Marlin Help

Patrick Meirmans

http://www.patrickmeirmans/com/software

August, 2009

About this document:

This Help-document is not a complete manual for Marlin, rather it is a combination of all the Help-files of Marlin. Because of this, many elements that you may expect from a manual are lacking such as a paginated index of all contents, a readable layout, and page-numbers. Also bear in mind that all the hyperlinks (blue underlined text) in the document are not working.

For questions about Marlin, do not hesitate to contact me at:
p.g.meirmans@uva.nl

Table of contents:

Creating simulations

Running simulations

- Starting runs
- What happens during a run?
- quantiNemo

Viewing the results

- Analyses
- Plotting
- XY plots
- Result maps
- Where do the files go?

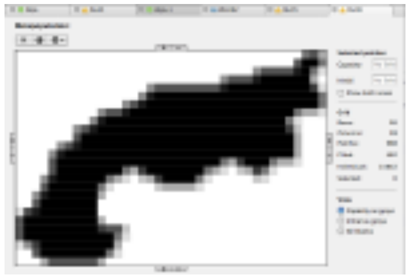
Miscellaneous

- Citing Marlin
- GenoDive
- License
- Cited references
- Version history

What Marlin does

Marlin is a program to create, run, analyse, and visualise spatially explicit population genetic simulations. It provides an intuitive user interface with which a metapopulation can be drawn by hand and the many different settings can be easily selected. But Marlin goes further than that and directly analyses and plots the results. This combination of simulation and analysis makes Marlin ideal for teaching and for scientists who are interested in doing simulations without having to learn command-line operations.

Create simulations

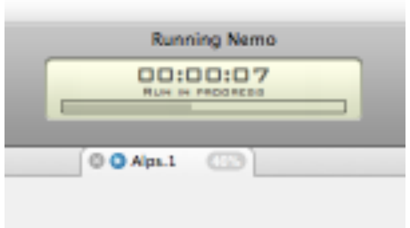


Marlin allows you to easily create geographically realistic metapopulation lay-outs. Populations can be added by simply clicking their location on a grid, after which the populations' maximum capacity and initial population size can be specified. For more complex scenarios, the metapopulation lay-out can be loaded from an image file, for example a species' distribution map.

Besides the direct drawing of the metapopulation, Marlin also allows you to easily change many life-history characters through it's graphical interface.

- Creating simulations

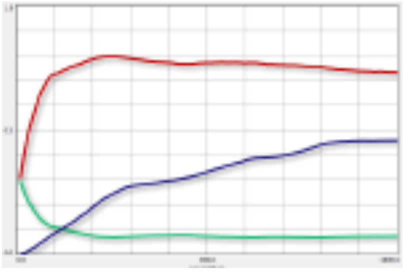
Run simulations



Technically, Marlin provides a front end for the simulation software QuantiNemo, developed by Samuel Neuenschwander. This provides a solid and thoroughly tested foundation for the simulations. However, Marlin focuses on neutral genetic markers and does not (yet) use QuantiNemo’s ability to simulate quantitative traits. Settings-files can be exported for use by QuantiNemo, for example for running simulations on a computer cluster. In this way, the user-friendly interface of Marlin can be combined with the power of the commandline.

- Running simulations

Analyse and visualise results



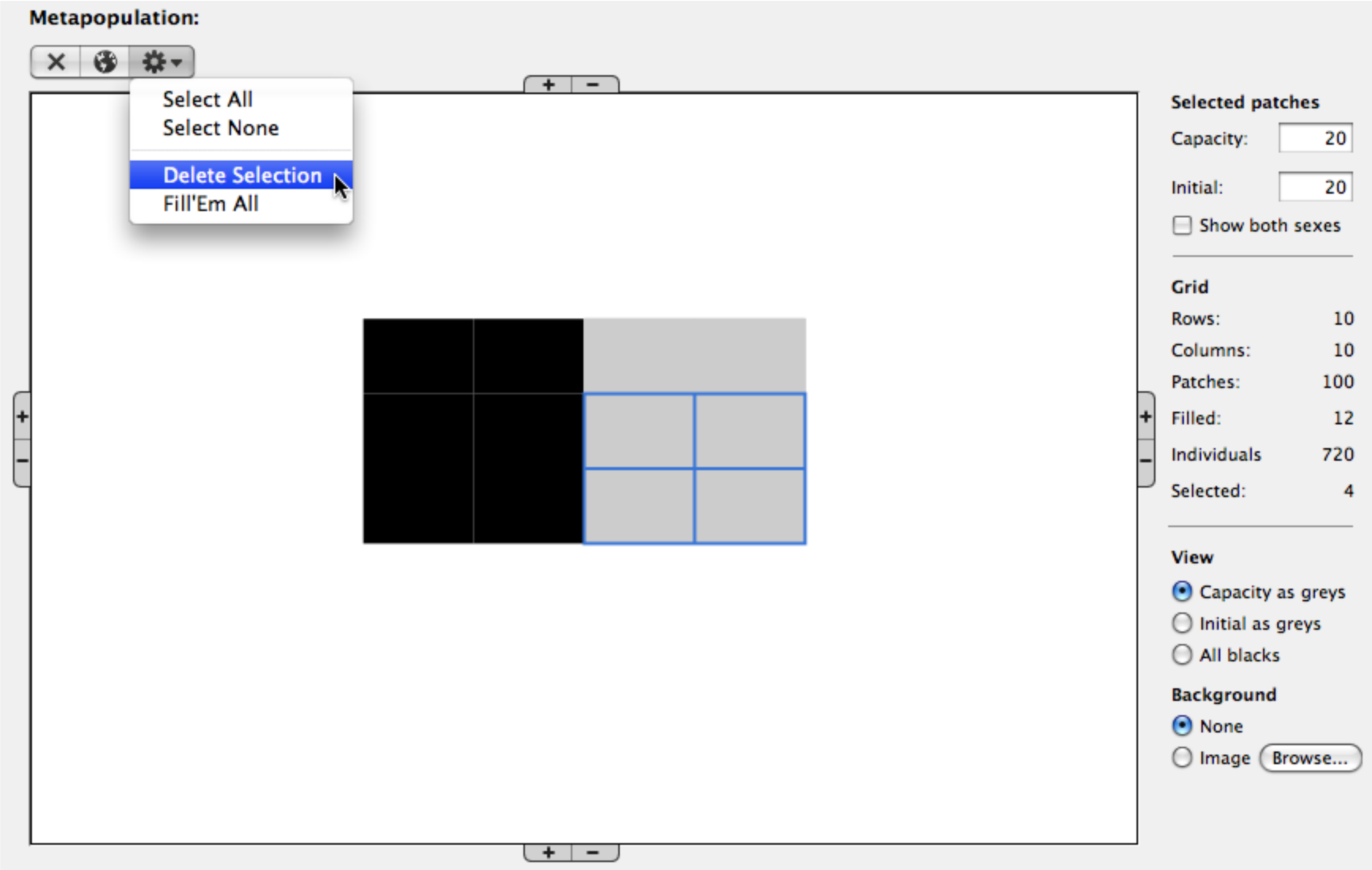
Marlin does not only create and run the simulations, it also immediately analyses and plots the results. Two types of plots are available: an xy-plot where the value of the computed summary statistics can be graphed over time, and a map where the results of some population genetics' analyses are shown for each population, graphically represented through the populations' colours.

- Viewing the results

Creating simulations

Drawing Populations

With Marlin you can directly draw the geographical lay-out of the metapopulation. Simply clicking an empty location will create a new population, after which the population's parameters can be set. This allows you to quickly create realistic geographical simulations. Alternatively, the lay-out of the metapopulation can be set using an image file (see here for more details on loading a map).



Metapopulation:

Selected patches
Capacity: 20
Initial: 20
☐ Show both sexes

Grid
Rows: 10
Columns: 10
Patches: 100
Filled: 12
Individuals 720
Selected: 4

View
● Capacity as greys
○ Initial as greys
○ All blacks

Background
● None
○ Image Browse...

To draw a metapopulation lay-out by hand:

- If necessary, select the metapopulation editor by clicking on the "Metapopulation" item in the source list on the left-hand side of the window.
- Clicking an empty spot in the drawing area will create a new population and will select it. The parameters of the new population will be set to those of the previously selected population, or to the default values if nothing was selected.
- Clicking a filled spot will select the underlying population. Shift-clicking or dragging the mouse with the button down can be used to select multiple popuations.
- The selected populations can be deleted by clicking the delete button ✕
- The carrying capacity (maximum population size) for the selected populations can be set by entering a value in the "Capacity" field to the right of the drawing area.
- The number of individuals that is present in the selected populations at the start of the simulation can be set by entering a value in the "Initial" box to the right of the drawing area.
- The carrying capacity and the initial capacity can both be set separately for males and females by switching on the "Show both sexes" button.
- The size of the whole metapopulation grid can be changed by adding rows or columns through clicking the + − buttons located at the four sides of the drawing area.
- Sometimes it may be handy to have an image at the background, that can be used as a basis for drawing the metapopulation. To enable this, select "Image" for the background and use the "Browse" button to locate the desired image file.

After setting the geographical lay-out of the metapopulation other simulation settings can be changed such as the breeding system of the simulated species, and, importantly for spatially explicit simulations, the migration of individuals between populations.

See also:

Loading a map
Creating simulations

Loading a metapopulation from a map

With Marlin it is possible to specify a metapopulation lay-out from an image file. This allows you to perform very realistic simulations of a species' distribution area. The pixels of the image will be interpreted as populations, with the possibility to set the carrying capacity or the initial capacity from the pixel's colour.

Alternatively, the lay-out of the metapopulation can drawn by hand by simply clicking in the drawing area to create populations (see here for more details on drawing a metapopulation by hand).

Preparing an image file:

- Use a standard image editor such as Photoshop, Graphic Converter, Gimp, or Acorn, to create an image file of the distribution area to be simulated.
- Be aware that small images work best as even moderately large images will give rise to very large, and therefore slow, simulations. For example, when this small image of the European Alps is used with a maximum of 100 individuals per population, the simulation contains 472 populations, with a total of nearly 40,000 individuals.

- The image should be saved in one of the following formats: tiff, png, gif, jpeg, or psd. Note that the compression used in certain formats (most notably jpeg) can cause a slight change in the image's colours.

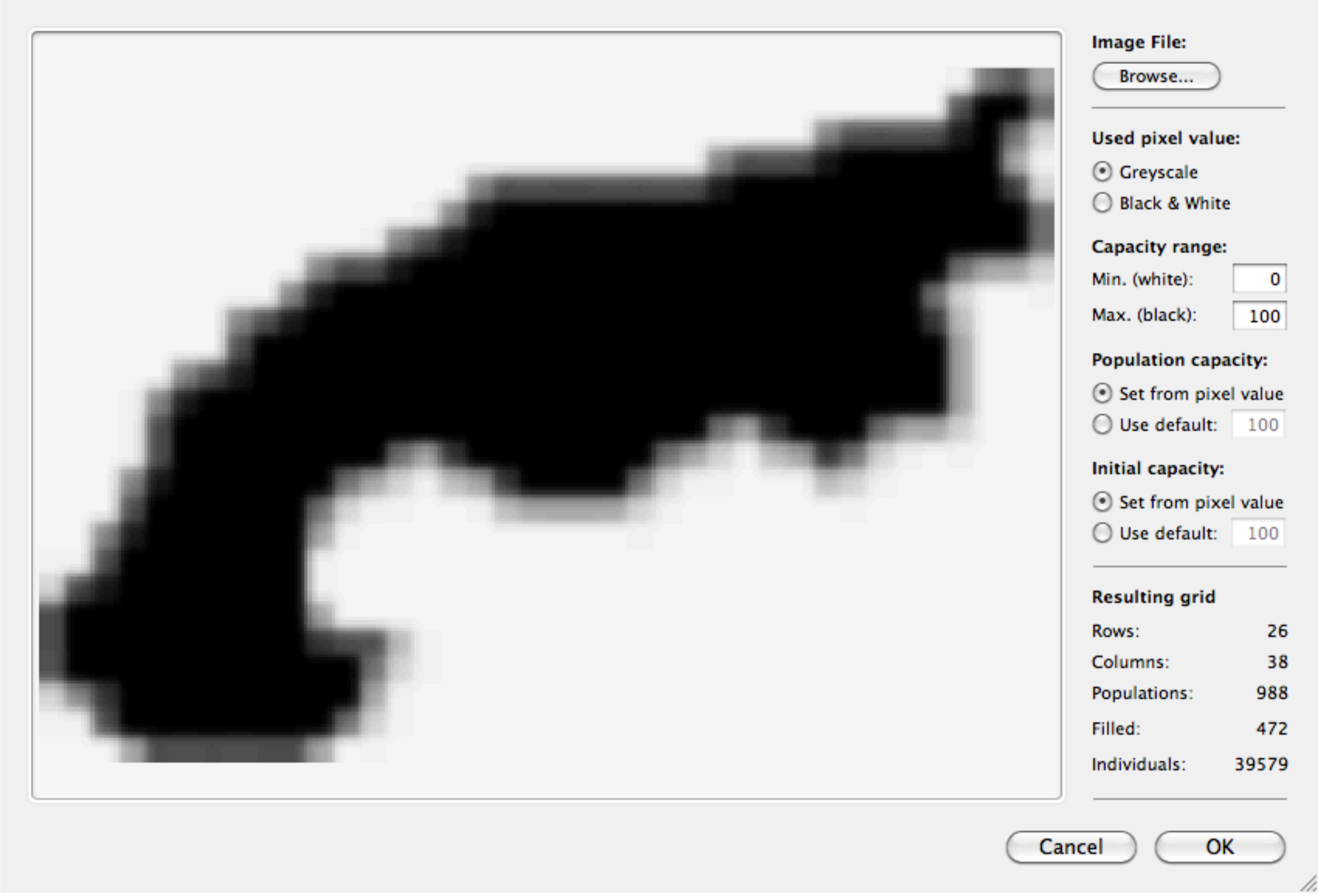


Image File:

Used pixel value:

● Greyscale

○ Black & White

Capacity range:

Min. (white): 0

Max. (black): 100

Population capacity:

● Set from pixel value

○ Use default: 100

Initial capacity:

● Set from pixel value

○ Use default: 100

Resulting grid

Rows:	26
Columns:	38
Populations:	988
Filled:	472
Individuals:	39579

To load an image file containing a map:

- If necessary, select the metapopulation editor by clicking on the "Metapopulation" item in the source list on the left-hand side of the window.
- Click the little globe button , or choose "Metapopulation from Image..." from the "Simulation" menu to start the map-loading dialog (see screenshot above).
- Drag the prepared image to the image field or click the Browse.. button to choose the image in an Open-dialog.
- Every pixel of the image will be interpreted as representing a population. The total size of the grid, the number of filled populations, and the total number of individuals can be checked under the "Resulting grid" section at the lower right corner of the dialog.
- The colours of the image can be interpreted either as greys or as black and white. Colours will be converted when necessary, though only internally without changing the image.
- The relation between the (converted) pixel colour and the population capacity can be changed by setting the capacity range. The range's minimum will be used for white pixels, the maximum will be used for black pixels. Grey pixels will result in populations with an intermediate capacity.
- For example, when the range has a minimum of 0 and a maximum of 60, a pixel with a 50% grey colour will give a population capacity of 30 individuals.
- Besides setting them from the pixel's colour value, the carrying capacity and/or the initial capacity can be fixed to a default value for every population. Setting the carrying capacity in this way only affects non-white pixels.
- Clicking OK will set the metapopulation grid to the currently selected run. If there was a previously existing metapopulation lay-out for the current run, this will be overwritten.

After setting the geographical lay-out of the metapopulation other simulations can be changed such as the breeding system of the simulated species, and, importantly for spatially explicit simulations, the migration of individuals between populations.

See also:

Drawing a metapopulation
Creating simulations

Simulation settings

Marlin presents a graphical interface for the program quantiNemo (Neuenschwander et al., 2008) that makes it easy to create simulations and change the settings. All available settings have been grouped into seven categories that can be found in the list on the left-hand side of the window. To change the settings for one of the categories simply click the name in the list and you will be presented with the available options.

By following the links below you will find an overview of available settings, but for a more extensive explanation you are advised to download a copy of the quantiNemo manual from: http://www2.unil.ch/popgen/softwares/quantinemo/

Available settings:

- Metapopulation: The spatial lay-out of the populations, plus the population parameters.
- Simulation: Generic settings that concern the simulation as a whole.
- Breeding: Breeding system and population growth/decline.
- Dispersal: Migration of individuals between populations.
- Neutral markers: Simulated genetic marker loci.
- Output: What to save and how often.
- Statistics: Summary statistics that will be calculated during the run.

See also:

Loading a map
Creating simulations

Metapopulation settings

The metapopulation section allows you to specify the spatial layout of the simulated metapopulation. Populations can be added by simply clicking an empty spot, after which the population's carrying capacity and initial size can be changed. The grid size can be changed by clicking the +- buttons on all four sides of the drawing area. For more information, see: drawing populations.

Besides drawing by hand, it is also possible to set the metapopulation lay-out from an image file that contains a geographical map by clicking the small globe button. For more information on this subject see: loading a map.

See also:

Simulation settings
Creating simulations

Overall simulation settings

The simulation section contains some generic settings that concern the simulation as a whole.

Available settings:

- *File name:* The name for the simulation; the name displayed in the simulation's tab in the tab-bar.
- *Number of generations:* The maximum length of the run; the actual run may be shorter than this when the metapopulation goes extinct.
- *Comments:* Here you can provide some extra comments about the simulation, such as the modelled species or scenario.
- *Extra settings:* Marlin provides only a subset of all parameters included in quantiNemo. In this box you can specify extra parameters in quantiNemo format, which will be included in the run. For more information, see the quantiNemo manual: http://www2.unil.ch/popgen/softwares/quantinemo/.
- *Random number seed:* By specifying a seed for the random number generator in this box, it is possible to exactly replicate a run. The default value of -1 will cause the seed to be taken from the system clock.

See also:

Simulation settings
Creating simulations

Breeding settings

The breeding section contains settings related to the breeding system and population growth/decline.

Available settings:

- *Offspring number model:* Determines the relationship between the breeding and the population size. "Carrying capacity" will keep the population size at carrying capacity, "Keep number" will keep the present population size, "Fecundity" will use the specified females' fecundity to determine the total number of offspring (possibly stochastically), and "Logistic growth" will use the specified growth rate (possibly stochastically).
- *Regulation model:* It is possible that after migration the population size exceeds the carrying capacity. This setting determines whether the individuals are randomly subsampled after migration to match the carrying capacity ("Random regulation") or not ("No regulation").
- *Mean fecundity:* The female's fecundity for use with the "Fecundity" offspring number models specified above.
- *Growth rate:* The population growth rate for use with the "Logistic growth" offspring number models specified above.
- *Extinction rate:* The probability that a population goes extinct in a generation.
- *Mating system:* Different possible mating systems of the simulated organism, some assume hermaphrodites, others two separate sexes.
- *Mating proportion:* Proportion of non-random mating, depending on the selected mating system. For example, when the mating system is set to selfing, a mating proportion of 0.05 will cause 5% of the offspring to be the result of selfing.
- *Sex ratio:* The ratio of males to females among the offspring for use with mating systems with two separate sexes. A value of 1 indicates a balanced sex ratio.
- *Number of mating males:* In many species, not all males have access to females (e.g. when males have harems). This allows setting how many males in a population are available for mating.

For more information, download the quantiNemo manual from: http://www2.unil.ch/popgen/softwares/quantinemo/

See also:

Simulation settings
Creating simulations

Dispersal settings

The dispersal section contains settings that specify the model of migration of individuals between populations. Currently, six different models are available to specify the dispersal:

Available dispersal models:

1. Stepping-stone model: Standard population genetics model where dispersal only takes place to neighbouring populations.
2. Island model: Standard population genetics models with equal dispersal to all other populations.
3. Two-phase model: Combination of the two standard models, where a part of the migrants disperse via an Island model and another part via the Stepping-stone model.
4. Exponential-power kernel: Dispersal of individuals follows an exponential power kernel, allowing you to set the shape of the kernel.
5. Double exponential-power kernel: A combination of two independent kernels, e.g. allows you to specify that a part of the migrants follow a fat-tailed kernel and another part a thin-tailed kernel.
6. Custom matrix: Allows you to use a custom dispersal matrix by pasting in a tab-delimited matrix.

See also:

Simulation settings
Creating simulations

Stepping-stone dispersal

This section of the dispersal settings dialog involves the standard Stepping-stone model, where individuals can only disperse to neighbouring populations.

Available settings:

- *Dispersal rate:* The fraction of individuals in a population that will disperse in a generation.
- *Sex specific dispersal:* Determines whether separate dispersal rates should be given for males and females.
- *Border model:* This determines how the border of the metapopulation acts. With a circle/torus border, the opposite borders are connected, creating a borderless landscape, without edge effects. With reflective borders, the individuals cannot migrate beyond the borders and bounce back into the landscape. With absorbing borders, individuals that move beyond the border die a agonizing death in the void that exists outside the landscape...
- *Number of neighbours:* The number of neighbouring populations to which individuals can migrate (either 4 or 8).

See also:

Dispersal methods
Simulation settings
Creating simulations

Island dispersal models

This section of the dispersal settings dialog involves the standard population genetics Island model of dispersal, where individuals can disperse to all other populations.

Available settings:

- *Dispersal rate:* The fraction of individuals in a population that will disperse in a generation.
- *Sex specific dispersal:* Determines whether separate dispersal rates should be given for males and females.
- *Model:* The dispersal model to use. There is a choice between a Migrant-pool model where all individuals migrate on their own, and a Propagule model where a part of the migration can take place in the form of propagules; clusters of comigrating individuals that come from the same natal population.
- *Propagule probability:* Under the propagule-pool Island model, this parameter specifies the fraction of migration that takes place in the form of propagules.

See also:

Dispersal methods
Simulation settings
Creating simulations

Two-phase dispersal model

This section of the dispersal settings dialog provides a combination of the two standard models, where a part of the migrants disperse via an Island model and another part via the Stepping-stone model. In general, with this model it makes most sense to set the Stepping-stone dispersal higher than the Island-model dispersal.

Available settings:

- *Stepping-stone dispersal:* The fraction of individuals in a population that can disperse only to neighbouring populations.
- *Island-model dispersal:* The fraction of individuals in a population that can disperses to any other population in the landscape (here, neighbouring populations are excluded).
- *Empty populations:* If there are many populations in the system with a carrying capacity of zero, many migrants may get lost in those empty patches. Checking this switch will cause these migrants to be placed back into the source population.
- *Number of neighbours:* For the Stepping-stone dispersal, the number of neighbouring populations to which individuals can migrate (either 4 or 8).

See also:

Dispersal methods
Simulation settings
Creating simulations

Exponential-power kernel

This section of the dispersal settings dialog involves an exponential power kernel (Robledo-Arnuncio & Garcia 2007), allowing you to set the shape of the kernel. This kernel defines the probability of migrating to another location at coordinates x and y relative to the current location as:

$$p(x,y) = \frac{b}{2\pi a^2\,\Gamma(2/b)}\exp\left(-\left(\frac{\sqrt{x^2+y^2}}{a}\right)^b\right).$$

Here, *a* is the kernel's range parameter, and *b* is the kernel's shape parameter. When *b*=2, the kernel is Gaussian (normal), and when *b*=1, the kernel is exponential. Lower values of *b* give a fatter-tail to the kernel.

Available settings:

- *Dispersal rate:* The fraction of individuals in a population that will disperse in a generation. Note that, depending on the kernel shape, the realised dispersal rate may be much lower than this.
- *Kernel width (a):* The kernel's range parameter, parameter *a* in the above equation.
- *Tail shape (b):* The kernel's shape parameter, parameter *b* in the above equation.
- *Empty populations:* If there are many populations in the system with a carrying capacity of zero, many migrants may get lost in those empty patches. Checking this switch will cause these migrants to be placed back into the source population.

See also:

Dispersal methods
Simulation settings
Creating simulations

Double exponential-power kernel

This section of the dispersal settings dialog allows you to specify two independent kernels which will be combined to define the dispersal matrix. Both kernels are from the exponential-power family (Robledo-Arnuncio & Garcia 2007) which defines the probability of migrating to another location at coordinates x and y relative to the current location as:

$$p(x,y) = \frac{b}{2\pi a^2\,\Gamma(2/b)}\exp\left(-\left(\frac{\sqrt{x^2+y^2}}{a}\right)^b\right).$$

Here, *a* is the kernel's range parameter, and *b* is the kernel's shape parameter. When *b*=2, the kernel is Gaussian (normal), and when *b*=1, the kernel is exponential. Lower values of *b* give a fatter-tail to the kernel.

Available settings:

- *First kernel / Second kernel:* The kernel settings can be separately be set for the two kernels. Setting the dispersal rate of one of the two to 0 will give the same kernel as the single exponential power kernel.
- *Dispersal rate:* The fraction of individuals in a population that will disperse in a generation. Note that, depending on the kernel shape, the realised dispersal rate may be much lower than this.
- *Kernel width (a):* The kernel's range parameter, parameter *a* in the above equation.
- *Tail shape (b):* The kernel's shape parameter, parameter *b* in the above equation.
- *Empty populations:* If there are many populations in the system with a carrying capacity of zero, many migrants may get lost in those empty patches. Checking this switch will cause these migrants to be placed back into the source population.

See also:

Dispersal methods
Simulation settings
Creating simulations

Custom dispersal matrix

This section of the dispersal settings dialog allows you to paste in a custom dispersal matrix. The matrix needs to be in a tab-delimited format (e.g. pasted from Excel). The matrix needs to be of size *num_populations* * *num_populations* with the elements d_{rc} of the matrix specifying the migration rate from population *r* to population *c*. All values on a row must sum to one, meaning no individuals can get lost.

To use a custom matrix, simply copy it from your favorite spreadsheet application into the text-box.

See also:

Dispersal methods
Simulation settings
Creating simulations

Neutral marker settings

The neutral marker section contains settings that specify the type of genetic markers that are simulated. At the moment, only neutral genetic markers are available.

Available settings:

- *Number of loci:* The number of neutral genetic marker loci that will be simulated.
- *Maximum number of alleles:* The maximum number of alleles that can be present at a single locus. Should be lower than 256. Note that depending on the mutation rate and other parameters, the actual number of alleles can be much lower.
- *Initial variation:* Determines the allele frequencies at the start of the simulations. The populations can either be initialised with maximally polymorph loci with respect to the allele frequencies ("Polymorph"), or all population can be fixed for the same allele ("Monomorph").
- *Mutation model:* The model used to mutate an allele from one state to another. Under a K-alleles model an allele will mutate from one state to a randomly chosen other state in the above defined range. This model is suitable for simulating allozymes, or, when the maximum number of alleles is set very low, for SNPs. Under Single Step Mutation model the current state of the allele is taken into account, and it will mutate only to either the next or the previous state. This model is applicable to microsatellites.
- *Mutation rate:* The probability that a mutation will occur at a locus per individual per generation.
- *Mutation shape:* The mutation rates can be varied among loci by drawing their mutation rates from a gamma distribution. This parameter sets the shape of the gamma distribution, while the above specified mutation rate determines the mean of the gamma distribution. When the shape parameter is set to 0, all loci will have the same mutation rate.

For more information, download the quantiNemo manual from: http://www2.unil.ch/popgen/softwares/quantinemo/

See also:

Simulation settings
Creating simulations

Output settings

The output section contains settings that specify how often and in which format the simulated marker data will be saved during the run.

Available settings:

- *Save genotypes:* The genetic data can be saved in either standard Fstat format (Goudet 1995) or in extended Fstat format, where for every individual genealogical information is appended after the genetic marker data. Alternatively, no genetic data can be saved.
- *Sex-specific output:* When there are two separate sexes, the output can be generated for only one of the two.
- *Age-specific output:* Generations are non-overlapping. Nevertheless, the age class of the output can be specified, giving output for only adults, only offspring, or both.
- *Log time:* The interval in generations at which the genotypes will be saved. For example a log time of 100 will cause the genetic data to be saved every 100 generations. Setting this value too low can cause the simulations to take up an inordinate amount of disk space.
- *Analyses to perform:* Marlin can automatically perform some analyses on the simulated datasets, which will be plotted in the results section (see here for more information).
- *Allele frequencies:* Allele frequencies will be calculated and are available for plotting. This option is only available when the maximum number of alleles is set to two.
- *K-means clustering:* Uses a simulated annealing approach to cluster populations into groups, and also determines which number of clusters best describes the data (Meirmans & Goudet in prep.). Can be very slow for large simulations.
- *Principal Components Analysis:* Uses a Principal Coordinates Analysis to ordinate the between-populations variation on a small number of orthogonal axes (Goudet 1999). Can be very slow for large simulations.

See also:

Simulation settings
Creating simulations

Statistics settings

The statistics section allows you to specify which summary statistics will be computed during the run, which are grouped into a number of categories. Some categories are available for adults as well as offspring, others are only available for adults. Given below are the descriptions of all statistics calculated per category, followed in brackets by their abbreviations, which are used for plotting. Note that some statistics (e.g. coancestry) can take a very long time to calculate and will slow down the simulations.

Available settings:

- *Demography:* Total number of individuals (nbInd), total number of females (nbFem), total number of males (nbMal), mean number of individuals per inhabited population (meanInd), mean number of females per inhabited population (meanFem), mean number of males per inhabited population (meanMal), sex ratio (sexRatio), number of inhabited patches (nbPops).
- *Patch extinction:* Proportion of populations that went extinct in the last generation (ext.rate).
- *Fecundity:* Mean realised female fecundity (fem.meanFec), variance of female fecundity (fem.varFec), mean realised male fecundity (mal.meanFec), variance of male fecundity (mal.varFec).
- *Kinship:* Proportion of full sibs (fsib), proportion of paternal half-sibs (phsib), proportion of maternal half-sibs (mhsib), proportion of non-sib (nsib), proportion of selfed offspring (self).
- *Migration:* Number of emigrants per population (emigrants), number of immigrants per population (immigrants), number of residents per populations (residents), the effective immigration rate (immigrate), the number of colonisers per extinct populations (colonisers), the effective colonisation rate of extinct populations (colon.rate).
- *Genotype coancestry:* Within population coancestry (theta), between population coancestry (alpha), within population within female coancestry (thetaFF), within population within male coancestry (thetaMM), within population between sexes coancestry (thetaFM), coancestry within full-sibs (coa.fsib), coancestry within paternal half-sibs (coa.phsib), coancestry within maternal half-sibs (coa.mhsib), coancestry within non-sib (coa.nsib).
- *Genetic diversity:* Mean number of alleles per locus over whole metapopulation (nbAll), mean number of alleles per locus per population (meanAll), number of fixed loci over whole metapopulation (nbFixLoc), mean number of fixed loci per population (meanFixLoc), observed heterozygosity (ho), expected heterozygosity (hs), total expected heterozygosity (ht) (Nei & Chesser 1983).
- *Fstats (Nei & Chesser):* global Fst (fst), global Fis (fis), global Fit (fit) (Nei & Chesser 1983).
- *Fstats (Weir & Cockerham):* global Fst (fst.wc), global Fis (fis.wc), global Fit (fit.wc) (Weir & Cockerham 1984).
- *Log time:* The interval in generations at which the statistics will be calculated. For example a log time of 10 will cause the statistics to be calculated every 10 generations.

For more information, download the quantiNemo manual from: http://www2.unil.ch/popgen/softwares/quantinemo/

See also:

Simulation settings
Creating simulations

Saving simulation settings

There are many situations in which you may want to store the settings of a simulation; e.g. to repeat a simulation multiple times, to share the settings with a colleague, or to distribute simulation settings among students in a practical. In Marlin there are three different ways in which you can store the settings of a simulation that you have created:

1. Storing them as a preset.
2. Exporting them to a .marlin settings file.
3. Exporting them to a .ini file for use with quantiNemo.

Presets

Presets are simulation settings that are stored with the Marlin preferences and can directly be used to create new simulations. Marlin has several default presets that represent interesting demographic scenarios. If you want to make your own preset from simulation settings that you specified, simply click the Presets button in the toolbar and provide a name in the dialog. To manage presets, click and hold the button until a menu pops up and choose the "Manage presets.." item.

To create a new run from an existing preset, click and hold the New Run button in the toolbar until a menu pops up, and then choose the preset's name from the list. Alternatively, you can choose the preset using the "New Run from Preset" item of the "File"-menu

Importing and exporting settings

Simulation settings can also be saved to a file which can then be transferred to other computers or emailed. These files have the extension .marlin and can only be read by Marlin. To save your settings in such a way, choose the "Save Settings..." item from the "File" menu. You will be presented with a standard Save dialog that allows you to specify the name and the location of the settings file. A settings file will also automatically be created in the results folder when a simulation is run (see: Where do the files go?).

To create a new run from saved settings, double click the settings file in the Finder, or choose the "Open Settings..." item from the "File" menu.

quantiNemo .ini files

The calculational backbone of Marlin is provided by the program quantiNemo (Neuenschwander et al., 2008). QuantiNemo is a command line program that reads the settings from a normal text file with the extension .ini. Normally, Marlin uses an embedded copy of quantiNemo to perform its simulations. However, it may be useful to run a separate copy of quantiNemo with the settings specified using Marlin, for example when running a large number of simulations on a computer cluster.

In Marlin, the settings can be saved as a .ini file by clicking the Preview/.ini button on the toolbar. This will show a new window that displays the settings in quantiNemo format. To save these settings, click the "Save As..." button at the bottom of this window. A .ini file will also automatically be created in the results folder when a simulation is run (see: Where do the files go?). Note that Marlin cannot read .ini files.

See also:

quantiNemo
What happens during a run?
Creating simulations

Running simulations

Starting runs

If you simply want to run a single simulation, you can start the run by either clicking the Start-button in the toolbar, or by choosing "Start Run" from the "Simulation" menu.

If you want Marlin to perform multiple runs, running these all simultaneously can require a lot of resources from your computer, resulting in very slow calculations. If the simulations are very big and require a lot of RAM-memory, running several runs at one time can even bring down your whole system. In such cases it is handier to use Marlin's queueing system to run simulations serially; new simulations can be added to the queue and will only start once the others are finished.

The run queue:

1. To add a run to the queue, choose "Add to Queue" from the "Simulation" menu.
2. When the number of running simulations is lower than the number of processor cores in the computer, the simulation is started immediately, otherwise it is added to the queue.
3. As soon as one of the running simulations has finished, the first simulation in the queue is started.
4. This process is continued until all queued simulations have run.
5. Simulations that have been added to the queue, and that have not yet started, can be removed from the queue by choosing "Remove from Queue" from the "Simulation" menu.
6. An overview of all the queued runs can be seen in the Runs drawer, which can be opened by choosing "Toggle Runs Drawer" from the "Window" menu.

See also:

What happens during a run?
Running simulations

What happens during a run?

For performing the actual simulations, Marlin uses the program quantiNemo developed by Sam Neuenschwander et al. (2008). A full copy of quantiNemo is included within Marlin itself, and this copy is run in the background when a users starts a simulation. Once quantiNemo has finished the simulation, Marlin takes over again and analyses the produced data and produces chart and maps. I have tried to make this process as transparant to the user as possible, so that it is possible to use Marlin without any knowledge of how to use quantiNemo. Nevertheless, it may be useful to know what is exactly happening during a run.

Marlin runs, step by step:

1. The user clicks the Start button or chooses Start Run from the Simulation menu (see starting runs for more info).
2. Marlin creates a temporary folder with the name of the run in ~/Library/Application Support/Marlin.
3. Marlin gathers all settings that have been specified by the user and writes these to a quantiNemo input file called quantiNemo.ini. This file contains all the information on the specified metapopulation layout and, if necessary, also a full dispersal matrix between all pairs of populations.
4. The embedded copy of quantiNemo is started as a separate background process and is told to read the settings file.
5. All files produced by quantiNemo are placed inside the temporary folder; the log-files, the files with the genetic marker data, and the files with the calculated statistics.
6. All messages from quantiNemo are intercepted by Marlin and displayed in the Results>Log section of the list on the left-hand side of the window. These messages are also used to update the progress indicator in the window's toolbar.
7. When quantiNemo is finished, Marlin starts several analyses. On computers with multiple processor cores, the analyses are distributed over all available cores to speed up the calculations.
8. The files with statistics that have been produced by quantiNemo are read and interpreted in order to produce the plots in the Results>Statistics section of the list on the left-hand side of the window.
9. All files with neutral genetic marker data are read and the specified analyses are performed on the genetic data. The results are used to produce the maps in the Results>Neutral Markers section of the list on the left-hand side of the window.
10. When Marlin quits, all temporary data will be deleted. This means that the results of your simulations will be lost unless you saved them to a different location (don't fear, you will be asked whether you want to save the results upon quit).

See also:

quantiNemo
Starting runs
Where do the files go?
Running simulations

quantiNemo

Although it may seem like Marlin is performing the simulations itself, it is in fact a separate program called quantiNemo that is doing all the hard work (Neuenschwander et al. 2008). Marlin simply provides a user interface for an embedded copy of quantiNemo that is run as a background process when a user starts a simulation.

More parameters

QuantiNemo is an amazingly powerful piece of software and the functionality provided by Marlin barely scratches the surface. Among others, quantiNemo is able to simulate quantitative genetic traits, as well as the underlying genetic loci and the selection acting on these traits. Furthermore, quantiNemo can take linkage between the loci into account and it is even possible to specify a complete linkage map. If you are interested in doing some large simulation projects, it is very likely that quantiNemo provides the functionality you need. For more information, and to download quantiNemo go to: http://www2.unil.ch/popgen/softwares/quantinemo/

QuantiNemo vs. Marlin

Despite quantiNemo's tremendous power, it has some limitations. For instance, all simulation parameters have to be set through a separate text file. While this provides great flexibility, the syntax and large number of available parameters may be bewildering to novel users. Setting the parameters through a text file also makes it difficult to specify detailed geographic scenarios. Furthermore, analysing and visualising the results of a simulation takes a significant effort, often requiring the use of at least two other programs. This is where Marlin comes in. With Marlin, the parameters are set through its intuitive user interface, realistic geographic scenarios can simply be read from a map, and once a simulation is finished, all produced files are directly analysed and plotted.

Symbiosis

Possibly the best work flow for doing a large simulation study is a combination of the two programs. When I am doing simulations myself I use Marlin for developing the simulations, i.e. to create the metapopulation lay-out, to set all life-history characteristics. I then perform several test runs and use Marlin's plotting capacities to check the results to see whether anything needs to be changed. For instance, I may have set the mutation rate too low for the population size, resulting in a lack of genetic variation. I may have to go to several iterations in order to get results that are in line with what I want to simulate. I then export the settings as a quantiNemo .ini-file and make some additional changes if necessary. Then I use these settings to run quantiNemo on a cluster computer to generate a large number of repeats.

See also:

Citing Marlin
What happens during a run?
Running simulations

Viewing the results

Analyses

The raw output of the simulations consists of files with genetic marker data in Fstat format, which can be saved for several generations (see output settings). Marlin can automatically read all the produced files and perform some simple analyses on them. The results of these analyses can be visualised on the metapopulation map. This provides an immediate visual representation of the distribution of the genetic variation over the metapopulation, and the strength of the population structure.

The analyses will be performed on all files with genetic data that have been produced during the run. If you choose to produce a lot of data files (e.g. producing a file once every few generations) the analyses may take a long time. This is especially the case for metapopulations that contains very large numbers of individuals.

Available analyses:

- *Summary statistics (always on):* Marlin will calculate for every non-empty population in the metapopulation several summary statistics (Nei 1987): the population size, observed heterozygosity, expected heterozygosity, and the inbreeding coefficient Fis. If the files have been saved in extended Fstat format (with genealogical information appended) Marlin also calculates the average age, the sex ratio, and the population's immigration rate.
- *Allele frequencies (optional):* For every population and for every locus the allele frequencies. This option is only available if the neutral markers have been set to have only two alleles, since drawing the allele frequencies quickly becomes unmanageable for multiple loci with several alleles per locus.
- *k-Means clustering (optional):* This is an algorithm that clusters the populations into *k* different groups, based on the allele frequencies (Meirmans & Goudet in prep.). The algorithm is run for different values of *k* and afterwards the best overall clustering is picked based on a summary statistic called pseudo-F. The method uses a simulated annealing approach to find the best clustering, i.e. the clustering with the highest among-groups sum of squares (analogous to maximising F_{ct}). Some settings for this analysis can be specified in the Marlin preferences panel.
- *Principal Components Analysis (optional)* Principal Components Analyses is a method that can reveal the internal structure of the data in a way which best explains the variance in the data. Marlin performs a PCA on the population allele frequencies (Goudet 1999); when the results are plot on the map, a population's axis score is represented by its colour on the map. The default number of retained PCA-axes in 5, but this can be changed in the Marlin preferences panel. Note that the PCA can be extremely slow for data sets with a large overall number of alleles.

Which of the analyses will be performed after the simulation can be changed in the "Output" section of the simulation settings. Furthermore, it is possible to redo the analyses by including the desired analyses and then clicking the button labeled "Redo analyses" in the "Maps" section of the results. Note that by doing this, the results of previously included analyses will be lost.

See also:

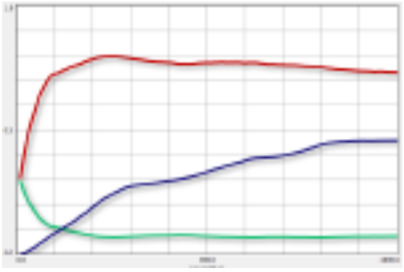
Output settings
Plotting
Viewing the results

Plotting results

Marlin can plot the values of many different summary statistics directly when the run is finished. This allows you for example to track the value of Fst as a function of the number of generations. Other combinations are also possible, such as plotting the relationship between Fis and the population size, or between the heterozygosity and the population size. Furthermore, Marlin can map the results, for example producing a spatial map of the heterozygosity following a colonization of an empty habitat. Maps can be created for different generations, allowing you to see the development of the population structure over time.

Both types of graphs are available under the Results-section of the source list in the left-hand side of the window. More information on these two types of graphs can be found by following the links below:

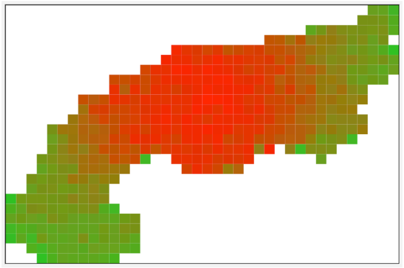
XY-plots of overall summary statistics.



On the XY-plots implemented in Marlin, a single variable can be set on the X-axis of the plot and several variables can be set on the Y-axis. Which variables are available for plotting depends on the statistics that were included for calculation during the run.

- XY-plots

Maps of per-population statistics.



Marlin can also present the results of the performed analyses as a map that is based on the simulation's metapopulation lay-out.

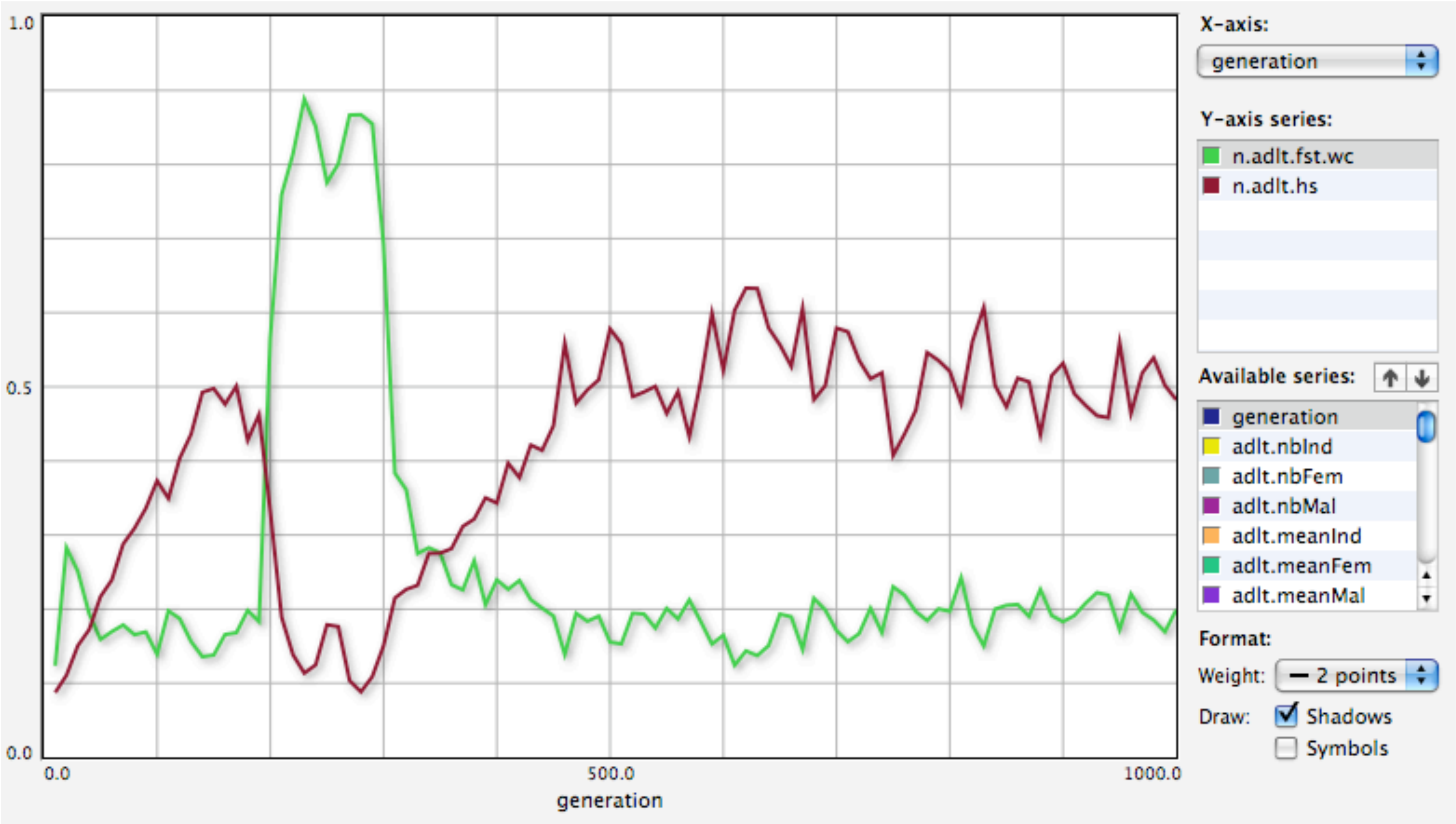
- Maps

See also:

Statistics
Analyses
Viewing the results

XY-plots

On the XY-plots implemented in Marlin, a single variable can be set on the X-axis of the plot and several variables can be set on the Y-axis. The scale of the axes cannot be changed manually, this is done automatically to fit the selected variables. Furthermore, some basic graphical settings can be changed, such as the stroke width, the shadow and the plotting of symbols. Which variables are available for plotting depends on the statistics that were included for calculation during the run (see statistics-settings).



X-axis: generation

Y-axis series:
- 🟩 n.adlt.fst.wc
- 🟥 n.adlt.hs

Available series:
- generation
- adlt.nbInd
- adlt.nbFem
- adlt.nbMal
- adlt.meanInd
- adlt.meanFem
- adlt.meanMal

Format:

Weight: 2 points

Draw: ☑ Shadows ☐ Symbols

Changing XY-plots:

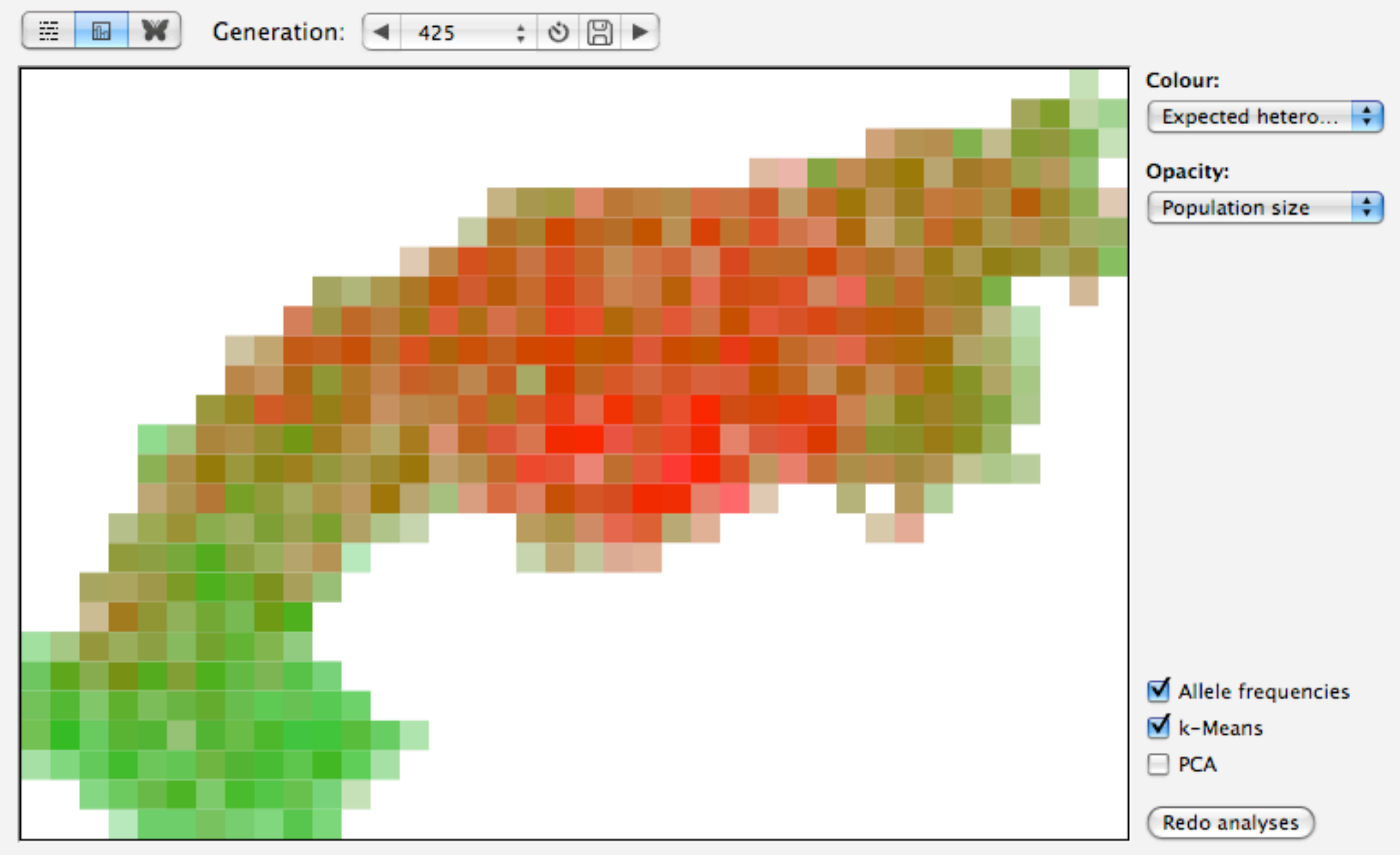
- The X-axis variable can be selected through the pop-up menu labeled "X-axis", located on the right-hand side of the window. By default, the X-axis is set to "generation".
- The current Y-axis variables are shown in the table labeled "Y-axis series". The variables can be changed using the buttons with up and down arrows located below this table. ↑↓
- To remove a variable, select it in the table and click the "down" arrow.
- To add a series, select it in the table labeled "Available series" and click the "up" arrow.

See also:

Plotting
Maps
Viewing the results

Maps

Marlin can also present the results of the performed analyses as a map that is based on the simulation's metapopulation layout. During the run, files containing the genetic marker data are produced at specified time-intervals. Afterwards, during the analysis-phase of the run, these files are analysed and the results are plot onto the metapopulation map, resulting in a separate map for every produced file.



Generation: 425

Colour: Expected hetero...

Opacity: Population size

☑ Allele frequencies
☑ k-Means
☐ PCA

Changing maps:

- The analysis results that are represented by the map colours can be changed using the pop-up button labaled "colour" on the right hand side of the window.
- On the map, every population is represented by a small rectangle, and the colour of this rectangle represents the population's value for the selected analysis results. For example, in the graph above this text the colour has been set to represent the expected heterozygosity, and populations with a low heterozygosity have a green colour while populations with a high heterozygosity have a red colour.
- In addition, it is possible to specify a result to represent the opacity of the used colour. For example, using the population size to set the opacity will make small populations appear lighter, and therefore will draw less attention to them.
- You can use the series of buttons labeled "Generation" to navigate through the maps created for all files with genetic marker data, corresponding to different generations. . The pop-up menu and the arrows allow you to select the generations, while the stopwatch-button will animate the map over the generations. The little disk icon allows you to save the currently displayed map.
- It is possible to redo the analyses by including the desired analyses and then clicking the button labeled "Redo analyses". Note that by doing this, the results of previously included analyses will be lost.

See also:

Plotting
XY-plots
Viewing the results

Where do the files go?

During a run,many files with output are produced, containing the calculated statistics, the genetic marker data, and the results of the analyses performed on these data. Obviously, all these files have to be stored somewhere. When a run is started, a folder carrying the name of the run is created in ~/Library/Application Support/Marlin/Runs. During the run, all produced files are saved to this folder. To avoid cluttering your hard drive with old simulations, Marlin cleans up its Application Support folder when the program is quit.

If you want to save the results to a different location, use the command Save Result As... in the File menu, or click the Save button in the program's toolbar. Depending on the simulation's settings, saving the Results can take quite some time as all produced files have to be copied to the new location. Note that the Results are not saved as a single file, but as a folder containing several files and folders.

Contents of the results folder:

- *SimulationSettings.marlin:* A file with the simulations settings that can be read by Marlin (see Presets). Double-clicking this file in the Finder will open the Results in Marlin. Note that if some of the other files are missing from the folder, the results cannot be opened and a new run is created with these settings instead.
- *QuantiNemo.ini:* A file with the simulations settings that can be read by quantiNemo, the program that provides the calculational backbone of Marlin (see quantiNemo). This file can be modified by any text editor and used to perform additional runs using quantiNemo from the command line.
- *quantinemo.log:* A log file produced by quantiNemo. This file contains some basic information on the run, such as the start and end time, the location of the produced output etc.
- *MarlinRun.log:* Another log file produced by quantiNemo. This file contains a replicate of the used settings, including some settings that have not been specified by Marlin. This file can also be used as an input file for quantiNemo, instead of the quantiNemo.ini file.
- *Statistics:* A folder that contains the statistics calculated by quantiNemo during the run. Of most interest here is the file named "MarlinRun_mean.txt", which contains the values of the calculated statistics at the time-intervals specified in the settings. The file named "MarlinRun_legend.txt" gives a short description of the calculated statistics.
- *Neutral_Marker_Genotypes:* A folder that contains the files with genetic marker data in Fstat format, produced by quantiNemo. The files all have a name with the format MarlinRun_g025.dat, where the number (here "025") indicates the generation during which the file was saved.
- *Analyses:* A folder that contains the results of the analyses that Marlin has performed on the produced genetic marker data. The naming of the files follows that of the files with marker data, but they have the extension ".gdv". The files contain a table of which the rows represent the populations and the columns the performed analyses. Missing data (e.g. due to empty populations) is coded either as -10 or -99999. Appended to the bottom of the file are three rows with the minimum, maximum, and type identifier that are used by Marlin to produce the maps. Note that these minimum and maximum values may not correspond to the actual minimum and maximum of the column, but rather are the extreme values used for the map's colour scale.

See also:

What happens during a run?
Viewing the results

Miscellaneous

How to cite Marlin:

No paper describing Marlin has been published yet, in the mean time you can cite Marlin like this:

- Meirmans, P.G. 2009. Marlin, a program for performing geographically realistic simulations. Available from http://www.patrickmeirmans.com/software.

If you use Marlin for a publication, please make sure that you also cite quantiNemo, which provides the calculational backbone of the program. QuantiNemo can be cited as follows:

- Neuenschwander, S., F. Hospital, F. Guillaume, J. Goudet. 2008. QuantiNemo: an individual-based program to simulate quantitative traits with explicit genetic architecture in a dynamic metapopulation. Bioinformatics 24:1552-1553.

Most of the analyses have been described in papers written by other people than me; please remember to cite these papers as well.

See also

Cited References

GenoDive

Marlin has the option to open the produced files with genetic marker data using the program GenoDive (Meirmans & Van Tienderen 2004). You can then use GenoDive to perform additional analyses on your data.

GenoDive is a Mac-only program for population genetic analyses and features many different types of statistical inferences. Included inferences are among others: a flexible Analysis of Molecular Variance, estimation of standardised coefficients of population differentiation, k-means clustering of populations using a simulated annealing approach, assigning genotypic identity (clones) to individuals, testing for clonal reproduction, testing Hardy-Weinberg equilibrium, calculation of the hybrid index for individuals, and different types of Mantel tests.

Opening results in GenoDive:

- After the run has finished, click "Neutral"-markers in the "Results" category in the source list on the left hand side of the window.
- Using the pop-up menu or the two arrows, select the generation for which you want to display the data in GenoDive.
- Click the GenoDive button in the window's toolbar.
- If you have a copy of GenoDive on your computer, it will open the selected file. If no copy of GenoDive is found, you will be asked whether you want to download it.

For more information on GenoDive, or to download the program, go to: http://patrickmeirmans.com/software.

See also:

Analyses
Plotting
Viewing the results

License

Marlin license

Copyright (c) 2007-2009, Patrick Meirmans

Distribution of Marlin is allowed and encouraged, but only as a complete package. Use and enjoy Marlin at your own risk. The author can never be kept responsible for any possible damage, loss of time or materials, resulting from the use of Marlin or incorrect information provided by Marlin:

THIS SOFTWARE IS PROVIDED ''AS IS'' AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

quantiNemo

Marlin contains an embedded, compiled, copy of the program quantiNemo, written by Sam Neuenschwander. QuantiNemo and its source code have been published under the GNU General Public License, and are available from http://www2.unil.ch/popgen/softwares/quantinemo/

Included source code

Marlin contains source code written by third party authors and kindly made available for general use. The licenses and copyrights of the included source code can be found below.

- **UKPrefsPanel**

  Copyright (c) 2003-07 by M. Uli Kusterer.

  You may redistribute, modify, use in commercial products free of charge, however distributing modified copies requires that you clearly mark them as having been modified by you, while maintaining the original markings and copyrights.

- **DSClickableURLTextField**

  Copyright (c) 2006 - 2007 Night Productions, by Darkshadow. All Rights Reserved.

  May be used freely, but keep my name/copyright in the header. There is NO warranty of any kind, express or implied; use at your own risk. Responsibility for damages (if any) to anyone resulting from the use of this code rests entirely with the user.

- **PSMRolloverButton**

  Copyright (c) 2005, Positive Spin Media

  All rights reserved.

  Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

  Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

  Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

  Neither the name of Positive Spin Media nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

  THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

- **PSMTabBarControl**

  Copyright (c) 2005, Positive Spin Media

  All rights reserved.

  Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

  Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

  Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

  Neither the name of Positive Spin Media nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

  THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

- **CTGradient**

  Copyright (c) 2006 Cotingent.

  Some rights reserved: http://creativecommons.org/licenses/by/2.5/

- **CTGradient**

  Copyright (c) 2006 Chad Weider.

  Some rights reserved: http://creativecommons.org/licenses/by/2.5/

- **RSVerticallyCenteredTextField**

  Copyright (c) 2006 Red Sweater Software

  Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

  The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

  THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

- **Sparkle**

  Copyright (c) 2006 Andy Matuschak

  Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

  The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

  THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

- **Apple Help template**

  Copyright (c) 2005 Andy Matuschak

  Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

  The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

  THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

- **MTwist**

  Copyright (c) 2001 Geoff Kuenning.

  Licensed under the GNU Lesser General Public License: http://www.gnu.org/licenses/lgpl-3.0.txt

See also

Version History
Requirements
How to cite Marlin

Cited References:

References cited in these Help-files:

- Goudet, J. 1995. FSTAT (Version 1.2): A Computer Program to Calculate F-Statistics. Journal of Heredity 86:485-486
- Goudet, J. 1999. PCAGEN, A Computer Package which Performs Principal Component Analysis (PCA) on Gene Frequency Data. Available from URL: http://www2.unil.ch/popgen/softwares/pcagen.htm.
- Meirmans, P.G., and P. H. Van Tienderen. 2004. GENOTYPE and GENODIVE: two programs for the analysis of genetic diversity of asexual organisms. Molecular Ecology Notes 4:792-794.
- Meirmans, P.G., and J. Goudet. Amova-based determination of the number of clusters in population genetic data. in prep.
- Nei, M. and R.K. Chesser. 1983. Estimation of fixation indexes and gene diversities. Annals of Human Genetics 47:253-259
- Nei, M. 1987. Molecular Evolutionary Genetics. Columbia University Press, New York
- Neuenschwander, S., F. Hospital, F. Guillaume, J. Goudet. 2008. QuantiNemo: an individual-based program to simulate quantitative traits with explicit genetic architecture in a dynamic metapopulation. Bioinformatics 24:1552-1553.
- Robledo-Arnuncio, J.J. and C. García. 2007. Estimation of the seed dispersal kernel from exact identification of source plants. Molecular Ecology 16:5098-5109.
- Weir, B.S. and C.C. Cockerham. 1984. Estimating f-statistics for the analysis of population structure. Evolution 38:1358-1370.

See also

How to refer to Marlin

Version History

Previous versions of Marlin

Marlin 1.4:

- Added more references to Help-files.
- More interface fixes.
- Added more types of image files for exporting charts.

Marlin 1.3:

- Uses new version of quantiNemo.
- Fixed several interface glitches introduced with Snow Leopard.
- Now works correctly on Tiger.

Marlin 1.2:

- Fixed several bugs that caused excessive memory use.
- Changed the way in which presets and runs results are stored, making them more accessible.
- Updated contact information.

Marlin 1.1:

- Added a queueing system to help run simulations in series.
- Included a new version of quantiNemo, which fixes several bugs.
- Separate dispersal settings for Island model and Stepping-Stone model.
- Statistics and analyses results can now also be displayed as tables.
- Changed some warning icons and texts.
- Can now set background image when manually drawing the metapopulation.
- Animated preferences window.
- Fixed bug in enabling icons in toolbar.
- Support for Growl notifications.
- Can change colours of y-axis series in statistics plots.

Marlin 1.0:

- First public release.

See also

License
Requirements
How to cite Marlin

Requirements

Though Marlin should be able to run just fine on any reasonable modern Mac, there are some specific requirements.

What you need to run Marlin:

- The minimum version of Mac OS X that Marlin can run on is OS 10.4.0 (Tiger). Given that Tiger is five years old at the time of writing, I hope that this is not much of a problem to anyone.
- Marlin is compiled as a so-called "universal binary" which means that it can run both on PPC and on Intel-based Macs. Unfortunately, my PPC-based iBook died a while ago, so that most testing has been done on Intel-based Macs running OS 10.5
- Furthermore, the quantiNemo backbone runs natively in 64-bits when supported by the hardware. This allows larger simulations (when sufficient RAM is available) and simulations generally run slightly faster in 64-bit mode.
- The amount of RAM-memory required depends mostly on your simulations. Simulations with a lot of large populations will a lot of RAM. For such simulations, it is not advisable to write the data in Fstat format very often as that will also take up a lot of RAM.

See also

Starting runs
How to refer to Marlin