Command Reference for MrBayes ver. 3.2.2

(c) John P. Huelsenbeck, Fredrik Ronquist
and Maxim Teslenko

****
* *
* 1. Command summary *
* *
****

Commands that are available from the command
line or from a MrBayes block include:

```
About            -- Describes the program
Acknowledgments  -- Shows program acknowledgments
Calibrate        -- Assigns dates to terminals or interior nodes
Charset          -- Assigns a group of sites to a set
Charstat         -- Shows status of characters
Citations        -- Citation of program, models, and methods
Comparetree      -- Compares the trees from two tree files
Constraint       -- Defines a constraint on tree topology
Ctype            -- Assigns ordering for the characters
Databreaks       -- Defines data breaks for autodiscrete gamma model
Delete           -- Deletes taxa from the analysis
Disclaimer       -- Describes program disclaimer
Exclude          -- Excludes sites from the analysis
Execute          -- Executes a file
Help             -- Provides detailed description of commands
Include          -- Includes sites
Link             -- Links parameters across character partitions
Log              -- Logs screen output to a file
Lset             -- Sets the parameters of the likelihood model
Manual           -- Prints a command reference to a text file
Mcmc             -- Starts Markov chain Monte Carlo analysis
Mcmcp            -- Sets parameters of a chain (without starting analysis)
Outgroup         -- Changes outgroup taxon
Pairs            -- Defines nucleotide pairs (doublets) for stem models
Partition        -- Assigns a character partition
Plot             -- Plots parameters from MCMC analysis
Prset            -- Sets the priors for the parameters
Propset          -- Sets proposal probabilities and tuning parameters
```

```
Quit             -- Quits the program
Report           -- Controls how model parameters are reported
Restore          -- Restores taxa
Set              -- Sets run conditions and defines active data partition
Showbeagle       -- Show available BEAGLE resources
Showmatrix       -- Shows current character matrix
Showmcmctrees    -- Shows trees used in mcmc analysis
Showmodel        -- Shows model settings
Showmoves        -- Shows moves for current model
Showparams       -- Shows parameters in current model
Showusertrees    -- Shows user-defined trees
Speciespartition -- Defines a partition of tips into species
Ss               -- Starts stepping-stone sampling
Ssp              -- Sets parameters of stepping-stone analysis (without starting)
Startvals        -- Sets starting values of parameters
Sump             -- Summarizes parameters from MCMC analysis
Sumss            -- Summarizes parameters from stepping-stone analysis
Sumt             -- Summarizes trees from MCMC analysis
Taxastat         -- Shows status of taxa
Taxset           -- Assigns a group of taxa to a set
Unlink           -- Unlinks parameters across character partitions
Version          -- Shows program version

Commands that should be in a NEXUS file (data
block, trees block or taxa block) include:

Begin            -- Denotes beginning of block in file
Dimensions       -- Defines size of character matrix
End              -- Denotes end of a block in file
Endblock         -- Alternative way of denoting end of a block
Format           -- Defines character format in data block
Matrix           -- Defines matrix of characters in data block
Taxlabels        -- Defines taxon labels
Translate        -- Defines alternative names for taxa
Tree             -- Defines a tree

Note that this program supports the use of the shortest unambiguous
spelling of the above commands (e.g., "exe" instead of "execute").
------------------------------------------------------------------------

************************************************************************
*                                                                      *
*  2. MrBayes commands                                                 *
*                                                                      *
************************************************************************

------------------------------------------------------------------------
About
```

This command provides some general information about the program.

Acknowledgments

This command shows the authors' acknowledgments.

Calibrate

This command dates a terminal or interior node in the tree. The format is

```
calibrate <node_name> = <age_prior>
```

where <node_name> is the name of a defined interior constraint node or the name of a terminal node (tip) and <age_prior> is a prior probability distribution on the age of the node. The latter can either be a fixed date or a date drawn from one of the available prior probability distributions. In general, the available prior probability distributions are parameterized in terms of the expected mean age of the distribution to facilitate for users. Some distributions put a positive probability on all ages above 0.0, while others include a minimum-age constraint and sometimes a maximum-age constraint. The available distributions and their parameters are:

```
calibrate <node_name> = fixed(<age>)
calibrate <node_name> = uniform(<min_age>,<max_age>)
calibrate <node_name> = offsetexponential(<min_age>,<mean_age>)
calibrate <node_name> = truncatednormal(<min_age>,<mean_age>,<stdev>)
calibrate <node_name> = lognormal(<mean_age>,<stdev>)
calibrate <node_name> = offsetlognormal(<min_age>,<mean_age>,<stdev>)
calibrate <node_name> = gamma(<mean_age>,<stdev>)
calibrate <node_name> = offsetgamma(<min_age>,<mean_age>,<stdev>)
```

Note that mean_age is always the mean age and stdev the standard deviation of the distribution measured in user-defined time units. This way of specifying the distribution parameters is often different from the parameterization used elsewhere in the program. For instance, the standard parameters of the gamma distribution used by MrBayes are shape (alpha) and rate (beta). If you want to use the standard parameterization, the conversions are as follows:

```
exponential distributon: mean = 1 / rate
gamma distributon:       mean    = alpha / beta
                         st.dev. = square_root( alpha / beta^2 )
lognormal distributon:   mean    = exp( mean_log + st.dev._log^2/2 )
                         st.dev. = square_root(( exp( st.dev._log^2 ) - 1 )
                                   * ( exp( 2*mean_log + st.dev._log^2 ))
```

The truncated normal distribution is an exception in that the mean_age and stdev parameters are the mean and standard deviation of the underlying non-truncated normal distribution. The truncation will cause the modified distribution to have a higher mean and lower standard deviation. The magnitude of that effect depends on how much of the tail of the distribution is removed.

Note that previous to version 3.2.2, MrBayes used the standard rate parameterization of the offset exponential. This should not cause a problem in most cases because the old parameterization will result in an error in more recent versions of MrBayes, and the likely source of the error is given in the error message.

For a practical example, assume that we had three fossil terminals named 'FossilA', 'FossilB', and 'FossilC'. Assume further that we want to fix the age of FossilA to 100.0 million years, we think that FossilB is somewhere between 100.0 and 200.0 million years old, and that FossilC is at least 300.0 million years old, possibly older but relatively unlikely to be more than 400.0 million years old. Then we might use the commands:

```
calibrate FossilA = fixed(100) FossilB = uniform(100,200)
calibrate FossilC = offsetexponential(300,400)
```

Note that it is possible to give more than one calibration for each 'calibrate' statement. Thus, 'calibrate FossilA=<setting> FossilB=<setting>' would be a valid statement.

To actually use the calibrations to obtain dated trees, you also need to set a clock model using relevant 'brlenspr' and 'nodeagepr' options of the 'prset' command. You may also want to examine the 'clockvarpr' and 'clockratepr' options. Furthermore, you need to activate the relevant constraint(s) using 'topologypr', if you use any dated interior nodes in the tree.

You may wish to remove a calibration from an interior or terminal node, which has previously been calibrated. You can do that using

```
calibrate <node_name> = unconstrained
```


Charset

This command defines a character set. The format for the charset command is

```
charset <name> = <character numbers>
```

For example, "charset first_pos = 1-720\3" defines a character set

called "first_pos" that includes every third site from 1 to 720.
The character set name cannot have any spaces in it. The slash (\)
is a nifty way of telling the program to assign every third (or
second, or fifth, or whatever) character to the character set.
This option is best used not from the command line, but rather as a
line in the mrbayes block of a file. Note that you can use "." to
stand in for the last character (e.g., charset 1-.\3).

Charstat

This command shows the status of all the characters. The correct usage
is

   charstat

After typing "charstat", the character number, whether it is excluded
or included, and the partition identity are shown. The output is paused
every 100 characters. This pause can be turned off by setting autoclose
to "yes" (set autoclose=yes).

Citations

This command shows a thorough list of citations you may consider using
when publishing the results of a MrBayes analysis.

Comparetree

This command compares the trees in two files, called "filename1" and
"filename2". It will output a bivariate plot of the split frequencies
as well as plots of the tree distance as a function of the generation. The
plots can be used to get a quick indication of whether two runs have con-
verged onto the same set of trees. The "Comparetree" command will also
produce a ".pairs" file and a ".dists" file (these file endings are added
to the end of the "Outputname"). The ".pairs" file contains the paired
split frequencies from the two tree samples; the ".dists" file contains the
tree distance values.

Note that the "Sumt" command provides a different set of convergence diag-
nostics tools that you may also want to explore. Unlike "Comparetree", "Sumt"
can compare more than two tree samples and will calculate consensus trees and
split frequencies from the pooled samples.

Options:

Relburnin   -- If this option is set to 'Yes', then a proportion of the

```
                samples will be discarded as burnin when calculating summary
                statistics. The proportion to be discarded is set with
                Burninfrac (see below). When the Relburnin option is set to
                'No', then a specific number of samples is discarded instead.
                This number is set by Burnin (see below). Note that the
                burnin setting is shared with the 'mcmc', 'sumt', 'sump' and
                'plot' commands.
Burnin        -- Determines the number of samples (not generations) that will
                be discarded when summary statistics are calculated. The
                value of this option is only relevant when Relburnin is set
                to 'No'.
BurninFrac    -- Determines the fraction of samples that will be discarded
                when summary statistics are calculated. The value of this
                option is only relevant when Relburnin is set to 'Yes'.
                Example: A value for this option of 0.25 means that 25% of
                the samples will be discarded.
Minpartfreq   -- The minimum probability of partitions to include in summary
                statistics.
Filename1     -- The name of the first tree file to compare.
Filename2     -- The name of the second tree file to compare.
Outputname    -- Name of the file to which 'comparetree' results will be
                printed.

Current settings:

Parameter       Options                 Current Setting
-------------------------------------------------------
Relburnin       Yes/No                  Yes
Burnin          <number>                0
Burninfrac      <number>                0.25
Minpartfreq     <number>                0.00
Filename1       <name>                  temp.t
Filename2       <name>                  temp.t
Outputname      <name>                  temp.comp

------------------------------------------------------------------------
------------------------------------------------------------------------
Constraint

This command defines a tree constraint. The format for the constraint
command is

   constraint <name> [hard|negative|partial] = <taxon list> [:<taxon list>]

There are three types of constraint implemented in MrBayes. The type of the
constraint is specified by using one of the three keywords 'hard', 'negative',
or 'partial' right after the name of the constraint. If no type is specified,
then the constraint is assumed to be 'hard'.
```

In a rooted tree, a 'hard' constraint forces the taxa in the list to form a monophyletic group. In an unrooted tree, the taxon split that separates the taxa in the list from other taxa is forced to be present. The interpretation of this depends on whether the tree is rooted on a taxon outside the list or a taxon in the list. If the outgroup is excluded , the taxa in the list are assumed to form a monophyletic group, but if the outgroup is included, it is the taxa that are not in the list that are forced together.

A 'negative' constraint bans all the trees that have the listed taxa in the same subtree. In other words, it is the opposite of a hard constraint.

A 'partial' or backbone constraint is defined in terms of two sets of taxa separated by a colon character. The constraint forces all taxa in the first list to form a monophyletic group that does not include any taxon in the second list. Taxa that are not included in either list can be placed in any position on the tree, either inside or outside the constrained group. In an unrooted tree, the two taxon lists can be switched with each other with no effect. For a rooted tree, it is the taxa in the first list that have to be monophyletic, that is, these taxa must share a common ancestor not shared with any taxon in the second list. The taxa in the second list may or may not fall in a monophyletic group depending on the rooting of the tree.

A list of taxa can be specified using a taxset, taxon names, taxon numbers, or any combination of the above, sepatated by spaces. The constraint is treated as an absolute requirement of trees, that is, trees that are not compatible with the constraint have zero prior (and hence zero posterior) probabilty.

If you are interested in inferring ancestral states for a particular node, you need to 'hard' constrain that node first using the 'constraint' command. The same applies if you wish to calibrate an interior node in a dated analysis. For more information on how to infer ancestral states, see the help for the 'report' command. For more on dating, see the 'calibrate' command.

It is important to note that simply defining a constraint using this command is not sufficient for the program to actually implement the constraint in an analysis. You must also enforce the constraints using 'prset topologypr = constraints (<list of constraints>)'. For more information on this, see the help on the 'prset' command.

Examples:

```
constraint myclade = Homo Pan Gorilla
```

Defines a hard constraint forcing Homo, Pan, and Gorilla to form a monophyletic group or a split that does not include any other taxa.

```
constraint forbiddenclade negative = Homo Pan Gorilla
```

Defines a negative constraint that associates all trees where Homon, Pan, and Gorilla form a monophyletic group with zero posterior probability. In other words, such trees will not be sampled during MCMC.

```
constraint backbone partial = Homo Gorilla : Mus
```

Defines a partial constraint that keeps Mus outside of the clade defined by the most recent common ancestor of Homo and Gorilla. Other taxa are allowed to sit anywhere in the tree. Note that this particular constraint is meaningless in unrooted trees. MrBayes does not assume anything about the position of the outgroup unless it is explicitly included in the partial constraint. Therefore, a partial constraint must have at least two taxa on each side of the ':' to be useful in analyses of unrooted trees. The case is different for rooted trees, where it is sufficient for a partial constraint to have more than one taxon before the ':', as in the example given above, to constrain tree space.

To define a more complex constraint tree, simply combine constraints into a list when issuing the 'prset topologypr' command.

Ctype

This command sets the character ordering for standard-type data. The correct usage is:

```
ctype <ordering>:<characters>
```

The available options for the <ordering> specifier are:

```
unordered    -- Movement directly from one state to another is
                allowed in an instant of time.
ordered      -- Movement is only allowed between adjacent characters.
                For example, perhaps only between 0 <-> 1 and 1 <-> 2
                for a three state character ordered as 0 - 1 - 2.
irreversible -- Rates of change for losses are 0.
```

The characters to which the ordering is applied is specified in manner that is identical to commands such as "include" or "exclude". For example,

```
ctype ordered: 10 23 45
```

defines characters 10, 23, and 45 to be of type ordered. Similarly,

```
   ctype irreversible: 54 - 67  71-92
```

defines characters 54 to 67 and characters 71 to 92 to be of type irreversible. You can use the "." to denote the last character, and "all" to denote all of the characters. Finally, you can use the specifier "\" to apply the ordering to every n-th character or you can use predefined charsets to specify the character.

Only one ordering can be used on any specific application of ctype. If you want to apply different orderings to different characters, then you need to use ctype multiple times. For example,

```
   ctype ordered: 1-50
   ctype irreversible: 51-100
```

sets characters 1 to 50 to be ordered and characters 51 to 100 to be irreversible.

The ctype command is only sensible with morphological (here called "standard") characters. The program ignores attempts to apply character orderings to other types of characters, such as DNA characters.

Databreaks

This command is used to specify breaks in your input data matrix. Your data may be a mixture of genes or a mixture of different types of data. Some of the models implemented by MrBayes account for nonindependence at adjacent characters. The autocorrelated gamma model, for example, allows rates at adjacent sites to be correlated. However, there is no way for such a model to tell whether two sites, adjacent in the matrix, are actually separated by many kilobases or megabases in the genome. The databreaks command allows you to specify such breaks. The correct usage is:

```
   databreaks <break 1> <break 2> <break 3> ...
```

For example, say you have a data matrix of 3204 characters that include nucleotide data from three genes. The first gene covers characters 1 to 970, the second gene covers characters 971 to 2567, and the third gene covers characters 2568 to 3204. Also, let's assume that the genes are not directly adjacent to one another in the genome, as might be likely if you have mitochondrial sequences. In this case, you can specify breaks between the genes using:

```
   databreaks 970 2567;
```

The first break, between genes one and two, is after character 970 and

the second break, between genes two and three, is after character 2567.

Delete

This command deletes taxa from the analysis. The correct usage is:

   delete <name and/or number and/or taxset> ...

A list of the taxon names or taxon numbers (labelled 1 to ntax in the order in the matrix) or taxset(s) can be used. For example, the following:

   delete 1 2 Homo_sapiens

deletes taxa 1, 2, and the taxon labelled Homo_sapiens from the analysis. You can also use "all" to delete all of the taxa. For example,

   delete all

deletes all of the taxa from the analysis. Of course, a phylogenetic analysis that does not include any taxa is fairly uninteresting.

Disclaimer

This command shows the disclaimer for the program. In short, the disclaimer states that the authors are not responsible for any silly things you may do to your computer or any unforseen but possibly nasty things the computer program may inadvertently do to you.

Exclude

This command excludes characters from the analysis. The correct usage is

   exclude <number> <number> <number>

or

   exclude <number> - <number>

or

   exclude <charset>

or some combination thereof. Moreover, you can use the specifier "\" to exclude every nth character. For example, the following

```
   exclude 1-100\3
```

would exclude every third character. As a specific example,

```
   exclude 2 3 10-14 22
```

excludes sites 2, 3, 10, 11, 12, 13, 14, and 22 from the analysis. Also,

```
   exclude all
```

excludes all of the characters from the analysis. Excluding all characters does not leave you much information for inferring phylogeny.

Execute

This command executes a file called <file name>. The correct usage is:

```
   execute <file name>
```

For example,

```
   execute replicase.nex
```

would execute the file named "replicase.nex". This file must be in the same directory as the executable.

Help

This command provides useful information on the use of this program. The correct usage is

```
   help
```

which gives a list of all available commands with a brief description of each or

```
   help <command>
```

which gives detailed information on the use of <command>.

Include

This command includes characters that were previously excluded from the analysis. The correct usage is

```
   include <number> <number> <number>
```

or

```
   include <number> - <number>
```

or

```
   include <charset>
```

or some combination thereof. Moreover, you can use the specifier "\" to include every nth character. For example, the following

```
   include 1-100\3
```

would include every third character. As a specific example,

```
   include 2 3 10-14 22
```

includes sites 2, 3, 10, 11, 12, 13, 14, and 22 from the analysis. Also,

```
   include all
```

includes all of the characters in the analysis. Including all of the characters (even if many of them are bad) is a very total-evidence-like thing to do. Doing this will make a certain group of people very happy. On the other hand, simply using this program would make those same people unhappy.

Link

This command links model parameters across partitions of the data. The correct usage is:

```
   link <parameter name> = (<all> or <partition list>)
```

The list of parameters that can be linked includes:

```
   Tratio          -- Transition/transversion rate ratio
   Revmat          -- Substitution rates of GTR model
   Omega           -- Nonsynonymous/synonymous rate ratio
   Statefreq       -- Character state frequencies
   Shape           -- Gamma shape parameter
   Pinvar          -- Proportion of invariable sites
   Correlation     -- Correlation parameter of autodiscrete gamma
   Switchrates     -- Switching rates for covarion model
   Brlens          -- Branch lengths of tree
```

```
   Topology        -- Topology of tree
   Speciationrates -- Speciation rates for birth-death process
   Extinctionrates -- Extinction rates for birth-death process
   Theta           -- Parameter for coalescence process
   Growthrate      -- Growth rate of coalescence process
   Aamodel         -- Aminoacid rate matrix
   Cpprate         -- Rate of Compound Poisson Process (CPP)
   Cppmultdev      -- Standard dev. of CPP rate multipliers (log scale)
   Cppevents       -- CPP events
   TK02var         -- Variance increase in TK02 relaxed clock model
   TK02branchrates -- Branch rates of TK02 relaxed clock model
   Igrvar          -- Variance increase in IGR relaxed clock model
   Igrbranchlens   -- Branch lengths of IGR relaxed clock model

For example,

   link shape=(all)

links the gamma shape parameter across all partitions of the data.
You can use "showmodel" to see the current linking status of the
characters. For more information on this command, see the help menu
for link's converse, unlink ("help unlink");
------------------------------------------------------------------------
------------------------------------------------------------------------
Log

This command allows output to the screen to also be output to a file.
The useage is:

   log start/stop filename=<name> append/replace

The options are:

Start/Stop     -- Starts or stops logging of output to file.
Append/Replace -- Either append to or replace existing file.
Filename       -- Name of log file (currently, the name of the log
                  file is "log.out").
------------------------------------------------------------------------
------------------------------------------------------------------------
Lset

This command sets the parameters of the likelihood model. The likelihood
function is the probability of observing the data conditional on the phylo-
genetic model. In order to calculate the likelihood, you must assume a
model of character change. This command lets you tailor the biological
assumptions made in the phylogenetic model. The correct usage is

   lset <parameter>=<option> ... <parameter>=<option>
```

For example, "lset nst=6 rates=gamma" would set the model to a general model of DNA substition (the GTR) with gamma-distributed rate variation across sites.

Options:

Applyto -- This option allows you to apply the lset commands to specific partitions. This command should be the first in the list of commands specified in lset. Moreover, it only makes sense to be using this command if the data have been partitioned. A default partition is set on execution of a matrix. If the data are homogeneous (i.e., all of the same data type), then this partition will not subdivide the characters. Up to 30 other partitions can be defined, and you can switch among them using "set partition=<partition name>". Now, you may want to specify different models to different partitions of the data. Applyto allows you to do this. For example, say you have partitioned the data by codon position, and you want to apply a nst=2 model to the first two partitions and nst=6 to the last. This could be implemented in two uses of lset:

```
lset applyto=(1,2) nst=2

lset applyto=(3) nst=6
```

The first applies the parameters after "applyto" to the first and second partitions. The second lset applies nst=6 to the third partition. You can also use applyto=(all), which attempts to apply the parameter settings to all of the data partitions. Importantly, if the option is not consistent with the data in the partition, the program will not apply the lset option to that partition.

Nucmodel -- This specifies the general form of the nucleotide substitution model. The options are "4by4" [the standard model of DNA substitution in which there are only four states (A,C,G,T/U)], "doublet" (a model appropriate for modelling the stem regions of ribosomal genes where the state space is the 16 doublets of nucleotides), "codon" (the substitution model is expanded around triplets of nucleotides--a codon), and "Protein" (triplets of nucleotides are translated to amino acids, which form the basis of the substitution model).

Nst -- Sets the number of substitution types: "1" constrains all of the rates to be the same (e.g., a JC69 or F81 model); "2" allows transitions and transversions to have potentially different rates (e.g., a K80 or HKY85 model); "6" allows all rates to be different, subject to the constraint of time-reversibility (e.g., a GTR model). Finally, 'nst' can be set to 'mixed', which

results in the Markov chain sampling over the space of all possible reversible substitution models, including the GTR model and all models that can be derived from it model by grouping the six rates in various combinations. This includes all the named models above and a large number of others, with or without name.

Code -- Enforces the use of a particular genetic code. The default is the universal code. Other options include "vertmt" for vertebrate mitocondrial DNA, "mycoplasma", "yeast", "ciliates", and "metmt" (for metazoan mitochondrial DNA except vertebrates).

Ploidy -- Specifies the ploidy of the organism. Options are "Haploid", "Diploid" or "Zlinked". This option is used when a coalescence prior is used on trees.

Rates -- Sets the model for among-site rate variation. In general, the rate at a site is considered to be an unknown random variable. The valid options are:

* equal -- No rate variation across sites.
* gamma -- Gamma-distributed rates across sites. The rate at a site is drawn from a gamma distribution. The gamma distribution has a single parameter that describes how much rates vary.
* adgamma -- Autocorrelated rates across sites. The marginal rate distribution is gamma, but adjacent sites have correlated rates.
* propinv -- A proportion of the sites are invariable.
* invgamma -- A proportion of the sites are invariable while the rate for the remaining sites are drawn from a gamma distribution.

Note that MrBayes versions 2.0 and earlier supported options that allowed site specific rates (e.g., ssgamma). In versions 3.0 and later, site specific rates are allowed, but set using the 'prset ratepr' command for each partition.

Ngammacat -- Sets the number of rate categories for the gamma distribution. The gamma distribution is continuous. However, it is virtually impossible to calculate likelihoods under the continuous gamma distribution. Hence, an approximation to the continuous gamma is used; the gamma distribution is broken into ncat categories of equal weight (1/ncat). The mean rate for each category represents the rate for the entire cateogry. This option allows you to specify how many rate categories to use when approximating the gamma. The approximation is better as ncat is increased. In practice, "ncat=4" does a reasonable job of approximating the continuous gamma.

Nbetacat -- Sets the number of rate categories for the beta distribution. A symmetric beta distribution is used to model the stationary frequencies when morphological data are used. This option specifies how well the beta distribution will be approximated.

Omegavar -- Allows the nonsynonymous/synonymous rate ratio (omega) to vary across codons. Ny98 assumes that there are three classes, with potentially different omega values (omega1, omega2, omega3): omega2 = 1; 0 < omega1 <1; and omega3 > 1. Like the Ny98 model, the M3 model has three omega classes. However, their values are less constrained, with omega1 < omega2 < omega3. The default (omegavar = equal) has no variation on omega across sites.

Covarion -- This forces the use of a covarion-like model of substitution for nucleotide or amino acid data. The valid options are "yes" and "no". The covarion model allows the rate at a site to change over its evolutionary history. Specifically, the site is either on or off. When it is off, no substitutions are possible. When the process is on, substitutions occur according to a specified substitution model (specified using the other lset options).

Coding -- This specifies how characters were sampled. If all site patterns had the possibility of being sampled, then "all" should be specified (the default). Otherwise "variable" (only variable characters had the possibility of being sampled), "noabsence" (characters for which all taxa were coded as absent were not sampled), and "nopresence" (characters for which all taxa were coded as present were not sampled. "All" works for all data types. However, the others only work for morphological (all/variable) or restriction site (all/variable/noabsence/nopresence) data.

Parsmodel -- This forces calculation under the so-called parsimony model described by Tuffley and Steel (1998). The options are "yes" or "no". Note that the biological assumptions of this model are anything but parsimonious. In fact, this model assumes many more parameters than the next most complicated model implemented in this program. If you really believe that the parsimony model makes the biological assumptions described by Tuffley and Steel, then the parsimony method is miss-named.

Default model settings:

Parameter	Options	Current Setting
Nucmodel	4by4/Doublet/Codon/Protein	4by4
Nst	1/2/6/Mixed	1
Code	Universal/Vertmt/Mycoplasma/ Yeast/Ciliates/Metmt	Universal
Ploidy	Haploid/Diploid/Zlinked	Diploid
Rates	Equal/Gamma/Propinv/Invgamma/Adgamma	Equal
Ngammacat	<number>	4
Nbetacat	<number>	5
Omegavar	Equal/Ny98/M3	Equal
Covarion	No/Yes	No

```
Coding       All/Variable/Noabsencesites/
             Nopresencesites                        All
Parsmodel    No/Yes                                 No
-------------------------------------------------------------------

-------------------------------------------------------------------------
Manual

This command allows you to generate a text file containing help information
on all the available commands. This text file can be used as an up-to-date
command reference. You can set the name of the text file using the
"filename" option; the default is "commref_mb<version>.txt".

Parameter       Options                  Current Setting
-----------------------------------------------------
Filename        <name>                   commref_mb3.2.2.txt

-------------------------------------------------------------------------
-------------------------------------------------------------------------
Mcmc

This command starts the Markov chain Monte Carlo (MCMC) analysis. The
posterior probability of phylogenetic trees (and other parameters of the
substitution model) cannot be determined analytically. Instead, MCMC is
used to approximate the posterior probabilities of trees by drawing
(dependent) samples from the posterior distribution. This program can
implement a variant of MCMC called "Metropolis-coupled Markov chain Monte
Carlo", or MCMCMC for short. Basically, "Nchains" are run, with
Nchains - 1 of them heated. The chains are labelled 1, 2, ..., Nchains.
The heat that is applied to the i-th chain is B = 1 / (1 + temp X i). B
is the power to which the posterior probability is raised. When B = 0, all
trees have equal probability and the chain freely visits trees. B = 1 is
the "cold" chain (or the distribution of interest). MCMCMC can mix
better than ordinary MCMC; after all of the chains have gone through
one cycle, two chains are chosen at random and an attempt is made to
swap the states (with the probability of a swap being determined by the
Metropolis et al. equation). This allows the chain to potentially jump
a valley in a single bound. The correct usage is

   mcmc <parameter> = <value> ... <parameter> = <value>

For example,

   mcmc ngen=100000 nchains=4 temp=0.5

performs a MCMCMC analysis with four chains with the temperature set to
0.5. The chains would be run for 100,000 cycles.
```

Options:

Ngen -- This option sets the number of cycles for the MCMC algorithm. This should be a big number as you want the chain to first reach stationarity, and then remain there for enough time to take lots of samples.

Nruns -- How many independent analyses are started simultaneously.

Nchains -- How many chains are run for each analysis for the MCMCMC variant. The default is 4: 1 cold chain and 3 heated chains. If Nchains is set to 1, MrBayes will use regular MCMC sampling, without heating.

Temp -- The temperature parameter for heating the chains. The higher the temperature, the more likely the heated chains are to move between isolated peaks in the posterior distribution. However, excessive heating may lead to very low acceptance rates for swaps between different chains. Before changing the default setting, however, note that the acceptance rates of swaps tend to fluctuate during the burn-in phase of the run.

Reweight -- Here, you specify three numbers, that respectively represent the percentage of characters to decrease in weight, the percentage of characters to increase in weight, and the increment. An increase/decrease in weight is acheived by replicating/removing a character in the matrix. This is only done to non-cold chains. The format for this parameter is "reweight=(<number>,<number>)" or "reweight=(<number>, <number>,<number>)".

Swapfreq -- This specifies how often swaps of states between chains are attempted. You must be running at least two chains for this option to be relevant. The default is Swapfreq=1, resulting in Nswaps (see below) swaps being tried each generation of the run. If Swapfreq is set to 10, then Nswaps swaps will be tried every tenth generation of the run.

Nswaps -- The number of swaps tried for each swapping generation of the chain (see also Swapfreq).

Samplefreq -- This specifies how often the Markov chain is sampled. You can sample the chain every cycle, but this results in very large output files. Thinning the chain is a way of making these files smaller and making the samples more independent.

Printfreq -- This specifies how often information about the chain is printed to the screen.

Printall -- If set to NO, only cold chains in a MCMC analysis are printed to screen. If set to YES, both cold and heated chains will be output. This setting only affects the printing to screen, it does not change the way values are written to file.

Printmax -- The maximum number of chains to print to screen.

Mcmcdiagn -- Determines whether acceptance ratios of moves and swaps will be printed to file. The file will be named similarly to the '.p' and '.t' files, but will have the ending '.mcmc'. If

more than one independent analysis is run simultaneously (see Nruns below), convergence diagnostics for tree topology will also be printed to this file. The convergence diagnostic used is the average standard deviation in partition frequency values across independent analyses. The Burnin setting (see below) determines how many samples will be discarded as burnin before calculating the partition frequencies. The Minpartfreq setting (see below) determines the minimum partition frequency required for a partition to be included in the calculation. As the independent analyses approach stationarity (converge), the value of the diagnostic is expected to approach zero.

Diagnfreq -- The number of generations between the calculation of MCMC diagnostics (see Mcmcdiagn above).

Diagnstat -- The statistic to use for run-time convergence diagnostics. Choices are 'Avgstddev' for average standard deviation of split frequencies and 'Maxstddev' for maximum standard deviation of split frequencies.

Savetrees -- If you are using a relative burnin for run-time convergence diagnostics, tree samples need to be deleted from split frequency counters as the cut-off point for the burnin moves during the run. If 'Savetrees' is set to 'No', tree samples to be discarded are read back in from file. If 'Savetrees' is set to 'Yes', the tree samples to be removed will be stored in the internal memory instead. This can use up a lot of memory in large analyses.

Minpartfreq -- The minimum frequency required for a partition to be included in the calculation of the topology convergence diagnostic. The partition is included if the minimum frequency is reached in at least one of the independent tree samples that are compared.

Allchains -- If this option is set to YES, acceptance ratios for moves are recorded for all chains, cold or heated. By default, only the acceptance ratios for the cold chain are recorded.

Allcomps -- If this option is set to YES, topological convergence diagnostics are calculated over all pairwise comparisons of runs. If it is set to NO, only the overall value is reported.

Relburnin -- If this option is set to YES, then a proportion of the sampled values will be discarded as burnin when calculating the convergence diagnostic. The proportion to be discarded is set with Burninfrac (see below). When the Relburnin option is set to NO, then a specific number of samples will be discarded instead. This number is set by Burnin (see below).

Burnin -- Determines the number of samples (not generations) that will be discarded when convergence diagnostics are calculated. The value of this option is only relevant when Relburnin is set to NO.

BurninFrac -- Determines the fraction of samples that will be discarded when convergence diagnostics are calculated. The value of

```
                this option is only relevant when Relburnin is set to YES.
                Example: A value for this option of 0.25 means that 25% of
                the samples will be discarded.
Stoprule     -- If this option is set to NO, then the chain is run the number
                of generations determined by Ngen. If it is set to YES, and
                topological convergence diagnostics are calculated (Mcmcdiagn
                is set to YES), then the chain will be stopped before the pre-
                determined number of generations if the convergence diagnostic
                falls below the stop value.
Stopval      -- The critical value for the topological convergence diagnostic.
                Only used when Stoprule and Mcmcdiagn are set to yes, and
                more than one analysis is run simultaneously (Nruns > 1).
Checkpoint   -- If this parameter is set to 'Yes', all the current parameter
                values of all chains will be printed to a check-pointing file
                every 'Checkfreq' generation of the analysis. The file will be
                named <Filename>.ckp and allows you to restart the analysis
                from the last check point. This can be handy if you are
                running a long analysis and want to extend it, or if there is
                a risk that a long analysis will be inadvertently interupted
                by hardware failure or other factors that are out of your
                control.
Checkfreq    -- The number of generations between check-pointing. See the
                'Checkpoint' parameter above for more information.
Filename     -- The name of the files that will be generated. Two files
                are generated: "<Filename>.t" and "<Filename>.p".
                The .t file contains the trees whereas the .p file con-
                tains the sampled values of the parameters.
Startparams  -- The starting values for the model parameters are set to
                arbitrary or random values when the parameters are created.
                These starting values can be altered using the 'Startvals'
                command. The 'Startparams=reset' option allows you to reset
                the starting values to the default at the start of the ana-
                lysis, overriding any previous user-defined starting values.
                Under the default option, 'current', the chains will use the
                current starting values.
Starttree    -- The starting tree(s) for the chain can either be randomly
                selected or user-defined. It might be a good idea to
                start from randomly chosen trees; convergence seems
                likely if independently run chains, each of which
                started from different random trees, converge to the same
                answer. If you want the chain to start from user-defined
                trees instead, you first need to read in your tree(s) from a
                Nexus file with a 'trees' block, and then you need to set the
                starting tree(s) using the 'Startvals' command. Finally, you
                need to make sure that 'Starttree' is set to 'current'. If
                you do not set the starting tree(s), the chains will start
                with random trees. Setting 'Starttree' to 'random' causes
                new starting trees to be drawn randomly at the start of the
```

```
                run, overwriting any previous user-defined starting trees.
Nperts       -- This is the number of random perturbations to apply to the
                user starting tree. This allows you to have something
                between completely random and user-defined trees start
                the chain.
Savebrlens   -- This specifies whether branch length information is
                saved on the trees.
Data         -- When Data is set to NO, the chain is run without data. This
                should be used only for examining induced priors. DO NOT SET
                'DATA' TO 'NO' UNLESS YOU KNOW WHAT YOU ARE DOING!
Ordertaxa    -- Determines whether taxa should be ordered before trees are
                printed to file. If set to 'Yes', terminals in the sampled
                trees will be reordered to match the order of the taxa in the
                data matrix as closely as possible. By default, trees will be
                printed without reordering of taxa.
Append       -- Set this to 'Yes' to append the results of the current run to
                a previous run. MrBayes will first read in the results of the
                previous run (number of generations and sampled splits) and
                will then continue that run where you left it off. Make sure
                that the output file names used in the previous run are the
                same as those in the current run.
Autotune     -- Set this to 'Yes' to autotune the proposals that change
                substitution model parameters. When set to 'No', the tuning
                parameters are fixed to their starting values. Note that the
                autotuning occurs independently for each chain. The target
                acceptance rate for each move can be changed using the
                'Propset' command.
Tunefreq     -- When a proposal has been tried 'Tunefreq' times, its tuning
                parameter is adjusted to reach the target acceptance rate
                if 'Autotune' is set to 'Yes'.

Parameter       Options               Current Setting
-------------------------------------------------------
Ngen            <number>              1000000
Nruns           <number>              2
Nchains         <number>              4
Temp            <number>              0.100000
Reweight        <number>,<number>     0.00 v 0.00 ^
Swapfreq        <number>              1
Nswaps          <number>              1
Samplefreq      <number>              500
Printfreq       <number>              500
Printall        Yes/No                Yes
Printmax        <number>              8
Mcmcdiagn       Yes/No                Yes
Diagnfreq       <number>              5000
Diagnstat       Avgstddev/Maxstddev   Avgstddev
Minpartfreq     <number>              0.10
```

Allchains	Yes/No	No
Allcomps	Yes/No	No
Relburnin	Yes/No	Yes
Burnin	<number>	0
Burninfrac	<number>	0.25
Stoprule	Yes/No	No
Stopval	<number>	0.05
Savetrees	Yes/No	No
Checkpoint	Yes/No	Yes
Checkfreq	<number>	100000
Filename	<name>	temp.out.<p/t>
Startparams	Current/Reset	Current
Starttree	Current/Random/ Parsimony	Current
Nperts	<number>	0
Data	Yes/No	Yes
Ordertaxa	Yes/No	No
Append	Yes/No	No
Autotune	Yes/No	Yes
Tunefreq	<number>	100

Mcmcp

This command sets the parameters of the Markov chain Monte Carlo (MCMC) analysis without actually starting the chain. This command is identical in all respects to Mcmc, except that the analysis will not start after this command is issued. For more details on the options, check the help menu for Mcmc.

Parameter	Options	Current Setting
Ngen	<number>	1000000
Nruns	<number>	2
Nchains	<number>	4
Temp	<number>	0.100000
Reweight	<number>,<number>	0.00 v 0.00 ^
Swapfreq	<number>	1
Nswaps	<number>	1
Samplefreq	<number>	500
Printfreq	<number>	500
Printall	Yes/No	Yes
Printmax	<number>	8
Mcmcdiagn	Yes/No	Yes
Diagnfreq	<number>	5000
Diagnstat	Avgstddev/Maxstddev	Avgstddev
Minpartfreq	<number>	0.10

Allchains	Yes/No	No
Allcomps	Yes/No	No
Relburnin	Yes/No	Yes
Burnin	<number>	0
Burninfrac	<number>	0.25
Stoprule	Yes/No	No
Stopval	<number>	0.05
Savetrees	Yes/No	No
Checkpoint	Yes/No	Yes
Checkfreq	<number>	100000
Filename	<name>	temp.out.<p/t>
Startparams	Current/Reset	Current
Starttree	Current/Random/ Parsimony	Current
Nperts	<number>	0
Data	Yes/No	Yes
Ordertaxa	Yes/No	No
Append	Yes/No	No
Autotune	Yes/No	Yes
Tunefreq	<number>	100

Outgroup

This command assigns a taxon to the outgroup. The correct usage is:

```
outgroup <number>/<taxon name>
```

For example, "outgroup 3" assigns the third taxon in the matrix to be the outgroup. Similarly, "outgroup Homo_sapiens" assings the taxon "Homo_sapiens" to be the outgroup (assuming that there is a taxon named "Homo_sapiens" in the matrix). Only a single taxon can be assigned to be the outgroup.

Pairs

This command is used to specify pairs of nucleotides. For example, your data may be RNA sequences with a known secondary structure of stems and loops. Substitutions in nucleotides involved in a Watson-Crick pairing in stems are not strictly independent; a change in one changes the probability of a change in the partner. A solution to this problem is to expand the model around the pair of nucleotides in the stem. This command allows you to do this. The correct usage is:

```
pairs <NUC1>:<NUC2>, <NUC1>:<NUC2>,..., <NUC1>:<NUC2>;
```

For example,

```
   pairs 30:56, 31:55, 32:54, 33:53, 34:52, 35:51, 36:50;
```

specifies pairings between nucleotides 30 and 56, 31 and 55, etc. Only nucleotide data (DNA or RNA) may be paired using this command. Note that in order for the program to actually implement a "doublet" model involving a 16 X 16 rate matrix, you must specify that the structure of the model is 16 X 16 using "lset nucmodel=doublet".

Partition

This command allows you to specify a character partition. The format for this command is

```
   partition <name> = <num parts>:<chars in first>, ...,<chars in last>
```

For example, "partition by_codon = 3:1st_pos,2nd_pos,3rd_pos" specifies a partition called "by_codon" which consists of three parts (first, second, and third codon positions). Here, we are assuming that the sites in each partition were defined using the charset command. You can specify a partition without using charset as follows:

```
   partition by_codon = 3:1 4 6 9 12,2 5 7 10 13,3 6 8 11 14
```

However, we recommend that you use the charsets to define a set of characters and then use these predefined sets when defining the partition. Also, it makes more sense to define a partition as a line in the mrbayes block than to issue the command from the command line (then again, you may be a masochist, and want to do extra work).

Plot

This command plots specified parameters in the .p file or one of the .p files created during an MCMC analysis. An x-y graph of the parameter over the course of the chain is created. The command can be useful for visually diagnosing convergence for many of the parameters of the phylogenetic model. The parameter to be plotted is specified by the "parameter" option. Several parameters can be plotted at once by using the "match" option, which has a default value of "perfect". For example, if you were to set "parameter = pi" and "match = consistentwith", then all of the state frequency parameters would be plotted. You can also set "match=all", in which case all of the parameters are plotted.

Note that the "Sump" command provides a different set of convergence diag-

nostics tools that you may also want to explore. Unlike "Plot", "Sump" can compare two or more parameter samples and will calculate convergence diagnostics as wel as parameter summaries for the pooled sample.

Options:

```
Relburnin     -- If this option is set to 'Yes', then a proportion of the
                 samples will be discarded as burnin when creating the plot.
                 The proportion to be discarded is set with Burninfrac (see
                 Burninfrac below). When the Relburnin option is set to 'No',
                 then a specific number of samples is discarded instead. This
                 number is set by Burnin (see below). Note that the burnin
                 setting is shared across the 'comparetree', 'sump' and 'sumt'
                 commands.
Burnin        -- Determines the number of samples (not generations) that will
                 be discarded when summary statistics are calculated. The
                 value of this option is only relevant when Relburnin is set
                 to 'No'.
Burninfrac    -- Determines the fraction of samples that will be discarded
                 when creating a plot. The value of this parameter is only
                 relevant when Relburnin is set to 'Yes'. Example: A value of
                 this option of 0.25 means that 25% of the samples will be
                 discarded.
Filename      -- The name of the file to plot.
Parameter     -- Specification of parameters to be plotted. See above for
                 details.
Match         -- Specifies how to match parameter names to the Parameter
                 specification. See above for details.
```

Current settings:

Parameter	Options	Current Setting
Relburnin	Yes/No	Yes
Burnin	<number>	0
Burninfrac	<number>	0.25
Filename	<name>	temp.p
Parameter	<name>	lnL
Match	Perfect/Consistentwith/All	Perfect

Prset

This command sets the priors for the phylogenetic model. Remember that in a Bayesian analysis, you must specify a prior probability distribution for the parameters of the likelihood model. The prior distribution represents your prior beliefs about the parameter before observation of the

data. This command allows you to tailor your prior assumptions to a large extent.

Options:

Applyto -- This option allows you to apply the prset commands to specific partitions. This command should be the first in the list of commands specified in prset. Moreover, it only makes sense to be using this command if the data have been partitioned. A default partition is set on execution of a matrix. If the data are homogeneous (i.e., all of the same data type), then this partition will not subdivide the characters. Up to 30 other partitions can be defined, and you can switch among them using "set partition=<partition name>". Now, you may want to specify different priors to different partitions of the data. Applyto allows you to do this. For example, say you have partitioned the data by codon position, and you want to fix the statefreqs to equal for the first two partitions but apply a flat Dirichlet prior to the statefreqs of the last. This could be implemented in two uses of prset:

```
prset applyto=(1,2) statefreqs=fixed(equal)

prset applyto=(3) statefreqs=dirichlet(1,1,1,1)
```

The first applies the parameters after "applyto" to the first and second partitions. The second prset applies a flat Dirichlet to the third partition. You can also use applyto=(all), which attempts to apply the parameter settings to all of the data partitions. Importantly, if the option is not consistent with the data in the partition, the program will not apply the prset option to that partition.

Tratiopr -- This parameter sets the prior for the transition/transversion rate ratio (tratio). The options are:

```
prset tratiopr = beta(<number>, <number>)
prset tratiopr = fixed(<number>)
```

The program assumes that the transition and transversion rates are independent gamma-distributed random variables with the same scale parameter when beta is selected. If you want a diffuse prior that puts equal emphasis on transition/transversion rate ratios above 1.0 and below 1.0, then use a flat Beta, beta(1,1), which is the default. If you wish to concentrate this distribution more in the equal-rates region,

then use a prior of the type beta(x,x), where the magnitude of x determines how much the prior is concentrated in the equal rates region. For instance, a beta(20,20) puts more probability on rate ratios close to 1.0 than a beta(1,1). If you think it is likely that the transition/transversion rate ratio is 2.0, you can use a prior of the type beta(2x,x), where x determines how strongly the prior is concentrated on tratio values near 2.0. For instance, a beta(2,1) is much more diffuse than a beta(80,40) but both have the expected tratio 2.0 in the absence of data. The parameters of the Beta can be interpreted as counts: if you have observed x transitions and y transversions, then a beta(x+1,y+1) is a good representation of this information. The fixed option allows you to fix the tratio to a particular value.

Revmatpr -- This parameter sets the prior for the substitution rates of the GTR model for nucleotide data. The options are:

```
prset revmatpr = dirichlet(<number>,<number>,...,<number>)
prset revmatpr = fixed(<number>,<number>,...,<number>)
```

The program assumes that the six substitution rates are independent gamma-distributed random variables with the same scale parameter when dirichlet is selected. The six numbers in brackets each corresponds to a particular substitution type. Together, they determine the shape of the prior. The six rates are in the order A<->C, A<->G, A<->T, C<->G, C<->T, and G<->T. If you want an uninformative prior you can use dirichlet(1,1,1,1,1,1), also referred to as a 'flat' Dirichlet. This is the default setting. If you wish a prior where the C<->T rate is 5 times and the A<->G rate 2 times higher, on average, than the transversion rates, which are all the same, then you should use a prior of the form dirichlet(x,2x,x,x,5x,x), where x determines how much the prior is focused on these particular rates. For more info, see tratiopr. The fixed option allows you to fix the substitution rates to particular values.

Revratepr -- This parameter sets the prior for each substitution rate of the GTR model subspace when 'nst' is set to 'mixed' (see the 'lset' command). The only option is

```
prset revratepr = symdir(<number>)
```

which will associate each independent rate in the rate matrix with a modified symmetric Dirichlet prior, where a singleton rate has the specified alpha parameter, while a rate that applies to n pairwise substitution types has an alpha that is n times the specified number. The higher the specified number, the more focused the prior will be on equal rates. The

default value is 1, which gives an effect similar to a flat Dirichlet.

Aamodelpr -- This parameter sets the rate matrix for amino acid data. You can either fix the model by specifying aamodelpr= fixed(<model name>), where <model name> is 'poisson' (a glorified Jukes-Cantor model), 'jones', 'dayhoff', 'mtrev', 'mtmam', 'wag', 'rtrev', 'cprev', 'vt', 'blosum', 'equalin' (a glorified Felsenstein 1981 model), or 'gtr'. You can also average over the first ten models by specifying aamodelpr= mixed. If you do so, the Markov chain will sample each model according to its probability. The sampled model is reported as an index: poisson(0), jones(1), dayhoff(2), mtrev(3), mtmam(4), wag(5), rtrev(6), cprev(7), vt(8), or blosum(9). The 'Sump' command summarizes the MCMC samples and calculates the posterior probability estimate for each of these models.

Aarevmatpr -- This parameter sets the prior for the substitution rates of the GTR model for amino acid data. The options are:

```
prset revmatpr = dirichlet(<number>,<number>,...,<number>)
prset revmatpr = fixed(<number>,<number>,...,<number>)
```

The options are the same as those for 'Revmatpr' except that they are defined over the 190 rates of the time-reversible GTR model for amino acids instead of over the 6 rates of the GTR model for nucleotides. The rates are in the order A<->R, A<->N, etc to Y<->V. In other words, amino acids are listed in alphabetic order based on their full name. The first amino acid (Alanine) is then combined in turn with all amino acids following it in the list, starting with amino acid 2 (Arginine) and finishing with amino acid 20 (Valine). The second amino acid (Arginine) is then combined in turn with all amino acids following it, starting with amino acid 3 (Asparagine) and finishing with amino acid 20 (Valine), and so on.

Omegapr -- This parameter specifies the prior on the nonsynonymous/ synonymous rate ratio. The options are:

```
prset omegapr = uniform(<number>,<number>)
prset omegapr = exponential(<number>)
prset omegapr = fixed(<number>)
```

This parameter is only in effect if the nucleotide substitution model is set to codon using the lset command (lset nucmodel=codon). Moreover, it only applies to the case when there is no variation in omega across sites (i.e., "lset omegavar=equal").

Ny98omega1pr -- This parameter specifies the prior on the nonsynonymous/ synonymous rate ratio for sites under purifying selection. The options are:

```
                    prset Ny98omega1pr = beta(<number>,<number>)
                    prset Ny98omega1pr = fixed(<number>)

                  This parameter is only in effect if the nucleotide sub-
                  stitution model is set to codon using the lset command
                  (lset nucmodel=codon). Moreover, it only applies to the
                  case where omega varies across sites using the model of
                  Nielsen and Yang (1998) (i.e., "lset omegavar=ny98"). If
                  fixing the parameter, you must specify a number between
                  0 and 1.
Ny98omega3pr  -- This parameter specifies the prior on the nonsynonymous/
                  synonymous rate ratio for positively selected sites. The
                  options are:

                    prset Ny98omega3pr = uniform(<number>,<number>)
                    prset Ny98omega3pr = exponential(<number>)
                    prset Ny98omega3pr = fixed(<number>)

                  This parameter is only in effect if the nucleotide sub-
                  stitution model is set to codon using the lset command
                  (lset nucmodel=codon). Moreover, it only applies to the
                  case where omega varies across sites according to the
                  NY98 model. Note that if the NY98 model is specified
                  that this parameter must be greater than 1, so you should
                  not specify a uniform(0,10) prior, for example.
M3omegapr     -- This parameter specifies the prior on the nonsynonymous/
                  synonymous rate ratios for all three classes of sites for
                  the M3 model. The options are:

                    prset M3omegapr = exponential
                    prset M3omegapr = fixed(<number>,<number>,<number>)

                  This parameter is only in effect if the nucleotide sub-
                  stitution model is set to codon using the lset command
                  (lset nucmodel=codon). Moreover, it only applies to the
                  case where omega varies across sites using the M3 model of
                  Yang et al. (2000) (i.e., "lset omegavar=M3"). Under the
                  exponential prior, the four rates (dN1, dN2, dN3, and dS)
                  are all considered to be independent draws from the same
                  exponential distribution (the parameter of the exponential
                  does not matter, and so you don't need to specify it). The
                  rates dN1, dN2, and dN3 are taken to be the order statistics
                  with dN1 < dN2 < dN3. These three rates are all scaled to
                  the same synonymous rate, dS. The other option is to simply
                  fix the three rate ratios to some values.
Codoncatfreqs -- This parameter specifies the prior on frequencies of sites
                  under purifying, neutral, and positive selection. The
```

```
                  options are:

                     prset codoncatfreqs = dirichlet(<num>,<num>,<num>)
                     prset codoncatfreqs = fixed(<number>,<number>,<number>)

                  This parameter is only in effect if the nucleotide sub-
                  stitution model is set to codon using the lset command
                  (lset nucmodel=codon). Moreover, it only applies to the
                  case where omega varies across sites using the models of
                  Nielsen and Yang (1998) (i.e., "lset omegavar=ny98")
                  or Yang et al. (2000) (i.e., "lset omegavar=M3")
                  Note that the sum of the three frequencies must be 1.
Statefreqpr    -- This parameter specifies the prior on the state freq-
                  uencies. The options are:

                     prset statefreqpr = dirichlet(<number>)
                     prset statefreqpr = dirichlet(<number>,...,<number>)
                     prset statefreqpr = fixed(equal)
                     prset statefreqpr = fixed(empirical)
                     prset statefreqpr = fixed(<number>,...,<number>)

                  For the dirichlet, you can specify either a single number
                  or as many numbers as there are states. If you specify a
                  single number, then the prior has all states equally
                  probable with a variance related to the single parameter
                  passed in.
Shapepr        -- This parameter specifies the prior for the gamma shape
                  parameter for among-site rate variation. The options are:

                     prset shapepr = uniform(<number>,<number>)
                     prset shapepr = exponential(<number>)
                     prset shapepr = fixed(<number>)

Pinvarpr       -- This parameter specifies the prior for the proportion of
                  invariable sites. The options are:

                     prset pinvarpr = uniform(<number>,<number>)
                     prset pinvarpr = fixed(<number>)

                  Note that the valid range for the parameter is between 0
                  and 1. Hence, "prset pinvarpr=uniform(0,0.8)" is valid
                  while "prset pinvarpr=uniform(0,10)" is not. The def-
                  ault setting is "prset pinvarpr=uniform(0,1)".
Ratecorrpr     -- This parameter specifies the prior for the autocorrelation
                  parameter of the autocorrelated gamma distribution for
                  among-site rate variation. The options are:

                     prset ratecorrpr = uniform(<number>,<number>)
```

```
                    prset ratecorrpr = fixed(<number>)

                 Note that the valid range for the parameter is between -1
                 and 1. Hence, "prset ratecorrpr=uniform(-1,1)" is valid
                 while "prset ratecorrpr=uniform(-11,10)" is not. The
                 default setting is "prset ratecorrpr=uniform(-1,1)".
Covswitchpr   -- This option sets the prior for the covarion switching
                 rates. The options are:

                    prset covswitchpr = uniform(<number>,<number>)
                    prset covswitchpr = exponential(<number>)
                    prset covswitchpr = fixed(<number>,<number>)

                 The covarion model has two rates: a rate from on to off
                 and a rate from off to on. The rates are assumed to have
                 independent priors that individually are either uniformly
                 or exponentially distributed. The other option is to
                 fix the switching rates, in which case you must specify
                 both rates. (The first number is off->on and the second
                 is on->off).
Symdirihyperpr - This option sets the prior for the stationary frequencies
                 of the states for morphological (standard) data. There can
                 be as many as 10 states for standard data. However, the
                 labelling of the states is somewhat arbitrary. For example,
                 the state "1" for different characters does not have the
                 same meaning. This is not true for DNA characters, for ex-
                 ample, where a "G" has the same meaning across characters.
                 The fact that the labelling of morphological characters is
                 arbitrary makes it difficult to allow unequal character-
                 state frequencies. MrBayes gets around this problem by
                 assuming that the states have a dirichlet prior, with all
                 states having equal frequency. The variation in the diri-
                 chlet can be controlled by this parameter--symdirihyperpr.
                 Symdirihyperpr specifies the distribution on the variance
                 parameter of the dirichlet. The valid options are:

                    prset Symdirihyperpr = uniform(<number>,<number>)
                    prset Symdirihyperpr = exponential(<number>)
                    prset Symdirihyperpr = fixed(<number>)
                    prset Symdirihyperpr = fixed(infinity)

                 If "fixed(infinity)" is chosen, the dirichlet prior is
                 fixed such that all character states have equal frequency.
Topologypr    -- This parameter specifies the prior probabilities of
                 phylogenies. The options are:

                    prset topologypr = uniform
                    prset topologypr = speciestree
```

```
                   prset topologypr = constraints(<list>)
                   prset topologypr = fixed(<treename>)

                If the prior is selected to be "uniform", the default,
                then all possible trees are considered a priori equally
                probable. The 'speciestree' option is used when the topology
                is constrained to fold inside a species tree together with
                other (gene) trees. The constraints option allows you to
                specify complicated prior probabilities on trees (constraints
                are discussed more fully in "help constraint"). Note that
                you must specify a list of constraints that you wish to be
                obeyed. The list can be either the constraints' name or
                number. Finally, you can fix the topology to that of a user
                tree defined in a trees block. Branch lengths will still be
                sampled as usual on the fixed topology.
Brlenspr      -- This parameter specifies the prior probability dist-
                ribution on branch lengths. The options are specified using:

                   prset brlenspr = <setting>

                where <setting> is one of

                   unconstrained:uniform(<num>,<num>)
                   unconstrained:exponential(<number>)
                   unconstrained:twoexp(<num>,<num>)
                   unconstrained:gammadir(<num>,<num>,<num>,<num>)
                   unconstrained:invgamdir(<num>,<num>,<num>,<num>)
                   clock:uniform
                   clock:birthdeath
                   clock:coalescence
                   clock:speciestreecoalescence
                   fixed(<treename>)

                Trees with unconstrained branch lengths are unrooted
                whereas clock-constrained trees are rooted. The option
                after the colon specifies the details of the probability
                density of branch lengths. If you choose a birth-death
                or coalescence prior, you may want to modify the details
                of the parameters of those processes (speciation rate,
                extinction rate and sample probability for the birth-death
                prior; population size and clock rate parameter for the
                coalescence prior). When gene trees are constrained to fold
                inside species trees, the appropriate branch length prior is
                'clock:speciestreecoalescence'. Under this model, it is poss-
                ible to control whether the population size is constant or
                varies across the species tree using the 'popvarpr' setting.
                Branch lengths can also be fixed but only if the topology is
                fixed.
```

For unconstrained branch lengths, MrBayes offers five alternative prior distributions. The first two are the simple 'uniform' and 'exponential' priors. The 'uniform' prior takes two parameters, the lower and upper bound of the uniform distribution, respectively. The 'exponential' prior takes a single parameter, the rate of the exponential distribution. The mean of the exponential distribution is the inverse of the rate. For instance, an 'exp(10)' distribution has an expected mean of 0.1.

MrBayes also offers three more complex prior distributions on unconstrained branch lengths. The two-exponential prior (Yang and Rannala 2005; Yang 2007) uses two different exponential distributions, one for internal and one for external branch lengths. The two-exponential prior is invoked using 'twoexp(<r_I>,<r_E>)', where '<r_I>' is a number specifying the rate of the exponential distribution on internal branch lengths, while '<r_E>' is the rate for external branch lengths. The expected prior mean for internal branch lengths is then 1/r_I, and for external ones it is 1/r_E. For instance, to set r_I = 100 and r_E = 10, use 'twoexp(100,10)'. The setting 'twoexp(10,10)' is equivalent to 'exp(10)'.

The compound Dirichlet priors 'gammadir(<a_T>,<b_T>,<a>,<c>)' and 'invgamdir(<a_T>,<b_T>,<a>,<c>)' specify a fairly diffuse prior on tree length 'T', and then partitions the tree length into branch lengths according to a Dirichlet distribution (Rannala et al. 2012). If 'T' is considered drawn from a gamma distribution with parameters a_T and b_T, and with mean a_T/b_T, we recommend setting a_T = 1; if it is instead considered drawn from an inverse gamma (invgamma) distribution with parameters a_T and b_T, and with mean b_T/(a_T -1), then we reccommend setting a_T = 3. In the latter case, b_T should be chosen so that the prior mean of T is reasonable for the data. In the former case, setting b_T = 0.1 (corresponding to a mean tree length of 10) should be appropriate for a wide range of tree lengths (at least in the interval 1 to 100). The concentration parameter a of the Dirichlet distribution is inversely related to the variance of the branch lengths, while c is the ratio of the prior means for the internal and external branch lengths. The default setting, a = c = 1, specifies a uniform Dirichlet distribution of branch lengths given the tree length. For instance, 'gammadir(1,0.1,1,1)' specifies a compound Dirichlet prior on branch lengths, where tree length is associated with a gamma distribution with mean 10, and branch length proportions are associated with a uniform Dirichlet distribution. The default prior for uncon-

```
                strained branch lengths is 'exp(10)'.
Treeagepr     -- This parameter specifies the prior probability distribution
                on the tree age when a uniform prior is used on the branch
                lengths of a clock tree.

                The options are:

                   prset treeagepr = <setting>

                where <setting> is one of

                   fixed(<age>)
                   uniform(<min_age>,<max_age>)
                   offsetexponential(<min_age>,<mean_age>)
                   truncatednormal(<min_age>,<mean_age>,<st.dev.>)
                   lognormal(<mean_age>,<st.dev.>)
                   offsetlognormal(<min_age>,<mean_age>,<st.dev.>)
                   gamma(<mean_age>,<st.dev.>)
                   offsetgamma(<min_age>,<mean_age>,<st.dev.>)

                These are the same options used for the 'Calibrate' command.
                Note that, unlike elsewhere in MrMayes, we always use the
                mean and standard deviation of the resulting age distribution
                rather than the standard parameterization, if different. This
                is to facilitate for the users who want to focus on the in-
                formation conveyed about the age. For those who wish to use
                the standard parameterization, there are simple conversions
                between the two. See the 'Calibrate' command for more infor-
                mation.

                The tree age is simply the age of the most recent common
                ancestor of the tree. If the clock rate is fixed to 1.0,
                which is the default, the tree age is equivalent to the
                expected number of substitutions from the root to the tip of
                the tree, that is, tree height. The tree age prior ensures
                that the joint probability for the uniform prior (or fossil-
                ization prior) model of branch lengths on a clock tree is
                proper. The default setting is 'gamma(1,1)'. If the root node
                in the tree is calibrated, the root calibration replaces the
                tree age prior.
Speciationpr  -- This parameter sets the prior on the net speciation rate,
                that is, lambda - mu in the birth-death model;
                or, lambda - mu - psi in the fossilized birth-death model.

                   prset speciationpr = uniform(<number>,<number>)
                   prset speciationpr = exponential(<number>)
                   prset speciationpr = fixed(<number>)
```

This parameter is only relevant if the birth-death process is selected as the prior on branch lengths.

Extinctionpr -- This parameter sets the prior on the relative extinction rate, that is, mu / lambda in the birth-death model; or, (mu + psi) / lambda in the fossilization model. Values of this parameter are in the range (0,1). Prior options are:

```
prset extinctionpr = beta(<number>,<number>)
prset extinctionpr = fixed(<number>)
```

This parameter is only relevant if the birth-death process is selected as the prior on branch lengths.

SampleStrat -- This parameter sets the strategy under which species where sampled in the analysis. This is used with the birth-death prior on trees (see H√∂hna et al, 2011).

```
prset samplestrat = random
prset samplestrat = diversity
prset samplestrat = cluster
```

Sampleprob -- This parameter sets the fraction of extant species that are sampled in the analysis. This is used with the birth-death prior on trees (see Yang and Rannala, 1997),

```
prset sampleprob = <number>
```

Popsizepr -- This parameter sets the prior on the population size component of the coalescent parameter. The options are:

```
prset popsizepr = uniform(<number>,<number>)
prset popsizepr = lognormal(<number>,<number>)
prset popsizepr = normal(<number>,<number>)
prset popsizepr = gamma(<number>,<number>)
prset popsizepr = fixed(<number>)
```

This parameter is only relevant if the coalescence process is selected as the prior on branch lengths. Note that the setting of 'ploidy' in 'lset' is important for how this parameter is interpreted.

Popvarpr -- In a gene tree - species tree model, this parameter determines whether the population size is the same for the entire species tree ('popvarpr = equal', the default), or varies across branches of the species tree ('popvarpr=variable').

Nodeagepr -- This parameter specifies the assumptions concerning the age of the terminal and interior nodes in the tree. The default model ('nodeagepr = unconstrained') assumes that all terminal nodes are of the same age while the age of interior nodes is unconstrained. The alternative ('nodeagepr = calibrated')

option derives a prior probability distribution on terminal and interior node ages from the calibration settings (see the 'calibrate' command). The 'nodeagepr' parameter is only relevant for clock trees.

Clockratepr -- This parameter specifies the prior assumptions concerning the base substitution rate of the tree, measured in expected number of substitutions per site per time unit. The default setting is 'Fixed(1.0)', which effectively means that the time unit is the number of expected substitutions per site. If you apply age constraints to the tree, the default setting changes automatically to 'Exponential(<x>)', where '<x>' is set such that the expectation of the exponential is ten times the age of the maximum age constraint. This will give you a very vague prior, which may or may not be adequate for your particular problem.

If you do not have any age calibrations in the tree, you can still calibrate the tree using 'Clockratepr'. For instance, if you know that your sequence data evolve at a rate of 0.20 substitutions per million years, you might calibrate the tree by fixing the substitution rate to 0.20 using

```
prset clockratepr = fixed(0.20)
```

after which the tree will be calibrated using millions of years as the unit.

You can also assign a prior probability distribution to the substitution rate, accommodating the uncertainty about its value. You can choose between normal, lognormal, exponential and gamma distributions for this purpose. For instance, if you would like to associate the substitution rate with a normal distribution truncated at 0, so that only positive values are allowed, and with mean 0.20 and standard deviation of 0.02, you would use

```
prset clockratepr = normal(0.20,0.02)
```

The lognormal distribution is parameterized in terms of the mean and standard deviation on the log scale (natural logs). For instance,

```
prset clockratepr = lognormal(-1.61,0.10)
```

specifies a lognormal distribution with a mean of log values of -1.61 and a standard deviation of log values of 0.10. In such a case, the mean value of the lognormal distribution is equal to e^(-1.61 + 0.10^2/2) = 0.20.

Note that the 'Clockratepr' parameter has no effect on non-clock trees.

Clockvarpr -- This parameter allows you to specify the type of clock you are assuming. The default is 'strict', which corresponds to the standard clock model where the evolutionary rate is constant throughout the tree. You can also use 'cpp', which invokes a relaxed clock model where the rate evolves according to a Compound Poisson Process (CPP) model (see Huelsenbeck et al., 2000) or 'tk02', which invokes the Brownian Motion (TK02) model described by Thorne and Kishino (2002). Finally, you can use a model where each branch has an independent rate drawn from a scaled gamma distribution, such that there is a specified variance in the effective height of the tree in the prior, the Independent Gamma Rate (IGR) model (LePage et al., 2007). Each of the relaxed clock models has additional parameters with priors. For the CPP model, it is 'cppratepr' and 'cppmultdevpr'; for the TK02 model, it is 'tk02varpr'; for the IGR model, it is 'igrvarpr'. The 'clockvarpr' parameter is only relevant for clock trees.

For backward compatibility, 'bm' is allowed as a synonym of 'tk02', and 'ibr' as a synonym of 'igr'.

Cppratepr -- This parameter allows you to specify a prior probability distribution on the rate of the Poisson process generating changes in the evolutionary rate in the CPP relaxed clock model. You can either fix the rate or associate it with an exponential prior using

```
prset cppratepr = fixed(<number>)
prset cppratepr = exponential(<number>)
```

For instance, if you fix the rate to 2, then on a branch with the length equual to one expresed in terms of average expected number of substitution per site, you expect to see, on average, two rate-modifying events.
If you put an exponential(0.1) on the rate, you will be estimating the rate against a prior probability distribution where the expected rate is 10 (= 1/0.1).

Cppmultdevpr -- This parameter allows you to specify the standard deviation of the log-normal distribution from which the rate multipliers of the CPP relaxed clock model are drawn. The standard deviation is given on the log scale. The default value of 1.0 thus corresponds to rate multipliers varying from 0.37 (1/e) to 2.7 (e) when they are +/- one standard deviation from the expected mean. The expected mean of the logarithm of the multipliers is fixed to 0, ensuring that the expected mean rate is 1.0. You can change the default value by using

```
                    prset cppmultdevpr = fixed(<number>)

                 where <number> is the standard deviation on the log scale.
TK02varpr     -- This parameter allows you to specify the prior probability
                 distribution for the variance of the rate multiplier in the
                 Thorne-Kishino ('Brownian motion') relaxed clock model.
                 Specifically, the parameter specifies the rate at which the
                 variance increases with respect to the base rate of the
                 clock. If you have a branch of a length corresponding to 0.4
                 expected changes per site according to the base rate of the
                 clock, and the tk02var parameter has a value of 2.0, then the
                 rate multiplier at the end of the branch will be drawn from a
                 lognormal distribution with a variance of 0.4*2.0 (on the
                 linear, not the logarithm scale). The mean is the same as the
                 rate multiplier at the start of the branch (again on the
                 linear scale).

                 You can set the parameter to a fixed value, or specify that
                 it is drawn from an exponential or uniform distribution:

                    prset tk02varpr = fixed(<number>)
                    prset tk02varpr = exponential(<number>)
                    prset tk02varpr = uniform(<number>,<number>)

                 For backward compatibility, 'bmvarpr' is allowed as a synonym
                 of 'tko2varpr'.
Igrvarpr      -- This parameter allows you to specify a prior on the variance
                 of the gamma distribution from which the branch lengths are
                 drawn in the independent branch rate (IGR) relaxed clock
                 model. Specifically, the parameter specifies the rate at
                 which the variance increases with respect to the base rate of
                 the clock. If you have a branch of a length corresponding to
                 0.4 expected changes per site according to the base rate of
                 the clock, and the igrvar parameter has a value of 2.0, then
                 the effective branch length will be drawn from a distribution
                 with a variance of 0.4*2.0.

                 You can set the parameter to a fixed value, or specify that
                 it is drawn from an exponential or uniform distribution:

                    prset igrvarpr = fixed(<number>)
                    prset igrvarpr = exponential(<number>)
                    prset igrvarpr = uniform(<number>,<number>)

                 For backward compatibility, 'ibrvarpr' is allowed as a syn-
                 onym of 'igrvarpr'.
Ratepr        -- This parameter allows you to specify the site specific rates
```

```
                model or any other model that allows different partitions to
                evolve at different rates. First, you must have defined a
                partition of the characters. For example, you may define a
                partition that divides the characters by codon position, if
                you have DNA data. You can also divide your data using a
                partition that separates different genes from each other.
                The next step is to make the desired partition the active one
                using the set command. For example, if your partition is
                called "by_codon", then you make that the active partition
                using "set partition=by_codon". Now that you have defined
                and activated a partition, you can specify the rate multi-
                pliers for the various partitions. The options are:

                   prset ratepr = fixed
                   prset ratepr = variable
                   prset ratepr = dirichlet(<number>,<number>,...,<number>)

                If you specify "fixed", then the rate multiplier for
                that partition is set to 1 (i.e., the rate is fixed to
                the average rate across partitions). On the other hand,
                if you specify "variable", then the rate is allowed to
                vary across partitions subject to the constraint that the
                average rate of substitution across the partitions is 1.
                You must specify a variable rate prior for at least two
                partitions, otherwise the option is not activated when
                calculating likelihoods. The variable option automatically
                associates the partition rates with a dirichlet(1,...,1)
                prior. The dirichlet option is an alternative way of setting
                a partition rate to be variable, and also gives accurate
                control of the shape of the prior. The parameters of the
                Dirichlet are listed in the order of the partitions that the
                ratepr is applied to. For instance, "prset applyto=(1,3,4)
                ratepr = dirichlet(10,40,15)" would set the Dirichlet para-
                meter 10 to partition 1, 40 to partition 3, and 15 to parti-
                tion 4. The Dirichlet distribution is applied to the weighted
                rates; that is, it weights the partition rates according to
                the number of included characters in each partition.
Generatepr    -- This parameter is similar to 'Ratepr' but applies to gene
                trees in the multispecies coalescent, whereas 'Ratepr' app-
                lies to partitions within genes.

Default model settings:

Parameter        Options                     Current Setting
-------------------------------------------------------------------
Tratiopr         Beta/Fixed                  Beta(1.0,1.0)
Revmatpr         Dirichlet/Fixed             Dirichlet(1.0,1.0,1.0,1.0,1.0,1.0)
Aamodelpr        Fixed/Mixed                 Fixed(Poisson)
```

```
Aarevmatpr       Dirichlet/Fixed              Dirichlet(1.0,1.0,...)
Omegapr          Dirichlet/Fixed              Dirichlet(1.0,1.0)
Ny98omega1pr     Beta/Fixed                   Beta(1.0,1.0)
Ny98omega3pr     Uniform/Exponential/Fixed    Exponential(1.0)
M3omegapr        Exponential/Fixed            Exponential
Codoncatfreqs    Dirichlet/Fixed              Dirichlet(1.0,1.0,1.0)
Statefreqpr      Dirichlet/Fixed              Dirichlet(1.0,1.0,1.0,1.0)
Shapepr          Uniform/Exponential/Fixed    Exponential(2.0)
Ratecorrpr       Uniform/Fixed                Uniform(-1.0,1.0)
Pinvarpr         Uniform/Fixed                Uniform(0.0,1.0)
Covswitchpr      Uniform/Exponential/Fixed    Uniform(0.0,100.0)
Symdirihyperpr   Uniform/Exponential/Fixed    Fixed(Infinity)
Topologypr       Uniform/Constraints/Fixed    Uniform
Brlenspr         Unconstrained/Clock/Fixed    Unconstrained:Exp(10.0)
Treeagepr        Gamma/Uniform/Fixed/         Gamma(1.00,1.00)
                 Truncatednormal/Lognormal/
                 Offsetlognormal/Offsetgamma/
                 Offsetexponential
Speciationpr     Uniform/Exponential/Fixed    Exponential(1.0)
Extinctionpr     Beta/Fixed                   Beta(1.0,1.0)
Fossilizationpr  Beta/Fixed                   Beta(1.0,1.0)
SampleStrat      Random/Diversity/Cluster     Random
Sampleprob       <number>                     1.00
Popsizepr        Lognormal/Gamma/Uniform/     Lognormal(4.6,2.3)
                 Normal/Fixed
Popvarpr         Equal/Variable               Equal
Nodeagepr        Unconstrained/Calibrated     Unconstrained
Clockratepr      Fixed/Normal/Lognormal/
                 Exponential/Gamma            Fixed(1.00)
Clockvarpr       Strict/Cpp/TK02/Igr          Strict
Cppratepr        Fixed/Exponential            Exponential(0.10)
Cppmultdevpr     Fixed                        Fixed(0.40)
TK02varpr        Fixed/Exponential/Uniform    Exponential(10.00)
Igrvarpr         Fixed/Exponential/Uniform    Exponential(10.00)
Ratepr           Fixed/Variable=Dirichlet     Fixed
Generatepr       Fixed/Variable=Dirichlet     Fixed
-----------------------------------------------------------------
```

Propset

This command allows the user to change the details of the MCMC samplers (moves) that update the state of the chain. The useage is:

```
   propset  <move_name>$<tuning-parameter>=<value>
```

Assume we have a topology parameter called 'Tau{all}', which is sampled by the move 'ExtTBR(Tau{all})' (note that the parameter name is included in the

move name). This move has three tuning parameters: (1) 'prob', the relative proposal probability (a weight defining its probability relative to other moves); (2) 'p_ext', the extension probability; and (3) 'lambda', the tuning parameter of the branch length multiplier. A list of the tuning parameters is available by using 'Showmoves' (see below). To change the relative proposal probability to 20 and the extension probability to 0.7, use:

```
propset etbr(tau{all})$prob=20 etbr(tau{all})$p_ext=0.7
```

This change would apply to all chains in all runs. It is also possible to set the tuning parameters of individual runs and chains using the format:

```
propset  <move_name>$<tuning-parameter>(<run>,<chain>)=<value>
```

where <run> and <chain> are the index numbers of the run and chain for which you want to change the value. If you leave out the index of the run, the change will apply to all runs; if you leave out the index of the chain, the change will similarly apply to all chains. To switch off the exttbr(tau{all}) move in chain 2 of all runs, use:

```
propset  etbr(tau{all})$prob(,2)=0
```

It is important to note that all moves are not available until the model has been completely defined. Any change to the model will cause all proposal tuning parameters to return to their default values. To see a list of all the moves that are currently switched on for the model, use 'showmoves'. You can also see other available moves by using 'showmoves allavailable=yes'. A list of the moves for each parameter in the model is available by using the command 'Showparams'. If you change proposal probabilities, make sure that all parameters that are not fixed in your model have at least one move switched on.

One word of warning: You should be extremely careful when modifying any of the chain parameters using 'propset'. It is quite possible to completely wreck any hope of achieving convergence by inappropriately setting the tuning parameters. In general, you want to set move tuning parameters such that the acceptance rate of the move is intermediate (we suggest targeting the range 10% to 70% acceptance, if possible). If the acceptance rate is outside of this range, the MCMC chain will probably not sample that parameter very efficiently. The acceptance rates for all moves in the cold chain(s) are summarized at the end of each run in the screen output. The acceptance rates (potentially for all chains, cold and heated) are also printed to the .mcmc file if Mcmc convergence diagnostics are turned on (using 'Mcmc' or 'Mcmcp').

Quit

This command quits the program. The correct usage is:

```
   quit

It is a very easy command to use properly.
------------------------------------------------------------------------
------------------------------------------------------------------------
Report

This command allows you to control how the posterior distribution is
reported. For rate parameters, it allows you to choose among several popular
parameterizations. The report command also allows you to request printing of
some model aspects that are usually not reported. For instance, if a node is
constrained in the analysis, MrBayes can print the probabilities of the
ancestral states at that node. Similarly, if there is rate variation in the
model, MrBayes can print the inferred site rates, and if there is omega varia-
tion, MrBayes can print the inferred omega (positive selection) values for
each codon. In a complex model with several partitions, each partition is
controlled separately using the same 'Applyto' mechanism as in the 'Lset' and
'Prset' commands.

Options:

Applyto   -- This option allows you to apply the report commands to specific
             partitions. This command should be the first in the list of
             commands specified in 'report'.
             For example,

                report applyto=(1,2) tratio=ratio

                report applyto=(3) tratio=dirichlet

             would result in the transition and transversion rates of the
             first and second partitions in the model being reported as a
             ratio and the transition and transversion rates of the third
             partition being reported as proportions of the rate sum (the
             Dirichlet parameterization).
Tratio    -- This specifies the report format for the transition and trans-
             version rates of a nucleotide substituion model with nst=2.
             If 'ratio' is selected, the rates will be reported as a ratio
             (transition rate/transversion rate). If 'dirichlet' is selected,
             the transition and transversion rates will instead be reported
             as proportions of the rate sum. For example, if the transition
             rate is three times the transversion rate and 'ratio' is selec-
             ted, this will reported as a single value, '3.0'. If 'dirichlet'
             is selected instead, the same rates will be reported using two
             values, '0.75 0.25'. The sum of the Dirichlet values is always 1.
             Although the Dirichlet format may be unfamiliar to some users,
             it is more convenient for specifying priors than the ratio
```

format.

Revmat -- This specifies the report format for the substitution rates of a GTR substitution model for nucleotide or amino acid data. If 'ratio' is selected, the rates will be reported scaled to the G-T rate (for nucleotides) or the Y-V rate (for amino acids). If 'dirichlet' is specified instead, the rates are reported as proportions of the rate sum. For instance, assume that the C-T rate is twice the A-G rate and four times the transversion rates, which are equal. If the report format is set to 'ratio', this would be reported as '1.0 2.0 1.0 1.0 4.0 1.0' since the rates are reported in the order rAC, rAG, rAT, rCG, rCT, rGT and scaled relative to the last rate, the G-T rate. If 'dirichlet' is selected instead, the same rates would have been reported as '0.1 0.2 0.1 0.1 0.4 0.1' since the rates are now scaled so that they sum to 1.0. The Dirichlet format is the parameterization used for formulating priors on the rates.

Ratemult -- This specifies the report format used for the rate multiplier of different model partitions. Three formats are available. If 'scaled' is selected, then rates are scaled such that the mean rate per site across partitions is 1.0. If 'ratio' is chosen, the rates are scaled relative to the rate of the first partition. Finally, if 'dirichlet' is chosen, the rates are given as proportions of the rate sum. The latter is the format used when formulating priors on the rate multiplier.

Tree -- This specifies the report format used for the tree(s). Two options are available. 'Topology' results in only the topology being printed to file, whereas 'brlens' causes branch lengths to to be printed as well.

Ancstates -- If this option is set to 'yes', MrBayes will print the probability of the ancestral states at all constrained nodes. Typically, you are interested in the ancestral states of only a few characters and only at one node in the tree. To perform such an analysis, first define and enforce a topology constraint using 'constraint' and 'prset topologypr = constraints (...)'. Then put the character(s) of interest in a separate partition and set MrBayes to report the ancestral states for that partition. For instance, if the characters of interest are in partition 2, use 'report applyto=(2) ancstates=yes' to force MrBayes to print the probability of the ancestral states of those characters at the constrained node to the '.p' file.

Siterates -- If this option is set to 'yes' and the relevant model has rate variation across sites, then the site rates, weighted over rate categories, will be reported to the '.p' file.

Possel -- If this option is set to 'yes' and the relevant model has omega variation across sites, the probability that each model site (codon in this case) is positively selected will be written to file.

Siteomega -- If this option is set to 'yes' and the relevant model has omega

variation across sites, the weighted omega value (over omega categories) for each model site will be reported to file.

Default report settings:

Parameter	Options	Current Setting
Tratio	Ratio/Dirichlet	Ratio
Revmat	Ratio/Dirichlet	Dirichlet
Ratemult	Scaled/Ratio/Dirichlet	Scaled
Tree	Brlens/Topology	Brlens
Ancstates	Yes/No	No
Siterates	Yes/No	No
Possel	Yes/No	No
Siteomega	Yes/No	No

Restore

This command restores taxa to the analysis. The correct usage is:

   restore <name and/or number and/or taxset> ...

A list of the taxon names or taxon numbers (labelled 1 to ntax in the order in the matrix) or taxset(s) can be used. For example, the following:

   restore 1 2 Homo_sapiens

restores taxa 1, 2, and the taxon labelled Homo_sapiens to the analysis. You can also use "all" to restore all of the taxa. For example,

   restore all

restores all of the taxa to the analysis.

Set

This command is used to set some general features of the model or program behavior. The correct usage is

   set <parameter>=<value> ... <parameter>=<value>

Available options:

Seed   -- Sets the seed number for the random number generator. The

```
                random number seed is initialized haphazardly at the beg-
                inning of each MrBayes session. This option allows you to
                set the seed to some specific value, thereby allowing you
                to exactly repeat an analysis. If the analysis uses swapping
                between cold and heated chains, you must also set the swap
                seed (see below) to exactly repeat the analysis.
Swapseed     -- Sets the seed used for generating the swapping sequence
                when Metropolis-coupled heated chains are used. This seed
                is initialized haphazardly at the beginning of each MrBayes
                session. This option allows you to set the seed to some
                specific value, thereby allowing you to exactly repeat a
                swap sequence. See also the 'Seed' option.
Dir          -- The working directory. Specifies the absolute or relative path
                to the working directory. If left empty, the working directory
                is the current directory.
Partition    -- Set this option to a valid partition id, either the number or
                name of a defined partition, to enforce a specific partition-
                ing of the data. When a data matrix is read in, a partition
                called "Default" is automatically created. It divides the
                data into one part for each data type. If you only have one
                data type, DNA for instance, the default partition will not
                divide up the data at all. The default partition is always
                the first partition, so 'set partition=1' is the same as
                'set partition=default'.
Speciespartition -- Set this option to a valid speciespartition id, either the
                number or name of a defined speciespartition, to enforce a
                specific partitioning of taxa to species. When a data matrix
                is read in, a speciespartition called "Default" is auto-
                matically created. It assigns one taxon for each species. The
                default speciespartition is always the first speciespartition,
                so 'set speciespartition=1' is the same as
                'set speciespartition=default'.
Autoclose    -- If autoclose is set to 'yes', then the program will not prompt
                you during the course of executing a file. This is particular-
                ly useful when you run MrBayes in batch mode.
Nowarnings   -- If nowarnings is set to yes, then the program will not prompt
                you when overwriting or appending an ouput file that is al-
                ready present. If 'nowarnings=no' (the default setting), then
                the program propts the user before overwriting output files.
Autoreplace  -- When nowarnings is set to yes, then MrBayes will by default
                overwrite output files that already exists. This may cause
                irrecoverable loss of previous results if you have not removed
                or renamed the files from previous runs. To override this be-
                havior, set autooverwrite to no, in which case new output will
                be appended to existing files instead.
Quitonerror  -- If quitonerror is set to yes, then the program will quit when
                an error is encountered, after printing an error message. If
                quitonerror is set to no (the default setting), then the
```

```
                program will wait for additional commands from the command
                line after the error message is printed.
Scientific   -- Set this option to 'Yes' to write sampled values to file in
                scientific format and to 'No' to write them in fixed format.
                Fixed format is easier for humans to read but you risk losing
                precision for small numbers. For instance, sampled values that
                are less than 1E-6 will print to file as '0.000000' if fixed
                format is used and 'precision' is set to 6.
Precision    -- Precision allows you to set the number of decimals to be prin-
                ted when sampled values are written to file. Precision must be
                in the range 3 to 15.
Usebeagle    -- Set this option to 'Yes' to attempt to use the BEAGLE library
                to compute the phylogenetic likelihood on a variety of high-
                performance hardware including multicore CPUs and GPUs. Some
                models in MrBayes are not yet supported by BEAGLE.
Beagledevice -- Set this option to 'GPU' or 'CPU' to select processor.
Beagleprecision -- Selection 'Single' or 'Double' precision computation.
Beaglescaling --  'Always' rescales partial likelihoods at each evaluation.
                  'Dynamic' rescales less frequently and should run faster.
Beaglesse    -- Use SSE instructions on Intel CPU processors.
Beagleopenmp -- Use OpenMP to parallelize across multi-core CPU processors.

Current settings:

Parameter          Options               Current Setting
-------------------------------------------------------
Seed               <number>              1376406472
Swapseed           <number>              1376406472
Dir                <name>                ""
Partition          <name>                ""
Speciespartition   <name>                ""
Autoclose          Yes/No                No
Nowarnings         Yes/No                No
Autoreplace        Yes/No                Yes
Quitonerror        Yes/No                No
Sientific          Yes/No                Yes
Precision          <number>              6
Usebeagle          Yes/No                No
Beagledevice       CPU/GPU               CPU
Beagleprecision    Single/Double         Double
Beaglescaling      Always/Dynamic        Always
Beaglesse          Yes/No                No
Beagleopenmp       Yes/No                No

------------------------------------------------------------------------
------------------------------------------------------------------------
Showbeagle
```

This command shows available BEAGLE resources.

Showmatrix

This command shows the character matrix currently in memory.

Showmcmctrees

This command shows the current trees used by the Markov chains.
is "showmcmctrees".

Showmodel

This command shows the current model settings. The correct usage is

   showmodel

After typing "showmodel", the modelling assumptions are shown on a
partition-by-partition basis.

Showmoves

This command shows the MCMC samplers (moves) that are switched on for the
parameters in the current model. The basic usage is

   showmoves

If you want to see all available moves, use

   showmoves allavailable=yes

If you want to change any of the tuning parameters for the moves, use the
'propset' command.

Showparams

This command shows all of the parameters in the current model. The basic
usage is

   showparams

The parameters are listed together with their priors, the available moves,
and the current value(s), which will be used as the starting values in the

next mcmc analysis.

Showusertrees

This command shows the currently defined user trees. The correct usage
is "showusertrees".

Speciespartition

Defines a partition of tips into species. The format for the speciespartition
command is

   Speciespartition <name> = <species name>:<taxon list> ,...,<sp nm>:<tx lst>

The command enumerates comma separated list of pairs consisting of 'species
name' and 'taxon list'. The 'taxon list' is a standard taxon list, as used by
the 'Taxset' command. This means that you can use either the index or the name
of a sequence ('taxon'). Ranges are specified using a dash, and a period can
be used as a synonym of the last sequence in the matrix.

For exammple: speciespartition species = SpeciesA: 1, SpeciesB: 2-.
Here, we name two species. SpeciesA is represented by a single sequence while
SpeciesB is represented by all remaining sequences in the matrix.
Each sequence is specified by its row index in the data matrix.

As with ordinary partitioning you may define multiple species partitioning
scheme. You have to use command 'set speciespartition' to enable use of one of
them.

Currently defined Speciespartitions:

Number Speciespartition name   Number of species

Ss

This command is used to start stepping-stone sampling, which is an efficient
and accurate method for estimating the marginal likelihood of the currently
specified model. It is considerably more accurate than the harmonic mean of
the likelihoods from a standard MCMC run on the model (calculated by the
'Sump' command) but it requires a separate MCMC-like run. To be more specific,
stepping-stone sampling uses importance sampling to estimate each ratio in a
series of discrete steps bridging the posterior and prior distributions.
The importance distributions that are used are called power posterior distri-

butions, and are defined as prior*(likelihood^beta). By varying beta from 1 to 0, we get a series of distributions that connect the posterior (beta = 1) to the prior (beta = 0).

The power posterior distributions are sampled using MCMC. First, we start a standard MCMC chain on the posterior distribution, and let it run until we have reached the criterion specified by the 'Burninss' option. After this, we step through the power posterior distributions until we reach the prior distribution. In each of the 'Nsteps' steps, we sample from a new power posterior distribution with a distinct beta value. The beta values correspond to 'Nsteps' evenly spaced quantiles in a Beta distribution with the parameters 'Alpha' and 1.0. For the first sampling step, the beta value is equal to the last quantile, i.e., it is close to 1.0. For each successive step, the beta value takes on the value of the next quantile, in decreasing order, until it reaches the value of 0.0. If you change value of 'FromPrior' from default 'No' to 'Yes' then the direction of power posterior change during SS analizes is opposite to the one described above, i.e. we start from sampling prior and finish close to posterior.

The 'Ss' procedure uses the same machinery as the standard 'Mcmc' algorithm, and shares most of its parameters with the 'Mcmc' and 'Mcmcp' commands. All 'Mcmc' parameters, except those related to burnin, have the same meaning and usage in the 'Ss' command as they have in the 'Mcmc' command. The 'Mcmc' burnin parameters are used to set up burnin within each step. The 'Ss' command also uses its own burnin parameter, 'Burninss' (see below for details). The 'Ss' command also has its own parameters for specifying the number of steps and the shape of the Beta distribution from which the beta values are computed (see below).

Note that the 'Ngen' parameter of 'Mcmc' is used to set the maximum number of generations processed, including both the burnin and the following steps in the stepping-stone sampling phase. For instance, assume that 'Burninss' is set to '-1', 'Nsteps' to '49', 'Ngen' to '1000000' and 'Samplefreq' to '1000'. We will then get 1,000 samples in total (1,000,000 / 1,000). These will fall into 50 bins, one of which represents the burnin and is discarded. Each step in the algorithm will thus be represented by 20 samples.

More information on 'Mcmc' parameters is available in the help for the 'Mcmc' and 'Mcmcp' commands. Only the exclusive 'Ss' parameters are listed below. These can only be set up using the 'Ss' command, while the parameters shared with 'Mcmc' and 'Mcmcp' can also be set up using those commands.

The correct usage is

```
   ss <parameter>=<value> ... <parameter>=<value>
```

Note that a command:

```
   ss <setting parameters shared with mcmc> <setting exclusive ss parameters>

would be equivalent to executing two commands:

  mcmcp <setting parameters shared with mcmc>;
  ss <setting exclusive ss parameters>;

For more information on the stepping-stone algorithm, see:

Xie, W., P. O. Lewis, Y. Fan, L. Kuo, and M.-H. Chen. 2011. Improving marginal
   likelihood estimation for Bayesian phylogenetic model selection. Systematic
   Biology 60:150-160.

Available options:
(NB: Only exclusive ss parameters listed here. For additional parameters, see
     help on 'mcmc' or 'mcmcp'.

Alpha        -- The beta values used in the stepping-stone sampling procedure
                correspond to evenly spaced quantiles from a Beta('Alpha',1.0)
                distribution. The parameter 'Alpha' determines the skewness of
                the beta values. If 'Alpha' is set to '1.0', the beta values
                would be spaced uniformly on the interval (0.0,1.0). However,
                better results are obtained if the beta values are skewed.
                Empirically, it was observed that 'Alpha' values in the range
                of 0.3 to 0.5 produce the most accurate results.
Burninss     -- Fixed number of samples discarded before sampling of the first
                step starts. 'Burninss' can be specified using either a pos-
                itive or a negative number. If the number is positive, it is
                interpreted as the number of samples to discard as burnin. If
                the number is negative, its absolute value is interpreted as
                the length of the burnin in terms of the length of each of the
                following steps in the stepping-stone algorithm. For instance,
                a value of '-1' means that the length of the burnin is the
                same as the length of each of the subsequent steps.
Nsteps       -- Number of steps in the stepping-stone algorithm. Typically, a
                number above 30 is sufficient for accurate results.
FromPrior    -- If it is set to 'Yes', it indicates that in the first step we
                sample from the prior, with each consequtive step we sample
                closer to the posterior. 'No' indicates the opposite direction
                of power posterior change, i.e. in the first step we sample
                close to the posterior, and with each consequtive step we
                sample closer to the prior.

Current settings:

Parameter          Options               Current Setting
-------------------------------------------------------
Alpha              <number>              0.40
```

```
BurninSS            <number>              -1
Nsteps              <number>              50
FromPrior            Yes/No               No

------------------------------------------------------------------------------
------------------------------------------------------------------------------
Ssp

This command sets the parameters of the stepping-stone sampling
analysis without actually starting the chain. This command is identical
in all respects to Ss, except that the analysis will not start after
this command is issued. For more details on the options, check the help
menu for Ss.

Current settings:

Parameter           Options               Current Setting
-------------------------------------------------------
Alpha               <number>              0.40
BurninSS            <number>              -1
Nsteps              <number>              50
FromPrior            Yes/No               No

------------------------------------------------------------------------------
------------------------------------------------------------------------------
Startvals

Use this command to change the current values for parameters in your model.
These values will be used as the starting values in the next mcmc analysis.
The basic format is:

   startvals <param>=(<value_1>,<value_2>,...,<value_n>)

for all substitution model parameters. The format is slightly different for
parameters that are written to a tree file:

   startvals <param>=<tree_name>

This version of the command will look for a tree with the specified name
among the trees read in previously when parsing a tree block. The information
stored in that tree will be used to set the starting value of the parameter.
The parameters that are set using this mechanism include topology and branch
length parameters, as well as relaxed clock branch rates, cpp events and
cpp branch rate multipliers.

The above versions of the command will set the value for all runs and chains.
You can also set the value for an individual run and chain by using the format
```

```
   startvals <param>(<run>,<chain>)=(<value_1>,...)
```

where <run> is the index of the run and <chain> the index of the chain. If the run index is omitted, the values will be changed for all runs. Similarly, if the chain index is omitted, all chains will be set to the specified value. For example, if we wanted to set the values of the stationary frequency parameter pi{1} to (0.1,0.1,0.4,0.4) for all chains in run 1, and to (0.3,0.3,0.2,0.2) for chain 3 of run 2, we would use

```
   startvals pi{1}(1,)=(0.1,0.1,0.4,0.4) pi{1}(2,3)=(0.3,0.3,0.2,0.2)
```

Sump

During an MCMC analysis, MrBayes prints the sampled parameter values to one or more tab-delimited text files, one for each independent run in your analysis. The command 'Sump' summarizes the information in this parameter file or these parameter files. By default, the root of the parameter file name(s) is assumed to be the name of the last matrix-containing nexus file. MrBayes also remembers the number of independent runs in the last analysis that you set up, regardless of whether you actually ran it. For instance, if there were two independent runs, which is the initial setting when you read in a new matrix, MrBayes will assume that there are two parameter files with the endings '.run1.p' and '.run2.p'. You can change the root of the file names and the number of runs using the 'Filename' and 'Nruns' settings.

When you invoke the 'Sump' command, three items are output: (1) a generation plot of the likelihood values; (2) estimates of the marginal likelihood of the model; and (3) a table with the mean, variance, and 95 percent credible interval for the sampled parameters. All three items are output to screen. The table of marginal likelihoods is also printed to a file with the ending '.lstat' and the parameter table to a file with the ending '.pstat'. For some model parameters, there may also be a '.mstat' file.

When running 'Sump' you typically want to discard a specified number or fraction of samples from the beginning of the chain as the burn in. This is done using the same mechanism used by the 'mcmc' command. That is, if you run an mcmc analysis with a relative burn in of 25 % of samples for convergence diagnostics, then the same burn in will be used for a subsequent sump command, unless a different burn in is specified. That is, issuing

```
sump
```

immediately after 'mcmc', will result in using the same burn in settings as for the 'mcmc' command. All burnin settings are reset to default values every time a new matrix is read in, namely relative burnin ('relburnin=yes') with 25 % of samples discarded ('burninfrac = 0.25').

```
Options:

Relburnin    -- If this option is set to 'Yes', then a proportion of the
                samples will be discarded as burnin when calculating summary
                statistics. The proportion to be discarded is set with
                'Burninfrac' (see below). When the 'Relburnin' option is set
                to 'No', then a specific number of samples is discarded
                instead. This number is set by 'Burnin' (see below). Note that
                the burnin setting is shared across the 'sumt', 'sump', and
                'mcmc' commands.
Burnin       -- Determines the number of samples (not generations) that will
                be discarded when summary statistics are calculated. The
                value of this option is only applicable when 'Relburnin' is
                set to 'No'.
Burninfrac   -- Determines the fraction of samples that will be discarded when
                summary statistics are calculated. The setting only takes
                effect if 'Relburnin' is set to 'Yes'.
Nruns        -- Determines how many '.p' files from independent analyses that
                will be summarized. If Nruns > 1 then the names of the files
                are derived from 'Filename' by adding '.run1.p', '.run2.p',
                etc. If Nruns=1, then the single file name is obtained by
                adding '.p' to 'Filename'.
Filename     -- The name of the file to be summarized. This is the base of the
                file name to which endings are added according to the current
                setting of the 'Nruns' parameter. If 'Nruns' is 1, then only
                '.p' is added to the file name. Otherwise, the endings will
                be '.run1.p', '.run2.p', etc.
Outputname   -- Base name of the file(s) to which 'Sump' results will be
                printed.
Hpd          -- Determines whether credibility intervals will be given as the
                region of Highest Posterior Density ('Yes') or as the interval
                containing the median 95 % of sampled values ('No').
Minprob      -- Determines the minimum probability of submodels to be included
                in summary statistics. Only applicable to models that explore
                submodel spaces, like 'nst=mixed' and 'aamodelpr=mixed'.

Current settings:

Parameter       Options                 Current Setting
-------------------------------------------------------
Relburnin       Yes/No                  Yes
Burnin          <number>                0
Burninfrac      <number>                0.25
Nruns           <number>                2
Filename        <name>                  temp.p<.run<i>.p>
Outputname      <name>                  <.pstat etc>
Hpd             Yes/No                  Yes
```

Minprob <number> 0.050

Sumss

This command summarizes results of stepping stone analyses. It is a tool to investigate the obtained results, and to help find the proper step burn-in. To get more help information on stepping-stone analyses, use 'help ss'.

During stepping-stone analysis, MrBayes collects the sampled likelihoods in order to estimate the marginal likelihood at the end. It also prints the sampled parameter values to one or more tab-delimited text files, one for each independent run in your analysis. The command 'Sumss' summarizes likelihood values stored in these parameter files and calculates marginal likelihood estimates. The names of the files that are summarized are exactly the same as the names of the files used for the 'sump' command. In fact, the 'filename' setting is a shared setting for the 'sump' and 'sumss' commands. That is, if you change the setting in one of the commands, it would change the setting in the other command as well.

When you invoke the 'Sumss' command, three items are output: (1) 'Step contribution table' - summarizes the contribution of each step to the overall estimate; (2) 'Step plot' - plot of the likelihood values for the initial burn-in phase or a chosen step in the stepping-stone algorithm; (3) 'Joined plot' - summarizes sampling across all steps in the algorithm.

Step contribution table
The printed table is similar to the one output to the .ss file. The main purpose of the table is to summarize marginal likelihood for different values of the step burn-in after the stepping stone analysis has finished. The burn-in is controlled by the 'Relburnin', 'Burnin' and 'Burninfrac' settings.
Note that during stepping-stone analyses, step contributions to marginal likelihood are calculated based on all generations excluding burn-in. 'Sumss' on the other hand makes estimates based only on the sampled generations. This may lead to slight difference in results compared to the one printed to the .ss file.

Step plot
The main objective of the plot is to provide a close look at a given step in the analysis. Which step is printed here is defined by the 'Steptoplot' setting. The plot could be used to inspect if the chosen step burn-in is appropriate for the given step. It could also be used to check if the initial burn-in phase has converged. Note that the amount of discarded samples is controled by the 'Discardfrac' setting, and not by the ordinary burn-in settings.

Joined plot
Different steps sample from different power posterior distributions. When we

switch from one distribution to another, it takes some number of generations before the chain settles at the correct stationary distribution. This lag is called a 'temperature lag' and if the corresponding samples are not removed, it will result in a biased estimate. It is difficult to determine the lag beforehand, but MrBayes allows you to explore different step burn-in settings after you have finished the stepping-stone algorithm, without having to rerun the whole analysis. The 'Joined plot' helps to facilitate the choice of the right step burn-in. The plot summarizes samples across all steps and gives you a quick overview of the whole analysis.

Specifically, the following procedure is used to obtain the joined plot. Each step has the same number N of samples taken. We number each sample 1 to N within steps according to the order in which the samples are taken. The first sample in each step is numbered 1, and the last sample is N. For each number i in [1,..., N], we sum up log likelihoods for all samples numbered i across all steps. The joined plot is a graph of the step number versus the normalized sums we get in the procedure describe above. This directly visualizes the temperature lag and allows you to select the appropriate step burn-in.

Ideally, after you discard the appropriate step burn-in, the graph should appear as white noise around the estimated value. If you see an increasing or decreasing tendency in the beginning of the graph, you should increase the step burn-in. If you see an increasing or decreasing tendency across the whole graph, then the initial burn-in phase was not long enough. In this case, you need to rerun the analysis with a longer initial burn-in.

To make it easier to observe tendencies in the plotted graph you can choose different levels of curve smoothing. If 'Smoothing' is set to k, it means that for each step i we take an average over step i and k neighboring samples in both directions, i.e., the k-smoothed estimate for step i is an average over values for steps [i-k,...,i+k].

Options:

Allruns -- If set to 'Yes', it forces all runs to be printed on the same graph when drawing joined and step plots. If set to 'No', each run is printed on a separat plot.

Askmore -- Long analyses may produce huge .p files. Reading in them may take several minutes. If you want to investigate different aspects of your analyses, it could be very inconvenient to wait for several minutes each time you want to get a new summary for different settings. If you set 'Askmore' to 'YES', sumss will read .p files only once. After responding to the original query, it will interactivaly ask you if you wish to produce more tables and plots for different settings of 'Burnin' or 'Smoothing' (see below).

Relburnin -- If this option is set to 'Yes', then a proportion of the

```
                samples from each step will be discarded as burnin when calcu-
                lsting summary statistics. The proportion to be discarded is
                set with 'Burninfrac' (see below). When the 'Relburnin' option
                is set to 'No', then a specific number of samples is discarded
                instead. This number is set by 'Burnin'. Note that the burnin
                settings --- 'Relburnin', 'Burnin', and 'Burninfrac' --- are
                shared across the 'sumt', 'sump', 'sumss' and 'mcmc' commands.
Burnin       -- Determines the number of samples (not generations) that will
                be discarded from each step when summary statistics are calcu-
                lated. The value of this option is only applicable when
                'Relburnin' is set to 'No'.
Burninfrac   -- Determines the fraction of samples that will be discarded from
                each step when summary statistics are calculated. The setting
                only takes effect if 'Relburnin' is set to 'Yes'.
Discardfrac  -- Determines the fraction of samples that will be discarded when
                a step plot is printed. It is similar to the 'Burninfrac' set-
                ting, but unlike 'Burninfrac' it is used only for better vis-
                ualization of the step plot. It has no effect on the number of
                samples discarded during marginal likelihood computation.
Filename     -- The name of the file to be summarized. This is the base of the
                file name to which endings are added according to the current
                setting of the 'Nruns' parameter. If 'Nruns' is 1, then only
                '.p' is added to the file name. Otherwise, the endings will
                be '.run1.p', '.run2.p', etc. Note that the 'Filename' setting
                is shared with 'sump' command.
Nruns        -- Determines how many '.p' files from independent analyses that
                will be summarized. If Nruns > 1 then the names of the files
                are derived from 'Filename' by adding '.run1.p', '.run2.p',
                etc. If Nruns=1, then the single file name is obtained by
                adding '.p' to 'Filename'.
Steptoplot   -- Defines which step will be printed in the step plot.If the
                value is set to 0, then the initial sample from the posterior
                will be used.
Smoothing    -- Determines smoothing of the joined plot (see above). A value
                equal to 0 results in no smoothing.

Current settings:

Parameter       Options                 Current Setting
-------------------------------------------------------
Allruns         Yes/No                  Yes
Askmore         Yes/No                  Yes
Relburnin       Yes/No                  Yes
Burnin          <number>                0
Burninfrac      <number>                0.25
Discardfrac     <number>                0.80
Filename        <name>                  temp.p<.run<i>.p>
Nruns           <number>                2
```

```
Steptoplot      <number>                0
Smoothing       <number>                0
------------------------------------------------------------------------------
------------------------------------------------------------------------------
```

Sumt

This command is used to produce summary statistics for trees sampled during a Bayesian MCMC analysis. You can either summarize trees from one individual analysis, or trees coming from several independent analyses. In either case, all the sampled trees are read in and the proportion of the time any single taxon bipartition (split) is found is counted. The proportion of the time that the bipartition is found is an approximation of the posterior probability of the bipartition. (Remember that a taxon bipartition is defined by removing a branch on the tree, dividing the tree into those taxa to the left and right of the removed branch. This set is called a taxon bipartition.) The branch length of the bipartition is also recorded, if branch lengths have been saved to file. The result is a list of the taxon bipartitions found, the frequency with which they were found, the posterior probability of the bipartition and, the mean and variance of the branch lengths or node depths, and various other statistics.

The key to the partitions is output to a file with the suffix '.parts'. The summary statistics pertaining to bipartition probabilities are output to a file with the suffix '.tstat', and the statistics pertaining to branch or node parameters are output to a file with the suffix '.vstat'.

A consensus tree is also printed to a file with the suffix '.con.tre' and printed to the screen as a cladogram, and as a phylogram if branch lengths have been saved. The consensus tree is either a 50 percent majority rule tree or a majority rule tree showing all compatible partitions. If branch lengths have been recorded during the run, the '.con.tre' file will contain a consensus tree with branch lengths and interior nodes labelled with support values. By default, the consensus tree will also contain other summary information in a format understood by the program 'FigTree'. To use a simpler format understood by other tree-drawing programs, such as 'TreeView', set 'Conformat' to 'Simple'.

MrBayes alo produces a file with the ending ".trprobs" that contains a list of all the trees that were found during the MCMC analysis, sorted by their probabilities. This list of trees can be used to construct a credible set of trees. For example, if you want to construct a 95 percent credible set of trees, you include all of those trees whose cumulative probability is less than or equal to 0.95. You have the option of displaying the trees to the screen using the "Showtreeprobs" option. The default is to not display the trees to the screen; the number of different trees sampled by the chain can be quite large. If you are analyzing a large set of taxa, you may actually want to skip the calculation of tree probabilities entirely by setting 'Calctreeprobs' to 'No'.

When calculating summary statistics you probably want to skip those trees that were sampled in the initial part of the run, the so-called burn-in period. The number of skipped samples is controlled by the 'Relburnin', 'Burnin', and 'Burninfrac' settings, just as for the 'Mcmc' command. Since version 3.2.0, the burn-in settings are shared across the 'Sumt', 'Sump' and 'Mcmc' commands. That is, changing the burn-in setting for one command will change the settings for subsequent calls to any of the other commands.

If you are summarizing the trees sampled in several independent analyses, such as those resulting from setting the 'Nruns' option of the 'Mcmc' command to a value larger than 1, MrBayes will also calculate convergence diagnostics for the sampled topologies and branch lengths. These values can help you determine whether it is likely that your chains have converged.

The 'Sumt' command expands the 'Filename' according to the current values of the 'Nruns' and 'Ntrees' options. For instance, if both 'Nruns' and 'Ntrees' are set to 1, 'Sumt' will try to open a file named '<Filename>.t'. If 'Nruns' is set to 2 and 'Ntrees' to 1, then 'Sumt' will open two files, the first named '<Filename>.run1.t' and the second '<Filename>.run2.t', etc. By default, the 'Filename' option is set such that 'Sumt' automatically summarizes all the results from your immediately preceding 'Mcmc' command. You can also use the 'Sumt' command to summarize tree samples in older analyses. If you want to do that, remember to first read in a matrix so that MrBayes knows what taxon names to expect in the trees. Then set the 'Nruns', 'Ntrees' and 'Filename' options appropriately if they differ from the MrBayes defaults.

Options:

Relburnin -- If this option is set to YES, then a proportion of the samples will be discarded as burnin when calculating summary statistics. The proportion to be discarded is set with Burninfrac (see below). When the Relburnin option is set to NO, then a specific number of samples is discarded instead. This number is set by Burnin (see below). Note that the burnin setting is shared across the 'sumt', 'sump', and 'mcmc' commands.

Burnin -- Determines the number of samples (not generations) that will be discarded when summary statistics are calculated. The value of this option is only relevant when Relburnin is set to NO.

BurninFrac -- Determines the fraction of samples that will be discarded when summary statistics are calculated. The value of this option is only relevant when Relburnin is set to YES. Example: A value for this option of 0.25 means that 25% of the samples will be discarded.

Nruns -- Determines how many '.t' files from independent analyses that will be summarized. If Nruns > 1 then the names of the files

```
                are derived from 'Filename' by adding '.run1.t', '.run2.t',
                etc. If Nruns=1 and Ntrees=1 (see below), then only '.t' is
                added to 'Filename'.
Ntrees        -- Determines how many trees there are in the sampled model. If
                'Ntrees' > 1 then the names of the files are derived from
                'Filename' by adding '.tree1.t', '.tree2.t', etc. If there
                are both multiple trees and multiple runs, the filenames will
                be '<Filename>.tree1.run1.t', '<Filename>.tree1.run2.t', etc.
Filename      -- The name of the file(s) to be summarized. This is the base of
                the file name, to which endings are added according to the
                current settings of the 'Nruns' and 'Ntrees' options.
Minpartfreq   -- The minimum probability of partitions to include in summary
                statistics.
Contype       -- Type of consensus tree. 'Halfcompat' results in a 50% major-
                ity rule tree, 'Allcompat' adds all compatible groups to such
                a tree.
Conformat     -- Format of consensus tree. The 'Figtree' setting results in a
                consensus tree formatted for the program FigTree, with rich
                summary statistics. The 'Simple' setting results in a simple
                consensus tree written in a format read by a variety of pro-
                grams.
Outputname    -- Base name of the file(s) to which 'sumt' results will be
                printed. The default is the same as 'Filename'.
Calctreeprobs -- Determines whether tree probabilities should be calculated.
Showtreeprobs -- Determines whether tree probabilities should be displayed on
                screen.
Hpd           -- Determines whether credibility intervals will be given as the
                region of Highest Posterior Density ('Yes') or as the inter-
                val containing the median 95 % of sampled values ('No').

Current settings:

Parameter       Options                 Current Setting
-------------------------------------------------------
Relburnin       Yes/No                  Yes
Burnin          <number>                0
Burninfrac      <number>                0.25
Nruns           <number>                2
Ntrees          <number>                1
Filename        <name>                  temp.t<.run<i>.t>
Minpartfreq     <number>                0.10
Contype         Halfcompat/Allcompat    Halfcompat
Conformat       Figtree/Simple          Figtree
Outputname      <name>                  temp.t.stat<.parts etc>
Calctreeprobs   Yes/No                  Yes
Showtreeprobs   Yes/No                  No
Hpd             Yes/No                  Yes
```


Taxastat

This command shows the status of all the taxa. The correct usage is

   taxastat

After typing "taxastat", the taxon number, name, and whether it is excluded or included are shown.

Taxset

This command defines a taxon set. The format for the taxset command is

   taxset <name> = <taxon names or numbers>

For example, "taxset apes = Homo Pan Gorilla Orang gibbon" defines a taxon set called "apes" that includes five taxa (namely, apes). You can assign up to 30 taxon sets. This option is best used not from the command line but rather as a line in the mrbayes block of a file.

Unlink

This command unlinks model parameters across partitions of the data. The correct usage is:

   unlink <parameter name> = (<all> or <partition list>)

A little background is necessary to understand this command. Upon execution of a file, a default partition is set up. This partition referenced either by its name ("default") or number (0). If your data are all of one type, then this default partition does not actually divide up your characters. However, if your datatype is mixed, then the default partition contains as many divisions as there are datatypes in your character matrix. Of course, you can also define other partitions, and switch among them using the set command ("set partition=<name/number>"). Importantly, you can also assign model parameters to individual partitions or to groups of them using the "applyto" option in lset and prset. When the program attempts to perform an analysis, the model is set for individual partitions. If the same parameter applies to different partitions and if that parameter has the same prior, then the program will link the parameters: that is, it will use a single value for the parameter. The program's default, then, is to strive for parsimony.

```
However, there are lots of cases where you may want unlink a parameter
across partitions. For example, you may want a different transition/
transversion rate ratio to apply to different partitions. This command
allows you to unlink the parameters, or to make them different across
partitions. The converse of this command is "link", which links to-
gether parameters that were previously told to be different. The list
of parameters that can be unlinked includes:

   Tratio          -- Transition/transversion rate ratio
   Revmat          -- Substitution rates of GTR model
   Omega           -- Nonsynonymous/synonymous rate ratio
   Statefreq       -- Character state frequencies
   Shape           -- Gamma shape parameter
   Pinvar          -- Proportion of invariable sites
   Correlation     -- Correlation parameter of autodiscrete gamma
   Switchrates     -- Switching rates for covarion model
   Brlens          -- Branch lengths of tree
   Topology        -- Topology of tree
   Speciationrates -- Speciation rates for birth-death process
   Ratemultiplier  -- Rate multiplier for partitions
   Extinctionrates -- Extinction rates for birth-death process
   Theta           -- Parameter for coalescence process
   Growthrate      -- Growth rate of coalescence process
   Aamodel         -- Aminoacid rate matrix
   Cpprate         -- Rate of Compound Poisson Process (CPP)
   Cppmultdev      -- Standard dev. of CPP rate multipliers (log scale)
   Cppevents       -- CPP events
   TK02var         -- Variance increase in TK02 relaxed clock model
   TK02branchrates -- Branch rates of TK02 relaxed clock model
   Igrvar          -- Variance increase in IGR relaxed clock model
   Igrbranchlens   -- Branch lengths of IGR relaxed clock model

For example,

   unlink shape=(all)

unlinks the gamma shape parameter across all partitions of the data.
You can use "showmodel" to see the current linking status of the
characters.
------------------------------------------------------------------------
------------------------------------------------------------------------
Version

This command shows the release version of the program.
------------------------------------------------------------------------

************************************************************************
*                                                                      *
```

```
*  3. 'Data' or 'tree' block commands (in #NEXUS file)                    *
*                                                                         *
***************************************************************************

---------------------------------------------------------------------------
Begin

This command is used to format data or commands in the program. The correct
usage is

   begin <data or mrbayes>;

The two valid uses of the "begin" command, then, are

   begin data;
   begin mrbayes;

The "data" specifier is used to specify the beginning of a data block; your
character data should follow. For example, the following is an example of
a data block for four taxa and ten DNA sites:

   begin data;
      dimensions ntax=4 nchar=10;
      format datatype=dna;
      matrix
      taxon_1  AACGATTCGT
      taxon_2  AAGGATTCCA
      taxon_3  AACGACTCCT
      taxon_4  AAGGATTCCT
      ;
   end;

The other commands -- dimensions, format, and matrix -- are discussed
in the appropriate help menu. The only thing to note here is that the
block begins with a "begin data" command. The "mrbayes" command is
used to enter commands specific to the MrBayes program into the file.
This allows you to automatically process commands on execution of the
program. The following is a simple mrbayes block:

   begin mrbayes;
      charset first  = 1-10\3;
      charset second = 2-10\3;
      charset third  = 3-10\3;
   end;

This mrbayes block sets off the three "charset" commands, used to
predefine some blocks of characters. The mrbayes block can be very useful.
For example, in this case, it would save you the time of typing the char-
```

acter sets each time you executed the file. Also, note that every "begin <data or mrbayes>" command ends with an "end". Finally, you can have so-called foreign blocks in the file. An example of a foreign block would be "begin paup". The program will simply skip this block. This is useful because it means that you can use the same file for MrBayes, PAUP* or MacClade (although it isn't clear why you would want to use those other programs).

Dimensions

This command is used in a data block to define the number of taxa and characters. The correct usage is

```
dimensions ntax=<number> nchar=<number>
```

The dimensions must be the first command in a data block. The following provides an example of the proper use of this command:

```
begin data;
   dimensions ntax=4 nchar=10;
   format datatype=dna;
   matrix
   taxon_1  AACGATTCGT
   taxon_2  AAGGATTCCA
   taxon_3  AACGACTCCT
   taxon_4  AAGGATTCCT
   ;
end;
```

Here, the dimensions command tells MrBayes to expect a matrix with four taxa and 10 characters.

End

This command is used to terminate a data or mrbayes block. The correct usage is

```
end;
```

For more information on this, check the help for the "begin" command.

Endblock

This is an older, deprecated version of "End", see that command.

Format

This command is used in a data block to define the format of the character matrix. The correct usage is

```
format datatype=<name> ... <parameter>=<option>
```

The format command must be the second command in a data block. The following provides an example of the proper use of this command:

```
begin data;
   dimensions ntax=4 nchar=10;
   format datatype=dna gap=-;
   matrix
   taxon_1  AACGATTCGT
   taxon_2  AAGGAT--CA
   taxon_3  AACGACTCCT
   taxon_4  AAGGATTCCT
   ;
end;
```

Here, the format command tells MrBayes to expect a matrix with DNA characters and with gaps coded as "-".

The following are valid options for format:

```
Datatype   -- This parameter MUST BE INCLUDED in the format command. Moreover, it must be the first parameter in the line. The datatype command specifies what type of characters are in the matrix. The following are valid options:
                 Datatype = Dna: DNA states (A,C,G,T,R,Y,M,K,S,W,H,B,
                            V,D,N)
                 Datatype = Rna: DNA states (A,C,G,U,R,Y,M,K,S,W,H,B,
                            V,D,N)
                 Datatype = Protein: Amino acid states (A,R,N,D,C,Q,E,
                            G,H,I,L,K,M,F,P,S,T,W,Y,V)
                 Datatype = Restriction: Restriction site (0,1) states
                 Datatype = Standard: Morphological (0,1) states
                 Datatype = Continuous: Real number valued states
                 Datatype = Mixed(<type>:<range>,...,<type>:<range>): A
                            mixture of the above datatypes. For example,
                            "datatype=mixed(dna:1-100,protein:101-200)"
                            would specify a mixture of DNA and amino acid
                            characters with the DNA characters occupying
                            the first 100 sites and the amino acid characters occupying the last 100 sites.
```

Interleave -- This parameter specifies whether the data matrix is in interleave format. The valid options are "Yes" or "No", with "No" as the default. An interleaved matrix looks like

```
format datatype=dna gap=- interleave=yes;
matrix
taxon_1  AACGATTCGT
taxon_2  AAGGAT--CA
taxon_3  AACGACTCCT
taxon_4  AAGGATTCCT

taxon_1  CCTGGTAC
taxon_2  CCTGGTAC
taxon_3  ---GGTAG
taxon_4  ---GGTAG
;
```

Gap -- This parameter specifies the format for gaps. Note that gap character can only be a single character and that it cannot correspond to a standard state (e.g., A,C,G,T,R,Y, M,K,S,W,H,B,V,D,N for nucleotide data).

Missing -- This parameter specifies the format for missing data. Note that the missing character can only be a single character and cannot correspond to a standard state (e.g., A,C,G,T,R,Y, M,K,S,W,H,B,V,D,N for nucleotide data). This is often an unnecessary parameter to set because many data types, such as nucleotide or amino acid, already have a missing character specified. However, for morphological or restriction site data, "missing=?" is often used to specify ambiguity or unobserved data.

Matchchar -- This parameter specifies the matching character for the matrix. For example,

```
format datatype=dna gap=- matchchar=.;
matrix
taxon_1  AACGATTCGT
taxon_2  ..G...--CA
taxon_3  .....C..C.
taxon_4  ..G.....C.
;
```

is equivalent to

```
format datatype=dna gap=-;
matrix
taxon_1  AACGATTCGT
```

```
                taxon_2  AAGGAT--CA
                taxon_3  AACGACTCCT
                taxon_4  AAGGATTCCT
                ;
```

The only non-standard NEXUS format option is the use of the "mixed", "restriction", "standard" and "continuous" datatypes. Hence, if you use any of these datatype specifiers, a program like PAUP* or MacClade will report an error (as they should because MrBayes is not strictly NEXUS compliant).

Matrix

This command specifies the actual data for the phylogenetic analysis. The character matrix should follow the dimensions and format commands in a data block. The matrix can have all of the characters for a taxon on a single line:

```
   begin data;
      dimensions ntax=4 nchar=10;
      format datatype=dna gap=-;
      matrix
      taxon_1  AACGATTCGT
      taxon_2  AAGGAT--CA
      taxon_3  AACGACTCCT
      taxon_4  AAGGATTCCT
      ;
   end;
```

or be in "interleaved" format:

```
   begin data;
      dimensions ntax=4 nchar=20;
      format datatype=dna gap=- interleave=yes;
      matrix
      taxon_1  AACGATTCGT
      taxon_2  AAGGAT--CA
      taxon_3  AACGACTCCT
      taxon_4  AAGGATTCCT

      taxon_1  TTTTCGAAGC
      taxon_2  TTTTCGGAGC
      taxon_3  TTTTTGATGC
      taxon_4  TTTTCGGAGC
      ;
   end;
```

Note that the taxon names must not have spaces. If you really want to indicate a space in a taxon name (perhaps between a genus and species name), then you might use an underline ("_"). There should be at least a single space after the taxon name, separating the name from the actual data on that line. There can be spaces between the characters.

If you have mixed data, then you specify all of the data in the same matrix. Here is an example that includes two different data types:

```
begin data;
   dimensions ntax=4 nchar=20;
   format datatype=mixed(dna:1-10,standard:21-30) interleave=yes;
   matrix
   taxon_1  AACGATTCGT
   taxon_2  AAGGAT--CA
   taxon_3  AACGACTCCT
   taxon_4  AAGGATTCCT

   taxon_1  0001111111
   taxon_2  0111110000
   taxon_3  1110000000
   taxon_4  1000001111
   ;
end;
```

The matrix command is terminated by a semicolon.

Finally, just a note on data presentation. It is much easier for others to (1) understand your data and (2) repeat your analyses if you make your data clean, comment it liberally (using the square brackets), and embed the commands you used in a publication in the mrbayes block. Remember that the data took a long time for you to collect. You might as well spend a little time making the data file look nice and clear to any that may later request the data for further analysis.

Taxlabels

This command defines taxon labels. It could be used within taxa block.

Translate

This command is used by MrBayes to specify the mapping between taxon names and taxon numbers in a Nexus tree file. For instance,

```
translate
```

```
      1 Homo,
      2 Pan,
      3 Gorilla,
      4 Hylobates;
```

establishes that the taxon labeled 1 in the trees that follow is Homo, the taxon labeled 2 is Pan, etc.

Tree

This command is used by MrBayes to write trees to a nexus tree file. Trees are written in the Newick format. For instance,

```
   tree ((1,2),3,4);
```

describes an unrooted tree with taxa 1 and 2 being more closely related to each other than to taxa 3 and 4. If branch lengths are saved to file, they are given after a colon sign immediately following the terminal taxon or the interior node they refer to. An example of an unrooted tree with branch lengths is:

```
   tree ((1:0.064573,2:0.029042):0.041239,3:0.203988,4:0.187654);
```

Trees that are rooted (clock trees) are written with a basal dichotomy instead of a basal trichotomy. If the tree described above had been rooted on the branch leading to taxon 4, it would have been represented as:

```
   tree (((1,2),3),4);
```
