Advanced methods in DNA sequence and mulilocus data analyses

Lesson 2 – DNA sequences II

(Eliška Záveská – Faculty of Science UK, Prague)

29. October 2016

Analyses of multiple sequence data datasets, incongruence tests, gene trees vs. species tree reconstruction, networks, detection of hybrid species

I. Test of congruence of multiple alignments [ILD test, Concaterpillar]
II. Visualization of gene tree incongruences using NeighbourNet [Splitstree]
III. Reconstruction of species tree under multicoalescent model [*BEAST]
IV. Reconstruction of species network (coalescence/hybridization) [Phylonet]

When reconstructing phylogeny using sequence data we have to keep in our minds that different parts of the genome experienced a bit different evolutionary history and therefore also sequence data from different parts of the genome can give us a bit different reflection of this situation. Our aim is to estimate the evolutionary history of groups of organisms (e.g. species), so we have to figure out, which markers give us the best insight into this question or how to analyze the data to get most likely species phylogeny.

In this session you will practice gene tree reconstructions (you can employ the experiences and manuals from the previous session 'DNA sequences I') and you will learn how to compare multiple gene trees that you reconstruct based on different regions in the genome and/or plastome (e.g. different coding/non-coding regions of nDNA or cpDNA). Except your own eyes we can employ several types of tests for congruence/incongruence of tree topology (and/or branch length), e.g. using programs PAUP (ILD test) or Concaterpillar. We can also plot two trees and visualize their incongruence using Dendroscope. Congruence tests generally tell you, how you should further treat your data – if you can „pretend“ that different gene regions share the same evolutionary history so you can analyze them as a concatenated dataset (perhaps just partitioned and analyze partitions under different evolutionary model) or if you should analyze the datasets separately because they do not reflect the same evolutionary histories and by analysing them as concatenated dataset you could violate assumptions of some analyses.

In parallel you can quickly visualize concatenated dataset (including all your particular sequence datasets, even when they are significantly incongruent) in Splitstree. The reconstructed NeighbourNetwork is useful to visualize stronger incongruences among datasets caused usually by presence of hybridogenous individuals/species.

When we know enough about our gene trees we can proceed to reconstruct the species tree that should reflect phylogenetic relationships among evolutionary units of our interest (commonly species or distinct intraspecific evolutionary lineages) better that various gene trees. For this task we can employ multispecies coalescent approach *BEAST implemented in software BEAST. Such approach is suitableonly in the case when there is/was no gene flow among the evolutionary units we are interested in (i.e. topological incongruences among gene trees should be caused only by deep coalescence/presence of ancestral polymorphism, not by hybridization). When we suspect that in our dataset can be also hybridogenous species, we should choose rather species network reconstruction, e.g. using recently developed Maximum pseudo-likelihood inference of species network implemented in Phylonet.

Aim: By following this manual you should get from multiple sequence alignments up to the final species tree or species network. Also, you should be able to detect, if your dataset includes hybrids.

HOMEWORK/PROJECT for 5 selected students:

- *run the following manual with two different starting datasets – one including hybrid species and one without hybrids (you will find the input data in the corresponding packages online)*

- *compare the results of ILD test, Neighbour-Net, *BEAST and PhyloNet analyses for those two datasets and discuss them whith focus on following questions:*

1) Which analysis shows the hybrid species in most straightforward way?

2) Which analysis migh be most negativelly affected when the hybrids are included?

3) Which analysis is most sensitive to the lack of information in particular markers in the analysis?

==========

I. Test of dataset incongruence – ILD test [PAUP]

To compare topology of two (or more) gene trees you can first just open your trees e.g. in Figtree, use your eyes and identify which individuals are placed in different position in different trees. This activity is worth to perform as you can roughly estimate how incongruent your datasets are and it will help you to interprete the results of further analyses. To evaluate congruence of the tree topologies statistically there are multiple possible approaches. One of the traditionally used tests – Incongruence length difference (ILD) test also known as „**Partitions homogeneity test**“ (Farris et al., 1994) – seems to be nowadays rather outdated (see attached literature) but it might be still useful. For some

modern alternatives see e.g. (Leigh et al., 2008) and the Optional manual in paragraph **Ia**.

This test we apply on concatenated dataset of multiple genes in for which we define partitions for each gene region (e.g. partition_1 – nDNA, partition_2 – cpDNA). Test cen be performed e.g. in PAUP.

- programs:
  - PAUP, http://people.sc.fsu.edu/~ dswofford/paup_test/

- test datasets (phylogeny of genus *Curcuma*, Zingiberaceae) – six alignments in fasta format. For the test, you can compare just few of them. E.g.
  - CHS_s_H_names.fas
  - GAPDH_s_H_names.fas
  - GLO3_s_H_names.fas
  - ITS_s_H_names.fas
  - Leafy_I_s_H_names.fas
  - cpDNA_s_H_names.fas

- run the analysis
  - first you have to prepare input nexus file that includes at least two partitions. Concatenate two fasta files (e.g. using Fabox) and note the start and end position of each alignment in the concatenated dataset. For instance, when you concatenate CHS and cpDNA datasets CHS partition will be 1-1032bp and cpDNA will be 1033-6788bp.
  - Convert concatenated fasta file to nexus and at the end of the file add the nexus block that will define the partitions according the example below.
  - Open **Paup.exe** and using **File → Open** load input nexus file and execute it.
  - When the window pop ups and ask for increase the 'Maxtrees' set up higher the value (e.g. 1000) and select option „Leave unchanged, and don't prompt“.
  - When analysis finish we will get p-value according to which we reject or accept the null hypothesis – H0: dataset is homogenous, i.e. all the partitions of the dataset reflect the same evolutionary hypothesis and same tree topology. If $p < 0.05$ we reject H0 and accept the alternative that compared partitions reflects different evolutionary history and we have to further analyze them as incongruent datasets.

EXERCISE: run more pair comaprisons between the datasets. Remove individuals: rosc1, rosc2, smit1, cand1, arom2, vama1 and run the same pair comparisons again with the reduced dataset. Have you encountered differences in the results? Can you explain it?

OPTIONAL: I.a Hierarchical likelihood-ratio test for phylogenetic congruence using Concaterpillar

To compare topology of two (or more) gene trees you can first just open your trees e.g. in Figtree, use your eyes and identify which individuals are placed in different position in different trees. This activity is worth to perform as you can roughly estimate how incongruent your datasets are and it will help you to interprete the results of further analyses. To evaluate congruence of the tree topologies statistically there are multiple possible approaches. One of the traditionally used tests – Incongruence length difference (ILD) test (Farris et al., 1994) – seems to be nowadays more or less outdated (see attached literature). Therefore we will here use one of the modern alternative – Concaterpillar (Leigh et al., 2008) - a hierarchical clustering method based on likelihood-ratio testing that identifies congruent loci for phylogenomic analysis. Concaterpillar also includes a test for shared relative evolutionary rates between genes indicating whether they should be analyzed separately or by concatenation.

- programs:
  - Concaterpillar v. 1.8a; http://leigh.net.nz/software.shtml
  - RaxML v. 7.0, 7.2 or 7.3; http://sco.h-its.org/exelixis/resource/download/software/RAxML-7.3.0.tar.bz
  - Python version < v.3 (e.g. v.2.7. works well) and module SciPy

- test datasets (phylogeny of genus *Curcuma,* Zingiberaceae) – **fasta** files renamed to have suffix *.seq that is requested by Concaterpillar:
  - CHS_s_H_names.seq
  - GAPDH_s_H_names.seq
  - GLO3_s_H_names.seq
  - ITS_s_H_names.seq
  - Leafy_I_s_H_names.seq
  - cpDNA_s_H_names.seq

- running the program:
  - alocate the input files and all python scripts from folder into single directory. This directory will be then your working directory for further analyses.
  - open the terminal and change directory to the working directory (i.e. folder where Concaterpillar scripts and input data are located). To change directory use command 'cd' and the PATH to the desired directory, e.g:

```
cd /media/ez/1TB_ez/_kurz_hodnoceni_dat_2016/concaterpillar-1.8a
```

Fig.1. You will find the terminal icon using search field in Ubuntu. Open the Terminal by double click.

- When you are in your working directory, you can check its content using command 'ls'
- open Concaterpillar manual using command:

```
gedit Concaterpillar-1.8-MANUAL.txt
```

- following the manual run first script 'ccpinstall.py' to locate RAxML executable

```
python ccpinstall.py
```

- usually the script ccpinstall.py finds the RaxML executable itself. If not, you have to specify the PATH where you installed RaxML.
- When RaxML is succesfully located, you can run Concaterpillar analysis using:

```
python concaterpillar.py -t -b -c 4 -m GTR
```

- consult chosen parameters (`-t -b -c 4 -m GTR`) with manual and make sure you know what they mean.
- The analysis take some time (minutes to many hours), depends mainly on number of input alignments and of course on parameters of your machine, especially on number of processors you can use for parallelizatiopn of RaxML, i.e. `-c` parameter in settings mentioned above)
- Even those six alignments from test data may take some hours to compare, so you can find the results in the folder 'RESULTS' that is part of the folder with test data.

- INTERPRETING the RESULTS
  - results of the Concaterpillar will be placed in your working directory where many new files and some new folders appear. First open the file 'results.ccp'. In this file you will find the set of alignments that are (up to some degree, depends on previous settings of p value) congruent in tree topology. In our example congruent are alignmets
    - 'GAPDH_s_H_names.seq' and 'cpDNA_s_H_names.seq'

        - 'CHS_s_H_names.seq' and 'ITS_s_H_names.seq'
        - 'GLO3_s_H_names.seq' and 'Leafy_I_s_H_names.seq'
    - comparable alignments are placed also in folders with names 'Bltest-set000', 'Bltest-set001' etc. Within each of those folders you can find the file named 'bltest-set000.ccp', 'bltest-set001.ccp' etc. where are further summarized results of the Branch length test for alignments placed within these folders. In our example none of the pair of alignments congruent in topology did pass the Branch length test.
    - In summary, our test data were divided in three groups that includes alignments with comparable topology, but within these groups alignments are not comparable because they significantly differ in branch lengths. Generally you should not analyze those dataset as one concatenated dataset. Perhaps you could concatenate the alignments that are congruent in topology, but you should defined partitions for particular regions and for which you will set up different model of evolution (inferred e.g. using jModeltest, as you did in Lesson I).

EXCERCISE: Think about possible reasons of those incongruencies (you can also read some papers, e.g. Leigh et al., 2008 that is attached). Results of further analyses will also help you to understand the results of congruence tests. Think about how you should analyze fulrther your dataset(s) and why.

I. Visualization of gene tree incongruences using NeighbourNet using software Splitstree

By constructing the *neighbour-network* in software Splitstree you can easily and fastly visualize conflicts in your dataset(s) caused e.g. by hybridization or another type of recombination. If you analyze single alignment you can detect mainly conflicts caused by recombination within the region you study. You can also concatenate multiple alignments of different regions and detect incongruences between these alignments. This analysis is very powerful explorative tool, but it is based only on visualization of *distances*, so you should always test/verify the results of this analysis by other approaches.

Neighbour-Net algorithm compares genetic distances (by default uncorrected p-distances) between individuals and plotting them into the networks. When there is no signal of conflict within dataset (i.e. all your alignments would give you approximately same gene tree topology) the Neighbour-Net will have topology similar to unrooted tree. If there is stronger conflict between the alignments, individuals that are causing the incongruence are placed in between the lineages where they belong based on particular datasets. In the example figure below the individuals with black labels are not unambigously placeable into any of the coloured group thet represent main evolutionary lineage and therefore they are suppose to be of hybrid origin.

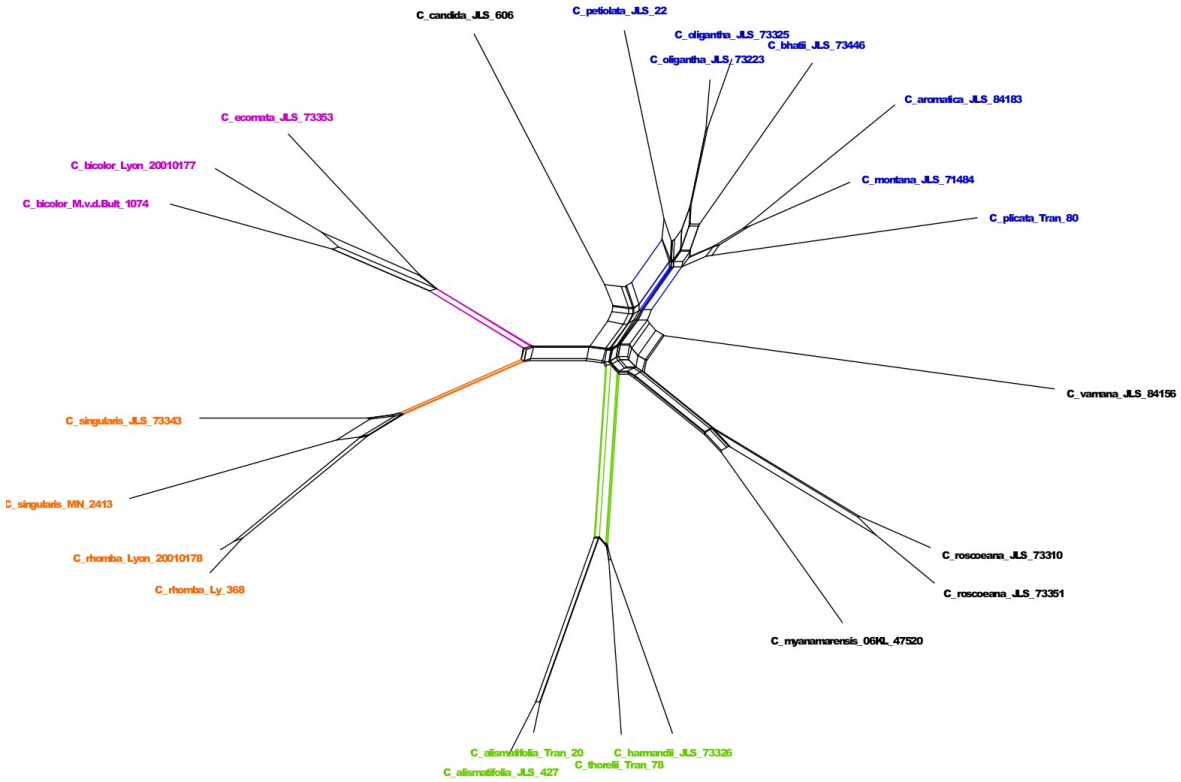

Fig. 2. NeighbourNet analysis based on markes/loci . Plausible hybridogenous species are in black and usually appears on a branch originating in 'box' structure in the network. Grouping of the species into main avolutionary lineages is highlighted in colours.

- programs:
  - Splitstree: (http://www.splitstree.org/

• test datasets (phylogeny of ge*nus Curcu*ma, Zingiberaceae) – the s**ame f**asta files that were used for Concaterpillar EXERCISE, but with different suffix.

  - CHS_s_H_names.fas
  - GAPDH_s_H_names.fas
  - GLO3_s_H_names.fas
  - ITS_s_H_names.fas
  - Leafy_I_s_H_names.fas
  - cpDNA_s_H_names.fas

• running the program:

  - input file for Splitstree is NEXUS, so you have to convert your fasta files to NEXUS first (using Mega or some web tool, as you did it in Lesson I)
  - To visualize conflicts between particular datasets you should concatenate two or more files into single concatenated alignment that you will then convert to nexus

- open Splitstree by double click on Splitstree icon
- open the NEXUS file: *File* → *Open* (choose file with suffix *.nex), the analysis runs automatically with default settings that are good enough for majority of rough analyses.
- If you want to exclude some individual from the analysis you can easily highlight the individual that you want to exclude in the network, right click and choose – *Exclude selected taxa*. If you want to modify included species more extensively you have to go to main menu bar and select *Data* → *Filter taxa.*
- Results of the analyses. i.e. figures of the networks, you can easily export to pdf using *File* → *Export Image*.

EXERCISE: After the analysis of concatenated test datasets or your own data try to say if you see some indications of conflict in the dataset or not. If yes, try to say what individuals cause the conflict(s). Then remove those individuals and compare the topology of the network.

II. Reconstruction of species tree under multicoalescent model *BEAST implemented in BEAST

When you see how different can be gene tree topologies based on different gene regions you might be interested in reconstruction of 'genuine' relationships between species (or distinct intraspecific evolutionary lineages) by reconstructing species tree.

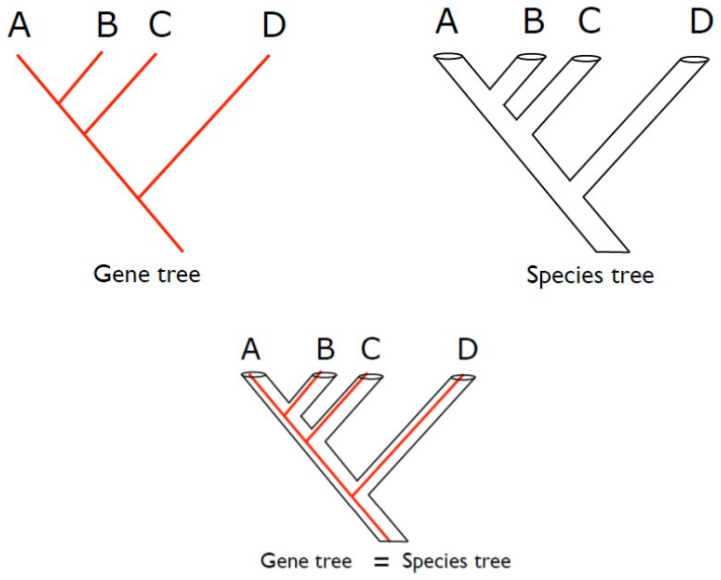

Fig. X. Gene tree vs. species tree. The equal topology of gene tree and species tree is rather exception than a rule.

Species tree analysis usually takes as a source data multiple gene alignments (or multiple gene tree topologies) and reconstructs the relationships among pre-defined species (or distinct intraspecific evolutionary lineages). Among most

popular way to reconstruct species tree belongs so called 'multi species coalescence' approach, which is the model for species tree where possible incongruences among particular gene trees are explained by incomplete lineage sorting (ILS).

The well known and broadly used tool to compute species tree under this model is *BEAST implemented in software BEAST that employs Bayesian inference. Disadvantage of this approach is the assumption of the model that there is no gene flow among the species you want to analyze. The alternative models where you are allowed to include hybridogenous taxa exists (see the chapter IV. In this manual) but they are generally much more complex and they might have other model assumptions that limitates usage of such tools.
In the example/EXERCISE below you will see how the analysis behave when there are also hybridogenous species present in the dataset. As a homework you can run the same analysis without the hybridogenous species and compare the results.

As an input for the *BEAST analysis you will need the alignments from at least two gene regions and to allow the analysis correctly estimate parameters of the coalescent model you should include at least two individuals per single species (or your desired evolutionary lineage).

- programs:
  - BEAST, preferable version 1.8.1: http://beast.bio.ed.ac.uk/
    - downloaded package includes BEAST java executable and several additional programs (also as executables) that will be useful to summarize and/or visualize the results of the analysis. The most important executables are:
    - BEAUti v1.7.5.exe
    - BEAST v1.7.5.exe
    - TreeAnnotator v1.7.5.exe
    - LogCombiner v1.7.5.exe

  - Tracer: http://tree.bio.ed.ac.uk/software/tracer/
    - Tracer v1.5.exe
  - FigTree: http://tree.bio.ed.ac.uk/software/figtree/
    - FigTree v1.3.1.exe
  - DensiTree; part of BEAST2 package: http://beast2.org/
    - DensiTree.exe

- test datasets (phylogeny of genus *Curcuma,* Zingiberaceae) – the same **fasta** files that were used for Concaterpillar EXERCISE, but with different suffix.
  - CHS_s_H_names.fas
  - GAPDH_s_H_names.fas
  - GLO3_s_H_names.fas
  - ITS_s_H_names.fas
  - Leafy_I_s_H_names.fas
  - cpDNA_s_H_names.fas

- running the program:
  - BEAST/BEAUti usually accepts fasta files as an input data format, but sometimes you need to convert input files to NEXUS format. Try first to import fasta and convert to NEXUS, if some error message comes from BEAUti
  - open BEAUti (**BEAUti v1.7.5.exe**) and import all input files using **File →Import Data**. Using mouse highlight all imported datasets and in the top left corner
  - You should set up that all the dataset should be unlinked, i.e. they might results in different tree topology and might fit to different evolutionary model (as already suggested by Concaterpillar results). Above the table with input data there you click on buttons “Unlink Subst. Models”, “Unlink Clock Models” a “Unlink Trees” that will ensure that we can set up different parameters/evolutionary models for particular datasets.
  - In the upper left corner tick the box “**Use Species Tree ancestral reconstruction (*BEAST) Heled & Drummond 2010**“. The window pop ups and ask you to choose the way you want to define the species for your analysis, i.e. how to group the individuals under the study into new OTUs that represents species. You can choose „**Create a new trait**“ to define species manually and click „ok“.
  - In next window you have to add the name of the species to each individual. If the name of the species is included in the name of the individual you can make this task faster using button **'Guess trait value'**. In pop-up window you can specify where in the label of individual is the name of the species. In exemplar data you have to choose option '**Defined by regular expression (REGEX)**' and write in the field '[a-z]{1,}' which means that all the small caps in the individual label correspond to name of the species.
  - In the tab „Sites“ we can set up model of DNA evolution for each locus (the model we estimate e.g. using jModeltest in advance)
    - Let's say that for all of our alignments the most suitable model is: „Substitution model“ – GTR, „Base frequencies“ – Estimated, „Site Heterogeneity Model“ – Gamma; so you can set it up like this.
    - To change the settings for other loci you have to click on the name of another locus in the column „Substitution Model“ on the left.
  - In the tab „Clocks“ we can set up model of molecular clocks for each locus. The most usefull choices in most of the cases are 'strict clock' and 'Lognormal relaxed clock (Uncorrelated)'. Between these two models you decide based on test of clocklikness of your data that you can perform e.g. in Mega (see manuals for Lesson I). Let's leave default 'strict clock' settings for all loci. Further it is important to check the all boxes except one in column 'Estimate'. The locus where the box is not check will have a fixed clock rate while the clock rates of the other loci will be scaled according the fixed one.

- In the tab „Trees“ you set up prior informations for estimation of gene tree and species tree model parameters. In section „Species Tree prior used to start all gene tree models“ you can leave default settings. In section „Tree Model“ it is necessary to specify for each locus „Ploidy type“, i.e. if the locus is „autosomal nuclear“ (in case of nuclear DNA loci) or „mitochondrial“ (in case of plastid loci). In the same section we can set up algorithm for construction of primary tree (e.g. „UPGMA starting tree“), but this settings do not have significant influence on the results.
- In the tab „Priors“ we can set up further prior information that can spead up the analysis and/or improve the results. In this case, however, we can leave default prior settings as they are except those that are highlighted in red font. The parameters highlighted in red regards to molecular clock rates for particular loci and here the default prior settings are not yet specified so we have to specify them. Click into the field with red font and in the pop-up window you change 'Prior Distribution' to 'Uniform' and upper value to '10'. Do the same for remaining loci. By this you shrink the value space in which the analysis will look for the best values when estimating the parameter. The aim is to shrink the interval in a way, that the interval still includes the most probable values of particular parameter but do not include values that are too unlikely and would just slow down the analysis that would have to explore entire big interval of unlikely values.
- In „MCMC“ tab we set up parameters of Bayesian analysis (BA), i.e.:
  - number of generations for which the BA will explore the tree space; this number mostly dependes on the amount of data in analysis – the bigger datasets (and the more loci), the more generations you need to bring the analysis to convergence. You can try default settings and by checking the analysis performance (in Tracer, see below) you can optimize/increase number of generations and run another analysis.
  - In the field „File name stem“ we introduce the name of your analysis. As you might run more analysis with similar but not exactly same settings it is worth to make the names of the analysis specific and self-explaining – it will save your time later on :).
- In the right bottom corner click the button „Generate BEAST File“ and another window pop-ups requesting you to confirm all settings. Confirm it and save the newly created *.xml file to the folder where you want to store the results of your analysis.

- Open executable of BEAST (**BEAST v1.7.5.exe**) and using „choose file“ select previously created *.xml file and run the analysis using „Run“
- To check/control performance of the analysis use program **Tracer**

- Open executable **Tracer v1.5.exe** and using **File →Import Trace file** import *.log file of the running or finished BEAST analysis. After loading log file you need to check especially the column 'ESS' in the left part of the window. In this column the ESS (Effective Sample Size) values of all parameters should be at least 200. The lower value of ESS indicates that the particular parameter haven't reach convergence yet and you should run the analysis longer (for more MCMC generations).
- If you click on tab 'Trace' in the right part of the window you should see balanced sampling of the parameter values around the mean parameter value along the process of the analysis. Like in figure below.

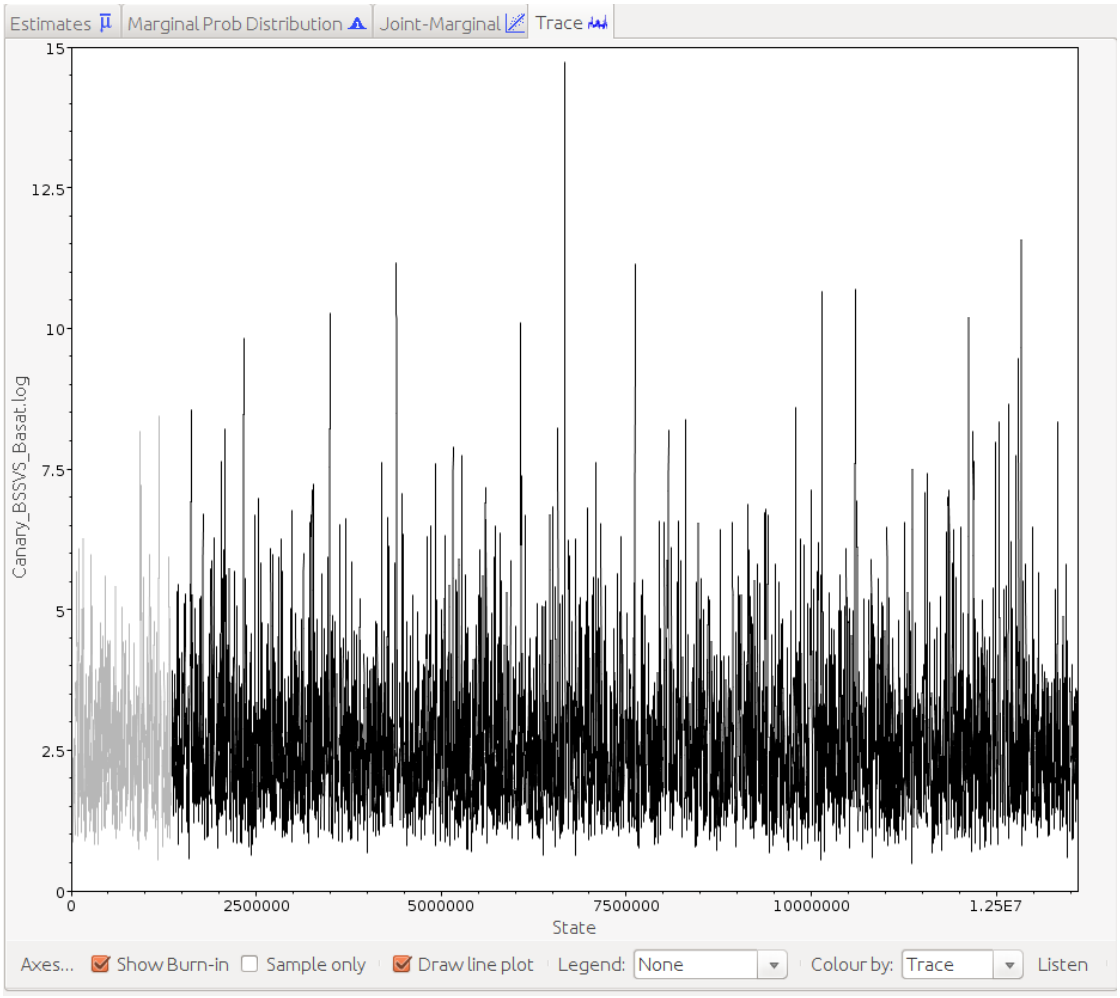

Fig. 3. Exemplar MCMC sampling of the parameter values in Bayesian analysis visualized in Tracer.

- During and/or after the analysis you can also check how your trees looks like using **Densitree**. The advantage of this tree visualizer is, that it plots all the trees in the file one over another so you can see the incongruences between the trees if there are some.

- Open executable **Densitree.exe** and using **File → Load** import the file resulting from the BEAST analysis that includes all species tree (the one with suffix *.species.trees). You should see picture similar to that shown below (Fig. 4). Unclear relationships between species might be coused either by lack of information in the data or by stronger conflicts between datasets caused by hybridization.

Fig. 4. Species tree of genus *Curcuma* representatives. Output of the *BEAST analysis visualized using Densitree.

- When the BEAST analysis is finished and log file in Tracer looks good, i.e. MCMC search has converged, you can construct consensus species tree based on the trees collected in file with suffix *.species.trees. For this task you will use **TreeAnnotator** anothe part of BEAST package.
- Open **TreeAnnotator v1.7.5.exe** and to the field “Burnin (as trees)” type the number of trees that should be removed as a burnin phase. In case you run 10 000 000 generations and you keep/save every 1000 tree and want first 25% of the trees to be removed as a burnin, you will type 2500 into that field. “Target tree type” should stay “Maximum clade credibility tree” and “Node heights” as a

"Median heights"Into the field". To the field "„Input Tree File" you choose the file with all species tree (i.e. that with suffix ***.species.tree**), in the field "Output file" you select the folder where you want to store the final consensus tree and type the new new name of this tree, e.g. with suffix ***.species_cons.tree**. Run the analysis using „Run".
    - Consensus species tree you can open e.g. in FigTree. To show the posterior probability (PP) values click to the tab „**Node labels**" in the left part of the window and **Display → posterior**.

EXERCISE: Check again the results of the species tree analysis in DensiTree and focus namely on the position of the species that had ambiguous position in Neighbour-Net analysis, i.e. *C. vamana* ('**vama**' in the analyses), *C. roscoeana* ('**rosc**' in the analyses), *C. myanmarensis* ('**smit**' in the analyses) and *C. candida* ('**cand**' in the analyses). Open the consensus specie tree in Figtree, check the PP values and again focus on the species mentioned above. For which of the species above you can infer their phylogenetic history unabmbiguously and with high probability?

IV. Reconstruction of species network in presence of incomplete lineage sorting (ILS) and hybridization using Phylonet

In the previous analysis using *BEAST we violated the assumption of the analysis when we involved possible hybridogenous species. Also, the results of *BEAST analysis did not gave us realiable answer to origin of those hybrid species. However, we run *BEAST analysis with hybrids to show you, what can happened if you include some hybrid species by accident or just because you don't know about their hybrid origin.

Until recently there were not many usefull approaches to reconstruct species tree for datasets that includes hybrids and account also for gene tree incongruence caused by ILS. Since ca last year there can be found more approaches for reconstruction of hybrid networks in presence of ILS applying different models and using different computation frameworks (e.g. Bayesian analysis, Maximum likelihood and others). Here we will use software PhyloNet that enables multiple analysis of species networks (each request a bit different type of input data and can have run under different conditions) as you can check here: https://wiki.rice.edu/confluence/pages/viewpage.action?pageId=8898533. We will use the approach that uses Maximum pseudo-likelihood inference of species network and gene tree topologies as an input data (Yu and Nakleh, 2015). It is relatively fast and reliable analysis but in my short experience with this approach it seems to be sensitive to quality of input gene trees.

- programs:
    - PhyloNet, https://wiki.rice.edu/confluence/pages/viewpage.action?pageId=8898533
    - Dendroscope, http://dendroscope.org/

- test datasets (phylogeny of genus *Curcuma,* Zingiberaceae) – gene trees in newick format based on the files used in previous analyses
    - CHS_s_H_names_ML.nwk
    - cpDNA_s_H_names_ML.nwk
    - GAPDH_s_H_names_ML.nwk
    - GLO3_s_H_names_ML.nwk
    - ITS_s_H_names_ML.nwk
    - Leafy_I_s_H_names_ML.nwk

- running the program:
    - first you have to create the input NEXUS file for PhyloNet analysis that will includes topology of all the gene trees and command to run the analysis.
    - You can **edit** already prepared nexus file that is also attached in the Phylonet folder ('**Curcuma_5trees_infer_network_MPL_5_reticulations.nexus**'). When you open this file you will find 5 gene trees topologies in section „BEGIN TREES;“. Instead of five trees we will analyze only two trees – gt0 and gt1. Replace the topologies that are in original nexus by the topologies of ITS and cpDNA gene tree in given newick test files. Further you have to change number of trees analyzed in section „BEGIN PHYLONET;“ from (gt0-gt4) to (gt0-gt1).
    - In section „BEGIN PHYLONET;“ there are commands for PhyloNet analysis and you can check what they do here:https://wiki.rice.edu/confluence/display/PHYLONET/InferNetwork_MPL. The most important is the command **'InferNetwork_MPL'** that calls chosen type of the analysis from PhyloNet package, then follows list of analyzed gene trees and **number of reticulation you expect to occur** (you can set up for test e.g. 3 instead of original 5). Using **'-a'** you can specify which individuals belongs to which species (similarly like in *BEAST), **'-pl'** moderates the number of processors used for the computations. By **'-di'** and name of the file you indicates that you want to save the results into this file (however it sometimes not working, do you have to copy the results from the screen).
    - Save your input nexus file in your PhyloNet working directory and add/copy into the same directory also Phylonet java executable (PhyloNet_3.6.0.jar).
    - PhyloNet needs to be run from command line. So in Windows write 'cmd' into search field and you should see icon for 'Command prompt'. Open it by clicking. Change directory to your working directory using:

```
cd X:\PATH\TO\YOUR\WORKING\DIRECTORY
```

- **PhyloNet needs Java** to run so you should have your java executable in a PATH to be able to call java from any location of your computer. In your working type `java -version` to see if you Java is available. If you get some error message Java can't be found. Either add the Java executable into the PATH (check some online help) or just copy Java executable into your working directory. Java executable can be ususally found in folder similar to C:\Program Files(x86)\Java\jre.1.8.x_xx\bin.
- Run PhyloNet using command

```
java -jar PhyloNet_3.6.0.jar input_file.nexus
```

- Analysis based on two trees and 3 reticulation could last for ca 1 hour, depends on the parameters of you computer.
- At the end of the analysis you will get the details of the **5 most likely networks and their log probabilities** on the screen. Copy those results and save it to the text file (e.g. via NotePad).
- To visualize the results using Dendroscope you will need just a smaller part of those results you just copied. Create a new empty file and copy there only network topologies that follows after words „**Visualize in Dendroscope :** „. Save the network topologies under new name with suffix *.nwk.

- Open **Dendroscope.exe** and using **File → Open** load the file with network topologies you just created. You will see five networks that should be (in ideal case) consistent in topology. To the plausible hybrids lead more than one blue branch.
- If you don't want to wait for your own results you can check the attached file

EXERCISE: Try to run PhyloNet analysis with more gene trees in input file and compare the results. Check also differences between gene trees and their resolution (amount of species in polytomy etc.). If you encountered differences between results of particular PhyloNet analyses check also the probabilities of particular networks and think about the possible reasons.