

# MrBayes: A program for the Bayesian inference of phylogeny

Program written by John P. Huelsenbeck<sup>1</sup> and Fredrik Ronquist<sup>2</sup>

Manual written by John P. Huelsenbeck, Fredrik Ronquist, and Barry Hall<sup>1</sup>

<sup>1</sup>*Department of Biology, University of Rochester,  
Rochester, NY 14627, U.S.A.*

<sup>2</sup>*Department of Systematic Zoology, Evolutionary Biology Centre,  
Uppsala University, Norbyv. 18D, SE-752 36 Uppsala, Sweden*

MrBayes is a program for the Bayesian inference of phylogeny. This manual explains Bayesian inference of phylogeny and how to use the program. The program has a command-line interface and should run on a variety of computer platforms. Note that the computer should be reasonably fast and should have a lot of memory (depending on the size of the data matrix, the program may require hundreds of megabytes of memory). The program is optimized for speed and not for minimizing memory requirements. This manual is divided into two parts. The first section gives an overview of the program with a brief tutorial to help you get started and the second section introduces Bayesian inference and presents some of the theory behind the program.

## *Conventions used in this manual*

What you see on the screen and what is in the input file is in a **sans serif font**. What you type is in a **bold font**.

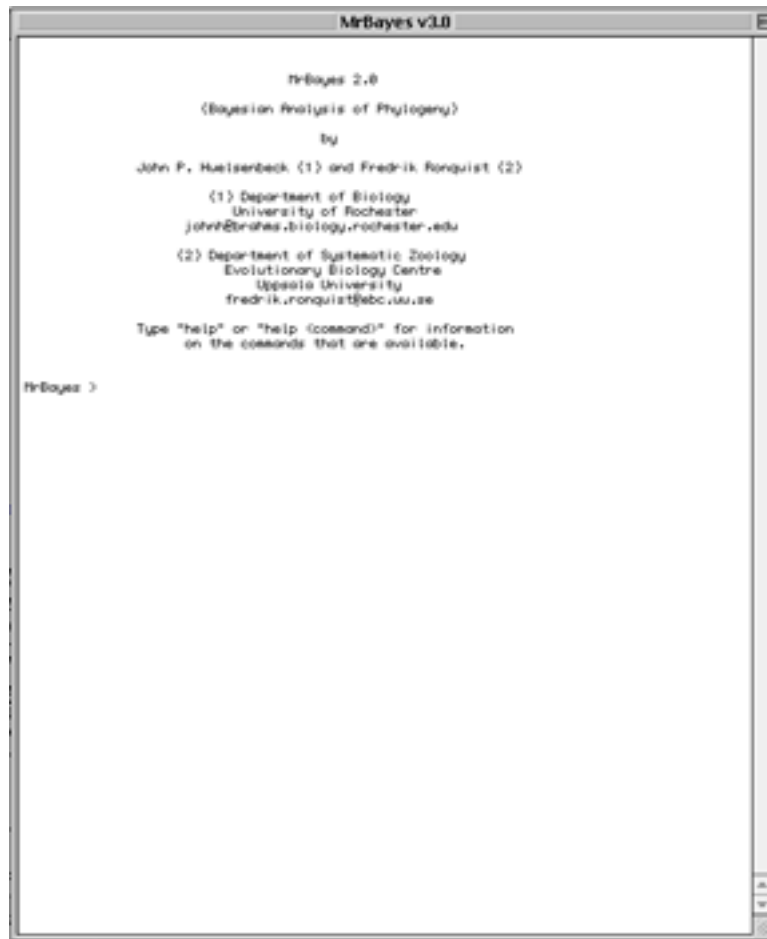
## Section I: Using MrBayes

### *Acquiring and installing MrBayes*

MrBayes is distributed without charge by download from <http://morphbank.ebc.uu.se/mrbayes/>. If someone has given you a copy of MrBayes, we strongly suggest that you download the current version from the above site. It is a plain-vanilla program that uses a command line interface and therefore behaves the same on all platforms—Macintosh, Windows and Unix. The Macintosh and Windows programs are ready to run immediately after being downloaded. If you decide to run the program on a computer running UNIX/LINUX, then you need to compile the program first. Simply untar the file `mrBayesSrc.tar` by typing **tar -xvf BayesSrc.tar**. This command extracts all of the files from the tar archive. You then need to compile the program. We have included a Make file that will automatically compile the program. You simply type **make** to compile the program. We assume as the default C compiler gcc. However, if you do not have gcc installed on your machine, you may need to change the compiler information in the Make file. The executable is called “mb”. To execute the program type **mb** or **./mb**.

### *Getting started using MrBayes*

Start MrBayes by double-clicking the application icon (or typing **mb**) and you will see the screen below:



### *Getting help*

At the **MrBayes>** prompt type **help** to see a list of the commands available in MrBayes. Most commands allow you to set values for a set of options. If you type **help <command>**, where <command> is any of the listed commands, you will see a description of each option and the choices for that option. A complete list of commands and options is given at the end of this manual.

The help facility also provides a way to see the current settings of the parameters. For instance, typing **help lset** not only lists each of the options and possible choices, at the end of the list it tells you the current state for each option.

### *Getting data into MrBayes*

The input file is a standard Nexus file of aligned nucleotide or amino acid sequences. A Nexus file is a simple text (ASCII) file that begins with NEXUS on the first line, followed by the DATA block. The DATA block begins with “**begin data;**” followed by the dimensions statement, the format statement, and matrix. The dimensions statement must contain **ntax**, the number of taxa, and **nchar**, the number of characters in each aligned sequence. The format statement must begin with **datatype=<DNA or RNA or Protein or Standard>**. The format statement may also contain **gap=-** (or whatever symbol is used for a gap in your alignment), **missing=?** (or whatever symbol is used for missing data in your file), **interleave=yes** when the data matrix is interleaved sequences, and **=.** (or whatever symbol is used for matching characters in the alignment). The format statement is followed by the word “**matrix**” on a separate line, followed by the aligned sequences. Each sequence begins with the sequence name separated from the sequence itself by at least one space. The data block is completed by an “**end;**” statement. Note that the begin data, dimensions, format, and end statements all end in a semicolon. That semicolon is essential and must not be left out.

The example below is a Nexus file in block (non-interleaved) format. Non-interleaved is the default, but you can put “**interleave=no**” in the format statement if you want to be sure.

#NEXUS

```
begin data;
  dimensions ntax=9 nchar=720;
  format datatype=rna interleave=no gap=-;
  matrix
FR      GGCAACGGU---GUG ... GCAGACCCACGCCUC
MS2     GGGAACGGA---GUG ... UCAGAUCCACGCCUC
GA      GGCAACGGU---UUG ... UCAGAUCCGCGACUC
SP      UCA---AAUAAAGCA ... UGGGAUCCUAGAGCA
NL95    UCG---AAUAAAGCA ... UGGGAUCCUAGGGUA
M11     CCUUUCAAAUAAAGCA ... UGGGAUCCUAGGGUA
MX1     CCUUUCAAAUAAAGCA ... UGGGAUCCUAGGGUA
QB      CCUUUUAUAAUAAAGCA ... UGGGAUCCUAGGGCC
PP7     GACAGC---CGGUUC ... CGGAUCCCUGACACG
  ;
end;
```

To put the data into MrBayes type **execute** <filename> at the MrBayes> prompt, where “<filename>” is the name of the input file. Note!: The input file must be located in the same folder (directory) as the MrBayes application and the name of the input file should not have blank spaces.

#### *Entering commands into MrBayes*

All of the commands can be entered at the command line at the MrBayes> prompt. At a minimum two commands, “**lset**” and “**mcmc**” are required. “lset” sets the parameters of the likelihood model, and “mcmc” both sets the parameters of the Markov Chain Monte Carlo (MCMC) analysis and initiates the analysis. Entering commands at the command line provides a way for the user to change parameters on the fly and is useful for quickly exploring the effects of such changes, but it is not the most convenient way to use MrBayes. It is much easier to enter commands via a MrBayes block in the input file.

#### *Entering commands via the input file*

Any command that can be entered via the command line can also be entered via the input file as a MrBayes statement. To do so add a MrBayes block after the “end;” statement that terminates the data block. A MrBayes block begins with “**begin mrbayes;**”, and ends with “**end;**”. As is the case for the data block, all statements end with a semicolon. Consider the example block below:

```
begin mrbayes;
  charset 1stpos = 1-720\3;
  charset 2ndpos = 2-720\3;
  charset 3rdpos = 3-720\3;
  partition bycodon = 3:1stpos,2ndpos,3rdpos;
  lset nst=6 rates=sitespec sitepartition=bycodon;
  usertree = (((FR,MS2),GA),((SP,NL95),((M11,MX1),QB)),PP7);
  mcmc ngen=1000 printfreq=100 samplefreq=10 nchains=4 savebrlens=yes;
end;
```

which was taken from the example file replicase.nex. The first four lines define character sets with the command **charset**. The line “**charset 1stpos = 1-720\3;**” defines a character set named “1stpos” as consisting of every third site from 1 through 721. The next two lines similarly define the second and third positions. The format for the charset command is **charset <name> = <character numbers>**. The next line defines a partition named “bycodon” consisting of three parts: 1stpos, 2ndpos, and 3rdpos. Note that the terms 1stpos, 2ndpos, and 3rdpos that were defined in the first three statements are used in the partition statement. A partition statement has the format **partition name = <number of partitions>: <sites in 1st partition>, sites in 2nd partition> etc;**. Together these lines have defined codons and the

sites within codons. The next line sets the parameters of the likelihood model with the command “lset”. “nst=6” sets the general time reversible model, “rates=sitespecific” sets the model for among site variation to being site specific, and “sitepartition=bycodon” sets the name of the site partition to use for site specific rate variation as “bycodon”. Note that a partition had to already be defined for this option to be implemented. The next very long line defines the starting tree using the Newick format. If you define a tree it must be strictly bifurcating or MrBayes will return an error message. You don’t need to define a tree. If no tree is defined MrBayes will use a random tree as its starting point. If you have a problem with a user tree just use the default random tree option. This may actually be preferable in many cases, particularly when monitoring convergence by comparing several independent runs. The final statement mcmc instructs MrBayes to run for 1000 generations (ngen=1000) print its results to the screen once every 100 generations (printfreq=100), to save the current tree to a file once every 10 generations (samplefreq=10), to run four simultaneous MCMC chains, and to save the branch lengths of the trees to the tree file. It also instructs MrBayes to start running the chains. With this block in place typing **execute <filename>**, e.g. **execute replicase.nex**, will produce the following output:

```
Running Markov chain
Starting likelihoods
 1 -- -6429.147 -6429.147
 2 -- -6388.254 -6388.254
 3 -- -6678.446 -6678.446
 4 -- -6597.156 -6597.156
100 -- -19215.03 -- (-6045.46) [-5929.31] (-6088.02) (-6238.81) 2s
200 -- -18522.17 -- (-5878.95) [-5819.39] (-5833.47) (-5834.11) 4s (4~2)
300 -- -18292.23 -- [-5732.10] (-5725.53) (-5821.09) (-5810.95) 6s
400 -- -18139.18 -- (-5712.11) (-5703.27) (-5780.78) [-5693.39] 8s
500 -- -18092.13 -- (-5702.02) (-5684.19) (-5762.62) [-5680.81] 9s
600 -- -18036.97 -- (-5696.36) (-5680.07) (-5701.56) [-5671.27] 11s
700 -- -18018.36 -- (-5682.72) (-5679.98) (-5700.30) [-5662.94] 13s (4~1)
800 -- -17981.18 -- (-5679.95) (-5666.39) (-5684.34) [-5649.37] 15s
900 -- -17970.38 -- (-5684.04) (-5663.26) (-5669.62) [-5647.99] 17s
1000 -- -17959.28 -- (-5684.56) [-5657.12] (-5657.73) (-5648.36) 19s (4~2)

Continue with chain? (yes/no):
```

MrBayes is asking whether you want to continue the run. If you type yes (the full word is required) MrBayes will ask for the number of additional generations you want. If you type anything other than yes the run terminates. Why might you want the run to continue? Primarily to ensure that you consider a sufficient number of trees after the likelihoods of the trees have converged on a stable value. In the output, above, the first column shows the generation number. The next column shows the sum of the natural log of the likelihoods for the trees in each of the four chains weighted according to the temperatures of the chains. The default temperature of the heated chains is determined by the **temp** parameter that is set by the **mcmc** command. The default setting is **temp=0.2**. (For an explanation of heated chains see section 2 of this manual) The next four columns show the likelihoods of the trees in each chain with the cold chain being indicated by square brackets. The final (rightmost) column shows the elapsed time in seconds. In three of the generations that were printed to the screen, there was a successful swap between chains, indicated by “(4~2)”, “(4~1)”, and “(4~2)” in generations 200, 700, and 1000, respectively.

Note that the ln likelihood sum (column 2) starts at -19215.03 and increases with each 100 generations to -17959.28. Eventually that number will converge upon a stable value. Once that stable value is reached the MrBayes is sampling trees according to their posterior probabilities (one hopes). The trees sampled after the log likelihoods reach a plateau can be used to make a consensus tree. Below we will discuss how to decide how many trees to include in that set, but at this point just consider how to answer the question “Continue with chain?”.

Since the log likelihood sum has not reached a stable value type “yes”. How many additional generations do you want to see? This is the \$64,000 question for MCMC analysis. We suggest the following: (1) run multiple chains, starting from random trees if possible and (2) run each chain well past the point where

apparent stationarity was reached. For example, if we run the replicase data set again, this time printing every 500th to the screen, we obtain:

```

500 -- -17993.91 -- [-5651.94] (-5704.28) (-5686.40) (-5658.09) 9s
1000 -- -17938.56 -- (-5653.61) (-5652.86) (-5671.55) [-5644.75] 18s
1500 -- -17911.89 -- (-5642.16) (-5657.28) (-5642.82) [-5643.71] 27s (4~3)
2000 -- -17900.22 -- [-5641.23] (-5642.17) (-5641.69) (-5643.72) 35s (4~2)
2500 -- -17905.10 -- (-5651.80) [-5642.50] (-5642.14) (-5640.15) 44s
3000 -- -17898.56 -- (-5650.13) (-5641.28) [-5635.48] (-5641.77) 53s
3500 -- -17899.50 -- (-5645.09) (-5645.03) [-5637.74] (-5640.57) 62s (1~4)
4000 -- -17888.31 -- [-5635.58] (-5637.56) (-5638.49) (-5643.47) 71s
4500 -- -17901.53 -- [-5638.01] (-5640.68) (-5647.19) (-5646.79) 80s (1~3)
5000 -- -17889.99 -- (-5638.23) (-5640.35) [-5636.40] (-5641.58) 89s (3~2)
5500 -- -17906.17 -- (-5642.21) (-5646.19) [-5648.80] (-5637.32) 98s (1~4)
6000 -- -17910.50 -- (-5642.70) (-5648.94) (-5656.92) [-5637.01] 106s (3~1)
6500 -- -17906.43 -- (-5643.42) [-5638.26] (-5647.00) (-5650.79) 116s
7000 -- -17895.80 -- (-5639.54) [-5634.80] (-5648.19) (-5641.47) 125s
7500 -- -17890.95 -- (-5642.34) (-5639.56) [-5632.56] (-5645.62) 134s
8000 -- -17897.81 -- (-5645.97) (-5641.58) (-5642.21) [-5637.28] 143s (2~3)
8500 -- -17903.34 -- (-5642.63) [-5639.20] (-5644.76) (-5647.57) 152s (1~2)
9000 -- -17892.54 -- (-5644.11) (-5638.13) (-5644.03) [-5634.35] 161s (1~3)
9500 -- -17897.95 -- [-5641.69] (-5637.69) (-5640.14) (-5646.34) 169s (2~1)
10000 -- -17893.30 -- (-5636.77) [-5641.33] (-5647.43) (-5635.27) 178s (4~1)
10500 -- -17891.36 -- (-5642.92) (-5640.35) (-5649.60) [-5629.41] 187s
11000 -- -17896.02 -- (-5648.41) [-5635.74] (-5640.64) (-5641.29) 196s
11500 -- -17907.58 -- (-5642.24) [-5634.30] (-5659.54) (-5646.23) 205s (3~4)
12000 -- -17879.13 -- (-5641.49) (-5635.24) (-5636.55) [-5630.35] 214s (2~3)
12500 -- -17902.92 -- (-5656.38) [-5636.68] (-5648.66) (-5633.30) 223s
13000 -- -17898.09 -- (-5645.50) (-5640.01) (-5645.86) [-5636.92] 232s (2~3)
13500 -- -17889.75 -- (-5647.46) [-5634.11] (-5642.97) (-5634.34) 241s
14000 -- -17889.64 -- (-5641.87) (-5640.95) (-5640.87) [-5633.40] 250s (1~3)
14500 -- -17900.36 -- (-5646.60) (-5643.01) [-5638.79] (-5642.06) 258s (3~4)
15000 -- -17909.86 -- (-5641.49) (-5651.88) (-5654.19) [-5637.49] 267s
15500 -- -17905.15 -- (-5654.59) (-5646.07) (-5643.89) [-5634.62] 276s
16000 -- -17894.56 -- (-5661.10) (-5634.41) (-5644.16) [-5629.49] 285s
16500 -- -17895.79 -- [-5635.66] (-5634.98) (-5647.15) (-5645.50) 294s (3~1)
17000 -- -17887.96 -- (-5635.99) [-5634.48] (-5647.32) (-5636.63) 303s (1~3)
17500 -- -17889.21 -- (-5637.32) (-5636.37) (-5646.93) [-5636.13] 311s
18000 -- -17898.34 -- (-5646.61) (-5639.04) (-5648.28) [-5635.52] 320s (1~3)
18500 -- -17891.88 -- (-5635.24) [-5638.67] (-5648.28) (-5637.44) 329s
19000 -- -17904.45 -- (-5637.69) [-5637.90] (-5647.41) (-5655.37) 338s (2~1)
19500 -- -17902.31 -- (-5646.84) (-5649.32) (-5641.51) [-5636.55] 346s
20000 -- -17891.69 -- [-5635.71] (-5648.45) (-5638.32) (-5637.95) 355s

```

Note that the chain appeared to reach apparent stationarity by about the 3000th generation. Thus, we discard the states of the chain that were sampled before generation 3000 as the “burn in” of the chain. Inferences of phylogeny could be based upon those trees sampled after the burn in.

At the start of the run MrBayes opened three files: mbout.t, mbout.p, and mbout.bp. These are the default names for the output files, but you can set your own name for the output files using the command `filename = <filename>` in the `mcmc` statement. The `mbout.bp` file is used by MrBayes in computing the summary of parameters after the run (see post-run analysis below). `mbout.p` is a file of the values of various parameters in tab delimited text format; this file can be read by a program such as Excel. `mbout.t` is the tree file, the file to which MrBayes writes the current tree at the frequency defined in the `mcmc` statement. That file is the source of the trees used to build a final consensus tree. The trees that are written before the ln likelihoods sum converges on a stable value are less likely, given the data, than are those written after

convergence; so it makes sense to base the consensus tree on those written after convergence. The question is: How many trees are necessary to obtain a good consensus tree? Unfortunately there is no good answer to that question other than “it depends on the data”. In general, more is better, but how many for how much better? Are 10000 trees better than 1000? Yes. How much better? One very pragmatic approach is to set `ngen` at a value that will allow the run to be completed overnight (or in whatever time interval you can live with). For the `adh.nex` example 16 hours would yield about 480,000 generations or 4800 trees, not an unreasonable number. Since 15000 generations were required for likelihood convergence it would make sense to discard the first 150 of the 4800 trees as the “burnin”.

#### *Postrun analysis*

*sumt.*— When the run is complete issuing the `sumt` command will summarize the results concerning the topology and branch lengths. The format is `sumt <filename> burnin = <number of trees to ignore> contype = <allcompat or halfcompat>`. For the `adh.nex` example the statement would be **`sumt filename=replicase.nex.t contype=allcompat burnin=300`**. Remember, burn in is the number of trees to be ignored, not the number of generations. Contype is the consensus type. Halfcompat is the equivalent of 50% majority rule in PAUP\*, while allcompat is the equivalent to the 50% majority rule with “Show frequencies of all observed bipartitions” ticked. If the consensus tree with branch lengths calculated by MrBayes is to be the same as that calculated by PAUP\* contype must correspond to the setting in PAUP\*.

`sumt` writes the bipartitions, the frequencies with which they were found, the probabilities of the bipartitions, and the mean and variance of the branch length (if branch lengths were recorded) to a file called `<filename>.parts`. If branch lengths were recorded it also writes a consensus phylogram based on mean branch lengths. Finally, a consensus tree with clade probabilities is shown on the screen. The consensus tree in `<filename>.con` can be printed with PAUP\*.

You can also construct a consensus tree by importing the tree file into PAUP\* and create a consensus tree in PAUP\*, including all but the trees discarded as the burn-in trees in the consensus. The numbers at the interior branches of a majority-rule consensus tree are the percent of the time that the clade occurs among the sampled trees; i.e. the (posterior) probability of that clade existing.

As an aside, one of the great advantages of Bayesian inference of phylogeny is its speed. After completing the `sumt` command and obtaining a consensus tree, you have gone a long ways towards not only finding a good estimate of phylogeny but also finding those parts of the tree that are well supported. Moreover, you have done this using full models of DNA substitution. This is roughly equivalent to doing a maximum likelihood search with bootstrapping.

*sump.*—The `sump` command summarizes the information in the file named `<filename>.bp` (default is `mbout.bp`). The output is to the screen, and provides the mean, variance, and 95% credible interval for the parameter. The format is `sump filename=<filename>.bp burnin=<number of trees to be ignored>`.

#### *Running MrBayes in the batch mode*

If you already know how many generations are required for convergence you can do the entire run in batch mode by modifying the `mrbytes` block as shown below. Modifications to the previous example are in boldface:

```
begin mrbayes;
  set autoclose=yes;
  charset 1stpos = 1-720\3;
  charset 2ndpos = 2-720\3;
  charset 3rdpos = 3-720\3;
  partition bycodon = 3:1stpos,2ndpos,3rdpos;
  lset nst=6 rates=sitespec sitepartition=bycodon;
  usertree = (((FR,MS2),GA),((SP,NL95),((M11,MX1),QB)),PP7);
  mcmc ngen=20000 printfreq=500 samplefreq=10 nchains=4 savebrlens=yes;
end;
```

The only unfamiliar command is `autoclose=yes`. Autoclose prevents MrBayes from asking if you want to continue the run for more generations after it has completed the assigned 20000 generations. That allows it to get on with `sumt` and `sump`.

#### *The semi-batch mode*

Sometimes it is useful to use a MrBayes block to set most of the parameters, but to vary other parameters from within the program by the command line. You can then use the `help` command to check out the settings and see how varying them affects things. You can use an `mcmc` statement in the `mrBayes` block instead of an `mcmc` statement to set the `mcmc` parameters without starting the run. If you do so, however, all of the post-run statements (`supt` and `sump`) must be deleted from the `mrBayes` block or you will get an error message.

In the semi-batch mode you execute the input file to set the parameters, but issue the `mcmc` command at the command line to start running the chain.

#### *Interrupting a run*

MrBayes can be interrupted at any time by holding down the Command Key while typing a period (Macintosh) or by typing Control-C (Unix).

#### *Some suggestions for lset parameters*

The above MrBayes block is a good choice for dna data that is an alignment of coding regions. For non-coding DNA or for RNA everything before the `lset` statement should be eliminated and `lset nst=6 rates=gamma` is a good starting point. For protein sequences `lset aamodel=jones` is a good starting point.

#### *Memory requirements*

You will quickly discover that MrBayes is a “memory hog”. The reason for this is not simply lazy programming, but the result of decisions made early on to make the program as fast as possible, memory-be-damned. It might be useful to know about how much memory will be required before running the program. The formula is

$$(\# \text{ chains}) \times 2 \times (2 \times \# \text{ taxa}) \times (\# \text{ site patterns}) \times (\# \text{ characters states}) \times (\# \text{ rate categories}) \times 8$$

This will give you the memory requirements in bytes. The number of site patterns is the number of unique patterns in the data. This number is reported before the MCMC analysis. The number of states is either 4 (DNA and RNA), 20 (amino acid), or 61 (codon models). The number of rate categories is the number of rate classes. Usually this is 1 (equal or site specific) and 4 (for the discrete gamma model). The default number of rate categories for gamma-distributed rate variation is `ncat=4`. This option can be changed using `lset ncat=<number>`.

#### *Time requirements*

The time required is a function of the processor speed of the computer, the number of taxa, the length of the sequences, the number of parameters that must be estimated in the model determined by `lset`, and the number of generations for the run. It is therefore almost impossible to predict how long any particular run will take. In general, for any given model, the required time varies linearly with sequence. For the coding sequence data in the `adh.nex` file, using the model as in `lset` above, running on a Macintosh G3 processor at 366 MHz, for 100,000 generations, 0.42 hours were required for 10 taxa, 1.05 hours for 20 taxa, 1.7 hours for 30 taxa, 2.4 hours for 40 taxa, and 3.6 hours were required for all 54 taxa. The time required for protein data or codon data is much greater. If the chain seems to be taking a long time, just remember how long it took you to collect the sequences in the first place!

#### *Ancestral States*

MrBayes can estimate the sequence at one or more internal nodes, i.e. sequences of the taxa that are ancestral to the extant taxa from which the data were obtained. To estimate an ancestral sequence it is necessary to add constraints to the MrBayes block of an existing input file. A constraint is an instruction to always consider a set of taxa as belonging to a clade. The format is “`constraint <name> = taxon1 taxon2`”

taxon3 ... taxoni;". In the phylogeny below we are interested in estimating the ancestral sequences at the nodes labeled con1 con7.

The tree phylogeny was estimated using the MrBayes block below:

```
begin mrbayes;
  charset 1st_pos = 1-867\3;
  charset 2nd_pos = 2-867\3;
  charset 3rd_pos = 3-867\3;
  partition by_codon = 3:1st_pos,2nd_pos,3rd_pos;
  lset nst=6 rates=sitespec sitepartition=by_codon;
  set autoclose=yes;
  mcmc ngen=1000000 printfreq=1000 samplefreq=100 nchains=4 savebrlens=yes
    startingtree=random filename=ALLDNA;
  sumt filename=ALLDNA.t burnin=500 contype=halfcompat;
end;
```

To reconstruct the ancestral sequences the block was modified as shown below in boldface:

```
begin mrbayes;
  charset 1st_pos = 1-867\3;
  charset 2nd_pos = 2-867\3;
  charset 3rd_pos = 3-867\3;
  partition by_codon = 3:1st_pos,2nd_pos,3rd_pos;
  constraint con1=TEM1 TEM6 TEM17 TEM54 TEM10 TEM28 TEM70 TEM76 TEM77 TEM79 TEM29 TEM43
    TEM2 TEM3 TEM8 TEM24 TEM60 TEM21 TEM22 TEM16 TEM59 TEM12 TEM53 TEM78 TEM20 TEM72
    TEM47 TEM68 TEM48;
  constraint con2= TEM1 TEM6 TEM17 TEM54 TEM10 TEM28 TEM70 TEM76 TEM77 TEM79 TEM29 TEM43;
  constraint con3=TEM2 TEM3 TEM8 TEM24 TEM60 TEM21 TEM22 TEM16 TEM59 TEM12 TEM53 TEM78;
  constraint con4=TEM2 TEM3 TEM8 TEM24 TEM60 TEM21 TEM22 TEM16 TEM59;
  constraint con5= shv1 shv2 shv5 shv7 shv18 shv9 shv8 shv24 shv26 shv27 shv28 shv2A
    shv12 shv15 shv13 shv25;
  constraint con6= shv1 shv2 shv5 shv7 shv18 shv9 shv8 shv24 shv26 shv27 shv28;
  constraint con7= shv1 shv2 shv5 shv7 shv18 shv9 shv8 shv24;
  lset nst=6 rates=sitespec sitepartition=by_codon enforcecon=yes inferanc=yes;
  set autoclose=yes;
  mcmc ngen=1000000 printfreq=1000 samplefreq=100 nchains=4 savebrlens=yes
    startingtree=random filename=ALLDNA;
  sumt filename=ALLDNA.t burnin=500 contype=halfcompat;
end;
```

Each constraint has a name (con1, con2, etc.) and each lists the taxa that are constrained. Notice that all of the taxa in con7 are also in con6. Each constraint will result in the estimation of the ancestral sequence for that node. In addition to the constraints themselves two additional parameters must be set with lset: enforcecon=yes and inferanc=yes. inferanc tells Mr. Bayes to infer ancestral states, while enforcecon makes MrBayes keep members of a constraint together as a clade. In practice, a tree must first be constructed without constraints, and probabilities of clades must be estimated at this stage. Once that is done the tree must be reconstructed with the constraints of interest. It is not legitimate to estimate probabilities of clades when constraints have been imposed because the constraint has forced the probability at the constrained node to be 100%. If the filename is alldna, then in addition to the alldna.t, alldna.p, alldna.bp, alldna.con and alldna.parts files MrBayes will write a file for each of the constraints named alldna.sum.1, alldna.sum.11, etc The files list the sites and the probabilities of, respectively, A, C, G or T at that site, and the most probable base at that site. For reasons known only to programmers the sites are numbered starting at zero instead of 1, so you will have to add 1 to each site position to make them correspond to standard numbering of a sequence. Summary of ancestral states at

```
constraint "con1"
```



```

0    0 -- 0.999942 0.000024 0.000017 0.000017 A
1    1 -- 0.000018 0.000017 0.000019 0.999945 T
2    2 -- 0.000017 0.000017 0.999943 0.000023 G
3    0 -- 0.999942 0.000024 0.000017 0.000017 A
4    3 -- 0.000017 0.000017 0.999948 0.000018 G
5    4 -- 0.000024 0.000019 0.000031 0.999927 T
6    5 -- 0.999944 0.000022 0.000017 0.000017 A

```

The most probable sequence and the log of the likelihood of that sequence are given at the bottom of each file.

### Problems and citations

MrBayes is provided free and as is. It is not guaranteed to be free of bugs. Moreover, we suggest that you run multiple chains in all of your analyses to confirm that the program is converging to the posterior probability of interest. You should know that Markov chain Monte Carlo is a fantastic technology that allows you to calculate posterior probabilities for really hard problems. However, the method is not foolproof, and you should take care to confirm your results.

If you experience any problems with the program, please feel free to email JPH at [johnh@brahms.biology.rochester.edu](mailto:johnh@brahms.biology.rochester.edu) or FR at [fredrik.ronquist@ebc.uu.se](mailto:fredrik.ronquist@ebc.uu.se). If you use the results of the program in a paper, please use the following citation:







HUELSENBECK, J. P, AND F. R. RONQUIST. In Press. MRBAYES: Bayesian inference of phylogeny. *Biometrics*.

Please send comments or suggestions about the program to [mrbayes@ebc.uu.se](mailto:mrbayes@ebc.uu.se). Your message will be forwarded to JPH and FR.

## Section II: An Introduction to Bayesian Inference of Phylogeny

### A simple example of Bayesian inference

We will illustrate Bayesian inference using a simple example involving dice. Consider a box with 100 dice, 90 of which are fair and 10 of which are biased. The probability of observing some number of pips after rolling a fair or biased die is given in the following table:

Observation	Fair	Biased
	$\frac{1}{6}$	$\frac{1}{21}$
	$\frac{1}{6}$	$\frac{2}{21}$
	$\frac{1}{6}$	$\frac{3}{21}$
	$\frac{1}{6}$	$\frac{4}{21}$
	$\frac{1}{6}$	$\frac{5}{21}$
	$\frac{1}{6}$	$\frac{6}{21}$

The probability of a high roll is larger for the biased dice than for the fair dice. Suppose that you draw a die at random from the box and roll it twice, observing a four on the first roll and a six on the second roll. What is the probability that the die is biased?

A Bayesian analysis combines one's prior beliefs about the probability of a hypothesis with the likelihood. The likelihood is the vehicle that carries the information about the hypothesis contained in the observations. In this case, the likelihood is simply the probability of observing a four and a six given that the die is biased or fair. Assuming independence of the tosses, the probability of observing a four and a six is

$$\Pr[\begin{smallmatrix} \blacksquare\blacksquare \\ \blacksquare\blacksquare \end{smallmatrix} | \text{Fair}] = \frac{1}{6} \times \frac{1}{6} = \frac{1}{36}$$

for a fair die and

$$\Pr[\begin{smallmatrix} \blacksquare\blacksquare \\ \blacksquare\blacksquare \end{smallmatrix} | \text{Biased}] = \frac{4}{21} \times \frac{6}{21} = \frac{24}{441}$$

for a biased die. The probability of observing the data is 1.96 times greater under the hypothesis that the die is biased. In other words, the ratio of the likelihoods under the two hypotheses suggests that the die is biased.

Bayesian inferences are based upon the posterior probability of a hypothesis. The posterior probability that the die is biased can be obtained using Bayes (1763) formula:

$$\Pr[\text{Biased} \mid \text{4, 6}] = \frac{\Pr[\text{4, 6} \mid \text{Biased}] \times \Pr[\text{Biased}]}{\Pr[\text{4, 6} \mid \text{Biased}] \times \Pr[\text{Biased}] + \Pr[\text{4, 6} \mid \text{Fair}] \times \Pr[\text{Fair}]}$$

where  $\Pr[\text{Biased}]$  and  $\Pr[\text{Fair}]$  are the prior probabilities that the die is biased or fair, respectively. As we set up the problem, a reasonable prior probability that the die is biased would be the proportion of the dice in the box that were biased. The posterior probability is then

$$\Pr[\text{Biased} \mid \text{4, 6}] = \frac{\frac{24}{441} \times \frac{1}{10}}{\frac{24}{441} \times \frac{1}{10} + \frac{1}{36} \times \frac{9}{10}} = 0.179$$

This means that our opinion that the die is biased changed from 0.1 to 0.179 after observing the four and six.

Depending upon one's viewpoint, the incorporation of prior beliefs about a parameter is either a strength or a weakness of Bayesian inference. It is a strength in as much as the method explicitly incorporates prior information in inferences about a hypothesis. However, it can often be difficult to specify a prior. For the dice example, it is easy to specify the prior as we provided information on the number of fair and biased dice in the box and also specify that a die was randomly selected. However, if we were to simply state that the die is either fair or biased, but did not specify a physical description of how the die was chosen, it would have been much more difficult to specify a prior specifying the probability that the die is biased. For example, one could have taken the two hypotheses to have been *a priori* equally probable or given much more weight to the hypothesis that had the die is fair as severely biased dice are rarely encountered (or manufactured) in the real world.

### Bayesian inference of phylogeny

Bayesian inference of phylogeny is based upon the posterior probability of a phylogenetic tree,  $\tau$ . The posterior probability of the  $i$ th phylogenetic tree,  $\tau_i$ , conditioned on the observed matrix of aligned DNA sequences ( $\mathbf{X}$ ) is obtained using Bayes formula:

$$f(\tau_i \mid \mathbf{X}) = \frac{f(\mathbf{X} \mid \tau_i) f(\tau_i)}{\sum_{j=1}^{B(s)} f(\mathbf{X} \mid \tau_j) f(\tau_j)}$$

[throughout, we denote conditional probabilities as  $f(\cdot \mid \cdot)$ ]. Here,  $f(\tau_i \mid \mathbf{X})$  is the posterior probability of the  $i$ th phylogeny and can be interpreted as the probability that  $\tau_i$  is the correct tree given the DNA sequence data. The likelihood of the  $i$ th tree is  $f(\mathbf{X} \mid \tau_i)$  and the prior probability of the  $i$ th tree is  $f(\tau_i)$ . The summation in the denominator is over all  $B(s)$  trees that are possible for  $s$  species. This number is  $B(s) = \frac{(2s-3)!}{2^{s-2}(s-2)!}$  for rooted trees,  $B(s) = \frac{(2s-5)!}{2^{s-3}(s-3)!}$  for unrooted trees, and  $B(s) = \frac{s!(s-1)!}{2^{s-1}}$  for labelled histories. Typically, an uninformative prior is used for trees, such that  $f(\tau_i) = \frac{1}{B(s)}$

*DNA sequence data.*—The program assumes as input an aligned matrix of  $s$  DNA sequences:

$$\mathbf{X} = \{x_{ij}\} = \begin{matrix} \text{Species 1} \\ \text{Species 2} \\ \text{Species 3} \\ \vdots \\ \text{Species } s \end{matrix} \begin{pmatrix} A & A & C & C & T \\ A & A & C & G & G \\ A & C & C & C & T \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ A & C & C & C & T \end{pmatrix}$$

The data matrix consists of the sequences for  $s$  species for  $c = 5$  sites from a gene ( $c$  is the length of the aligned DNA sequences). The observations at the first site are  $\mathbf{x}_1 = \{A, A, A, \dots, A\}'$ . In general, the information at the  $i$ th site in the matrix is denoted  $\mathbf{x}_i$ .

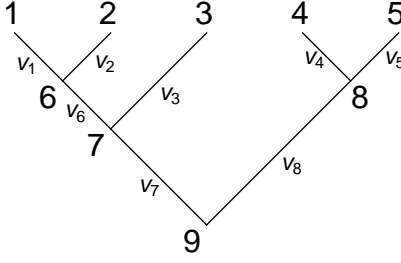


Figure 1.—An example of a phylogenetic tree for  $s = 5$  species. The branch lengths are denoted  $v_i$ .

*Phylogenetic models.*—What is the probability of observing the data at the  $i$ th site? To calculate this probability, we assume a phylogenetic model. A phylogenetic model consists of a tree ( $\tau_i$ ) with branch lengths specified on the tree ( $v_i$ ) and a stochastic model of DNA substitution. Figure 1 shows an example of a phylogenetic tree of  $s = 5$  species. The tips of the tree are labeled  $1, 2, \dots, s$  and the internal nodes of the tree are labeled  $s + 1, s + 2, \dots, 2s - 1$ ; the root of the tree is always labeled  $2s - 1$ . The lengths of the branches are denoted  $v_i$  and are in terms of the number of substitutions expected to occur along the  $i$ th branch. In general, the ancestor of node  $k$  will be denoted  $\sigma(k)$ ; the ancestor of node 4 is  $\sigma(4) = 8$ . The ancestor of the root is  $\sigma(2s - 1) = \emptyset$ .

The second part of the phylogenetic model consists of a stochastic model of DNA substitution. Here, the typical assumption is that DNA substitution follows a time-homogeneous Poisson process. The heart of the model is a matrix specifying the instantaneous rate of substitution from one nucleotide state to another:

$$\mathbf{Q} = \{q_{ij}\} = \begin{pmatrix} \cdot & r_{AC} & r_{AG} & r_{AT} \\ r_{CA} & \cdot & r_{CG} & r_{CT} \\ r_{GA} & r_{GC} & \cdot & r_{GT} \\ r_{TA} & r_{TC} & r_{TG} & \cdot \end{pmatrix}$$

where the matrix specifies the rate of change from nucleotide  $i$  (row) to nucleotide  $j$  (column). The nucleotides are in the order A, C, G, T. The diagonals of the matrix are specified such that the rows each sum to 0. This matrix of instantaneous rates specifies the most general of nucleotide substitution, and is referred to as the general non-reversible model of DNA substitution (Yang, 1994a). Because the rate of substitution and time are confounded, the  $\mathbf{Q}$  matrix is rescaled such that  $-\sum \pi_i q_{ii} = 1$  for all  $i$  (making the average rate of substitution 1). Over a branch of length  $v$  the transition probabilities are calculated as  $\mathbf{P}(v, \boldsymbol{\theta}) = \{p_{ij}(v, \boldsymbol{\theta})\} = e^{\mathbf{Q}v}$ . The parameters of the substitution model are contained in a vector  $\boldsymbol{\theta}$ . The stationary frequencies of the nucleotides (denoted  $\boldsymbol{\pi} = \{\pi_A, \pi_C, \pi_G, \pi_T\}$ ) are obtained by letting the substitution process run for a very long time ( $v \rightarrow \infty$ ). The model is non-reversible because  $\pi_i r_{ij} \neq \pi_j r_{ji}$ . Importantly, for non-reversible models, the probability of observing the data changes depending upon where the root of the phylogenetic tree is placed.

The most general time-reversible model of DNA substitution has instantaneous rates

$$\mathbf{Q} = \{q_{ij}\} = \begin{pmatrix} \cdot & \pi_C r_{AC} & \pi_G r_{AG} & \pi_T r_{AT} \\ \pi_A r_{AC} & \cdot & \pi_G r_{CG} & \pi_T r_{CT} \\ \pi_A r_{AG} & \pi_C r_{CG} & \cdot & \pi_T r_{GT} \\ \pi_A r_{AT} & \pi_C r_{CT} & \pi_G r_{GT} & \cdot \end{pmatrix}$$

and is referred to as the GTR model (Tavaré, 1986). Other commonly used models of DNA substitution are simply restrictions of the GTR model. For example, the model proposed by Kimura (1980) has instantaneous rates

$$\mathbf{Q} = \{q_{ij}\} = \begin{pmatrix} \cdot & 1 & \kappa & 1 \\ 1 & \cdot & 1 & \kappa \\ \kappa & 1 & \cdot & 1 \\ 1 & \kappa & 1 & \cdot \end{pmatrix}$$

where  $\kappa$  is the transition/transversion rate ratio.

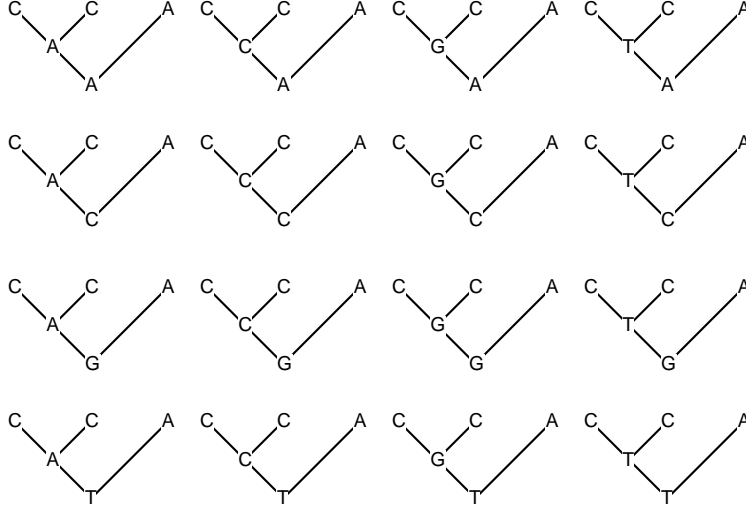


Figure 2.—The 16 possible assignments of nucleotides to the internal nodes of a tree of  $s = 3$  species. The observations at the site are  $\mathbf{x}_i = \{C, C, A\}$  and the unobserved nucleotides at the internal nodes of the tree are denoted  $\mathbf{y}$ .

*The likelihood of a phylogeny.*—The phylogenetic model consists of a tree  $(\tau_i)$  with branch lengths  $(\mathbf{v}_i)$  and a stochastic model of DNA substitution that is specified by a matrix of instantaneous rates. The probability of observing the data at the  $i$ th site in the aligned matrix is a sum over all possible assignments of nucleotides to the internal nodes of the tree:

$$f(\mathbf{x}_i | \tau_j, \mathbf{v}_j, \boldsymbol{\theta}) = \sum_{\mathbf{y}} \left[ \pi_{y_{2s-1}} \left( \prod_{k=1}^s p_{y_{i\sigma(k)}, x_{ik}}(v_k, \boldsymbol{\theta}) \right) \left( \prod_{k=s+1}^{2s-2} p_{y_{i\sigma(k)}, y_{ik}}(v_k, \boldsymbol{\theta}) \right) \right]$$

Here,  $y_{ij}$  is the (unobserved) nucleotide at the  $j$ th node for the  $i$ th site. The summation is over all  $4^{s-1}$  ways that nucleotides can be assigned to the internal nodes of the tree. Figure 2 illustrates the possible nucleotide assignments for a simple tree of  $s = 3$  species. Felsenstein (1981) introduced a pruning algorithm that efficiently calculates the summation. Often, the rate at the site is assumed to be drawn from a gamma distribution (Yang, 1993). This allows one to relax the assumption that the rate of substitution is equal across all sites. If gamma-distributed rate variation is assumed, then the probability of observing the data at the  $i$ th site becomes:

$$f(\mathbf{x}_i | \tau_j, \mathbf{v}_j, \boldsymbol{\theta}, \alpha) = \int_0^\infty \left\{ \sum_{\mathbf{y}} \left[ \pi_{y_{2s-1}} \left( \prod_{k=1}^s p_{y_{i\sigma(k)}, x_{ik}}(v_k r, \boldsymbol{\theta}) \right) \left( \prod_{k=s+1}^{2s-2} p_{y_{i\sigma(k)}, y_{ik}}(v_k r, \boldsymbol{\theta}) \right) \right] \right\} f(r | \alpha) dr$$

where  $f(r | \alpha)$  is the density of the rate  $r$  under the gamma model (Yang, 1993). The parameter  $\alpha$  is the shape parameter of the gamma distribution (here, the shape and the scale parameters of the gamma distribution are both set to  $\alpha$ ). Typically, this integral is impossible to evaluate. Hence, an approximation first suggested by Yang (1994b) is used in which the continuous gamma distribution is broken into  $K$  categories, each with equal weight. The mean rate from each category represents the rate for the entire category. The probability of observing the data at the  $i$ th site then becomes:

$$f(\mathbf{x}_i | \tau_j, \mathbf{v}_j, \boldsymbol{\theta}, \alpha) = \sum_{n=1}^K \left\{ \sum_{\mathbf{y}} \left[ \pi_{y_{2s-1}} \left( \prod_{k=1}^s p_{y_{i\sigma(k)}, x_{ik}}(v_k r_n, \boldsymbol{\theta}) \right) \left( \prod_{k=s+1}^{2s-2} p_{y_{i\sigma(k)}, y_{ik}}(v_k r_n, \boldsymbol{\theta}) \right) \right] \right\} \frac{1}{K}$$

Another method for relaxing the assumption of equal rates across sites is to assign different sites to categories, and then estimate the rate for each category separately. For example, sites can be assigned to categories according to codon position, and the rate for each codon separately estimated. Although this

approach assumes that all of the sites within a category have the same rate, it often does a better job of explaining DNA sequences than the gamma model. The probability of observing the data at the  $i$ th site is

$$f(\mathbf{x}_i|\tau_j, \mathbf{v}_j, \boldsymbol{\theta}, r_i) = \sum_{\mathbf{y}} \left[ \pi_{y_{2s-1}} \left( \prod_{k=1}^s p_{y_{i\sigma(k)}, x_{ik}}(v_k r_i, \boldsymbol{\theta}) \right) \left( \prod_{k=s+1}^{2s-2} p_{y_{i\sigma(k)}, y_{ik}}(v_k r_i, \boldsymbol{\theta}) \right) \right]$$

under the site-specific model or rate variation across sites. The rate for the  $i$ th site is  $r_i$ .

Assuming independence of the substitutions across sites, the probability of observing the aligned matrix of DNA sequences is

$$f(\mathbf{X}|\tau_j, \mathbf{v}_j, \boldsymbol{\theta}, \alpha) = \prod_{i=1}^c f(\mathbf{x}_i|\tau_j, \mathbf{v}_j, \boldsymbol{\theta}, \alpha)$$

Importantly, the likelihood can be calculated under a number of different models of character change. For example, the codon model describes the substitution process over triplets of sites (a codon) and allows the estimation of the nonsynonymous/synonymous rate ratio (Goldman and Yang, 1994; Muse and Gaut, 1994). Similarly, models of DNA substitution have been described that allow nonindependent substitutions to occur in stem regions of rRNA genes (Schöniger and von Haeseler, 1994). Finally, one can calculate likelihoods for amino acid (Adachi and Hasegawa, 1992, 1996a, 1996b), restriction site (Smouse and Li, 1987), and, more recently, morphological data (Lewis, 2001).

*Bayesian inference of phylogeny.*—As described so far, the likelihood depends upon several unknown parameters; generally, the phylogeny, branch lengths, and substitution parameters are unknown. The method of maximum likelihood estimates these parameters by finding the values of the parameters which maximize the likelihood function. Currently, programs such as PAUP\* (Swofford, 1998), PAML (Yang, 1997), and PHYLIP (Felsenstein, 1993) estimate phylogeny using the method of maximum likelihood.

Bayesian inference is based instead upon the posterior probability of the parameter. As described above, the posterior probability of the  $i$ th tree is

$$f(\tau_i|\mathbf{X}) = \frac{f(\mathbf{X}|\tau_i)f(\tau_i)}{\sum_{j=1}^{B(s)} f(\mathbf{X}|\tau_j)f(\tau_j)}$$

where the likelihood function is integrated over all possible values for the branch lengths and substitution parameters:

$$f(\mathbf{X}|\tau_i) = \int_{v_i} \int_{\theta} \int_{\alpha} f(\mathbf{X}|\tau_i, \mathbf{v}_i, \boldsymbol{\theta}, \alpha) f(\mathbf{v}_i) f(\boldsymbol{\theta}) f(\alpha) d\mathbf{v}_i d\boldsymbol{\theta} d\alpha$$

Bayesian inference of phylogeny has been described by Rannala and Yang (1996), Mao and colleagues (Mao, 1996; Mao and Newton, 1997; Mao et al., 1999), and Li (1996).

*Markov chain Monte Carlo.*—Typically, the posterior probability cannot be calculated analytically. However, the posterior probability of phylogenies can be approximated by sampling trees from the posterior probability distribution. Markov chain Monte Carlo (MCMC) can be used to sample phylogenies according to their posterior probabilities. The Metropolis-Hastings-Green (MHG) algorithm (Green, 1995; Hastings, 1970; Metropolis et al., 1953) is an MCMC algorithm that has been used successfully to approximate the posterior probabilities of trees (Larget and Simon, 1999; Yang and Rannala, 1997).

The MHG algorithm works as follows. Let  $\Psi = \{\tau, \mathbf{v}, \boldsymbol{\theta}, \alpha\}$  be a specific tree, combination of branch lengths, substitution parameters, and gamma shape parameter. The MHG algorithm constructs a Markov chain that has as its stationary frequency the posterior probability of interest (in this case, the joint posterior probability of  $\tau$ ,  $\mathbf{v}$ ,  $\boldsymbol{\theta}$ , and  $\alpha$ ). The current state of the chain is denoted  $\Psi$ . If this is the first generation of the chain, then the chain is initialized (perhaps by randomly picking a state from the prior). A new state is then proposed,  $\Psi'$ . The probability of proposing the new state given the old state is  $f(\Psi'|\Psi)$  and the probability of making the reverse move (which is never actually made) is  $f(\Psi|\Psi')$ . The new state is accepted with probability

$$\begin{aligned} R &= \min \left( 1, \frac{f(\Psi'|\mathbf{X})}{f(\Psi|\mathbf{X})} \times \frac{f(\Psi|\Psi')}{f(\Psi'|\Psi)} \right) \\ &= \min \left( 1, \frac{f(\mathbf{X}|\Psi')f(\Psi')/f(\mathbf{X})}{f(\mathbf{X}|\Psi)f(\Psi)/f(\mathbf{X})} \times \frac{f(\Psi|\Psi')}{f(\Psi'|\Psi)} \right) \end{aligned}$$

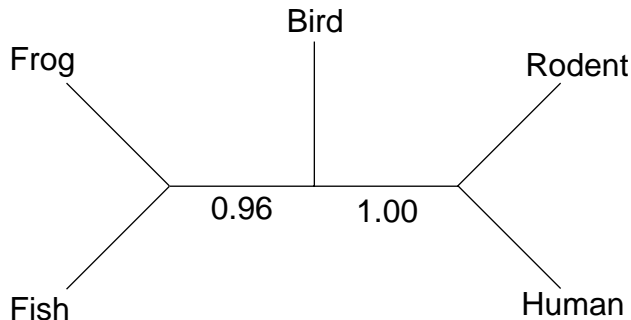


Figure 3.—The tree with the maximum posterior probability for the analysis of the *c-myc* sequences. The numbers at the internal branches represent the posterior probability that the clade is correct.

$$= \min \left( 1, \underbrace{\frac{f(\mathbf{X}|\Psi')}{f(\mathbf{X}|\Psi)}}_{\text{Likelihood Ratio}} \times \underbrace{\frac{f(\Psi')}{f(\Psi)}}_{\text{Prior Ratio}} \times \underbrace{\frac{f(\Psi|\Psi')}{f(\Psi|\Psi)}}_{\text{Proposal Ratio}} \right)$$

A uniform random variable between 0 and 1 is drawn. If this number is less than  $R$ , then the proposed state is accepted and  $\Psi = \Psi'$ . Otherwise, the chain remains in the original state. This process of proposing a new state, calculating the acceptance probability, and either accepting or rejecting the proposed move is repeated many thousands of times. The sequence of states visited forms a Markov chain. This chain is sampled (either every step, or the chain is “thinned” and samples are taken every so often). The samples from the Markov chain form a valid, albeit dependent, sample from the posterior probability distribution (Tierney, 1994). As described here, the Markov chain samples from the joint probability density of trees, branch lengths, and substitution parameters. The marginal probability of trees can be calculated by simply printing to a file the trees that are visited during the course of the MCMC analysis. The proportion of the time any single tree is found in this sample is an approximation of the posterior probability of the tree.

A variant of MCMC called “Metropolis-coupled Markov chain Monte Carlo” (or MCMCMC; Geyer, 1991) runs several chains, some number of which are heated. A heated Markov chain has the posterior probability of a tree raised to some power  $\beta$ . For example, if  $n$  chains are run, the heat for the  $i$ th chain might be  $\beta = \frac{1}{1+T(i-1)}$ ; when  $i = 1$ ,  $\beta = 1$  corresponding to the unheated or cold chain. This type of heating scheme is called incremental heating because  $\beta$  becomes smaller as  $i$  becomes larger. Heated Markov chains can more easily cross deep valleys; the effect of heating is to fill in valleys and lower peaks. Hence, a heated Markov chain better explores the parameter space. The MCMCMC algorithm lets every chain go one step. Then, a swap of the states between two chains is attempted. If the swap is accepted, then the states for the two chains are exchanged. If a swap occurs between a heated chain and the cold chain, the cold chain might cross a large valley that it would normally cross with only a very small probability.

*An example of Bayesian inference of phylogeny.*—Here we will demonstrate Bayesian inference of phylogeny for a simple example of five species. The DNA sequences are *albumin* and *c-myc* sequences sampled from a fish, frog, bird, rodent, and human (*albumin*: Actinopterygii, *Salmo salar*, X52397; Amphibia, *Xenopus laevis*, M18350; Aves, *Gallus gallus*, X60688; Rodentia, *Rattus norvegicus*, J00698; Primates, *Homo sapiens*, L00132; *c-myc*: Actinopterygii, *Salmo gairdneri*, M13048; Amphibia, *Xenopus laevis*, M14455; Aves, *Gallus gallus*, M20006; Rodentia, *Rattus norvegicus*, Y00396; Primates, *Homo sapiens*, V00568). There are a total of  $B(5) = 15$  unrooted trees possible for the five sequences. The prior probability of any single tree, then, is  $\frac{1}{15} = 0.067$ .

We first analyzed the *c-myc* DNA sequences using MrBayes. The HKY85+ $\Gamma$  model of DNA substitution was assumed (Hasegawa et al., 1985; Yang, 1994b). This model allows there to be a different rate of transitions and transversions, different stationary nucleotide frequencies, and among-site rate variation (as described by a discrete gamma distribution). The Markov chain was run for 100,000 generations and sampled every 100 generations. The first 10,000 generations of the chain were discarded; the chain was started from

a random tree and branch lengths and it took some time for the chain to reach apparent stationarity. Hence, inferences were based upon a sample of 900 trees. Figure 3 summarizes the results of the analysis. The tree with the largest posterior probability was (Fish,Frog,(Bird,(Rodent,Human))) and the posterior probability of this tree was 0.964. Figure 3 shows the posterior probability of the clades on the tree with the maximum posterior probability.

One of the advantages of Bayesian inference of phylogeny is that the results are easy to interpret. For example, the sum of the posterior probabilities of all trees will sum to 1. Moreover, the posterior probability of any single clade is simply the sum of the posterior probabilities of all trees that contain that clade. Finally, a credible set of trees can be formed by ordering all of the trees from largest to smallest posterior probability and then adding those trees with the highest posterior probability to a set until the cumulative posterior probability is 0.95. A 95% credible set of trees for the *c-myc* gene would contain only one tree.

The posterior probability of trees can form the prior for any subsequent analysis of the species. For example, let us imagine that the *albumin* sequences were analyzed after the *c-myc* sequences. The posterior probabilities of phylogenies from the analysis of the *c-myc* sequences is the prior for the analysis of the *albumin* sequences. The posterior probability of the trees after analysis of the *albumin* sequences is shown in the table. The posterior probability of  $\tau_1$  is now 0.996. Our beliefs about the phylogeny of the five species have changed throughout the analysis. For example, our initial belief about the the phylogeny  $\tau_1$  was 0.067. After observing the *c-myc* sequences, our belief that this is the true phylogeny increased from 0.067 to 0.964. The *albumin* sequences strengthened our beliefs about this phylogeny. The final posterior probability of this phylogeny was 0.996. This probability could form the prior probability for tree 1 for any subsequent analysis.

$i$	$\tau_i$	$f(\tau_i)$	$f(\tau_i c-myc)$	$f(\tau_i albumin)$
1	(Fish,Frog,(Bird,(Rodent,Human)))	0.067	0.964	0.996
2	(Fish,Frog,(Rodent,(Bird,Human)))	0.067	0.000	0.000
3	(Fish,Frog,(Human,(Rodent,Bird)))	0.067	0.000	0.000
4	(Fish,Bird,(Frog,(Rodent,Human)))	0.067	0.012	0.003
5	(Fish,Bird,(Rodent,(Frog,Human)))	0.067	0.000	0.000
6	(Fish,Bird,(Human,(Rodent,Frog)))	0.067	0.000	0.000
7	(Fish,Rodent,(Bird,(Frog,Human)))	0.067	0.000	0.000
8	(Fish,Rodent,(Frog,(Bird,Human)))	0.067	0.000	0.000
9	(Fish,Rodent,(Human,(Bird,Frog)))	0.067	0.000	0.000
10	(Fish,Human,(Bird,(Rodent,Frog)))	0.067	0.000	0.000
11	(Fish,Human,(Frog,(Rodent,Bird)))	0.067	0.000	0.000
12	(Fish,Human,(Rodent,(Frog,Bird)))	0.067	0.000	0.000
13	(Frog,Bird,(Fish,(Rodent,Human)))	0.067	0.023	0.001
14	(Frog,Human,(Fish,(Rodent,Bird)))	0.067	0.000	0.000
15	(Frog,Rodent,(Fish,(Bird,Human)))	0.067	0.000	0.000

One modification of the analysis of the vertebrate sequences would be to modify the prior probabilities of trees. There is overwhelming morphological and paleontological evidence that the correct phylogeny for fish, frogs, birds, rodents, and humans is tree  $\tau_1$ . Hence, a systematist might reflect this prior information as a different prior probability on the trees. For example, he or she may decide to put almost all of the prior probability on  $\tau_1$  and very little prior probability on the other trees.

### Acknowledgments

Peter Foster gave us the routines for exponentiating the  $Q$  matrix and Ziheng Yang provided the routines for manipulating gamma distributions. Barry Hall, Johan Nylander, and Jon Bollback provided feedback on the program and did some initial testing. NSF (DEB-0075406) and the Swedish Natural Science Research Council provided much needed financial support.

### References

- ADACHI, J., AND M. HASEGAWA. 1992. Amino acid substitution of proteins coded for in mitochondrial DNA during mammalian evolution. *Jpn. J. Genet.* 67:187–197.

- ADACHI, J., AND M. HASEGAWA. 1996a. MOLPHY: Programs for molecular phylogenetics, I. PROTML. Maximum likelihood inference of protein phylogeny. *Comput. Sci. Monogr.* 27:1–77.
- ADACHI, J., AND M. HASEGAWA. 1996b. MOLPHY, version 2.3: Programs for molecular phylogenetics based on maximum likelihood. *Comput. Sci. Monogr.* 28:1–150.
- BAYES, T. 1763. An essay towards solving a problem in the doctrine of chances. *Philosophical Transactions of the Royal Society of London* 53:370–418. Reprinted, E. S. Pearson and M. G. Kendall (eds.). 1970. Pages 131–153 in *Studies in the History of Statistics and Probability*. Charles Griffin, London.
- FELSENSTEIN, J. 1981. Evolutionary trees from DNA sequences: A maximum likelihood approach. *Journal of Molecular Evolution* 17:368–376.
- FELSENSTEIN, J. 1993. *PHYLIP (Phylogeny Inference Package) version 3.5c*. Distributed by the author. Department of Genetics, University of Washington, Seattle.
- GEYER, C. J. 1991. Markov chain Monte Carlo maximum likelihood. Pages 156–163 in *Computing Science and Statistics: Proceedings of the 23rd Symposium on the Interface*. (E. M. Keramidas, ed.). Fairfax Station: Interface Foundation.
- GOLDMAN, N., AND Z. YANG. 1994. A codon-based model of nucleotide substitution for protein-coding DNA sequences. *Molecular Biology and Evolution* 11:725–736.
- GREEN, P. J. 1995. Reversible jump Markov chain Monte Carlo computation and Bayesian model determination. *Biometrika* 82:711–732.
- HASEGAWA, M., H. KISHINO, AND T. YANO. 1985. Dating the human-ape splitting by a molecular clock of mitochondrial DNA. *Journal of Molecular Evolution* 22:160–174.
- HASTINGS, W. K. 1970. Monte Carlo sampling methods using Markov chains and their applications. *Biometrika* 57:97–109.
- KIMURA, M. 1980. A simple method for estimating evolutionary rates of base substitutions through comparative studies of nucleotide sequences. *Journal of Molecular Evolution* 16:111–120.
- LARGET, B., AND D. SIMON. 1999. Markov chain Monte Carlo algorithms for the Bayesian analysis of phylogenetic trees. *Molecular Biology and Evolution* 16:750–759.
- LEWIS, P. O. 2001. Maximum likelihood phylogenetic inference: Modeling discrete morphological characters. *Systematic Biology*.
- LI, S. 1996. *Phylogenetic tree construction using Markov chain Monte carlo*. Ph. D. dissertation, Ohio State University, Columbus.
- MAU, B. 1996. *Bayesian phylogenetic inference via Markov chain Monte carlo methods*. Ph. D. dissertation, University of Wisconsin, Madison.
- MAU, B., AND M. NEWTON. 1997. Phylogenetic inference for binary data on dendrograms using Markov chain Monte Carlo. *Journal of Computational and Graphical Statistics* 6:122–131.
- MAU, B., M. NEWTON, AND B. LARGET. 1999. Bayesian phylogenetic inference via Markov chain Monte carlo methods. *Biometrics* 55:1–12.
- METROPOLIS, N., A. W. ROSENBLUTH, M. N. ROSENBLUTH, A. H. TELLER, AND E. TELLER. 1953. Equations of state calculations by fast computing machines. *J. Chem. Phys.* 21:1087–1091.
- MUSE, S. V., AND B. S. GAUT. 1994. A likelihood approach for comparing synonymous and nonsynonymous nucleotide substitution rates with application to the chloroplast genome. *Molecular Biology and Evolution* 11:715–724.



- RANNALA, B., AND Z. YANG. 1996. Probability distribution of molecular evolutionary trees: a new method of phylogenetic inference. *Journal of Molecular Evolution* 43:304–311.
- SCHÖNIGER, M., AND A. VON HAESLER. 1994. A stochastic model and the evolution of autocorrelated DNA sequences. *Molecular Phylogenetics and Evolution* 3:240–247.
- SWOFFORD, D. L. 1998. *PAUP\*: Phylogenetic Analysis Using Parsimony and Other Methods*. Sinauer Associates, Sunderland, MA.
- TAVARÉ, S. 1986. Some probabilistic and statistical problems on the analysis of DNA sequences. Pages 57–86 in *Lectures in Mathematics in the Life Sciences, vol. 17*.
- TIERNEY, L. 1994. Markov chains for exploring posterior distributions (with discussion). *Annals of Statistics* 22:1701–1762.
- YANG, Z. 1993. Maximum likelihood estimation of phylogeny from DNA sequences when substitution rates differ over sites. *Molecular Biology and Evolution* 10:1396–1401.
- YANG, Z. 1994a. Estimating the pattern of nucleotide substitution. *Journal of Molecular Evolution* 39:105–111.
- YANG, Z. 1994b. Maximum likelihood phylogenetic estimation from DNA sequences with variable rates over sites: Approximate methods. *Journal of Molecular Evolution* 39:306–314.
- YANG, Z. 1997. TITLE. *CABIOS* 15:555.
- YANG, Z., AND B. RANNALA. 1997. Bayesian phylogenetic inference using DNA sequences: a Markov chain Monte carlo method. *Molecular Biology and Evolution* 14:717–724.