Species Trees and Species Delimitation with SNAPP: A Tutorial and Worked Example

Adam D. Leaché
Department of Biology, University of Washington, Seattle, United States
Burke Museum of Natural History and Culture, Seattle, United States

Remco R. Bouckaert
Centre for Computational Evolution, University of Auckland, Auckland, New Zealand
Max Planck Institute for the Science of Human History, Jena, Germany

Updated for the 2018 Workshop on Population and Speciation Genomics by Huw Ogilvie, who you should blame for anything that is wrong or that doesn't make sense...

Contents

1 Objective

This tutorial will help you become familiar with conducting species tree inference and species delimitation in a Bayesian framework using biallelic markers (AFLP or SNP data) with `SNAPP`. You be using example SNP data from *Hemidactylus* geckos with the implementation of `SNAPP` available for the `BEAST 2` platform (Bouckaert et al., 2014). The tutorial includes instructions for installing the required packages on your computer, setting up the XML file, and testing species delimitation models using marginal likelihood estimation and Bayes factors.

A secondary goal is to provide guidance on how to make informed choices concerning the priors and other settings in `SNAPP`. It is deceptively easy (and often enticing) to run the program with default settings and to ignore the biological meanings of the various priors and settings, but this is dangerous and typically leads to inaccurate results. Comparing marginal likelihood estimates obtained from “default” versus “realistic” settings can be dramatic, and can lead to different rankings of species delimitation models (even in cases where the species tree topologies look similar).

2 Version, Author information, and Acknowledgements

This tutorial was mostly written by Adam Leaché and Remco Bouckaert. David Bryant, Jamie Oaks, and CJ Battey helped troubleshoot the tutorial. The layout of the tutorial is a modified version of a divergence time tutorial written by Jamie Oaks, which he borrowed from Tracy Heath. This work is licensed under a Creative Commons Attribution 4.0 International License.

3 Background Information

Most coalescent methods for estimating species trees are multilocus methods, which explicitly model a separate gene tree for each locus. Methods such as `StarBEAST2` (Ogilvie et al., 2017) and `BPP` (Rannala and Yang, 2017) jointly infer the species tree and the coalescent history of each gene, but are difficult to scale up for use with hundreds or thousands of loci. An alternative is to use a method such as `ASTRAL` (Mirarab et al., 2014; Mirarab and Warnow, 2015; Zhang et al., 2017) which uses previously computed gene tree topologies as input, but this class of methods cannot be used to estimate species divergence times, population sizes or to infer the root absent an outgroup.

`SNAPP` is a method which estimates species trees directly from biallelic markers (e.g., SNP or AFLP data), bypassing the necessity of having to explicitly integrate or sample the gene trees at each locus. The method works by estimating the probability of allele frequency change across ancestor/descendent nodes. The result is a posterior distribution for the species tree, species divergence times, and effective population sizes, all obtained without the estimation of gene trees. The method works well for relatively small numbers of species (the maximum is probably near 20 due to computational constraints).

Incorporating coalescent analysis into studies of species delimitation is now standard practice. Comparing candidate species delimitation models that contain different numbers of species, or different allocations of populations to species, is relatively easy in a Bayesian framework. The general approach is to estimate the marginal likelihood (Baele et al., 2012) of each competing species delimitation model, rank models by marginal likelihood, and use Bayes factors (Kass and Raftery, 1995) to assess support for model rankings. This approach, called Bayes factor delimitation (BFD), was first implemented by Grummer et al. (2013) with DNA sequences in the program `*BEAST`. The approach was modified to work with SNP data (BFD*) using the program `SNAPP` (Leaché et al., 2014).

BFD* estimates the species tree and evaluates the species delimitation model at the same time, while allowing the user to compare models that contain different numbers of species and different assignments of samples to species. This is useful when the goal is to compare predefined species delimitation models or competing taxonomies. However, one drawback is that the user needs to predefine the number of species and sample assignments. This prevents the method from searching among all possible species assignments, an obvious disadvantage for studies aiming to discover cryptic diversity. Another major limitation is that the method does not explicitly consider gene flow, isolation by distance, selection, or several other important biological processes; however, these limitations are shared by many current methods. For example, failing to sample admixed populations often favors models containing more species, whereas including admixed populations will support more models containing fewer species. Distinguishing among these problematic scenarios requires paying close attention to both sample selection and prior settings. Finally, when evaluating results, remember to consider other aspects of the biology, ecology, and geography of "species" before jumping to conclusions.

4 Programs Used in This Tutorial

You will use the free, open-source phylogenetics platform BEAST 2 (Bayesian Evolutionary Analysis Sampling Trees; http://beast2.org), for estimating species trees. BEAST comes with several utility programs including BEAUTi, which you will use to manage package plugins (also called add-ons), and TreeAnnotator, for summarizing output files. You will also be using the programs Tracer (http://tree.bio.ed.ac.uk/software/tracer) and FigTree (http://tree.bio.ed.ac.uk/software/figtree) for evaluating, summarizing, and viewing results.

5 The Data

You will be analyzing SNP data for geckos in the *Hemidactylus fasciatus* species complex. Details on how the data were collected are provided in Leaché et al. (2014). For this tutorial, we will use a data matrix containing 129 SNPs that is also available for download on Dryad. Allopatric divergence seems to be the primary mechanism causing speciation in this group. These geckos are restricted to rainforest habitats, and their distributions match those of the major blocks of rainforest in West and Central Africa (Figure 1).

For this species delimitation example, you will test models based on historical connections between adjacent rainforest blocks. These models differ in the number of species, and how samples are assigned to species. The base model has four species (Figure 1a). The alternative models are grouped into three classes: (1) lumping: populations are collapsed into the same species, (2) splitting: populations are partitioned into separate species, (3) reassigning: population(s) are allocated into a different species.

6 Tutorial

6.1 Downloading the programs

Step 1: Download BEAST from http://beast2.org and install it on your computer. This tutorial is written for the Mac OS X version of BEAST 2.4.5 or greater. Please make sure to keep BEAST and its packages up to date – at the time of writing, the most recent version is 2.4.7.

You will be using BEAST to run SNAPP, although it is possible to run SNAPP on its own. However, you have to use BEAST in order to combine SNAPP and marginal likelihood estimation into the same analytical framework. Thus, without BEAST you would not be able to conduct species delimitation with SNAPP.

Step 2: After downloading and unzipping this archive you should have a "BFD" tutorial folder on your computer. This tutorial contains the files and folders shown in Box 1. The ***data*** folder contains the gecko SNP data in binary format (necessary for SNAPP). If you are unsure of how to convert your own SNP data from nucleotide to binary format, please read the documentation A rough guide to SNAPP (Section 4. Preparing Input File). You can find scripts for converting SNP data into SNAPP input format at the phrynomics project

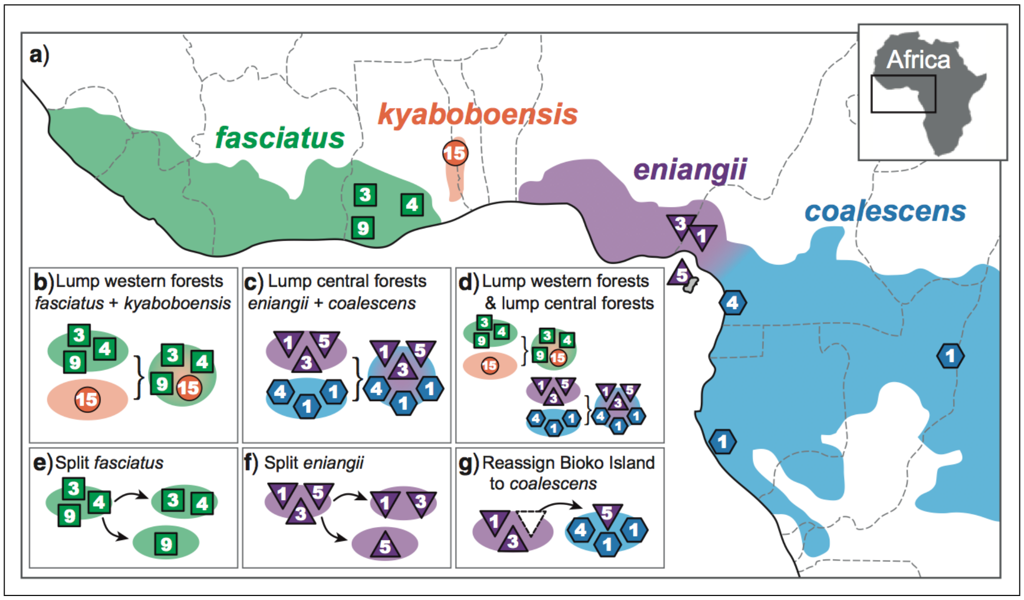


Figure 1: Geographic sampling of geckos (numbers in symbols indicate sample sizes). Starting taxonomy is shown in (a). BFD* is used to test the alternative species delimitation models outlined in (b) – (g).

site at GitHub. You can also find help at the `BEAST` Google users group. The ***xml*** folder contains seven xml files (named according to the species delimitation models in Figure 1) that are ready to run in `BEAST`.

- `BFD/`
  - `BFD-tutorial.pdf`
  - `data/`
    - `hemi129.nex`
    - `smallhemi129.nex`
  - `xml/`
    - `RunA.xml`
    - `RunB.xml`
    - `RunC.xml`
    - `RunD.xml`
    - `RunE.xml`
    - `RunF.xml`
    - `RunG.xml`

Box 1: The files included in this tutorial. The data folder contains the SNP data in binary format. Ready-to-run XML files are included in the xml folder.

6.2 Setting up the XML file with `BEAUTi`

Step 3: Begin by launching the `BEAUTi` program that comes with `BEAST`. If you are using Mac OS or Windows, you should be able to do this by double clicking on the application. On Linux, open a terminal and `cd` into the extracted BEAST folder, then launch BEAUTi using the command `bin/beauti`. If everything is working correctly, a window should appear that looks something like Figure 2.

Step 4: You need to add functionality to `BEAST` in order to estimate species trees with SNP data and to perform model selection. Begin by using the drop-down menu ***File*** →***Manage Packages.*** A window should appear that looks something like Figure 3. Select and install the packages **SNAPP** and **Model_Selection**.

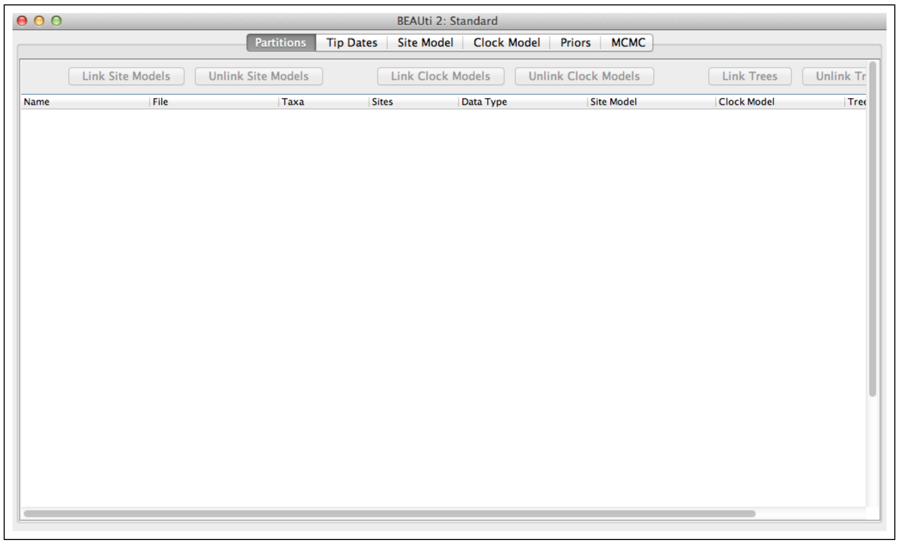
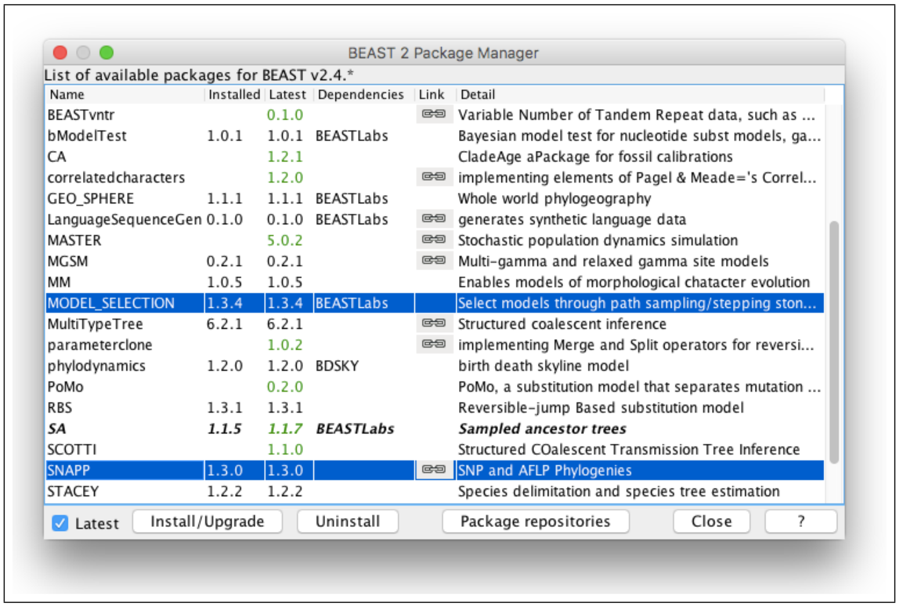


Figure 2: BEAUTi window launched from `BEAST`.

You can then exit the window by clicking the **"Close"** button.



List of available packages for BEAST v2.4.*

Name	Installed	Latest	Dependencies	Link	Detail
BEASTvntr		0.1.0		🔗	Variable Number of Tandem Repeat data, such as ...
bModelTest	1.0.1	1.0.1	BEASTLabs		Bayesian model test for nucleotide subst models, ga...
CA		1.2.1			CladeAge aPackage for fossil calibrations
correlatedcharacters		1.2.0		🔗	implementing elements of Pagel & Meade='s Correl...
GEO_SPHERE	1.1.1	1.1.1	BEASTLabs		Whole world phylogeography
LanguageSequenceGen	0.1.0	0.1.0	BEASTLabs	🔗	generates synthetic language data
MASTER		5.0.2		🔗	Stochastic population dynamics simulation
MGSM	0.2.1	0.2.1		🔗	Multi-gamma and relaxed gamma site models
MM	1.0.5	1.0.5			Enables models of morphological chatacter evolution
MODEL_SELECTION	1.3.4	1.3.4	BEASTLabs		Select models through path sampling/stepping ston...
MultiTypeTree	6.2.1	6.2.1		🔗	Structured coalescent inference
parameterclone		1.0.2		🔗	implementing Merge and Split operators for reversi...
phylodynamics	1.2.0	1.2.0	BDSKY		birth death skyline model
PoMo		0.2.0			PoMo, a substitution model that separates mutation ...
RBS	1.3.1	1.3.1			Reversible-jump Based substitution model
SA	***1.1.5***	***1.1.7***	***BEASTLabs***		***Sampled ancestor trees***
SCOTTI		1.1.0			Structured COalescent Transmission Tree Inference
SNAPP	1.3.0	1.3.0		🔗	SNP and AFLP Phylogenies
STACEY	1.2.2	1.2.2			Species delimitation and species tree estimation

Figure 3: `BEAUTi` package manager for `BEAST`.

Step 5: Tell `BEAUTi` that you are setting up a `SNAPP` analysis, which will change the menu options and allow us to import SNP data. Begin by using the drop-down menu ***File*** →***Template, SNAPP.*** This should change the appearance of the `BEAUTi` window to look something like Figure 4.

Step 6: Import the SNP data (the **smallhemi129.nex** file) using the drop-down menu ***File*** →***Import Alignment*** or ***File*** →***Add Alignment*** (the exact name depends on your operating system). Once the data are successfully loaded into `BEAUTi` you should see a list of the samples included in the data file (Figure 5.)

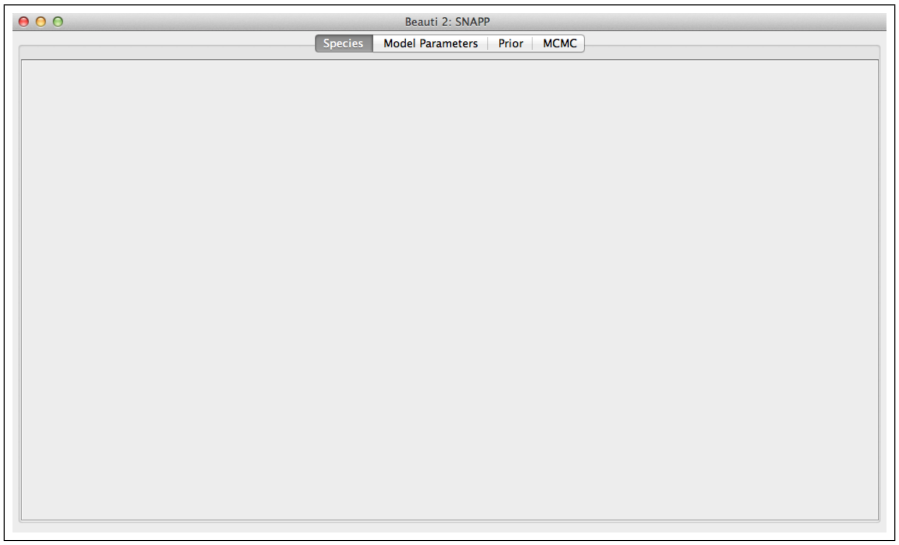
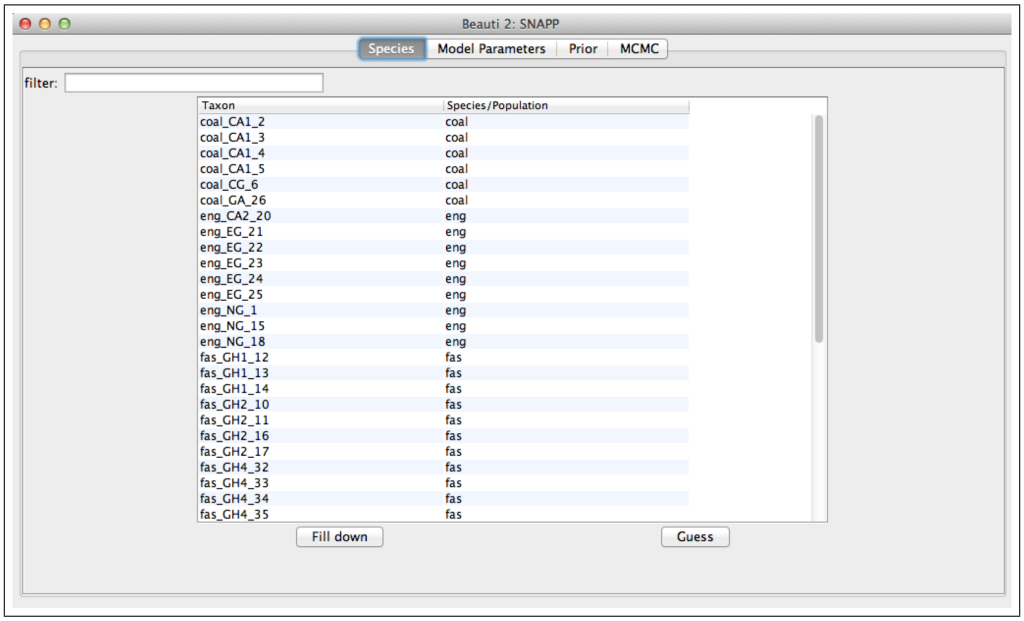


Figure 4: BEAUTi window after importing the SNAPP template. Notice that the menu tabs have changed.

Figure 5 shows the complete set of samples if you load the full data set from **hemi129.nex**, but we recommend loading the small data set because otherwise you might not finish this tutorial in the allocated time frame. In a real analysis, how many samples should you include? The number of samples slows down SNAPP much more than the number of SNPs. Therefore, if your analyses are going too slow, then it is typically better to randomly subsample down to an even proportion of sequences from each species instead of removing SNPs. Reducing the number of samples by half will more than halve the analysis time. When setting up a new analysis, start with a small number of samples for each species (for example, 4 samples per species), which will enable you to make quicker progress. Increase the number of samples once your analyses are returning reasonable results. If you are overambitious with your sampling, then your analyses will become unbearably slow.



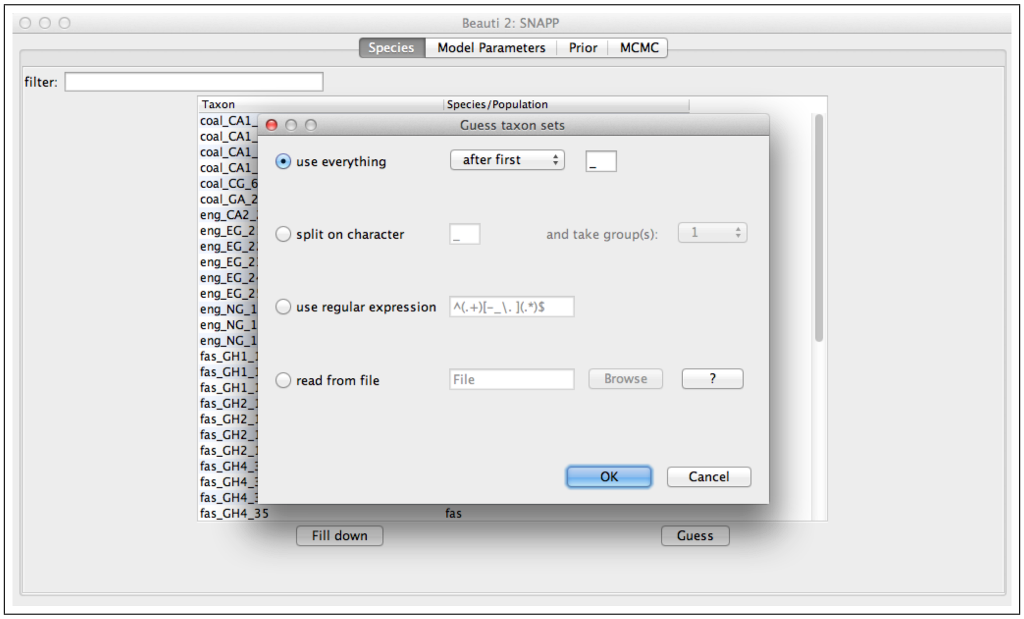
Taxon	Species/Population
coal_CA1_2	coal
coal_CA1_3	coal
coal_CA1_4	coal
coal_CA1_5	coal
coal_CG_6	coal
coal_GA_26	coal
eng_CA2_20	eng
eng_EG_21	eng
eng_EG_22	eng
eng_EG_23	eng
eng_EG_24	eng
eng_EG_25	eng
eng_NG_1	eng
eng_NG_15	eng
eng_NG_18	eng
fas_GH1_12	fas
fas_GH1_13	fas
fas_GH1_14	fas
fas_GH2_10	fas
fas_GH2_11	fas
fas_GH2_16	fas
fas_GH2_17	fas
fas_GH4_32	fas
fas_GH4_33	fas
fas_GH4_34	fas
fas_GH4_35	fas

Figure 5: The data successfully loaded by BEAUTi.

Step 7: There are several ways to designate species assignments. You can automatically designate species names using the names already present in the data files. The species names can be pre-defined this way by including a "delimiter" that allows the species name to be parsed from the rest of the sequence name. The gecko data file uses an underscore "_" to separate the species name (on the left) from the rest of the sequence name (on the right) as follows:

```
eng_NG_1
coal_CA1_2
coal_CA1_3
coal_CA1_4
coal_CA1_5
coal_CG_6
kya_GH3_7
kya_GH3_8
...
```

Several options for assigning species names are available using the "Guess" button. The screen should look similar to Figure 6. You can even import a custom mapping file that links each sample to a species using the "read from file" option.



fas_GH4_35 fas

● use everything after first _

○ split on character _ and take group(s): 1

○ use regular expression ^(.+)[-_\.](.*)$

○ read from file File

Figure 6: The species assignment options that appears after you select the "Guess' button.

To extract the species names in this tutorial, keep "use everything" selected and leave the underscore in the text box, but change "after first" to **"before first"**. Click the "Ok" button to return to the ***Species*** window. Be sure that each Taxon has a Species/Population name.

Step 8: Next, set up our model under the ***Mutation Parameters*** tab (Figure 7). Be sure to read the documentation A rough guide to `SNAPP` to learn more about the model options. Briefly, the parameters are as follows:

```
Mutation Rate U: instantaneous rate of mutating from the 0 allele to the 1 allele.
Mutation Rate V: instantaneous rate of mutating from the 1 allele to the 0 allele.
Coalescence Rate:  population size parameter with one value for each node in the tree.
```

Recommendations: Set mutation rates u and v = 1, and disable sampling by unchecking the "Sample" box adjacent to the u parameter. For SNP data where the "0" and "1" alleles are arbitrarily assigned from the data, it usually makes no sense to uncouple these rates.

Alternatively – if you know what you are doing! – you can click the **"Calc mutation rates"** button to get a direct estimate of u and v. Either way, you typically do not have to estimate these parameters during the MCMC. Coalescent Rate: make sure the Sample box is ticked so that MCMC sampling of this parameter is enabled. If you do not sample, then you assume that all population sizes are the same, which is unrealistic. The coalescent rate is 2/theta, and the number is simply the starting value used to initialize the analysis. Do not confuse the coalescent rate with the theta prior. The theta prior is described in detail in the next section.

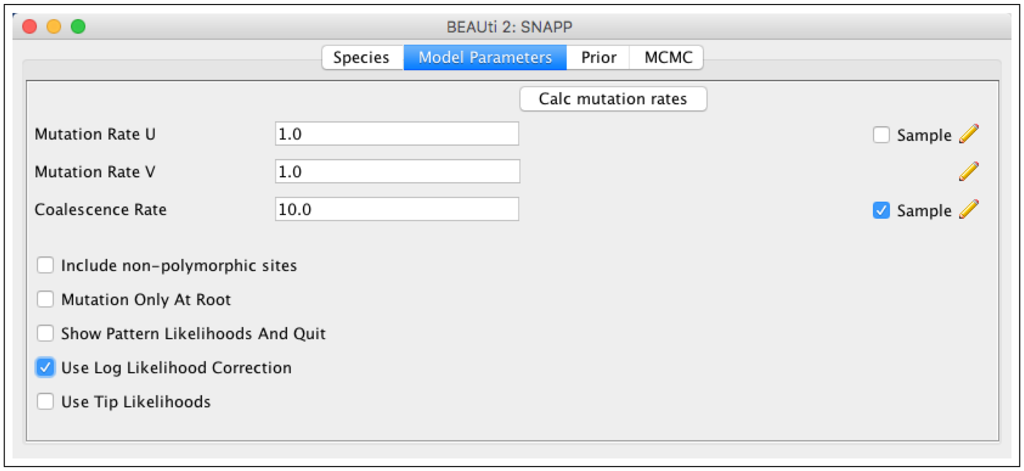
The "Include non-polymorphic" checkbox is used in cases where invariant sites have been included in the data. The likelihood calculations are different if SNAPP assumes that all constant sites have been removed. The example dataset for this tutorial, like typical SNP datasets, only includes variable site so make sure that the box is not checked.

The "Mutation Only At Root" checkbox indicates conditioning on zero mutations, except at root (default false). As a result, all gene trees will coalesce in the root only, and never in any of the branches. This option allows you to emulate the model used by Nielsen (1998) and RoyChoudhury et al. (2008).

The "Show Pattern Likelihoods And Quit" checkbox is handy if you just want to print out the likelihoods for all patterns in the starting state and then quit.

The "Use Log Likelihood Correction" checkbox is for calculating corrected likelihood values for Bayes factor test of different species assignments (the calculation is almost instantaneous, and it will not slow down your analysis). This is a species assignment and delimitation tutorial so make sure this box is checked!

Unless things break, leave "Use Tip Likelihoods" and "Implementation" at their default settings.



Mutation Rate U 1.0 □ Sample

Mutation Rate V 1.0

Coalescence Rate 10.0 ☑ Sample

□ Include non-polymorphic sites

□ Mutation Only At Root

□ Show Pattern Likelihoods And Quit

☑ Use Log Likelihood Correction

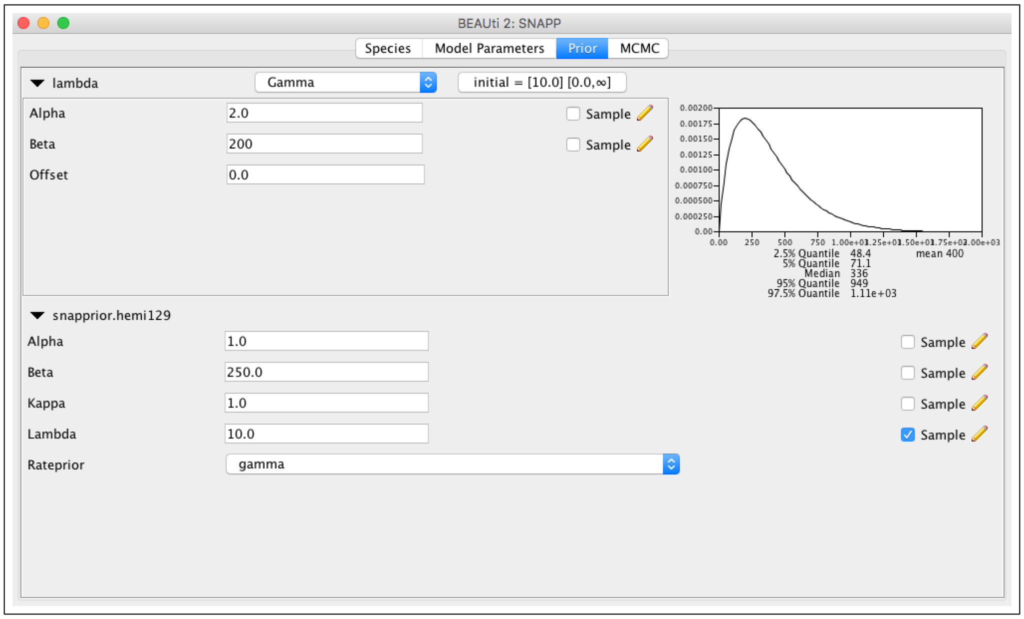
□ Use Tip Likelihoods

Figure 7: The Mutation Parameters options.

Step 9: Next, move to the ***Prior*** tab to specify the priors. Again, read the documentation A rough guide to SNAPP to learn more about these priors. It is important to be aware of the biological meaning of these priors. One problem with SNAPP is that it is deceptively easy to set up an analysis using default options, but those defaults are almost certainly inappropriate for your particular study.

Lambda (or λ) refers to the speciation rate in the Yule model. The default prior on lambda of 1/X is uninformative and improper. This is not necessarily bad if you truly have no prior knowledge of what this rate should be. However, improper priors **cannot** be used for Bayes factor analyses, and so must be changed for this tutorial. Species of the genus *Hemidactylus* seem to bifurcate every 0.0025 substitutions per site or so, so change the prior on lambda to a Gamma distribution, and set the beta scale parameter to 200. This corresponds to a mean of 400, or 400 speciation events for every substitution per site of tree length (Figure 8). The alpha shape parameter of 2 corresponds to a modal but broad distribution, so we are not letting this prior swamp the signal present in the data. Change the initial value of lambda—under "snapprior"—to a more realistic initial value like 10. The initial value won't change your results, but it may speed up the MCMC a little.

Rateprior sets the prior distribution used for theta. Recall that for a diploid population, theta = 4Nu, where N is the effective population size and u is the per-generation mutation rate. If theta=0.004, you expect to observe 0.4%



lambda Gamma initial = [10.0] [0.0,∞]

Alpha 2.0 □ Sample

Beta 200 □ Sample

Offset 0.0

2.5% Quantile 48.4 mean 400
5% Quantile 71.1
Median 336
95% Quantile 949
97.5% Quantile 1.11e+03

snapprior.hemi129

Alpha 1.0 □ Sample

Beta 250.0 □ Sample

Kappa 1.0 □ Sample

Lambda 10.0 ☑ Sample

Rateprior gamma

Figure 8: Using a gamma prior for Lambda.

variation between two randomly sampled alleles in a population. Another way to think about this is in the expected number of substitutions; "theta=0.004" means that for two randomly sampled phased sequences within a population you expect to observe 4 SNPs in 1,000 bases.

The "snapprior" parameters alpha, beta and kappa are contextually dependant on the distribution chosen for Rateprior. Using the default gamma distribution the kappa paremeter is ignored, and the alpha and beta parameters correspond to the shape and *rate* of the distribution (rather than the default shape and *scale* used elsewhere in BEAUTi). When using a gamma distribution here, the mean of the prior on theta will therefore be alpha ÷ beta.

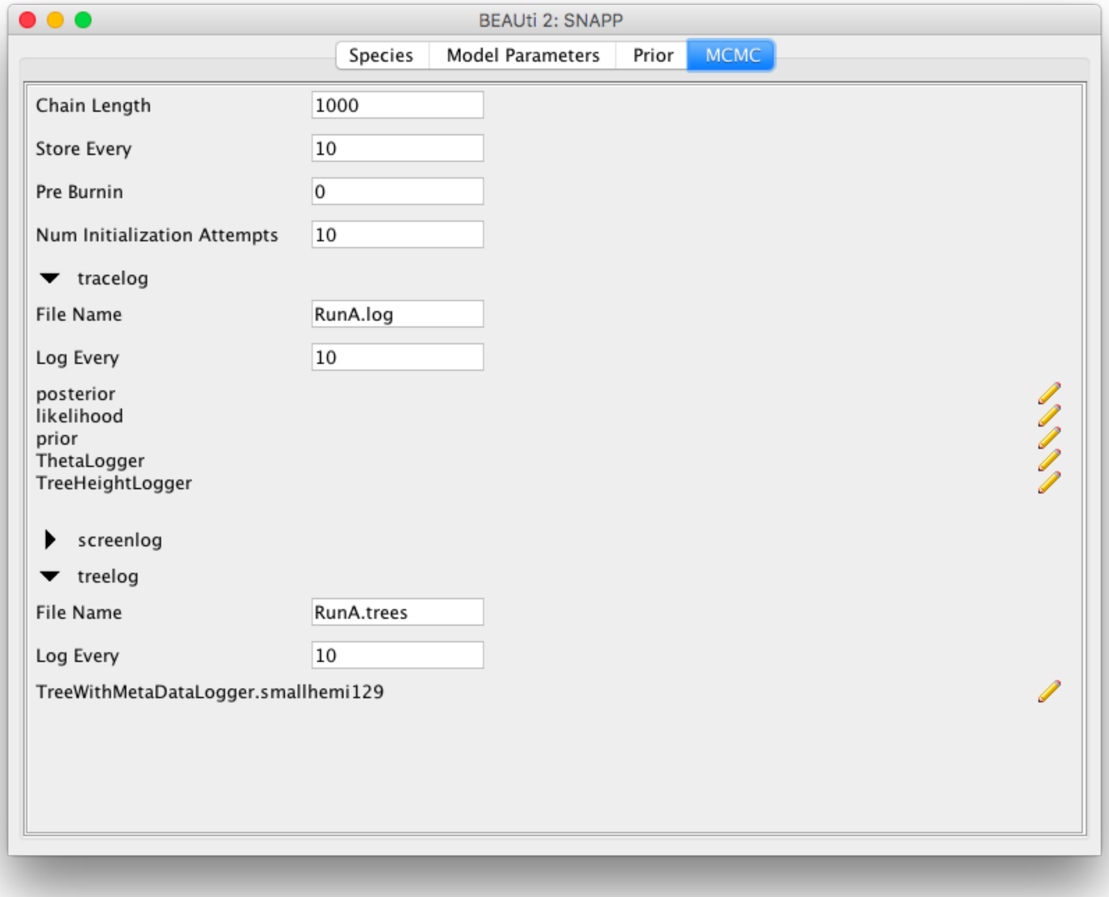
For this tutorial, set alpha to 1 and beta to 250. This corresponds to a decaying distribution with a mean of 0.004, or an expectation of 4 heterozygous sites per 1,000 bases, a reasonable value for the data set in this tutorial. As with the speciation rate, the posterior values of theta will be strongly influenced by the data.

Step 10: Next, move to the ***MCMC*** tab. Change the following settings:

```
Chain Length:   1000
Store Every:   10
tracelog:File Name:   RunA.log
tracelog:Log Every:   10
screenlog:Log Every:   10
treelog:File Name:   RunA.trees
treelog:Log Every:   10
```

Leave the remaining options at their default values (Figure 9). These MCMC values are way to low, and a thorough analysis requires much more computational time. The MCMC run times are intentionally kept short (and the data files reduced) in this tutorial. These short analyses should run in approximately 2 – 4 minutes depending on the number of processors available on your computer. Thorough analyses of the full data takes 2 – 6 days, depending on the number of species in the model, and generally require at least 100,000 generations. Running multiple independent chains using different starting seeds and comparing results is a good way to ensure that the analyses are converging.

Next, save the file using ***File*** →***Save. . . .*** Another subwindow will appear for specifying the name and location for saving the XML file. Name the file "RunA.xml" and place it in a new folder called "RunA" you may create in the "BFD" tutorial folder.



Chain Length 1000

Store Every 10

Pre Burnin 0

Num Initialization Attempts 10

▼ tracelog

File Name RunA.log

Log Every 10

posterior
likelihood
prior
ThetaLogger
TreeHeightLogger

▶ screenlog

▼ treelog

File Name RunA.trees

Log Every 10

TreeWithMetaDataLogger.smallhemi129

Figure 9: The MCMC settings.

6.3 Running the stepping stone analysis with `BEAST`

Step 11: There are two ways to set up the stepping stone analysis; through a GUI, and by editing the XML. The GUI is more convenient but makes it a bit harder to transfer the analysis to a cluster, and since stepping stone analyses are typically very computational intensive, it often makes sense to run them on a cluster. In this step, we explain how to set up the analysis using the GUI, and in the next two steps it is explained how to set up an XML file through a text editor.

First, start the BEAST app-store by selecting the File/Launch apps menu in BEAUti (alternatively, double click the AppStore icon in the BEAST folder). A window similar to Figure 10 should pop up.

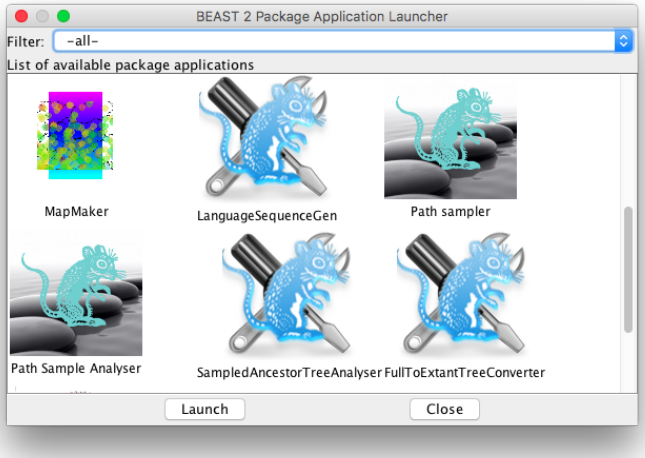

BEAST 2 Package Application Launcher
Filter: -all-
List of available package applications
MapMaker
LanguageSequenceGen
Path sampler
Path Sample Analyser
SampledAncestorTreeAnalyser
FullToExtantTreeConverter
Launch
Close

Figure 10: BEAST app store.

Select the Path sampler icon, and hit the Launch button. A new window pops up with the GUI for path sampling/stepping stone analysis similar to Figure 11.

If you prefer to start from the command line, you can use the following in a terminal:

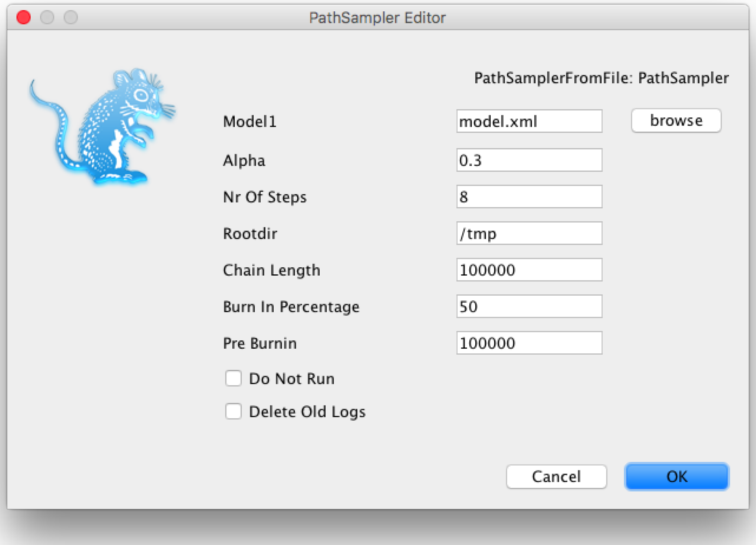
```
/path/to/beast/bin/appstore PathSampler
```

We need to change some of the settings for this tutorial:

Model1: Use the browse button to select the file with MCMC analysis you just set up in BEAUti.
Alpha: Changing this can help – or hurt – the efficiency of the numerical integration of the marginal likelihood, leave at the default.
Nr Of Steps: The number of steps to use, more is better but slower. Change to 12.
rootdir: Folder for storing output. Change to the **full** path of the "RunA" folder you created earlier. For example, on my computer this is "/home/me/Documents/beast-docs/BFD/RunA" (without the quote marks).
Chain Length: The length of each chain for each step. Change it to 1000. Longer is better, but 1000 steps will fit within the tutorial time.
Burn In Percentage: Burn-In percentage used for analyzing the log files, leave at the default.
Pre Burnin: Number of samples that are discarded for the first step, but not the others. Change to 0.
Do Not Run: Create the necessary XML files and exit, useful for running on clusters. Leave off.
Delete Old Logs: Delete any previously created log files. Leave off.

Now just click OK to run the stepping stone sampling for the "RunA" species delimitation hypothesis.

Step 11b. Manually editing the XML file for marginal likelihood estimation – OPTIONAL and ALTERNATIVE to Step 11. Instructions for setting up marginal likelihood estimation using path/stepping stone sampling are provided at the BEAST website. The procedure involves (1) typing in some short codes in a few places, (2) replacing some words, and (3) copying and pasting some sections around. Specific instructions are below:



PathSamplerFromFile: PathSampler

Model1	model.xml
Alpha	0.3
Nr Of Steps	8
Rootdir	/tmp
Chain Length	100000
Burn In Percentage	50
Pre Burnin	100000

□ Do Not Run

□ Delete Old Logs

Figure 11: GUI for path sampling/stepping stone analysis.

Open your XML file in a text editor. Search and replace the opening run statement (located about half way through the file) with an mcmc statement by changing “**<run ...>**” into “**<mcmc ...>**”. Next, type a new closing mcmc statement, “**</mcmc>**”, just before the closing run statement, “**</run>**”, located at the end of the file.

Now you are ready to insert the path/stepping stone sampling commands. You will need to insert the following block of text into your XML file immediately above the opening “<mcmc ...>”’ element:

```
 <run spec='beast.inference.PathSampler'
chainLength="1000"
alpha='0.3'
rootdir='/path/to/BFD/RunA/'
burnInPercentage='50'
preBurnin="0"
deleteOldLogs='true'
nrOfSteps='12'>
cd $(dir)
java -cp $(java.class.path) beast.app.beastapp.BeastMain $(resume/overwrite) -java -seed $(seed) beast.xml
```

Important: If you copy and paste this section into your XML file, be sure to check that the symbols paste correctly. Also, make sure that the root folder path (rootdir) exists on your computer.

These path sampling parameters are way to low, and a thorough analysis requires much more computational time. Stable marginal likelihood estimates usually require at least 48 steps (sometimes 100), chainLength = 100,000 (sometimes 1,000,000), and preBurnin=10,000 (sometimes 100,000). The MCMC run times are intentionally kept short in this tutorial to obtain quick (but meaningless) results. The run time on a MacBook Pro 2.3GHz i7 processor with 16GB of memory is approximately 2.5 minutes, and this is running 8 concurrent steps (= 8 threads). Increasing the number of threads will speed up the analysis by running more concurrent path sampling steps, but this requires more memory (this analysis uses about 12GB of memory). For large-scale analyses, many users find that they run out of memory before processors.

The path sampling parameters that you just entered into your XML file are as follows:

chainLength: MCMC sample length for each path sampling step.
alpha: parameter used to space out path sampling steps.

rootdir: folder for storing output. Be sure that the folder exists before starting the run.
burnInPercentage: burn-In percentage used for analyzing the log files.
preBurnin: number of samples that are discarded for the first step, but not the others.
deleteOldLogs: delete existing log files from rootdir
nrOfSteps: the number of path sampling steps to use

You can execute the XML file in `BEAST` using the GUI or the command line. If you are using Mac OS or Windows, you should be able to launch the `BEAST` GUI by double clicking on the application icon. After the `BEAST` window appears, click the `Choose File...` button, and select the XML file you just created (Figure 12). Increase the `Thread pool size` to speed up your analysis. Running SNAPP with multiple threads can increase speeds, but experimenting with the number of threads is required to get the best performance.

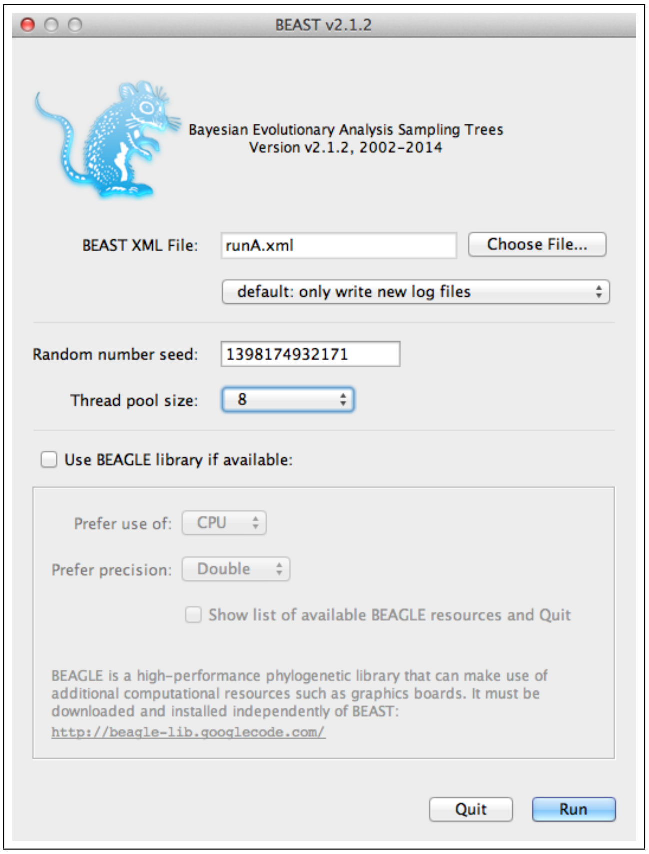
Click `Run`. The analysis should take about 10 minutes. You can also run `BEAST` from the command line. Open the **Terminal** Application and navigate to the "RunA" folder. To execute the file, type the following at the command line:

```
/path/to/beast/bin/beast -threads 8 RunA.xml
```

or

```
beast -threads 8 RunA.xml
```

if you have already copied the `BEAST` executable to your path. Caution: setting the number of `threads` beyond the maximum number available on your computer can have serious drawbacks, and you will probably not have enough memory to support all of those separate analyses.



BEAST v2.1.2

Bayesian Evolutionary Analysis Sampling Trees
Version v2.1.2, 2002-2014

BEAST XML File: runA.xml

default: only write new log files

Random number seed: 1398174932171

Thread pool size: 8

□ Use BEAGLE library if available:

Prefer use of: CPU

Prefer precision: Double

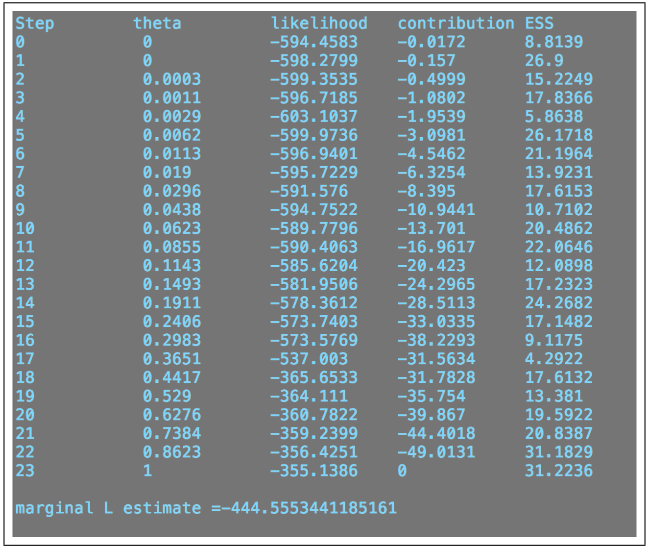
□ Show list of available BEAGLE resources and Quit

BEAGLE is a high-performance phylogenetic library that can make use of additional computational resources such as graphics boards. It must be downloaded and installed independently of BEAST:
http://beagle-lib.googlecode.com/

Figure 12: The `BEAST` GUI window.

6.4 Inspecting path sampling results

Step 12: At the end of your analysis, the path sampling results will be displayed on the screen. An example is shown in Figure 13. Each row shows the results from one path sampling step. The example in Figure 13 shows the results from a path sampling analysis with 24 steps. You will use the value after "marginal L estimate" to compare models.



```
Step        theta        likelihood   contribution ESS
0            0           -594.4583    -0.0172      8.8139
1            0           -598.2799    -0.157       26.9
2            0.0003      -599.3535    -0.4999      15.2249
3            0.0011      -596.7185    -1.0802      17.8366
4            0.0029      -603.1037    -1.9539      5.8638
5            0.0062      -599.9736    -3.0981      26.1718
6            0.0113      -596.9401    -4.5462      21.1964
7            0.019       -595.7229    -6.3254      13.9231
8            0.0296      -591.576     -8.395       17.6153
9            0.0438      -594.7522    -10.9441     10.7102
10           0.0623      -589.7796    -13.701      20.4862
11           0.0855      -590.4063    -16.9617     22.0646
12           0.1143      -585.6204    -20.423      12.0898
13           0.1493      -581.9506    -24.2965     17.2323
14           0.1911      -578.3612    -28.5113     24.2682
15           0.2406      -573.7403    -33.0335     17.1482
16           0.2983      -573.5769    -38.2293     9.1175
17           0.3651      -537.003     -31.5634     4.2922
18           0.4417      -365.6533    -31.7828     17.6132
19           0.529       -364.111     -35.754      13.381
20           0.6276      -360.7822    -39.867      19.5922
21           0.7384      -359.2399    -44.4018     20.8387
22           0.8623      -356.4251    -49.0131     31.1829
23           1           -355.1386    0            31.2236

marginal L estimate =-444.5553441185161
```

Figure 13: The path sampling output at the end of the analysis.

6.5 Setting up new species delimitation models

Step 13: Now that you have one XML file up and running it is easy to make new XML files for each species delimitation model. To prepare a new file for species delimitation, make a few slight modifications to the existing RunA.xml file: (1) save a copy of the xml file as RunB.xml and save it in a new folder called "RunB", (2) find and replace "RunA" with "RunB" in the xml file so that you don't accidentally overwrite any of your previous results, (3) change the species assignments.

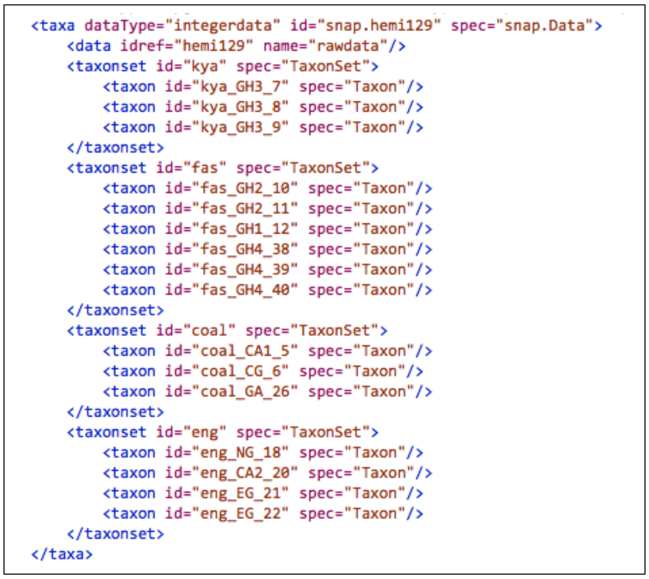
This last part requires changing the number and/or composition of taxonset features. Each taxonset begins with **"<taxonset ...>"** and ends with **"</taxonset>"** (Figure 14). To lump species, simply combine the taxon names into a single taxonset feature. To split a species, simple create a new taxonset with an appropriate new "id" name, and cut and paste the relevant **"<taxon ... />"** elements. To reassign a taxon to another species you can cut and paste the taxon to a different taxonset. Although you could create a new XML with the alternative species assignments from scratch by rerunning `BEAUTi`, it is much more efficient to duplicate your already existing XML and edit the taxonsets.

While the taxon names in the data and XML files are a little cryptic, it should be straightforward to lump species based on the hypotheses in Figure 1. For splitting and reassignment, you should transfer the following individuals:

H. fasciatus: Split (RunE) Ankasa Conversation Area samples, labeled as fas_GH4_38, _39 and _40.
H. eniangii: Split (RunF) or reassign (RunG) Bioko Island individuals, labeled as eng_EG_21 and _22.

After you have set up an XML file for each hypothesis you can calculate the marginal likelihoods by running through Step 11 again for each XML individually.

Step 14: After you run each of the alternative species delimitation models you can rank them by their marginal likelihood estimate (MLE). You can also calculate Bayes factors to compare the models. The Bayes factor (BF) is a model selection tool that is simple and well suited for the purposes of comparing species delimitation models. Calculating the BF between models is simple. To do so, simply subtract the MLE values for two models, and then



```
<taxa dataType="integerdata" id="snap.hemi129" spec="snap.Data">
    <data idref="hemi129" name="rawdata"/>
    <taxonset id="kya" spec="TaxonSet">
        <taxon id="kya_GH3_7" spec="Taxon"/>
        <taxon id="kya_GH3_8" spec="Taxon"/>
        <taxon id="kya_GH3_9" spec="Taxon"/>
    </taxonset>
    <taxonset id="fas" spec="TaxonSet">
        <taxon id="fas_GH2_10" spec="Taxon"/>
        <taxon id="fas_GH2_11" spec="Taxon"/>
        <taxon id="fas_GH1_12" spec="Taxon"/>
        <taxon id="fas_GH4_38" spec="Taxon"/>
        <taxon id="fas_GH4_39" spec="Taxon"/>
        <taxon id="fas_GH4_40" spec="Taxon"/>
    </taxonset>
    <taxonset id="coal" spec="TaxonSet">
        <taxon id="coal_CA1_5" spec="Taxon"/>
        <taxon id="coal_CG_6" spec="Taxon"/>
        <taxon id="coal_GA_26" spec="Taxon"/>
    </taxonset>
    <taxonset id="eng" spec="TaxonSet">
        <taxon id="eng_NG_18" spec="Taxon"/>
        <taxon id="eng_CA2_20" spec="Taxon"/>
        <taxon id="eng_EG_21" spec="Taxon"/>
        <taxon id="eng_EG_22" spec="Taxon"/>
    </taxonset>
</taxa>
```

Figure 14: Example of the taxonset features in the XML file.

multiply the difference by two:

$$\text{BF} = 2 \times (\text{MLE}_1 - \text{MLE}_0) \tag{1}$$

A positive BF value indicates support in favor of model 1, while a negative BF value indicates support in favor of model 0. To calculate Bayes factors in this tutorial, the current taxonomy (RunA) will always be model 0.

The strength of support from BF comparisons of competing models can be evaluated using the framework of Kass and Raftery (1995). The BF scale is as follows: $0 < \text{BF} < 2$ is not worth more than a bare mention, $2 < \text{BF} < 6$ is positive evidence, $6 < \text{BF} < 10$ is strong support, and $\text{BF} > 10$ is decisive.

The results for the seven gecko models are provided in Table 2. The model that splits *eniangii* into two species (RunF) is the top-ranked model. It has the highest MLE value, and it is supported in favor of the current taxonomy model (RunA). The BF in support for model F is decisive compared to model A. It is important to emphasize that these results are *tragically deficient* in terms of the MCMC analysis. Much, much longer runs are required to obtain stable results.

Table 2: Stepping stone sampling results for the seven species delimitation models shown in Figure 1. Positive BF values indicate support for the alternative model, and negative BF values indicate support for the current taxonomy.

Model	Species	MLE	Rank	BF
RunA, current taxonomy	4	-816.41	2	–.–
RunB, lump western forests	3	-846.21	5	-59.60
RunC, lump central forests	3	-883.16	6	-133.50
RunD, lump western and central forests	2	-899.22	7	-165.62
RunE, split *fasciatus*	5	-840.17	4	-47.52
RunF, split *eniangii*	5	-766.03	1	100.76
RunG, reassign Bioko Island	4	-818.87	3	-4.92

MLE = Marginal likelihood estimate
BF = Bayes factor

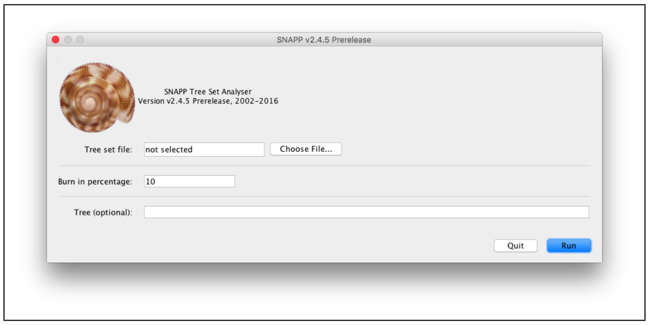
6.6 Summarizing the trees using TreeAnnotator.

Step 15: TreeAnnotator will summarize the posterior distribution of species trees and identify the topology with the best posterior support, and summarize the divergence times for each node in the tree. Launch the `TreeAnnotator` program. For the `Target tree type` field, choose **Maximum clade credibility tree**. For the `Node heights` field, choose **Median heights**. You will typically only want to summarize the results corresponding to theta=1, which is the step where trees are sampled from the posterior distribution. Check the SNAPP screen output to verify which path sampling step corresponds to theta=1 (this changes with different versions of SNAPP). Select the `Input Tree File` button and select the file "RunA.trees" in the stepN subfolder of "RunA" that corresponds to theta=1. Select the `Output File` button to specify the output folder and a file name, e.g. "RunA-MCC.tree". Click `Run`

6.7 Visualizing the tree in FigTree

Step 16: Launch the `FigTree` program, and load the "RunA-MCC.tree" file you just created with `TreeAnnotator`. Check the `Branch Labels` option and select **posterior** for the ***Branch labels*** →***Display*** fields. Check the `Node Bars` option and select **height_95%_HPD** for the ***Node bars*** →***Display*** field.

You can also get a summary of some tree statistics using the TreeSetAnalyser, which you can launch from the BEAST app launcher, and looks something like Figure 15



SNAPP v2.4.5 Prerelease

SNAPP Tree Set Analyser
Version v2.4.5 Prerelease, 2002-2016

Tree set file: not selected

Burn in percentage: 10

Tree (optional):

Figure 15: Tree set analyser utility.

7 Additional Information/Resources

Other resources for `BEAST` and `SNAPP` are available on the web page https://www.beast2.org/snapp/.

8 Bibliography

Baele, G., P. Lemey, T. Bedford, A. Rambaut, M. A. Suchard, and A. V. Alekseyenko. 2012. Improving the accuracy of demographic and molecular clock model comparison while accommodating phylogenetic uncertainty. Molecular Biology and Evolution 29:2157–2167.

Bouckaert, R., J. Heled, D. Kühnert, T. Vaughan, C.-H. Wu, D. Xie, M. A. Suchard, A. Rambaut, and A. J. Drummond. 2014. BEAST 2: A software platform for Bayesian evolutionary analysis. PLOS Computational Biology 10:e1003537.

Grummer, J. A., R. W. Bryson, and T. W. Reeder. 2013. Species delimitation using Bayes factors: simulations and application to the *Sceloporus scalaris* species group (Squamata: Phrynosomatidae). Systematic Biology 63:119–133.

Kass, R. E. and A. E. Raftery. 1995. Bayes factors. Journal of the American Statistical Association 90:773–795.

Leaché, A. D., M. K. Fujita, V. N. Minin, and R. Bouckaert. 2014. Species delimitation using genome-wide SNP data. Systematic Biology 63:534–542.

Mirarab, S., R. Reaz, M. S. Bayzid, T. Zimmermann, M. S. Swenson, and T. Warnow. 2014. ASTRAL: Genome-scale coalescent-based species tree estimation. Bioinformatics 30:i541–i548.

Mirarab, S. and T. Warnow. 2015. ASTRAL-II: Coalescent-based species tree estimation with many hundreds of taxa and thousands of genes. Bioinformatics 31:i44–i52.

Nielsen, R. 1998. Maximum likelihood estimation of population divergence times and population phylogenies under the infinite sites model. Theoretical Population Biology 53:143–151.

Ogilvie, H. A., R. R. Bouckaert, and A. J. Drummond. 2017. StarBEAST2 brings faster species tree inference and accurate estimates of substitution rates. Molecular Biology and Evolution 34:2101–2114.

Rannala, B. and Z. Yang. 2017. Efficient Bayesian species tree inference under the multispecies coalescent. Systematic Biology 66:823–842.

RoyChoudhury, A., J. Felsenstein, and E. A. Thompson. 2008. A two-stage pruning algorithm for likelihood computation for a population tree. Genetics 180:1095–1105.

Zhang, C., E. Sayyari, and S. Mirarab. 2017. ASTRAL-III: Increased scalability and impacts of contracting low support branches. Pages 53–75 *in* 15th International Workshop on Comparative Genomics (J. Meidanis and L. Nakhleh, eds.).