

**COUNT: Evolutionary Analysis of
Phylogenetic Profiles and Other
Numerical Characters
User's Guide**

Miklós Csűrös

Department of Computer Science and Operations Research
Université de Montréal
Montréal, Québec, Canada

July 15, 2009

License

The COUNT software package is distributed under the terms of the BSD license, as shown below.

Copyright © 2009, Miklós Csűrös
All rights reserved. Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
3. Neither the name of the *Université de Montréal* nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Contents

1	Overview	4
1.1	Introduction	4
1.2	Availability	4
2	Mathematical background	6
2.1	Introduction	6
2.2	Parsimony	6
2.3	Phylogenetic birth-and-death model	7
2.3.1	Rates	7
2.3.2	Inparalogs and xenologs	8
2.3.3	Posteriors	8
2.3.4	Expected values	9
2.3.5	Absent families	10
3	Using COUNT	12
3.1	Basic design concepts	12
3.1.1	Errors	12
3.1.2	Sessions and the work area	13
3.1.3	Browsers, primary items and views	14
3.1.4	Tree displays	14
3.1.5	Table displays: column rearrangements and row sorting	15
3.1.6	Comments in input and output files	16
3.2	Sessions	16
3.2.1	Organismal phylogeny	16
3.2.2	Saving your work	17
3.3	Data	18
3.3.1	Family size table	18
3.3.2	Family annotations	19

3.3.3	Family selections, filtering, and absence/presence transformations	20
3.3.4	Saving tables	22
3.4	Rates	22
3.4.1	Rate model file	22
3.4.2	Rates panel	23
3.4.3	Rates panel: table	24
3.4.4	Rates panel: graphical display of rate variation	24
3.4.5	Rates panel: tree display	25
3.4.6	Rate model optimization	26
3.4.7	Rate optimization: model type	28
3.4.8	Rate optimization: model parameters	28
3.4.9	Rate optimization: computing	29
3.5	Analysis panels	30
3.5.1	Analysis panel: family table	31
3.5.2	Analysis panel: lineage table	32
3.5.3	Analysis panel: tree display	32
3.5.4	Analysis: Dollo parsimony	33
3.5.5	Analysis: Wagner parsimony	34
3.5.6	Analysis: Posteriors	35
3.5.7	Analysis: Propensity for gene loss	37
4	Test data	39
5	Command-line usage	41
5.1	Overview	41
5.2	Executables	42
5.2.1	Output	42
5.2.2	Comments	42
5.2.3	Data formats	42
5.3	Wagner parsimony	42
5.4	Model parameters	43
5.5	Inference of ancestral gene content	45

Chapter 1

Overview

1.1 Introduction

The COUNT is a software package for the evolutionary analysis of homolog family sizes, or other numerical census-type characters along a phylogeny.

1.2 Availability

COUNT is written entirely in Java, and, thus, can be used in different operating systems, including Mac OS X, Microsoft Windows, and various Unix/Linux versions. The software is packaged in a JAR file, and can be executed in Java versions 1.6 and above.

Mac OS X I have written the software using a Mac, and went to some extent to integrate the Java executable into a native-looking application. The JAR file is bundled as `Count.app`, which you can just run directly by double-clicking on it.

Microsoft Windows You need to have a Java Virtual Machine on your computer in order to run COUNT. You could download, for instance, Sun's Java Runtime Environment from <http://www.java.com/>, which is the JRE I used in the testing. You will probably need to enable larger memory usage for the JVM than the default setting, which you get by double-clicking on the JAR file. You can launch COUNT via the provided MS-DOS batch file that sets the heap space for the JVM to 1000 Megabytes. Edit the batch file manually, if necessary.

Unix/Linux You can run COUNT from the command line, launching `java -jar Count.jar`. You will probably need to enable larger memory usage for the JVM than the default setting, which you can do by launching COUNT as `java -Xmx1024M -jar Count.jar`. The `-Xmx` option here sets the Java heap space to 1 Gigabytes: you can experiment with other settings appropriate for your computer and data set.

Chapter 2

Mathematical background

2.1 Introduction

The methods implemented in COUNT aim to facilitate the evolutionary analysis of gene content evolution. The principal data consist of the distribution of homolog family sizes across multiple genomes. In particular, the data correspond to a table $[\Phi_{fj}: f = 1, \dots, n; j = 1, \dots, m]$, where Φ_{fj} is the number of homologous genes to family f that are found in genome j . The vector $\Phi_f = (\Phi_{fj}: j = 1, \dots, m)$ is the so-called *phylogenetic profile* of family f . *Ancestral reconstruction* is the problem of inferring family sizes at inner nodes of a given evolutionary tree over a subset of the genomes $j = 1, \dots, m$. In a *parsimony* approach, the phylogenetic profile is extended to inner nodes by minimizing a penalty function over the implied size changes on the tree edges. A *likelihood* approach assumes an explicit probabilistic model for phylogenetic profiles. COUNT implements so-called phylogenetic birth-and-death models, in which family size evolution on an edge is governed by a linear birth-and-death model traditionally employed in the contexts of queuing systems and population growth. After optimizing the parameters of such a model on a data set, ancestral reconstruction can be carried out by computing posterior probabilities for the family sizes at inner nodes.

2.2 Parsimony

Parsimony COUNT implements a parsimony method known as asymmetric Wagner parsimony. The method is described in details elsewhere [Csűrös, M. “Ancestral reconstruction by asymmetric Wagner parsimony over continuous charac-

ters and squared parsimony over distributions.” *Sixth Annual RECOMB Satellite Workshop on Comparative Genomics*, Springer LNCS **5267**:72–86, 2008. DOI 10.1007/978-3-540-87989-3_6]. The key idea is to penalize the changes of gene family size differently in cases of losses and gains. Specifically, a change from x to y on an edge is penalized either by $(x - y)$ when $x > y$, or by $g(y - x)$ when $y > x$. The g parameter sets the relative penalty of a gain vs. a loss. In classic Wagner parsimony [Farris, J. S. “Methods for computing Wagner trees.” *Systematic Zoology*, **19**(1):83–92, 1970] $g = 1$, but if gene losses happen more often than gains, then some $g > 1$ may be a more adequate choice.

2.3 Phylogenetic birth-and-death model

2.3.1 Rates

A *phylogenetic birth-and-death model* assumes that a stochastic process acts on each edge, determining the evolution of homolog family size. The process on an edge is characterized by three parameters, denoted by κ , μ , and λ . A family of size n decreases by a rate of $n\mu$ and increases by a rate of $(\kappa + n/\lambda)$. In the context of a homolog gene family, μ is the individual gene loss rate (uniform across members of the family), λ is the individual gene duplication rate (uniform across members of the family), and κ is the rate of gene gain by any mechanism, including innovation and lateral gene transfer. The model and the associated computational techniques are described in details elsewhere [Csűrös, M. and I. Miklós. “A probabilistic model for gene content evolution with duplication, loss, and horizontal transfer.” *Tenth Annual International Conference on Research in Computational Molecular Biology (RECOMB)*, Springer LNCS **3909**:206–220, 2006. DOI 10.1007/11732990_18; Csűrös, M. and I. Miklós. “Mathematical framework for phylogenetic birth-and-death models.” arXiv:0902.0970 [q-bio/PE], 2009]. In the most general model, the process parameters (κ, μ, λ) differ across edges, and depend on the gene family. Specifically, the linear birth-and-death process on edge e for family f has rate parameters

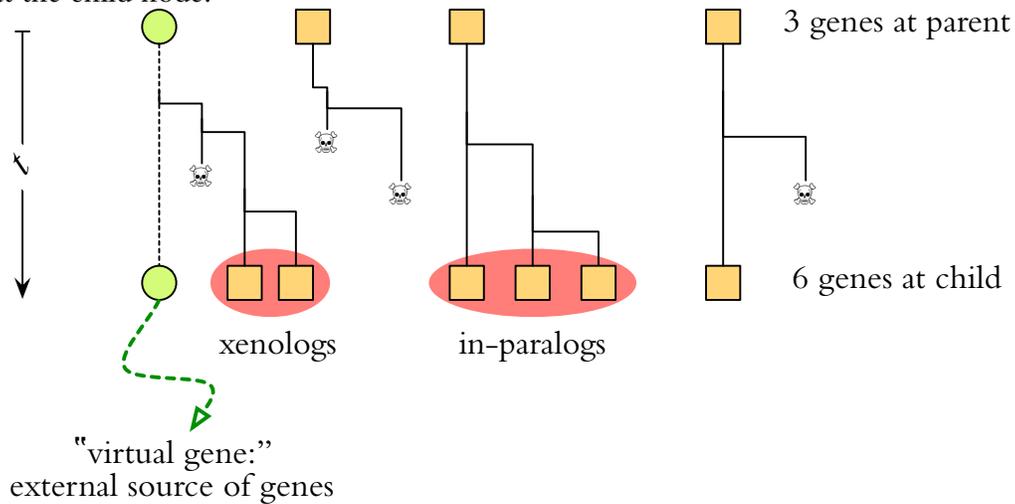
$$\kappa = \hat{\kappa}_e \kappa_f, \quad \mu = \hat{\mu}_e \mu_f, \quad \lambda = \hat{\lambda}_e \lambda_f,$$

and runs for a duration of $\hat{t}_e t_f$. Edge length is \hat{t}_e ; $\hat{\kappa}_e, \hat{\mu}_e, \hat{\lambda}_e$ are lineage-specific average rate parameters; $t_f, \kappa_f, \mu_f, \lambda_f$ are family-specific rate factors. These latter are assumed to be either constant, or have a discretized Gamma distribution [Yang, Z. “Maximum likelihood phylogenetic estimation from DNA sequences with variable rates over sites: approximate methods.” *Journal of Molecular Evolution*,

39:306–314,1994]. For gain and duplication rates, there is a possibility for mixing in 0-rate categories: $\lambda_f = 0$ or $\kappa_f = 0$ with prior probabilities $\pi_\lambda^{(0)}$ and $\pi_\kappa^{(0)}$ set during optimization. Then the family-specific duplication and gain rate factors have the discretized Gamma distribution in the non-zero-rate categories. Model parameters are set by likelihood optimization in COUNT. Given model parameters yield exactly computable likelihoods and posterior probabilities for ancestral gene content: see Csűrös and Miklós [arXiv:0902.0970 [q-bio/PE], 2009].

2.3.2 Inparalogs and xenologs

A key notion in the likelihood computation is that of partitioning the family members at a child node into *xenolog* and *inparalog groups*. The xenolog group consists of the members that have no ancestor at the parent node, i.e., their ancestor appeared in a gain event within the lineage leading to the child node. Each family member at the ancestor node has a corresponding inparalog group formed by its descendants at the child node.



2.3.3 Posteriors

Ancestral reconstruction can be carried out by computing posterior probabilities for the family sizes at inner nodes. Let $\xi[u]$ denote the family size at node u . The vector

$$\xi = (\xi[u] : \text{all nodes } u) \quad (2.1)$$

is a so-called phylogenetic character. The phylogenetic birth-and-death model defines the distribution of ξ . For a phylogenetic profile Φ , COUNT computes posterior probabilities for different characteristics conditioned on Φ , i.e., on the event

$$\xi \models \Phi = \left\{ \xi[j] = \Phi_j \text{ for every leaf } j \right\}.$$

In particular, the following statistics are computed.

Statistics	Applicability	Definition
presence	every node u	$\mathbb{P}\left\{ \xi[u] > 0 \mid \xi \models \Phi \right\}$
multiple members	every node u	$\mathbb{P}\left\{ \xi[u] > 1 \mid \xi \models \Phi \right\}$
gain	every edge uv	$\mathbb{P}\left\{ \xi[u] = 0; \xi[v] > 0 \mid \xi \models \Phi \right\}$
loss	every edge uv	$\mathbb{P}\left\{ \xi[u] > 0; \xi[v] = 0 \mid \xi \models \Phi \right\}$
expansion	every edge uv	$\mathbb{P}\left\{ \xi[u] = 1; \xi[v] > 1 \mid \xi \models \Phi \right\}$
contraction	every edge uv	$\mathbb{P}\left\{ \xi[u] > 1; \xi[v] = 1 \mid \xi \models \Phi \right\}$

(2.2)

2.3.4 Expected values

A great advantage of using posterior probabilities is that they can be summed together to obtain expectations, which are excellent aggregate characteristics for family dynamics and ancestral lineages. COUNT uses five characteristics for family dynamics, shown in the following table.

Statistics	Definition
gains	$\mathbb{E} \sum_{uv} \{ \xi[u] = 0; \xi[v] > 0 \} = \sum_{uv} \mathbb{P}\{ \xi[u] = 0; \xi[v] > 0 \}$
losses	$\mathbb{E} \sum_{uv} \{ \xi[u] > 0; \xi[v] = 0 \} = \sum_{uv} \mathbb{P}\{ \xi[u] > 0; \xi[v] = 0 \}$
expansions	$\mathbb{E} \sum_{uv} \{ \xi[u] = 1; \xi[v] > 1 \} = \sum_{uv} \mathbb{P}\{ \xi[u] = 1; \xi[v] > 1 \}$
contractions	$\mathbb{E} \sum_{uv} \{ \xi[u] > 1; \xi[v] = 1 \} = \sum_{uv} \mathbb{P}\{ \xi[u] > 1; \xi[v] = 1 \}$
arrivals	$\text{gains} + \mathbb{P}\{ \xi[\text{root}] > 0 \},$

(2.3)

where all probabilities are conditioned on the observation $\xi \models \Phi$.

If $(\Phi_{fj} : f = 1, \dots, n)$ is a set of phyletic profiles for n families, then ancestral lineages can be characterized by expectations. Let ξ_f denote phylogenetic character for family f . The number of families that were present at an ancestral node u is inferred as the conditional expected value

$$\mathbb{E} \left[\sum_{f=1}^n \{ \xi_f[u] > 0 \} \mid \forall f : \xi_f \models \Phi_f \right] = \sum_{f=1}^n \mathbb{P} \left\{ \xi_f[u] > 0 \mid \xi_f \models \Phi_f \right\}.$$

COUNT computes the following lineage-specific characteristics.

Statistics	Applies to	Definition
presence	nodes u	$\sum_{f=1}^n \mathbb{P} \left\{ \xi_f[u] > 0 \mid \xi_f \models \Phi_f \right\}$
multi-members	nodes u	$\sum_{f=1}^n \mathbb{P} \left\{ \xi_f[u] > 1 \mid \xi_f \models \Phi_f \right\}$
gains	edges uv	$\sum_{f=1}^n \mathbb{P} \left\{ \xi_f[u] = 0; \xi_f[v] > 0 \mid \xi_f \models \Phi_f \right\}$
losses	edges uv	$\sum_{f=1}^n \mathbb{P} \left\{ \xi_f[u] > 0; \xi_f[v] = 0 \mid \xi_f \models \Phi_f \right\}$
expansions	edges uv	$\sum_{f=1}^n \mathbb{P} \left\{ \xi_f[u] = 1; \xi_f[v] > 1 \mid \xi_f \models \Phi_f \right\}$
contractions	edges uv	$\sum_{f=1}^n \mathbb{P} \left\{ \xi_f[u] > 1; \xi_f[v] = 1 \mid \xi_f \models \Phi_f \right\}$

(2.4)

2.3.5 Absent families

An *absent family* is a family that has a phylogenetic profile $\Phi = \mathbf{0} = (0, 0, \dots, 0)$, i.e., it has no members at any terminal node. Absent families are immaterial in parsimony analyses, but the phylogenetic birth-and-death model assigns a well-defined probability p_0 to the all-0 profile. The likelihood optimization assumes that the data set does not include any families with an all-0 profile, and corrects the likelihood formula appropriately. Even for an absent family, there is a small probability that the family history includes at least one ancestral presence. If p_0 is the probability of an all-0 profile, then the number of absent families is estimated as

$$n_0 = n \frac{p_0}{1 - p_0}.$$

COUNT can take absent families into account when inferring lineage-specific statistics. If absent families are included in the reconstruction, then the statistics are modified in the following way.

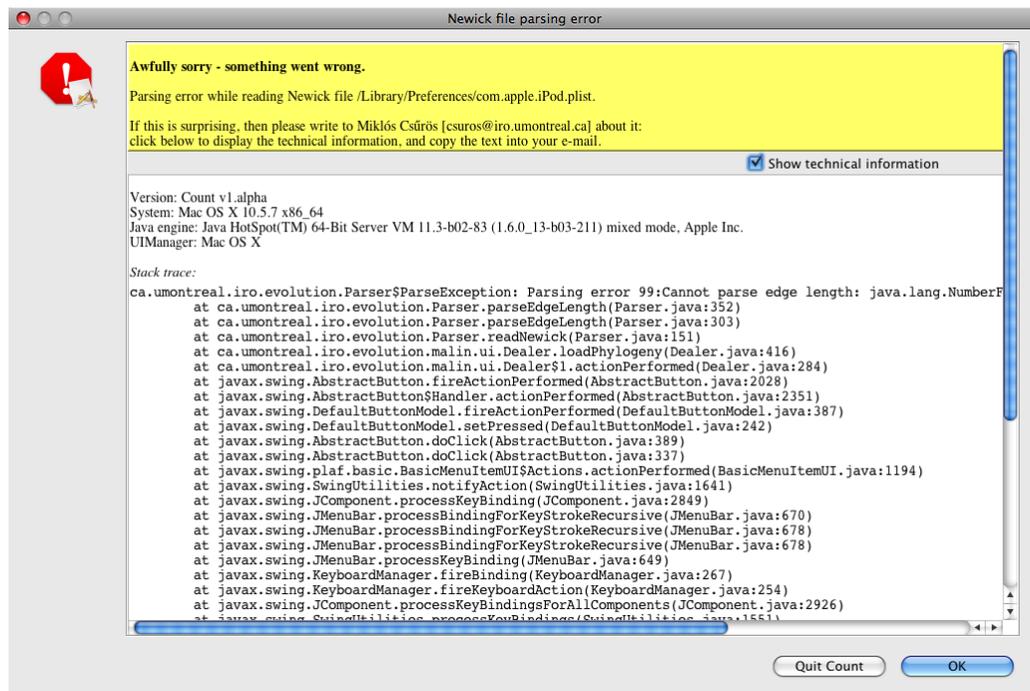
Statistics	Definition	
presence'	presence + $n_0 \cdot \mathbb{P}\{\xi[u] > 0 \mid \xi \models \mathbf{0}\}$	
multi-members'	multi-members + $n_0 \cdot \mathbb{P}\{\xi[u] > 1 \mid \xi \models \mathbf{0}\}$	
gains'	gains + $n_0 \cdot \mathbb{P}\{\xi[u] = 0; \xi[v] > 0 \mid \xi \models \mathbf{0}\}$	(2.5)
losses'	losses + $n_0 \cdot \mathbb{P}\{\xi[u] > 0; \xi[v] = 0 \mid \xi \models \mathbf{0}\}$	
expansions'	expansions + $n_0 \cdot \mathbb{P}\{\xi[u] = 1; \xi[v] > 1 \mid \xi \models \mathbf{0}\}$	
contractions'	contractions + $n_0 \cdot \mathbb{P}\{\xi[u] > 1; \xi[v] = 1 \mid \xi \models \mathbf{0}\}$	

Chapter 3

Using COUNT

3.1 Basic design concepts

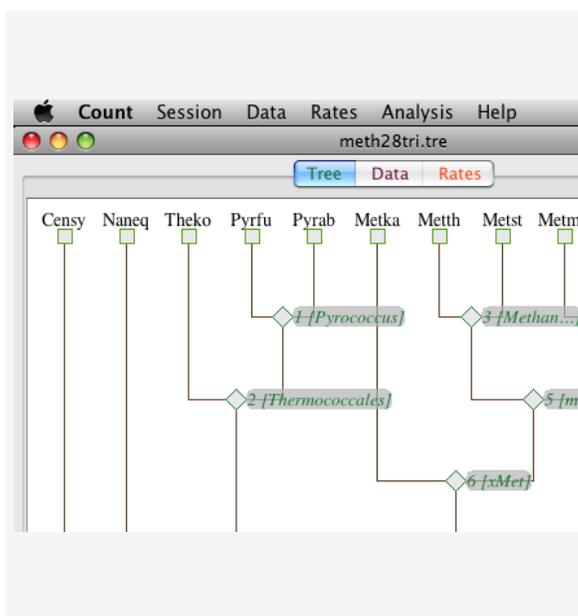
3.1.1 Errors



It may happen that something goes wrong. COUNT displays a window with the error message in such cases. If you think that the error was caused by a programming bug, or you would like to ask me for help with it, then send me an email,

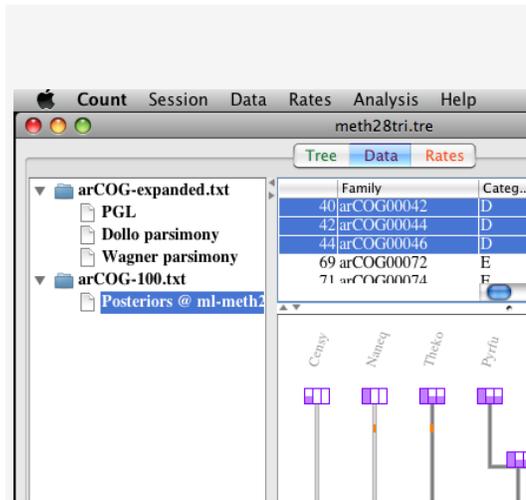
with the detailed technical error message included in the message body. The technical details are shown only when you select the corresponding checkbox.

3.1.2 Sessions and the work area



COUNT operates with *sessions*: each session is associated with a fixed species phylogeny. More than one session may be open at one time: e.g., the same data set may be analyzed with different phylogenies simultaneously. A session has three main components, represented by the tabs of the displayed workspace: a species phylogeny (Tree), a *browser* for data sets and analysis results (Data) and a browser for probabilistic models (Rates). The current session's name is displayed as the window title.

3.1.3 Browsers, primary items and views



The Data and Rates tabs are attached to browser displays. A browser consists of a hierarchy on the left, and an information panel on the right, corresponding to the item selected in the hierarchy. Primary items in the hierarchy (depicted as folders of a file system) are data tables (under the Data tab), or rate models (under the Rates tab). Primary items may have *views*, which correspond to various analysis tasks. Views are descendant nodes in the hierarchy (depicted as documents or bullets, depending on the operating system).

Nodes of the hierarchy have small associated popup menus which you can bring up by right-clicking on them (or by Ctrl-click on a Mac). The popup menu items include the removal of the node from the browser, and possibly saving options. Intron tables and rate models can be saved, and views are typically *exported* into text files. (The difference is that exported views cannot be loaded later, but saved tables and rate models can.)

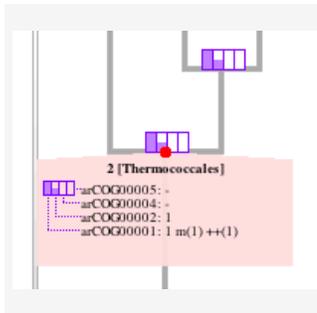
The browsers operate with split panes. You can resize the panes by dragging the dividers with the mouse, or even expand a pane completely by clicking on the little triangles on the bottom or the far right of the dividers.

3.1.4 Tree displays

There are several graphical displays that show analysis results on the session's phylogeny.



Tree displays have some associated control elements in the bottom tool bar. Most importantly, there is a zooming spinner on the bottom right, where you can set a relative magnification factor (displayed as a percentage).



You can select a tree node by clicking on it. In that way, you can get some more specific information about the selected node: it depends on the context what that information exactly is. (Here, ancestral inference results are shown in Wagner parsimony analysis.) You can select at most one node at a time. In order to deselect all nodes, click somewhere away from the tree nodes within the tree display.

3.1.5 Table displays: column rearrangements and row sorting

Most results are shown in tree displays together with table displays. Row selection in the tables affects the tree display and node selections in the tree display may affect the row selection in a table for lineages. Columns can be rearranged at will by dragging the column headers.

Family	Categ...	Description	#lin	#mem	d
arCOG02271	K	Predicted transcriptional...	10	14	
arCOG03296	K	Predicted transcriptional...	10	23	
arCOG04362	K	Predicted transcriptional...	10	22	
arCOG00732	K	Transcriptional regulator...	11	14	
arCOG00921	K	Predicted transcriptional...	11	12	
arCOG01128	K	Transcriptional activator...	11	17	
arCOG01679	K	Transcriptional regulator...	11	14	
arCOG02272	K	Predicted transcriptional...	11	46	

Table displays can be sorted row-wise by clicking on a column header. The column by which the table is sorted has a mark next to it, also indicating the direction (increasing or decreasing order). Here, families are sorted by the number of lineages (#lin column) they are represented in.

1.0	0.008	0.08	0.01	0.01
1.1	0.007	0.08	0.001	0.01
1.1	0.42	0.16	0.01	0.01
1.1	.	0.02	0.03	0.01
1.1	0.	0.41673003373875717	.	00
1.1	0.001	.	0.007	0.00
1.0	0.001	0.39	0.03	0.02

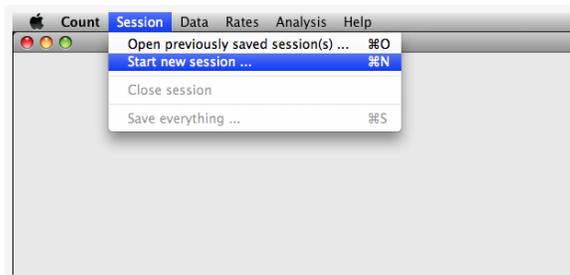
COUNT works with high-precision numerical values internally (Java's double), but the tables use rounding. Zero is denoted by a dot. The cell's tool tip gives the exact value used internally.

3.1.6 Comments in input and output files

Output and input files may contain comments in lines starting with #. Such lines are ignored on input, and do not contain essential information. If necessary (e.g., in order to prepare for import into Excel), they are easily filtered out in Unix on the command-line.

```
grep -v '#' file > stripped
```

3.2 Sessions



Data analysis in COUNT starts with opening a new session (Menu: Session → Start new session...).

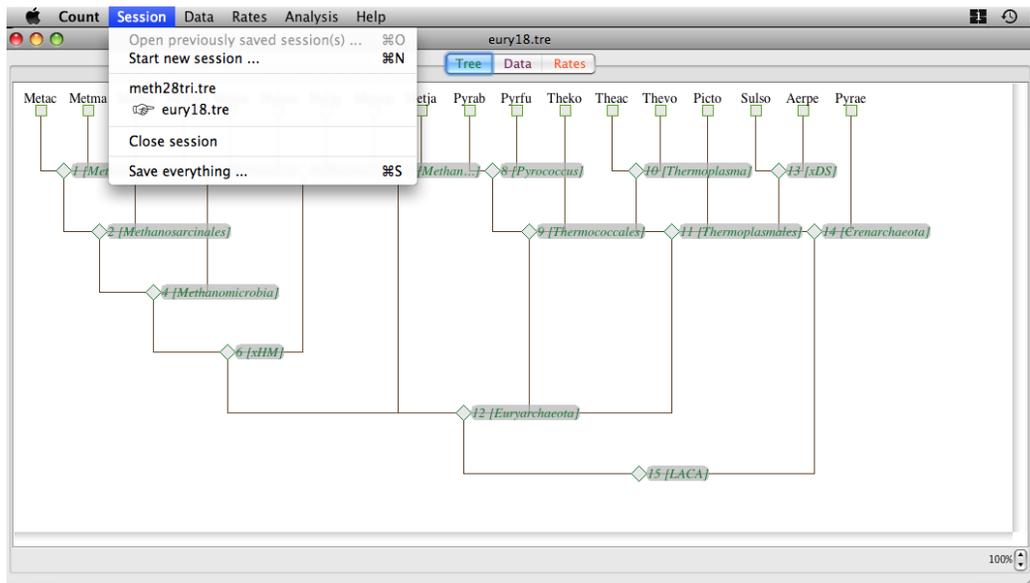
3.2.1 Organismal phylogeny

A session is opened by loading an organismal phylogeny. The phylogeny is expected to be in Newick format (<http://evolution.genetics.washington.edu/phylip/newicktree.html>) used by Phylip and other fine software packages for molecular evolution. The branch lengths of this phylogeny are ignored in most cases, except when computing Propensity for Gene Loss (PGL). The inner nodes of the tree may have more than two children: COUNT can deal with arbitrary multifurcations.

The phylogeny is displayed under the Tree tab. For convenience in the graphical user interface, it is recommended that you use short names (3-4 letters) for the terminal taxa. The inner nodes of the tree are numbered as 1,2,... (in a postorder

traversal). It may be useful to name the inner nodes of the phylogeny, which is possible in Newick format:

```
((Natph, Halsp, Halwa) Halobacteriales,
  ((Metcu, Methu, Metla) Methanomicrobiales,
    (Metbu, Metsa, (Metac, Metma, Metba) Methanosarcina) Methanosarcinales
  ) Methanomicrobia
) root;
```



The open sessions are listed under the Session menu, where they can be selected to switch back and forth between them. There is a pointing hand icon next to the current active session (shown in the window title). The active session can be disposed of by closing it (Menu: Session → Close session).

3.2.2 Saving your work

It is possible to save all your sessions, together with all the rate models, data tables, and analysis views at once: Session → Save everything The saved file is in a machine-readable format (XML), and can be loaded later to restore your analysis pipeline: Session → Open previously saved session(s) Note that the menu point is only available when there are no sessions open yet. The saved XML file can be compressed with Gzip (`gzip saved-sessions.xml` in the command line), and loaded later directly: COUNT checks for a `.gz` extension, and uncompresses the file on the fly.

3.3 Data

3.3.1 Family size table

The input data, on which various analyses can be performed, is a family size **table**. The table of phylogenetic profiles is a TAB-delimited text file. Every row corresponds to a homolog gene family, with the exception of the first row that gives the column headers. The first column is the family name. The second, third, etc. columns correspond to terminal taxa of the phylogenetic tree. The column headers must specify the terminal taxon names, in an arbitrary order. Columns with taxons missing from the phylogeny are ignored.

```
family Aerpe Arcfu Calma Censy Halma Halsp
arCOG00001 1 2 1 0 0 0
arCOG00002 1 0 1 1 1 1
arCOG00004 0 0 0 0 1 1
```

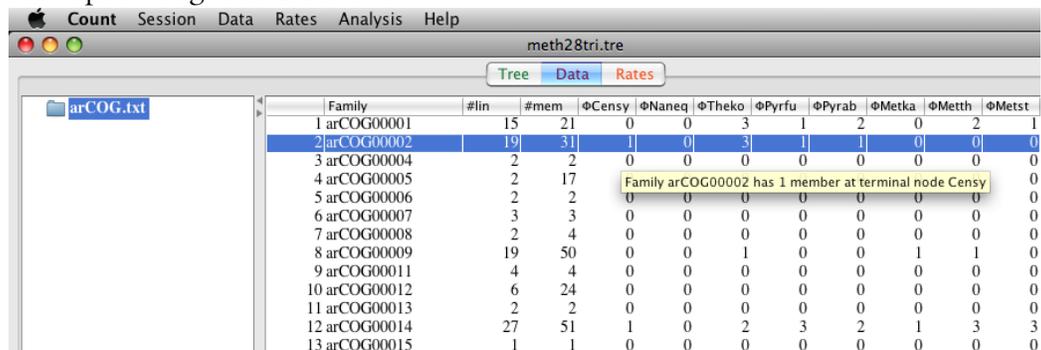
Note that the relevant files of the COG, KOG and arCOG databases can be used immediately, without any input format conversion.

The parsimony analyses can handle missing entries in the table, denoted by ?, but other programs treat missing data as a family size of 0.

A family size table can be opened from the menu:

Data → Open table

The opened table is displayed in the Data tab's browser as a primary item. The displayed columns include #lin and #mem, which are the total number of terminal taxa with at least one member, and total number of family members, respectively. Other columns are the family indices (original order in the file), family names, and numbr of homologs per terminal taxon (column $\Phi\sigma$ for taxon σ). Notice that more detailed information about elements of the graphical display is available in tool tips throughout COUNT.



Family	#lin	#mem	Φ Censy	Φ Naneq	Φ Theko	Φ Pyrpu	Φ Pyrab	Φ Metka	Φ Metth	Φ Metst
1 arCOG00001	15	21	0	0	3	1	2	0	2	1
2 arCOG00002	19	31	1	0	3	1	1	0	0	0
3 arCOG00004	2	2	0	0	0	0	0	0	0	0
4 arCOG00005	2	17	0	0	0	0	0	0	0	0
5 arCOG00006	2	2	0	0	0	0	0	0	0	0
6 arCOG00007	3	3	0	0	0	0	0	0	0	0
7 arCOG00008	2	4	0	0	0	0	0	0	0	0
8 arCOG00009	19	50	0	0	1	0	0	1	1	0
9 arCOG00011	4	4	0	0	0	0	0	0	0	0
10 arCOG00012	6	24	0	0	0	0	0	0	0	0
11 arCOG00013	2	2	0	0	0	0	0	0	0	0
12 arCOG00014	27	51	1	0	2	3	2	1	3	3
13 arCOG00015	1	1	0	0	0	0	0	0	0	0

3.3.2 Family annotations

Annotations define family properties: the default annotation is simply the family name. Further annotations (e.g., COG functional category) can also be included in COUNT. Note that annotations are text fields: numerical annotations are not supported. There are two ways to annotate families: an annotated family size table can be opened (Data → Open annotated table...), or additional family annotations can be loaded from a separate file (Data → Load family annotations...).

If you load an annotated table

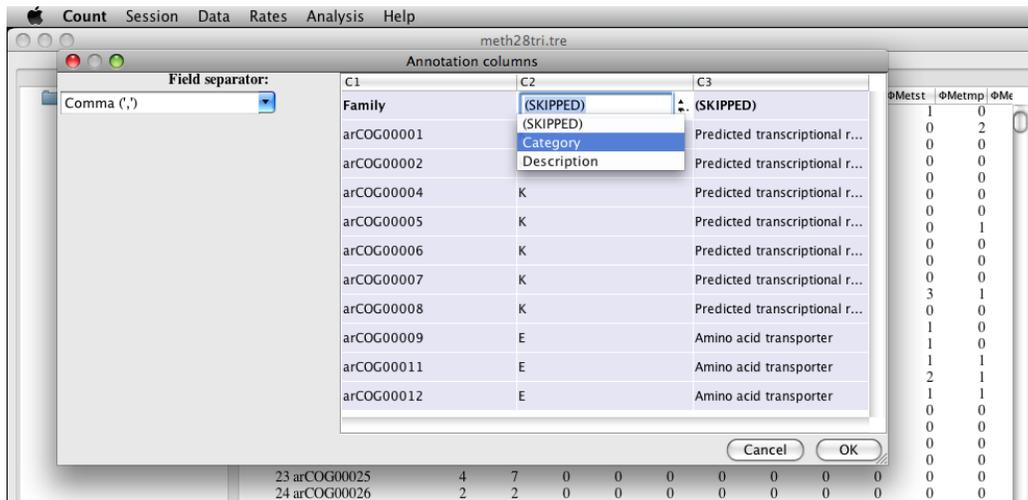
Data → Open annotated table... ,

then every column with a header not corresponding to a terminal taxon name is considered as an annotation column. You can have, for instance, “Category” and “Description” columns. Column headers must be unique (you cannot have two “Category” columns). You can also load a simple family size table. In that particular case, there will be an annotation column for every organism that is not present at the terminal taxa of the session’s phylogeny, although family sizes there will be treated as text values. Family name is always taken from the first column of the input file.

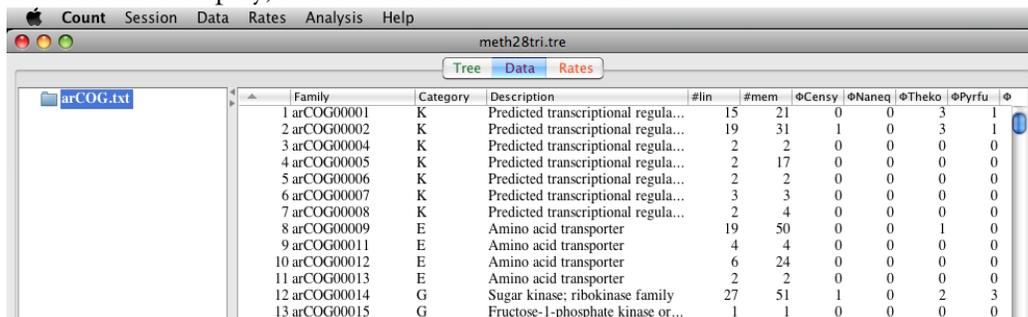
You can also add annotations from a separate file at any time

Data → Load family annotations... .

The annotations file is a text table, where the fields can be separated by comma (.csv extension), TAB (.txt files usually), or any other character. After the file is selected, the text format, and the imported columns are selected through a popup dialog. Note that COG family definition files (such as arCOGdef.csv) can be used immediately.

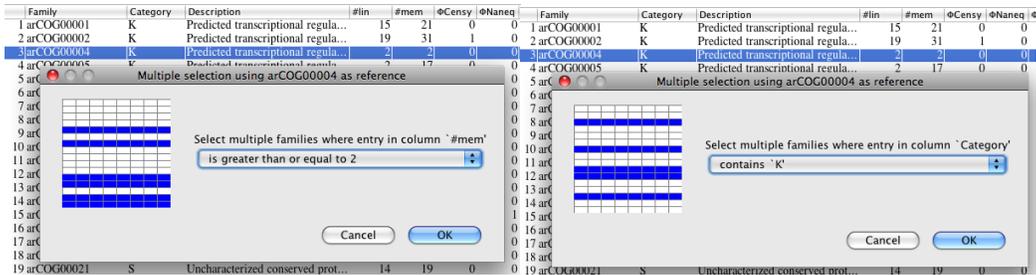


The first column of the annotation file must give the family name: families can be listed in arbitrary order. You can select the annotation columns that you would like to import through the dialog. The displayed column headers can be edited. They are either SKIPPED, meaning that they will not be imported, or have a different title from “SKIPPED.” There are two predefined options: “Category” and “Description,” but you can select any other text, as long as the columns have different names. The imported annotation columns are added to the currently selected table display, and all its descendant views.



3.3.3 Family selections, filtering, and absence/presence transformations

In display tables for family size and analysis results, you can select multiple families using the mouse directly, or by using logical selection criteria. Selection criteria are displayed by double-clicking on a table cell, in a popup menu.



If you double-click on a numerical column, then the selection options are “equal,” “less than or equal to,” and “greater than or equal to,” with the reference value taken from the cell you clicked on. If you double-click on a text column (annotations and family name), then the selection options are “equals” and “contains.” In this way, you can select families with a particular functional category, or size, or taxon representation. Or, using the table displaying some analysis results, you can define selection criteria based on presence at ancestral nodes, or other inferred characteristics.



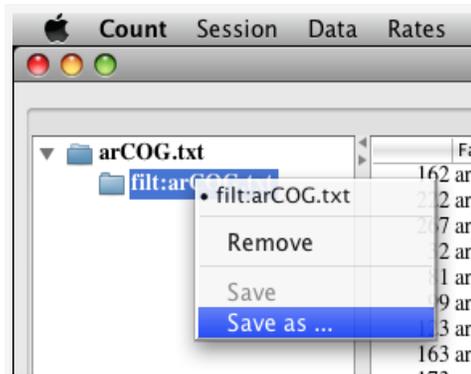
Selected families can be extracted into a new table (Data → Extract selected families into a new table). The filtered table appears as a descendant view in the Data browser.

Finally, family sizes can be transformed into binary profiles

Data → Transform numerical profiles into binary (presence/absence) profiles

Every positive value is replaced by '1' in the result. The binary table is displayed as a descendant view in the Data browser.

3.3.4 Saving tables



You may want to save the tables you created through filtering and binary transformations. Family size tables can be saved through the popup menu for the corresponding node in the data browser.

3.4 Rates

COUNT uses phylogenetic birth-and-death models (§2.3) in probabilistic inference. Rates can be set by optimization on the currently selected family size table, or previously computed rate models can be loaded from a *rate model file*. Rate models are displayed as *rate panels* under the Rates tab.

3.4.1 Rate model file

Model parameters are given in a human-readable¹ text file. The simplest way of browsing the rate file is to strip comments (`grep -v '#'`) which gives a TAB-delimited table that can be imported into Excel or other spreadsheet program. The table columns are \hat{t}_e , $\hat{\lambda}_e$, $\hat{\mu}_e$, $\hat{\kappa}_e$, followed by debug columns. Every row corresponds to an edge in the phylogeny, enumerated in a postorder traversal. (The debug information starts with the name of the edge's child node.) The rates and edge lengths are normalized by the total gene loss rate so that $\hat{\mu}_e = 1$ on every edge.

```
# length (t)      duplication (lambda)      loss (mu)      transfer (kappa) // ...
1.5820017453696325  0.38702704740849914  1.0  0.012376046605348303 // ...
4.139742169467891  0.08598802637984121  1.0  0.0021845554103576185 // ...
0.23206654225569434  0.47774576195966995  1.0  0.016986771452090894 // ...
0.157023248086316  0.569613362320818  1.0  0.012144222758613127 // ...
```

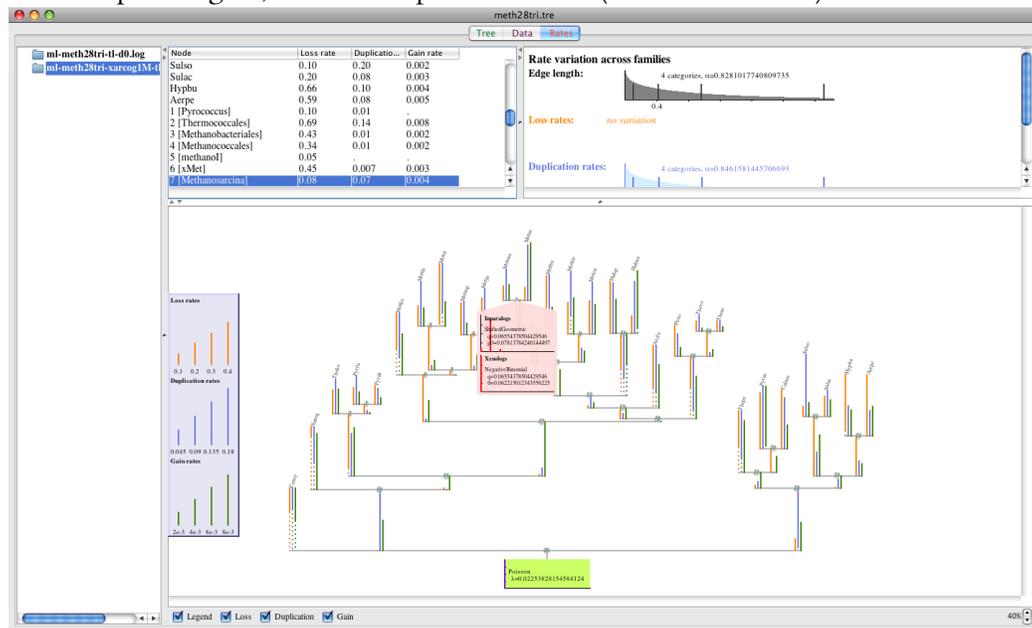
¹At least, technically.

The last lines of the rates file give the remaining model parameters such as the distribution of family-specific rate factors $t_f, \kappa_f, \mu_f, \lambda_f$, and the family size distribution at the root. In the example below, gene loss μ_f (“loss”) and gain κ_f (“transfer”) are constant, but gene duplication λ_f (“duplication”) and duration t_f (“length”) have Gamma distributions (shape parameters of 0.846... and 0.828...) discretized using 4 categories. The family size at the root has a Poisson distribution with mean 0.022....

```
|variation      duplication      4      0.8461581445706693      0.0
|variation      loss      1      1.0      0.0
|variation      transfer      1      1.0      0.0
|variation      length      4      0.8281017740809735
|root      Poisson      0.02253828154584124
```

3.4.2 Rates panel

The information panel for a rate model consists of three parts: a table showing numerical values of gain/loss/duplication rates (on the upper left), a graphical illustration of rate categories (on the upper right), and a graphical illustration of branch-specific gain, loss and duplication rates (in the lower half).



3.4.3 Rates panel: table

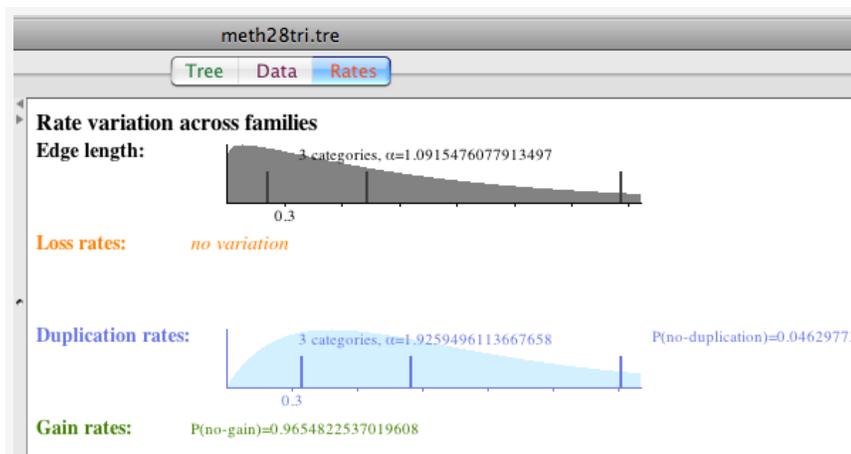
Node	Loss rate	Duplicatio...	Gain rate
Thevo	0.10	0.05	0.001
Theac	0.10	0.03	0.001
Thepe	1.0	0.26	0.008
Pyrae	0.53	0.20	0.01
Calma	0.56	0.16	0.004
Sulso	0.10	0.20	0.002
Sulac	0.20	0.08	0.003
Hypbu	0.66	0.10	0.004
Aerpe	0.59	0.08	0.005
1 [Pyrococcus]	0.10	0.01	.
2 [Thermococcales]	0.69	0.14	0.008
3 [Methanobacteriales]	0.43	0.01	0.002
4 [Methanococcales]	0.34	0.01	0.002

The table on the upper left-hand side gives the branch-specific prototypical loss, gain and duplication rates ($\hat{\mu}_e \hat{t}_e$, $\hat{\kappa}_e \hat{t}_e$ and $\hat{\lambda}_e \hat{t}_e$, respectively) for each branch. Branches are specified by the nodes they lead to. There are no rates next to the root node.

If a table row is selected, then the corresponding tree node is highlighted in the tree display on the bottom.

3.4.4 Rates panel: graphical display of rate variation

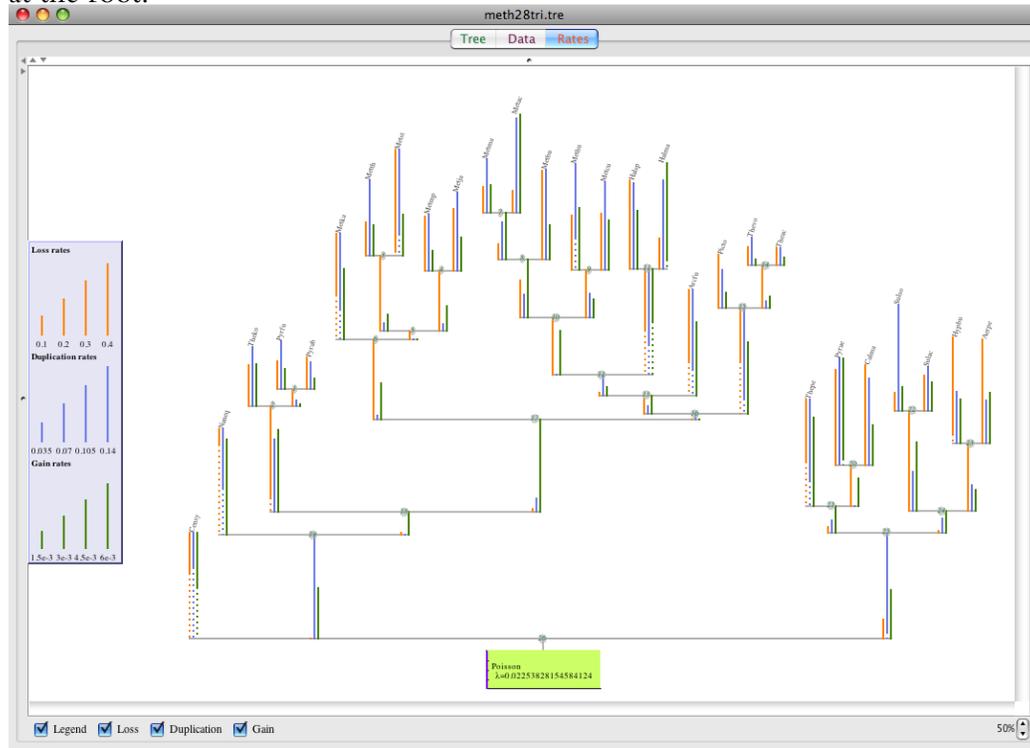
The prior distributions of family-specific rate factors are defined by possible no-gain ($\kappa_f = 0$) and no-duplication categories ($\lambda_f = 0$), and possible rate factors for the discretized Gamma distributions in case of edge length (t_f), loss rate (μ_f), duplication rate (λ_f) and gain rate (κ_f). The Gamma distribution plots also give the shape parameter α , and shade the corresponding continuous distribution.



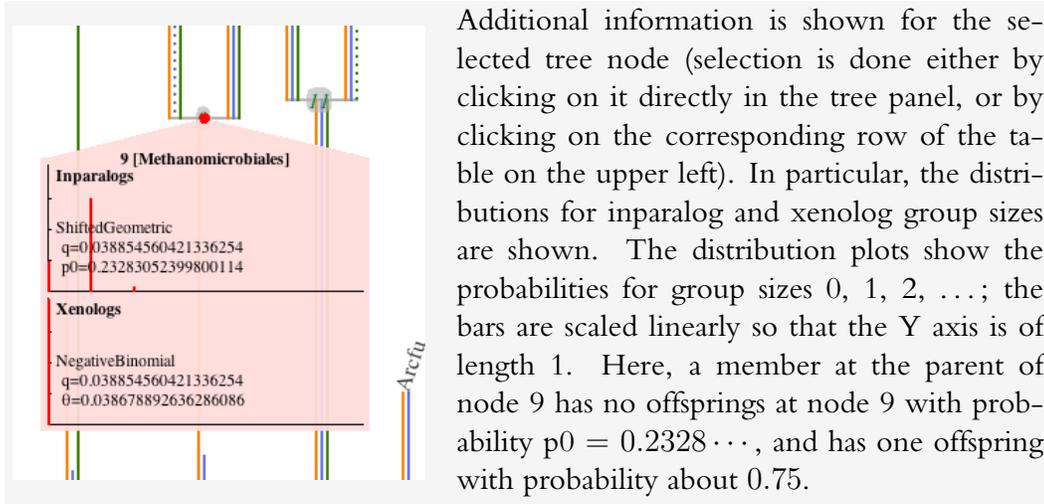
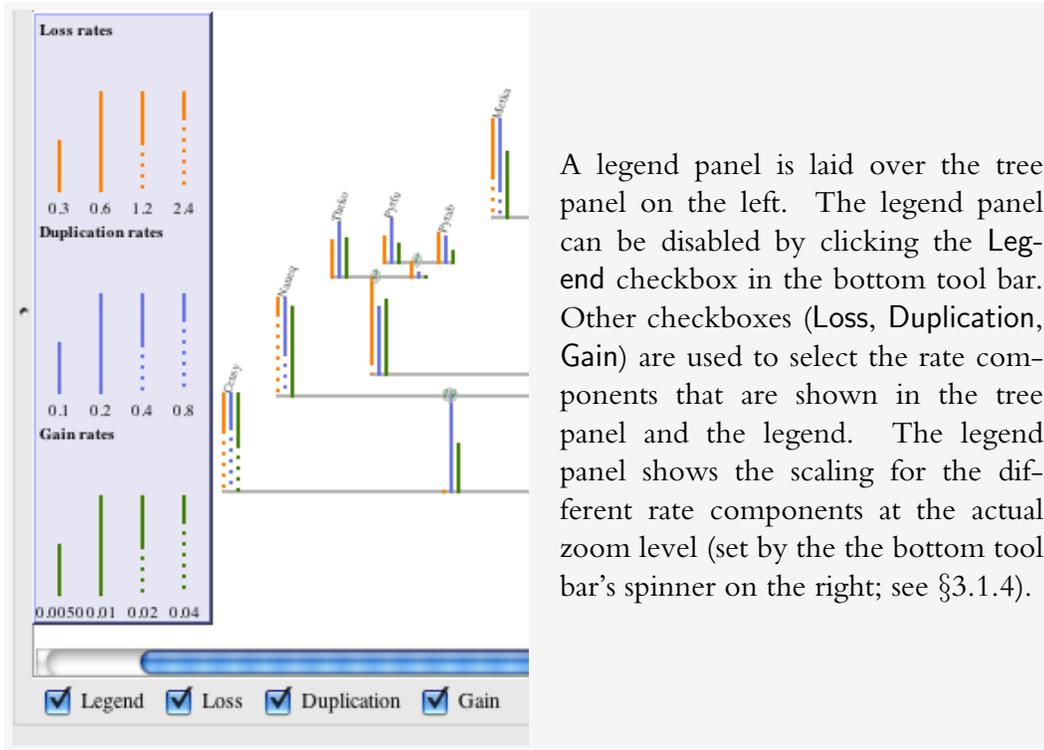
The upper right-hand side of the rates panel shows the variation of family-specific rate factors.

3.4.5 Rates panel: tree display

The bottom part of the rates panel displays branch-specific model components ($\hat{\mu}_e \hat{t}_e$, $\hat{\kappa}_e \hat{t}_e$ and $\hat{\lambda}_e \hat{t}_e$). The tree panel also shows the prior family size distribution at the root.



Rates may vary much across different lineages, and therefore it is not possible to have proportional branch lengths in the display. Instead, an “informative ellipsis” is used for long branches: the ratio between the solid part of the branch and the entire plotted branch length equals the ratio of the displayed branch length and the true branch length. For instance, if the displayed branch length corresponds to a loss rate of $\mu = \hat{\mu}_e \hat{t}_e = 1$, and the true loss rate is $\mu = 4$, then one-quarter of the displayed branch is solid and the rest is dotted.



3.4.6 Rate model optimization

COUNT computes the model parameters of the phylogenetic birth-and-death model by the numerical optimization of the likelihood. The likelihood com-

putation assumes that there are no all-0 profiles in the data set. It is therefore recommended that you first filter those families out before optimizing the likelihood. The simplest way to do that is to sort the family table by lineage-weight of the profile (column #lin), and double-click on a cell with #lin = 1. The popup selection includes the option of #lin \geq 1. Selected families can be then filtered into a separate table (Data \rightarrow Extract selected families. . . , see §3.3.3)

Prior to proceeding to the actual computation, optimization parameters need to be set in the window that appears after selecting the menu point Rates \rightarrow Optimize rates. . . .

Optimization parameters are grouped under the tabs Model type and Model parameters.

3.4.7 Rate optimization: model type

First, the initial model needs to be selected: this can be COUNT's predefined null model, or a previously computed rate model. This latter option is offered only if a rate model is selected in the Rates panel. The optimized model and parameters are initialized using the selected initial model.

Second, the optimized model architecture needs to be selected: gain-loss-duplication, duplication-loss, gain-loss, and pure loss. The most general model is the gain-loss-duplication model, where there is no restriction on the lineage-specific rates. In a duplication-loss model, all gain rates are zero ($\hat{\kappa}_e = 0$); in a gain-loss model, all duplication rates are zero ($\hat{\lambda}_e = 0$). In a pure loss model, both gain and duplication rates are zero.

Third, the type of the prior distribution at the root needs to be selected: this may be Poisson, negative binomial, or Bernoulli (point) distribution.

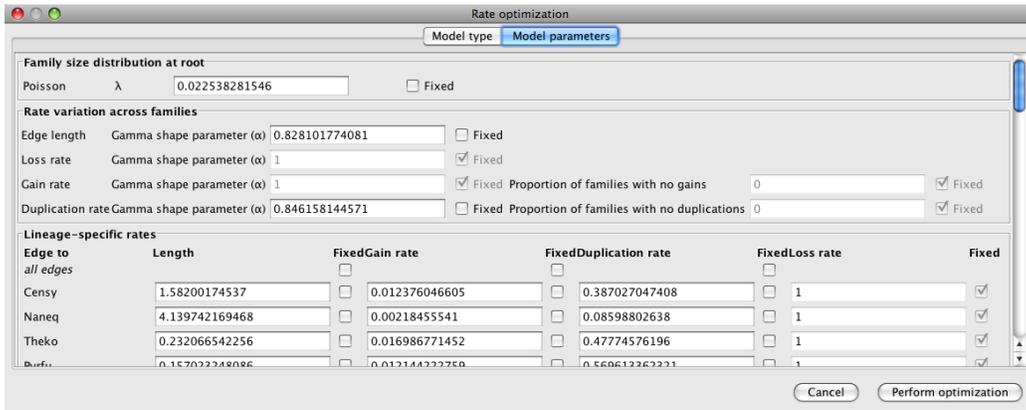
Fourth, it must be selected if duplication and gain rates may differ between tree edges. If, say, the "Same gain/loss ratio in all lineages" checkbox is selected, then the optimization assumes that $\hat{\kappa}_e = \kappa$ for some common gain rate κ , and optimizes the single model parameter κ along with \hat{t}_e and possibly $\hat{\lambda}_e$.

Fifth, the type of the rate variation across families needs to be chosen: this includes the number of discrete Gamma categories (= 1 if there is no Gamma variation), and possible no-duplication and no-gain categories.

The final set of parameters comprises computational parameters for the numerical optimization. The optimization proceeds in rounds: all model parameters are optimized once in each round. The optimization stops after the given maximum of optimization rounds, or earlier, when in two consecutive rounds, the log-likelihood (natural logarithm) changes by less than the given convergence threshold.

3.4.8 Rate optimization: model parameters

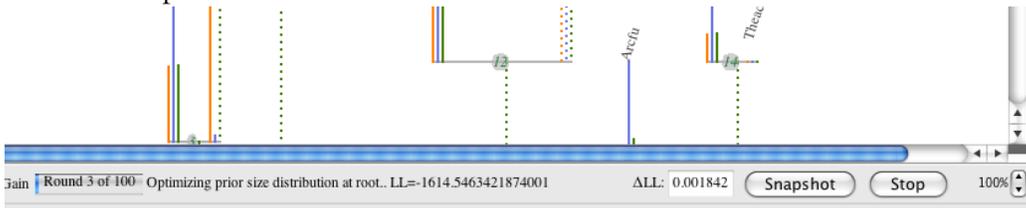
Under the "Model parameters" tab, you can set the initial values for all model parameters, as well as exclude certain parameters from the optimization. In order to exclude some parameter from the optimization, select its "Fixed" checkbox.



The model parameters include the prior family size distribution at the root (one or two parameters for Poisson, negative binomial, or Bernoulli distribution), parameters for rate variation across families (the set of tunable parameters depends on the rate variation type selected under the “Model type” tab), and lineage-specific rates. For technical reasons, the rates and edge lengths are scaled in such a way that the loss rate equals 1 on every edge. Note that the edge lengths of the input phylogeny are ignored in the probabilistic inference. Lineage-specific rates can be fixed individually, or all at once using the “master” checkboxes in the “all edges” row.

3.4.9 Rate optimization: computing

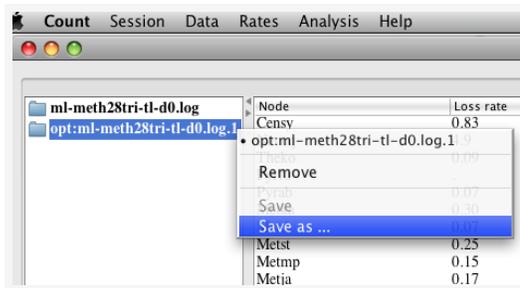
The actual optimization process starts when the Perform optimization button is pressed. The progress of the optimization can be followed in the rate model display that shows up.



The optimization is launched in a background process, and you can continue working with COUNT, performing other analysis steps. The rate model display will be updated continuously in the course of the optimization process. The display for the optimized model appears in the Rates browser. The bottom tool bar in optimized rate model displays includes progress indicators: a progress bar showing the current round, information about the current optimization step, the value of

the log-likelihood (LL), and its increase in the last round (ΔLL). You cannot use the rate model during optimization, and you should not save it (because it changes during the save). Instead, you can take *snapshots* of the current rate model during optimization, which will appear as descendant nodes of the optimized model display in the Rates browser. You can save the snapshot into a file, or perform ancestral reconstruction with it. The bottom tool bar has two specific buttons: one for taking snapshots of the current rate model during optimization, and another button for canceling the process (“Stop”).

After the optimization finished, you can use the optimized rate model as any other rate model: you can save it, or perform analyses with it.



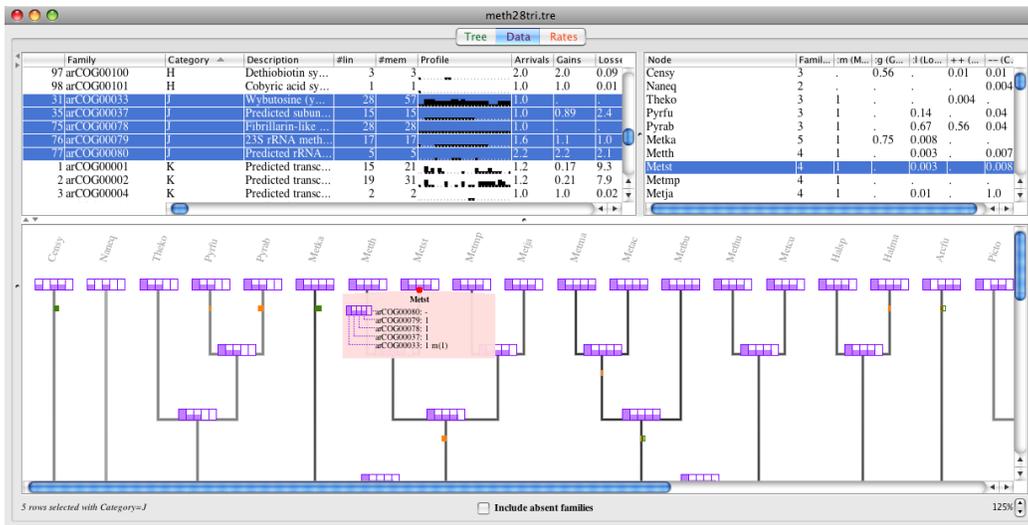
Rate models (including snapshots and finished optimizations) can be saved through the popup menu for the corresponding node in the Rates browser.

3.5 Analysis panels

You can perform ancestral inference and analyze family dynamics using the options available under the Analysis menu point. Namely, you can perform analysis by

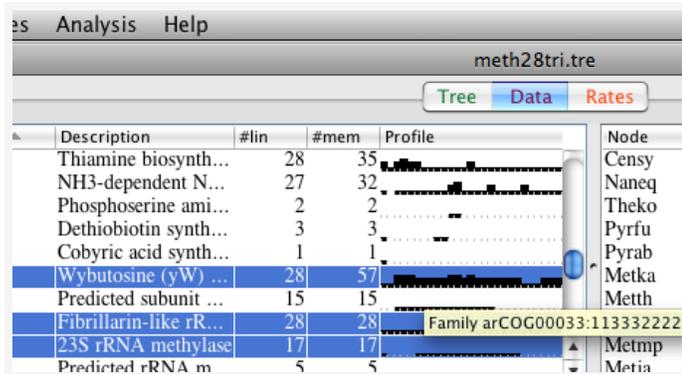
- Dollo parsimony,
- Wagner parsimony,
- posterior probabilities by the selected phylogenetic birth-and-death model, and
- Propensity for Gene Loss (PGL).

The analysis panels have the same basic design concept: they all consist of three parts. The three parts are (1) a table for family-specific information on the upper left, (2) a table for lineage-specific information on the upper right, and (3) a tree display on the bottom.



3.5.1 Analysis panel: family table

The family table on the upper left has a row for each family. Its columns include family index, family name, possibly family annotations, number of terminal lineages the family is present in (#lin), total number of members at terminal lineages (#mem), and phylogenetic profile. Additional columns are specific to the analysis methods.



The profile is depicted graphically: black bars show presence, with a height that is proportional to the logarithm of the family size at each node. The tool tip for a profile cell gives the exact numerical profile.

Multiple rows can be selected in the family table in the usual manner for your operating system (e.g., shift+click for range selection, Cmd-A or Ctrl-A for selecting all rows, etc.). The lineage table on the upper right shows sums (of gains, losses etc.) across the selected families by lineages. The tree display on the bottom illustrates the history of the selected families.

3.5.2 Analysis panel: lineage table

The lineage table on the upper right gives aggregate information over the selected families. Table rows correspond to lineages.

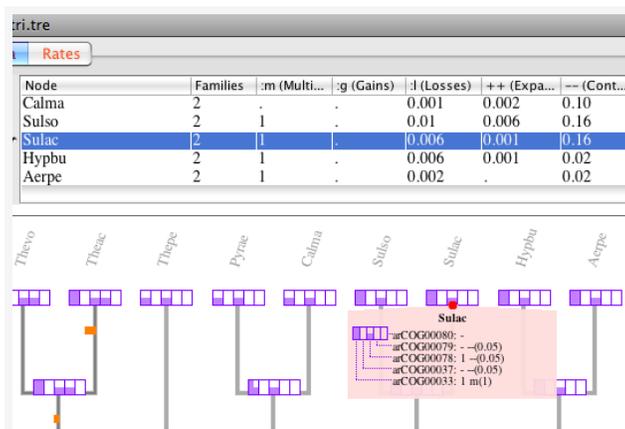
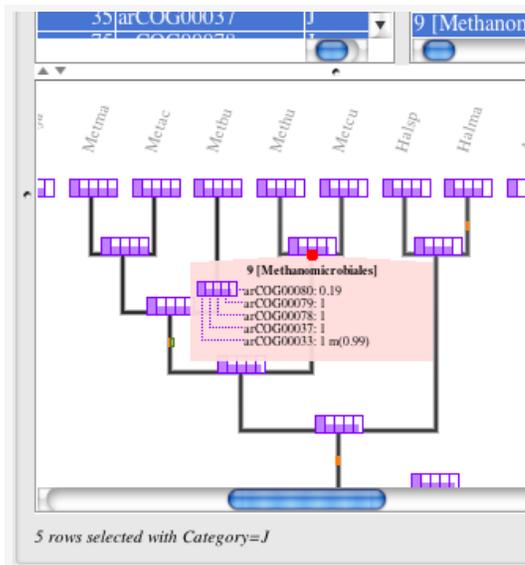


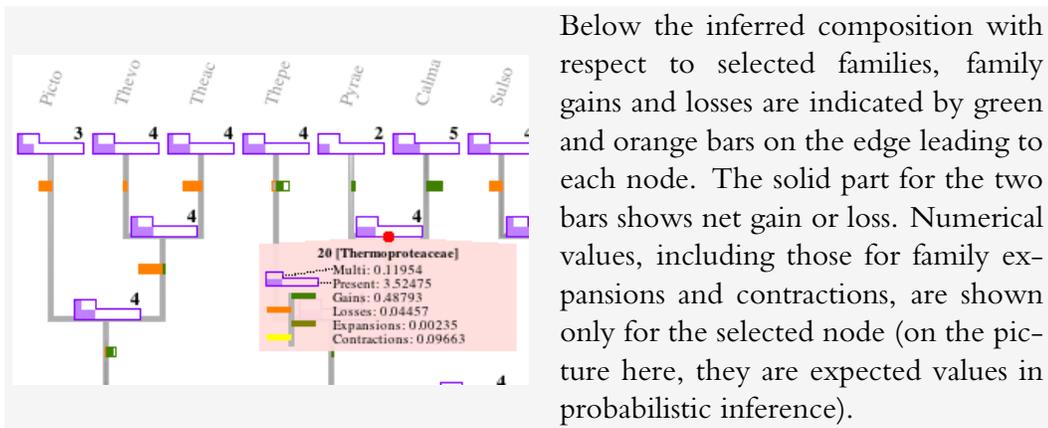
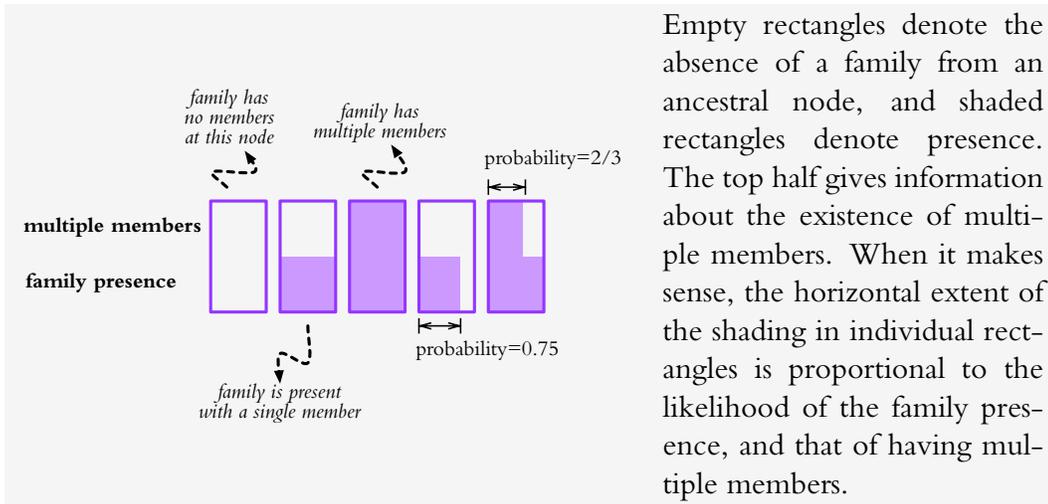
Table columns may include total number of families present (Families) and total number of multi-member families (:m) present at the node, as well as event totals on the edge leading to the node: family gains (:g), family losses (:l), expansions (++) and contractions (--).

3.5.3 Analysis panel: tree display

The tree display on the bottom illustrates the inferred history of the selected families.



The bottom tool bar in analysis panels gives information about the current selection in the family table. For less than seven selected families, their presence at nodes is illustrated individually, otherwise only aggregated values are shown by horizontal bars. Nodes in the tree panel can be selected directly, or through the lineage table. There is additional information displayed at the selected node.

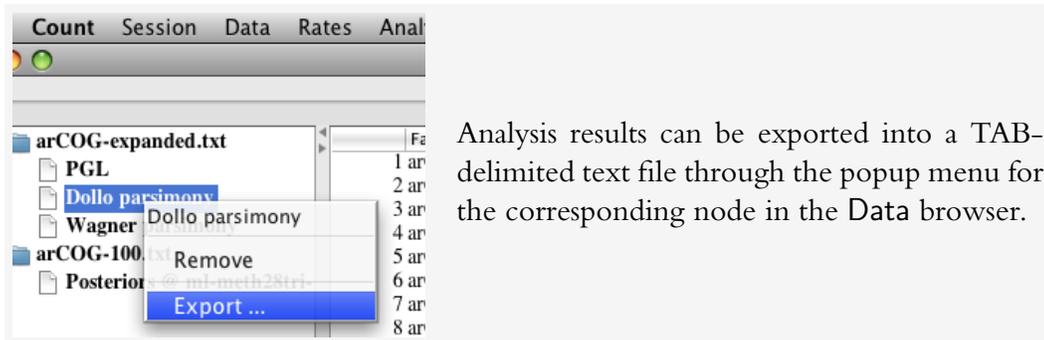
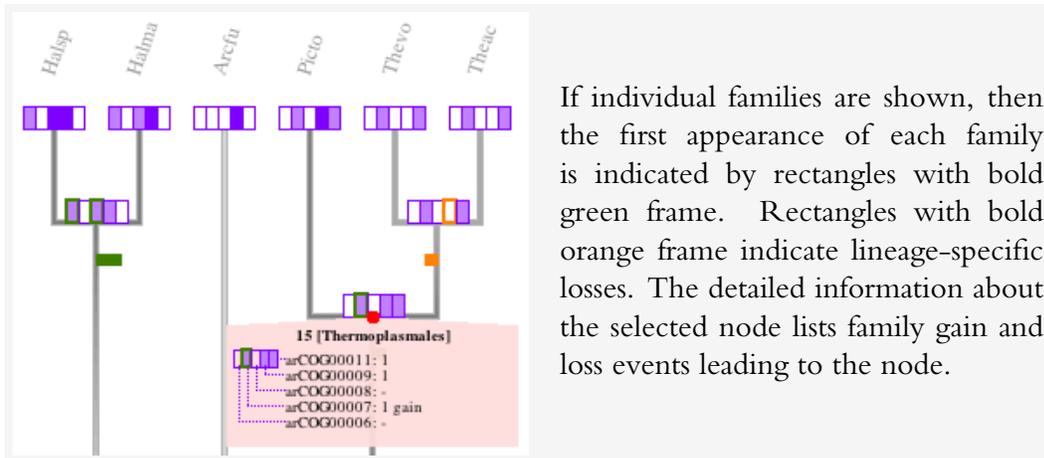


3.5.4 Analysis: Dollo parsimony

Ancestral presence in Dollo parsimony is inferred by assuming that each family appeared only once, and the presence-absence pattern is explained by lineage-specific losses. The ancestral reconstruction by Dollo parsimony is accessed through the menu

Analysis → Family history by Dollo parsimony.

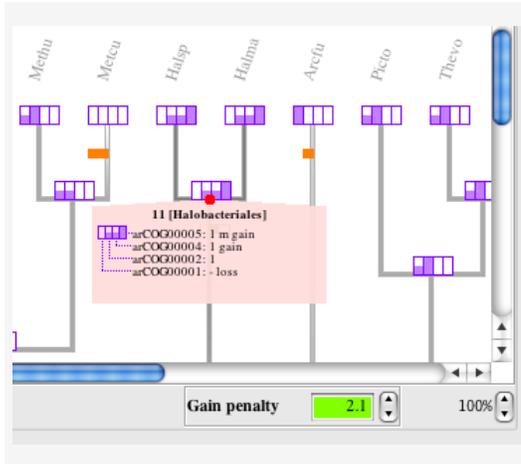
Multiple members are ignored in Dollo parsimony. Therefore, the lineage table does not give expansion and contraction counts, and the tree display illustrates family-level characteristics only (presence, gain and loss).



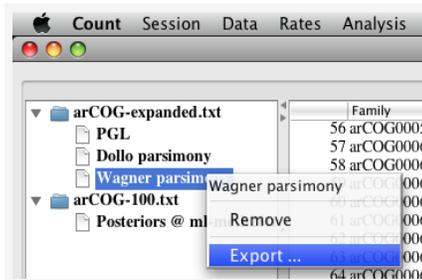
3.5.5 Analysis: Wagner parsimony

Wagner parsimony penalizes the loss and gain of individual family members, and infers the history with the minimum penalty. The ancestral reconstruction by Wagner parsimony is accessed through the menu

Analysis → Family history by Wagner parsimony.



The bottom tool bar includes a spinner for setting the gain penalty (loss penalty=1) in asymmetric Wagner parsimony. Recomputing the history with a new gain penalty value may take some time: the process is launched in the background. At the selected tree node, there is detailed information about individual families: presence/absence (1/-), multiple members (m), and family events on the edge leading to the node.

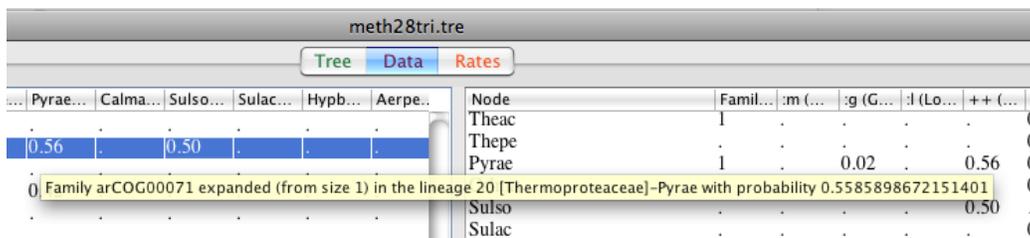


Analysis results can be exported into a TAB-delimited text file through the popup menu for the corresponding node in the Data browser.

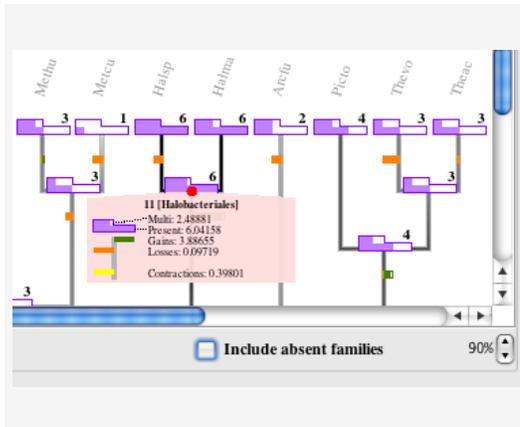
3.5.6 Analysis: Posteriors

Ancestral reconstruction by posteriors (see page 8) is accessed through the menu

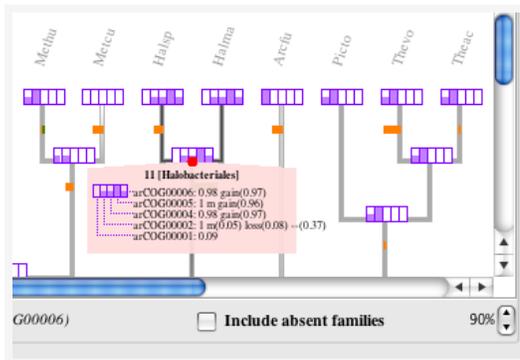
Analysis → Family history by posterior probabilities.



The family table on the upper left shows the computed ancestral reconstruction statistics (see (2.2) on page 9) and family dynamics (see (2.3) on page 9). Hover with the mouse over a cell to see the explanation of the displayed value.



At the selected tree node, there is detailed information about lineage-specific statistics (see (2.4) on page 10). The bottom tool bar includes a checkbox for including absent families in the lineage-specific statistics (see (2.5) on page 11). Computing the history may take substantial time (minutes or even hours): the process is launched in the background, and you can continue working with COUNT.



When individual families are shown at the selected node, then the presence probability is followed by the statistics (see (2.2) on page 9) with non-negligible values, including multi-member families (m), gains, losses, expansions (++) and contractions (--). Probabilities different from 0 and 1 are given in parentheses.

You can extract the posterior reconstruction into a TAB-delimited text file through the popup menu of the corresponding item in the Data browser. After selecting the file, you can specify which columns the file should include. The choices include posterior probabilities of rate categories, various statistics for the ancestral reconstruction and the family dynamics (see (2.2) and (2.3)), as well as the family annotations. The output file will have an additional row for the absent profile if the corresponding checkbox is selected. The output file format is discussed in more detail in Section 5.5.

Column selection for exporting posterior reconstruction

Select the columns to include: *(exporting into posteriors-100.txt)*

<input checked="" type="checkbox"/> Family profile	<input type="checkbox"/> Annotations	<input type="checkbox"/> Include absent families
<input checked="" type="checkbox"/> Presence at ancestral nodes	<input checked="" type="checkbox"/> Multiple homologs at ancestors	<input checked="" type="checkbox"/> Rate categories
<input checked="" type="checkbox"/> Lineage-specific family gains	<input checked="" type="checkbox"/> Lineage-specific family losses	<input checked="" type="checkbox"/> Totals across lineages
	<input checked="" type="checkbox"/> Lineage-specific expansions	<input checked="" type="checkbox"/> Lineage-specific contractions

3.5.7 Analysis: Propensity for gene loss

COUNT can compute the so-called PGL (propensity for gene loss) index, introduced by Krylov et al. [Krylov, Wolf, Rogozin, Koonin. “Gene loss, protein sequence divergence, gene dispensability, expression level, and interactivity are correlated in eukaryotic evolution.” *Genome Research*, **13**:2229–2235, 2003]. PGL is defined for a family as

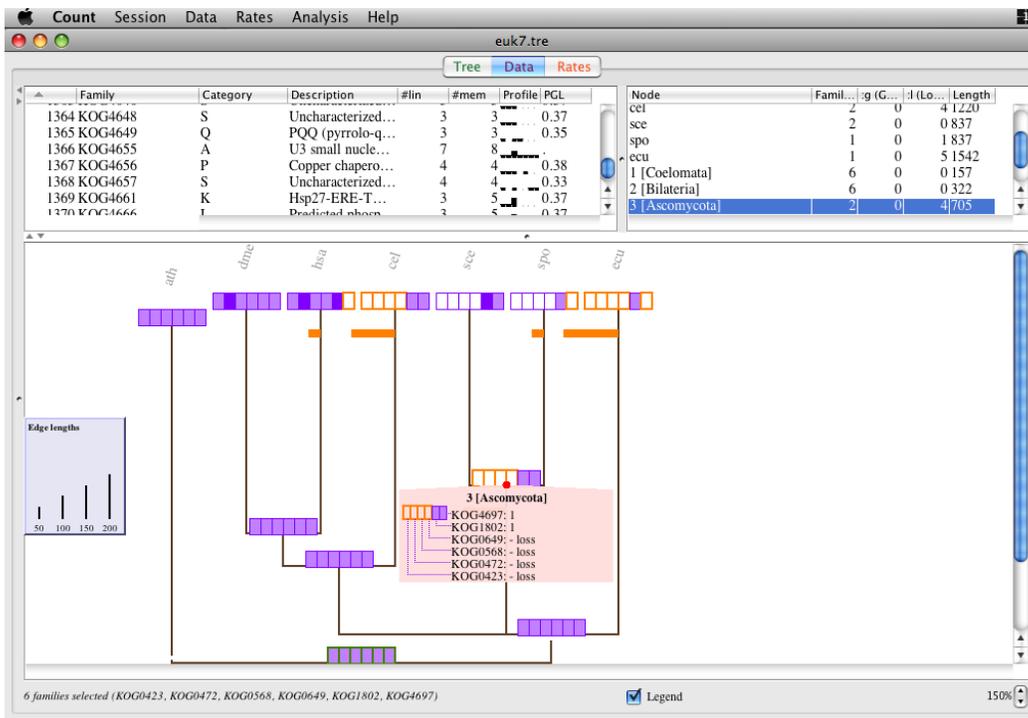
$$\frac{\sum_e \text{loss}(e) \cdot \text{length}(e)}{\sum_e \text{length}(e)},$$

where $\text{length}(e)$ is edge length (e.g., time between speciation events measured in million years), and $\text{loss}(e)$ is an indicator for the optimal Dollo parsimony reconstruction: $\text{loss}(e) = 1$ if the reconstruction posits a loss on edge e , otherwise $\text{loss}(e) = 0$. The summation goes over the edges in the subtree rooted at the first appearance of the family. COUNT uses the edge lengths in the session’s main phylogeny.

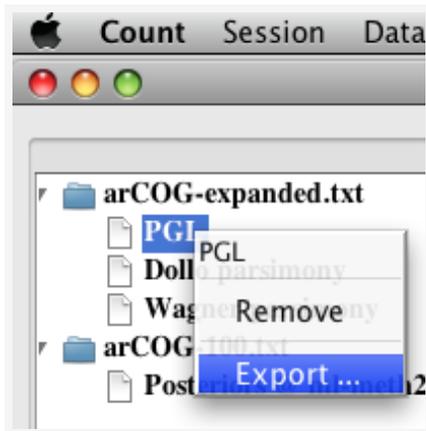
Typically, one is interested in families that originate at the same ancestor. In COUNT, you can select all such families by performing Dollo parsimony reconstruction (Analysis → Family history by Dollo parsimony), and double-clicking on a family with presence at the ancestral node o you are interested in. The popup selection menu includes the option $o \geq 1$ if you clicked on a cell with value ‘1’ in the column o . Extract the filtered rows into a new table (Data → Extract selected families. . .), and calculate PGL on that table only.

PGL computation is accessed through the menu

Analysis → PGL: propensity for gene loss (Krylov-Wolf-Rogozin-Koonin).



The tree display in PGL has an overlaid legend for edge length. The legend can be disabled by selecting the checkbox in the bottom tool bar.



Analysis results can be exported into a TAB-delimited text file through the popup menu for the corresponding node in the Data browser.

Chapter 4

Test data

I included some test data in the distribution, so that you can try out different functionalities, and verify the syntax of different input files. Test data are packaged in `test.tar.gz`, which expands into files in a `test` directory. The tests include a data set for archaeal gene content evolution [Csűrös M, Rogozin IB, and Koonin EV, “Streamlining and large ancestral genomes in Archaea inferred with a phylogenetic birth-and-death model,” *Molecular Biology and Evolution*, <http://mbe.oxfordjournals.org/cgi/content/abstract/msp123>, 2008], and a data set for gene loss propensity in eukaryotes [Krylov, Wolf, Rogozin, Koonin. “Gene loss, protein sequence divergence, gene dispensability, expression level, and interactivity are correlated in eukaryotic evolution.” *Genome Research*, 13:2229–2235, 2003].

Archaeal data set. The archaeal data set includes the following files.

`meth28tri.tre`: Newick-format phylogeny for 28 Archaea.

`meth28tri.table`: Family size table for arCOGs and lineage-specific families.

`meth28tri.rates`: Rate file.

`meth28tri.xml.gz`: Saved session that includes ancestral reconstruction by posteriors.

Open the session file (select `meth28tri.xml.gz` immediately after launching COUNT under the menu point `Session` → `Open previously saved session(s) ...`), or start a new session yourself (select `meth28tri.tre` under the menu point `Session` →

Start new session...), and load the family size table after (select `meth28tri.table` under the menu point `Data` → `Open table ...`).

Eukaryotic data set. The eukaryotic data set includes the following files.

`KOGs-euk7.tre`: Newick-format phylogeny for 7 eukaryotes.

`KOGs-annotated.txt`: Annotated (category and description) family size table for KOGs.

`KOGs-euk7.xml`: Saved session that includes PGL.

Open the session file (select `KOGs-euk7.xml` immediately after launching `COUNT` under the menu point `Session` → `Open previously saved session(s) ...`), or start a new session yourself (select `KOGs-euk7.tre` under the menu point `Session` → `Start new session...`), and load the annotated family size table after (select `KOGs-annotated.txt` under the menu point `Data` → `Open annotated table...`).

Chapter 5

Command-line usage

5.1 Overview

The command-line interface includes the modules shown in the table below.

Analysis method	Program
Ancestral gene content by Wagner parsimony	AsymmetricWagner
Parameter optimization for phylogenetic birth-and-death models	ML
Ancestral gene content by phylogenetic birth-and-death model	Posteriors

COUNT is written in Java (SE 6), and is packaged in the JAR file `Count.jar`. Every module `P` (where `P=ML,AsymmetricWagner,Posteriors`) can be launched in the command shell by

```
java -Xmx2048M -cp Count.jar ca.umontreal.iro.evolution.genecontent.P a b ...
```

Here “java” is the Java tool on your operating system: JDK 6 is required for using COUNT¹. The Java option `-Xmx2048M` allocates 2Gbytes of memory: this can be adjusted to match your hardware: instead of 2048M, one can write 1024M (1 Gbytes), 512M (512 Mbytes), 4096M (4 Gbytes), etc. I will discuss the application-specific parameters (“`a b ...`”) for each program `P` later.

¹I think that the command-line programs will run under JDK 5.0 too, but I have not tested that.

5.2 Executables

5.2.1 Output

The programs write to the standard output (`stdout`). You will probably want to redirect the output:

```
java ... P a b ... > result
```

where `result` is the name of a text file.

5.2.2 Comments

Output and input files may contain comments in lines starting with `#`. Such lines are ignored on input, and do not contain essential information. If necessary (e.g., in order to prepare for import into Excel), they are easily filtered out in the usual manner

```
grep -v '#' result > stripped
```

5.2.3 Data formats

The programs work with three basic data formats: a family size **table**, a phylogenetic **tree**, and phylogenetic birth-and-death model parameters (**rates**).

5.3 Wagner parsimony

Wagner parsimony is computed by the `AsymmetricWagner` application:

```
...AsymmetricWagner [options] tree table
```

The application-specific options are the following.

- `gain` g Relative penalty of a gain with respect to loss. When $g > 1$, scattered distributions are explained by multiple losses; conversely, $g < 1$ favors multiple gains (i.e., lateral transfers). Setting $g = 1$ is traditional Wagner parsimony.
- `max.paralogs` m The table rows are filtered by size: the total number of homologs is limited at m . In other words, only families with $m \geq \sum_{j=1}^m \Phi_{fj}$ are used in the inference.

The output consists of three parts: family sizes at ancestral taxa (lines starting with “# FAMILY”), genome sizes (lines starting with “# PRESENT”), and lineage-specific gene family size changes (lines starting with “# CHANGE”). I recommend splitting these three parts using simple scripts to get TAB-delimited tables:

```
grep '# FAMILY' result > families.txt
```

A “# FAMILY” line lists the family sizes at each taxon that minimize the parsimony penalty of the reconstruction. Further columns give the number of lineages where the family was lost (“Losses”) or newly appeared (“Gains”), or where it expanded (“Expansions”) and reduced (“Reductions”) in size.

```
# FAMILY name Natph Halsp Halwa ... root Gains Losses Expansions Reductions
# FAMILY arCOG00001 0 0 0 ... 1 0 1 1 0
# FAMILY arCOG00002 1 1 1 ... 2 0 0 0 3
```

Lines starting with “# PRESENT” aggregate the same information across different families: for every taxon, a line gives the number of families with positive size, as well as the total of the family sizes (i.e., number of all genes). Lines starting with “# CHANGE” give aggregate information on lineage-specific changes.

5.4 Model parameters

Parameters of a phylogenetic birth-and-death model can be computed by the ML program, which maximizes the likelihood using numerical optimization. Rates and edge lengths are normalized in ML so that $\hat{\mu}_e = 1$ holds on every edge e . The optimization is invoked by the following syntax.

```
... ML [options] tree table [rates]
```

(Bracketed arguments [...] are optional.) The following application-specific options can be used.

- max.paralogs m The table rows are filtered by size: the total number of homologs is limited at m . In other words, only families with $m \geq \sum_{j=1}^m \Phi_{fj}$ are used in the inference.
- opt.rounds R Sets the maximum number of iterations in the optimization. (I found that $R = 100$ is sufficiently large.)

- opt_eps ϵ Sets the convergence threshold for the optimization. The procedure stops if the log-likelihood decreases by a relative value of ϵ in two consecutive iterations. (I think that $\epsilon = 0.1, 0.01$ are conservative enough.)
- uniform_gain true Enforces the same gain rate on all edges ($\hat{\kappa}_e$ does not vary with e).
- uniform_duplication true Enforces the same duplication rate on all edges ($\hat{\lambda}_e$ does not vary with e).
- uniform_length true Enforces the same edge length everywhere (\hat{t}_e does not vary with e).
- gain_k k Number of discrete categories for the Gamma distribution of the family-specific gain rate factor (κ_f).
- loss_k k Number of discrete categories for the Gamma distribution of the family-specific loss rate factor (μ_f).
- duplication_k k Number of discrete categories for the Gamma distribution of the family-specific duplication rate factor (λ_f).
- length_k k Number of discrete categories for the Gamma distribution of the family-specific edge length multiplier (t_f).

If a rate file is specified (`rates`), then the initial values for the optimization are taken from there. Using more than $k = 3, 4$ categories may slow down the optimization dramatically. It is a good idea to perform the optimization in model hierarchy. Starting with a simple model (`-uniform_duplication true -gain_k 1 -loss_k 1 -duplication_k 1`), more and more rate variation can be introduced, in order to shorten the total time for optimization, and to get useful partial results along the way. For instance, given a tree in `ex.tre` and a table in `ex.txt`, the following series of commands culminate in a 4×4 -category rate variation model.

```
ML -uniform_duplication true ex.tre ex.txt > ex1.r
ML ex.tre ex.txt ex1.r > ex2.r
ML -max_paralogs 100 -length_k 3 ex.tre ex.txt ex2.r > ex3.r
ML -max_paralogs 100 -length_k 3 -duplication_k 3 ex.tre ex.txt ex3.r > ex4.r
ML -max_paralogs 10000 -length_k 4 -duplication_k 4 ex.tre ex.txt ex4.r > ex5.r
```

Here, `ex1.r` is the simplest model, `ex2.r` has lineage-specific rates, `ex3.r` overlays three categories for family-specific edge length variation, `ex4.r` mixes in three categories for family-specific duplication rate variation, and `ex5.r` refines the rate variation to 4×4 categories. The intermediate results filter out extremely large families (`-max_paralogs`): the run time grows quadratically with the largest family. For large tables and complicated models, the optimization may take long, up to 1–2 CPU days. The application specific switch `-v true` turns on verbose logging. In verbose mode, the ML program regularly reports on the progress of the optimization in lines starting with `***`. I usually launch the optimization by piping the output through `tee`: `...ML -v true ... | tee ex.r &`. Then, the output tracks the current iteration round of the optimization (“round”), the current value of the [negative] log-likelihood (“ll”), as well as the most recent decrease of the log-likelihood (“delta”).

```
...
***ML.o round 3 ll 112419.67853100887 delta 0.22343630842806306
...
```

5.5 Inference of ancestral gene content

The `Posteriors` application infers ancestral gene content by posterior probabilities in a phylogenetic birth-and-death model. In particular, the following values are computed for each taxon (tree node) or lineage (tree edge), and each family.

- **Family size.** The program computes two probability values: whether a family has/had 1 (p_1) or multiple ($p_{>1}$) members at a given (ancestral or terminal) taxon. The absence probability (family size of 0) can be computed as $(1 - p_1 - p_{>1})$. For taxon x , the posterior probabilities p_1 and $p_{>1}$ are listed under column headers $x:1$ and $x:m$, respectively. (Here it comes handy if the input tree’s internal nodes were named explicitly: x will be that name instead of a machine-generated code.)
- **Family gain and loss.** The program computes the probabilities that in a given lineage, the family size changed from a positive number to 0 (loss), or from 0 to a positive number (gain). The loss and gain probabilities for the edge leading to taxon x are given under the column headers $x:loss$ and $x:gain$, respectively.
- **Expansions and reductions.** The program also computes the probabilities for size changes in retained families along each lineage. Expansions (size

change from 1 to something larger) and reductions (size change from 2 or more to 1) on the edge leading to taxon x are listed under the headers x :`expansion` and x :`reduction`.

- **Rate categories.** If the model has non-constant family-specific rate variations, then the output of `Posteriors` includes the posterior probabilities for families belonging into discrete categories. The columns for these probabilities have headers in the syntax of Cc/p , where c is the category's machine-generated identifier (a positive integer), and p is a point in the lattice defined by the Cartesian product of the discrete category indices. For example, `C43/e2,d3,l0,t0`, is rate class 43, in which the edge length (`e`), duplication rate (`d`), loss rate (`l`) and gain rate (`t`) category indices are 2,3,0 and 0, respectively.

Not that the program reports the posterior probabilities also for the empty phylogenetic profile in a line where family name is `ABSENT`.

```
Family ... Censy:1 Censy:m Censy:gain Censy:loss Censy:expansion Censy:reduction ...
...
arCOG00001 ... 0.0 0.0 0.0 0.0 0.0 0.9985936233867079
```

The application is launched in the following syntax.

```
...Posteriors [options] tree table rates
```

Available options:

- `max_paralogs` m The table rows are filtered by size: the total number of homologs is limited at m . In other words, only families with $m \geq \sum_{j=1}^m \Phi_{fj}$ are used in the inference.
- `lineage_totals` `true` The output gives only lineage-specific expected totals, without posterior probabilities for individual families. The expected values include the correction for absent families (an unknown number of families with an all-0 profile) For example, the $x:1$ column gives the inferred estimate for the number of families with exactly 1 member at node x .