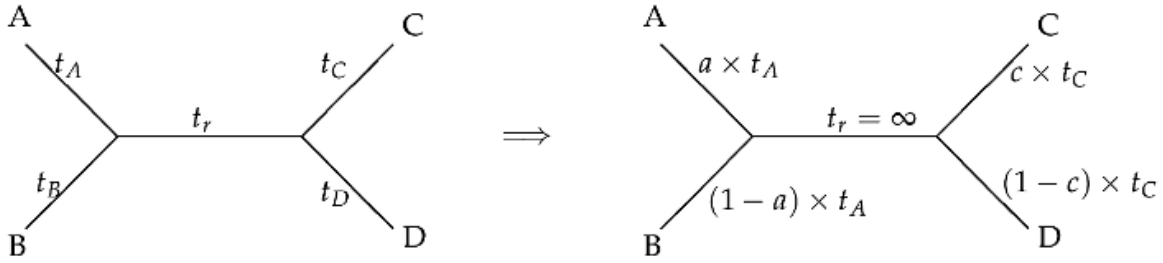
Supplementary Material - Simulations of Theobald's common origin test

Independent origin as a special case of the common origin

First we should notice that for a given fixed evolutionary model, the phylogenetic trees representing the hypothesis of independent origins can be represented by a single phylogenetic tree with the corresponding vanishing branches with infinite lengths.
Therefore the independent origins model is a particular case of the model of common origin where for each independent origin three internal branches are fixed - one at ∞, representing the *de novo* appearance, and one at each of its side become redundant by the pulley principle.

The following diagram represent the common origin model at the left and the independent model at the right, where we can see that after the "removal" of the internal branch the remaining branches have one less degree of freedom since the likelihood is the same whenever their sum is the same. The parameters a and c are constants between zero and one and a natural choice is $a = c = 1$, while A, B, C and D are subtrees.



The justification for fixing the branch length at inifinity comes from the fact that the Markov chains used in evolutionary models converge to their equilibrium distributions. That is, the

probability $P(x \mid z, t, M)$ of going from state z to state x in time t under evolutionary model M becomes independent of z when $t \to \infty$ and approaches π_x, the equilibrium frequency of x.

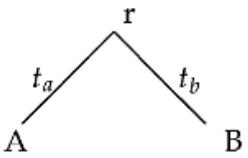
The likelihood $P(X \mid T, \mathbf{t}, M)$ of a phylogenetic tree T with branch length vector $\mathbf{t}$ arbitrarily rooted at r can be calculated for a given alignment column X as

$$P(X \mid T, \mathbf{t}, M) = \sum_z \pi_z L_r(z \mid \mathbf{t}, M) \quad (1)$$

where $L_r(z \mid \mathbf{t}, M)$ is the partial likelihood of node r for amino acid state z, and can be calculated recursively by

$$L_r(z \mid \mathbf{t}, M) = \left[\sum_x P(x \mid z, t_a, M) L_A(x \mid \mathbf{t}, M)\right]\left[\sum_y P(y \mid z, t_b, M) L_B(y \mid \mathbf{t}, M)\right] \quad (2)$$

Assuming the subtree with branch lengths $t_a, t_b \in \mathbf{t}$ represented by



Under the independent hypothesis $P(x \mid z, t_a, M) = \pi_x$ and $P(y \mid z, t_b, M) = \pi_y$ since t_a and t_b go to infinity, and therefore we have that equation 2 reduces to

$$L_r(z \mid \mathbf{t}, M) = \left[\sum_x \pi_x L_A(x \mid \mathbf{t}, M)\right]\left[\sum_y \pi_y L_B(y \mid \mathbf{t}, M)\right] = W \quad (3)$$

which is independent of z, and thus the site likelihood of equation 1 is

$$P(X \mid T, \mathbf{t}, M) = \sum_z \pi_z L_r(z \mid \mathbf{t}, M) = \sum_z \pi_z W = W \sum_z \pi_z = W \quad (4)$$

By comparing each of the two terms in equation 3 and equation 1 we can see that W is the product of the site likelihoods of two independent trees, arbitrarily rooted at A and B.

The extension for distinct models over the tree is straightforward, with the caveat that despite it can be handled by sequence simulation programs like INDELible, it is not implemented yet in popular phylogenetic reconstruction methods.

Artificial protein sequences with significant BLAST similarity

We reevaluated the likelihood of the synthetic datasets created by D. Theobald and described in subsections 4.2 and 4.3 of his Supplementary Information. We calculated the log-likelihood (LnL) of the datasets under both hipothesis using codeml and phyml. It is important to use the same program to compare the LnL values under the two hypothesis since there might be roundoff errors and other errors associated to the approximations. We report here the analyses conducted with phyml, but we also provide the scripts for working with codeml.

In subsection 4.2 of the original article he uses two sequences (only one branch), and in subsection 4.3 he works with three sequences (three internal branches with only one possible topology). Our strategy for both

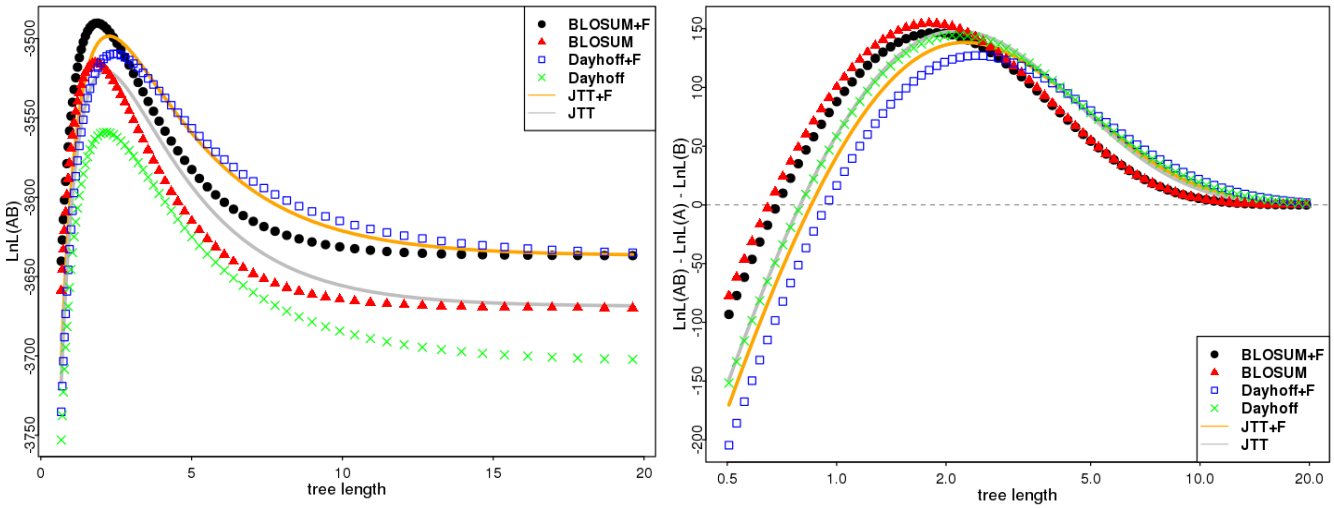
cases was to calculate the LnL over a grid of values for the branch lengths under a fixed model, where the independent hypothesis can be represented by an infinite branch length (in practice we set it to a very large value).

Two sequences

Phyml can not work directly with ony two sequences (or at least we are unaware of such an option), but we can circumvent that by creating a quartet such that the each pair of neighbour sequences are identical and have a branch length of zero. That is, a tree of the form ((A:0, A:0):x, (B:0, B:0)) where x is the internal branch length that will be made to vary.

We generated a log-spaced grid of 130 values for the branch length ranging from 0.2 to 108, and calculated the LnL under the Blosum62, Dayhoff and JTT replacement matrices where for each matrix we used the model-defined equilibrium frequencies or the empirically-derived ones ("+F"). We furthermore assumed that there was no gamma heterogeneity of rates among sites. Since it is essencial that we use the same software (in our case, phyml) to calculate the LnL under the common origin and independent origin hypoteses, we approximate the infinite branch length of the independent origins by a large value (in phyml it is explicitly set a t 100).

The results are summarized in the figure below. The left panel represent the LnL values for each branch length, under the six models. We can see that under all models the maximum likelihoood (ML) value is achieved at a finite, small branch length (between 1.8 and 2.4). To understand how better each branch length fares compared to the independent origin hypothesis, we subtracted each LnL value by the LnL value at "infinity" which in our case is the branch length equal to 108 (that is truncated to 100 by phyml). This likelihood ratio test is represented on the right panel.



It is worth noticing that there are branch length values that have a lower likelihood than the infinite length one. For example, the independent hypothesis would be favored for all lengths shorter than 0.64 under the Blosum62 matrix (or below 0.67 under Blosum62+F), which might explain the result obtained by D. Theobald using such combination of model and branch length.

implementation directory: *test002*

This directory contains the script run_phyml.py, based on phyml. The output is on file table_LnL.txt that contains the columns:

1. tree length (defined by grid);
2. tree length (used by phyml, which should correspond to the grid value except for exceedingly large values, that are truncated by phyml);
3. BLOSUM62 matrix using empirical equilibrium frequencies;
4. BLOSUM62 matrix using model-defined equilibrium frequencies;
5. Dayhoff matrix using empirical equilibrium frequencies;
6. Dayhoff matrix using model-defined equilibrium frequencies;

7. JTT matrix using empirical equilibrium frequencies;
8. JTT matrix using model-defined equilibrium frequencies;

Here we assume no gamma heterogeneity of rates. The program `plot_blen.R` can then be used in R to plot the graphics.

extra: ***test001***

This contains the script that does the codeml analysis (codeml belongs to the PAML package). The analysis can be reproduced by running the script `run_paml_grid_blen.sh`, that depends on the phylip file `seqpair.phy` with two sequences. The log-likelihood(LnL) values will be output to file `paml2.txt`, which contains the columns: tree length, branch length (half the tree length), and four LnL values representing models BLOSUM62+F, BLOSUM62, DAYHOFF+F, DAYHOFF respectively.

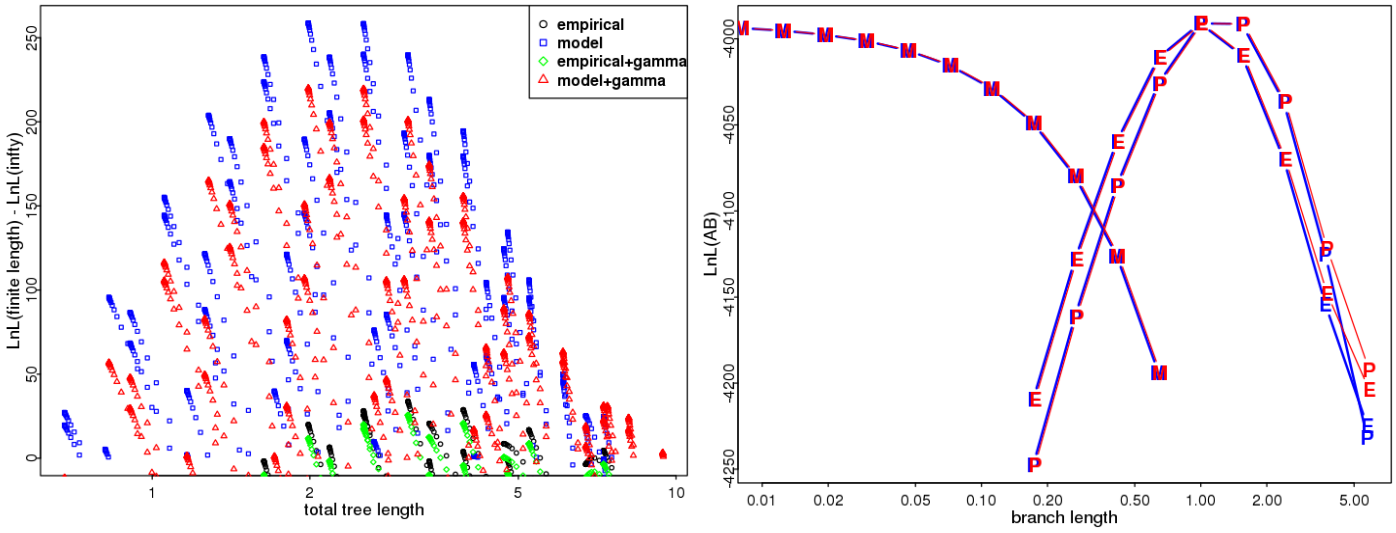
The BLOSUM62 matrix values was imported from MrBayes since it was not available in PAML. Notice that contrary to phyml, codeml does work with a pair of sequences. The program `plot_blen.R` can then be used in R to plot the graphics.

Three sequences

For the three sequences presented in subsection 4.3 of D. Theobald's Supplementary Material, we again employed the above strategy of calculating LnL values over a grid of values, with the caveat that in the case of a triplet we have three branches over which to vary their lenghts.

For each branch we varied its value between 0.0001 and 100, producing 29 points almost evenly spaced on a logarithmic scale. We assumed the WAG replacement matrix under four distinct scenarios: presence or absence of gamma-distributed rate heterogeneity, and model-defined versus empirically-derived equilibrium amino-acid frequencies.

The results are shown below, where again we assume that a large enough length is equivalent to the independent hypothesis. At the left we have the difference between the LnL of "short" trees (small value for the sum of the branch lengths) and the best "large" tree (ML tree with sum of branch lengths larger than 20). Each point is a combination of branch lengths, and at the right we have the LnL values for the individual branch lengths where the others were fixed at their ML values. The labels "M", "E", and "P" refer respectively to the branches of the artifical sequences "mytu1", "esco1" and "pogi1".



Here again we observe that in fact the common origin hypothesis would be favored, and we suspect that D. Theobald might have used distinct programs to do the calculations or the program failed to exploit the whole parameter space.

implementation directory: ***test004***

The program run_phyml.py is based on phyml software and will use the input file seqtri.phy and create an output file called table_LnL.txt. The R script plot_blen.R will then plot the results. The output file will contain the columns:

1. total tree length (sum of the three branch lengths)
2. length of branch leading to “esco1” sequence
3. length of branch leading to “mytu1” sequence
4. length of branch leading to “pogi1” sequence
5. LnL value for WAG model with empirical frequencies and homogeneous rates
6. LnL value for WAG model with model-based frequencies and homogeneous rates
7. LnL value for WAG model with empirical frequencies and variable rates following gamma
8. LnL value for WAG model with model-based frequencies and variable rates following gamma

extra: *test003*

The codeml-based script run_pamltri_grid_blen.sh will create an output file paml3.txt with columns:

1. length of branch leading to “esco1” sequence
2. length of branch leading to “mytu1” sequence
3. length of branch leading to “pogi1” sequence
4. LnL value for WAG model with empirical frequencies and homogeneous rates
5. LnL value for Blosum62 model with empirical frequencies and homogeneous rates
6. LnL value for Blosum62 model with model-based frequencies and homogeneous rates

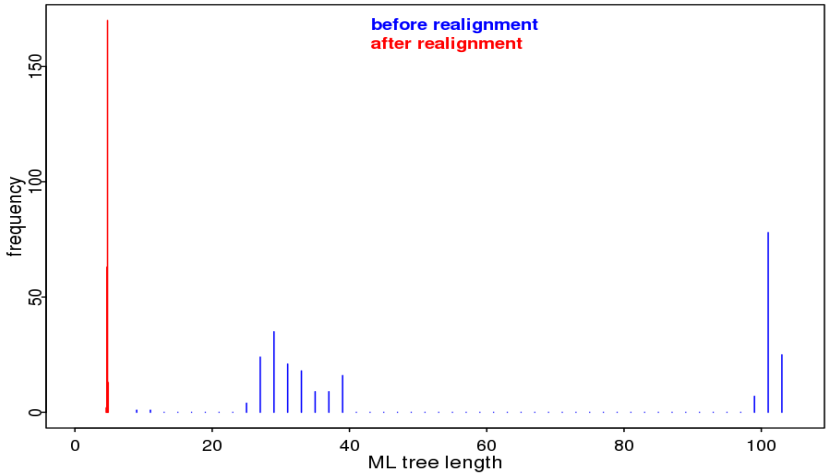
There’s no script to interpret the output, but on Unix we can always use “sort -n -k paml3.txt” or read it into R.

Randomly shuffled data

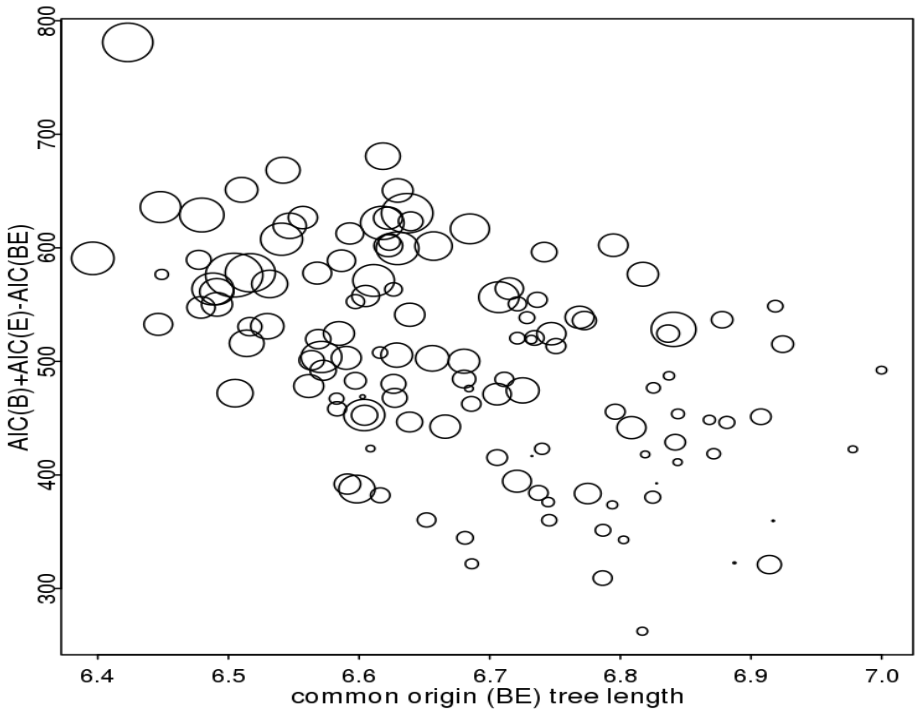
We replicated the analyses described in subsection 3.1 of the D. Theobald’s Suppl. Material, but this time under the more realistic setting of aligning the shuffled sequences. By shuffling the columns of one (or both) of the clades it is natural that the phylogenetic signal will decay, but such a dataset would be a very unlikely candidate for phylogenetic analysis given its very low similarity. By “shuffling” we understand interchanging at random the columns of all sequences belongin to one group -- such that the alignment is preserved within the group but disrupted between groups. Any phylogenetic analysis in such cases should start by optimizing the alignment for the sequences.

We reshuffled the Eukarya dataset 128 times, each time realigning it (optimizing the alignment) against the Bacteria dataset, and comparing the resulting phylogeny with those if we separate again the two groups. We always compare the datsets after shuffling **and** realigning since the inclusion of gaps (and even the order of computation) might interfere in the final likelihood values.

On the figure below we see a preliminary analysis using one simple model (Blosum62, no rate heterogeneity) and estimating the ML branch lenghts before and after realigning the shuffled sequences. We can see that before the realignment the ML trees have very long branches (blue lines), favoring the independent hypothesis as reported by D. Theobald. But the total tree length can be greatly decreased by simply optimizing the alignment (red lines), which raises the suspicion of a common origin being favored. We cannot, however, compare the likelihoods before and after the realignment because they represent distinct data for the phylogenetic reconstruction program.



Below we have a figure summarizing the main results of the analysis: for the 128 shuffled and realigned datasets, we have the total tree length versus the AIC, all strongly supporting the common origin. The circle size is proportiional to the realigned dataset length, which was between 7108 and 7331 (around 9% longer than the original 6591 amino acid sites alignment).



implementation directory: *test006*

The script 40.reshuffle_prottest.py will do iteractively the following steps (based on the best trees from the original datasets):

1. shuffle the columns of the Eukarya (4 sequences) alignment
2. create an dataset with 8 sequences, by piling up the 4 taxa from Bacteria and 4 from the shuffled Eukarya
3. optimize the alignment of this eight-taxa dataset with the program muscle
4. use prottest to find the best model and AIC values under the fixed tree for this realigned dataset
5. use prottest for this realigned dataset after splitting the Eukarya and Bacteria sequences

The output will be stored on a file named like tableLnL_.txt with, respectively, the ML total tree lengths of B, E, and BE, the sequence length of the realigned dataset, and the AIC defined as AIC(B)+AIC(E)-AIC(BE). Therefore positive values favor the common origin hypothesis.

extra: *test005*

The program `create_align.py` iteractively shuffles the dataset (one of two groups, Archea of Bacteria) and calculates the ML trees before and after realigning the sequences, using a Blosum62 model with no rate heterogeneity. The output contains the columns:

1. index
2. ML tree length (sum of branch lengths) before realigning
3. LnL of ML tree before realignment
4. ML tree length (sum of branch lengths) after realignment
5. LnL of ML tree after realignment
6. sequences size (number of columns) after realignment

This analysis mainly shows that realigning the reshuffled
The program `calc_original_align.py` simply finds the ML trees for the original (without reshuffling) datasets.

Simulations under H0 and H1

For us the null hypothesis (H0) is the independent origins case, while the common origin represents the alternative hypothesis (H1). We used the concatenated data set comprised of Bacteria and Eukarya kindly provided by D. Theobald, and using prottest found the best models and parameters. We did this for three data sets: only bacteria (B) comprised of four sequences, only eukarya (E) with four sequences as well and both bacteria and eukarya (BE) with eight sequences in total, where each alignment has 6591 sites. We subsequently used the optimal phylogenetic trees and branch lengths as input for the simulations of the respective data sets. Curiously the best model found was LG+I+G+F for all three alignments, while Theobald reports rtREV+I+G+F for BE and rtREV+G+F for each quartet -- but this is due to the newer version of phyml that we used (v3.0, against v2.4.5 used by him).

extra: test010
The script `10.align_and_phyml.py` will simulate datasets of length 6591 amino acid sites under the ML 8 taxa tree for Bacteria and Eukarya. Then it will calculate the Likelihood Ratio Test (LRT) between the separate quartets (datasets B and E) and the 8 taxa together (dataset BE), before and after realigning. The ln(LRT) = LnL(B) + LnL(E) - LnL(BE), and the model used in the simulations and in the inference is WAG+G+F. Not surprisingly the common origin was (correctly) strongly favored, and the realigned datasets were very similar to the non-aligned ones.

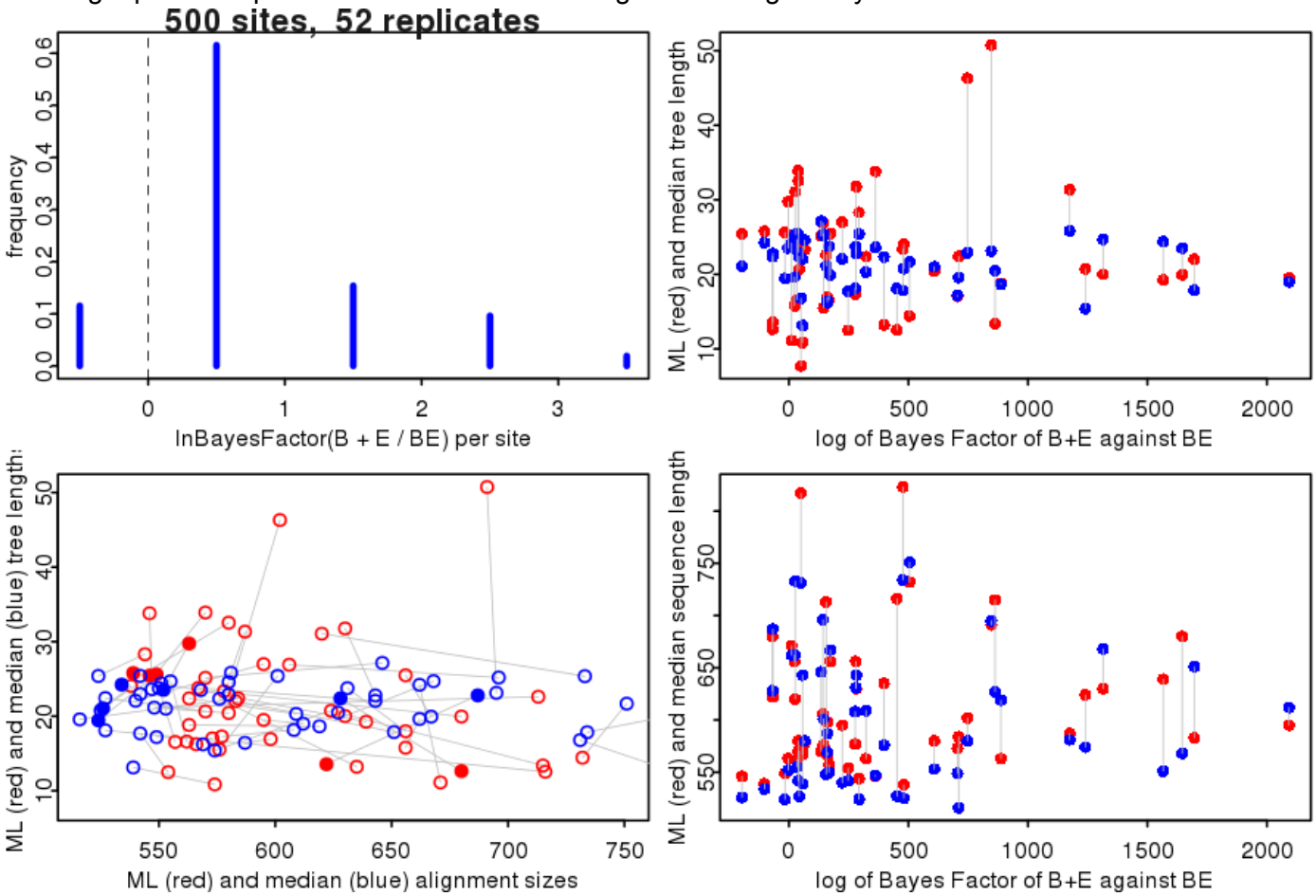
(...)

Using bali-phy under H0

We simulated independent data sets under the bacterial and eukarian phylogenies of 500 sites each, under the model LG+G+F but with slightly different gamma and frequency parameters for each (e.g. shape parameter of 1.43 for bacterial and 1.21 for eukaryan simulations). One replicate is then composed of three data sets, as before: each quartet (B and E) independently and a eight-taxa data set produced by joining both simulated quartets into a single one.

We then use bali-phy to estimate the posterior distribution of alignments, branch lengths and shape parameter assuming the LG+G model under a fixed topology (but variable branch lengths). For each data set we sample the parameters 500 times over a run of 10^5 MCMC iterations, and calculate the marginal likelihood *marg_lik(data)* as the harmonic mean of the sample likelihoods (in log scale). The log of the Bayes Factor for the independent origins hypothesis is then *log[marg_lik(B)] + log[marg_lik(E)] - log[marg_lik(BE)]* such that positive values favor the independent origins and negative values indicate a common origin.

On the picture below we see the results summarized over 52 replicates. The panel at the top-left shows the histogram of log Bayes factor values per site. That is, the log of the Bayes factor divided by the posterior median of the alignment length for BE data set (the nonscaled values can be seen on both panels on the right). The panel on the botom left plots the posterior BE alignment sizes against posterior total tree length for the BE simulation. There are several ways of summarizing the posterior distribution by one value, and we have chosen the maximum likelihood estimate (ML over all MCMC samples) which is depicted in red and the posterior median estimate represented in blue. They are connected to show that they refer to the same replicate (different estimates from same posterior distribution). The closed circles represent the replicates with negative log Bayes Factors.
In the right panel we plot each of these variables against the log of Bayes Factors.



Our convergence analysis was done on a pilot data set by visual inspection of the time series and by calculating the potential scale reduction factor, but we did not systematically checked for convergence of all replicates.