Phylogenetic tree building methods

method of building trees	type of data: distances	type of data: DNA sequences or other characters
clustering algorithm	UPGMA neighbor-joining tree	
optimality criterion	minimum evolution tree	parsimony maximum likelihood Bayesian analysis

Maximum likelihood (ML) method

The likelihood of an evolutionary tree (L) is the probability that the observed DNA sequences will evolve in a given tree topology and in a given evolutionary model.
The goal of the maximum likelihood method is to find a tree with the largest possible L. In this method, the starting tree is constructed using the parsimony method or the nearest neighbor method, and then the nearby trees are searched for a more likely tree than the one we have available.
It is also possible to search all possible trees, but this, as with the maximum parsimony trees, is time consuming and realistic only with fewer sequences.

Maximum likelihood (ML) methods

Optimality criterion: **ML methods evaluate phylogenetic hypotheses in terms of the probability that a proposed model of the evolutionary process and the proposed unrooted tree would give rise to the observed data. The tree found to have the highest ML value is considered to be the preferred tree.**

Advantages:

- Are based on explicit model of evolution.
- Usually the most 'consistent' of the methods available.
- Can be used for character (can infer the exact substitutions) and rate analysis.
- Can be used to infer the sequences of the extinct (hypothetical) ancestors.
- Can help account for branch-length effects.

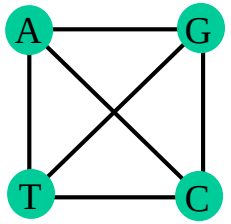
Disadvantages:

- Are based on explicit model of evolution.
- Are not as simple and intuitive as many other methods.
- **Are computationally very intense** (Iimits number of taxa and length of sequence).
- **Slow**
- Violations of the assumed model can lead to incorrect trees.

Models of the evolution of DNA sequences

Parameters that affect sequence evolution:

- base frequencies
- types of substitution (transitions, transversions)
- heterogeneity of the rate of substitutions



purines

pyrimidines

Classical substitution models:

Jukes-Cantor (1969)

Kimura 2 parameter (1980)

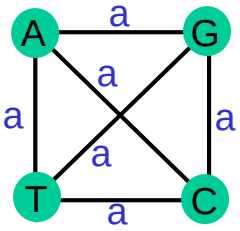
Felsenstein (1981)

Hasegawa, Kishino & Yano (1985)

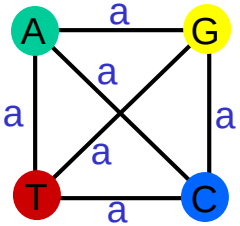
General time-reversible model (Lavane et al. 1984)

Each model is based on different assumptions

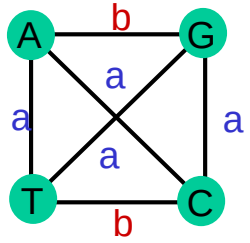
Substitution models



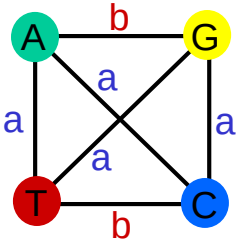
JC – equal substitution rates;
equal base frequencies



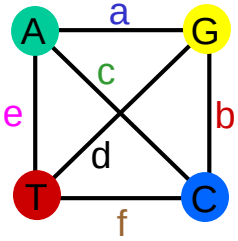
F81 – equal substitution rates;
unequal base frequencies



K2P – two different substitution rates;
equal base frequencies



HKY – two different substitution rates;
unequal base frequencies



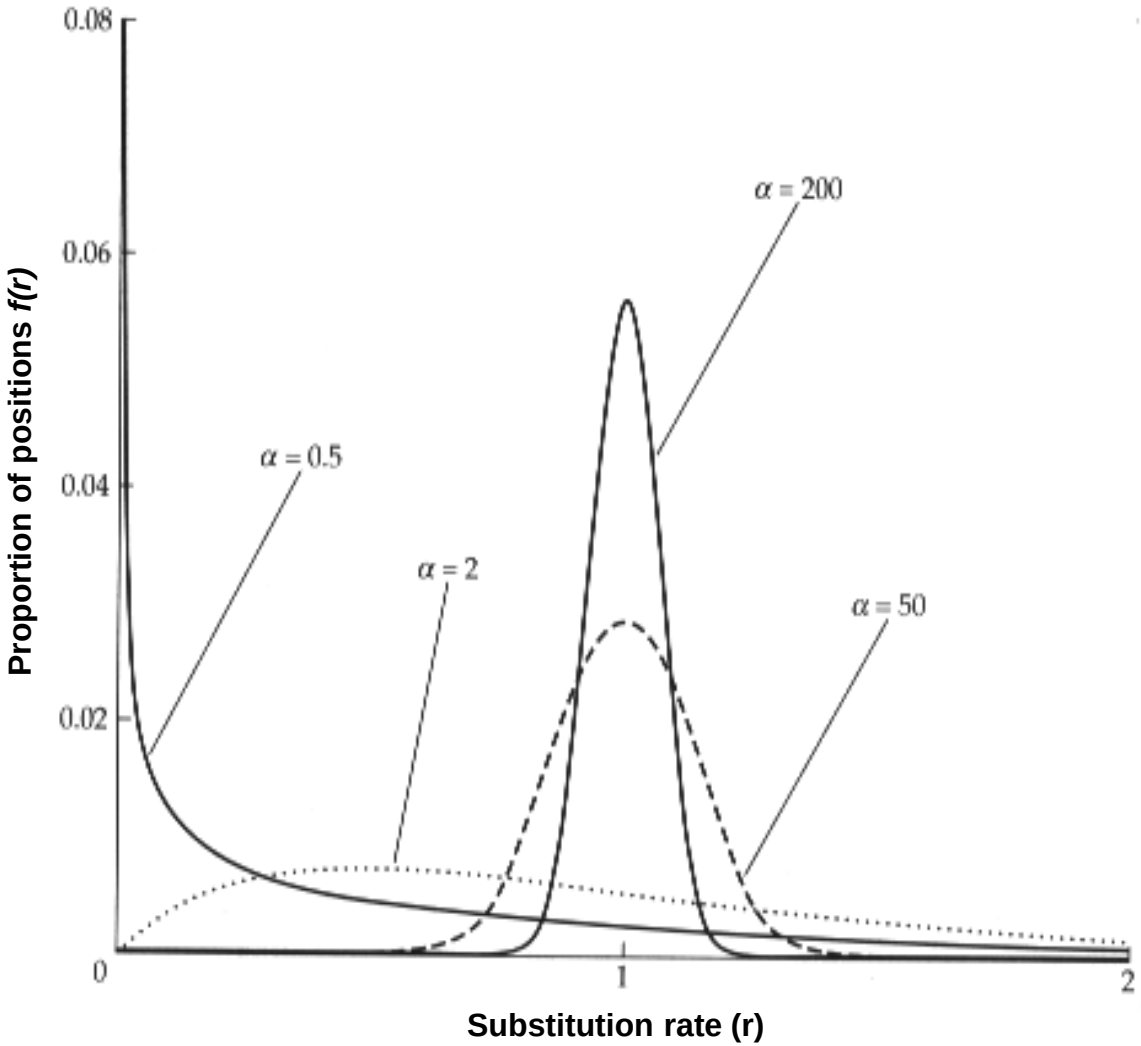
GTR – 6 different substitution rates;
unequal base frequencies

Increasing amount of model parameters

Function of **gama distribution (Γ)** *f(r)* substitution rate on positions of DNA sequences

α – shape parameter, α ≤ 1 – the function has a L shape, α > 1 – the function has a bell-shaped shape

β – scale parameter

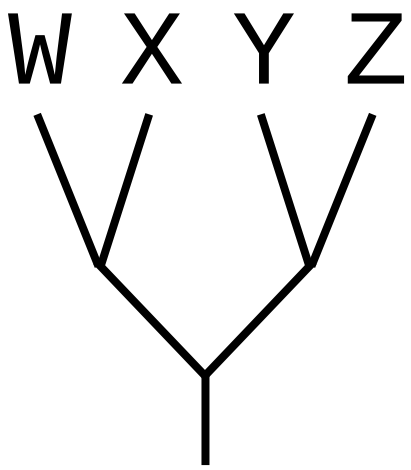


Yang, Z., 1996, Trends Ecol. Evol. 11: 367-372.

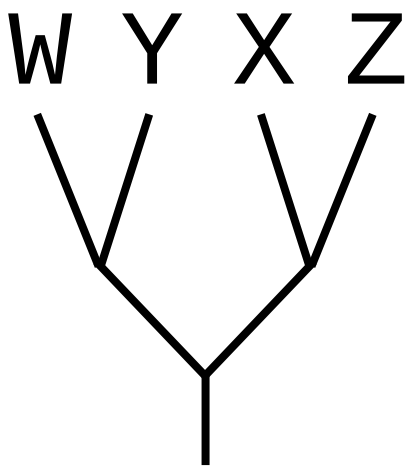
Maximum likelihood (ML) method

Possible trees

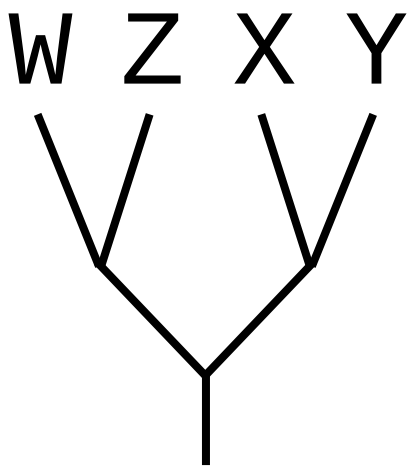
```
Sequence W: A C G C G T T G G G
Sequence X: A C G C G T T G G G
Sequence Y: A C G C A A T G A A
Sequence Z: A C A C A G G G A A
```



Tree1



Tree 2

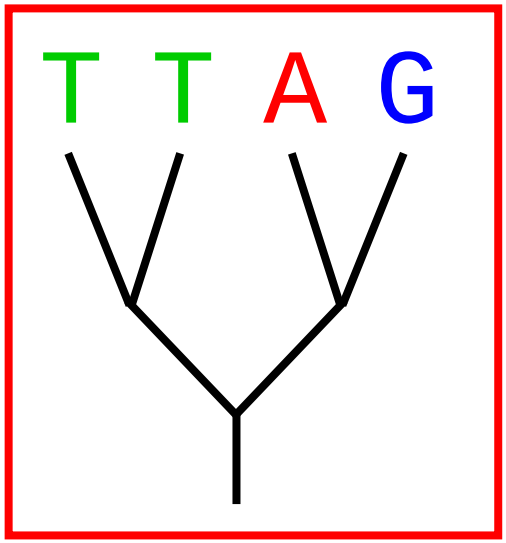


Tree 3

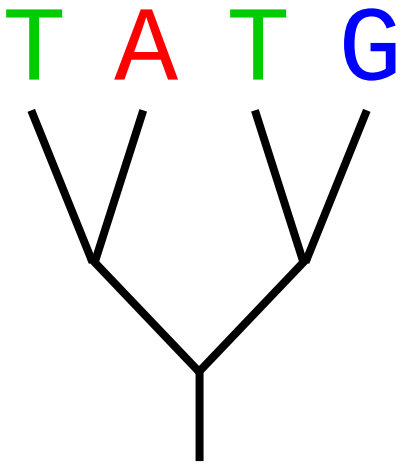
Maximum likelihood (ML) method

Possible trees

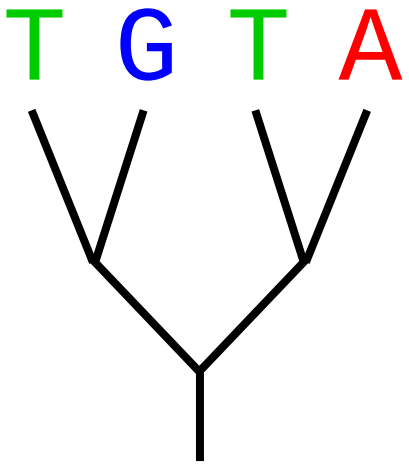
```
Sequence W: A C G C G T T G G G
Sequence X: A C G C G T T G G G
Sequence Y: A C G C A A T G A A
Sequence Z: A C A C A G G G A A
```



Tree 1

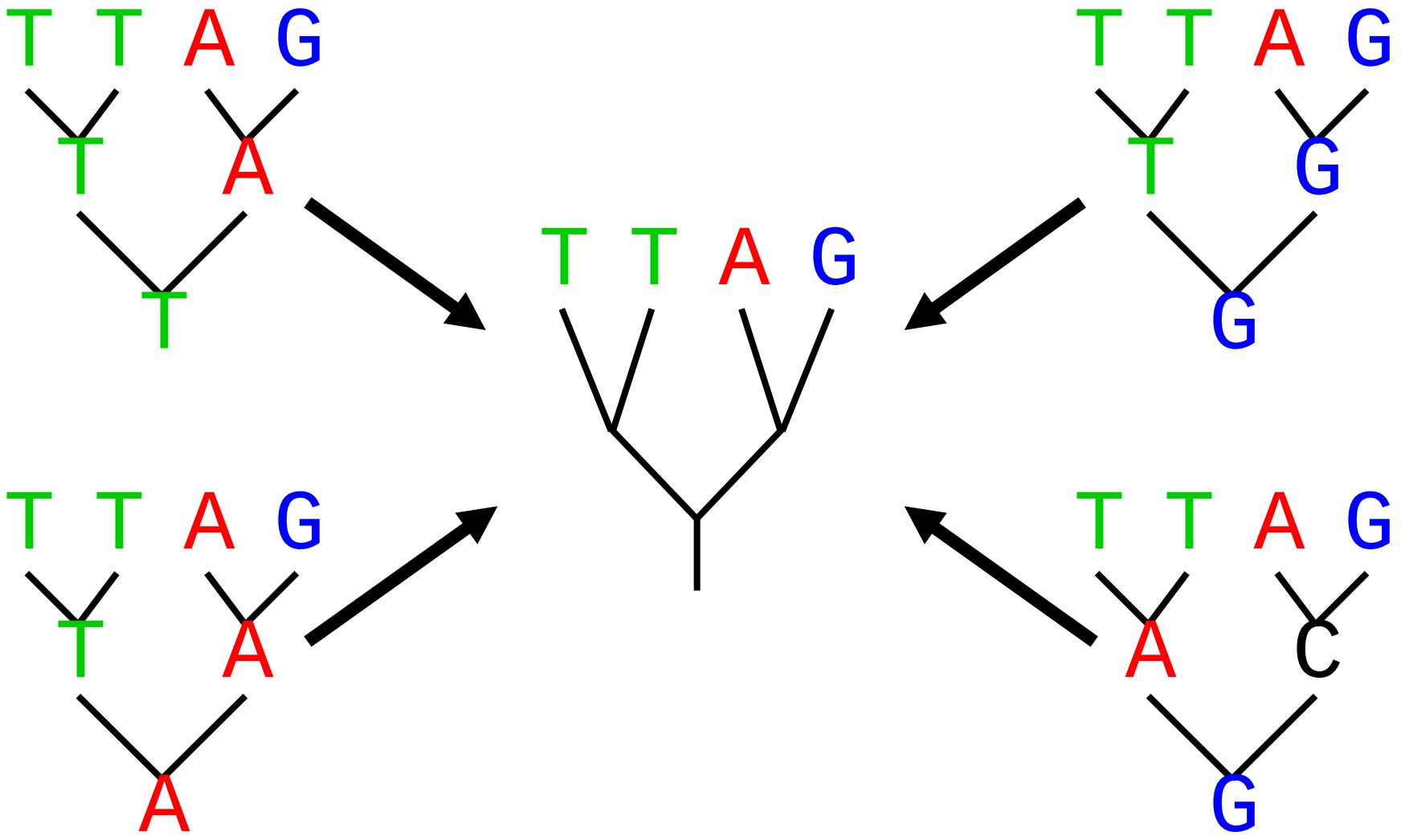


Tree 2



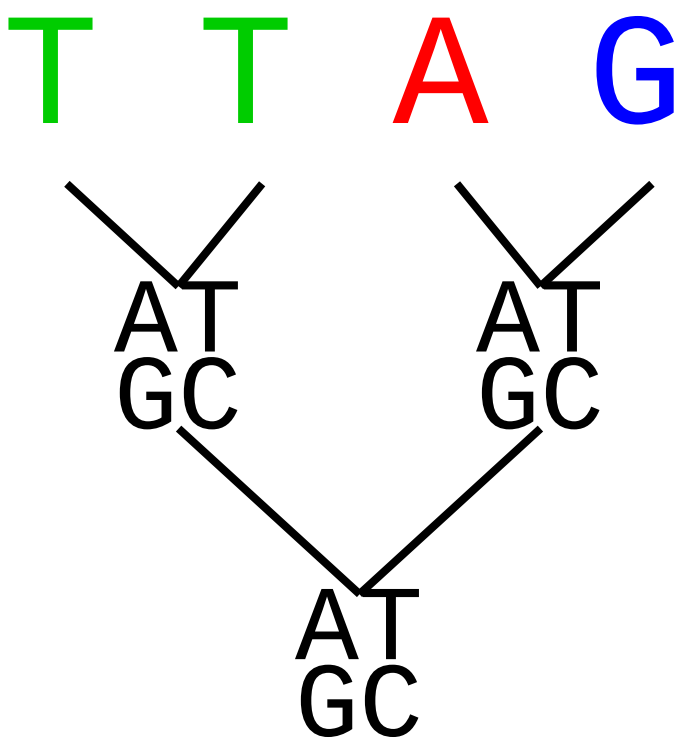
Tree 3

Maximum likelihood (ML) method
Some possible evolutionary paths (to the same tree)



Maximum likelihood (ML) method

Some possible evolutionary paths (to the same tree)



# of Possible Paths / OTU / Position:

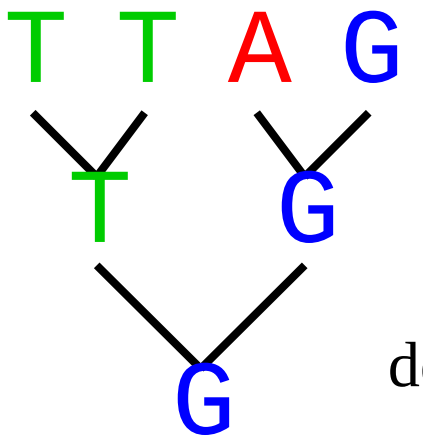
$= (\text{Number of States})^{(\text{Number of Nodes})}$

$= (\text{Number of States})^{(\text{Number of OTU } -1)}$

$= 4^3 = 64$

Maximum likelihood (ML) method

One path likelihood



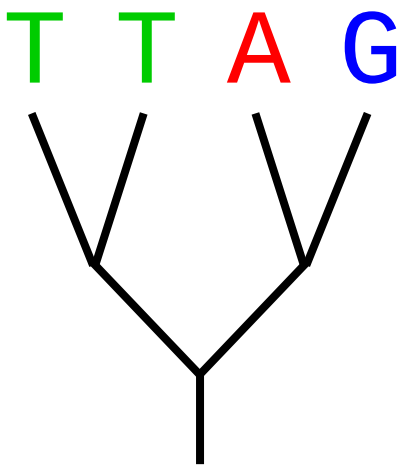
depends on particular model

$L(path) = L(root) \times \Pi\, L(branches)$

$= [\pi_G] \times [1-u(a\pi_G+b\pi_G+c\pi_G)]^2 \times [1-u(a\pi_T+b\pi_T+c\pi_T)]^2 \times [uf\pi_G] \times [ub\pi_G]$

Maximum likelihood (ML) method

Likelihood of one tree from one position of the DNA sequence



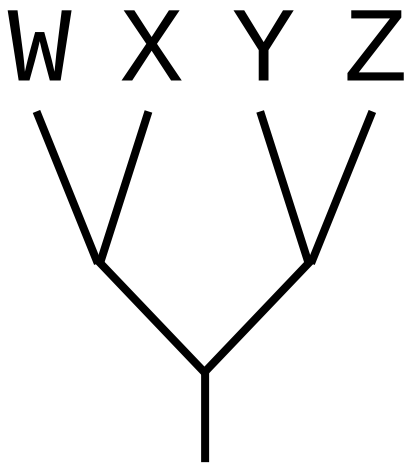
L(one position tree1) = Σ L(all possible Evolutionary Paths to tree 1)

= L(tree1) + L(tree2) + L(tree3) + … + L(tree64)

- trees of the same topology in terms of taxa at the ends of the branches but with different nucleotides in the nodes

Maximum likelihood (ML) method

Likelihood of one tree from the whole DNA sequence



L(Sequence tree) = Π L(tree of particular position)

Choose the tree with the Maximum Likelihood.

(# possible trees)x(seq length)x(# states)^(# OTUs)

Maximum likelihood (ML) method

ML analysis starts with a tree created e.g. NJ and then search the nearby trees for those with higher credibility.

For larger data files, heuristic ML analysis is very computer intensive, in addition, support for individual clades should be verified by the bootstrap method, which multiplies the time required for analysis.

If the ML analysis finds a local optimum, it does not mean that it is also a global optimum in the tree space.

Maximum likelihood, ML method

ML analysis includes:
- testing of evolutionary models for a given data set, calculation of scores (log likelihood scores) with respect to the model and data - jmodeltest2
- selection of a suitable model based on log likelihood scores according to certain test criteria - jmodeltest2
- calculation (search) of the most probable tree according to selected model - PAUP

jmodeltest2 (https://github.com/ddarriba/jmodeltest2/releases)

- program for calculation and comparison of log likelihood scores of tested models, selection of a suitable evolutionary model for a given data set

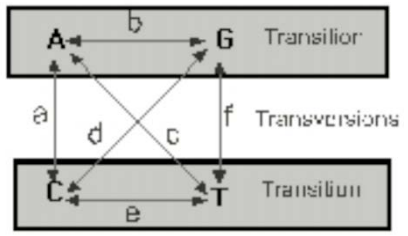
Darriba D, Taboada GL, Doallo R, Posada D. 2012. jModelTest 2: more models, new heuristics and parallel computing. Nature Methods 9(8), 772.

Guindon S and Gascuel O (2003). A simple, fast and accurate method to estimate large phylogenies by maximum-likelihood". Systematic Biology 52: 696-704.

models are expressed by parameters: - base frequencies, - types of substitution and their rates, - homogeneity / heterogeneity of mutation rates at different positions, - proportion of invariable positions

Table 2. Model parameters. The substitution codes are just two ways of indicating the substitution scheme. Any of these models can ignore rate variation or include invariable sites (+I), rate variation among sites (+G), or both (+I+G).

Model	Free parameters	Base frequencies	Substitution rates	Substitution code 1	Substitution code 2
JC	0	equal	a=b=c=d=e=f	000000	aaaaaa
F81	3	unequal	a=b=c=d=e=f	000000	aaaaaa
K80	1	equal	a=c=d=f, b=e	010010	abaaba
HKY	4	unequal	a=c=d=f, b=e	010010	abaaba
TNef	2	equal	a=c=d=f, b, e	010020	abaaca
TN	5	unequal	a=c=d=f, b, e	010020	abaaca
K81	2	equal	a=f, c=d, b=e	012210	abccba
K81uf	5	unequal	a=f, c=d, b=e	012210	abccba
TIMef	3	equal	a=f, c=d, b, e	012230	abccda
TIM	6	unequal	a=f, c=d, b, e	012230	abccda
TVMef	4	equal	a, c, d, f, b=e	012314	abcdbe
TVM	7	unequal	a, c, d, f, b=e	012314	abcdbe
SYM	5	equal	a, c, d, f, b, e	012345	abcdef
GTR	8	unequal	a, c, d, f, b, e	012345	abcdef

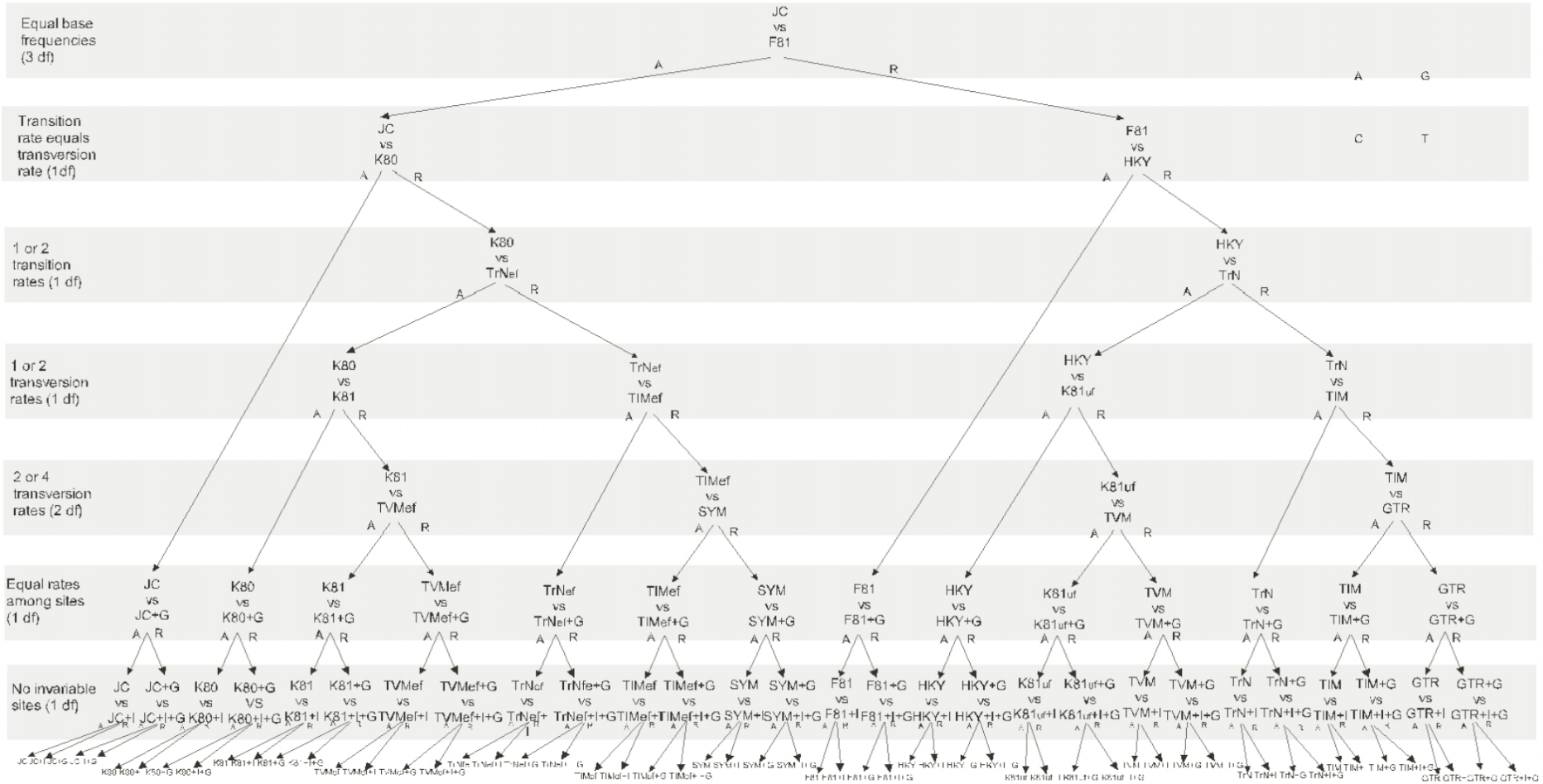


Mutation rates a(A-C), b(A-G), c(A-T), d(C-G), e(C-T), f(G-T)

Table 1. Model names. Some models have no reference (TNef, K81uf, TIMef, TIM, TVMef, TVM), they are just some variations of some existing models, and they were no developed, only named, by D. Posada.

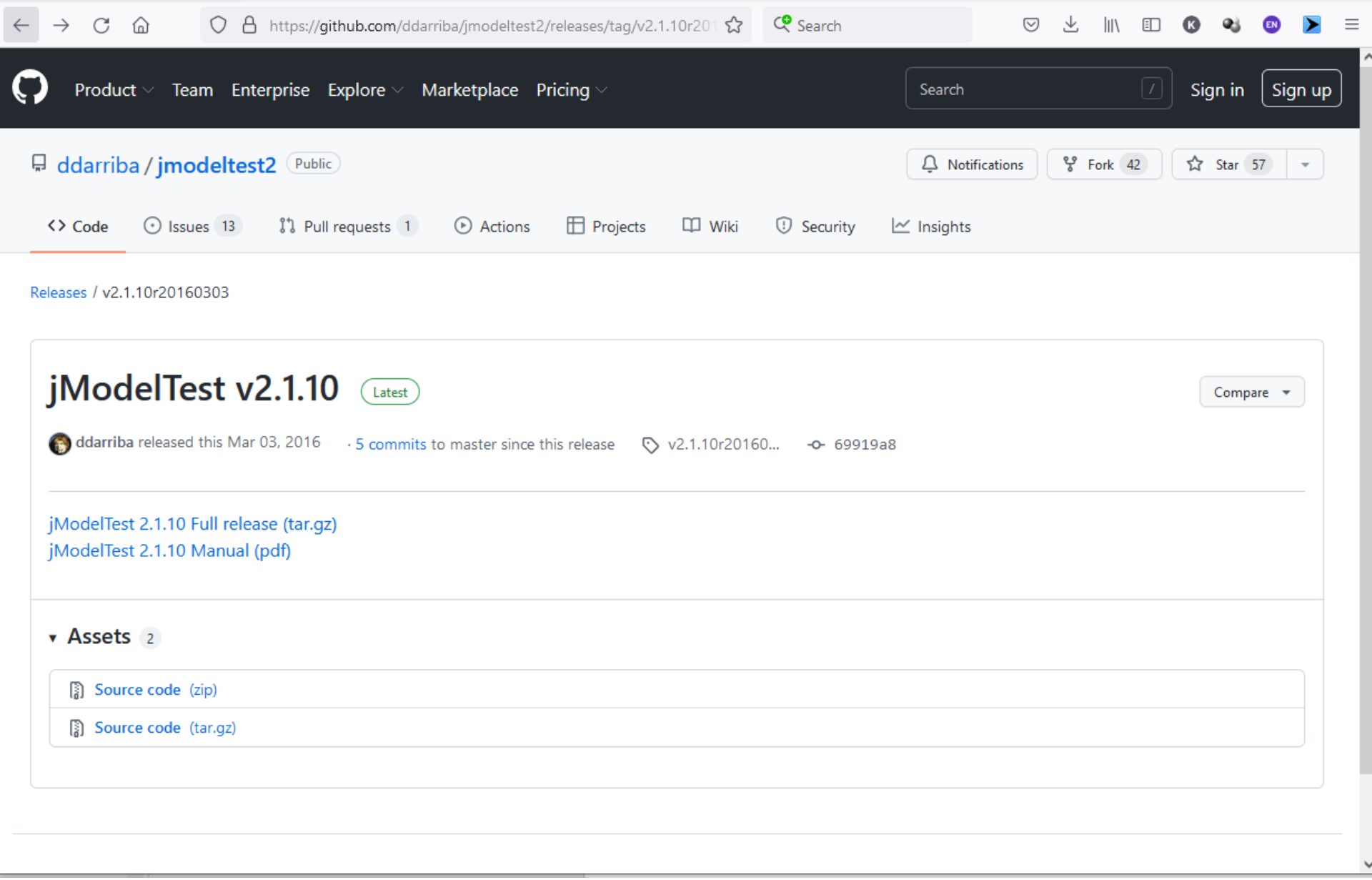
Model	Name
JC	Jukes and Cantor (Jukes and Cantor, 1969)
F81	Felsenstein 81 (Felsenstein, 1981)
K80	Kimura 80 (=K2P) (Kimura, 1980)
HKY	Hasegawa, Kishino, Yano 85 (Hasegawa, Kishino and Yano, 1985)
TNef	Tamura-Nei equal frequencies
TN	Tamura-Nei (Tamura and Nei, 1993)
K81	Two transversion-parameters model 1 (=K81=K3P) (Kimura, 1981)
K81uf	Two transversion-parameters model 1 unequal frecuencies
TIMef	Transitional model equal frequencies
TIM	Transitional model
TVMef	Transversional model equal frequencies
TVM	Transversional model
SYM	Symmetrical model (Zharkihk, 1994)
GTR	General time reversible (=REV) (Tavaré, 1986)

Modeltest 3.0 hierarchy

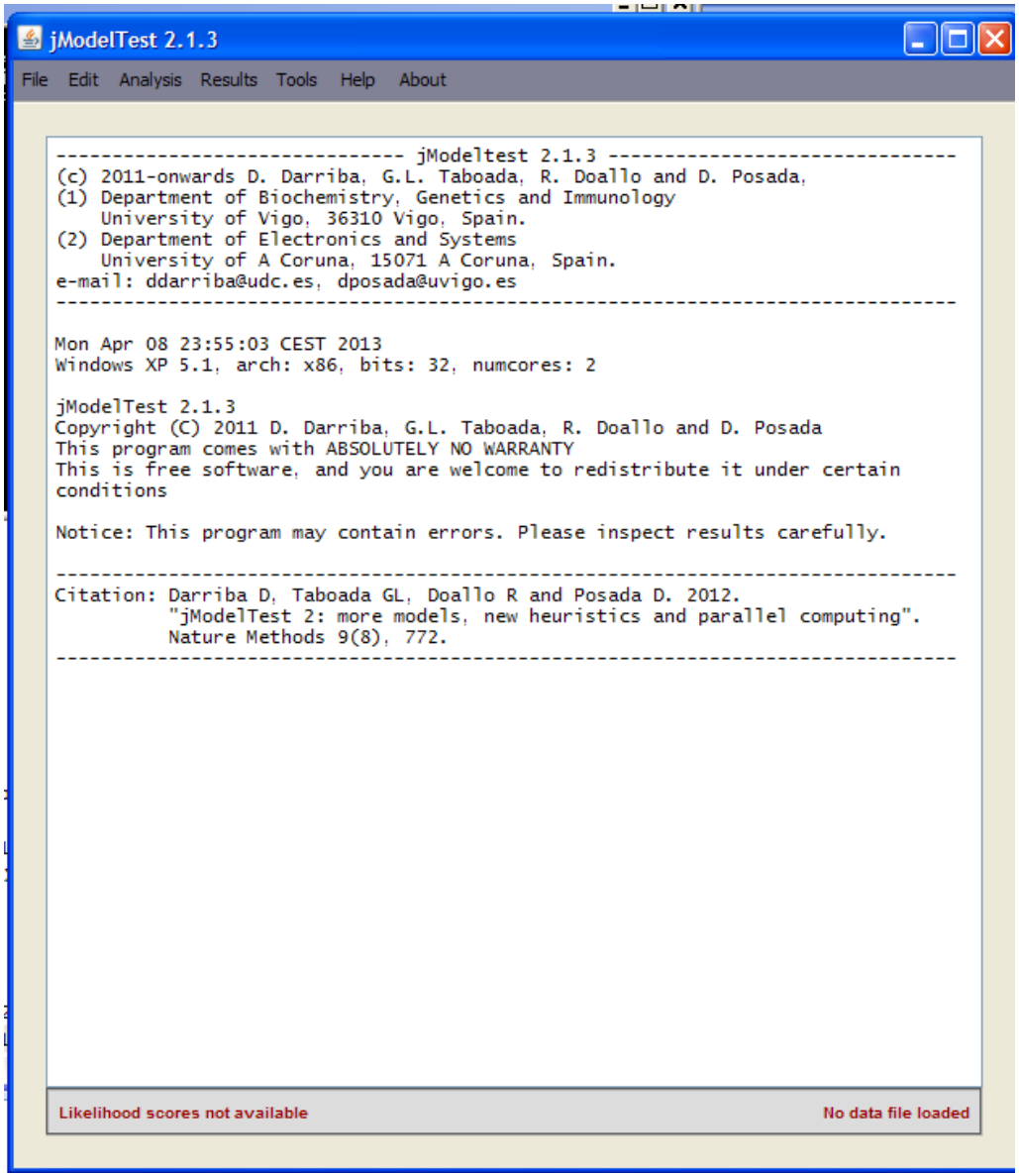


	JC	K80	TrNef	K81	TVMef	TIMef	SYM	F81	HKY	TrN	K81uf	TVM	TIM	GTR
Base frequencies	$p_A = p_C = p_G = p_T$	$p_A = p_C = p_G = p_T$	$p_A = p_C = p_G = p_T$	$p_A = p_C = p_G = p_T$	$p_A = p_C = p_G = p_T$	$p_A = p_C = p_G = p$	$p_A = p_C = p_G = p$	p_A, p_C, p_G, p_T	$p_A = p_C = p_G = p$	$p_A = p_C = p_G = p_T$	p_A, p_C, p_G, p_T	p_A, p_C, p_G, p_T	p_A, p_C, p_G, p_T	p_A, p_C, p_G, p_T
Substitution rates	a=b=c=d=e=f	a=c=d=f, b=e	a=c=d=f, b, e	a=f, b=e, c=d	a, c, d, f, b=e	a=f, c=d, b, c	a, b, c, d, e, f	a=b=c=d=e	a=c=d=f, b=c	a=c=d=f, b, e	a=f, b=e, c=d	a, c, d, f, b=e	a=f, c=d, b, e	a, b, c, d, e, f
Free Parameters	0	1	2	2	4	3	5	3	4	5	5	7	6	8

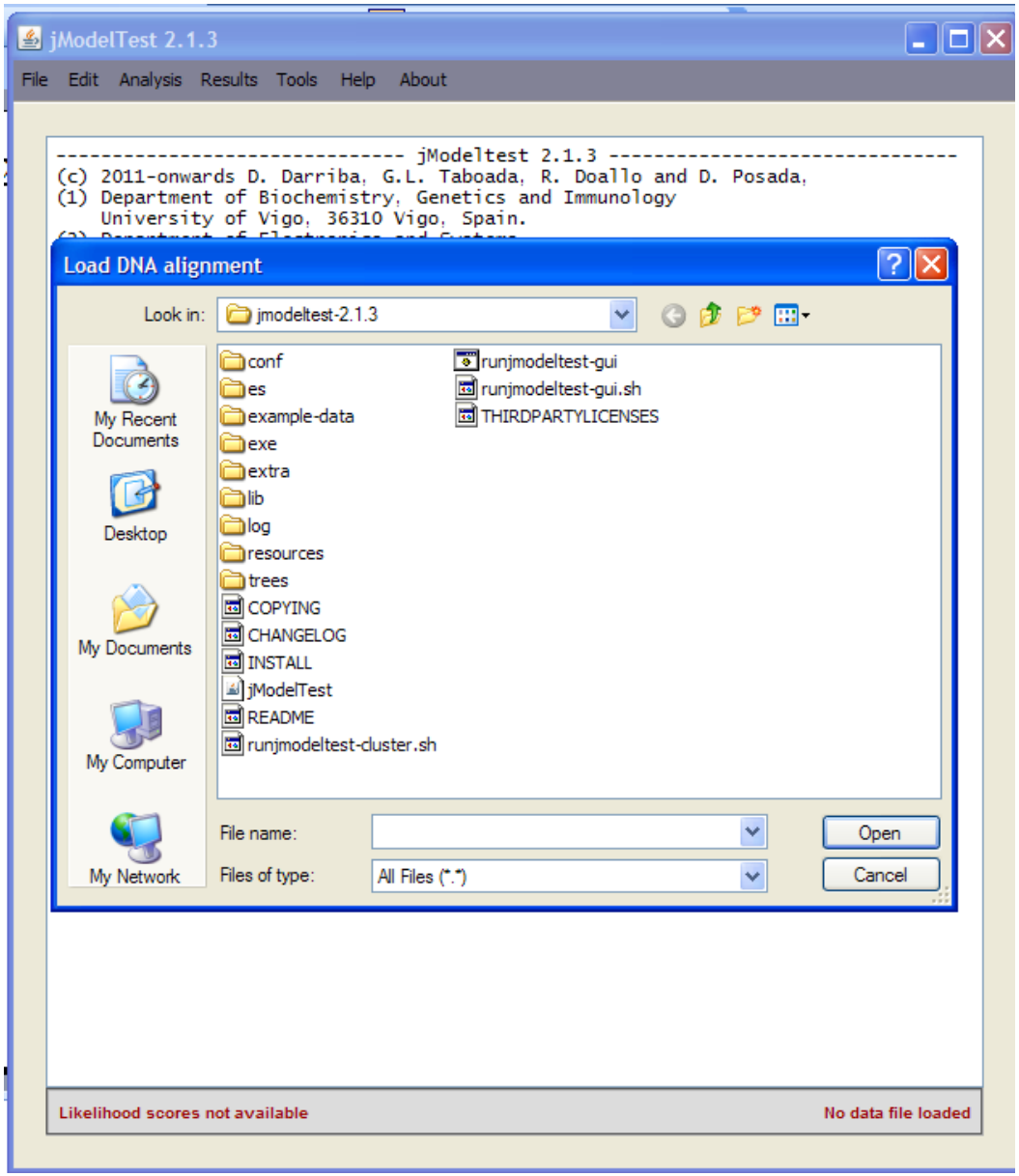
https://github.com/ddarriba/jmodeltest2/releases/tag/v2.1.10r20160303



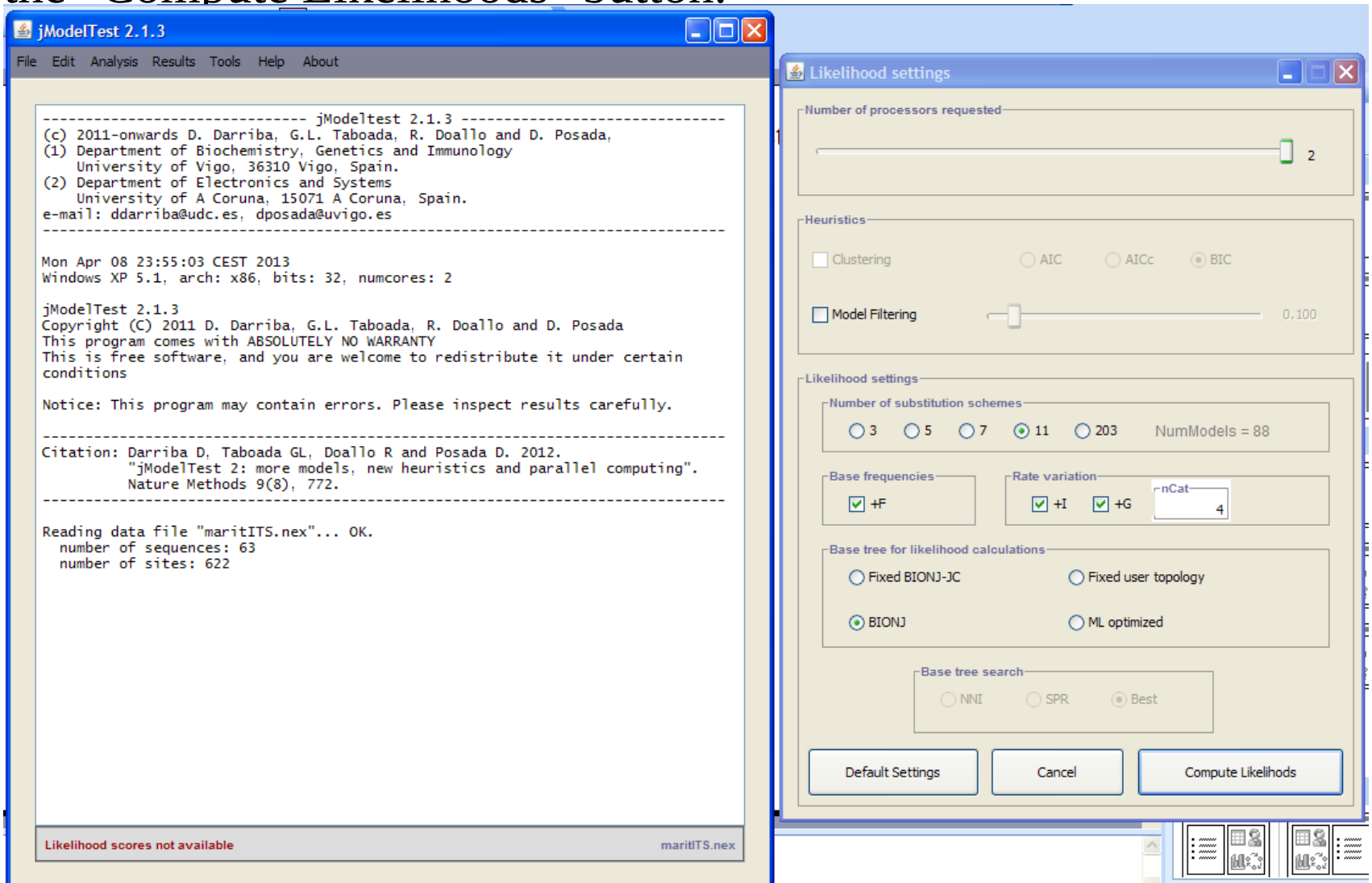
1. Execute the script for the Graphical User Interface (runjmodeltest-gui.sh). The main jModelTest frame should pop up on the screen:

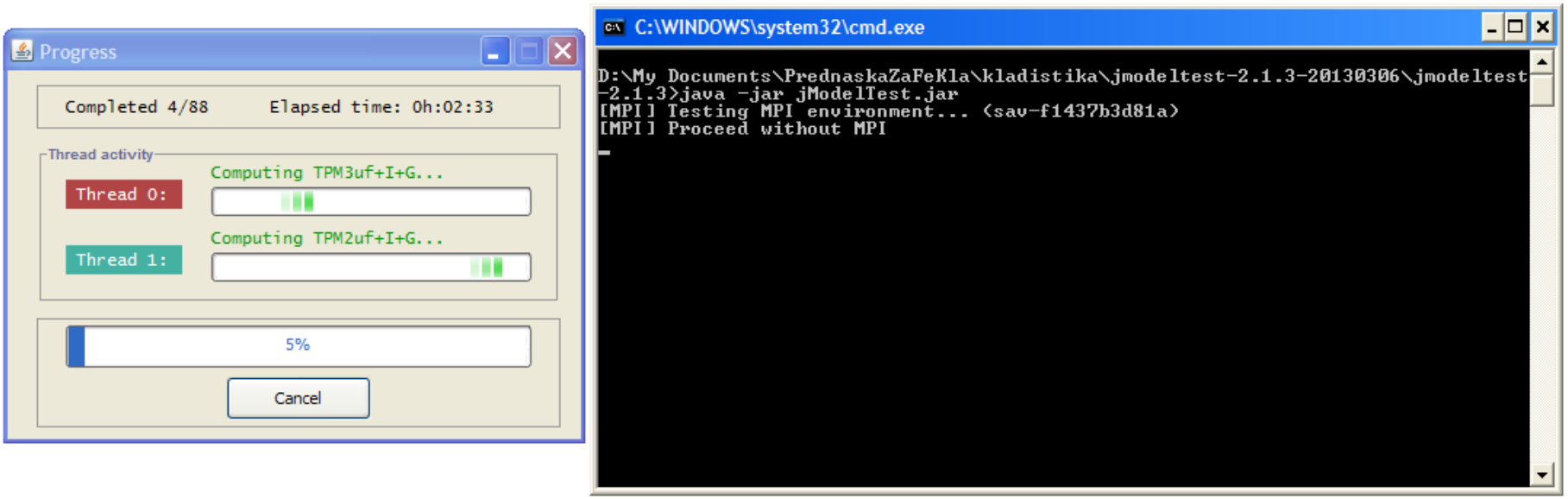


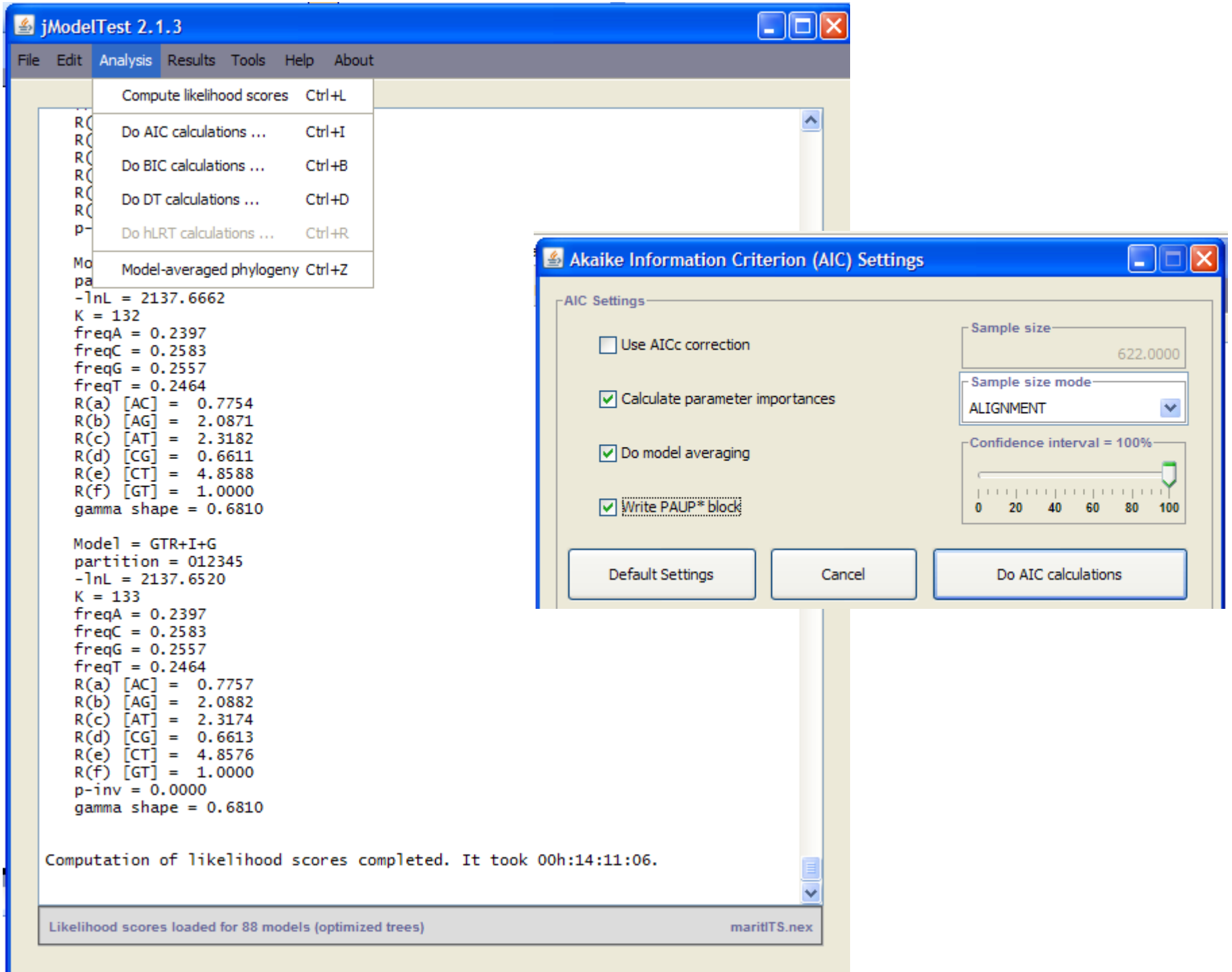
2. Load an input alignment file using the **File/Load Alignment** option.



3. Go to **Analysis/Compute Likelihood Scores** and select the candidate models and the options for model optimization (optionally you can set a base topology from a file). Press Enter or the "Compute Likelihoods" button.




Progress
Completed 4/88
Elapsed time: 0h:02:33
Thread activity
Thread 0:
Computing TPM3uf+I+G...
Thread 1:
Computing TPM2uf+I+G...
5%
Cancel
C:\WINDOWS\system32\cmd.exe
D:\My Documents\PrednaskaZaFeKla\kladistika\jmodeltest-2.1.3-20130306\jmodeltest-2.1.3>java -jar jModelTest.jar
[MPI] Testing MPI environment... (sav-f1437b3d81a)
[MPI] Proceed without MPI


jModelTest 2.1.3
File Edit Analysis Results Tools Help About
Compute likelihood scores Ctrl+L
Do AIC calculations ... Ctrl+I
Do BIC calculations ... Ctrl+B
Do DT calculations ... Ctrl+D
Do hLRT calculations ... Ctrl+R
Model-averaged phylogeny Ctrl+Z
-lnL = 2137.6662
K = 132
freqA = 0.2397
freqC = 0.2583
freqG = 0.2557
freqT = 0.2464
R(a) [AC] = 0.7754
R(b) [AG] = 2.0871
R(c) [AT] = 2.3182
R(d) [CG] = 0.6611
R(e) [CT] = 4.8588
R(f) [GT] = 1.0000
gamma shape = 0.6810
Model = GTR+I+G
partition = 012345
-lnL = 2137.6520
K = 133
freqA = 0.2397
freqC = 0.2583
freqG = 0.2557
freqT = 0.2464
R(a) [AC] = 0.7757
R(b) [AG] = 2.0882
R(c) [AT] = 2.3174
R(d) [CG] = 0.6613
R(e) [CT] = 4.8576
R(f) [GT] = 1.0000
p-inv = 0.0000
gamma shape = 0.6810
Computation of likelihood scores completed. It took 00h:14:11:06.
Likelihood scores loaded for 88 models (optimized trees)
maritITS.nex
Akaike Information Criterion (AIC) Settings
AIC Settings
Use AICc correction
Calculate parameter importances
Do model averaging
Write PAUP* block
Sample size
622.0000
Sample size mode
ALIGNMENT
Confidence interval = 100%
0 20 40 60 80 100
Default Settings
Cancel
Do AIC calculations

mutual testing of models based on log likelihood scores, selection of a suitable model according to testing criteria - AIC (Akaike information criterion), hLRT (hierarchical likelihood ratio test), BIC (Bayesian information criterion)

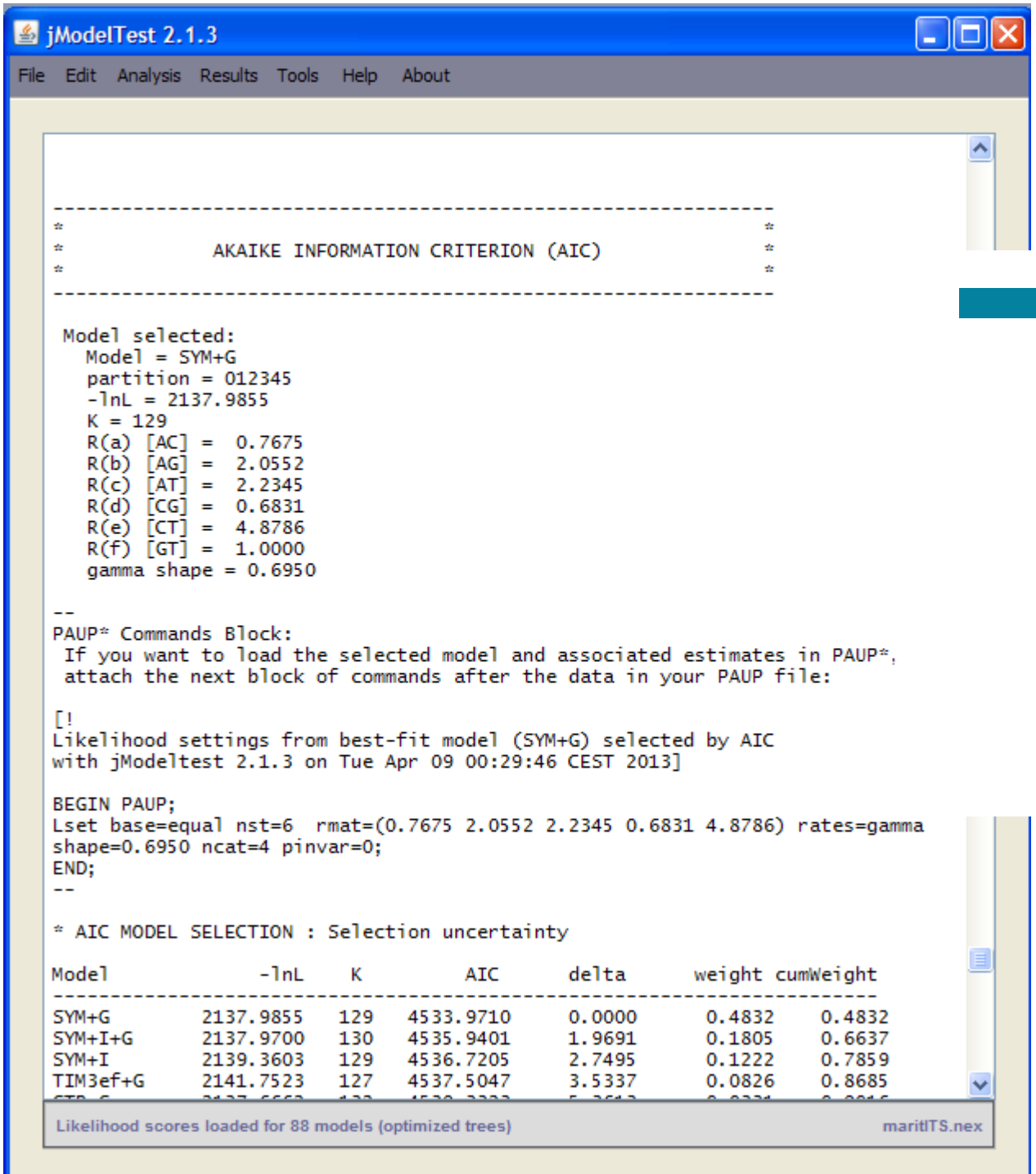
AIC - simultaneous comparison of all models

hLRT - mutual comparison of two models

BIC - calculation by Bayes method using MCMC

not the model with the highest score, but with the most optimal (i.e. if adding parameters, increasing complexity only slightly increases the score, a less complex model is chosen)

sometimes they design different models, it is up to the user who chooses (the preferred is usually AIC)



Model selected

Model	TIM3ef+G		BIC
partition	012032		
-lnL	2133.4446		
K	128		
freqA	-	R(a)	0.4421
freqC	-	R(b)	1.2723
freqG	-	R(c)	1.0000
freqT	-	R(d)	0.4421
ti/tv	-	R(e)	3.0879
		R(f)	1.0000
p-inv	-	gamma	0.7800

```
BEGIN PAUP;
Lset base=equal nst=6  rmat=(0.4421 1.2723 1.0000 0.4421
3.0879 1.0000) rates=gamma shape=0.7800 ncat=4 pinvar=0;
END;
```

lset ... specification of the selected model

Base ... frequency of bases A, C, G (T)

Nst ... number of types of substitutions

Rmat ... rates of mutations a (AC), b (AG), c (AT), d (CG), e (CT), f (GT)

Rates ... gamma distribution function (rate of mutations at nucleotide positions)

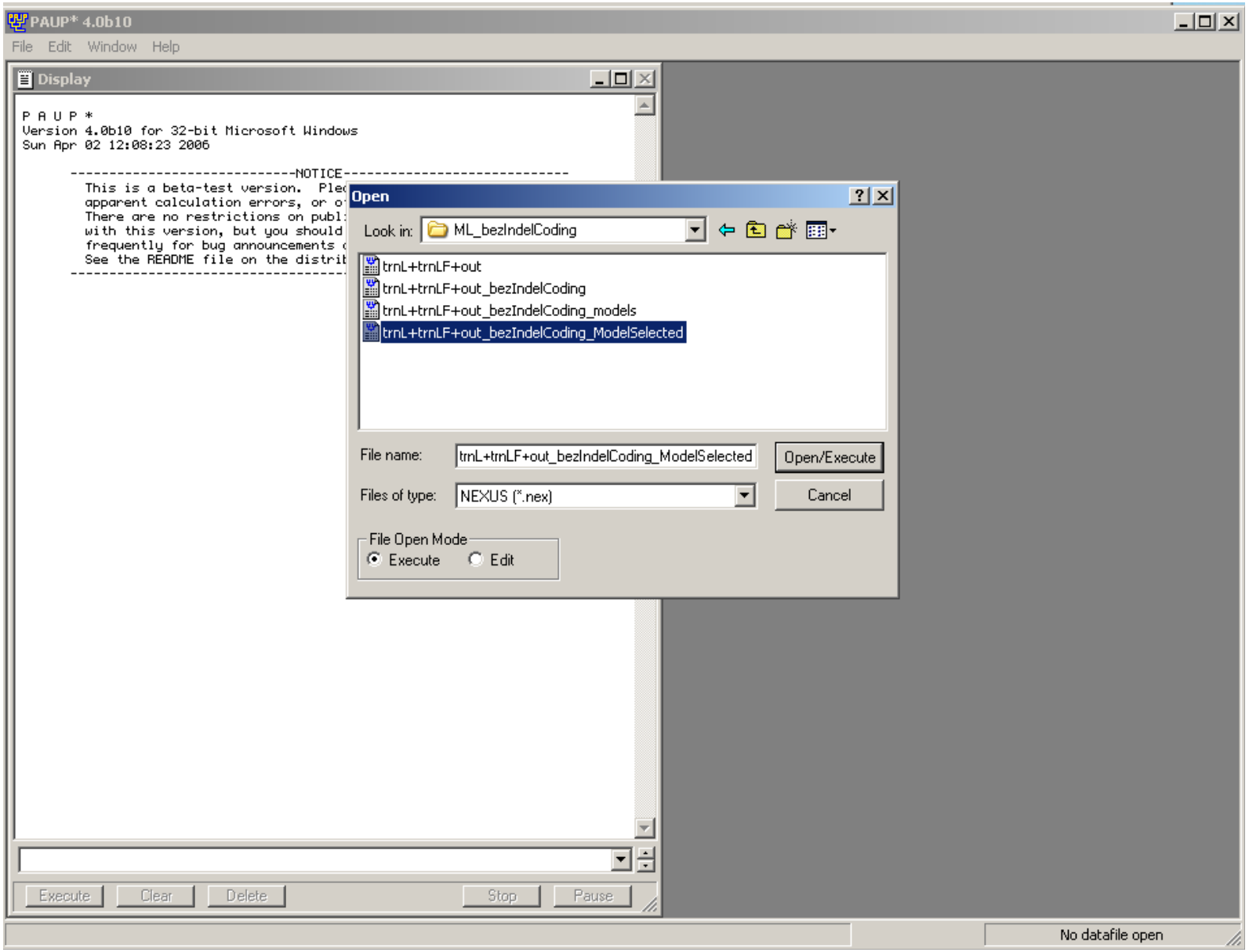
Shape ... function shape parameter

Pinvar ... proportion of invariable positions

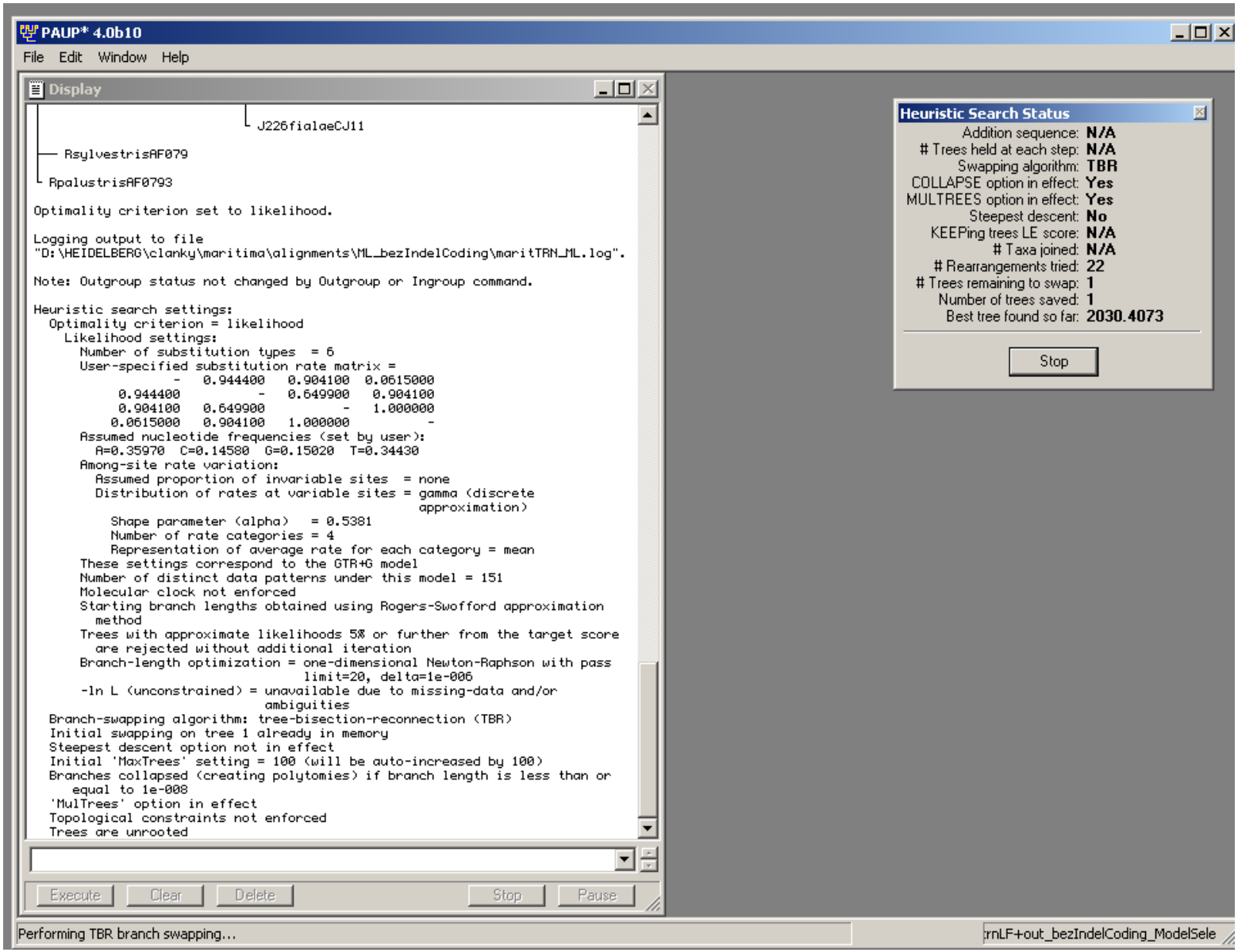
5. *nexus* file with added specification of selected model and commands for ML calculation

```
J215maritCC3·····ACCTACTAAGTGATAACTTTCAAATTCAGAGAAACCCTGGAATTAACAACGGGCAATCCTGAGCCAAATCCTTGTTTACGCGAACAAACC
RsylvestrisAF079352aAF362644·····NNNNNNNNNNNGATAACTTTCAAATTCAGAGAAACCCTGGAATTAACAATGGGCAATCCTGAGCCAAATCCTTGTTTACGCAAACAAACC
RpalustrisAF079351aAF362669·····NNNNNNNNNNNGATAACTTTCAAATTCAGAGAAACCCTGGAATTAACAATGGGCAATCCTGAGCCAAATCCTTGTTTACGCAAACAAACC
;¶
END;¶
¶
begin·PAUP;¶
set·autoclose=yes·increase=auto;¶
set·criterion=distance;¶
dset·distance=logdet;¶
outgroup·RsylvestrisAF079352aAF362644·RpalustrisAF079351aAF362669;¶
nj;¶
¶
set·autoclose=yes·increase=auto;¶
set·criterion=likelihood·notifybeep=yes;¶
log·file=maritTRN_ML.log;¶
Lset··Base=(0.3597·0.1458·0.1502)··Nst=6··Rmat=(0.9444·0.9041·0.0615·0.6499·0.9041)··Rates=gamma··Shape=0.5381··Pinvar=0;¶
outgroup·RsylvestrisAF079352aAF362644·RpalustrisAF079351aAF362669;¶
hsearch·start=1·nreps=10;¶
savetrees·brlens=yes·append=yes·file=maritTRN_ML.trees;¶
end;¶
¶
```

6. calculation of the ML analysis in the PAUP program

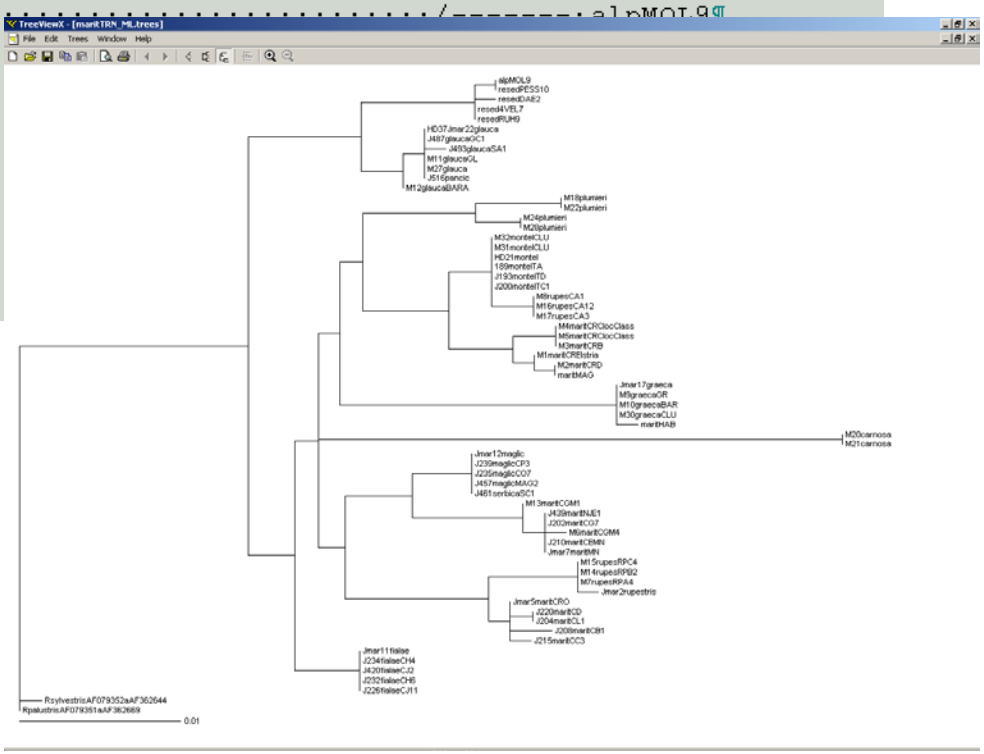


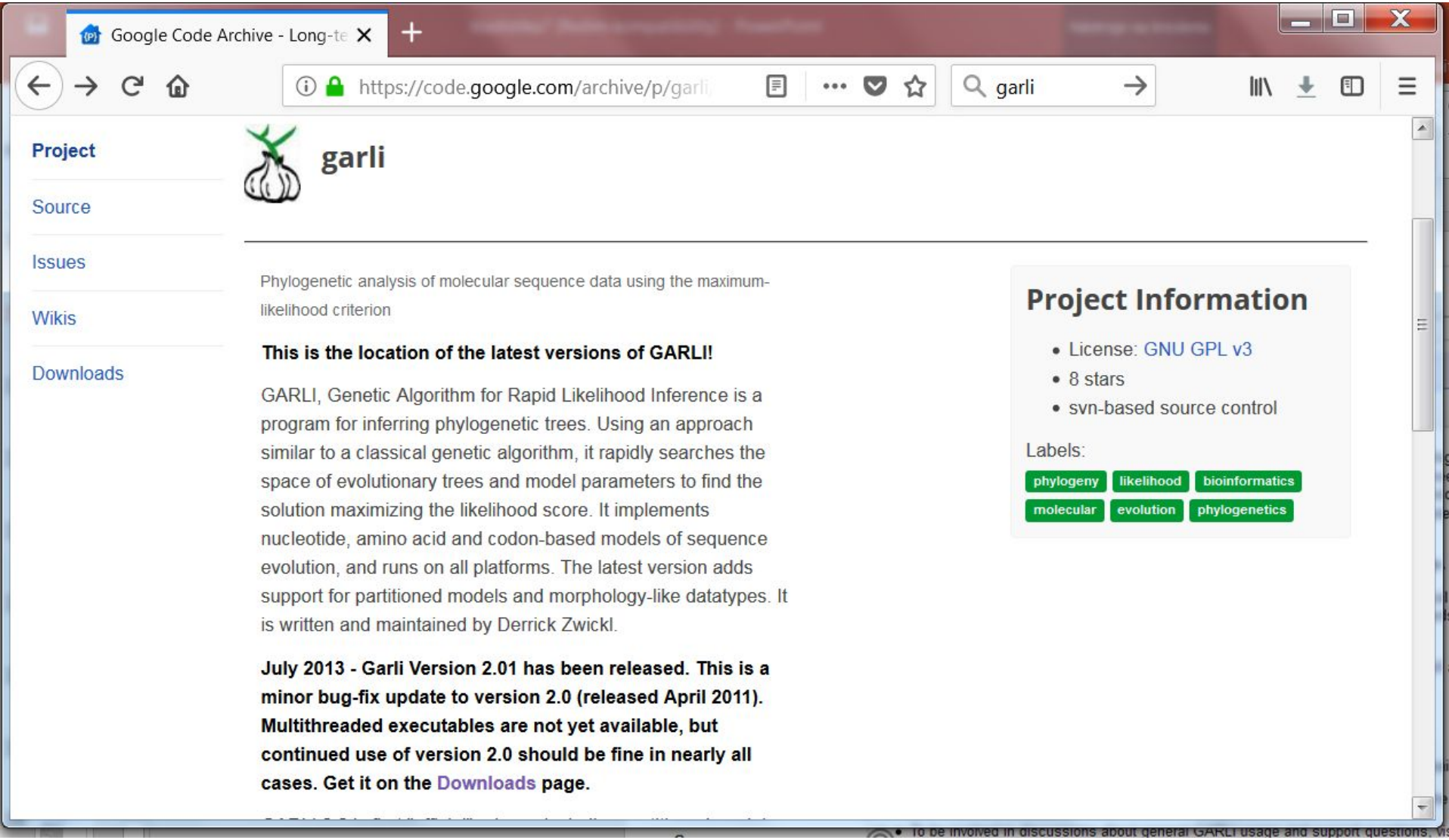
6. calculation of the ML analysis in the PAUP program



6. calculation of the ML analysis in the PAUP program

```
Heuristic search completed
   Total number of rearrangements tried = 57678
   Score of best tree(s) found = 2030.40733
   Number of trees retained = 1
   Time used = 29:48:50.0

1 tree appended to file "maritTRN_ML.trees"

Processing of file "trnL+trnLF+out_bezInd#3EC2F.txt" completed.

Tree number 1 (rooted using user-specified outgroup)

/-------------------------------------------------------------------------- RsylvestrisAF079
|
|.....................................................................
|.....................................................................
|.....................................................................
|.....................................................................
|.....................................................................
|.....................................................................
|.....................................................................
|.....................................................................
|.....................................................................
|.....................................................................
```





http://code.google.com/p/garli/

DOWNLOAD

Source code

LINKS

Publication

Documentation

Google group

GUI

Web services

Helper scripts

RAxML - Randomized Axelerated Maximum Likelihood

New RAxML citation

When using RAxML please cite the following paper: A. Stamatakis: "RAxML Version 8: A tool for Phylogenetic Analysis and Post-Analysis of Large Phylogenies". In *Bioinformatics*, 2014, open access.

Latest code download

Get the most up-to-date RAxML version from github.

Documentation

- **new RAxML version 8.0.0 manual** PDF
- copy of the old v704 manual: PDF
- For a basic step by step tutorial using some more recent features see RAxML step-by-step tutorial
- For a basic step by step tutorial by Pavlos Pavlidis on how to install and run RAxML on a Linux cluster see RAxML on cluster step-by-step tutorial
- For a video explaining the evolutionary placement algorithm for short reads see Alexis talking about evolutionary placement of short reads
- Video of Alexis talking about evolutionary placement of short reads at the Joint Genome Institute

http://sco.h-its.org/exelixis/web/software/raxml/

The RAxML v8.0.X Manual

by Alexandros Stamatakis
Heidelberg Institute for Theoretical Studies
May 23 2014

Structure of this manual

http://sco.h-its.org/exelixis/resource/download/NewManual.pdf

III. RAxML web-servers and Graphical User Interfaces

While there exist several web-servers that allow you to run RAxML, I am directly involved in running three of them.

1. The Cipres Portal web server: http://www.phylo.org/sub_sections/portal/
2. The web-server at vital IT in Switzerland: http://embnet.vital-it.ch/raxml-bb/
3. A dedicated server for the Evolutionary Placement Algorithm: http://epa.h-its.org/raxml

There is no official graphical user interface supported by me, but a GUI has been developed by researchers at the research museum in Frankfurt, which is available here: http://sourceforge.net/projects/raxmlgui/

Note that, I will not provide any sort of support for the GUI, you need to contact the original authors for this.

Home / Browse / Science & Engineering / Bio-Informatics / raxmlGUI

raxmlGUI

Brought to you by: imichalak, silvestro1

Summary | Files | Reviews | Support | Wiki | Mailing Lists | Hosted Apps | News | Discussion

★ 5.0 Stars (2)

147 Downloads (This Week)

Last Update: 2013-12-06

Tweet 5 | +1 2 | Like 26

Download raxmlGUI1.3.zip

Browse All Files

Description

A userfriendly graphical front-end for phylogenetic analyses using RAxML (Stamatakis, 2006). Please cite: Silvestro, Michalak (2012) - raxmlGUI: a graphical front-end for RAxML. Organisms Diversity and Evolution 12, 335-337. DOI: 10.1007/s13127-011-0056-0

http://sourceforge.net/projects/raxmlgui/

program TCS

http://bioresearch.byu.edu/tcs

Clement M, Posada D and Crandall K. 2000. TCS: a computer program to estimate gene genealogies. Molecular Ecology 9(10): 1657-1660

Templeton, A.R., Crandall, K.A., Sing, C.F., 1992. A cladistic analysis of the phenotypic associations with haplotypes inferred from restriction endonuclease mapping and DNA sequence data. III. Cladogram estimation. Genetics 132, 619–633

Templeton, A.R., 1998. Nested clade analyses of phylogeographic data: testing hypotheses about gene flow and population history. Mol. Ecol. 7, 381–397.

haplotype network formation (cpDNA data) - statistical parsimony

program TCS

http://bioresearch.byu.edu/tcs/

BYU | BRIGHAM YOUNG UNIVERSITY | COMPUTER SCIENCE DEPARTMENT

Computational Science Laboratory

HOME PROJECTS PEOPLE PAPERS CONTACT

Navigation

- Home
- Projects
  - TCS
  - MSA
  - AIRnet
  - GNUMAP
  - Pathgen
  - Promatch
  - Psoda
  - ScaffoldScaffolder
- People
- Papers
- Contact
- Lab Wiki
- Presentations

Computational Science Laboratory

The Computational Science Laboratory investigates new algorithms for solving problems in computational biology, computational chemistry and computational physics. Emphasis is placed on finding efficient algorithms that can be run on parallel computers.

Resources

The Laboratory currently uses computational resources from the Fulton Supercomputing Center as well as 600-800 idle desktop nodes across the BYU campus that are managed by the DOGMA system.

Software

Several software packages have been developed by members of the laboratory including:

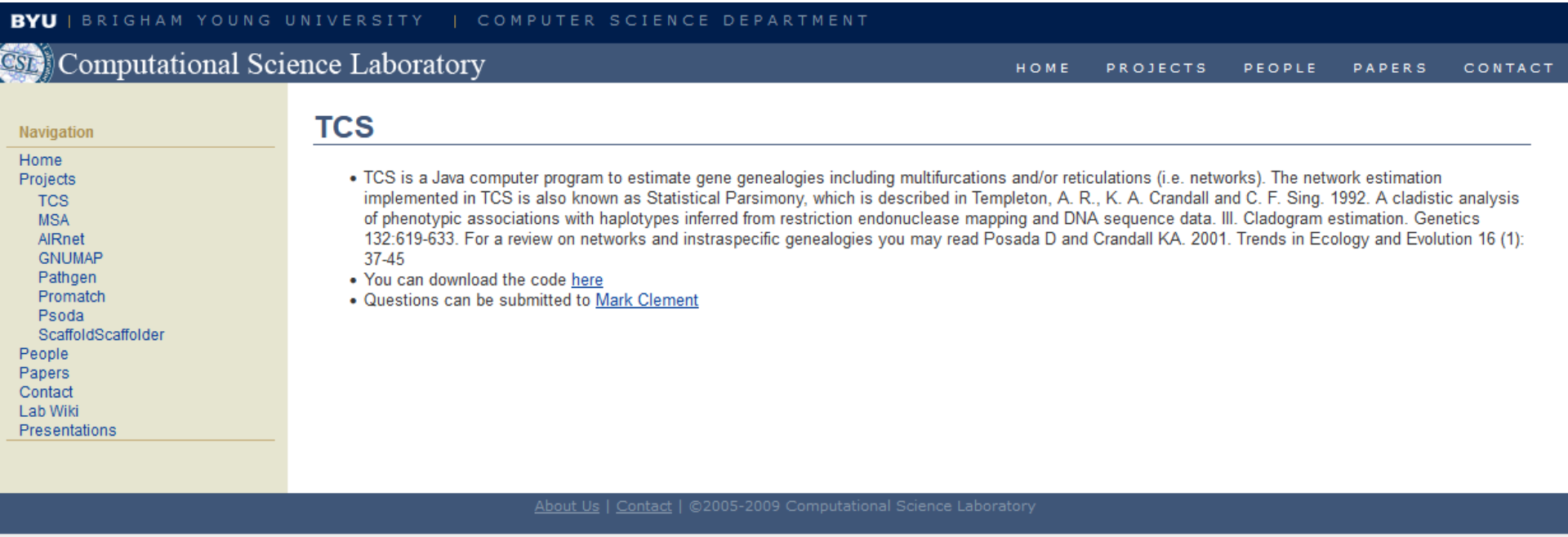
- GNUMAP maps reads from next-generation sequencers to genomic data.
- DOGMA uses idle computers as well as clusters and supercomputers for distributed scientific computing.
- The PSODA project provides a phylogenetic search tool that reads the same data file format as PAUP*.
- Members of the laboratory collaborate with the Department of Integrative Biology to develop software to analyze biochemical properties when performing alignment or phylogeny search. This software performs a detailed analysis of selection on amino acid properties using user-defined phylogenetic trees.
- The TCS project estimates gene genealogies including multifurcations and/or reticulations (i.e. networks). The network estimation implemented in TCS is also known as Statistical Parsimony, which is described in Templeton, A. R., K. A. Crandall and C. F. Sing. 1992.
- The ScaffoldScaffolder project is for haplotype-aware scaffolding of contigs. It also includes modules for solving the Contig Orientation Problem.

Funding

The laboratory is currently supported by an NSF grant to investigate hexapod phylogeny.

program TCS

http://bioresearch.byu.edu/tcs/



TCS 1.21 (30 June 2005)

2000-2005 © Mark Clement, Jacob Derington (Brigham Young University, USA)
Steve Woolley (Washington University, USA) and David Posada (University of Vigo, Spain).

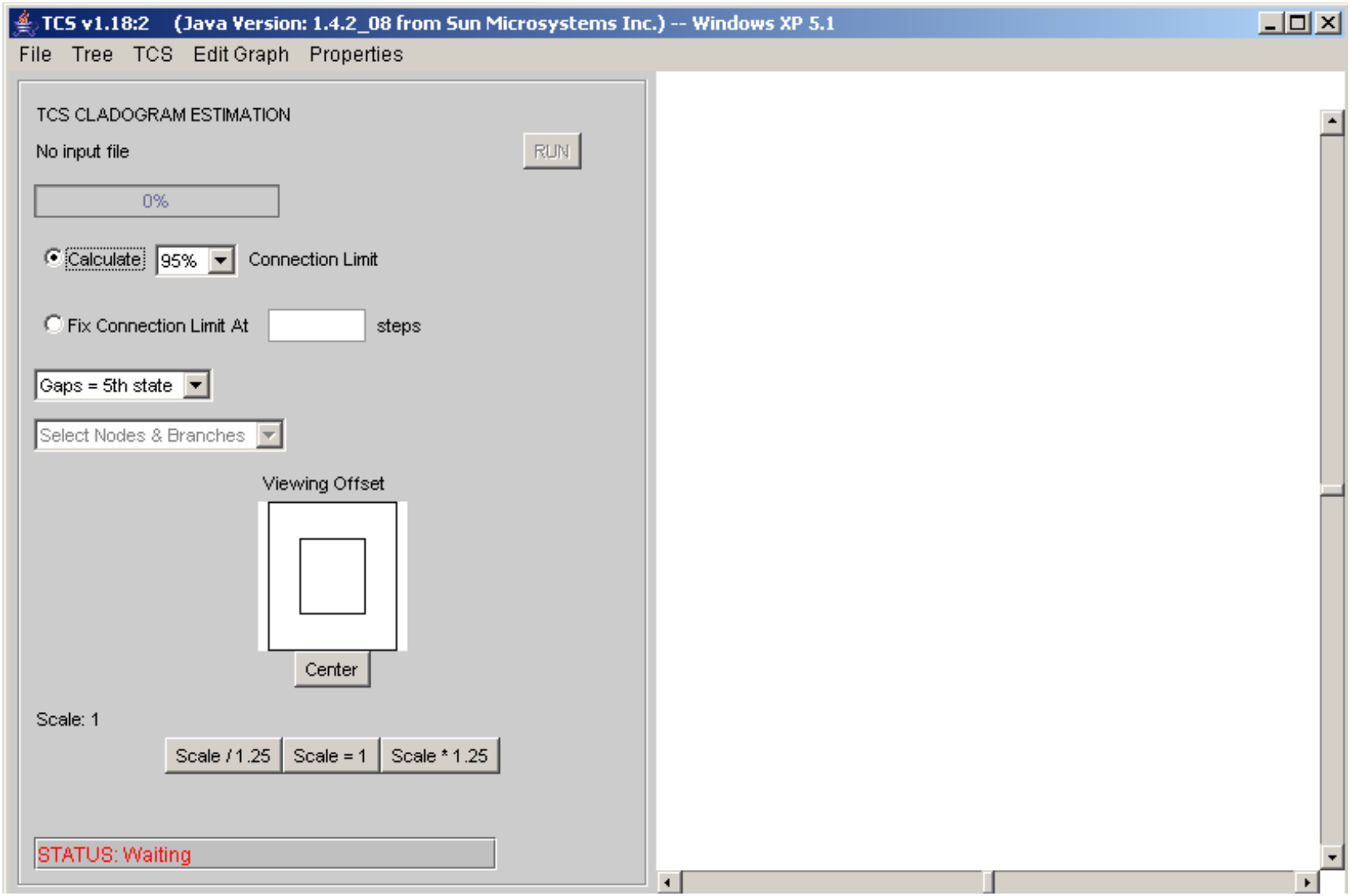
dposada@uvigo.es
http://darwin.uvigo.es/software/tcs.html

DISCLAIMER

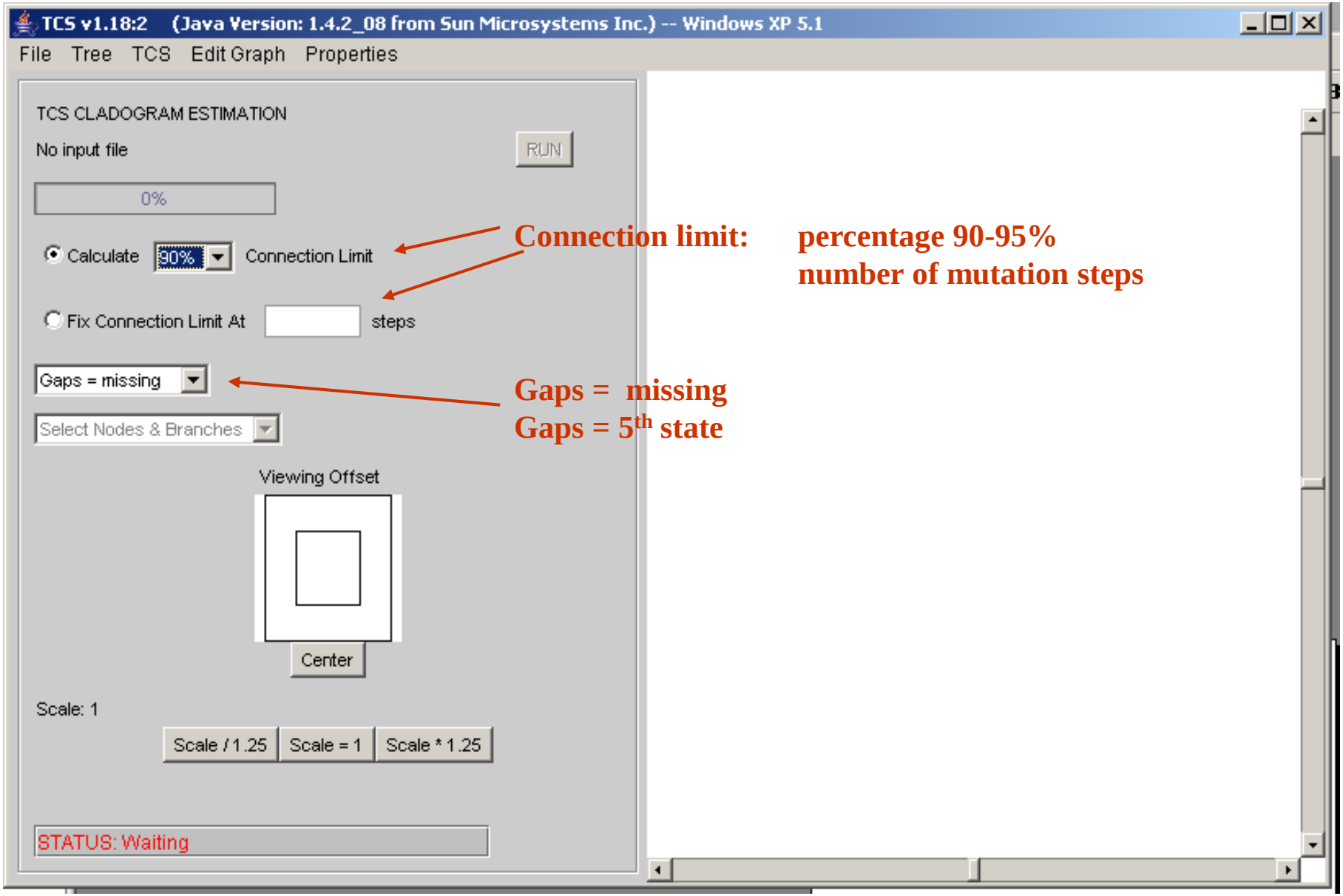
HISTORY

Version 1.21 (30 June 2005) Fixed the mapping code to correctly deal with gaps as defined in the GUI, either as 5th state or as missing (IUPAC ambiguity characters are treated as missing data)

program TCS


TCS v1.18:2 (Java Version: 1.4.2_08 from Sun Microsystems Inc.) -- Windows XP 5.1
File Tree TCS Edit Graph Properties
TCS CLADOGRAM ESTIMATION
No input file
RUN
0%
Calculate
95%
Connection Limit
Fix Connection Limit At
steps
Gaps = 5th state
Select Nodes & Branches
Viewing Offset
Center
Scale: 1
Scale / 1.25
Scale = 1
Scale * 1.25
STATUS: Waiting

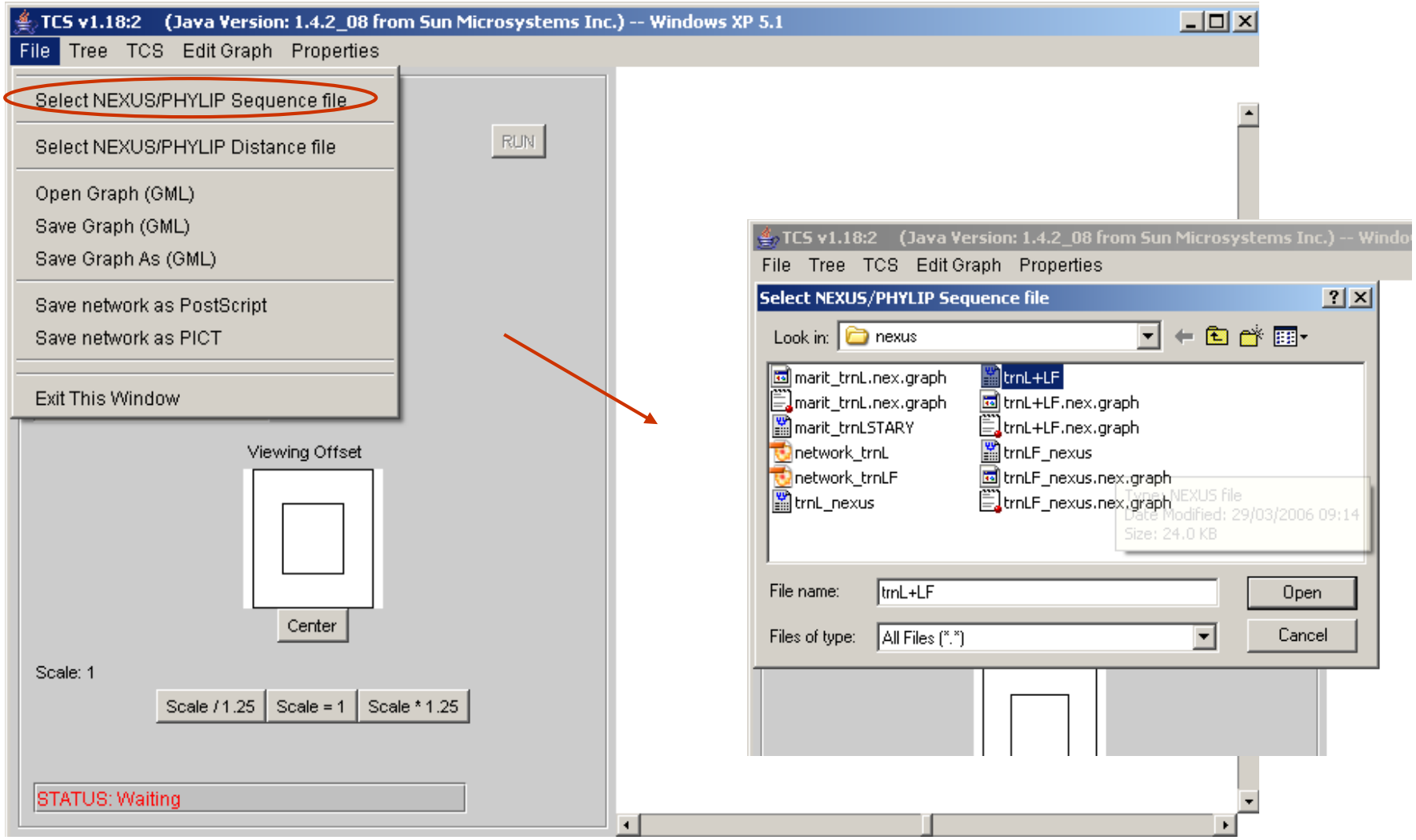
TCS



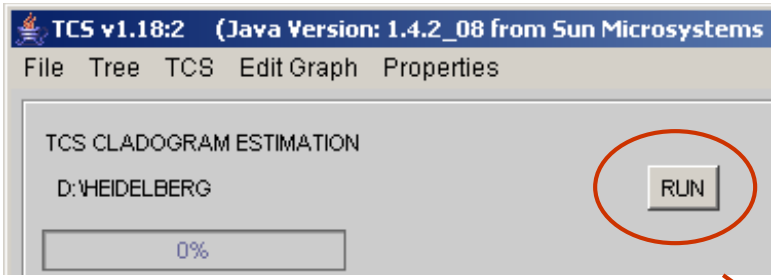
Connection limit: **percentage 90-95%**
number of mutation steps

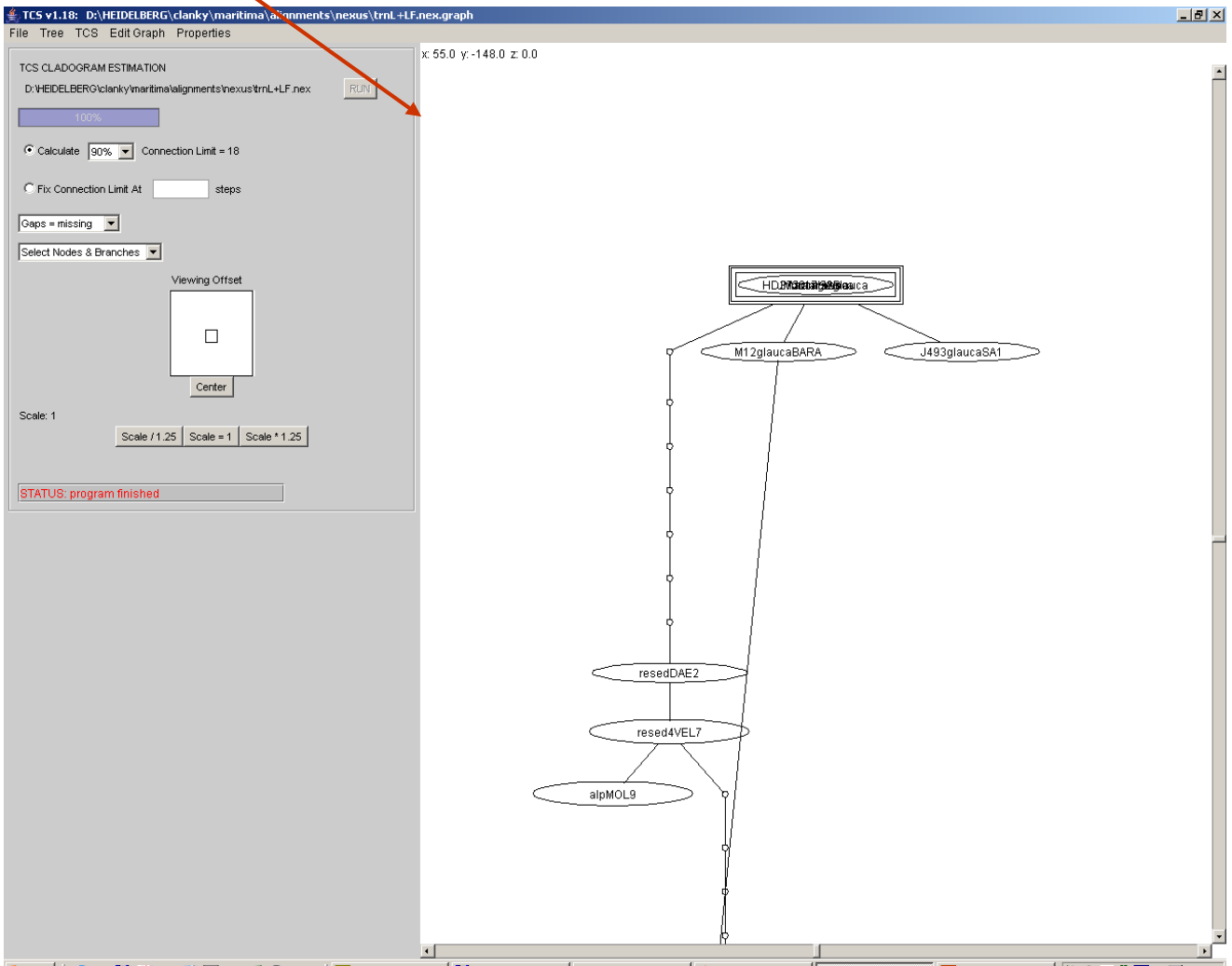
Gaps = missing
Gaps = 5th state

TCS

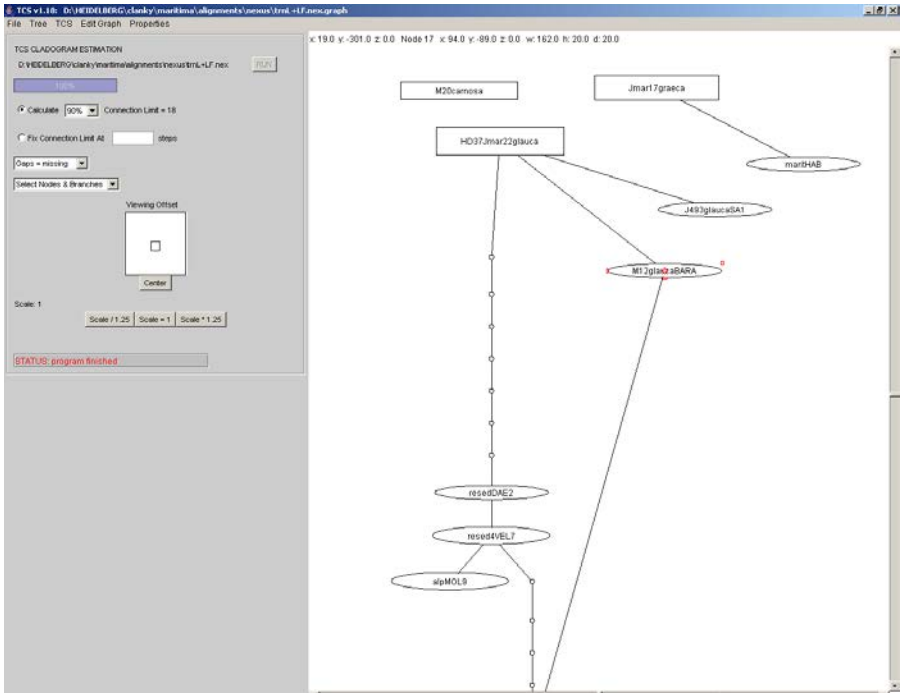

TCS v1.18:2 (Java Version: 1.4.2_08 from Sun Microsystems Inc.) -- Windows XP 5.1
File Tree TCS Edit Graph Properties
Select NEXUS/PHYLIP Sequence file
Select NEXUS/PHYLIP Distance file
Open Graph (GML)
Save Graph (GML)
Save Graph As (GML)
Save network as PostScript
Save network as PICT
Exit This Window
RUN
Viewing Offset
Center
Scale: 1
Scale / 1.25
Scale = 1
Scale * 1.25
STATUS: Waiting
Select NEXUS/PHYLIP Sequence file
Look in: nexus
marit_trnL.nex.graph
marit_trnL.nex.graph
marit_trnLSTARY
network_trnL
network_trnLF
trnL_nexus
trnL+LF
trnL+LF.nex.graph
trnL+LF.nex.graph
trnLF_nexus
trnLF_nexus.nex.graph
trnLF_nexus.nex.graph
Type: NEXUS file
Date Modified: 29/03/2006 09:14
Size: 24.0 KB
File name: trnL+LF
Open
Files of type: All Files (*.*)
Cancel

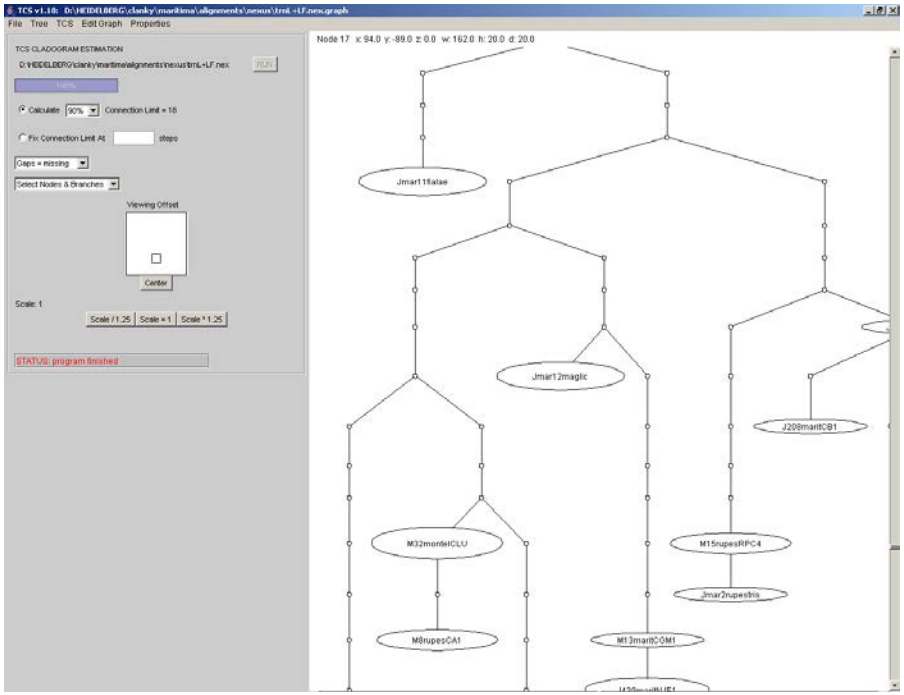
TCS

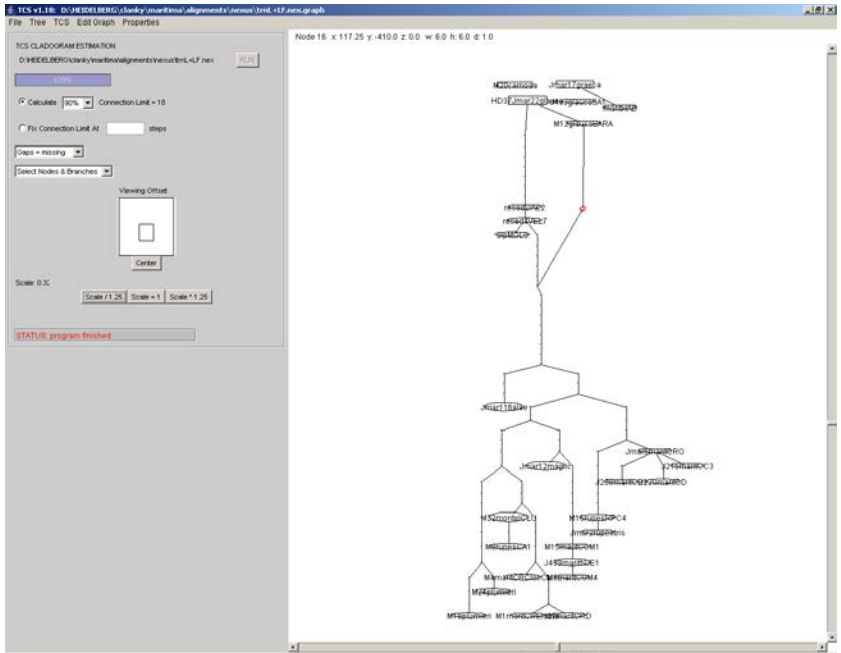

TCS v1.18:2 (Java Version: 1.4.2_08 from Sun Microsystems
File Tree TCS Edit Graph Properties
TCS CLADOGRAM ESTIMATION
D:\HEIDELBERG
RUN
0%


File Tree TCS Edit Graph Properties
TCS CLADOGRAM ESTIMATION
D:\HEIDELBERG\clanky\maritima\alignments\nexus\trnL+LF.nex
RUN
100%
Calculate 90% Connection Limit = 18
Fix Connection Limit At steps
Gaps = missing
Select Nodes & Branches
Viewing Offset
Center
Scale: 1
Scale / 1.25
Scale = 1
Scale * 1.25
STATUS: program finished
x: 55.0 y: -148.0 z: 0.0
M12glaucaBARA
J493glaucaSA1
resedDAE2
resed4VEL7
alpMOL9

TCS


TCS v1.18b: D:\HEIDELBERG\clanky\maritima\alignments\nexus\tmL+LF.nex.graph
File Tree TCS Edit Graph Properties
TCS CLADOGRAM ESTIMATION
Calculate 90% Connection Limit = 18
Fix Connection Limit At steps
Gaps = missing
Select Nodes & Branches
Viewing Offset
Center
Scale: 1
Scale / 1.25 Scale = 1 Scale * 1.25
STATUS: program finished
x: 19.0 y: -301.0 z: 0.0 Node 17 x: 94.0 y: -89.0 z: 0.0 w: 162.0 h: 20.0 d: 20.0
M20camosa
Jmar17graeca
HD37Jmar22glauca
marltHAB
J493glaucaBA1
M12glaucaBARA
resedDAE2
resed4VEL7
alpMOL9


TCS v1.18b: D:\HEIDELBERG\clanky\maritima\alignments\nexus\tmL+LF.nex.graph
File Tree TCS Edit Graph Properties
TCS CLADOGRAM ESTIMATION
Calculate 90% Connection Limit = 18
Fix Connection Limit At steps
Gaps = missing
Select Nodes & Branches
Viewing Offset
Center
Scale: 1
Scale / 1.25 Scale = 1 Scale * 1.25
STATUS: program finished
Node 17 x: 94.0 y: -89.0 z: 0.0 w: 162.0 h: 20.0 d: 20.0
Jmar11falae
Jmar12maglic
J208marltCB1
M32marltCLU
M15rupesRPC4
Jmar2rupesbis
M8rupesCA1
M13marltCGM1

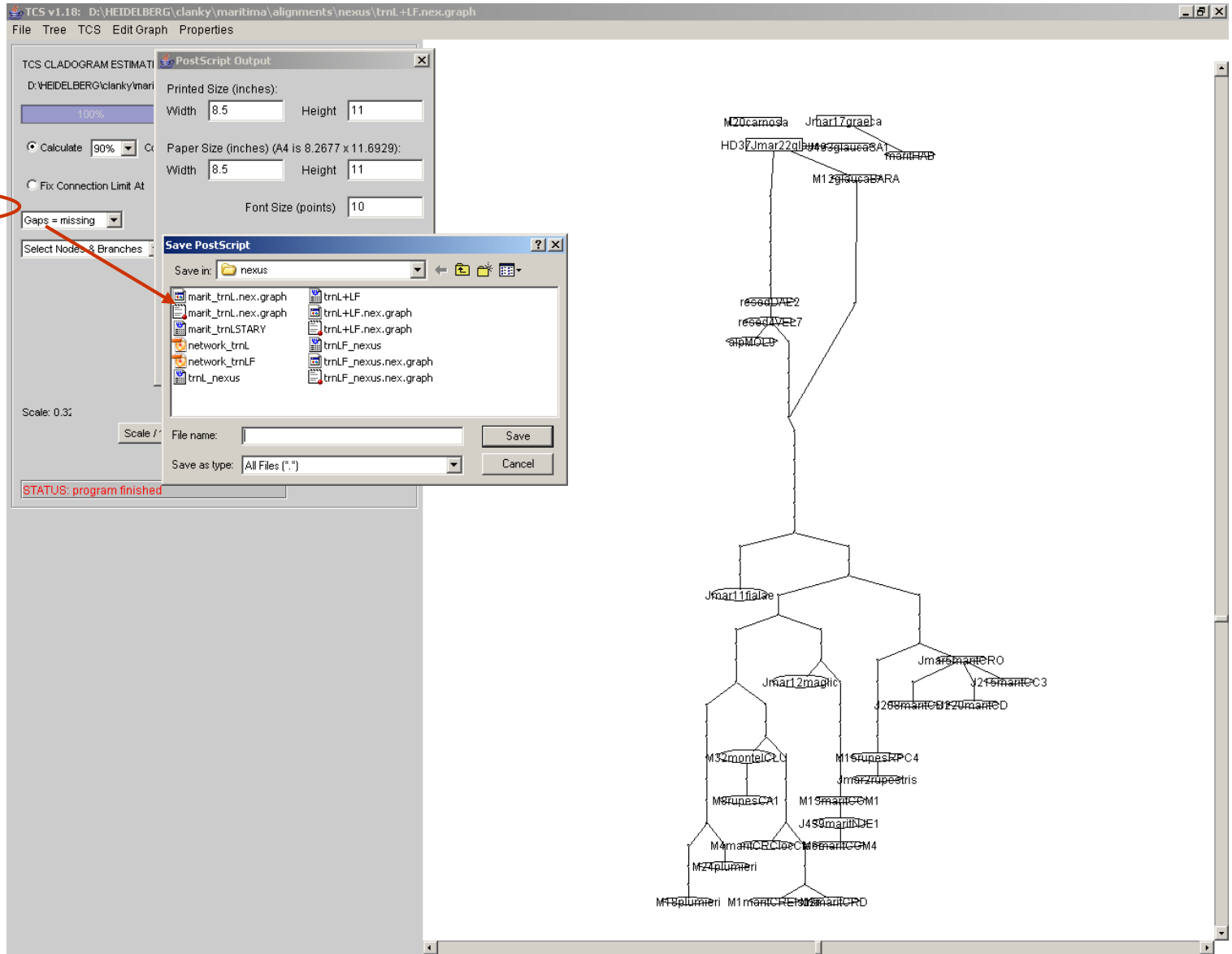

TCS v1.18b: D:\HEIDELBERG\clanky\maritima\alignments\nexus\tmL+LF.nex.graph
File Tree TCS Edit Graph Properties
TCS CLADOGRAM ESTIMATION
Calculate 90% Connection Limit = 18
Fix Connection Limit At steps
Gaps = missing
Select Nodes & Branches
Viewing Offset
Center
Scale: 0.3
Scale / 1.25 Scale = 1 Scale * 1.25
STATUS: program finished
Node 16 x: 117.25 y: -410.0 z: 0.0 w: 6.0 h: 6.0 d: 1.0

Saving haplotype network

as - *.gml file

as - *.ai file

TCS



log file:

TCS

PSPad - [D:\HEIDELBERG\clanky\maritima\alignments\nexus\trnL+LF.nex.graph.log]

File Projects Edit Search View Format Tools HTML Settings Window Help

trnL+LF.nex.graph.log

```
TCS v1.18
Wed Mar 29 11:54:05 CEST 2006
Datafile = D:\HEIDELBERG\clanky\maritima\alignments\nexus\trnL+LF.nex
Current OS = windows
Data in NEXUS format
Number of sequences: 63
Length of sequences: 898
_____

PARSIMONY PROBABILITY
For 1 step(s), » P(90%) = 0.9996570572825279
For 2 step(s), » P(90%) = 0.9984714516172535
For 3 step(s), » P(90%) = 0.9967988577313984
For 4 step(s), » P(90%) = 0.9945796040481115
For 5 step(s), » P(90%) = 0.9918080152866031
For 6 step(s), » P(90%) = 0.9884885688352988
For 7 step(s), » P(90%) = 0.9846259749890844
For 8 step(s), » P(90%) = 0.9802255475326238
For 9 step(s), » P(90%) = 0.9752935014113066
For 10 step(s), » P(90%) = 0.9698369419017302
For 11 step(s), » P(90%) = 0.9638638396427711
For 12 step(s), » P(90%) = 0.9573830110395306
For 13 step(s), » P(90%) = 0.9504040969377248
For 14 step(s), » P(90%) = 0.9429375388 42432
For 15 step(s), » P(90%) = 0.9349945529108609
For 16 step(s), » P(90%) = 0.9265871018348107
For 17 step(s), » P(90%) = 0.9177278647087628
For 18 step(s), » P(90%) = 0.9084302049891934
For 19 step(s), » P(90%) = 0.8987081366588069

RUN SETTINGS
Calculated maximum connection steps at 90% = 18
Gaps treated as missing data
```