Species Trees with Relaxed Molecular Clocks

Estimating per-species substitution rates using StarBEAST2

Joseph Heled, Remco Bouckaert, Walter Xie,
Alexei J. Drummond and Huw A. Ogilvie

1 Background

In this tutorial we demonstrate the use of StarBEAST2, a fully Bayesian method of species tree estimation and a replacement for BEAST (Heled and Drummond 2010). StarBEAST2 is many times faster than BEAST, and also supports applying a relaxed clock to the species tree. This enables estimating the substitution rates of extant and ancestral species under a multispecies coalescent model.

You will need to download and install the following software:

- **BEAST** - this package contains BEAST, BEAUti, TreeAnnotator, DensiTree, and other programs. This tutorial is written for BEAST 2 (Bouckaert et al. 2014) version 2.4.x, which is available for download from `http://beast2.org/`.
- **Tracer** - this program is used to explore the output of BEAST (and other Bayesian MCMC programs). It graphically and quantitively summarizes the distributions of continuous parameters and provides diagnostic information. At the time of writing, the current version is 1.6, which is available for download from `http://tree.bio.ed.ac.uk/software/tracer/`.
- **FigTree** - this is an application for displaying and printing molecular phylogenies, in particular those obtained using BEAST. At the time of writing, the current version is 1.4.2, which is available for download from `http://tree.bio.ed.ac.uk/software/figtree/`.

2 BEAST

This tutorial will guide you through the analysis of seven loci sampled from 26 individuals representing eight species of pocket gophers, a data set which was originally gathered and analysed by Belfiore et al. 2008. The objective of this tutorial is to estimate the species tree that is most probable given the multi-individual multi-locus sequence data. The species tree has eight taxa, whereas each gene tree has 26 taxa. StarBEAST2 will co-estimate seven gene trees embedded in a shared species tree (Heled and Drummond 2010).

The first step will be to convert a NEXUS file with a DATA or CHARACTERS block into a BEAST XML input file. This is done using the program BEAUti (Bayesian Evolutionary Analysis Utility). This is a user-friendly program for setting the evolutionary model and options for the MCMC analysis. The second step is to actually run BEAST using the input file that contains the data, model and settings. The final step is to explore the output of BEAST in order to diagnose problems and to summarize the results.

BEAUti

Run BEAUti by double clicking on its icon, or by launching the BEAUTi executable file from the command line in Linux.

Set up BEAUti for StarBEAST2

StarBEAST2 can be easily installed from within BEAUTi. First choose the **File/Manage Packages** menu item, which will then display the list of packages available for installation (Figure 1). Select the StarBEAST2 package and then install it by clicking the **Install/Upgrade** button. You must restart BEAUTi after installing any new packages for new features to become available.

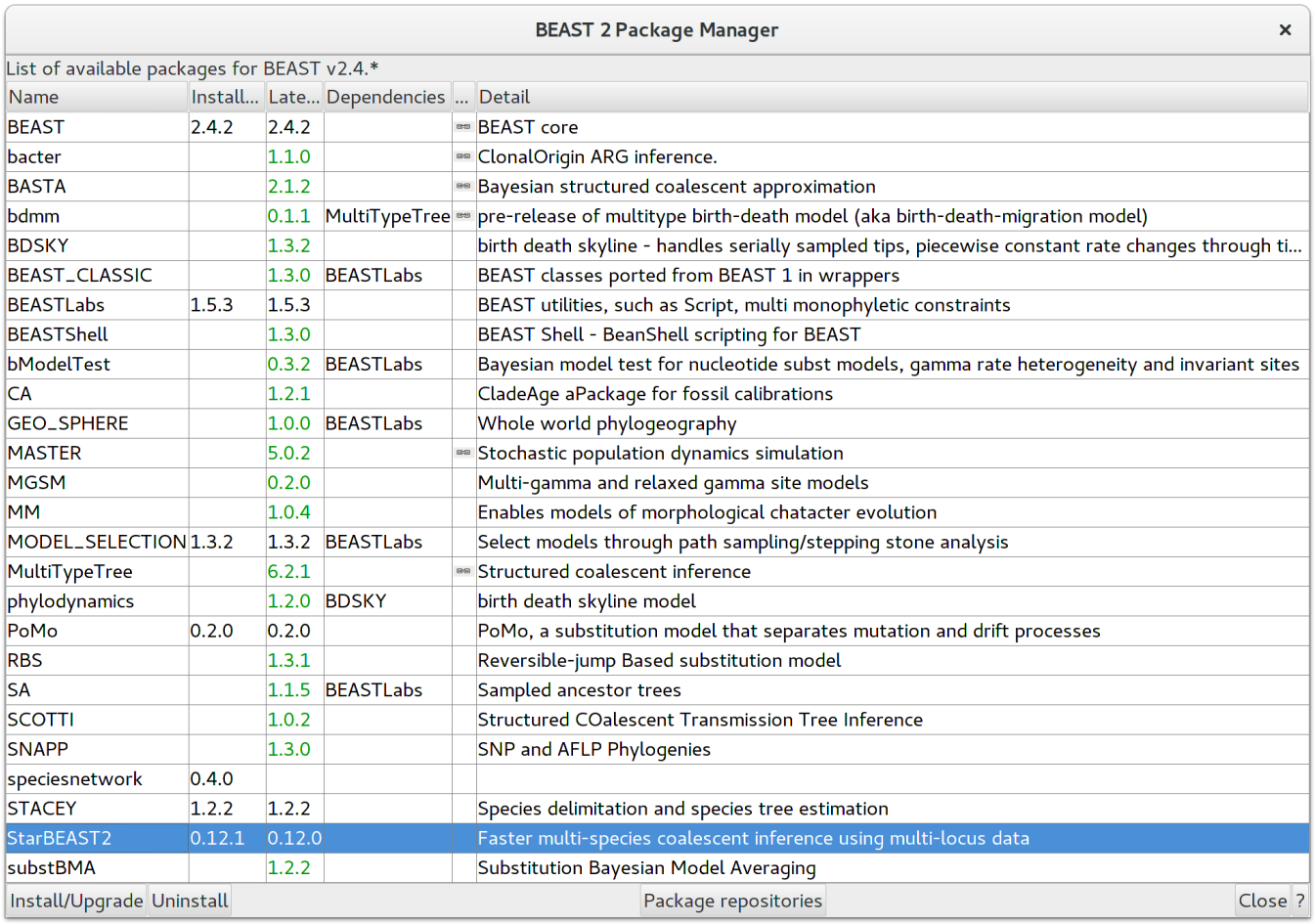

Figure 1: Install StarBEAST2 from within BEAUTi.

StarBEAST2 includes a series of templates for multispecies coalescent analyses. These include the **StarBeast2** template for strict clock or gene tree relaxed clock analyses, and various **SpeciesTree...** templates for species tree relaxed clock analyses. Currently three types of relaxed clocks are supported by StarBEAST2; the uncorrelated lognormal clock (UCLN), the uncorrelated exponential clock (UCED), and the random local clock (RLC) which we will use for this tutorial. The first thing to do is selecting that template by choosing the **File/Template/SpeciesTreeRLC** menu item (Figure 2). When changing a template, BEAUti deletes all previously imported data and starts with a new empty template. So, if you already loaded some data, a warning message pops up indicating that this data will be lost if you switch templates.

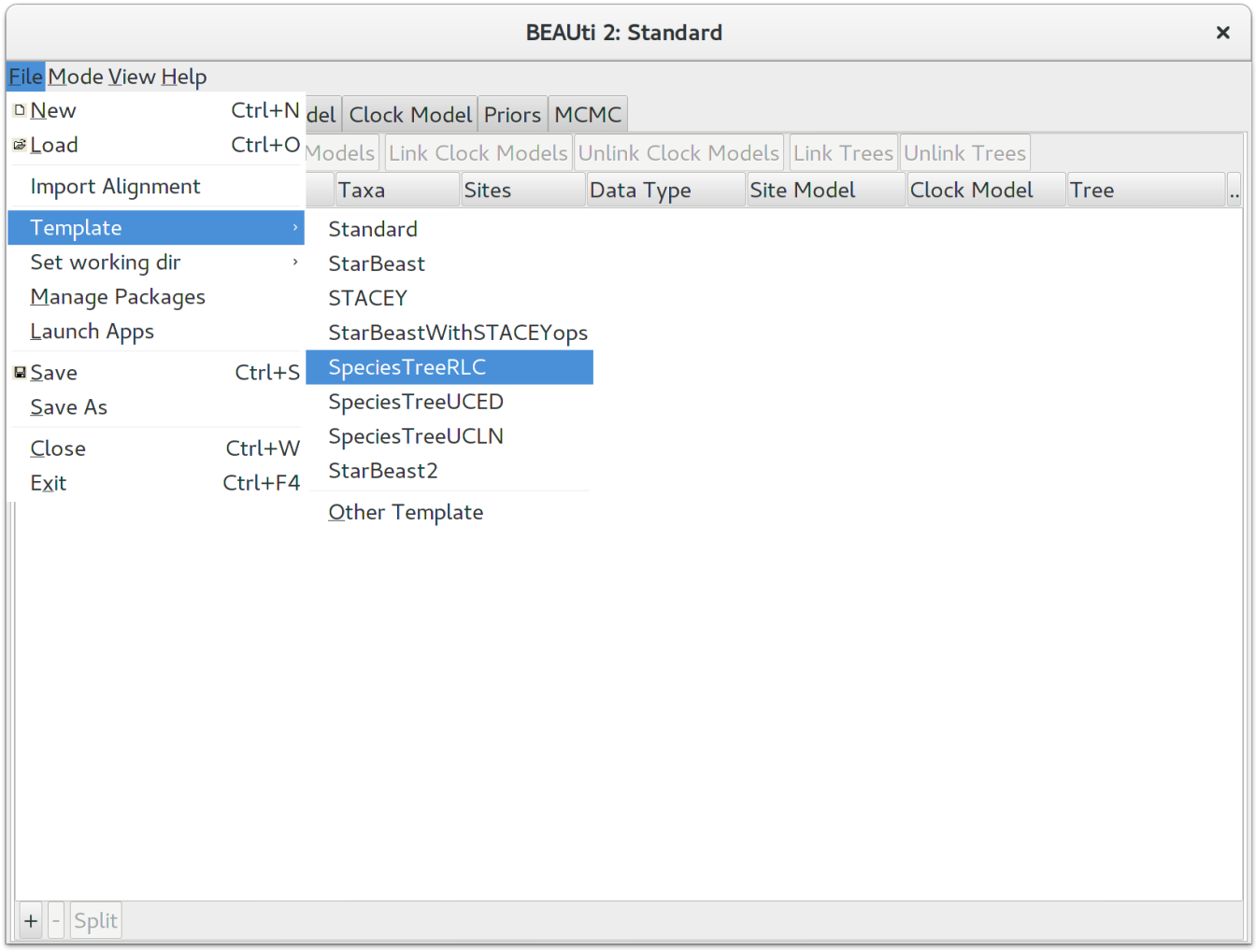

Figure 2: Select a species tree template in BEAUti.

Allow clock rates to vary

By default BEAUTi fixes the clock rate of the first partition to 1, so that the rates of other loci are estimated relative to the first locus. This is generally inappropriate for StarBEAST2 analyses, so it is very important to disable this behaviour by deselecting the **Mode/Automatic set clock rate** menu item (Figure 3).

Loading the NEXUS files

StarBEAST2 supports multiple individuals per-species and multiple loci (we use the term locus to refer to a genomic sequence, and gene when referring to the evolutionary tree for a given locus). The data for each locus is stored as one alignment in its own NEXUS file. Taxa names in each alignment have to be unique, but duplicates across alignments are fine.

To load a NEXUS format alignment, click the button with the plus symbol (+) in the lower left corner of the main **Partitions** tab. For this tutorial, navigate to the **examples/nexus** subfolder inside the **beast** application folder, and select all of the first seven NEXUS files. They should be numbered 26, 29, 47, 53,

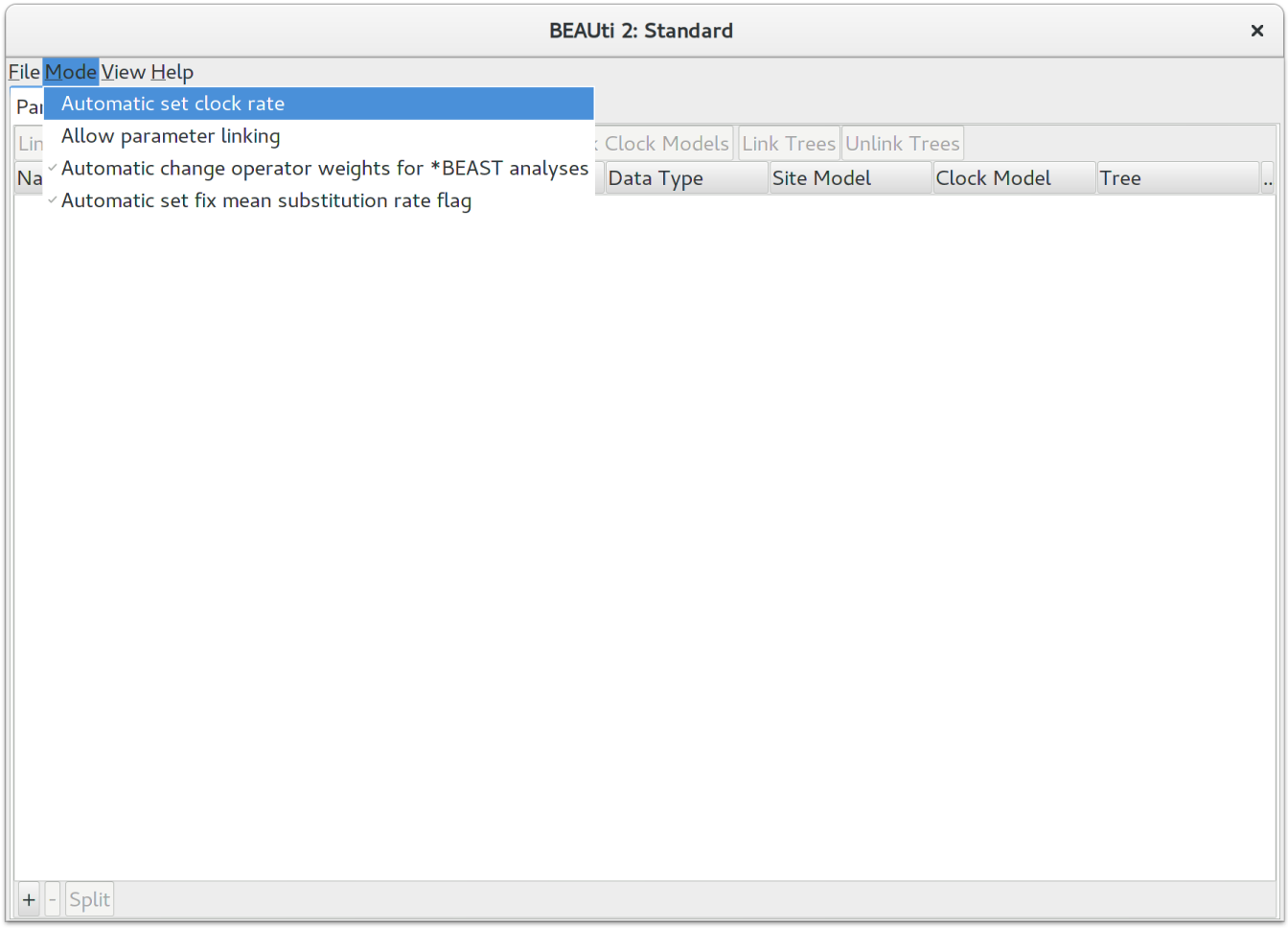

Figure 3: Disable automatic setting of clock rates.

59, 64, and 72 (Figure 4).

Each file contains an alignment of sequences of from an independent locus. The file **26.nex** looks like this (sequences have been truncated):

```
#NEXUS
[TBO26oLong]
BEGIN DATA;
  DIMENSIONS  NTAX =26 NCHAR=614;
  FORMAT DATATYPE = DNA GAP = - MISSING = ?;
  MATRIX
  Orthogeomys_heterodus       ATTCTAGGCAAAAAG-AGCAATGC...
  Thomomys_bottae_awahnee_a   ??????????????????ATGC...
  Thomomys_bottae_awahnee_b   ??????????????????ATGC...
  Thomomys_bottae_xerophilus  ??????????????????ATGC...
...

;
END;
```

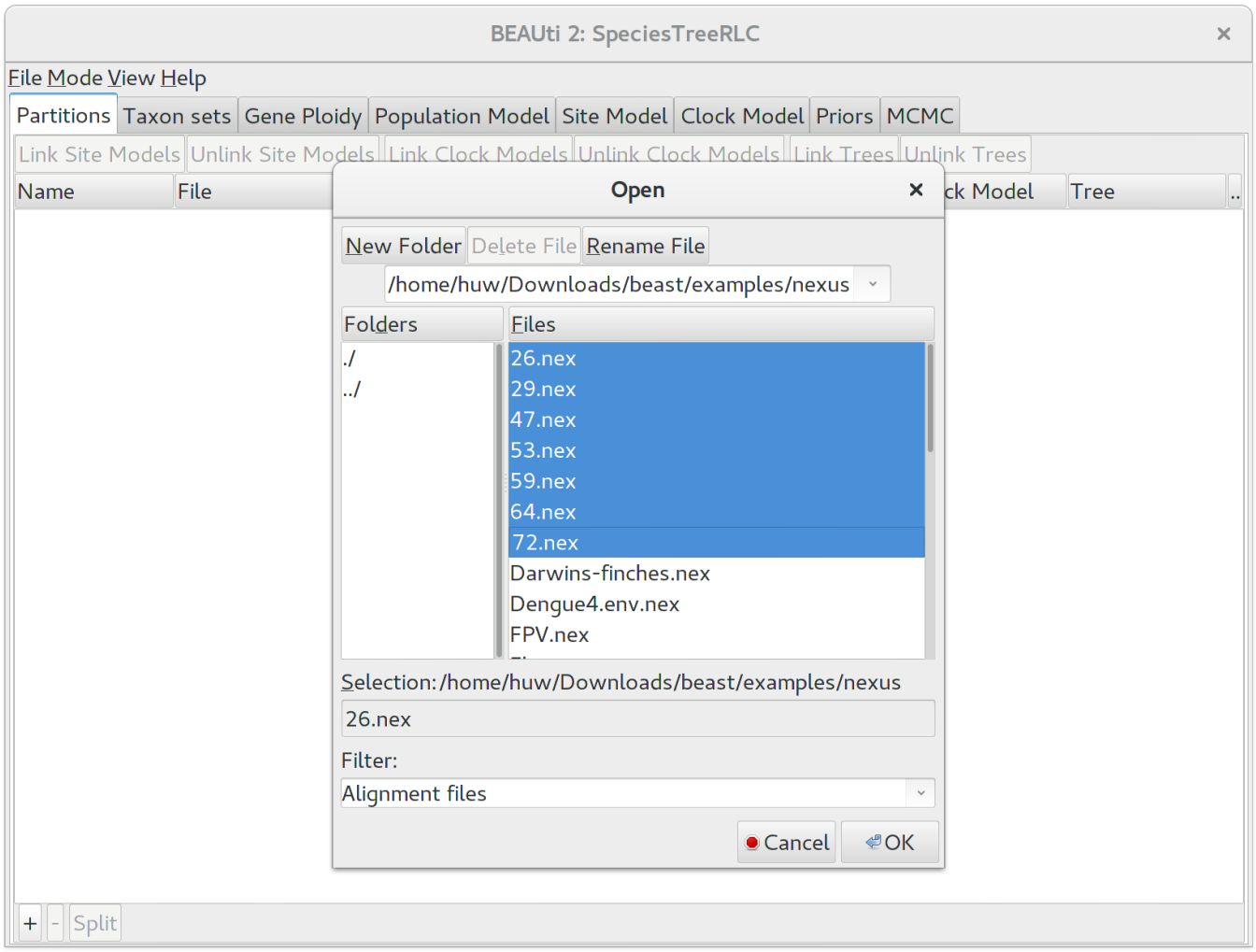

Figure 4: Selecting NEXUS alignment files to import.

Once loaded, the imported alignments are displayed in the **Partitions** panel. You can double click any alignment (partition) to show its detail. For multi-locus analyses, BEAST can link or unlink substitutions models across the loci by clicking buttons on the top of the panel. The default of StarBEAST2 is to unlink all models: substitution, clock, and trees. Note that you should only unlink the tree model across data partitions that are actually genetically unlinked. For example, in most organisms all the mitochondrial genes are effectively linked due to a lack of recombination and they should be set up to use the same tree model in any multispecies coalescent analysis.

Assigning the correct species to each sequence

Each taxon in a StarBEAST2 analysis is associated with a species or similar OTU. Typically the species name is already embedded inside the taxon label. The species name should be easy to extract; place it either at the beginning or the end, separated by a "special" character which does not appear in names. For example, `aria_334259, coast_343436` (using an underscore) or `10x017b.wrussia, 2x305b.eastis` (using a dot).

We need to tell BEAUti somehow which lineages in the alignments go with taxa in the species tree. Select

the Taxon Set panel, and a list of taxa from the alignments is shown together with a default guess by BEAUti. In this case, the guess is not very good, so we want to change this. You can manually change each of the entries in the table, or press the guess button and a dialog is shown where you can choose from several ways to try to detect the taxon from the name of the lineages, or have a mapping stored in a file. In this case, splitting the name on the underscore character '_' and selecting the second group will give us the mapping that we need (Figure 5).

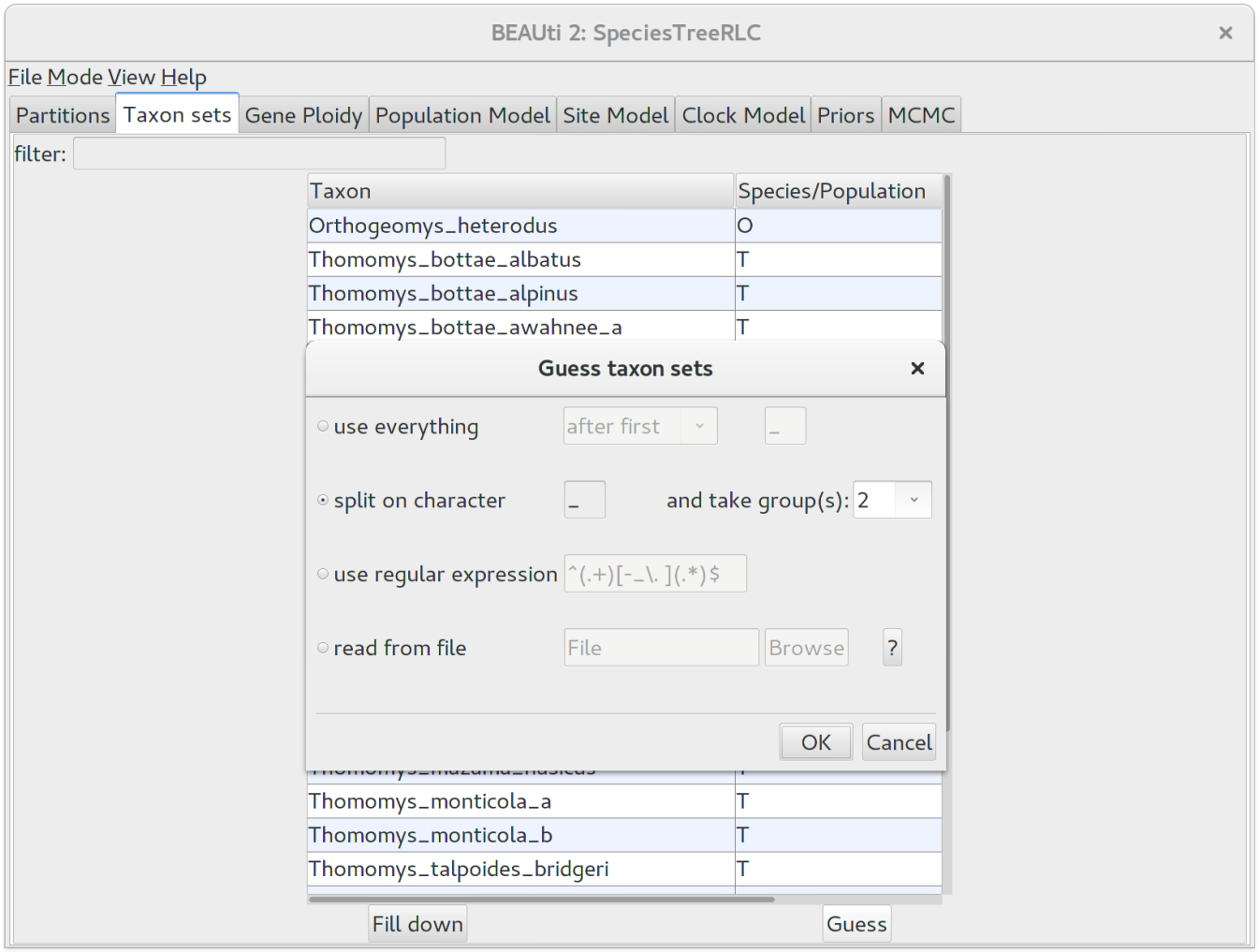

Figure 5: Assigning species to sequences in BEAUti using the guess dialog.

Alternatively, the mapping can be read from a trait file. A proper trait file is tab delimited. The first row is always `traits` followed by the keyword `species` in the second column and separated by tab. The rest of the rows map each individual taxon name to a species name: the taxon name in the first column and species name in the second column separated by tab. For example:

```
traits  species
taxon1  speciesA
taxon2  speciesA
taxon3  speciesB
...
```

Adjusting the ploidy of each gene

Ploidy should be based on the mode of inheritance for each gene. By convention, nuclear genes in diploids are given a ploidy of 2.0. Because mitochondrial and Y chromosome genes are haploid even in otherwise diploid organisms, and also inherited only through the mother or the father respectively, their effective population size N_e is only one quarter that of nuclear genes. Therefore if nuclear gene ploidy is set to 2.0, mitochondrial or Y chromosome gene ploidy should be set to 0.5. In this analysis all genes are from nuclear loci and their ploidy should be left at the default value of 2.0.

Selecting the method of population size integration

StarBEAST2, like BEAST before it, can estimate the effective population sizes for extant and ancestral species. However by default StarBEAST2 analytically integrates over population sizes which is faster than making explicit estimates. If you do need to estimate population sizes, change the population model to **Constant Populations**. For this tutorial, keep the default model which is **Analytical Population Size Integration**.

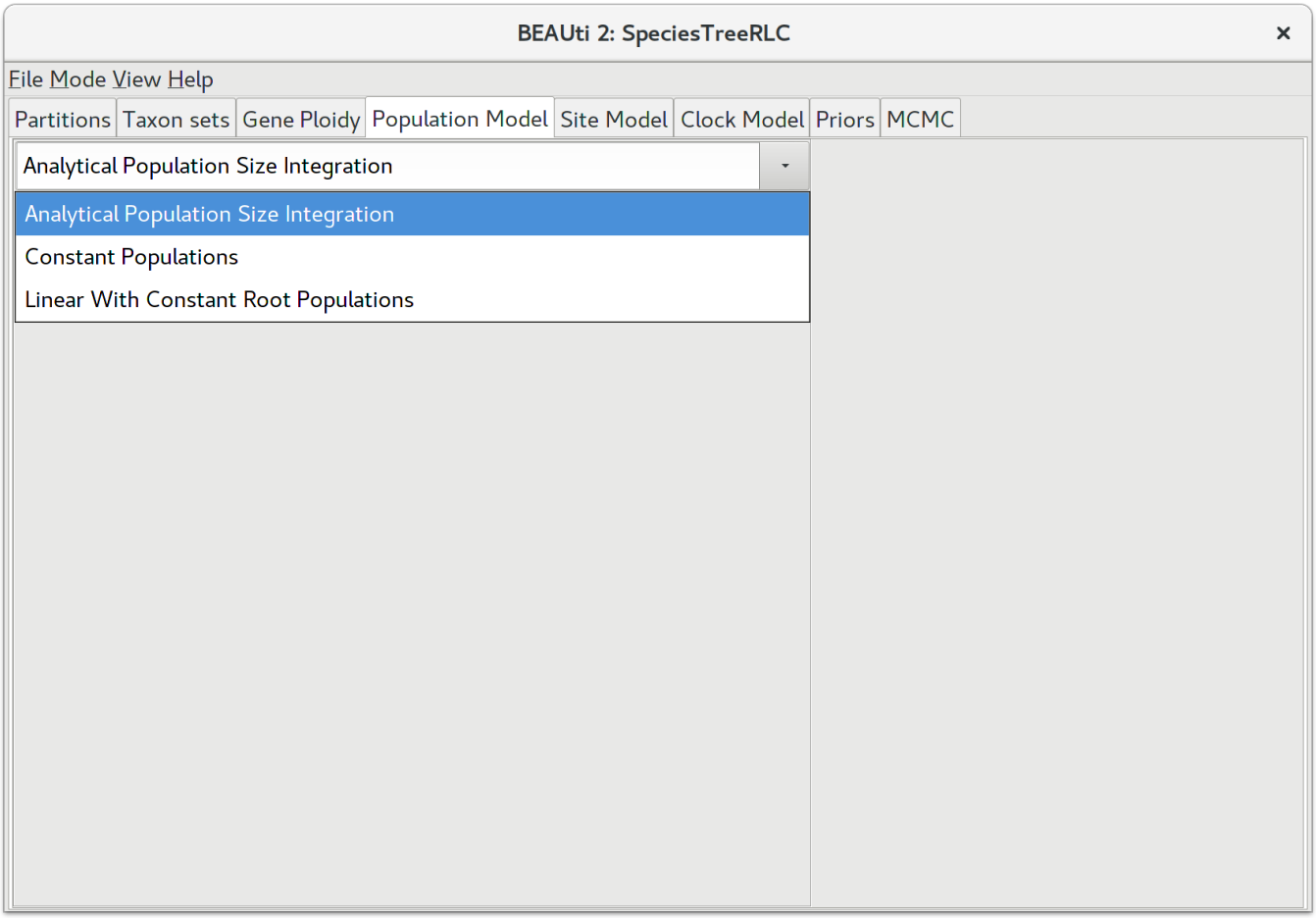

Figure 6: The choice of population models used by StarBEAST2.

Setting the substitution model

The next thing to do is to click on the **Site Model** tab at the top of the main window. This will reveal the evolutionary model settings for BEAST. Exactly which options appear depend on whether the data are nucleotides, or amino acids, or binary data, or general data. The settings that will appear after loading the data set will be the default values so we need to make some changes.

Many of the models may be familiar to you. For this analysis, we will select each substitution model listed on the left side in turn to make the following change: select "HKY" for substitution model (**Subst Model** in Figure 7). Make sure to repeat this step for every partition listed on the left side.

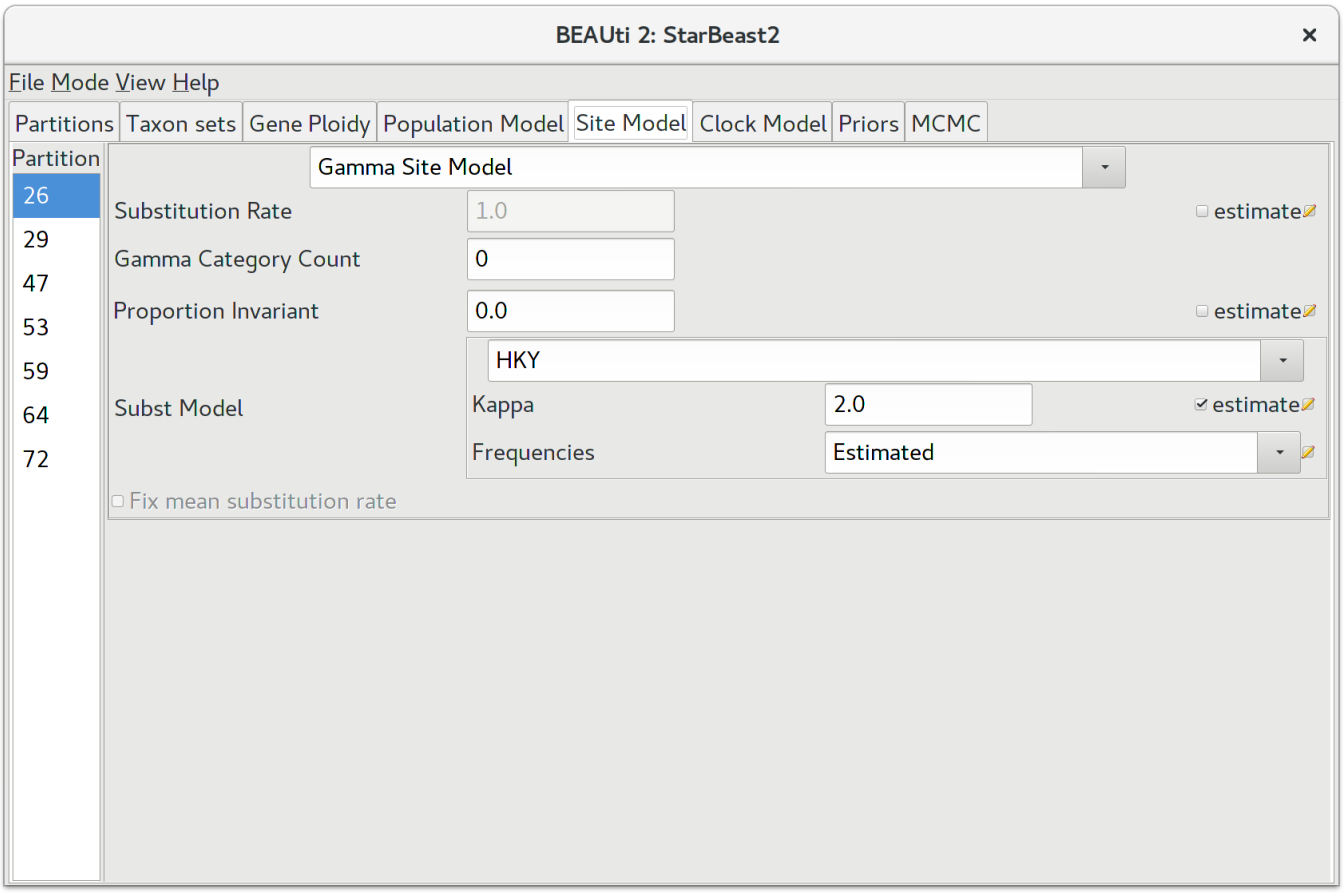

Figure 7: Setting up substitution and site models for the gopher alignments.

Setting the clock model

Click on the **Clock Model** tab at the top of the main window. In this panel you can configure the mean clock rate for each locus. If you followed the earlier instructions to disable automatic setting of clock rates, the mean clock rate "Clock.rate" of all partitions should have the **estimate** box ticked (Figure 8).

The default prior for mean clock rates in StarBEAST2 is a lognormal distribution with a mean (in real space) of 1. Trees estimated using this prior will have a time axis in units of substitutions. This will not be appropriate when using fossil (or other external) calibrations. You might instead use a 1/X prior, which is uninformative and will allow the calibration(s) to guide the clock rates.

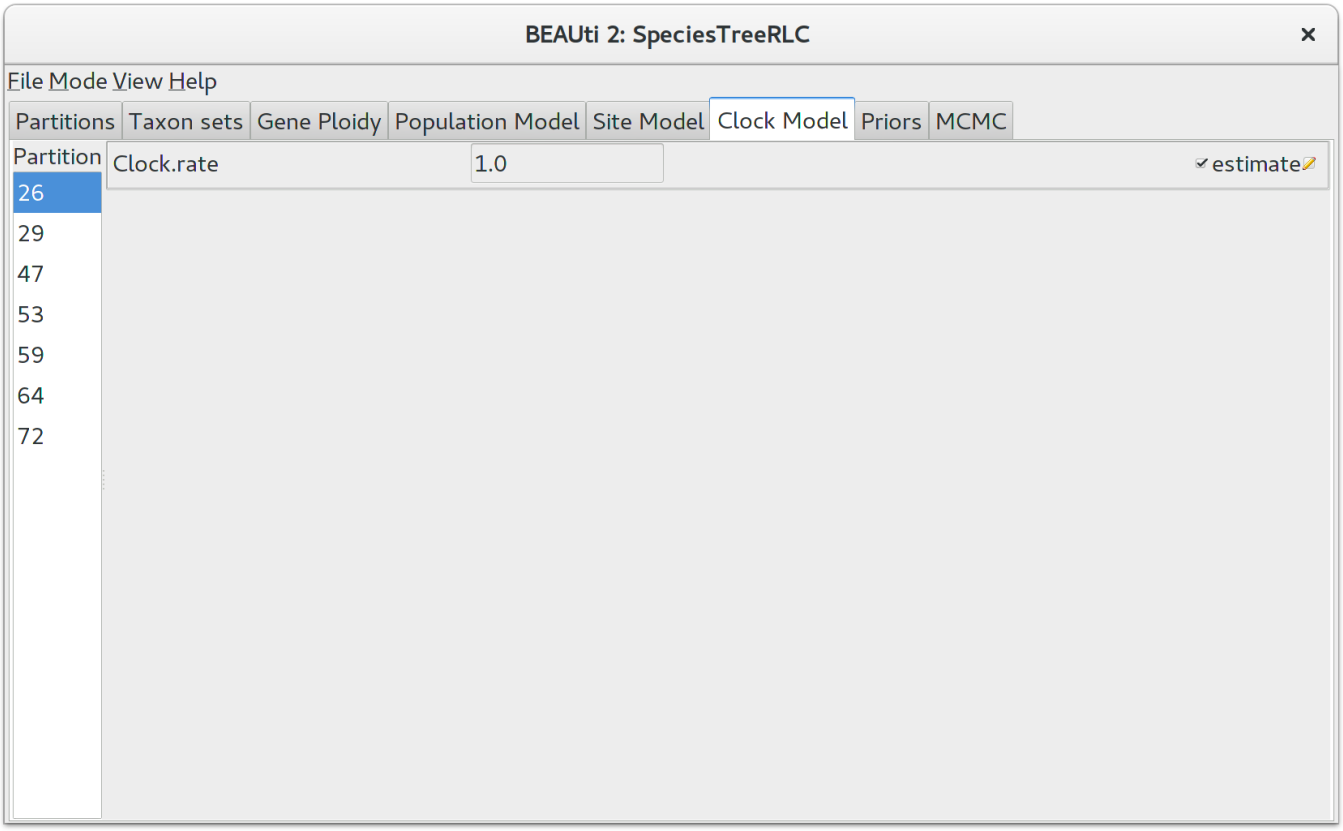

Figure 8: The default when automatic clock rate setting is disabled.

Priors

The **Priors** panel allows priors to be specified for each parameter in the model. Click the top-leftmost arrow to expand the options available for the default "Yule Model", and set the speciation rate (called for some silly historical reason "Birth Diff Rate") to 180.0. Uncheck the **estimate** box to make this a fixed parameter (Figure 9). For real analyses you should almost certainly estimate this value, but a fixed value will help us complete the tutorial in a reasonable time frame.

It would be biologically implausible for closely related species to have very large differences in substitution rates, so we should constrain the per-species branch rates to a reasonable range of values. Click the button next to "branchRates.Species" to define this range. Change "Lower" to 0.1 and "Upper" to 10.0, which means that the fastest branch rate can not be more than 100× that of the slowest branch rate (Figure 10).

Setting the MCMC options

The next tab, **MCMC**, provides more general settings to control the length of the MCMC and the file names.

First up is the **Chain Length**. This the number of steps BEAST will complete before stopping an MCMC chain. The appropriate length of the chain depends on the size of the data set, the complexity of the model and on the accuracy of the answer required. The default value of 10,000,000 is entirely arbitrary and should be adjusted according to the size of your data set. For this tutorial keep the default chain

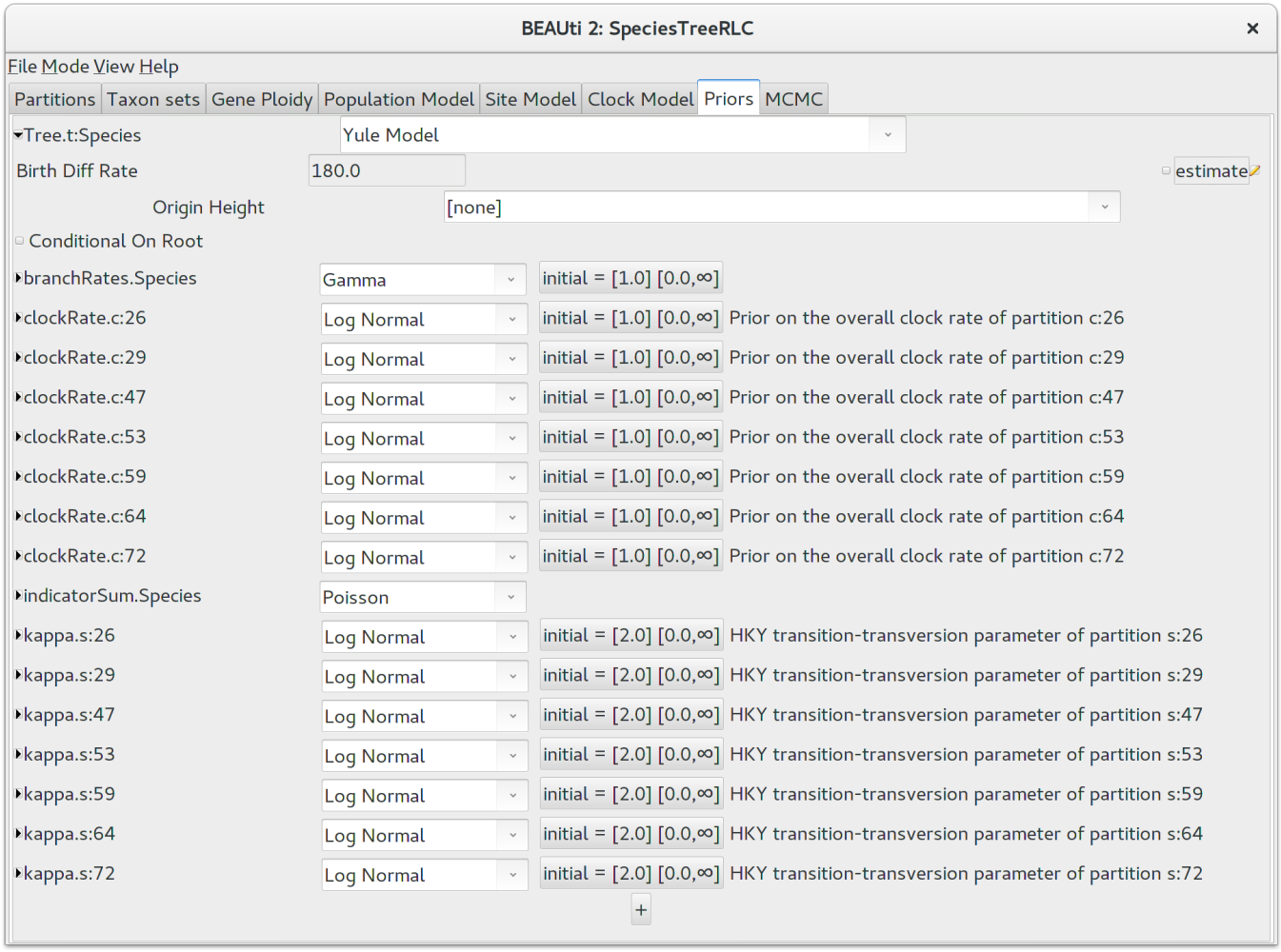

Figure 9: Fixing the speciation rate.

length, which should finish within 10 to 20 minutes on a modern computer (Figure 11).

The other options specify how the parameter values in the Markov chain should be displayed on the screen and recorded in the log file. The **screenlog** output is simply for monitoring the programs progress so can be set to any value (although if set too small, the sheer quantity of information being displayed on the screen will actually slow the program down). For the **tracelog** and **treelog** files, the value should be set relative to the total length of the chain. Sampling too often will result in very large files with little extra benefit in terms of the precision of the analysis. Sample too infrequently and the log file will not contain much information about the distributions of the parameters. You probably want to aim to store no more than 10000 samples so this should be set to no less than chain length $\div$ 10000. For this exercise, leave the default **Store Every** and **Log Every** settings in place.

If you are using Windows then we suggest you add the suffix ".txt" to the tracelog, speciesTreeLog, and other treelog file names (e.g. "starbeast.log.txt" and "species.trees.txt") so that Windows recognizes them as text files.

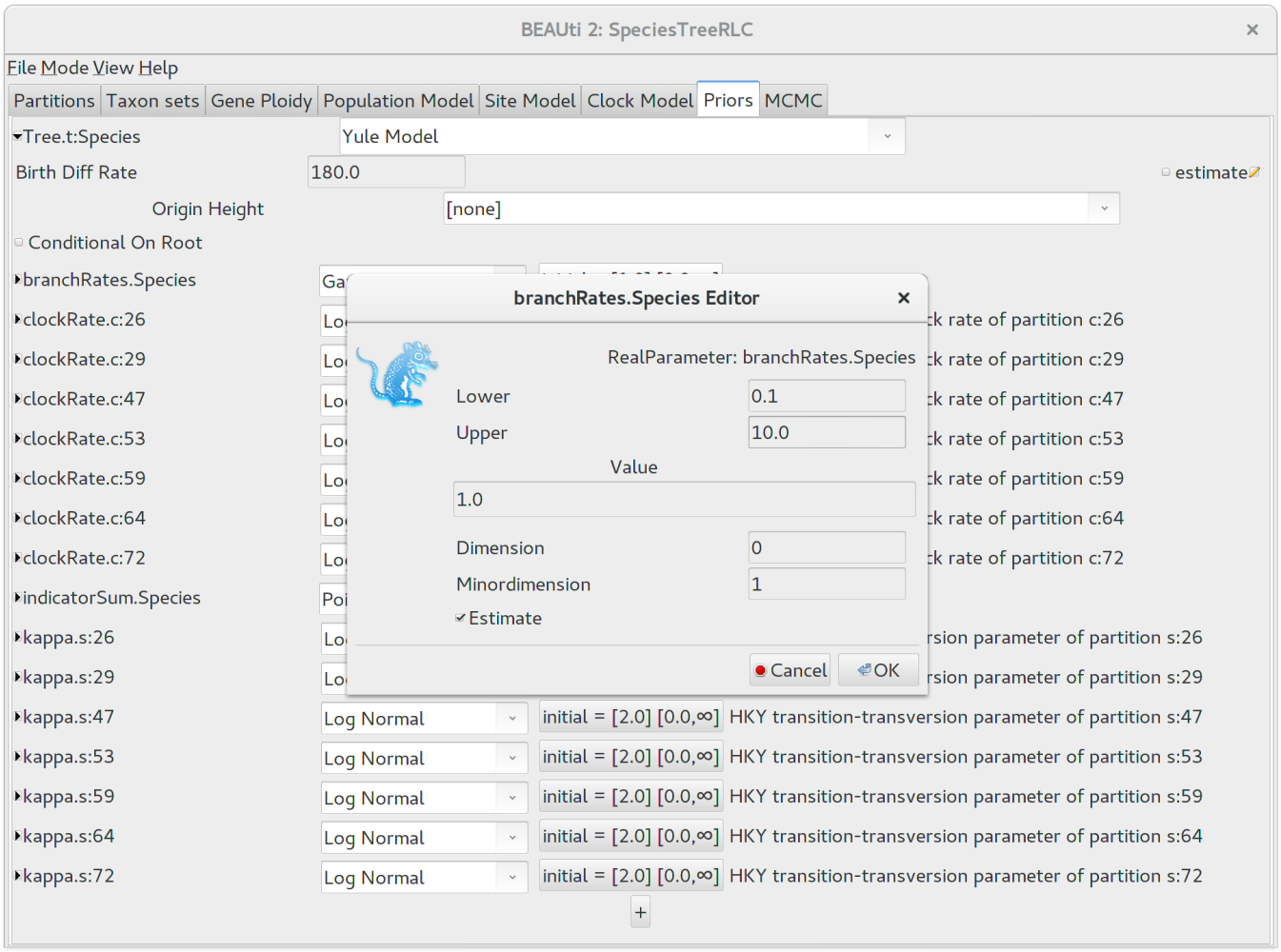

Figure 10: Setting reasonable limits on species branch rates.

Generating the BEAST XML file

We are now ready to create the BEAST XML file. To do this, either select the **File/Save** or **File/Save As** menu options. Save the file with an appropriate name (we usually end the filename with ".xml", e.g. "pocket-gophers-rlc.xml"). We are now ready to run the file through BEAST.

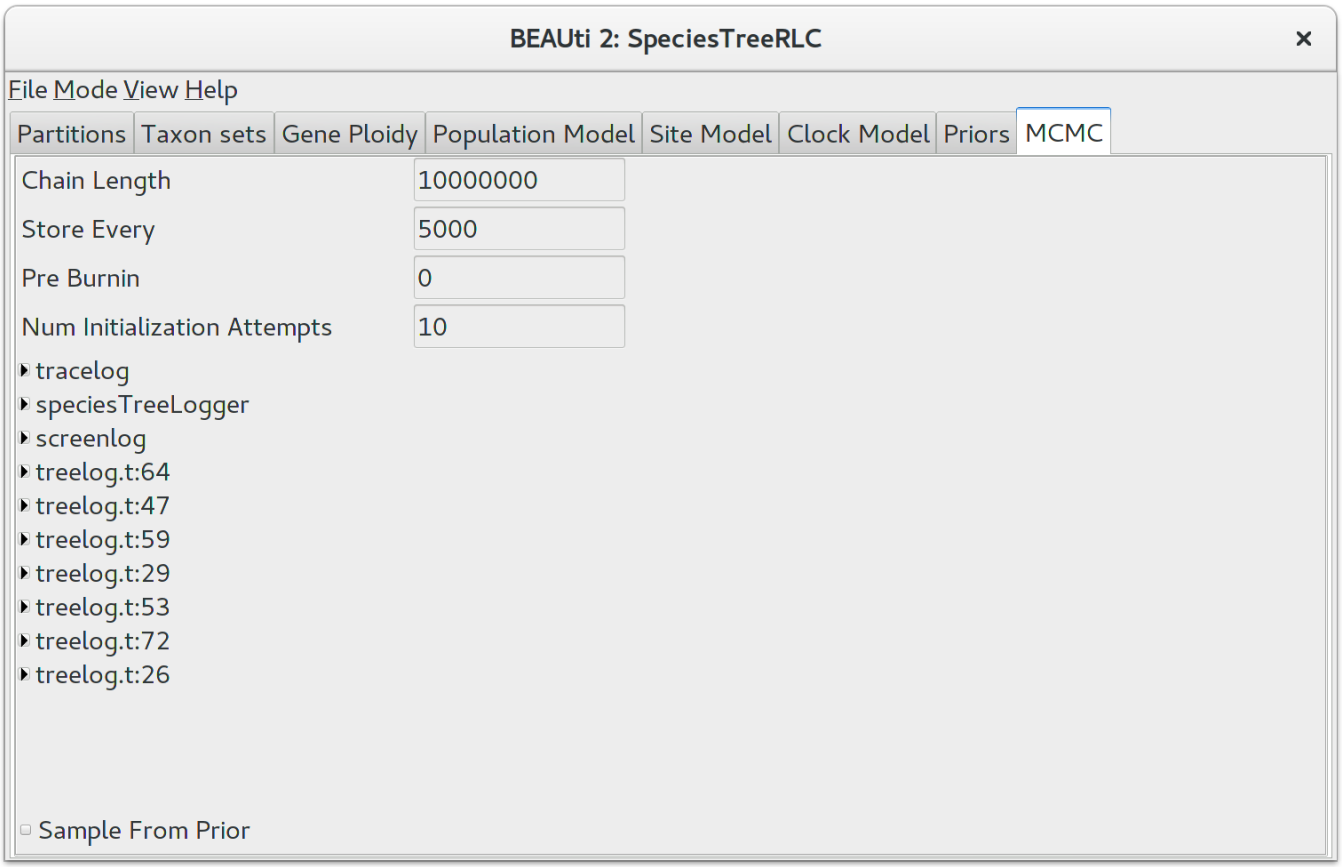

Figure 11: Setting up the MCMC paremeters.

Running BEAST

Now run BEAST and when it asks for an input file, provide your newly created XML file as input by clicking **Choose File...**, and then click **Run**. In Linux BEAST will immediately launch a file opening dialog box, which is to select the BEAST XML to run. BEAST will then run until it has finished reporting information to the screen. The actual results files are saved to the disk in the same location as your input file. The output to the screen will look something like this:

```
                BEAST v2.4.5 Prerelease, 2002-2016
            Bayesian Evolutionary Analysis Sampling Trees
                      Designed and developed by
 Remco Bouckaert, Alexei J. Drummond, Andrew Rambaut & Marc A. Suchard
...

         Sample      posterior ESS(posterior)     likelihood          prior
              0    -10073.1177              N    -10189.1746       -45.5630 --
           5000     -9577.5483         2.0        -9954.8872       -13.0309 --
...

        9995000     -9108.4194       647.2        -9813.0864        -3.5710 51s/Msamples
       10000000     -9103.9109       647.7        -9815.5796        -9.9556 51s/Msamples

Operator                                                          Tuning    #accept    #reject      Pr(m)  Pr(acc|m)
ScaleOperator(popMeanScale.Species)                               0.3812       3583       8190     0.0012     0.3043
ScaleOperator(branchRateScaler.Species)                           0.1723      56881     156125     0.0213     0.2670
BitFlipOperator(indicatorFlipper.Species)                              -       4242     101781     0.0106     0.0400
starbeast2.NodeReheight2(Reheight.t:Species)                           -     124045     761693     0.0886     0.1400
starbeast2.CoordinatedUniform(coordinatedUniform.t:Species)            -      63702     114019     0.0177     0.3584
starbeast2.CoordinatedExponential(coordinatedExponential.t:Species)    0.0028      60245     116930     0.0177     0.3400
ScaleOperator(TreeScaler.t:Species)                               0.9615       5514      29691     0.0035     0.1566
ScaleOperator(TreeRootScaler.t:Species)                           0.6990       8246      27291     0.0035     0.2320
Uniform(UniformOperator.t:Species)                                     -      21992     155918     0.0177     0.1236
SubtreeSlide(SubtreeSlide.t:Species)                              0.0019      24596     152409     0.0177     0.1390
Exchange(Narrow.t:Species)                                             -      18270     159466     0.0177     0.1028
Exchange(Wide.t:Species)                                               -       2363     174072     0.0177     0.0134
WilsonBalding(WilsonBalding.t:Species)                                 -        685     176237     0.0177     0.0039
UpDownOperator(updown.all.Species)                                0.3839      19511      51365     0.0071     0.2753
ScaleOperator(clockRateScaler.c:59)                               0.4593       7996      28057     0.0035     0.2218
UpDownOperator(clockUpDownOperator.c:59)                          0.7573       8155      27221     0.0035     0.2305
ScaleOperator(TreeScaler.t:59)                                    0.7809       8030      27602     0.0035     0.2254
ScaleOperator(TreeRootScaler.t:59)                                0.3936       8495      26738     0.0035     0.2411
Uniform(UniformOperator.t:59)                                          -     101539      75250     0.0177     0.5744
SubtreeSlide(SubtreeSlide.t:59)                                   0.0064      27799     149206     0.0177     0.1571
Exchange(Narrow.t:59)                                                  -      90635      86420     0.0177     0.5119
Exchange(Wide.t:59)                                                    -       5767     171547     0.0177     0.0325
WilsonBalding(WilsonBalding.t:59)                                      -       6033     171715     0.0177     0.0339
...

ScaleOperator(KappaScaler.s:59)                                   0.2986       3401       8449     0.0012     0.2870
...

DeltaExchangeOperator(FrequenciesExchanger.s:59)                  0.0990       3902      13818     0.0018     0.2202
...

     Tuning: The value of the operator's tuning parameter, or '-' if the operator can't be optimized.
    #accept: The total number of times a proposal by this operator has been accepted.
    #reject: The total number of times a proposal by this operator has been rejected.
      Pr(m): The probability this operator is chosen in a step of the MCMC (i.e. the normalized weight).
  Pr(acc|m): The acceptance probability (#accept as a fraction of the total proposals for this operator).

End likelihood: -9103.910915654225
```

3 Analyzing the results

Run the program called **Tracer** to analyze the output of BEAST. When the main window has opened, choose **Import Trace File...** from the **File** menu and select the file that BEAST has created called "starbeast.log". On the left hand side is a list of the different quantities that BEAST has logged. There are traces for the the various probabilities and likelihoods as well as estimates of various discrete and continuous parameters. The first and most important trace — the "posterior" — is the log of the product of gene tree phylogenetic likelihoods, the coalescent probability of gene trees within the species tree, and all prior probabilities. Selecting a trace on the left brings up analyses for this trace on the right hand side depending on tab that is selected. Select the statistic named "sum(indicators.Species)" — this is the total number of substitution rate changes along the species tree. You should now see a window like in Figure 12.

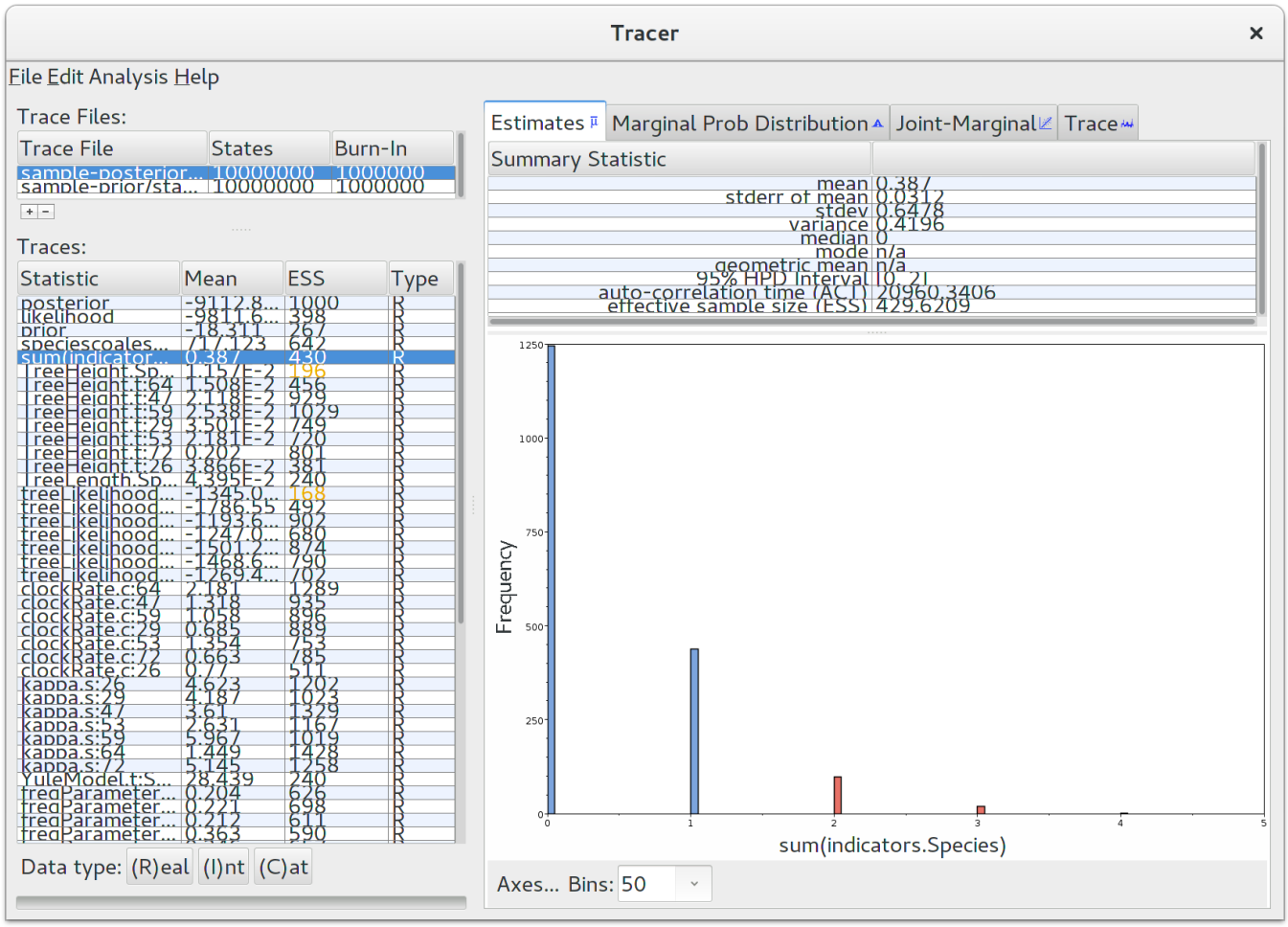

Figure 12: Tracer with the gopher data.

Remember that MCMC is a stochastic algorithm so the actual numbers will not be exactly the same. Tracer will plot a (marginalized) posterior distribution for the selected parameter and also give you statistics such as the mean and median. The "95% HPD interval" stands for *highest posterior density interval*, and represents the most compact interval on the selected parameter that contains 95% of the posterior probability. It is also known as a *credibility interval*, and can be thought of as a Bayesian analog to a confidence interval. The HPD for the sum of rate changes suggests that either 0, 1 or 2 rate changes have occured.

A note on prior distributions

For any Bayesian analysis, it is very important to compare your findings with the prior distribution. The default prior distribution for the number of substitution rate shifts for a random local clock is a Poisson distribution with the λ parameter fixed at $\ln(2) \approx 0.69$. The prior probability of zero rate changes for the default distribution is equal to 50%. Tracer reported that around 1250 samples had zero rate shifts, out of $9000000 \div 5000 = 1800$ post-burnin posterior samples. This means that after adding data, our belief in a strict clock increased from 50% to about 70%, a very modest change. The data in this tutorial suggests that a strict clock applies to pocket gophers, but falls far short of any standard of proof.

Obtaining an estimate of the phylogenetic tree

BEAST also produces a sample of plausible trees. These can be summarized using the program **TreeAnnotator**. This will take the set of trees and identify a single tree that best represents the posterior distribution. It will then annotate this selected tree topology with the mean ages of all the nodes as well as the 95% HPD interval of divergence times for each clade in the selected tree. It will also calculate the posterior clade probability for each node. Run the **TreeAnnotator** program and set it up to look like in Figure 13.

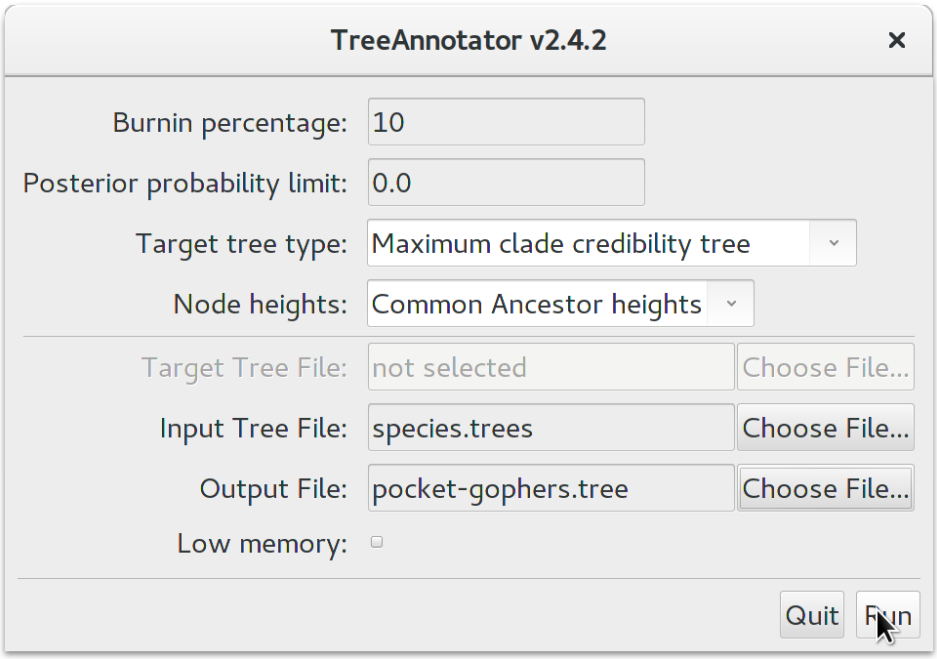

Figure 13: Using TreeAnnotator to summarise the tree set.

The **Burnin percentage** is the proportion of trees to remove from the start of the sample; for this tutorial, set a 10% burnin as shown in Figure 13.

The **Posterior probability limit** option specifies a limit such that if a node is found at less than this frequency in the sample of trees (i.e., has a posterior probability less than this limit), it will not be annotated.

For **Target tree type** you can either choose a specific tree from a file or ask TreeAnnotator to find a tree in your sample. The default option, **Maximum clade credibility tree**, finds the tree with the highest product of the posterior probability of all its nodes.

Keep "Common Ancestor heights" for **Node heights**. This sets the heights (ages) of each node in the tree to the mean height of the most recent common ancestor across the entire set of trees in the posterior.

For the input file, select the trees file that BEAST created (by default this will be called "species.trees") and select a file for the output (here we have called it "pocket-gophers.tree"). Now press **Run** and wait for the program to finish.

Viewing the species tree(s)

Finally, we can look at the tree in another program called **FigTree**. Run this program, and open the "pocket-gophers.tree" file by using the Open command in the File menu. The tree should appear. You can now try selecting some of the options in the control panel on the left. Try selecting **Node Bars** to get node age error bars. Turn on **Node Labels** and select "posterior" to get it to display the posterior probability for each node, and also turn on **Branch Labels** and select "rate_95%_HPD" to display the 95% HPD of the relative substitution rate for each species tree branch. You should end up with something like Figure 14.

Notice that the HPD interval for per-species substitution rates all include 1.0, concordant with our previous observation that there may be no changes to the overall substitution rate along this species tree.

As a more Bayesian alternative to FigTree, you can load the entire species tree set into DensiTree. Open the "species.trees" file in DensiTree and set up the cloudogram as follows:

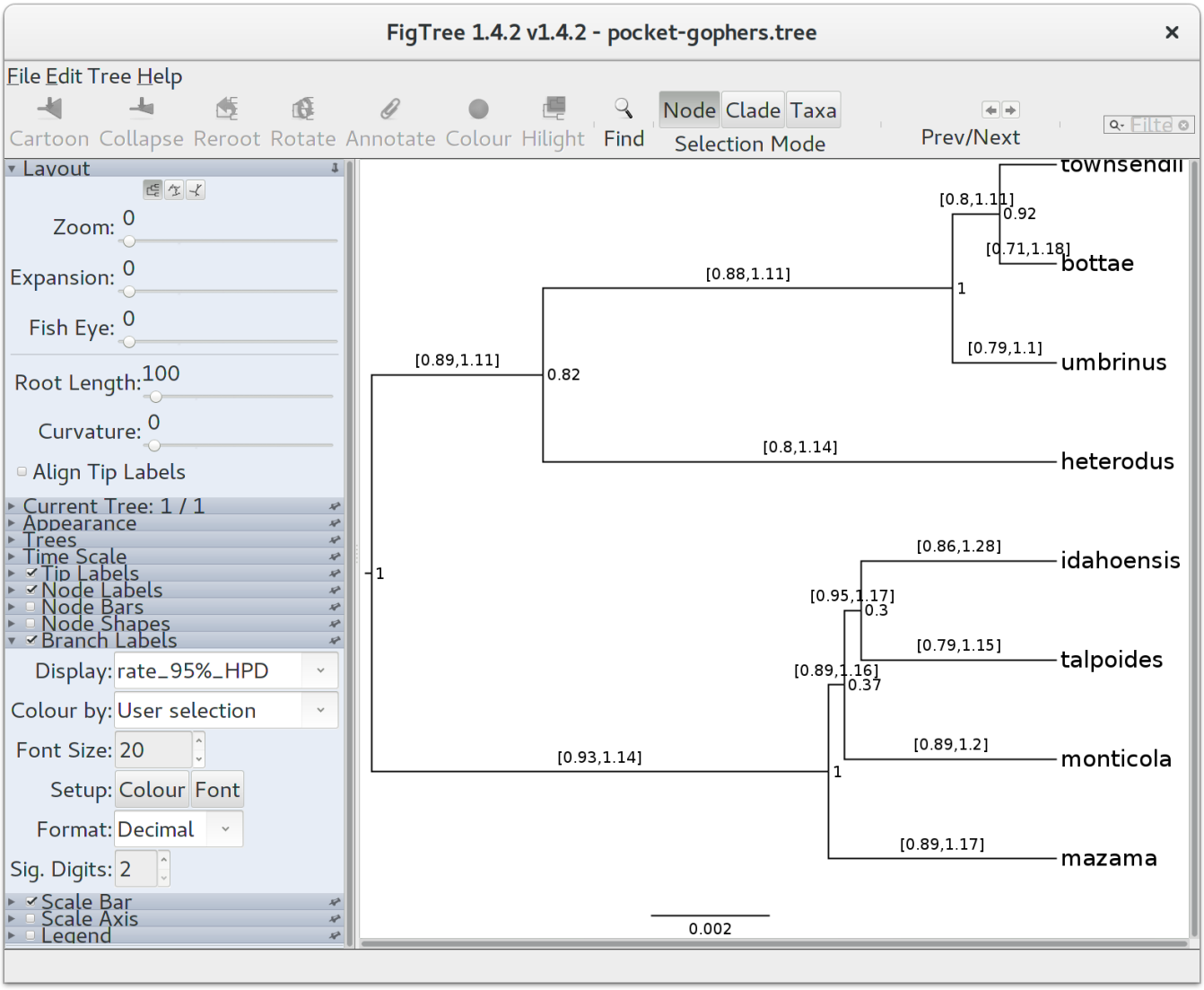

Figure 14: Figtree representation of the species tree.

- Select the **Central** geometry from the set of options in the top-right of the main screen.
- Under **Show**, check the "Root Canal" tree to guide the eye.
- Under **Clades**, check "Show Clades", display the means and 95% HPDs using "draw", and display the posterior support using "text".
- Now, too many clades are shown, and most are not of interest. Check "Selected only", then open the clade toolbar using the **Window/View clade toolbar** menu item. Select each clade (i.e. items with more than one species) with majority posterior support (i.e. with over 50% support) by using the shift key.

The image should look something like Figure 15. Notice that there is about 16% support for *heterodus* to be an outgroup, and about 82% for heterodus to be in a clade with *bottea*, *umbinus* and *townsendii*. Can you explain where the remaining 2% went?

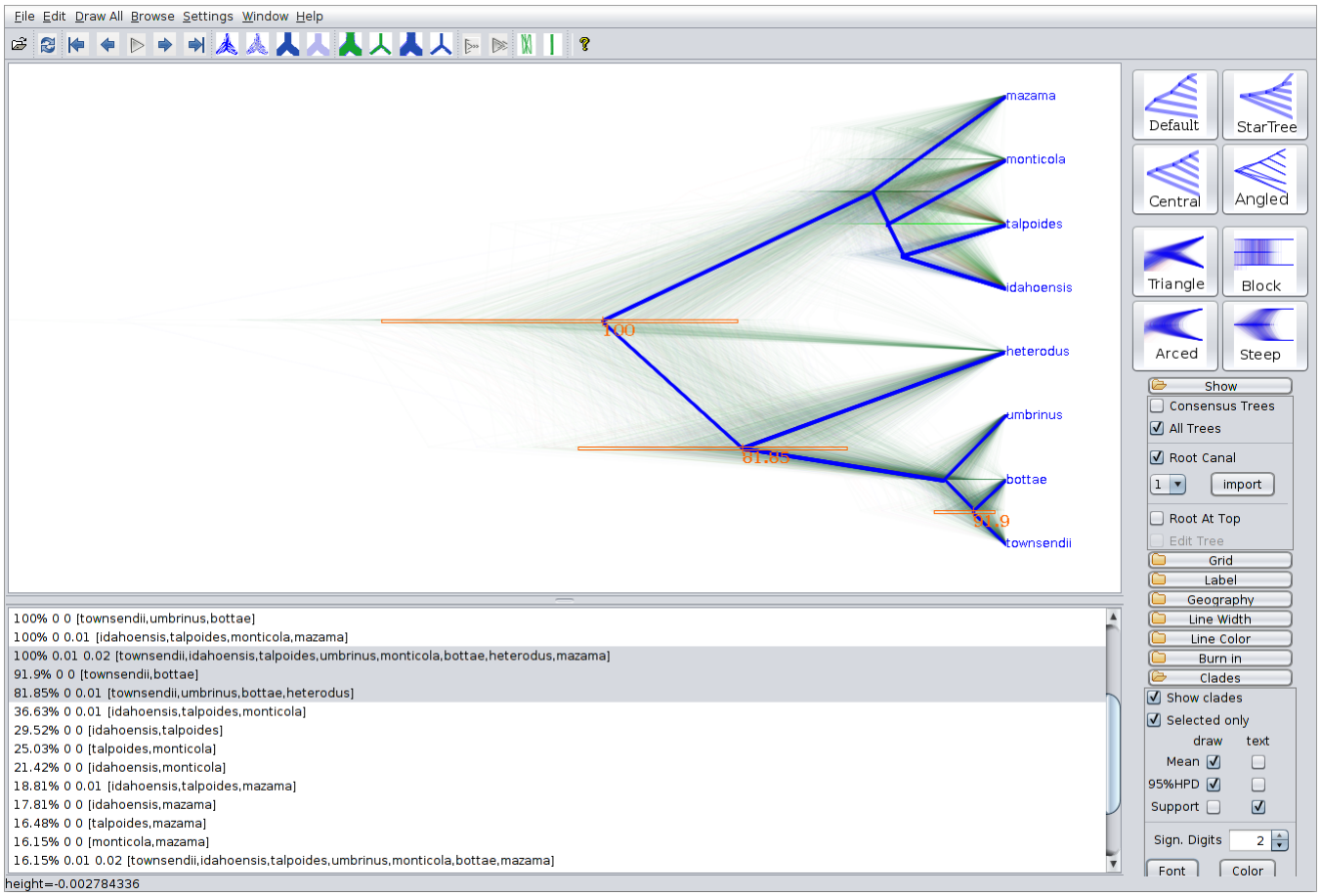

Figure 15: DensiTree representation of the species tree.

Version dated: August 28, 2018

Relevant References

Belfiore, N, L Liu, and C Moritz. 2008. Multilocus phylogenetics of a rapid radiation in the genus thomomys (rodentia: geomyidae). *Systematic Biology* 57: 294.

Bouckaert, R, J Heled, D Kühnert, T Vaughan, C-H Wu, D Xie, MA Suchard, A Rambaut, and AJ Drummond. 2014. Beast 2: a software platform for Bayesian evolutionary analysis. *PLoS Computational Biology* 10: 1–6.

Heled, J and AJ Drummond. 2010. Bayesian inference of species trees from multilocus data. *Molecular Biology and Evolution* 27: 570–580.