Phylogenetics and Molecular Evolution/Filogenética e Evolução Molecular

Octávio S. Paulo
Computational Biology and Population Genomics Group (CoBiG2)

Ciências ULisboa
Faculdade de Ciências da Universidade de Lisboa

CoBiG²
http://cobig2.fc.ul.pt
Computational
Biology & Population
Genomics **G**roup

cE3c
centre for ecology, evolution
and environmental changes

Software for computing phylogenetic networks.

SplitsTree4

Link to BETA version of SplitsTree5 (for testing purposes).

Link to talk given at ISMB18.

Description

SplitsTree4 is the leading application for computing unrooted phylogenetic networks from molecular sequence data. Given an alignment of sequences, a distance matrix or a set of trees, the program will compute a phylogenetic tree or network using methods such as split decomposition, neighbor-net, consensus network, super networks methods or methods for computing hybridization or simple recombination networks.

SplitsTree4 is an all-new implementation of the SplitsTree3 software written in Java. Superficially, this new program is similar to previous versions of this software. However, there are substantial differences. The new program has many new features, see the manual for details.

If you use SplitsTree in work in any way, then please cite the following paper: D. H. Huson and D. Bryant, **Application of Phylogenetic Networks in Evolutionary Studies**, Mol. Biol. Evol., 23(2):254-267, 2006.
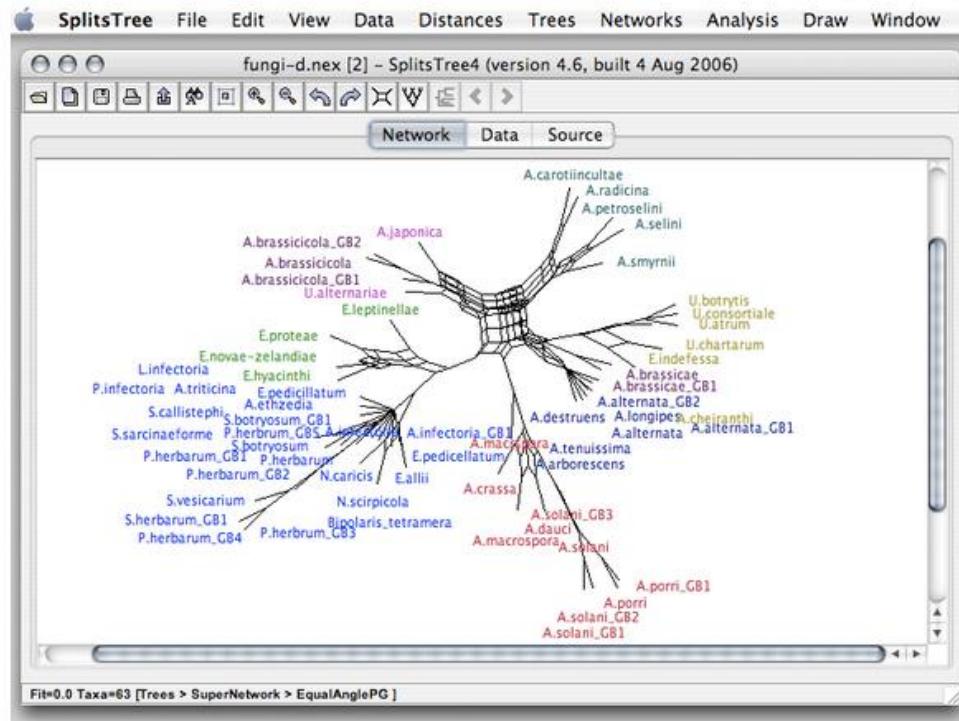

Huson and Bryant 2006 MBE

2

SplitsTree 4

Daniel H. Huson and David Bryant

September 22, 2023
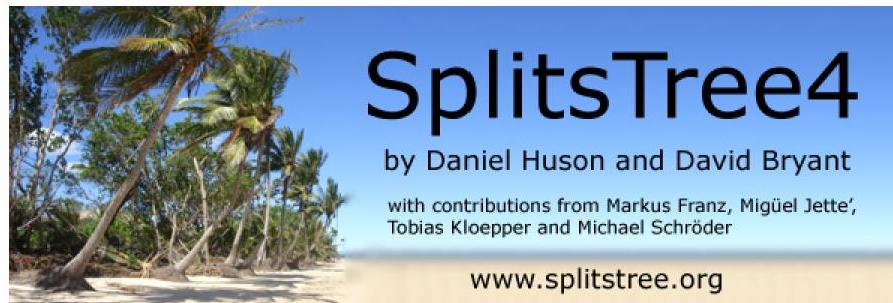
SplitsTree 4
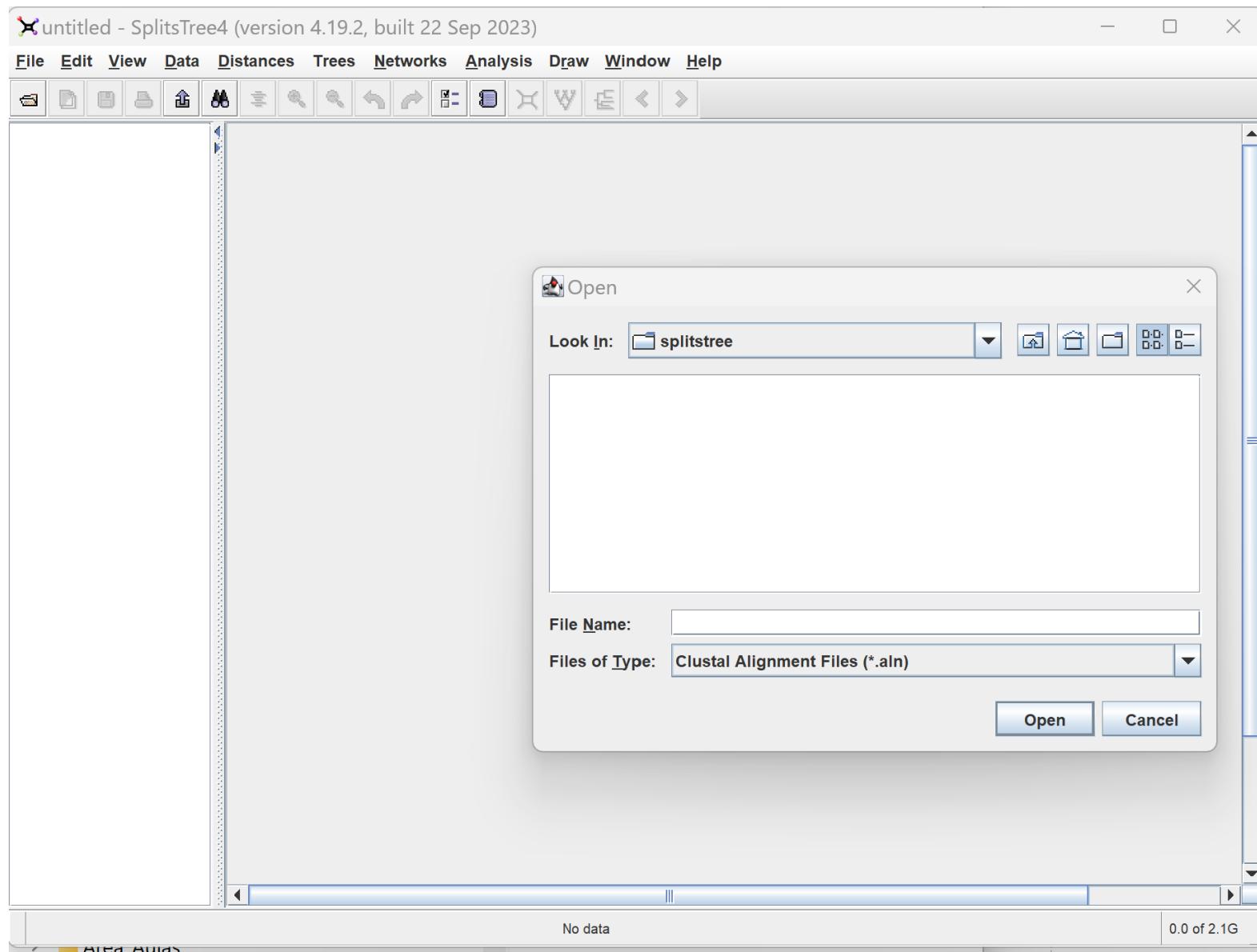
SplitsTree 4

9 Building and Drawing Networks

The `Network` menu provides methods for computing phylogenetic networks from character sequences, distances and trees.

Methods that compute a `split` `network` directly from character sequences provided in the `Characters` block are `ParsimonySplits`, `MedianNetwork`, `MedianJoining` and `SpectralSplits`. Note that the Median network method requires binary sequences. However, given DNA or RNA, this program will detect all sites that contain precisely two character-states and will build a Median network from these.

The MedianJoining method computes an unrooted network from binary sequences, DNA and other multi-state sequences. This is an implementation of the algorithm described in [4]. In the case of non-binary sequences, the resulting network will not be a split network.

Two methods for computing split networks from distances provided in the `Distances` block are `SplitDecomposition` and `NeighborNet`.

If a set of phylogenetic trees in the `Trees` block all contain the full set of taxa listed in the `Taxa` block, then the `ConsensusNetwork` method can be applied to compute a consensus network. If, however, the `Trees` block contains partial trees, that is, trees that do not necessarily all involve identical sets of taxa, then the `SuperNetwork` or `FilteredSuperNetwork` method can be used to compute a super network.

The *Draw* menu determines which algorithm is used to construct the final visualization of the `tree` or `network`. Existing methods are `EqualAngle`, `RootedEqualAngle` and `Phylogram`. Additionally, the `Draw→Hide Selected Splits` can be used to remove selected splits from the network and the `Draw→Select Trees` can be used to highlight different trees contained in a split network.

SplitsTree 4

[1] Sarah C. Ayling and Terence A. Brown. Novel methodology for construction and pruning of quasi-median networks. *BMC Bioinformatics*, 9:115, 2008.

[2] H.-J. Bandelt and A. W. M. Dress. A canonical decomposition theory for metrics on a finite set. *Advances in Mathematics*, 92:47–105, 1992.

[3] H.-J. Bandelt, P. Forster, B. C. Sykes, and M. B. Richards. Mitochondrial portraits of human population using median networks. *Genetics*, 141:743–753, 1995.

[4] Hans-Jürgen Bandelt, Peter Forster, and Arne Röhl. Median-joining networks for inferring intraspecific phylogenies. *Molecular Biology and Evolution*, 16:37–48, 1999.

starBeast

File   template starBeast3

Octávio S. Paulo - Filogenética

starBeast

File template starBeast3

Octávio S. Paulo - Filogenética
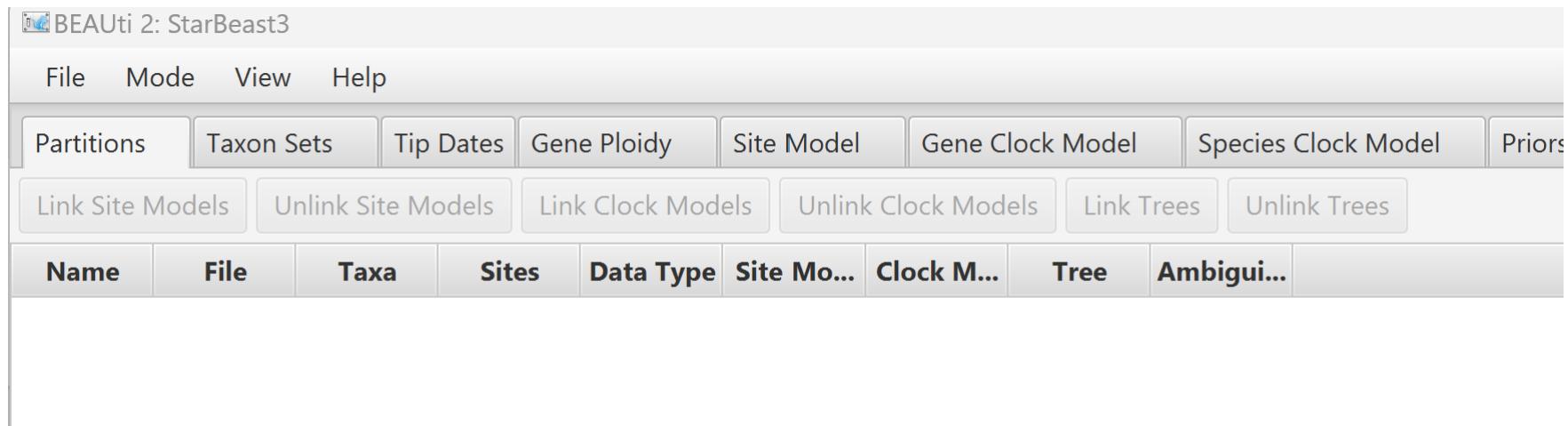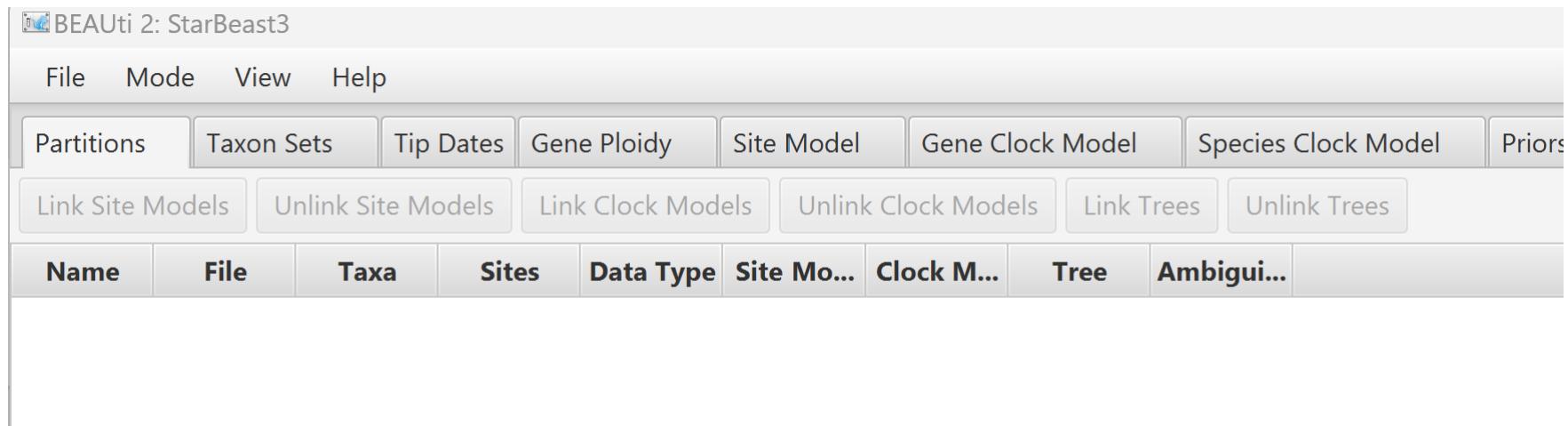
starBeast

File Import alingments   (nexus, um por gene)

Octávio S. Paulo - Filogenética

starBeast

Default unlink

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 a *BEAST analysis. It could also be that for this analysis a linked model would perform better.

Taxon set

Taxon set

Taxon set

Gene Ploidy

Octávio S. Paulo - Filogenética

Site model

Octávio S. Paulo - Filogenética

Site model

Octávio S. Paulo - Filogenética

Site model

Octávio S. Paulo - Filogenética

Gene clock model

Octávio S. Paulo - Filogenética

Species clock model

Species clock model

Octávio S. Paulo - Filogenética

Priors

Octávio S. Paulo - Filogenética

Priors

Priors

Octávio S. Paulo - Filogenética

Priors

Octávio S. Paulo - Filogenética

MCMC

500/5000
Octávio S. Paulo - Filogenética

MCMC

speciesTreeLogger

File Name	gopher2.species.trees.txt
Log Every	10000
Mode	tree
Sort	none
☐ Sanitise Headers	
☐ SpeciesTreeLoggerX	

5000

Octávio S. Paulo - Filogenética

screenlog

File Name	gopher2.txt
Log Every	10000
Mode	autodetect
Sort	none

5000

Xml to beast

Save xml in folder where results are going to be

Octávio S. Paulo - Filogenética

starBeast

Operator	Tuning	#accept	#reject	Pr(m)	Pr(acc|m)	
starbeast3.operators.NodeReheight2(Reheight.t:Species)	-	135835	374296	0.10224	0.26627	
starbeast3.operators.CoordinatedExponential(CoordinatedExponential.t:Species)	0.00358	323306	345760	0.05112	0.48322	
starbeast3.operators.CoordinatedUniform(CoordinatedUniform.t:Species)	-	254745	374765	0.10224	0.40467	
kernel.BactrianScaleOperator(TreeRootScaler.t:Species)	0.19119	5548	45552	0.01022	0.10857	
kernel.BactrianNodeOperator(BactrianNodeOperator.t:Species)	0.97070	59922	125784	0.01022	0.32267	
AdaptableOperatorSampler(AdaptableTopologyOperator.lengths.Species)	-	596564	1107649	0.34079	0.35005	
starbeast3.operators.PopSizeGibbsSampler(PopSizeGibbsSampler.Species)	-	851018	0	0.17040	1.00000	
AdaptableOperatorSampler(AdaptableOperatorSampler.popmean:Species)	-	22738	62841	0.01704	0.26570	
kernel.BactrianScaleOperator(clockRateScaler.c:29)	0.31773	447	1217	0.00034	0.26863	
kernel.BactrianScaleOperator(clockRateScaler.c:47)	0.25029	413	1278	0.00034	0.24423	
kernel.BactrianScaleOperator(KappaScaler.s:26)	0.61274	448	1197	0.00034	0.27234	
kernel.BactrianScaleOperator(KappaScaler.s:29)	0.63997	436	1222	0.00034	0.26297	
kernel.BactrianScaleOperator(KappaScaler.s:47)	0.48794	480	1243	0.00034	0.27858	
AdaptableOperatorSampler(AdaptableOperatorSampler.YuleModel.t:Species)	-	25544	59911	0.01704	0.29892	
Uniform(UniformOperator.t:Species)	-	11553	39800	0.01022	0.22497	
WilsonBalding(WilsonBalding.t:Species)	-	2078	254007	0.05112	0.00811	
Exchange(Wide.t:Species)	-	5521	251025	0.05112	0.02152	
Exchange(Narrow.t:Species)	-	3089	14078	0.00341	0.17994	
kernel.BactrianSubtreeSlide(BactrianSubtreeSlide.t:Species)	0.00251	20430	234698	0.05112	0.08008	Try decreasing size to about 0.001
starbeast3.operators.ParallelMCMCTreeOperator(ParallelMCMCTreeOperator)	-	0	0	0.01000	NaN	

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).

Total calculation time: 331.362 seconds
End likelihood: -3937.277984628074

Octávio S. Paulo - Filogenética

trace

Treeannotator

Input .trees
Output .tree

Octávio S. Paulo - Filogenética

Figtree

Densitree

Octávio S. Paulo - Filogenética