a segregating QTL) anywhere on the genome is 5%. More practically, this means that you will conclude that a QTL is present when the LOD is above the value 3.8.

Another way of getting the significance threshold is to do the permutation test. This is often thought of to be more correct, because the method in the paper by Van Ooijen (1999) is based on the assumption that the trait is distributed according to the normal distribution and this might not be true for the data you are analysing (however, it is true

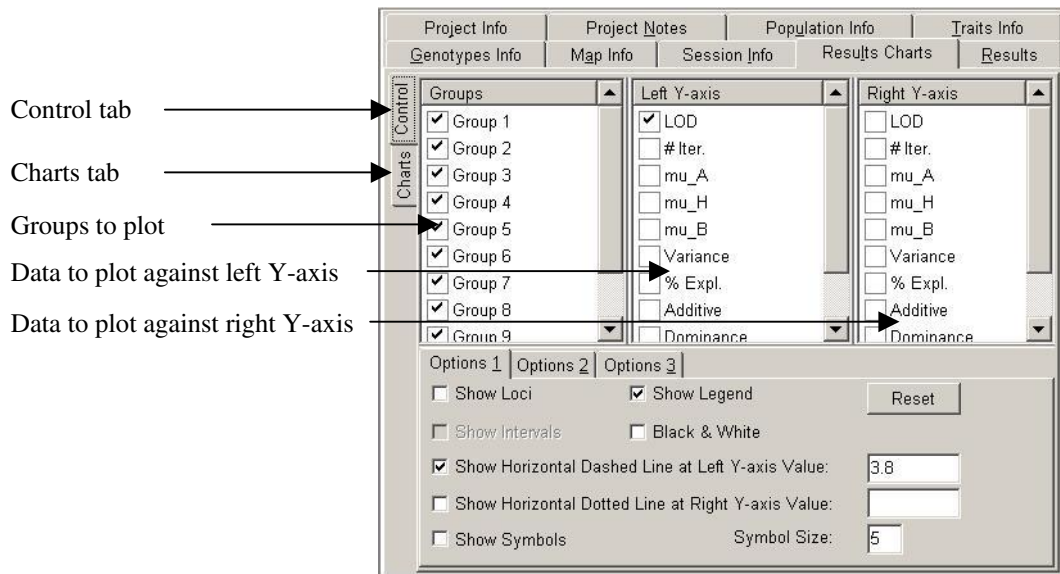


Figure 4. Results Charts tabsheet with subordinate Control tabsheet visible

here for the simulated DemoF2 data set). In the permutation test the significance threshold is determined on the actual data: each iteration the quantitative trait values are permuted over the individuals, thereby releasing any possible association with the markers. Subsequently the permuted data are analysed by interval mapping and the maximum LOD scores are recorded. By doing this repeatedly, preferably at least 10.000 times (because the results are quite variable), the frequency distribution of the LOD is determined based on the actual data of which we are certain that there is not any association between any segregating QTL and a marker (due to the permutations).

Try the permutation test, but because it takes so long for all computations, first set the number of permutations for this time to the low value of 100:

- Use the *Analysis Options* of the *Options* menu.
- Set the number of permutations to 100, and click *OK*.
- Select the *Permutation Test* in the *Analysis* selector.
- Verify that the trait *qtrait* and the ten linkage groups are still *specially* selected (if not restore this) and subsequently click on the *Calculate* button. Notice that new session nodes are created. The calculations will take some time to complete.
- Inspect the results in the *Results* tabsheet. Look up the interval value in the group *GW* at the relative cumulative value (*Rel.cum.count* column) of 0.9500 (or a value close). See the [Permutation test section](#) (p. 19) for details on how to deal with these results. Redo this permutation test a few times and make notes of each estimated threshold

value. You will notice that it varies around the 3.8 value calculated above, which is a correct value because the DemoF2 data are based upon a simulated normal distribution and thus agrees with the required assumption of normality.

We decide we want to use the 3.8 LOD as a genome wide significance threshold. Now let's go back to the interval mapping results charts.

- Select the *Control* tabsheet. On the *Options 1* tabsheet, enter the above threshold value 3.8 in the field *Show Horizontal Dotted Line at Left Y-axis Value*.
- Check the *Show Loci* option on the *Options 1* tabsheet.
- Select the *Charts* tabsheet again, and notice that it is easy to spot the regions with a significant LOD score: groups 1, 3, 4 and 5 have significant scores, group 2 is just below the threshold (on the *Results* tabsheet significant it shows a maximum LOD of 3.43 at locus m34).
- Find the markers closest to the highest LOD scores on these groups 1, 3, 4 and 5, and verify this using the *Results* tabsheets. These are m9, m54, m75 and m97. Also take notice of the *explained variance (% Expl.)* at these markers.

In interval mapping the association at a certain map position is tested against the residual variance: the larger the genetic effect associated with a position is in relation to the residual variance, the more significant is the test. However, when several QTL are segregating, some of the residual variance will be determined by the other segregating QTLs. If we could take these QTLs into account while testing for a QTL at a certain position, then the residual variance would be reduced and as a consequence the test would become more powerful. This is achieved by taking the markers that we think are associated with a QTL as cofactors in the so-called *MQM Mapping* analysis (also called *composite interval mapping*):

- Select the *Map Info* tabsheet.
- Check the boxes in the *Cofactor* column for markers m9, m54, m75 and m97.
- Set the *Analysis* selector on *MQM Mapping*.
- Verify that the trait *qtrait* and the ten linkage groups are still *pecially* selected (if not restore this) and subsequently click on the *Calculate* button.

Now inspect the results of MQM mapping using the charts; the LOD significance threshold can be taken as the same value 3.8, use this value for the horizontal dashed line in the charts; check *Cofactors* in the *Left Y-axis* list on the *Control* tabsheet. Find the linkage groups with a value above this threshold. These should be groups 1, 2, 3, 4 and 5; group 2 is now also included, with a very significant LOD score as well! Notice that all LODs are much larger values than with interval mapping: because there (apparently) are QTLs with a larger explained variance the power of the analysis has increased by taking the closest markers as cofactors with the MQM mapping analysis.

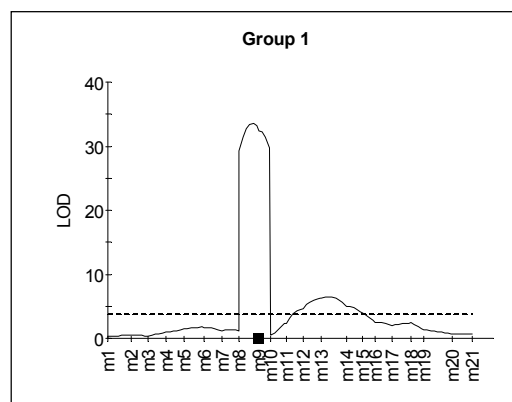
Subsequently you can try to improve the results. Examine all five linkage groups and find the marker closest to the LOD peak and modify the set of cofactors correspondingly. The set of cofactor markers should now be: m9, m33, m53, m75 and m96. Redo the MQM mapping analysis. When you inspect the results, you will notice that the chosen cofactor markers are now still closest to the current LOD peaks; that is what you would want to have as a final result, however...

There is however one important aspect that you must see: on linkage group 1 there are LOD values above the significance threshold some distance away from the cofactor marker m9 ([Figure 5](#)). This is an indication that the cofactor m9 is a distance away from the real QTL position or there may be more QTLs on this group. In order to study how many and where these QTLs are, you can use the automatic cofactor selection procedure in combination with the cofactor markers already determined on groups 2 to 5:

- Select the *Map Info* tabsheet.
- Verify that only m9, m33, m53, m75 and m96 are checked in the *Cofactor* column.
- Click on the *Cofactors Tool* button ☒.
- Check *Group 1* in the list *Act on checked groups*; select the action *Check all loci on indicated groups*; click on the *Do it!* button, and close the *Cofactors Tool*.
- Verify that now all loci of group 1 are checked as cofactor, and further only m33, m53, m75 and m96.
- Set the *Analysis* selector on *Automatic Cofactor Selection* and click on the *Calculate* button.

Automatic cofactor selection uses a backward elimination procedure to see which markers show a significant association and which do not; all not-significant markers are removed, so you end up with only significant cofactor markers. When you inspect the results, you will see that on group 1 of all markers the two markers m9 and m13 remain as being significant, while the cofactor markers on the other linkage groups also remain as significant. Now you would like to redo the MQM mapping analysis, but with the

Figure 5. LOD profile on linkage group 1 with LOD values above the significance threshold (3.8) some distance away from the cofactor marker m9



present set of cofactors:

- Make a note from the Results tabsheet of the file name under which the selected set of cofactors is stored;
(it should be something like: "Session 5 (ACS)_DemoF2_qtrait.cof", and it should reside in the project directory).
- Click on the *Cofactors Tool* button .
- Select the action *Load cofactor setting from file*, and click on the *Do it!* button.
- You are asked whether you wish to clear the currently checked loci; choose *Yes*.
- Subsequently, you will get a dialog in which you are prompted for a cofactors file; point the dialog to the project directory, pick the proper cofactors file, click on *Open*, and close the *Cofactors Tool*.
- Verify that the boxes for m9, m13, m33, m53, m75 and m96 are checked; this can be done easily after sorting the *Cofactors* column: click on the column header and all checked loci will be together on top (or bottom).
- Set the *Analysis* selector on *MQM Mapping*, and click on the *Calculate* button.

When you inspect the new results, you will notice that nowhere outside the intervals flanking the chosen cofactor markers are LODs significant, and that the cofactor markers are at the highest LOD positions. On group 1 you have detected two (significant) QTLs about 20 cM apart. This was only due to the fact that we discovered significant LODs outside the region where another QTL was detected and we decided to try automatic cofactor selection starting with all markers on group 1. These high LODs just outside a QTL region are caused by the fact that the second QTL has a large genetic effect while the preselected cofactor m9 was a fair distance away. In other circumstances we might have missed out on the second QTL using the approach we have taken here. For instance, if we would have used marker m11 instead of m9 we would not have observed the significant LODs outside the region of m11. This would be an example of a *ghost QTL*: one non-existing QTL would be detected in the middle of the two real QTLs both having genetic effects acting in the same direction (often called QTLs in coupling phase).

The approach we have followed up to the automatic cofactor selection can be seen as a *forward selection* procedure: do interval mapping and fix significant regions using markers as cofactors. As we have seen, a drawback of the approach is the possibility to obtain ghost QTLs, but it also has the possibility to miss out on linked QTLs with counteracting genetic effects, i.e. QTLs in repulsion phase (not present in the DemoF2 example). Contrarily, the automatic cofactor selection procedure with its *backward elimination* does have the potential to discover linked QTLs in coupling and in repulsion. Therefore, a more systematic approach employing automatic cofactor selection is to be recommended in order not to make any mistakes with linked QTLs if these happen to be present.

One type of a more systematic approach could be to do forward selection with interval mapping followed by automatic cofactor selection starting with the cofactor markers at the detected significant LOD peaks as a fixed set, extended with all markers on a single linkage group (or a subset at a certain distance), and do this for each linkage group. When nothing new is detected you are finished, but when new (linked) QTLs are detected you should start over again because you have increased in power due to the newly detected QTL.

Another type of a more systematic approach would be to start off right from the beginning with automatic cofactor selection. Due to missing genetic information (missing marker scores, dominant scores, low information segregation types) and usually limited population size, there are often insufficient degrees of freedom and memory (RAM) to accommodate for all or many loci in the starting set of the automatic selection procedure; this can be circumvented by doing linkage groups one by one and fixing the detected results. Let's try this approach with the current DemoF2 data set:

1. Verify that the trait *qtrait* and the ten linkage groups are still *specially* selected, and set the *Analysis* selector on *Automatic Cofactor Selection*.
2. Click on the *Cofactors Tool* button ☒.
3. Select the action *Clear all loci*, and click on the *Do it!* button.
4. Check *Group 1* in the list *Act on checked groups*; select the action *Check all loci on indicated groups*; click on the *Do it!* button, and close the *Cofactors Tool*. Now you have checked all loci on group 1 and no other loci; verify this.
(Remark: In the DemoF2 population the information is sufficient to do automatic cofactor selection with all loci of a group as a starting set. Whenever your own data set have insufficient information you will receive error messages about insufficient memory, or the calculations proceed extremely slow. In such cases you should reduce the number of selected loci for the starting set by picking loci say every 20 cM. If subsequently a significant locus is detected you may try to improve the result by using a starting set of just a few loci neighbouring the significant locus.)
5. Click on the *Calculate* button.
6. When finished, load the resulting set of cofactors from the file using the *Cofactors Tool* button while clearing the currently checked loci prior to the loading.
7. Repeat the steps 4 to 6, each time going to the next linkage group until all groups are done. This way all groups are searched one by one while retaining the significant loci. When significant loci *are* detected, the analysis gains in power along the way. Therefore, theoretically you should go back to group 1 again and keep on repeating steps 4 to 6 until nothing changes anymore.
8. However, it is advisable to do a round of MQM mapping first, because that will reveal some false positive cofactors since the P-value is set to a non-stringent value (0.02) as not to miss out on linked QTLs. The set of cofactors that you should have found at this point is: m9, m13, m32, m49 (instead of m53 !), m75, m96, these are nearly all the same as found in the forward selection approach above, and some additional loci on

groups 7, 9 and 10: m145, m146, m172, m173, m180, m183, m193.

9. Load this set of cofactors and do MQM mapping. Plot the cofactors and draw the significance line at 3.8 LOD, and inspect the results charts.
10. The LOD scores on groups 7, 9 and 10 are all below the significance threshold, therefore we can remove their cofactor loci as being false positives. Do this, redo the MQM mapping, and inspect the results. You will see that on none of the groups 7, 9 and 10 significant LOD scores are computed, so the loci can indeed be regarded as false positives.
11. Another fact that you should observe is the significant region some distance away from cofactor m49 on group 3. Similarly to what we did to tackle the comparable problem of [Figure 5](#), select all markers on this group and do automatic cofactor selection. The result will be that m49 will be swapped for m53.
12. Now we should return to step 7, and redo groups 1 to 5, to try to see if the results improve or change. Actually, the previous step 11 may well have been a part of this. In fact groups 6 to 10 should also be redone. If you do this you will notice that several false positives will emerge which may be unmasked with MQM mapping as in step 10. The final result should have the following set of cofactor markers: m9, m13, m33 (instead of m32), m53, m75, m96; the MQM results with this set you have already obtained above.

This result is identical to what we have found earlier above, but now with a more systematic approach. If you open the DemoF2.loc file (in the DemoData directory) with *Notepad*, you will see at what positions QTLs were located in the simulation: in all cases the detected cofactor markers are one of the markers flanking the QTL, which is the best we could have found.

As a final remark, the DemoF2 data set is a nice simulated data set where all markers are scored, none is missing, the scores are 100% correct, the map is accurate, and there are 6 segregating QTLs with quite large effects. In real life you will have to do with marker scores that contain (unknown) errors, you will have missing marker scores, and as a consequence some uncertainty or even errors in your linkage map. The result will be that the QTL analysis will not be as straightforward as in this tutorial. With MapQTL 5 you have quite a powerful tool to analyse the data that you have obtained from your experiments; the software cannot, however, improve the quality of its input data, that area remains your responsibility.

Mapping theory

Interval mapping

The implemented QTL mapping procedure is a maximum likelihood approach to the segregation of a mixture of probability distributions (cf. Titterton et al, 1985; McLachlan & Basford, 1988). Under the hypothesis that a single QTL is segregating (thus, a single-QTL model), the mixture consists of Q distributions (components), one for each QTL genotype $q = 1 \dots Q$, with Q depending on the type of population (CP: $Q = 4$; F2, RIX: $Q = 3$; the other types: $Q = 2$). The component distributions are assumed to be normal, with means μ_q and common variance σ^2 . The mapping function of Haldane is assumed, which means that recombination events are mutually independent. Map distances between markers are taken as fixed.

For a given (assumed) position on the genome of a segregating QTL the probabilities π_q for each of the Q components are calculated for each individual in the population from the marker genotypes and the linkage map. If, for instance, the current QTL position on the genome coincides with a marker and the marker genotype of the individual is completely known, then one of the q component probabilities π_q equals 1 while the others are 0. If, as an instance of another extreme, the current position is in between markers while the genotypes of these two so-called flanking markers of the individual are not completely known (such as c in an F2, or hk of segregation type $\langle hkhkhk \rangle$ in a CP), then the genotypes of the linked markers beyond the flanking markers and their map positions are used to obtain the probabilities. Thus, genetic information from markers surrounding the current assumed QTL map position is used to calculate the most accurate values of the component probabilities.

In the mixture model the mixture density $f(x_n)$ for individual $n = 1 \dots N$ is the sum of the products of the component densities $f_q(x_n)$ with their probabilities π_q :

$$f(x_n) = \sum_{q=1}^Q \pi_q f_q(x_n),$$

thus, the likelihood for the population under the hypothesis that a QTL is segregating, L_1 , is:

$$L_1 = \prod_{n=1}^N f(x_n) = \prod_{n=1}^N \sum_{q=1}^Q \pi_q f_q(x_n).$$

With the EM algorithm, the likelihood, or actually its logarithm, is maximised and the parameters μ_q and σ^2 are estimated (Dempster et al, 1977). The EM algorithm is an iterative procedure, in which at each iteration the logarithm of the likelihood (the log-likelihood function) is calculated. The iterations stop once the relative (!) change in the log-likelihood function has become smaller than the so-called *functional tolerance value*, which means that the algorithm has converged. To prevent endless iterations in the case of non-convergence the iterations stop when the number of iterations has reached the *maximum number of iterations*. The functional tolerance value and the maximum number of iterations are parameters that can be set, although their default values usually suffice.

The L_1 likelihood is compared to the likelihood under the null-hypothesis, L_0 , which is similar to L_1 except that there is just a single component in the mixture, $Q = 1$, i.e. no QTL is assumed to be segregating. The comparison is done using the so-called *LOD score* ($\text{LOD} = \log$ of odds; Barnard, 1949) as a test statistic:

$$\text{LOD} = {}^{10}\log (L_1 / L_0) .$$

Some prefer the equivalent test statistic called the *deviance*, which uses the natural instead of the 10-base logarithm (the deviance is also called the *likelihood ratio test statistic*):

$$D = 2 * {}^e\log (L_1 / L_0) \approx 4.605 * \text{LOD} \approx \text{LOD} / 0.21715 .$$

The test statistic can be compared to a significance threshold to decide upon presence or absence of a QTL (see the [LOD significance threshold section](#), p. 38).

If the markers directly flanking the QTL do not provide complete information, genetic information from markers surrounding the assumed QTL map position is used to calculate the component probabilities. The way this is done varies slightly over the population types. For all population types except CP, markers with unknown genotype (u) are completely ignored (markers with unknown genotype are sometimes referred to as missing markers). If for a certain individual a QTL flanking marker has an unknown genotype, then the first not-unknown marker beyond this missing marker will be treated as the flanking marker, of course taking its distance to the current QTL position into account. For the F2 and Rix population types marker genotypes can also be dominant, which means that the genotype is just partially unknown (or partially known). MapQTL tries to resolve these so-called incomplete genotypes as much as possible, by taking into

account the markers beyond it on the map. However, to prevent endless calculations in case neighbouring markers also have incomplete genotypes, a limit is set upon the number of markers used beyond the flanking markers. The corresponding parameter is called the *maximum number of neighbouring markers used*; when set to 0 only the genotypes of the markers flanking the current QTL position are used. Missing markers are ignored in the counting of these neighbouring markers used. In the (hopefully rare) case that all of the linked marker genotypes are unknown and cannot be used to calculate the component probabilities, then their average values over the population are used (the genotyped part of the population in case of selective genotyping). It must be noted that in the case of analysing unmapped loci, no use can be made of linked markers, so that for incompletely known genotypes here no further information can be obtained.

In the calculations of the component probabilities for an RIx population, use is made of exact composite three locus genotype probabilities (i.e. for the QTL in between two flanking markers). This is necessary because recombination events are not completely statistically independent (except in the generation of RI2). These computations involve matrix multiplications for every generation after the RI2, and are somewhat heavy. By ignoring the dependence the composite three locus genotype probabilities can be approximated by multiplying a single locus probability and two transition probabilities for each transition to a neighbouring locus, as is done for all other population types (where there is independence and it is therefore correct). This approach generally leads to very similar outcomes, but the computations are faster. It can be chosen with the *Use genotype probability approximation in RIx* parameter.

In a CP population markers of various segregation type can be used. Here, like dominant genotypes in an F2, several genotypes of a number of segregation types provide partially incomplete information. Similar to the approach for F2 and RIx neighbouring markers are used to calculate the component probabilities. This method was called *all-markers mapping* by Maliepaard & Van Ooijen (1994). In contrast to F2 and RIx, the completely unknown markers (uu) are not completely ignored in the implementation for CP: they are included in the counting of the number of neighbouring markers used, although of course they do not contribute to the resolution of partially unknown genotypes.

Genotypic information coefficient

The power of interval mapping lies in the possibility to make use of genetic information of markers in the region of the QTL. When markers lie further away, when many marker scores are missing, or when markers are scored dominantly, the power reduces because the genotypic information is smaller. In order to quantify the genotypic information across the genome the *genotypic information coefficient (GIC)* is defined, analogous to

the *marker information content* defined by Knott et al (1997) and Reyes-Valdés & Williams (2002). The *GIC* can have a value in the range from 0 to 1, 0 meaning there is no marker information at all, 1 meaning that there is complete or maximum marker information: the marker lies on top of the putative QTL while all individuals in the population have a codominant score. The derivation of the *GIC* is as follows. A QTL generates an amount of genotypic variance, V_Q . This variance can be partitioned into the variance to be explained by genotypic (marker) information, V_M , and the variance remaining due to uncertainty in the genotypic (marker) information, V_R . From this partitioning the *GIC* is defined as:

$$GIC = V_M / V_Q = (V_Q - V_R) / V_Q.$$

For each population type the *GIC* has specific formula's. For an F2 we use a QTL with the heterozygote strictly intermediate, expected means of the A, H and B genotypes of the QTL are +1, 0 and -1, respectively. From the neighbouring markers the probabilities for the QTL genotypes can be calculated: π_A , π_H , π_B . Using these probabilities, expected means and the standard formula for the variance, V_R is the following sum over N individuals:

$$\begin{aligned} V_R &= \sum_{n=1}^N (\pi_A^2 + \pi_H^2 + \pi_B^2 - (\pi_A + \pi_H + \pi_B)^2) \\ &= \sum_{n=1}^N (\pi_A + \pi_B - (\pi_A - \pi_B)^2). \end{aligned}$$

It can be seen that this variance is due to uncertainty in the QTL genotype: when there is no marker information the probabilities π_A , π_H , π_B are the Mendelian expectations 0.25, 0.5 and 0.25, respectively, resulting in $V_Q = V_R = 0.5N$, whereas when the marker is on top of the QTL the probabilities for any individual are either $[\pi_A=1, \pi_H=0, \pi_B=0]$, or $[\pi_A=0, \pi_H=1, \pi_B=0]$, or $[\pi_A=0, \pi_H=0, \pi_B=1]$, resulting in $V_R=0$, i.e. there is no variation due to uncertainty. Applying the above definition of *GIC*, the formula for the F2 is:

$$GIC = 1 - 2 \sum_{n=1}^N (\pi_A + \pi_B - (\pi_A - \pi_B)^2) / N.$$

This formula also applies to population type RIx. For the population types with two genotypes (BC1, DH, DH1, HAP, HAP1) the following formula applies:

$$GIC = 1 - 4 \sum_{n=1}^N (\pi_X \pi_Y) / N,$$

with X and Y being the two genotypes that go with the population type. For population type CP it is more interesting to look at the two parents separately; thus, you have the first

and second parent *GIC*'s (the program output also gives the mean of the two). The first and second parent *GIC*'s have the same formula for populations with two genotypes above, but here the QTL genotype probabilities are sums of two underlying genotype probabilities, for the first parent *GIC*: $\pi_X = \pi_A = \pi_{AC} + \pi_{AD}$ and $\pi_Y = \pi_B = \pi_{BC} + \pi_{BD}$, and for the second parent *GIC*: $\pi_X = \pi_C = \pi_{AC} + \pi_{BC}$ and $\pi_Y = \pi_D = \pi_{AD} + \pi_{BD}$. The resulting formula's are identical to those given by Knott et al (1997), though at first sight they may appear different. Finally, when there is selective genotyping (see below), the standard Mendelian expectations are used as the QTL genotype probabilities for the ungenotyped individuals; this results in a maximum *GIC* value equal to the fraction of genotyped individuals.

Selective genotyping

When the marker genotypes are determined of only a selected part of the segregating population, e.g. the 20% individuals with the highest and the 20% with the smallest values of the quantitative trait, this is called selective genotyping. Generally, it is considered to be a way of enhancing the power of QTL detection and mapping, and also a cost reduction method. However, when there are several QTLs the employment of a multiple-QTL model becomes problematic; it is supposed that the genetic effects of the distinct loci will be estimated with a large (negative) covariance. Therefore, selective genotyping is implemented only for interval mapping in MapQTL, not for MQM mapping. The analysis of selective genotyping with MapQTL is easily arranged by putting the data of the genotyped individuals in the top of the quantitative data file and those of the not-genotyped ones below. The top individuals should correspond to the individuals in the loc-file.

In the calculation of the likelihood the not-genotyped individuals obtain probabilities for the possible QTL genotypes based upon the usual expected segregation ratio (see [Table 8](#), p. 47), corrected for the segregation ratio acquired with the genotyped individuals. However, when this correction leads to illegal values (negative or larger than 1) the uncorrected expected segregation ratio will be used. When this happens, the program reports this as *modifying mixing proportions of ungenotyped individuals*.

MQM mapping

The approximate multiple-QTL model used in MQM mapping is an extension of the single-QTL model presented in the Interval mapping section. In this model selected cofactors take over the role of nearby QTLs; this forms the approximation aspect of the model. In the model a single segregating QTL is fitted in a background of cofactors.

Genetic effects of the separate QTLs, i.e. the single fitted QTL plus the others as represented by cofactors, are modelled as additive fixed effects, there is no QTL-by-QTL interaction (=epistasis). Just as in the single-QTL mixture model, the mixture density for an individual is the sum of the products of the Q component densities with their probabilities. The important distinction is that each individual now has a separate set of component densities. These component densities are normal densities with common variance σ^2 , but the component means are based on both QTL genotype and cofactor genotype: the means μ_{nq} of the Q distributions of individual n consist of the means of the distributions associated with the QTL genotypes μ_q that are adjusted with the genetic effects associated with the genotypes of its cofactors. Depending on the population type, these genetic effects are the additive and/or dominance effects, or for a CP population the deviations from the ac genotype in phase type $\{00\}$, all effects as defined in the [Interval mapping output section](#) of the Using MapQTL chapter (p. 15). For instance, when we have an F2 individual modelled with two cofactor markers, on which it has genotypes b and h , respectively, the three means μ_{nq} are modelled as:

$$\mu_{nq} = \mu_q + (-\alpha_1) + \delta_2,$$

in which α_1 is the additive genetic effect associated with the first cofactor and δ_2 is the dominance effect of the second. As in the single-QTL model the parameters μ_q , σ^2 and the genetic effects associated with each cofactor are estimated with the EM algorithm. The estimates of the cofactor genetic effects (the regressors) are printed in the cofactor monitor output file. The presence of a single segregating QTL is tested by comparing the model with QTL to the nearly identical model but without QTL, both models having the same cofactors. The LOD score or deviance is used for this purpose.

The problem of incomplete genotypes flanking the QTL in the single-QTL model of interval mapping, of course, also occurs for the single fitted QTL in the approximate multiple-QTL model, and is solved in the same fashion using the markers beyond the QTL flanking markers. A new difficulty arises, however, when cofactor genotypes are incomplete. This is solved by taking into account all possible genotypes given the incomplete genotype and calculating the probabilities of these so-called complete genotypes using the markers linked to the cofactor. If necessary, this can be done for more than just one cofactor. Subsequently, the mixture model is extended into more components, one for each combination of fitted QTL genotype and possible complete cofactor genotype. The means of the component distributions are the separate μ_q 's adjusted with the genetic effects associated with the complete cofactor genotypes. The component probabilities are determined by calculating the joint probabilities of the possible complete cofactor genotypes combined with the QTL genotypes.

For instance, if in a model with one cofactor an F2 individual has a dominant cofactor genotype c , then there are two possible genotypes for the cofactor, h and b , while there

are three possible QTL genotypes. Thus, for this individual the mixture will consist of six ($=2 \times 3$) components. With α and δ defined as the additive and dominance genetic effects, respectively, associated with the cofactor, the means of the three distributions, that correspond to the cofactor possibility h , will be:

$$\mu_{hq} = \mu_q + \delta,$$

while those corresponding to possibility b will be:

$$\mu_{bq} = \mu_q + (-\alpha).$$

With the probabilities for cofactor possibilities h and b (based on the surrounding markers) defined as π_h and π_b , respectively, the component probabilities, that correspond to the cofactor possibility h , will be (due to independence of recombination events within and between linkage groups):

$$\pi_{hq} = \pi_h \pi_q,$$

and those corresponding to possibility b will be:

$$\pi_{bq} = \pi_b \pi_q.$$

This simple relation doesn't hold if the cofactor is on the same chromosome as the QTL and the markers in between QTL and cofactor do not provide complete information. In such cases joint probabilities of cofactor genotype and QTL genotype must be calculated as multipoint genotype probabilities. In fact, independence is only employed in these calculations between chromosomes, in all other cases, e.g. with more cofactors (with possibly unknown genotypes) on a single chromosome, multipoint genotype probabilities are determined.

If the cofactor genotype in the example would be unknown (u), then the mixture would consist of nine components. And when the model contains more cofactors and the individual also has a dominant genotype on a second cofactor, then the number of components is multiplied by two. Thus, the number of components in the mixture is multiplied by the number of possible genotypes for each cofactor.

The number of parameters, that have to be estimated, doesn't change with incomplete cofactor genotypes. The number of components and thus the number of probabilities, that have to be calculated, however, grows in a multiplicative fashion with cofactor incompleteness. It is important to realise this, because choosing several cofactors, that have high amounts of incompleteness (e.g. RAPDs in an F2), can dramatically reduce the speed of the program.

A not very obvious phenomenon with MQM mapping is that the estimated genetic effect and the percentage of explained variance are rather biased when the fitted QTL lies not

very far from a linked (!) cofactor marker that is included in the model to absorb the effects of another QTL. This is due to the fact that under the null-hypothesis the linked cofactor marker will absorb part of the same genetic effect that is being fitted to a QTL under the alternative hypothesis.

LOD significance threshold

Several papers address the subject of what significance threshold to use for the LOD score in a QTL mapping experiment (e.g. Lander & Botstein, 1989; Van Ooijen, 1992; Feingold et al, 1993; Rebaï et al, 1994; Jansen, 1994; Lander & Kruglyak, 1995; Kruglyak & Lander, 1995; Doerge & Rebaï, 1996). For MapQTL, extensive computer simulations were carried out to come towards a convenient method of calculating the appropriate threshold for the genome size and population type under study; this research was published by Van Ooijen, 1999. MapQTL also offers the permutation test for interval mapping, with which the significance threshold can be determined based on the actual data rather than on assumed normally distributed data.

Data files

General

MapQTL uses plain text files to load the data that must be analysed. A plain text file can be made with any text editor program. MapQTL uses several types of data files, each containing different kinds of information. Besides the actual data the files contain instructions that guide the program through the information.

First, there is the *locus genotype file* (also called *loc-file*), which contains the genotype codes for the loci of a single segregating population. Then there is the *map file* containing the map positions of all loci. Thirdly, the *quantitative data file* (also called *qua-file*), which as the name suggests, holds the data of the quantitative traits of all individuals. And finally, there is the *cofactors file* (also called *cof-file*) containing the names of the markers that should be used as cofactors in MQM mapping analysis. The loc-file and the map file have the same formats as are used for JoinMap 3.0 (Van Ooijen & Voorrips, 2001); although the format for population type CP is different from JoinMap 2.0 (Stam & Van Ooijen, 1995), MapQTL also reads and interprets JoinMap 2.0 CP type files correctly.

Data file characteristics

Here we give some important general features with respect to the data files for MapQTL. The various data files themselves will be described in detail in subsequent sections.

For the sake of readability the data files may contain extra so-called *whitespace* wherever found appropriate; this is not allowed, however, within the various instructions, indicators, locus and file names, etc.. Whitespace is a sequence of one or more of the next characters: space, tab, newline (linefeed), carriage-return, vertical-tab and formfeed. The software is indifferent to the use of *lower-* or *uppercase*, both in the instructions and in the actual information. It is possible, and good practice as well, to put relevant comment

in a data file. To make a *comment line* place a semicolon ";" at the beginning of the line; to put comment somewhere in a line, place whitespace followed by a semicolon. Anything on the line behind the semicolon will be ignored by MapQTL.

The layout of the various files is either line-structured or sequential. The choice for a particular layout has to do with readability (by eye) and the amount of data that belongs together. Good readability is a proper measure for the prevention of errors. But occasionally some data groups may be so large that they don't fit on a single line. *Line-structured* means that data belonging together have to reside on the same single line. For instance in the map file, the locus name and its map position must be on a single line. *Sequential* means that the data are read from left to right, from top to bottom, and there is no requirement to group data on a single line. For instance in the locus genotype file, the genotype codes belonging to a single locus determined in a large population may not fit on a single line, and often have to be continued over several lines. Of course, it is a good measure to obtain proper readability by suitable spacing.

Some data files contain in the top of the file instructions regarding the contents of the data file, e.g. the number of individuals and the number of loci. This part of the file is called the *header*. The program is indifferent to the order in which the various instructions in the header are given. The header always has a sequential structure.

Some data elements are of *fixed length*, while others are of *variable length*. For instance, locus names may be up to twenty characters long, but they may also be shorter. In order to read variable-length data fields they must be separated from other data fields by whitespace. On the other hand, fixed-length data fields need not be separated by whitespace, although it is allowed (and often to be recommended). For instance, the genotype codes of individuals from one population are all the same size, two characters for cross pollinators (CP) and one for other population types, and may be given without spacing (though this will result in poor readability).

The names of loci, linkage groups, traits and populations, and also the missing value indicator, may be up to twenty characters long. Names cannot include spaces. The (full path) names of files may be up to 255 characters long. Lines may be up to 1000 characters wide (this only applies to line-structured data).

Locus genotype file

The locus genotype file (*loc-file*) contains the information of the loci for a single segregating population. It has a sequential structure. The header of the file contains four instructions on the contents of the data body. The data body contains the actual genotype

information for each locus and for all individuals. The four instructions define the name of the population (which is for administrative use only), the type of the population, the number of loci, and the number of individuals. These instructions can be given in any order within the header. The syntax of the four instructions is:

```
name = NAME
popt = POPT
nloc = NLOC
nind = NIND
```

where NLOC and NIND are the numbers of loci and individuals, respectively, NAME is the name of the population (which cannot contain spaces), and POPT is the code for the population type, which must be one of the codes given in [Table 2](#).

What happens if NIND or NLOC are incorrect? If NIND is incorrect, then MapQTL will try to interpret part of a locus name as a genotype code, which in general will lead to an error message. If NLOC is larger than the actual number of loci in the file, then MapQTL will try to read beyond the end of the file, which will also lead to an error message. If NLOC is smaller than the actual number, then it will issue a warning that there are more data in the file. You might want to exploit this feature to park loci that you do not want to be used.

The data body contains the information for all loci and individuals, grouped per locus. The data group for a locus consists of the name of the locus, followed by the genotype codes of all individuals. In between the locus name and the genotypes there can optionally be up to three additional instructions, depending on the type of population. MapQTL is indifferent to the order of these instructions. The instructions are concerned with the type of segregation of the locus (SEG) (for population type CP), the linkage phases of the locus (PHASE) (for population types CP, DH and HAP), and the type of classification for the locus (CLAS). In short, the syntax of a data group for a locus is (optional is indicated with `[]`):

```
<locus name>  [SEG]  [PHASE]  [CLAS]  <NIND genotypes>
```

It is important to note that it is absolutely essential that the order of the individuals is identical over all loci in the file. The genotype codes for population types F2, BC1 and RIX are given in [Table 3](#). Those for population types DH1 and HAP1 are identical to these, albeit that the heterozygous and dominant genotypes are excluded ([Table 4](#)). The genotype codes for a DH or HAP population are identical to those for DH1 and HAP1, but have a slightly different meaning, since the parentage of the alleles is not relevant ([Table 5](#)).

Table 2. Population type codes

Type	Description
F2	an F2 population: the result of selfing the F1 of a cross between two fully homozygous diploid parents
BC1	a first generation backcross population: the result of crossing the F1 of a cross between two fully homozygous diploid parents to one of the parents
RIx	a population of recombinant inbred lines in the x-th generation: the result of inbreeding an F2 by single seed descent; RI2 is equivalent to an F2
DH	a doubled haploid population: the result of doubling the gametes of one heterozygous diploid individual, linkage phases originally (possibly) unknown
DH1	a doubled haploid population produced from the gametes of the F1 of a cross between two homozygous diploid parents
HAP	a haploid population: the gametes (or derived individuals) of one heterozygous diploid individual, linkage phases originally (possibly) unknown
HAP1	a haploid population derived from the F1 of a cross between two fully homozygous diploid parents
CP	a population resulting from a cross between two heterogeneously heterozygous and homozygous diploid parents, linkage phases originally (possibly) unknown

For interval and MQM mapping in a DH or HAP population MapQTL requires known linkage phases of the alleles of the loci. (The nonparametric mapping does not need linkage phases, because it analyses the loci one by one.) These linkage phases can be obtained automatically with JoinMap 3.0. The genotype coding scheme is based on the loci to be in coupling in the parent, i.e. the *a* 's come from the same one grandparent, the *b* 's from the other grandparent. However, to allow for linkage phase differences a linkage phase indicator is used, a *phase type*. Such a phase type must be one of the following single-letter codes between curly brackets:

{0} or {1}.

For a locus with a phase type 1 the grandparental origin is switched, i.e. the *a* 's originate from the other grandparent, the *b* 's from the one grandparent. Locus pairs with the same phase code are assumed to be in coupling in the parent, and in repulsion otherwise.

Table 3. Genotype codes for population types F2, BC1 and RIx

Code	Description
a	homozygote as the one parent
b	homozygote as the other parent
h	heterozygote (as the F1)
c	not genotype a (dominant b-allele) (not possible for a BC1)
d	not genotype b (dominant a-allele) (not possible for a BC1)
–	genotype unknown
.	genotype unknown
u	genotype unknown

Remark:

A BC1 must be coded either with a's and h's, or with h's and b's.

Table 4. Genotype codes for population types DH1 and HAP1

Code	Description
a	homozygote or haploid as the one parent
b	homozygote or haploid as the other parent
–	genotype unknown
.	genotype unknown
u	genotype unknown

Table 5. Genotype codes for population types DH and HAP

Code	Description
a	the one genotype
b	the other genotype
–	genotype unknown
.	genotype unknown
u	genotype unknown

For population type CP the type of segregation may vary across the loci. Up to four different alleles may be segregating. Therefore, a code indicating the *segregation type* must be given in between the locus name and the genotypes. The segregation type codes are shown in [Table 6](#). The two characters left of the "x" in these codes represent the alleles of the first parent, the two on the right represent those of the second parent; each distinct allele is represented with a different character. The genotypes for a CP population must be coded with two characters, representing the two alleles, per individual. The coding depends on the segregation type, and is shown in [Table 7](#). MapQTL is indifferent to the order of the alleles, so: `ac` is equivalent to `ca`. In all cases the ".", the "-", and the `u` are treated as equivalent, so: `h.` and `hu` are both equivalent to `h-`. Although not required, it is recommended as a good measure against errors to separate the genotype codes of individuals with a space. The two-character codes themselves may not be separated with whitespace. The CP coding scheme is enhanced from the previous version 3.0 and from JoinMap 2.0, but this older format is interpreted correctly by the present version.

Analogous to the population types DH and HAP, MapQTL requires known linkage phases of the alleles of the loci for interval and MQM mapping in a CP type population. The genotype coding scheme is based on the alleles on the same position within the segregation type codes to be in coupling in the parent, i.e. the `a`, `e`, `h` and `l` alleles from the first parent come from the same one grandparent, the `b`, `f`, `k` and `m` alleles from the first parent from the other grandparent. However, to allow for linkage phase differences a linkage phase indicator is used similar to DH and HAP, but here we need a two-digit *phase type*, of which the first relates to the one parent and the second to the other. The phase type must be one of the next two-letter codes between curly brackets:

```
for the seg. type <lmxll>:    {0-} or {1-},
for the seg. type <nnxnp>:    {-0} or {-1},
for the other seg. types:     {00}, {01}, {10} or {11}.
```

Locus pairs with the same digit in the first position of their phase types are assumed to be in coupling in the first parent, and in repulsion in the first parent otherwise; for the second position the relation is likewise about the second parent. For instance, if a locus L is of type `<hkxhk>` {00} and another locus M is `<abxcd>` {01}, this means that in the first parent the `h`-allele of L and the `a`-allele of M are in coupling (and thus also their `k`- and `b`-alleles), and that in the second parent the `h`-allele of L is in repulsion with the `c`-allele of M (and thus in coupling with the `d`-allele of M). The phase type must be given in between the locus name and the genotypes. Here too, the linkage phases can be obtained automatically with JoinMap 3.0.

For the nonparametric mapping the program classifies the genotypes according to the

Table 6. Segregation type codes for population type CP

Code	Description
<abxcd>	locus heterozygous in both parents, four alleles
<efxeg>	locus heterozygous in both parents, three alleles
<hkxhk>	locus heterozygous in both parents, two alleles
<lmxll>	locus heterozygous in one parent
<nnxnp>	locus heterozygous in other parent

Table 7. Genotype codes for a CP population, depending on the locus segregation type

Seg. type	Possible genotypes
<abxcd>	ac, ad, bc, bd, -- (no dominance allowed)
<efxeg>	ee, ef, eg, fg, -- (no dominance allowed)
<hkxhk>	hh, hk, kk, h-, k-, --
<lmxll>	ll, lm, --
<nnxnp>	nn, np, --

Remarks:

1. each character a to p represents a distinct allele; "-" means unknown allele
2. h- and k- are dominant genotypes:
h- means hh or hk, and
k- means kk or hk
3. "." and u are treated equivalent to "-"

usual genotype classes. However, you may wish to classify in another way, e.g. when there is dominance. A classification type can optionally be given in the loc-file in between the locus name and the genotypes to force a certain classification. The classification type codes are given in [Table 8](#). The classification type must only be given, when a classification other than the default is desired. In fact, this is only necessary when there is dominance, or in the case of population type RIX. MapQTL does not allow classification types other than the default and optional types for the population and/or segregation type. If there is only the default classification type, then a classification type need not and cannot be given. The defaults and the options are shown in [Table 9](#). [Examples 1](#) and [2](#) are demonstrations of a locus genotype file.

Example 1. A locus genotype file for an F2 population

```

; 12 March 1995
; this is a ridiculously small data file
; but it serves only as an example

name = some_demo!
popt = F2                ; these data are from an F2 population
nloc = 2                 ; the file contains data on two loci
nind = 6                 ; and six plants

RFLP05                  ; this is a locus name
  aahba b               ; these are the genotypes of the six plants
RFLP67 (a,c)           ; classify this locus into a and c
  accac a

```

Example 2. A locus genotype file for a CP type population

```

; 12 March 1995
; this is another ridiculously small data file
; again, just an example

name = what_a_demo!
popt = CP                ; it is a CP type of population
nloc = 3                 ; it contains data on three loci
nind = 7                 ; and seven plants

RFLP21 <efxeg> {01}      ; marker RFLP21 segregates with
                        ; three alleles
  ef ee eg fg fg ef eg  ; genotypes of the seven plants
RAPD17 <hkxhk> (h-,kk) {00} ; classify into h- and kk
  h- h- kk h- kk h-    ; the seven genotypes in
                        ; identical order as for RFLP21
RFLP34 <nnxnp> {-1}     ; the linkage phase at this seg.
                        ; type defines it only for the
                        ; second parent
  nn np np np --       ; the autoradiogram was unclear
  nn np                 ; for plantnr 5

```

Map file

The map file contains the map positions of all loci. The nonparametric mapping of MapQTL examines the loci one by one; the map positions are only used to sort the loci. For interval and MQM mapping the map positions are used to calculate recombination frequencies, necessary for the calculation of the likelihood.

The map file is strictly line-structured and there is no header. Linkage groups must be started with the instruction *group* or *chrom* on a separate line. On the subsequent lines the loci with their map positions must be given in ascending order, one locus with its position

Table 8. Classification type codes. *Ratio* is the expected segregation ratio

Code	Ratio	Classification into genotype classes
(a,b)	1:1	a and b
(a,h)	1:1	a and h
(a,c)	1:3 *	a and c; h and b will be included in class c
(h,b)	1:1	h and b
(b,d)	1:3 *	b and d; a and h will be included in class d
(a,h,b)	1:2:1 *	a, h, and b
(ac,ad,bc,bd)	1:1:1:1	ac, ad, bc, and bd
(ee,ef,eg,fg)	1:1:1:1	ee, ef, eg, and fg
(hh,k-)	1:3	hh and k-; hk and kk will be included in class k-
(h-,kk)	3:1	h- and kk; hh and hk will be included in class h-
(hh,hk,kk)	1:2:1	hh, hk, and kk
(ll,lm)	1:1	ll and lm
(nn,np)	1:1	nn and np

**for RI_x the ratios are adjusted according to the generation number x*

Table 9. Default and optional classification types

Pop. type	Seg. type	Default	Optional
<i>A classification type is NOT ALLOWED in the data file:</i>			
BC1		(a,h) or (h,b) *	none
DH		(a,b)	none
DH1		(a,b)	none
HAP		(a,b)	none
HAP1		(a,b)	none
CP	<abxcd>	(ac,ad,bc,bd)	none
	<efxeg>	(ee,ef,eg,fg)	none
	<lmxll>	(ll,lm)	none
	<nnxnp>	(nn,np)	none
<i>Classification types are ALLOWED in the data file:</i>			
F2		(a,h,b)	(a,c) or (b,d)
RI _x		(a,b)	(a,h,b), (a,c) or (b,d)
CP	<hkxhk>	(hh,hk,kk)	(h-,kk) or (hh,k-)

** automatically determined*

Example 3. A map file

```
; the file is completely line-structured
group a
;<locus> <map position>
  rapd02    0.0
  rapd86    11.1
  rapd08    15.2
  rapd22    17.3
group b
  rapd54    0.0
  rapd66    15.2
  rapd18    22.3
```

per line. It is not required to start at map position 0.0. A following linkage group must start again with the group-instruction. Next to the group-instruction MapQTL attempts to read a group name of up to twenty characters (no spaces), which, if available, will be used in the output. A small map file is demonstrated in [Example 3](#).

Quantitative data file

This file holds the data of the quantitative traits of all individuals. It has a sequential structure. The header of the file contains three instructions on the contents of the data body, followed by the names of the traits. The data body contains the actual information for each trait and for all individuals. The three instructions define the numbers of traits and individuals and the text that indicates a missing value. These instructions can be given in any order. The syntax of the three instructions is:

```
ntrt = NTRT
nind = NIND
miss = MISS
```

where NTRT and NIND are the numbers of traits and individuals, respectively, and MISS is the missing value indicator, i.e. a text string that is used to indicate missing values (cannot contain spaces, maximum length is 20). There are no maximum values for NTRT and NIND (of course, reading the file becomes time consuming at extreme values of NTRT), but NIND must be equal to or larger than (selective genotyping, see below) the value of NIND in the corresponding loc-file. These instructions must be followed by the names of all the traits, after which the data body must be given. It is allowed (and even advisable, see below) to include non-numerical traits. After loading the data into a MapQTL project the non-numerical traits will show up in green in the Populations tabsheet, and no analysis can be done on such a trait. The names of the traits may be up to 9 characters long (cannot contain spaces), but only the first twenty will be used. Data

Example 4. A quantitative data file

```

; tiny experiment tomato/mildew '94

ntrt = 3 ; three traits
nind = 5 ; five genotypes
miss = * ; missing values are indicated with a '*'

nr      ; indiv. number; non-numerical 'trait', not to analyse for QTLs !
length  ; length to the 5th leaf averaged over 3 cuttings
mildew  ; mildew is the disease score averaged over 3 cuttings

94_1    69    3.2
94_2    75    4.5
94_3    54    2.1
94_4    66    *
94_5    71    1.1

```

fields of non-numerical traits may also be up to 99 characters long. Because the trait name is used as part of the output file name, a trait name should only contain legal file naming characters; this is operating system dependent.

In the data body the values of all the traits are given, grouped per individual (Contrarily, in the loc-file the data are grouped per locus.). Although the layout is completely sequential, for explanatory reasons it is best to look at the data body as a matrix. In this matrix the columns represent the traits, while the rows represent the individuals.

What happens if NTRT or NIND are incorrect? If NIND or NTRT are larger than the actual numbers, MapQTL will try to read beyond the end of the file, which will lead to an error message. If NIND is smaller (it is not allowed to be smaller than the NIND-value in the loc-file), then the traits will be read correctly for all NIND individuals, but the program will find more data after the last individual and will issue a warning. If NTRT is incorrect, then for most individuals the wrong trait will be read; this may remain undetected, unless there is a trait that consists of text instead of numerical values, in which case the unsuccessful attempt to interpret text as a numerical value will lead to an error message. So, you might consider including a text trait just for the sake of error detection.

It is important to note, that it is absolutely essential that the order of the individuals is identical to the order in the loc-file, i.e. the X-th individual in the loc-file must be the same as the X-th individual in the qua-file. In the case of selective genotyping, i.e. of only a part of the population (of which of all individuals a quantitative trait was determined) the genotypes of the markers were determined, the first individuals must correspond to the individuals in the loc-file, while the not-genotyped individuals are appended.

[Example 4](#) demonstrates a small qua-file.

Cofactors file

This file is used to feed MapQTL with the names of the loci to be used as cofactors in the MQM analysis. It is line-structured. The header consists of one instruction only. It is:

```
ncof = NCOF
```

where NCOF is the number of loci in the file. This instruction is followed by the names of the loci, each on a separate line. [Example 5](#) demonstrates a small cof-file.

Default file name extensions

For ease of use we have introduced default file name extensions for the various files. The default extensions are given in [Table 10](#).

Example 5. A cofactors file

```
ncof = 3      ; three cofactor markers
; group 2
rapd23
rapd19
; group 5
rflp11
```

Table 10. Default file name extensions

File	Extension
cofactors file	.cof
cofactor monitor file	.cmo
locus genotype file	.loc
map file	.map
output data file	.mqo
project directory	.mqd
project file	.mqp
quantitative data file	.qua

Lists and references

List of figures

Figure 1. User interface	2
Figure 2. <i>Populations</i> tabsheet with the DemoF2 population	22
Figure 3. <i>Maps</i> tabsheet with the DemoF2 map	22
Figure 4. <i>Results Charts</i> tabsheet with subordinate <i>Control</i> tabsheet visible	25
Figure 5. LOD profile on linkage group 1 with LOD values above the significance threshold (3.8) some distance away from the cofactor marker m9	27

List of tables

Table 1. Analysis abbreviations	13
Table 2. Population type codes	42
Table 3. Genotype codes for population types F2, BC1 and RIx	43
Table 4. Genotype codes for population types DH1 and HAP1	43
Table 5. Genotype codes for population types DH and HAP	43
Table 6. Segregation type codes for population type CP	45
Table 7. Genotype codes for a CP population, depending on the locus segregation type	45
Table 8. Classification type codes. <i>Ratio</i> is the expected segregation ratio	47
Table 9. Default and optional classification types	47
Table 10. Default file name extensions	50

List of examples

Example 1. A locus genotype file for an F2 population	46
Example 2. A locus genotype file for a CP type population	46

Example 3. A map file	48
Example 4. A quantitative data file	49
Example 5. A cofactors file	50

References

- Barnard, G.A., 1949.
Statistical inference. J. R. Statist. Soc. Ser. B 11: 115-139.
- Churchill, G.A. & R.W. Doerge, 1994.
Empirical threshold values for quatitative trait mapping. Genetics 138: 963-971.
- Dempster, A.P., N.M. Laird & D.B. Rubin, 1977.
Maximum likelihood from incomplete data via the EM algorithm. J. R. Statist. Soc. Ser. B 39: 1-38.
- Doerge, R.W. & A. Rebaï, 1996.
Significance thresholds for QTL interval mapping tests. Heredity 77: 459-464.
- Feingold, E., P.O. Brown & D. Siegmund, 1993.
Gaussian models for genetic linkage analysis using complete high-resolution maps of identity-by-descent. Am. J. Hum. Genet. 53: 234-251.
- Fijneman, R.J.A., S.S. De Vries, R.C. Jansen & P. Demant, 1996.
Complex interactions of new quantitative trait loci, Sluc1, Sluc2, Sluc3, and Sluc4, that influence the susceptibility to lung cancer in the mouse. Nature Genetics 14: 465-467.
- Jansen, R.C., 1993.
Interval mapping of multiple quantitative trait loci. Genetics 135: 205-211.
- Jansen, R.C., 1994.
Controlling the type I and type II errors in mapping quantitative trait loci. Genetics 138: 871-881.
- Jansen, R.C. & P. Stam, 1994.
High resolution of quantitative traits into multiple loci via interval mapping. Genetics 136: 1447-1455.
- Jansen, R.C., J.W. Van Ooijen, P. Stam, C. Lister & C. Dean, 1995.
Genotype-by-environment interaction in genetic mapping of multiple quantitative trait loci. Theor. Appl. Genet. 91: 33-37.
- JoinMap
<http://www.joinmap.nl>.
- Knott, S.A., D.B. Neale, M.M. Sewell & C.S. Haley, 1997.
Multiple marker mapping of quantitative trait loci in an outbred pedigree of loblolly pine. Theor. Appl. Genet. 94: 810-820.
- Kruglyak, L. & E.S. Lander, 1995.
A nonparametric approach for mapping quantitative trait loci. Genetics 139: 1421-1428.
- Lander, E.S. & D. Botstein, 1989.
Mapping Mendelian factors underlying quantitative traits using RFLP linkage maps. Genetics 121: 185-199.

- Lander, E.S. & L. Kruglyak, 1995.
Genetic dissection of complex traits: guidelines for interpreting and reporting linkage results. *Nature Genetics* 11: 241–247.
- Lehmann, E.L., 1975.
Nonparametrics. McGraw-Hill, New York.
- Maliepaard, C. & J.W. Van Ooijen, 1994.
QTL mapping in a full-sib family of an outcrossing species. In: Van Ooijen, J.W. & J. Jansen (Eds.). *Biometrics in Plant Breeding: Applications of Molecular Markers*, Proceedings of the Ninth Meeting of the EUCARPIA Section Biometrics in Plant Breeding, 6–8 July 1994, Wageningen, the Netherlands, pp. 140–146.
- MapQTL
<http://www.mapqtl.nl>.
- Martínez, O. & R.N. Curnow, 1992.
Estimating the locations and the sizes of the effects of quantitative trait loci using flanking markers. *Theor. Appl. Genet.* 85: 480–488.
- McLachlan, G.J. & K.E. Basford, 1988.
Mixture models: inference and applications to clustering. Marcel Dekker, New York.
- Rebaï, A., B. Goffinet & B. Mangin, 1994.
Approximate thresholds of interval mapping tests for QTL detection. *Genetics* 138: 235–240.
- Reyes-Valdés, M.H. & C.G. Williams, 2002.
A haplotype approach to founder-origin probabilities and outbred QTL analysis. *Genet. Res. Camb.* 80: 231–236.
- Snedecor, G.W. & W.G. Cochran, 1980.
Statistical Methods, Seventh edition. Iowa State University Press, Ames, Iowa.
- Stam, P. & J.W. Van Ooijen, 1995.
JoinMap (tm) version 2.0: Software for the calculation of genetic linkage maps. Plant Research International, Wageningen, the Netherlands.
- Titterton, D.M., A.F.M. Smith & U.E. Makov, 1985.
Statistical analysis of finite mixture distributions. Wiley, New York.
- Van Ooijen, J.W., 1992.
Accuracy of mapping quantitative trait loci in autogamous species. *Theor. Appl. Genet.* 84: 803–811.
- Van Ooijen, J., H. Sandbrink, C. Purimahua, R. Vrielink, R. Verkerk, P. Zabel & P. Lindhout, 1993.
Mapping quantitative genes involved in a trait assessed on an ordinal scale: A case study with bacterial canker in *Lycopersicon peruvianum*. In: J. I. Yoder (Ed.) *Molecular Biology of Tomato*, Technomic Publishing Co. Inc., Lancaster PA, Chpt 7, pp 59–74.
- Van Ooijen, J.W., 1999.
LOD significance thresholds for QTL analysis in experimental populations of diploid species. *Heredity* 83: 613–624.
- Van Ooijen, J.W. & C. Maliepaard, 1996.
MapQTL (tm) Version 3.0, Software for the calculation of QTL positions on genetic maps. Plant Research International, Wageningen, the Netherlands.
- Van Ooijen, J.W. & R.E. Voorrips, 2001.
JoinMap® 3.0, Software for the calculation of genetic linkage maps. Plant Research International, Wageningen, the Netherlands.

Van Ooijen, J.W., M.P. Boer, R.C. Jansen & C. Maliepaard, 2002.
MapQTL® 4.0, Software for the calculation of QTL positions on genetic maps. Plant
Research International, Wageningen, the Netherlands.

Index

(?)	14	default file name extensions	50
.emf	11	degrees of freedom	13, 18
.mqd	3, 8	demonstration data	3
.mqp	3, 8	deviance	15, 18, 32
acs	13, 18	dominance	14, 17
all-markers mapping	33	enhanced windows meta file	4, 11
analysis abbreviations	13	environment options	8
analysis options	8	example data files	8
analysis selector	11	execute analysis	8
automatic cofactor selection	18	exit program	8
calculate	3, 12	expected segregation ratio	35, 47
case sensitivity	39	export	4, 11
chart	10	file	
charts	4, 11	cof-	39
classification type	41, 45	cofactors	39, 50
classification type codes	47	loc-	39, 40
cofactor	4, 10	locus genotype	39, 40, 46, 50
cofactor monitor	17	map	39, 46, 48, 50
cofactors	16	output data	50
cofactors file	19, 39, 50	project	7, 8, 50
cofactors tool	4, 10	qua-	39
cof-file	39	quantitative data	39, 48, 49, 50
comment line	40	file name extensions	50
common traits	3, 9	find tool	11
component	31	fixed-length	40
contents-and-results panel	2, 7, 9, 10	functional tolerance value	15, 32
control	4, 11	genome wide	12
copy to clipboard	4, 11	genotype codes	39, 41, 43, 45
ctrl+a	4	genotypes	12
data file characteristics	39	genotypes info	10
data files	39	genotypic information coefficient	33
default	11	gic	33
default classification type	45	group name	2, 48
default classification types	47	group-instruction	48

H0	14, 17	modifying mixing proportions	35
H1	14, 17	mqm	13, 16
header	40	mqm mapping	16, 35
help menu	4, 8	name-instruction	41
im	13, 14	navigation panel	2, 3, 7, 9
installation	1	ncof-instruction	50
interval mapping	14, 31	new project	3, 8
key combinations	7	nind-instruction	41, 48
keyboard shortcuts	7	nloc-instruction	41
kruskal-wallis	12	non-numerical data	3, 9, 49
kurtosis	15	nonparametric	12
kw	12, 13	nr	10
layout	40	ntrt-instruction	48
length of names	2, 40	number of permutations	19
license file	1	open project	8
likelihood	14	optional classification types	47
likelihood ratio test statistic	32	output data file	50
line-structured	40	page setup	4, 11
linkage group	11	permutation test	19, 38
linkage groups	9	phase type	41, 42, 44
linkage phase	41, 44, 46	phase type codes	42, 44
load data	3, 8	plain text	39
loc-file	39, 40	popt-instruction	41
locus genotype file	39, 40, 46, 50	population info	10
locus name	2	population name	2
lod	14, 32	population type	41
lod significance threshold	38	population type codes	42
lowercase	39	populations	3, 9
map file	39, 46, 48, 50	preset default	8
map info	4, 10	print	4, 11
mapping function	31	print preview	4, 11
mapping step size	15	print setup	4, 11
mapqtl project	3, 8	program directory	1, 8
mapqtl.lic	1	program settings directory	2, 8
mapqtl5.exe	7	project	2, 3, 7, 8
maps	3, 9	project backup	8
maximum length of names	2, 40	project directory	3, 8, 50
maximum number of iterations	15, 32	project file	3, 7, 8, 50
maximum number of neighbouring markers used	15, 33	project info	3, 10
memory	2	project notes	3, 10
miss-instruction	48	pt	13, 19
		p-value	13, 18

qua-file	39	results	4, 10
quantitative data file	39, 48, 49, 50	results charts	4, 11
ram	2	session info	12
remove	10	sessions	4, 9
reset	8, 11	traits info	10
reset to default	8	trait name	2
restricted mqm mapping	17	traits info	10
results	4, 10	treeview	3, 9
results charts	4, 11	two-lod support interval	14
right-clicking	9	unmapped	10, 11, 12
rmqm	13, 17	uppercase	39
save as default	8	use genotype probability approximation	
segregation ratio	35, 47	in rix	33
segregation type	41, 44	user interface	2
segregation type codes	45	variable-length	40
selective genotyping	15, 17, 35	variance	15
selector	11	whitespace	39
sequential	40		
session	4		
session info	12		
sessions	4, 9		
settings directory	2, 8		
setup.exe	1		
shift+del	10		
significance level	13		
significance threshold	19		
skewness	15		
sort table	10		
special keys	7		
specially selected	9		
splitter	10, 11		
table	10		
tabsheet	2, 7		
charts	4, 11		
control	4, 11		
genotypes info	10		
map info	4, 10		
maps	3, 9		
population info	10		
populations	3, 9		
project info	3, 10		
project notes	3, 10		