

AntMap Ver. 1.1

**A software for constructing genetic linkage maps by
ant colony optimization algorithm**

Developed by
Hiroyoshi IWATA

Last update: May 19, 2005

AntMap ver. 1.1

Copyright © 2004 National Agriculture and Bio-oriented Research Organization.

Source code and executables of AntMap are available under GNU general public license (GPL) at <http://cse.naro.affrc.go.jp/iwatah/antmap/index.html>. AntMap comes with ABSOLUTELY NO WARRANTY, and you are welcome to redistribute it under certain conditions; see Appendix A (GNU General Public License) for details.

AntMap Website:

<http://cse.naro.affrc.go.jp/iwatah/antmap>

Correspondence to the developer:

Hiroyoshi Iwata (Ph.D)

Department of Information Science and Technology

National Agricultural Research Center

3-1-1 Kannondai, Tsukuba, Ibaraki 305-8666, Japan

Tel: +81-29-838-7025

Fax: +81-29-838-8551

E-mail: iwatah@affrc.go.jp

Web page: <http://cse.naro.affrc.go.jp/iwatah>

Table of Contents

1. Brief Overview
2. Ant Colony Optimization (ACO)
3. Locus Ordering by ACO algorithm
4. System requirements
5. Installation
6. Input File Format
7. Short Tutorial
8. AntMap Window
9. AntMap Menus
10. AntMap Options
11. How to cite AntMap
12. References
13. Appendix A: GNU General Public License (GPL)
14. Appendix B: Copyright notice of the C++ class (MTRand) for Mersenne Twister random number generator

Brief Overview

Locus ordering is an essential procedure in genome mapping. When the number of loci is large, it is quite difficult to determine the optimum order with an exhaustive search of all possible orders. The problem of searching for the optimum order has been recognized as a special case of the traveling salesman problem (TSP), i.e., given a set of cities and distances for each pair of them, find a roundtrip of minimal total length visiting each city exactly once. In recent years, Ant Colony Optimization (ACO) (Dorigo and Stutzle 2004), which is a set of algorithms inspired by the behavior of real ant colonies, has been successfully used to solve discrete optimization problems, such as TSP. We developed a novel system based on ACO for locus ordering in genome mapping (Iwata and Ninomiya *in preparation*). In our system, loci and absolute value of log likelihood (or recombination fraction) between loci were regarded as TSP cities and distance between cities, respectively. We tested the system using a simulated segregation population, and found it is highly efficient for linkage grouping as well as locus ordering in genome mapping (Iwata and Ninomiya 2004).

To commoditize our newly-developed system, we developed software named AntMap for constructing linkage map by the system. AntMap performs segregation test, linkage grouping and locus ordering, and constructs a linkage map quite rapidly and nearly automatically. Rapidity of the algorithm based on ACO enables us to conduct a bootstrap test of estimated order. With the aid of this software, researchers can save their time and labor, and can obtain a linkage map whose reliability is indicated by bootstrap values. Another advantage of AntMap is the fact that AntMap is open source; that is, source code and executables of AntMap are available under GNU General Public License (GPL) (see Appendix A for details). Java and C++ objects that code our newly-developed system will be utilized effectively for other applications as well as AntMap.

Ant Colony Optimization (ACO)

Sorry! Now this section is on a back-burner.

New Algorithm for Locus Ordering

See Iwata and Ninomiya (in prep.).

System Requirements

- Windows, Linux, Solaris or Mac OS.
- Java 2 Platform Standard Edition (J2SE) Java Runtime Environment (JRE) (ver. 1.4 or higher).

If you have no J2SE JRE (ver. 1.4 or higher) already installed on your system, you should install the JRE to execute AntMap on your system. To install JRE, please see <http://java.sun.com/j2se/1.4/download.html>.

Installation

1. Download the zipped file named “AntMap.zip”, which contains all the AntMap files. The file is available at <http://cse.naro.affrc.go.jp/iwatah/antmap>.

The files contained in “AntMap.zip” are as follows:

AntMap.jar executable jar file

(It can be executed on any OS on which the JRE is installed)

AntMap.exe native executable file for Windows

AntMap-linux native executable file for Linux

AntMap-solaris native executable file for Solaris

AntMap-macx native executable file for Mac OS X

org_agmodel_bioinfo_antmap_AntColonyJNI.dll

dynamic link library used by AntMap

(It is used only when AntMap is executed on Windows)

sample.raw a sample file

AntMapSource.jar archive file which contains source code

license.txt text file which describes GNU GPL

Manual.pdf **this file** (user’s guide for AntMap).

2. Extract the folder named “AntMap” from the “AntMap.zip”.
3. You can put the extracted folder anywhere you want.
4. (Optional) If your OS is Windows, make a short cut to “AntMap.exe” on your desktop for convenience.

Input File Format

Input file format of AntMap is identical to *.raw files required by MapMaker (Lander et al. 1987). A concise description of the format of *.raw files can be found in <http://www.rfcgr.mrc.ac.uk/Registered/Help/mapmaker/>.

AntMap can analyze data derived from progeny of several types of crosses, including:

- (1) F2 intercross
- (2) F2 backcross (e.g., BC1)
- (3) Recombinant inbred lines by self-mating
- (4) Doubled haploid lines

The first line of input file indicates the crossing types as:

```
data type xxxx
```

where `xxxx` should be one of the followings:

```
f2 intercross  
f2 backcross  
ri self  
dh
```

The current version of AntMap does not support two types of cross, F3 intercross by self-mating (`f3 self`) and recombination inbred lines by sib-mating (`ri sib`), which are supported by MapMaker EXP.

Short Tutorial

Step 0: Start AntMap



Fig. 1

When you use AntMap on Windows, start AntMap with double-clicking the “AntMap” icon (Fig. 1). For other operating systems (i.e., platforms), See below.

Box 1.

Linux and Solaris

Before executing “AntMap-linux” or “AntMap-solaris”, you should change mode of these files to be executable. To do that, type

```
chmod 755 AntMap-xxxx
```

on your command line system (“xxxx” should be “linux” or “solaris”). After changing the mode of files, you can execute AntMap by clicking the “AntMap-linux” or “AntMap-solaris” file icon.

Mac OS X

Note that you can execute “AntMap-macx” from the command line, but cannot execute by clicking the “AntMap-macx” file icon.

AntMap can also be executed by using the executable jar file “AntMap.jar” on any platforms (Linux, Solaris and Mac OS as well as Windows). To execute the jar file, run:

```
java -jar AntMap.jar
```

on your command line system. Some platforms may have bindings already set up such that you can execute the jar file just by clicking on the “AntMap.jar” file icon, which will run the command line equivalent. Note that you should change mode of the jar file to be executable when you are on “Linux” or “Solaris” platforms as described in Box 1.

Step 1: Open an input file.

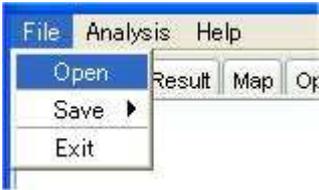


Fig. 2

Open an input file in MapMaker format (*.raw) through “File-Open” menu (Fig. 2). Here, open “sample.raw” contained in the “antmap” folder.

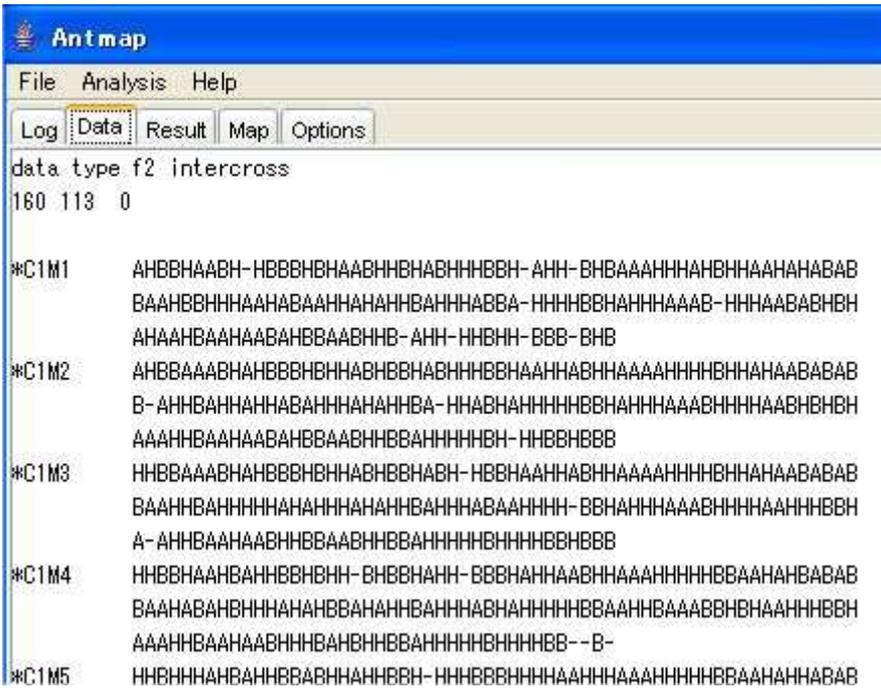


Fig. 3

After opening the file, contents of the file will appear in the “Data” panel (Fig. 3).

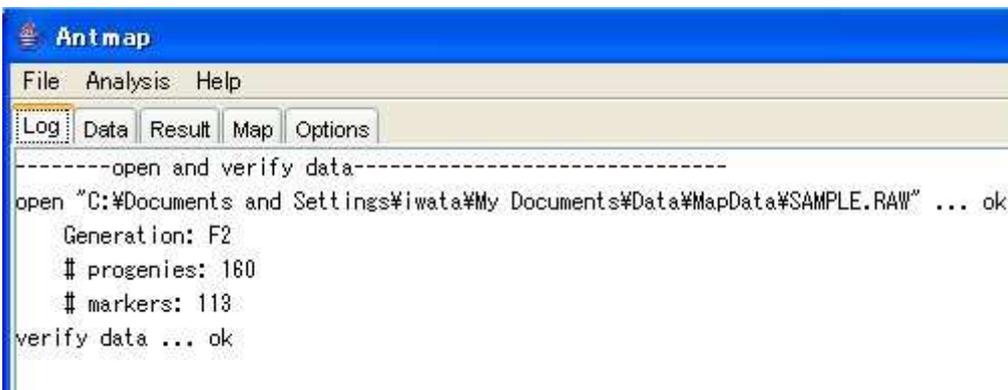


Fig. 4

Click the “Log” tab, and you can see a summary of the input data (Fig. 4).

Step 2: Segregation ratio test.

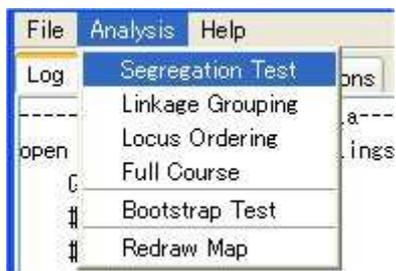
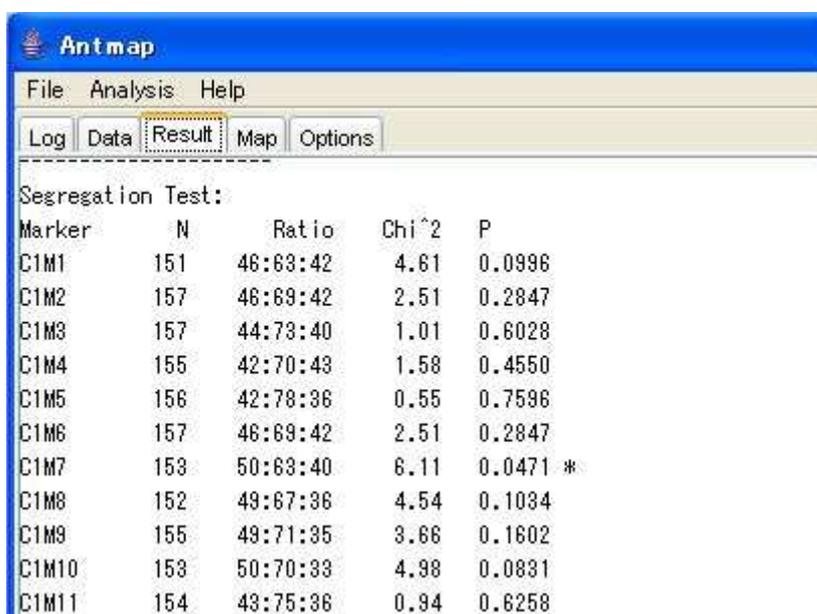


Fig. 5

Select "Segregation Test" from the "Analysis" menu (Fig. 5). Then you can see the results of segregation ratio tests in the "Result" panel (Fig. 6).

A screenshot of the 'Result' panel in the Antmap software. The panel displays the results of a Segregation Test for 11 markers (C1M1 to C1M11). The results are presented in a table with columns for Marker, N, Ratio, Chi^2, and P. The P-value for C1M7 is marked with an asterisk (*).

Marker	N	Ratio	Chi ²	P
C1M1	151	46:63:42	4.61	0.0996
C1M2	157	46:69:42	2.51	0.2847
C1M3	157	44:73:40	1.01	0.6028
C1M4	155	42:70:43	1.58	0.4550
C1M5	156	42:78:36	0.55	0.7596
C1M6	157	46:69:42	2.51	0.2847
C1M7	153	50:63:40	6.11	0.0471 *
C1M8	152	49:67:36	4.54	0.1034
C1M9	155	49:71:35	3.66	0.1602
C1M10	153	50:70:33	4.98	0.0831
C1M11	154	43:75:36	0.94	0.6258

Fig. 6

Step 3: Linkage grouping

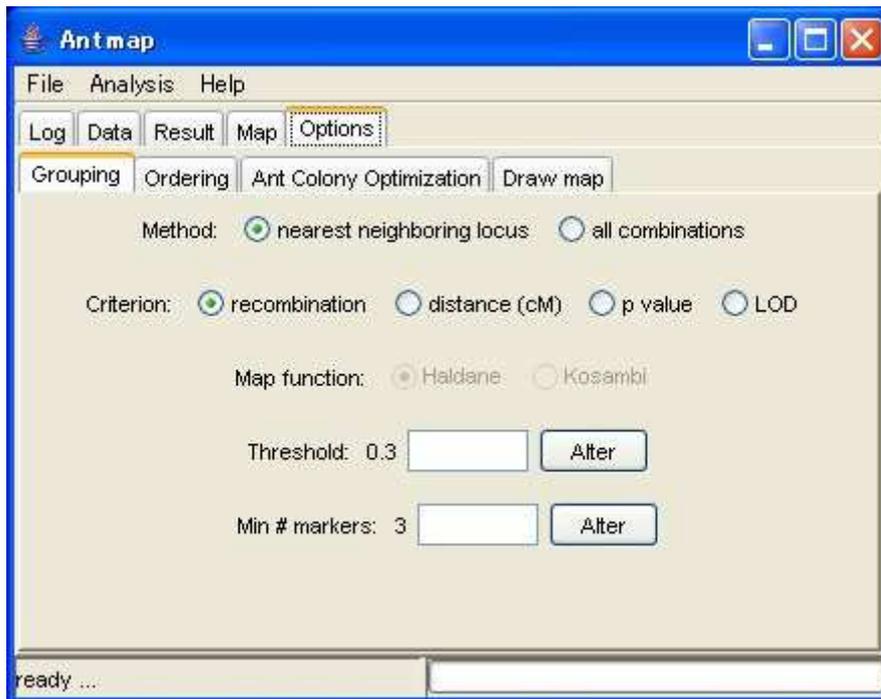


Fig. 7

Click the “Options” tab. Then you can see the “Grouping” option panel (Fig. 7).

You can choose one of the two grouping methods: “nearest neighboring locus” and “all combinations”. The former makes a group by sequentially combining a locus which shows the smallest recombination value against it. This algorithm has been implemented by MAPL (Ukai et al. 1991). The latter will produce similar results with “group” command of MapMaker.

You can also choose the grouping criterion, threshold value and the minimum number of markers for a single group.

Here, we will keep these options unchanged except for the threshold value.

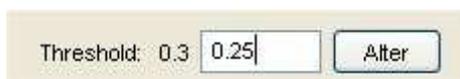


Fig. 8

Type “0.25” into an input area, and push the “Alter” button. Then you can change the threshold value from 0.3 to 0.25.

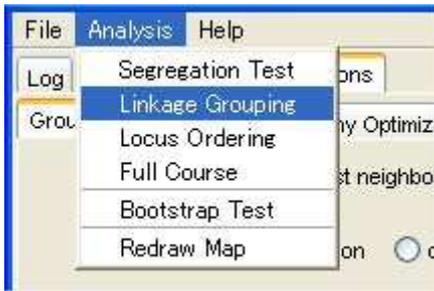


Fig. 9

Select the “Linkage Grouping” from the “Analysis” Menu. Then you can see the results of linkage grouping in the “Result” panel (Fig. 10).

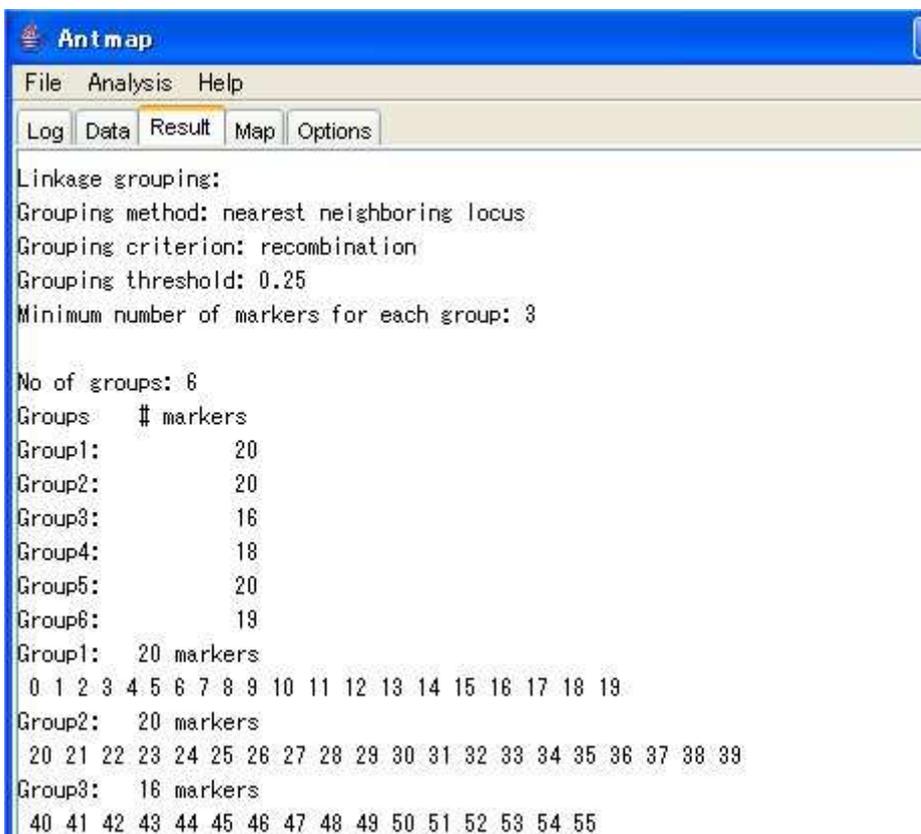


Fig. 10

When you analyze your data, you may not be able to achieve a good separation of markers to linkage groups from the start. In such a case, please find a good set of the threshold value, criterion and method through try-and-errors.

Step 4: Locus ordering

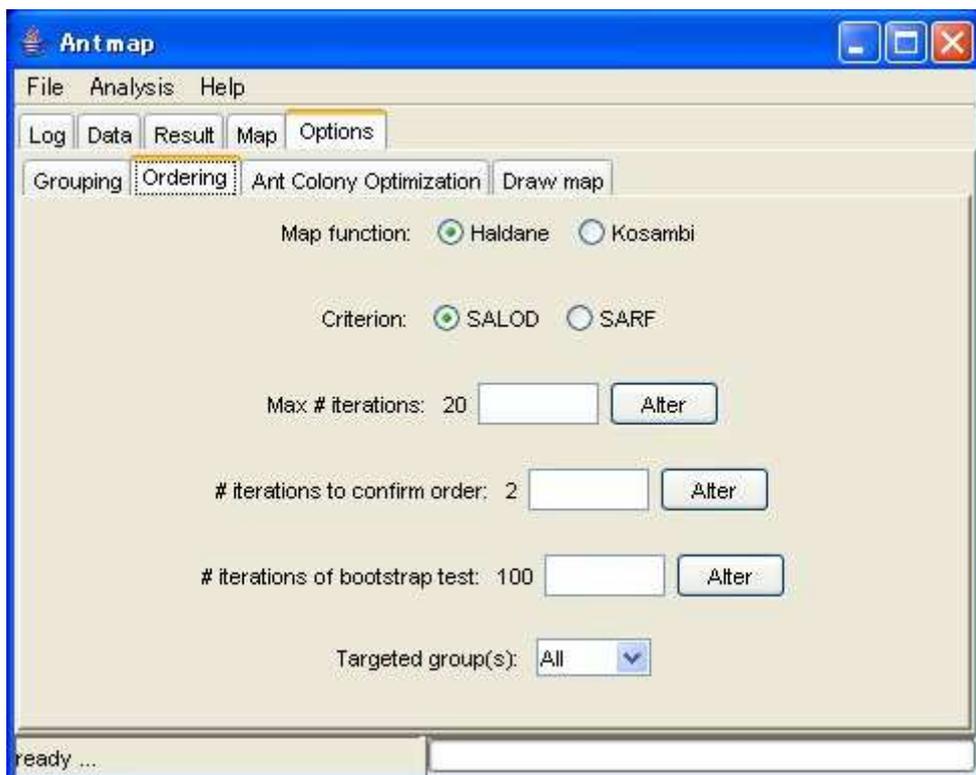


Fig. 11

Click the “Options” tab, and click the “Ordering” tab. Then you can see the “Ordering” option panel (Fig. 11).

In the locus ordering, you can choose one of the two criteria: log-likelihood and “SARF”. “SARF” is an abbreviation for “Sum of Adjacent Recombination Fractions” (Liu 1998). AntMap will search a locus order which maximizes log-likelihood or minimizes “SARF”.

You can also choose the maximum number of iterations and the number of iterations to confirm order. The details of these options are given in the “AntMap Options” section.

A map function for calculating a map distance between adjacent markers can be selected from “Haldane” (Haldane 1919) or “Kosambi” (Kosambi 1944) functions.

Here, we will keep these options unchanged.

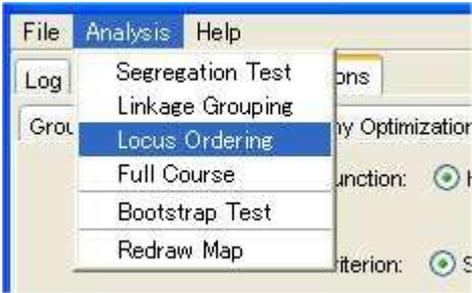


Fig. 12

Select the “Locus Ordering” from the “Analysis” Menu. Then you can see the results of locus ordering in the “Result” panel (Fig. 13).

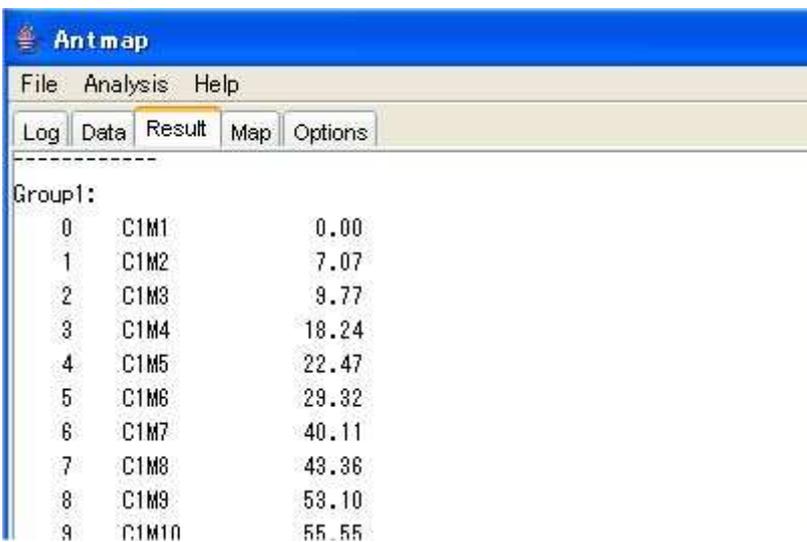


Fig. 13

You can also obtain a graphic of linkage map in the “Map” panel (Fig. 14).

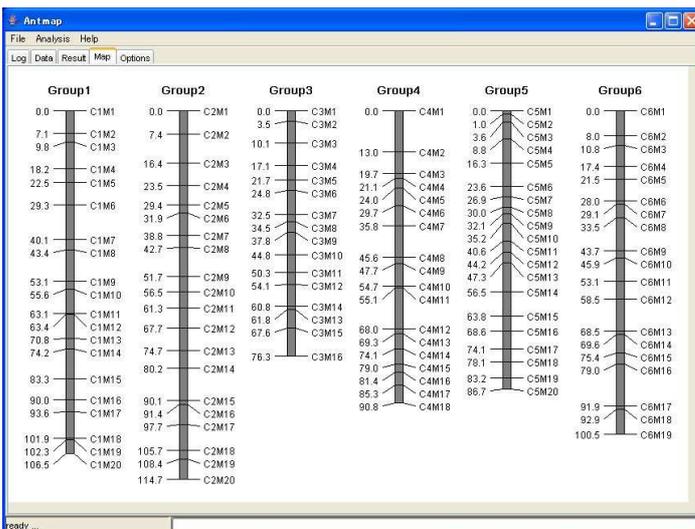


Fig. 14

Step 5: One-step mapping

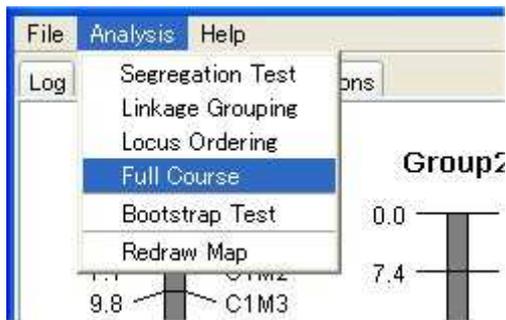


Fig. 15

Select “Full Course” from the “Analysis” Menu. Then, you can overall process from segregation ratio test (Step 2) to locus ordering (Step 4) at once.

Step 6: Redraw a linkage map

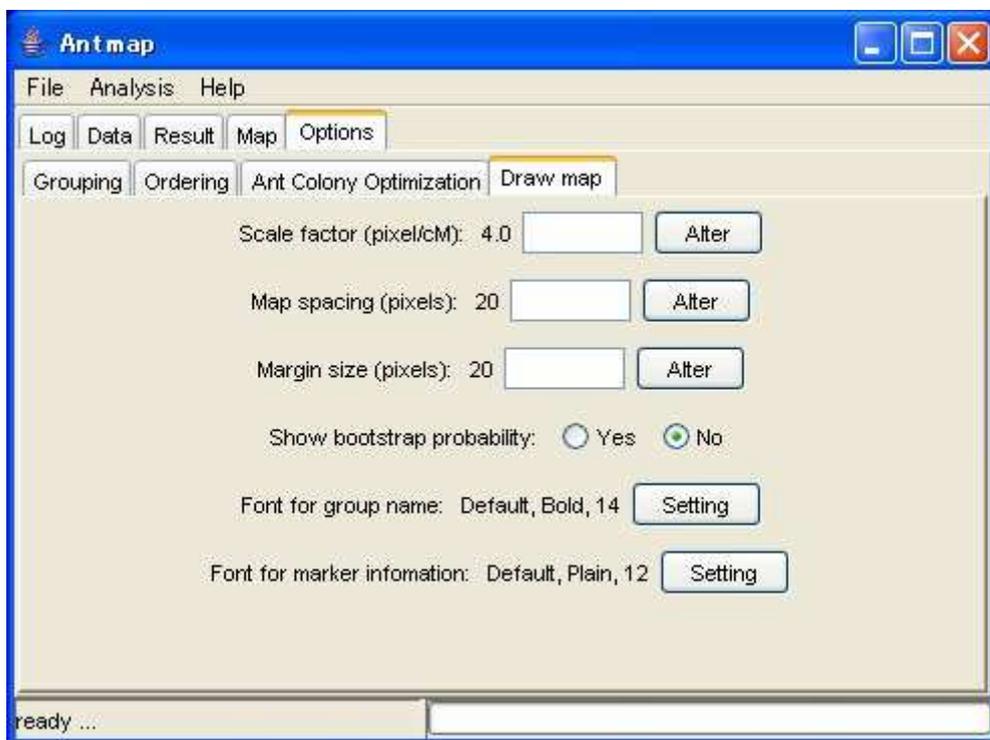


Fig. 16

Click the “Options” tab, and click the “Draw map” tab. Then you can see the “Draw map” option panel (Fig. 16).

Here, we will change the “Scale factor” option. Drawing size of linkage map can be changed through this option. Here, type “2” into an input area, and click the “Alter” button (Fig. 17).

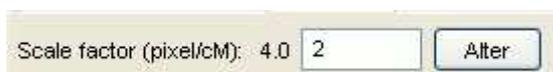


Fig. 17

After changing the option value from 4 to 2, select “Redraw Map” from the “Analysis” menu. Then, you can obtain a smaller linkage map than one obtained previously (Fig. 18).

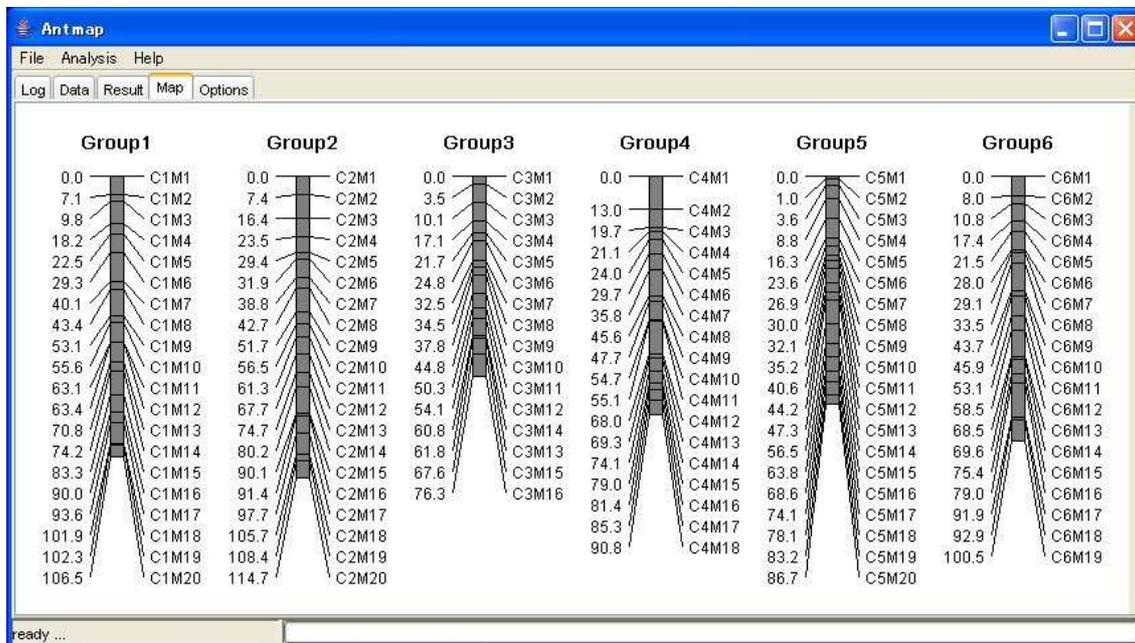


Fig. 18

Step 7: Bootstrap test for locus order

You can evaluate the reliability of estimated locus order by using bootstrap test. Bootstrap test (or bootstrapping) is a method for estimating the sampling distribution of an estimator by resampling with replacement from the original sample. In a bootstrap test, a random sample of size n is drawn from the original sample of size n , and estimates are obtained from the random sample. After repeating (iterating) this operation many times (e.g., 100-1000 times), the stability of estimates (e.g., standard error or confidence interval of estimators) is evaluated. For the details of bootstrap test, please see a good textbook such as Manly (1998). In the bootstrap test for locus order, we can obtain probability that a locus is located at its estimated order (Liu 1998).

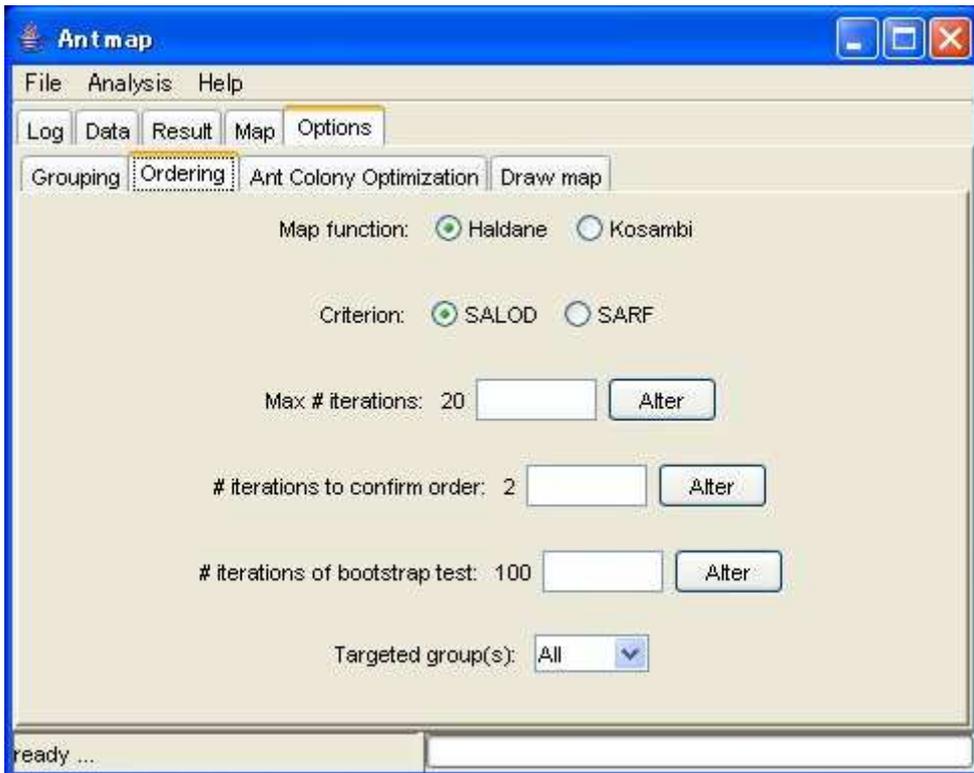


Fig. 19

Click the “Options” tab, and click the “Ordering” tab. Then you can see the “Ordering” option panel (Fig. 19).

You can change the number of iterations (repeats) of bootstrapping. To get a good estimate of percentage of correct locus order, 100 may be sufficient.

You can also choose a group which is targeted in the bootstrap test. Here, we will choose only Group3 to save our time (Fig. 20).

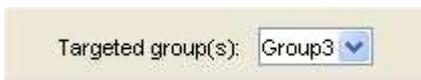


Fig. 20

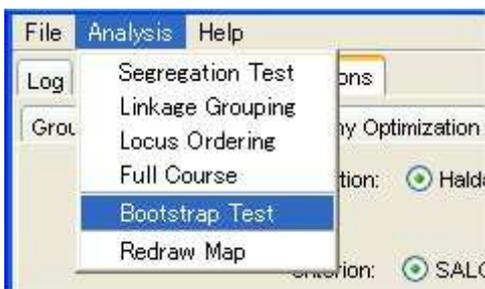


Fig. 21

Select the “Bootstrap Test” from the “Analysis” Menu (Fig. 21). Then you can see

the results of bootstrap test for locus order in the “Result” panel (Fig. 22).

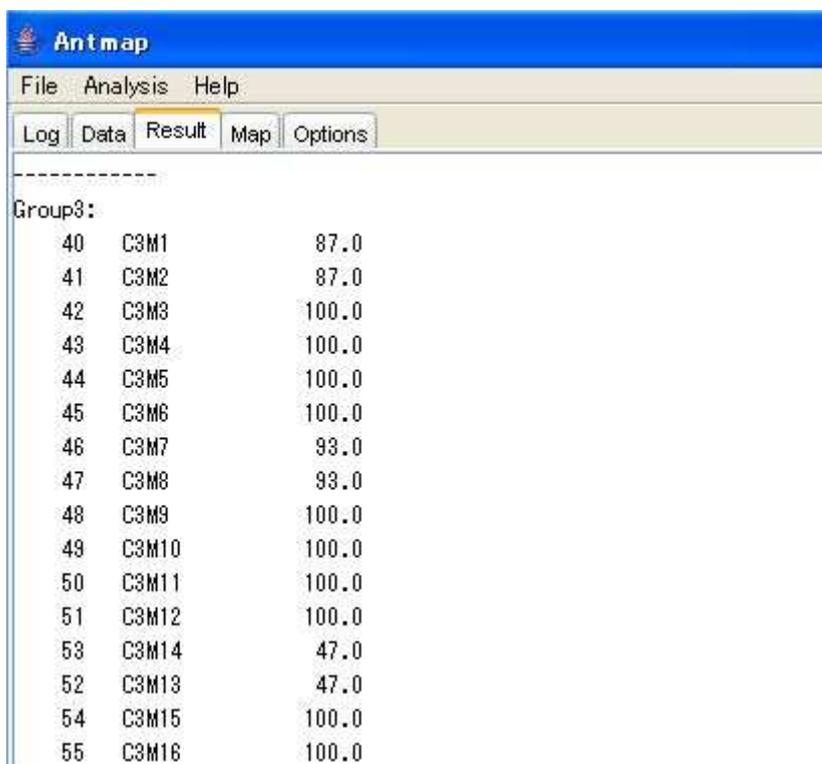


Fig. 22

You can also obtain a graphic of linkage map with bootstrap values in the “Map” panel (Fig. 23).

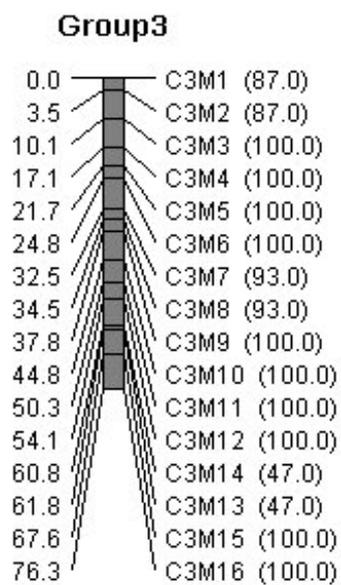


Fig. 23

FYI: The bootstrap test for all linkage groups may take long time even by high-end PC. Thus, you have better set your computer to perform this test at your lunch time or after going home.

Step 8: Save results of linkage mapping

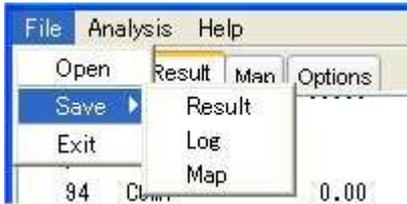
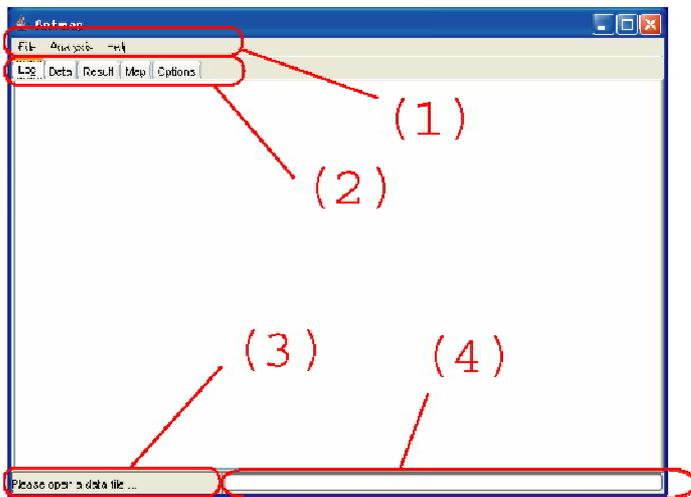


Fig. 24

You can save information in “Result”, “Log” and “Map” panels through the “Save” submenu in the “File” Menu. The information in “Result” and “Log” is saved as a text file. The information in “Map” (i.e., a graphic of linkage map) is saved as a JPEG (*.jpg) file.

AntMap Window



(1) AntMap Menus

See the next section.

(2) Tabs for Panels

1. "Log" Panel

Analysis log is output in this panel.

2. "Data" Panel

Input data are shown in this panel.

3. "Result" Panel

Results of linkage mapping are shown in this panel.

4. "Map" Panel

A graphical image of estimated linkage map is shown in this panel.

5. "Options" Panel

Setting of options for linkage mapping can be checked and changed through this panel.

(3) Status Indicator

Status of the software is indicated here as a text message.

(4) Progress Bar

A bar which indicates progress in calculation requested by the user.

AntMap Menus

AntMap operates interactively through menu selections made by a user. Explanations for the menus are as follows.

File menu

- | **Open**
 Open an input file
- | **Save**
 - | Result
 Save contents in the "Result" panel.
 - | Log
 Save contents in the "Log" panel.
 - | Map
 Save a graphic in the "Map" panel.
- | **Exit**
 Quit AntMap

Analysis menu

- | **Segregation Test**
 Conduct segregation ratio test
- | **Linkage Grouping**
 Conduct linkage grouping
- | **Locus Ordering**
 Conduct locus ordering
- | **Full Course**
 Conduct overall process of linkage mapping
 (segregation test, linkage grouping, locus ordering)
- | **Bootstrap Test**
 Conduct bootstrap test for locus order
- | **Redraw Map**
 Redraw a graphics of linkage map

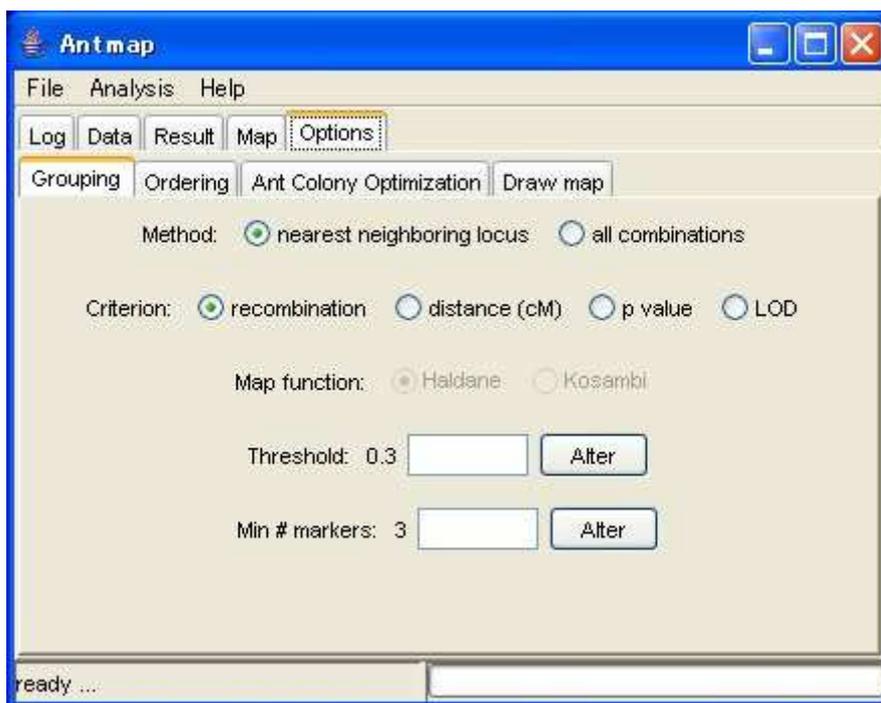
Help menu

- | **About**
 Show the version information of AntMap.
- | **About GPL**
 Show GNU General Public License (GPL)

AntMap Options

A user can check setting of options in linkage mapping and change them through the “Options” panel. In this panel, there are four child-panels which can be switched by clicking tabs. Explanations for the options are as follows.

“Grouping” options



Method – Method for linkage grouping

You can choose one of the two grouping methods: “nearest neighboring locus” and “all combinations”. The former makes a group by sequentially combining a locus which shows the smallest recombination value against it. This algorithm has been implemented by MAPL (Ukai et al. 1991). The latter will produce similar results with “group” command of MapMaker.

Criterion – Criterion for linkage grouping

You can choose one of the four grouping criteria: “recombination”, “distance (cM)”, “p value” and “LOD”. “recombination” and “distance (cM)” are recombination value and map distance (in centi-Morgan) between two loci, respectively. “p value” is a p value for chi-square test statistic under the assumption of no linkage between two loci and no segregation distortion for both loci. “LOD” is a LOD score which is defined as the following formula:

$$\text{LOD score} = \log \left[\frac{\text{probability of the observed data with a given recombination value}}{\text{probability of the observed data under no linkage}} \right].$$

Map function – Map function to be used

When you choose “distance (cM)” as the criterion, you can choose one of two mapping functions for calculating map distance. “Haldane” is the function derived by Haldane (1919) and “Kosambi” derived by Kosambi (1944).

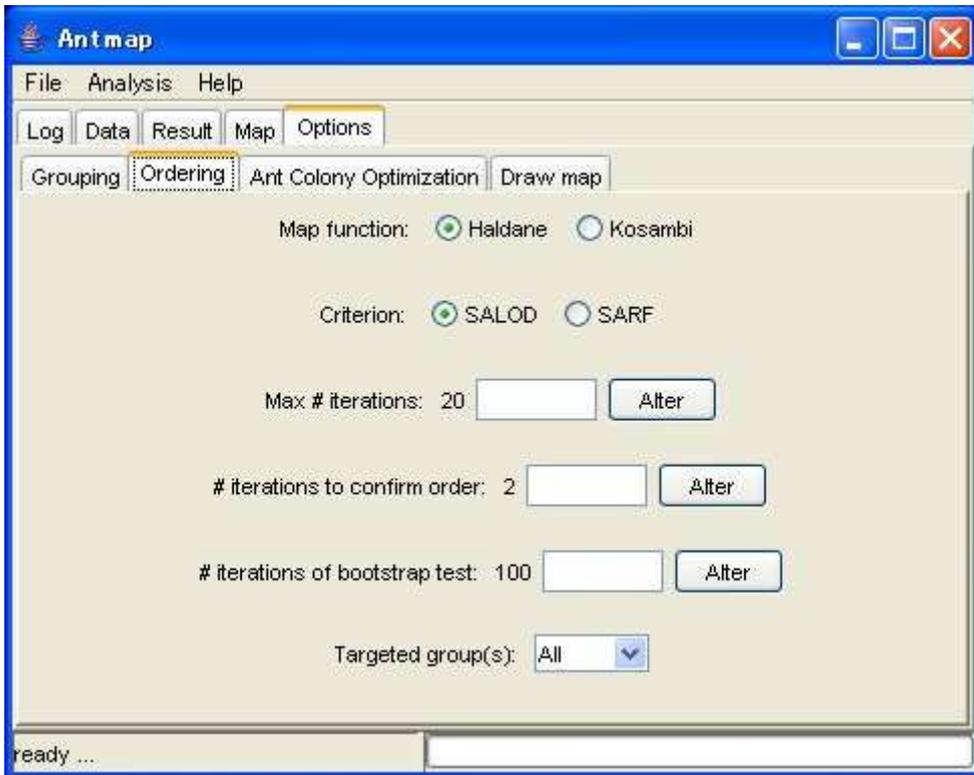
Threshold – Threshold value for grouping

When you choose “LOD” as the criterion, this value is used as the lower threshold for grouping. When you choose the other criteria, it is used as the upper threshold. **Please set this value appropriately for the criterion chosen by you.** For example, the default value “0.3” may be appropriate for the “recombination” criterion, but is too severe for the “distance (cM)” criterion and is too loose for the “p value” and “LOD” criteria.

Min # markers – Minimum number of markers in a linkage group

A linkage group which contain less than this number of markers is not considered as a linkage group. When you set this number to 1, an independent locus (i.e., a locus which is not linked to the other loci) is considered as a single linkage group.

“Ordering” options



Map function – Map function to be used

You can choose one of two mapping functions for calculating map distance between loci. “Haldane” is the function derived by Haldane (1919) and “Kosambi” derived by Kosambi (1944).

Criterion – Criterion for locus ordering

You can choose one of two optimization criteria for locus ordering. “SALOD” is an abbreviation of “Sum of Adjacent LOD score”, and “SARF” is an abbreviation of “Sum of Adjacent Recombination Fractions” (Liu 1998). AntMap will search a locus order which maximizes “SALOD” or minimizes “SARF”.

Max # iterations – Maximum number of iterations

iterations to confirm order – Number of iterations to confirm locus order

A meta-heuristics algorithm such as ACO does not necessarily reach a good solution to a given problem. Thus, AntMap obtains a number of solutions by repeating solution search process, and finally the best solution among them is considered as a solution to the given problem. “Max # iterations” is the

maximum number of iterations of the solution search (i.e., the maximum number of solutions). When the best solution at some point in time has been obtained n times, it is thought to be a good solution for the given problem, and thus the further iterations of the solution search can be omitted. “# iterations to confirm order” is the number of iterations to confirm locus order, i.e., the value of n in the preceding sentence.

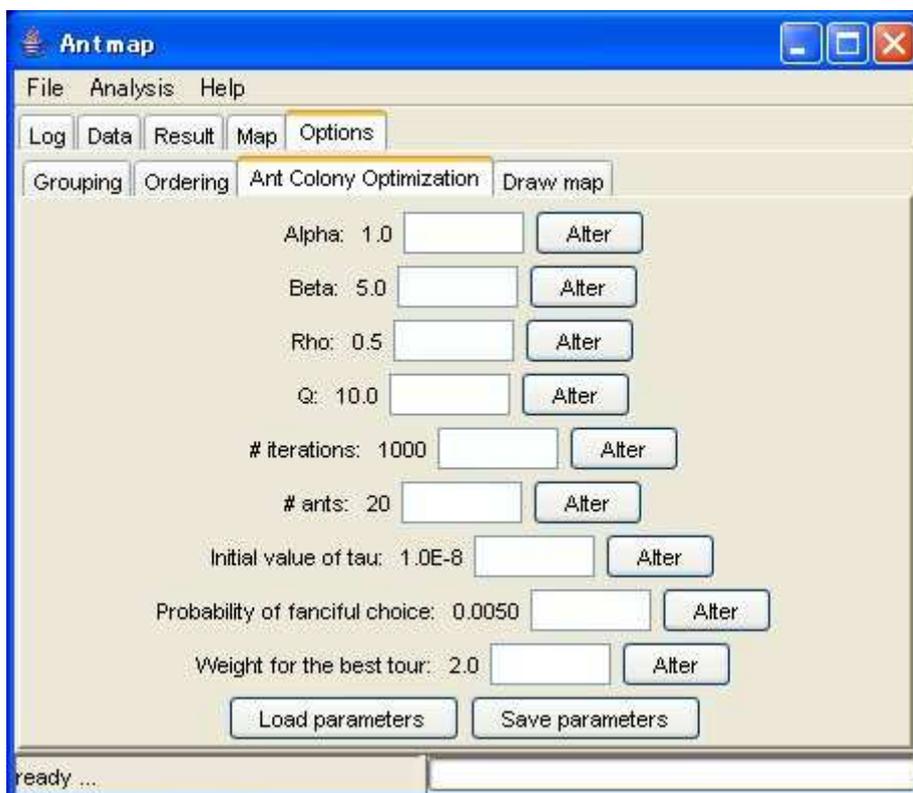
iterations of bootstrap test – Number of iterations in bootstrap test

You can change the number of iterations (repeats) of bootstrapping. To get a good estimate of percentage of correct locus order, 100 may be sufficient.

Targeted group(s) – Targeted group(s) in both locus ordering and bootstrap test

You can choose a linkage group which is targeted in locus ordering and bootstrap test for locus order.

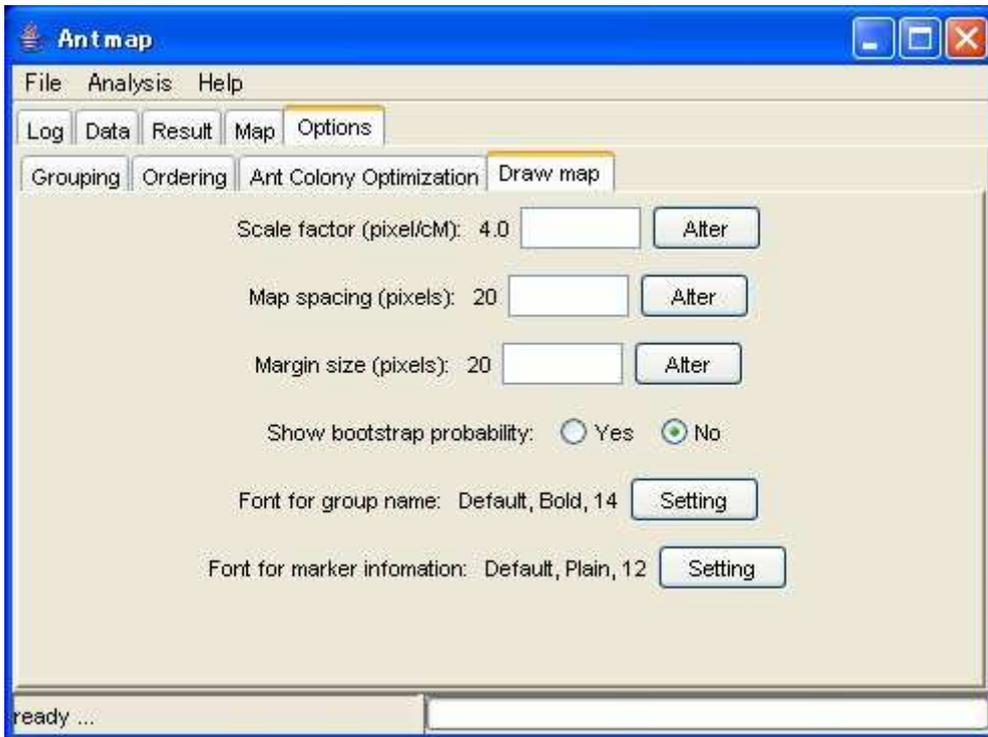
“Ant Colony Optimization” options



These options are for parameters in Ant Colony Optimizations. Most users may not have to change these parameters, because they have been almost optimized for locus ordering problems. If you are interested in the effects of these parameters, please change them and perform locus ordering. You may be able to find out a better set of parameter values than the default setting. The parameter set adjusted by you can be saved to a file (click the “Save parameters” button) and loaded again (click the “Load parameters” button) in your subsequent analyses.

The details of these parameters are given in the “Ant Colony Optimization (ACO)” and “Locus Ordering by ACO algorithm” sections.

“Draw Map” options



Scale factor (pixel/cm) – Scale for drawing a graphic of linkage map

A larger value of this option makes distance between loci larger in the graphic, and vice versa. When you construct a high-density linkage map, it may be better to enlarge this value.

Map spacing (pixels) – Space between linkage groups

Margin size (pixels) – Margin size in a graphic of linkage map

You can adjust the size of margin and space in a graphic of linkage map.

Show bootstrap probability – Show bootstrap probability of locus order

When “Yes” is checked, bootstrap probabilities in bootstrap test of locus order are appeared in a graphic of linkage map.

Font for group name – Font attributes for linkage group names

Font for marker information – Font attribute for marker information

You can choose font attributes for linkage group names and marker information (i.e., marker location, name and bootstrap probability, through this option).

How to cite AntMap

AntMap is open source software. However, please cite AntMap when you intend to publish results from AntMap. Now we are preparing a paper reporting our study related to AntMap. At the current moment, please cite AntMap as follows:

Iwata, H. and S. Ninomiya (2004) Ant colony optimization for linkage grouping and locus ordering in genome mapping. Plant and Animal Genome XII. San Diego. USA.

It is to be noted that the developer is not responsible for any effect you will have by using this software AntMap.

References

- Dorigo, M and T. Stutzle (2004) Ant Colony Optimization. The MIT Press, London.
- Haldane, J.B.S. (1919) The combination of linkage values and the calculation of distances between the loci of linked factors. *Journal of Genetics* 8: 299-309.
- Iwata, H. and S. Ninomiya (2004) Ant colony optimization for linkage grouping and locus ordering in genome mapping. *Plant and Animal Genome XII*. San Diego. USA.
- Iwata, H. and S. Ninomiya (in prep.) AntMap: software for constructing genetic linkage maps using an ant colony optimization algorithm.
- Kosambi, D.D. (1944) The estimation of map distances from recombination values. *Annals of Eugenics* 12: 172-175.
- Lander, E.S., P. Green, J. Abrahamson, A. Barlow, M.J. Daly, S.E. Lincoln and L. Newburg (1987) MAPMAKER: an interactive computer package of constructing primary genetic linkage maps of experimental and natural populations. *Genomics* 1: 174-181.
- Liu, B-H. (1998) *Statistical Genomics: Linkage, Mapping, and QTL Analysis*. CRC Press, New York.
- Manly, B.F.J. (1998) *Randomization, Bootstrap and Monte Carlo Methods in Biology*, Second edition. Chapman & Hall, London.
- Ukai, Y., R. Ohsawa and A. Saito (1991) MAPL: A package of microcomputer programs for RFLP linkage mapping. *Rice Genetics Newsletters* 8: 155-157.

Appendix A: GNU General Public License (GPL)

GNU GENERAL PUBLIC LICENSE
Version 2, June 1991

Copyright (C) 1989, 1991 Free Software Foundation, Inc.
51 Franklin St, Fifth Floor, Boston, MA 02110-1301 USA
Everyone is permitted to copy and distribute verbatim copies
of this license document, but changing it is not allowed.

Preamble

The licenses for most software are designed to take away your freedom to share and change it. By contrast, the GNU General Public License is intended to guarantee your freedom to share and change free software--to make sure the software is free for all its users. This General Public License applies to most of the Free Software Foundation's software and to any other program whose authors commit to using it. (Some other Free Software Foundation software is covered by the GNU Library General Public License instead.) You can apply it to your programs, too.

When we speak of free software, we are referring to freedom, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for this service if you wish), that you receive source code or can get it if you want it, that you can change the software or use pieces of it in new free programs; and that you know you can do these things.

To protect your rights, we need to make restrictions that forbid anyone to deny you these rights or to ask you to surrender the rights. These restrictions translate to certain responsibilities for you if you distribute copies of the software, or if you modify it.

For example, if you distribute copies of such a program, whether gratis or for a fee, you must give the recipients all the rights that you have. You must make sure that they, too, receive or can get the source code. And you must show them these terms so they know their rights.

We protect your rights with two steps: (1) copyright the software, and (2) offer you this license which gives you legal permission to copy, distribute and/or modify the software.

Also, for each author's protection and ours, we want to make certain that everyone understands that there is no warranty for this free software. If the software is modified by someone else and passed on, we want its recipients to know that what they have is not the original, so that any problems introduced by others will not reflect on the original authors' reputations.

Finally, any free program is threatened constantly by software patents. We wish to avoid the danger that redistributors of a free program will individually obtain patent licenses, in effect making the program proprietary. To prevent this, we have made it clear that any patent must be licensed for everyone's free use or not licensed at all.

The precise terms and conditions for copying, distribution and modification follow.

GNU GENERAL PUBLIC LICENSE
TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License applies to any program or other work which contains a notice placed by the copyright holder saying it may be distributed under the terms of this General Public License. The "Program", below, refers to any such program or work, and a "work based on the Program" means either the Program or any derivative work under copyright law: that is to say, a work containing the Program or a portion of it, either verbatim or with modifications and/or translated into another language. (Hereinafter, translation is included without limitation in the term "modification".) Each licensee is addressed as "you".

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running the Program is not restricted, and the output from the Program is covered only if its contents constitute a work based on the Program (independent of having been made by running the Program). Whether that is true depends on what the Program does.

1. You may copy and distribute verbatim copies of the Program's source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and give any other recipients of the Program a copy of this License along with the Program.

You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

2. You may modify your copy or copies of the Program or any portion of it, thus forming a work based on the Program, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

a) You must cause the modified files to carry prominent notices stating that you changed the files and the date of any change.

b) You must cause any work that you distribute or publish, that in whole or in part contains or is derived from the Program or any part thereof, to be licensed as a whole at no charge to all third parties under the terms of this License.

c) If the modified program normally reads commands interactively when run, you must cause it, when started running for such interactive use in the most ordinary way, to print or display an announcement including an appropriate copyright notice and a notice that there is no warranty (or else, saying that you provide a warranty) and that users may redistribute the program under these conditions, and telling the user how to view a copy of this License. (Exception: if the Program itself is interactive but does not normally print such an announcement, your work based on the Program is not required to print an announcement.)

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Program, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based on the Program, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the Program.

In addition, mere aggregation of another work not based on the Program with the Program (or with a work based on the Program) on a volume of a storage or distribution medium does not bring the other work under the scope of this License.

3. You may copy and distribute the Program (or a work based on it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you also do one of the following:

- a) Accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,
- b) Accompany it with a written offer, valid for at least three years, to give any third party, for a charge no more than your cost of physically performing source distribution, a complete machine-readable copy of the corresponding source code, to be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,
- c) Accompany it with the information you received as to the offer to distribute corresponding source code. (This alternative is allowed only for noncommercial distribution and only if you received the program in object code or executable form with such an offer, in accord with Subsection b above.)

The source code for a work means the preferred form of the work for making modifications to it. For an executable work, complete source code means all the source code for all modules it contains, plus any associated interface definition files, plus the scripts used to control compilation and installation of the executable. However, as a special exception, the source code distributed need not include anything that is normally distributed (in either source or binary form) with the major components (compiler, kernel, and so on) of the operating system on which the executable runs, unless that component itself accompanies the executable.

If distribution of executable or object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place counts as distribution of the source code, even though third parties are not compelled to copy the source along with the object code.

4. You may not copy, modify, sublicense, or distribute the Program except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense or distribute the Program is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

5. You are not required to accept this License, since you have not signed it. However, nothing else grants you permission to modify or distribute the Program or its derivative works. These actions are prohibited by law if you do not accept this License. Therefore, by modifying or distributing the Program (or any work based on the Program), you indicate your acceptance of this License to do so, and all its terms and conditions for copying, distributing or modifying the Program or works based on it.

6. Each time you redistribute the Program (or any work based on the Program), the recipient automatically receives a license from the original licensor to copy, distribute or modify the Program subject to these terms and conditions. You may not impose any further restrictions on the recipients' exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties to this License.

7. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Program at all. For example, if a patent license would not permit royalty-free redistribution of the Program by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Program.

If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the section is intended to apply and the section as a whole is intended to apply in other circumstances.

It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the integrity of the free software distribution system, which is implemented by public license practices. Many people have made generous contributions to the wide range of software distributed through that system in reliance on consistent application of that system; it is up to the author/donor to decide if he or she is willing to distribute software through any other system and a licensee cannot impose that choice.

This section is intended to make thoroughly clear what is believed to be a consequence of the rest of this License.

8. If the distribution and/or use of the Program is restricted in certain countries either by patents or by copyrighted interfaces, the original copyright holder who places the Program under this License may add an explicit geographical distribution limitation excluding those countries, so that distribution is permitted only in or among countries not thus excluded. In such case, this License incorporates the limitation as if written in the body of this License.

9. The Free Software Foundation may publish revised and/or new versions of the General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Program specifies a version number of this License which applies to it and "any later version", you have the option of following the terms and conditions either of that version or of any later version published by the Free Software Foundation. If the Program does not specify a version number of this License, you may choose any version ever published by the Free Software Foundation.

10. If you wish to incorporate parts of the Program into other free programs whose distribution conditions are different, write to the author to ask for permission. For software which is copyrighted by the Free Software Foundation, write to the Free Software Foundation; we sometimes make exceptions for this. Our decision will be guided by the two goals of preserving the free status of all derivatives of our free software and of promoting the sharing and reuse of software generally.

NO WARRANTY

11. BECAUSE THE PROGRAM IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE PROGRAM "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH YOU. SHOULD THE PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

12. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR REDISTRIBUTE THE PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE PROGRAM (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE PROGRAM TO OPERATE WITH ANY OTHER PROGRAMS), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Programs

If you develop a new program, and you want it to be of the greatest possible use to the public, the best way to achieve this is to make it free software which everyone can redistribute and change under these terms.

To do so, attach the following notices to the program. It is safest to attach them to the start of each source file to most effectively convey the exclusion of warranty; and each file should have at least the "copyright" line and a pointer to where the full notice is found.

```
<one line to give the program's name and a brief idea of what it does.>  
Copyright (C) <year> <name of author>
```

```
This program is free software; you can redistribute it and/or modify  
it under the terms of the GNU General Public License as published by  
the Free Software Foundation; either version 2 of the License, or  
(at your option) any later version.
```

```
This program is distributed in the hope that it will be useful,  
but WITHOUT ANY WARRANTY; without even the implied warranty of  
MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the  
GNU General Public License for more details.
```

```
You should have received a copy of the GNU General Public License  
along with this program; if not, write to the Free Software  
Foundation, Inc., 51 Franklin St, Fifth Floor, Boston, MA 02110-1301 USA
```

Also add information on how to contact you by electronic and paper mail.

If the program is interactive, make it output a short notice like this when it starts in an interactive mode:

```
Gnomovision version 69, Copyright (C) year name of author  
Gnomovision comes with ABSOLUTELY NO WARRANTY; for details type `show w'.  
This is free software, and you are welcome to redistribute it  
under certain conditions; type `show c' for details.
```

The hypothetical commands `show w' and `show c' should show the appropriate parts of the General Public License. Of course, the commands you use may be called something other than `show w' and `show c'; they could even be mouse-clicks or menu items--whatever suits your program.

You should also get your employer (if you work as a programmer) or your school, if any, to sign a "copyright disclaimer" for the program, if necessary. Here is a sample; alter the names:

```
Yoyodyne, Inc., hereby disclaims all copyright interest in the program  
'Gnomovision' (which makes passes at compilers) written by James Hacker.
```

```
<signature of Ty Coon>, 1 April 1989  
Ty Coon, President of Vice
```

This General Public License does not permit incorporating your program into proprietary programs. If your program is a subroutine library, you may consider it more useful to permit linking proprietary applications with the library. If this is what you want to do, use the GNU Library General Public License instead of this License.

Appendix B: Copyright notice of the C++ class (MTRand) for Mersenne Twister random number generator

AntMap uses the C++ class (MTRand) for Mersenne Twister random number generator, which was written by Richard J. Wagner based on code by Makoto Matsumoto, Takuji Nishimura and Shawn Cokus, in the dynamic library named "org_agmodel_bioinfo_antmap_AntColonyJNI.dll". A copyright notice of this class is as follows:

```
MersenneTwister.h
```

```
Mersenne Twister random number generator -- a C++ class MTRand  
Based on code by Makoto Matsumoto, Takuji Nishimura, and Shawn Cokus  
Richard J. Wagner v1.0 15 May 2003 rjwagner@writeme.com
```

```
The Mersenne Twister is an algorithm for generating random numbers. It  
was designed with consideration of the flaws in various other generators.  
The period,  $2^{19937}-1$ , and the order of equidistribution, 623 dimensions,  
are far greater. The generator is also fast; it avoids multiplication and  
division, and it benefits from caches and pipelines. For more information  
see the inventors' web page at http://www.math.keio.ac.jp/~matumoto/emt.html
```

Reference

```
M. Matsumoto and T. Nishimura, "Mersenne Twister: A 623-Dimensionally  
Equidistributed Uniform Pseudo-Random Number Generator", ACM Transactions on  
Modeling and Computer Simulation, Vol. 8, No. 1, January 1998, pp 3-30.
```

```
Copyright (C) 1997 - 2002, Makoto Matsumoto and Takuji Nishimura,  
Copyright (C) 2000 - 2003, Richard J. Wagner  
All rights reserved.
```

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

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. 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.
3. The names of its contributors may not 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.
```

The original code included the following notice:

When you use this, send an email to: matumoto@math.keio.ac.jp
with an appropriate reference to your work.

It would be nice to CC: rjwagner@writeeme.com and Cokus@math.washington.edu
when you write.