Fast and accurate branch lengths estimation for phylogenomic trees

Manuel Binet, Olivier Gascuel, Celine Scornavacca, Emmanuel J.P. Douzery, Fabio Pardi

RESEARCH

Fast and accurate branch lengths estimation for phylogenomic trees

Manuel Binet[1,2,3], Olivier Gascuel[1,2], Celine Scornavacca[2,3], Emmanuel J.P. Douzery[3] and Fabio Pardi[1,2*]

*Correspondence: pardi@lirmm.fr
[1]Laboratoire d'Informatique de Robotique et de Microélectronique de Montpellier (LIRMM), CNRS, Université de Montpellier, France
Full list of author information is available at the end of the article

Abstract

Background: Branch lengths are an important attribute of phylogenetic trees, providing essential information for many studies in evolutionary biology. Yet, part of the current methodology to reconstruct a phylogeny from genomic information — namely supertree methods — focuses on the topology or structure of the phylogenetic tree, rather than the evolutionary divergences associated to it. Moreover, accurate methods to estimate branch lengths — typically based on probabilistic analysis of a concatenated alignment — are limited by large demands in memory and computing time, and may become impractical when the data sets are too large.

Results: Here, we present a novel phylogenomic distance-based method, named ERaBLE (Evolutionary Rates and Branch Length Estimation), to estimate the branch lengths of a given reference topology, and the relative evolutionary rates of the genes employed in the analysis. ERaBLE uses as input data a potentially very large collection of distance matrices, where each matrix is obtained from a different genomic region — either directly from its sequence alignment, or indirectly from a gene tree inferred from the alignment. Our experiments show that ERaBLE is very fast and fairly accurate when compared to other possible approaches for the same tasks. Specifically, it efficiently and accurately deals with large data sets, such as the OrthoMaM v8 database, composed of 6,953 exons from up to 40 mammals.

Conclusions: ERaBLE may be used as a complement to supertree methods — or it may provide an efficient alternative to maximum likelihood analysis of concatenated alignments — to estimate branch lengths from phylogenomic data sets.

Keywords: phylogenomics; supertree; branch lengths; gene rates; distance-based; least-squares

Background

With the continuous growth of genome sequencing capabilities, phylogenetic inference is increasingly based on large collections of genomic regions, if not entire genomes [1–3]. We have entered the era of phylogenomics — the study of evolution at a genomic scale.

New methodological challenges arise in this field. Clearly, the large amount of data — sequences from several taxa and large collections of genes — makes computational efficiency essential. Besides quantity, the nature of the data is also a concern, and it is extremely important to correctly account for the distinctive features of a typical phylogenomic data set: for example the heterogeneity in the evolution of genomic

regions [4–9], and the fact that each region is typically sequenced in a subset of the taxa under analysis, with only partial overlap between different subsets [10, 11].

In this paper, we focus on the problem of how to efficiently and accurately estimate the branch lengths of a tree in a phylogenomic context, a problem for which, to date, only computationally-intensive techniques appear to be available. Yet, evolutionary distance information is essential to answer several important biological questions, from molecular dating [12, 13] of events such as speciations, to the reconciliation of gene trees with a species tree [14], or to the measure of biodiversity in conservation biology [15]. Another goal here is the efficient estimation of the relative rates of evolution of different genomic regions. This information – strictly linked to branch lengths – is also very useful, for example to recognize the diverse selective pressures acting on different parts of the genome [16, 17]. Tree inferences in a phylogenomic context fall roughly into three frameworks: the supertree, the superalignment and the medium-level framework. We consider them in relation to our goals of branch length and gene rate estimation.

Supertree approaches [18, 19] combine the information from several phylogenetic trees into a larger phylogeny. A strength of these methods is that the source trees can come from different types of data, such as DNA or protein sequences, or even morphological data. In a phylogenomic context, each source phylogeny is inferred from a different locus, with standard methods such as maximum likelihood, maximum parsimony or distance-based approaches. Within this category, MRP (Matrix Representation with Parsimony) [20, 21] and its derived methods (e.g., SuperFine [22]) are to date the most widely used approaches. In its standard form, MRP does not use branch length information in the source trees (if present), a limitation that is shared by most supertree methods — with very few exceptions, such as BWD (Build with Distances) [23], ACS (Average Consensus Supertree) [24] and SDM (Super Distance Matrix) [25]. As a consequence, virtually all supertree approaches are unable to provide meaningful estimates for branch lengths (MRP may provide branch weights, but these should be interpreted as a measure of evidence, not evolutionary change), or any estimate at all for gene rates.

Superalignment methods are the other classical approach for phylogenomic tree inference. They concatenate all available genomic sequence alignments into a unique alignment (often called a *character supermatrix*), which is then analyzed with standard or specially-tailored phylogenetic reconstruction algorithms [26]. These methods — whose accuracy relies on the use of state-of-the-art statistical inference techniques (typically maximum likelihood or Bayesian methods) — naturally model branch lengths and across-site rate heterogeneity. However they are computationally demanding, and may become impractical if computing time or memory are limited, or when the concatenated alignment is very large. Moreover, heterogeneity in the evolutionary processes at different genomic regions — which is readily handled in a supertree context — may require the use of models such as partition models [8, 9] or mixture models [6, 7]. These models, however, further increase the number of parameters to estimate, and consequently computational costs.

Lastly, the *medium-level* [25, 27, 28] framework combines the information originating from the different loci at a level that is intermediate between sequence alignments and complete gene trees. For example, this intermediate level may consist of partial trees — such as quartets [29, 30] — or pairwise distances between gene

sequences [24, 25]. Specifically, distance-based methods naturally account for and can estimate branch lengths, and in some cases they can even estimate gene rates [5, 25]. Moreover, they are relatively light computationally. The method we present here, ERaBLE (Evolutionary Rates and Branch Length Estimation), falls within this category. Unlike other phylogenomic methods, however, its goal is not tree estimation, and ERaBLE should be used to complement existing approaches that do not estimate branch lengths and/or gene rates. Alternatively, it can be used on its own when the evolutionary relationships among the species under consideration are largely known.

Note that distance-based phylogenomic methods such as ERaBLE, ACS [24] and SDM [25] can be used both in the medium-level framework – when the input distances are directly estimated from genomic alignments – but also in the supertree framework – when the input distances require prior inference of a collection of gene trees. We will see examples of this in our experiments (Results and discussion section).

The methodology we propose here can be seen as a generalization of classical weighted least squares (WLS) branch length estimation, to the case where multiple distance matrices are estimated from different genomic regions. In fact if only one matrix is given, ERaBLE behaves exactly like WLS. WLS fits the branch lengths of a tree T so as to make the distances between its leaves as close as possible to the input distances. Formally, it minimises the criterion $\sum_{i<j} w_{ij}(\delta_{ij} - d_{ij}^T)^2$, where the δ_{ij} denote the input distances, the d_{ij}^T are the distances between the leaves of T (determined by the lengths assigned to its branches), and the weights $w_{ij} > 0$ express the confidence in the estimate δ_{ij}. When multiple distance matrices are provided, we face the problem that, due to rate heterogeneity among the alignments, their distances cannot readily be compared to d_{ij}^T. ERaBLE thus applies a rescaling of the input distances, in order to use them for branch length estimation. Compared to WLS, this entails surprisingly little computational overhead.

In the following, we first describe our new method and the data sets on which we compared its performance to that of other possible approaches for the same task (Methods section). Then, we present the results of our experiments on these data sets (Results and discussion section).

Methods

In this paper, we assume that the analysis focuses on a collection of orthologous genomic regions, or genes, $G_1, G_2, \ldots, G_m$, whose evolution mostly differs because of rate heterogeneity. In other words, the trees describing their evolution are topologically compatible [31]. This is an optimal scenario for the methods we describe here, but it does not preclude their application to real-world datasets where this assumption will be necessarily violated to some degree. Gene tree topological incompatibilities may in fact arise due to incomplete lineage sorting [32, 33], gene duplication and loss [34], or even lateral gene transfer (see [35] for an excellent review of these phenomena). An even stronger assumption, which is useful to clarify the meaning of branch lengths and rates at a genomic level, is that of the *proportional model* [4, 36], which we describe further below.

Defining phylogenomic branch lengths

The length of branch e in the gene tree for G_k, denoted here $b_e^{(k)}$, generally represents the average (or expected) number of substitutions per site, occurred between the endpoints of e. If we let x and y denote these endpoints, we can rewrite this as:

$$b_e^{(k)} = \frac{s_{xy}^{(k)}}{N_k}, \tag{1}$$

where $s_{xy}^{(k)}$ is the (expected) number of substitutions in G_k occurred between x and y, and N_k is the sequence length of gene G_k.

We wish to give the same meaning to the branch lengths of the phylogenomic (or species) tree representing the evolution of genes $G_1, G_2, \ldots, G_m$. If we define the length of branch e in this tree as the average (or expected) number of substitutions per site between its endpoints x and y, we then have:

$$b_e = \frac{\sum_{k=1}^{m} s_{xy}^{(k)}}{\sum_{k=1}^{m} N_k}. \tag{2}$$

This definition determines the relationship between the branch lengths in the species tree and those in the gene trees. If we let $N = \sum_{k=1}^{m} N_k$, and use equation (1), then equation (2) can be rewritten as:

$$b_e = \frac{1}{N} \sum_{k=1}^{m} N_k b_e^{(k)} \tag{3}$$

In other words, branch length b_e in the species tree is equal to an average of the corresponding branch lengths $b_e^{(k)}$ in the gene trees, weighted by the lengths of the gene sequences.

Note that in this paper we assume that genes are sampled in different, partially overlapping sets of taxa, meaning that a branch in a gene tree will in general correspond to a path in the species tree. Thus, in equation (3), and in the equations that follow, it is more accurate to interpret b_e and $b_e^{(k)}$ as lengths of paths connecting the same nodes across all trees, depending on the taxa sampled for each gene.

The proportional model

In order to provide a stronger link among branch lengths in gene trees and in the species tree, and to set a meaningful scale for the gene rate estimates, we now introduce the *proportional model* [4, 36], an implicit assumption of many phylogenomics methods [4, 5, 36], including ours. This model assumes that each gene G_k induces the same tree up to a multiplicative constant for branch lengths, r_k, representing its evolutionary rate (and up to removal of branches leading to taxa for which G_k is not sampled). In other words, if we let $b_e^{(k)}$ denote the length of a branch e (or a path, see above) in the gene tree for G_k, then

$$\frac{b_e^{(k)}}{r_k} \text{ is constant for all } k = 1, \ldots, m. \tag{4}$$

This model is a rough approximation of biological reality, as typically the relative values of the gene rates $r_1, r_2, \ldots, r_m$ may vary over time — a phenomenon known as heterotachy [37]. Nevertheless, this simple model greatly restricts the number of parameters to estimate and leads to robust analyses.

The proportional model, as specified by equation (4), defines *relative* rates, that is, it determines r_k up to a multiplicative constant. Here, we take r_k as the rate of G_k, relative to the "phylogenomic rate", that is, we require $r_k = b_e^{(k)}/b_e$. Equation (3) then implies that the weighted average of gene rates must be 1. In fact, by substituting $b_e^{(k)}$ with $r_k b_e$ into equation (3), and dividing both sides by b_e, we obtain:

$$\frac{1}{N}\sum_{k=1}^{m} N_k r_k = 1. \tag{5}$$

We will use this equation later on, to set a meaningful scale for the gene rates output by our method (and others). The same rescaling will be applied to the returned branch lengths, as they are strictly linked to the rates.

The ERaBLE method

The new method presented here, ERaBLE (*Evolutionary Rates and Branch Length Estimation*), simultaneously estimates gene rates and the branch lengths of a phylogenomic tree of given topology, using a collection of distance matrices — one distance matrix per gene G_k. As we illustrate in our experiments (*Results* section), these distance matrices can either be directly estimated from pairwise alignments of the gene sequences, or they can be calculated from gene trees inferred for each G_k. A C++ implementation of ERaBLE is available on the web at http://www.atgc-montpellier.fr/erable/.

Let L_k designate the set of taxa for which the sequence of G_k is available. For $i, j \in L_k$, let then $\delta_{ij}^{(k)}$ denote the input distance for gene G_k between taxa i and j. Given a tree topology $\mathcal{T}$ with leaves labelled by the taxa in $L = \bigcup_{k=1}^{m} L_k$, the goal is to estimate the branch lengths of $\mathcal{T}$ and the evolutionary rates of the m genes under consideration. $\mathcal{T}$ can either reflect a well-known phylogeny for the taxa in L, or it can be inferred prior to ERaBLE's execution, for example using MRP or other supertree methods. We do not make any assumption on the degree of overlap between the taxon sets L_k. Extremely sparse data sets may not determine a unique optimal solution to our estimation problem, but this does not prevent the application of ERaBLE.

Now let $\hat{b}_e$ denote the estimated length for branch e. This determines the *additive distance* $\widehat{d}_{ij}$ between any two taxa i and j, simply defined as the sum of the $\hat{b}_e$ for all e in the path between i and j in $\mathcal{T}$. For mathematical convenience, we choose to estimate the inverses of gene rates: we refer to $\hat{\alpha}_k$, the estimate for $1/r_k$, as the *scale factor* of gene G_k. ERaBLE thus seeks the values of $\hat{b}_e$, for all branches in $\mathcal{T}$, and of $\hat{\alpha}_k$, for $k = 1, 2, \ldots, m$, that solve the following optimization problem:

$$\begin{cases} \text{minimize} & Q(\hat{\alpha}, \hat{b}) = \displaystyle\sum_{k=1}^{m} \sum_{\{i,j\} \subset L_k} w_{ij}^{(k)} (\hat{\alpha}_k \delta_{ij}^{(k)} - \widehat{d}_{ij})^2, \\ \text{subject to} & \displaystyle\sum_{k=1}^{m} Z_k \hat{\alpha}_k = \sum_{k=1}^{m} Z_k. \end{cases} \tag{6}$$

ERaBLE can efficiently solve this problem for any choice of positive values for $w_{ij}^{(k)}$ and Z_k. Below, we explain the rationale behind the objective function $Q(\hat{\alpha}, \hat{b})$ and the constraint in problem (6), and provide practical choices for $w_{ij}^{(k)}$ and Z_k. Then, we briefly describe the algorithm that allows ERaBLE to efficiently solve problem (6). Details are provided in Additional file 1. Lastly, we show how to rescale the optimal values for $\hat{b}_e$ and $\hat{\alpha}_k$, so that they comply with their definitions in equations (3) and (5).

The objective function. As predicted by the proportional model, we would like the distances in the phylogenomic tree to be approximately equal to the gene-specific distances, up to the multiplicative factor r_k. Thus, we would like to set the $\hat{b}_e$ and $\hat{\alpha}_k$, so that:

$$\hat{d}_{ij} \approx \frac{\delta_{ij}^{(k)}}{r_k} \approx \hat{\alpha}_k \delta_{ij}^{(k)} \qquad \text{for all } k \in \{1, 2, \ldots, m\} \text{ and } i, j \in L_k.$$

The optimisation criterion $Q(\hat{\alpha}, \hat{b})$ provides a score for the discrepancy between the $\hat{d}_{ij}$ and the scaled distances $\hat{\alpha}_k \delta_{ij}^{(k)}$. It is a WLS criterion, where $w_{ij}^{(k)}$ is a strictly positive weight indicating the confidence given to the distance estimate $\delta_{ij}^{(k)}$, and which ideally is inversely proportional to its variance. In our experiments, we have chosen the simple approach of setting $w_{ij}^{(k)} = N_k$ (i.e., the length of the alignment for gene G_k), but ERaBLE is capable of using more sophisticated weightings (e.g., [5, 38]).

WLS is a special case of GLS, a class of criteria that account for the covariances between the $\delta_{ij}^{(k)}$. However, GLS criteria are rarely used for phylogenetic inference, because of the computational complexity of optimizing them, and because of the difficulty of evaluating the covariances. WLS is a good compromise, and it is notably used in the well-known algorithm of Fitch and Margoliash [39] and in FastME [40].

Criterion $Q(\hat{\alpha}, \hat{b})$ is similar to those by Bevan et al. [5] and Criscuolo et al. [25]. The optimisation problems in these papers, however, seek optimal values for $\hat{d}_{ij}$ directly, without assuming any relationship between these distances and a tree (namely without assuming additivity). ERaBLE, instead, assumes a particular topology $\mathcal{T}$, and constrains the distances $\hat{d}_{ij}$ to be additive with respect to $\mathcal{T}$, meaning that its problem unknowns are the branch lengths in $\mathcal{T}$.

The constraint. $Q(\hat{\alpha}, \hat{b})$ is trivially minimized by setting all $\hat{\alpha}_k = 0$, and all $\hat{b}_e = 0$. In order to obtain more meaningful solutions, while ensuring mathematical tractability, we adopt a linear constraint over the $\hat{\alpha}_k$: the constraint in (6) is in fact the most general form for such a linear constraint. In Additional file 1, we show that the right-hand side in this constraint is irrelevant to the end results, as it only determines their scale, which is subsequently reset by the step described in *Rescaling the outputs* below.

As to the choice for Z_k, the two simplest approaches are to set $Z_k = 1$ [4, 25] or $Z_k = N_k$. The latter results in a constraint that is similar in spirit to equation (5) above, as it constrains more strongly the rates (or more precisely their inverses) of long genes. However, our experiments have shown that both these approaches can incur in significant over-estimation of the scale factors $\hat{\alpha}_k$ for genes appearing in a small subset L_k of closely related taxa. In Additional file 2, we show a small example

where the reasons for this are evident. In order to deal with this problem, we have chosen to set $Z_k = N_k \sum_{i,j \in L_k} \delta_{ij}^{(k)}$ in all the experiments below, an approach that at the same time puts a stronger constraint on the scale factors of long genes – like (5) above – and that we have experimentally verified to largely fix the over-estimation problem for the $\hat{\alpha}_k$.

Solving the problem. The one in (6) is a classic quadratic programming problem, which can be solved using Lagrange multipliers [41]. As we show in Additional file 1, this yields a system of $\mathcal{O}(n+m)$ linear equations in $\mathcal{O}(n+m)$ unknowns (all the $\hat{b}_e$ and the $\hat{\alpha}_k$), where n is the number of taxa in L, and m is the number of genes. Calculating naïvely the coefficients of this system and solving it would require $\mathcal{O}(mn^4 + (n+m)^3)$ time and $\mathcal{O}((n+m)^2)$ auxiliary memory (i.e., not including the memory to store the input), but careful adaptation of techniques for WLS branch length calculation [5, 42, 43] leads to a reduction of the algorithm's complexity to $\mathcal{O}(mn^2 + n^3)$ time and $\mathcal{O}(mn + n^2)$ auxiliary memory. In Additional file 1, we describe this algorithm in detail.

Given that problem (6) can be seen as a generalization, for several distance matrices, of standard WLS branch length estimation, it is interesting to note that, for $m = \mathcal{O}(n)$, their computational complexities coincide — as standard WLS requires $\mathcal{O}(n^3)$ time and $\mathcal{O}(n^2)$ memory [42]. If instead $m >> n$, which is the most common scenario in phylogenomics, an attractive aspect of ERaBLE is that its complexity grows linearly in m, which makes it particularly suited to analyze phylogenomic data sets from large collections of genes (typically several thousands) sampled across a moderate number of taxa (few hundreds at most). This is indeed the scenario that we have tested in the experiments in the Results and discussion section, where m varies from 500 to about 7,000 and $n = 40$.

Finally, we remark that for some data sets the optimal solution of problem (6) may not be unique. This can happen when some pairs of taxa do not co-occur in any input distance matrix (note that this is a necessary but not sufficient condition for multiplicity of solutions). All such cases are recognized by ERaBLE, and the user is notified of the existence of multiple alternative solutions beyond the one returned.

Rescaling the outputs. Equation (5) shows that, as a consequence of their definition, the gene rates should have a weighted average of 1. We thus require that the estimated rates also satisfy this property, meaning that we need to rescale the $\hat{\alpha}_k$ so that the inverses of the new scale factors satisfy equation (5). In other words, we multiply the $\hat{\alpha}_k$ obtained by solving problem (6) by a *correction factor* c such that

$$\frac{1}{N} \sum_{k=1}^{m} \frac{N_k}{c \cdot \hat{\alpha}_k} = 1$$

By solving this equation for c, we obtain:

$$c = \frac{1}{N} \sum_{k=1}^{m} \frac{N_k}{\hat{\alpha}_k} \tag{7}$$

Moreover, note that in order for $\hat{d}_{ij} \approx c \cdot \hat{\alpha}_k \delta_{ij}^{(k)}$ to still hold, the same rescaling by c must be applied to the estimated branch lengths. In conclusion, ERaBLE returns:

$$\frac{1}{c \cdot \hat{\alpha}_k} \quad \text{and} \quad c \cdot \hat{b}_e$$

as estimates of r_k and b_e — the rate of gene G_k and the phylogenomic length of branch e, respectively.

Other phylogenomic distance-based methods

In our experiments, we have compared ERaBLE to a number of other approaches that bioinformaticians and evolutionary biologists may adopt in order to estimate gene rates and the branch lengths of a species tree in a phylogenomic context. These approaches are implemented as analysis pipelines, and described in detail in the Results and discussion section. While some of these pipelines implement standard techniques such as maximum-likelihood or distance-based analysis of a concatenated alignment, most pipelines are based on two phylogenomic distance-based methods that we now describe.

SDM (Super Distance Matrix) [25] has the objective to construct a distance matrix summarizing the topological signal in a collection of gene-specific distance matrices. This "average" matrix can then be used to infer a phylogenomic tree, using distance-based methods based on a single matrix. SDM applies two transformations to the input matrices — it multiplies each of them by a scale factor, and adds a scalar to each column and row (thus extending or shrinking external branches in the underlying gene tree) — with the goal of bringing them as close as possible to each other. The matrices thus obtained are then averaged to obtain a matrix that can then be analyzed with other distance-based methods. Our experiments use SDM*, a variant of SDM that only applies the scale factor transformation to the input matrices, which avoids altering the ratio between the lengths of internal and external branches in the reconstructed tree. We note that the implementation of SDM* includes a preprocessing step that corrects the input matrices to make them satisfy the triangle inequality. Since this step, as expected, affected negatively the estimation of branch lengths (but helps that of the tree topology), we removed it from the original code. In our experiments, the average matrix produced by SDM* is used to estimate the branch lengths of a fixed topology $\mathcal{T}$ using standard OLS, and gene rate estimates are obtained by taking the inverses of the scale factors returned by SDM*. Average distances and scale factors are rescaled as described for ERaBLE, that is, multiplied by the correction factor c in equation (7) above.

DistR [5] was conceived to estimate gene rates from a collection of distance matrices, and from the alignments used to calculate the distances. DistR uses the alignments to approximate the variances of the input distances, with the classical formulae by Bulmer [38]. These variances are then used in a distance-based optimization problem akin to that solved by SDM* — the main difference being the constraint on the scale of the results. DistR returns estimates for the gene rates, and, as a byproduct, a distance matrix that we use to estimate the branch lengths of a fixed topology $\mathcal{T}$ using standard OLS, as done for SDM*. No rescaling of the outputs was conducted for DistR, as it automatically produces rates and distances at a meaningful scale.

Data sets

In this section, we describe the data sets that we have used in our experiments to evaluate the performance of ERaBLE and competing methods. The first data set consists of 500 simulated replicates: for each replicate, we take a random tree over 40 taxa, and for each tree we simulate sequence data for 500 genes, which are only present in a random subset of taxa, and evolve at different rates. The second data set consists of the 6,953 exon alignments for 40 mammals in OrthoMaM v8 [44]. Detailed descriptions follow.

Simulated data. Each of the 500 replicates is obtained as follows.

- *Gene trees.* A tree T^0 is taken randomly (without replacement) from the 5,000 trees on $n = 40$ taxa in the original test data set for PhyML [45]. This tree is then rescaled to a total branch length of 1, by dividing all branch lengths by their sum. Call the resulting tree T^1. We then construct $m = 500$ gene trees $T_1, \ldots, T_m$ by multiplying the lengths of all the branches in T^1 by factors $t_1, \ldots, t_m$ randomly drawn from a continuous uniform distribution on the interval $[0.4, 9]$. This interval gives biologically realistic branch lengths [45].
- *Sequence generation.* For each gene tree T_k, we generate a DNA alignment consisting of $n = 40$ sequences of length N_k, where N_k is an integer drawn uniformly from the interval $[200, 600]$. We chose relatively short sequences to avoid making the simulated data sets too informative, so as to be able to discriminate among the estimation accuracies of the methods tested. Each alignment is generated with Seq-Gen [46], using T_k and the model K2P+Γ, with ratio between transition and transversion rates $R = 2$ (equivalent to $\kappa = 4$ [47, Sec. 1.2.4]) and with a continuous gamma distribution with shape parameter 1, to model rate heterogeneity across sites.
- *Missing data.* To simulate the partial overlap in the gene presence/absence patterns typical of real data sets, for each alignment we randomly remove a number of sequences. More precisely, for each of the m alignments generated in the previous step, we draw a parameter p uniformly between 0 and 1, and then we suppress each sequence with probability p. If the number of remaining sequences in L_k is less than 4, then we leave 4 sequences chosen randomly out of the 40, so as to guarantee a minimum amount of data to estimate the rate for that gene.
- *Model tree definition.* We call the tree that we wish to reconstruct the "model tree", and we denote it by T. Clearly, T must be the same as T^0 and T^1, up to their scale, and up to the removal of the taxa missing from all the simulated alignments. In order to define the correct scale of the model tree, we define tree T^2, with the same topology as T^0 and T^1, and branch lengths defined by $b_e = \frac{1}{N} \sum_{k=1}^{m} N_k b_e^{(k)}$, where $b_e^{(k)}$ denotes the length of e in T_k, and $N = \sum_{k=1}^{m} N_k$. Note that this is the same as equation (3), whose justification is amply given above. Finally, we obtain the model tree T by taking the restriction of T^2 on the set of taxa $L = \bigcup_{k=1}^{m} L_k$.
- *Model rates definition.* Similarly to the model tree, the "model gene rates" must be the same as $t_1, \ldots, t_m$ up to their scale. The absolute values of $t_1, \ldots, t_m$ are in fact unrecoverable from the data. By imposing equation (5)

to the rescaled rates, we must have:

$$r_k = \frac{t_k}{F}, \quad \text{where } F = \frac{1}{N}\sum_{k=1}^{m} N_k t_k.$$

OrthoMaM data set. OrthoMaM (v8) [44] consists of a collection of single-copy orthologous phylogenetic markers, selected among the genomes of the 40 mammals in the Ensembl v73 database [48]. We downloaded the entire set of the 6,953 nucleotide exon alignments in OrthoMaM v8, filtered with trimAl [49]. Alignment lengths N_k range from 231 to 17,103 (median: 702), and each alignment contains a variable subset L_k of taxa, with $4 \leq |L_k| \leq 40$ (median: 27).

Results and discussion

In order to compare the performance of ERaBLE to that of other approaches, we have conducted a number of experiments on the data sets described in the Methods section. For each of the 500 simulated replicates and for the OrthoMaM data set, we compare the branch length and gene rate estimates obtained by a number of competing approaches, including ERaBLE. For the OrthoMaM data set (6,953 genes), which is an order of magnitude larger than the simulated replicates (500 genes), we also compare their running times and memory usage.

Since, to the best of our knowledge, no tool is readily available for the simultaneous estimation of branch lengths and gene rates in a phylogenomic context, for our comparisons we have assembled a number of pipelines from existing methods. Besides ERaBLE, these methods include SDM* [25] and DistR [5], which however were conceived for other tasks than ours. We refer to the Methods for a brief description of how we adapted these tools to our goals. We describe the pipelines below.

Analysis protocol

The OrthoMaM data set and each replicate in the simulated data set have the same structure: they consist of m gene alignments $A_1, A_2, \ldots, A_m$ over the taxon sets $L_1, L_2, \ldots, L_m$ ($m = 500$ for the simulated data sets, $m = 6,953$ for OrthoMaM). In addition to these inputs, the tested methods are also provided with a reference topology $\mathcal{T}$, over the set of taxa $L = \bigcup_{k=1}^{m} L_k$, to which they aim to assign branch lengths. For simulated data, $\mathcal{T}$ is the topology of the model tree T, whereas for OrthoMaM $\mathcal{T}$ is the mammalian tree topology in Additional file 5. The outputs are a tree estimate $\hat{T}$ with topology $\mathcal{T}$, and gene rate estimates $\hat{r}_1, \ldots, \hat{r}_m$.

The tested methods are classified in the three frameworks described in the Background section: supertree, superalignment and medium-level. Note that for distance estimation, as well as for maximum likelihood (ML) tree reconstruction, we use the model TN93+Γ, as it is the most complex nucleotide substitution model for which an analytic formula for pairwise distance estimation is available. In the following, we denote by Γ_c the continuous Gamma distribution used for pairwise distance estimation, and by Γ_8 the discrete Gamma distribution based on 8 categories, which we adopt for ML tree inference. Also note that for pairwise distance estimation, the shape parameter for the Gamma distribution cannot be estimated from the data,

and thus must be set to a realistic value by the user [47] (more detail on this point below). All tested methods are depicted schematically in Fig. 1 and their names together with short descriptions can be found in Table 1. A detailed description follows.

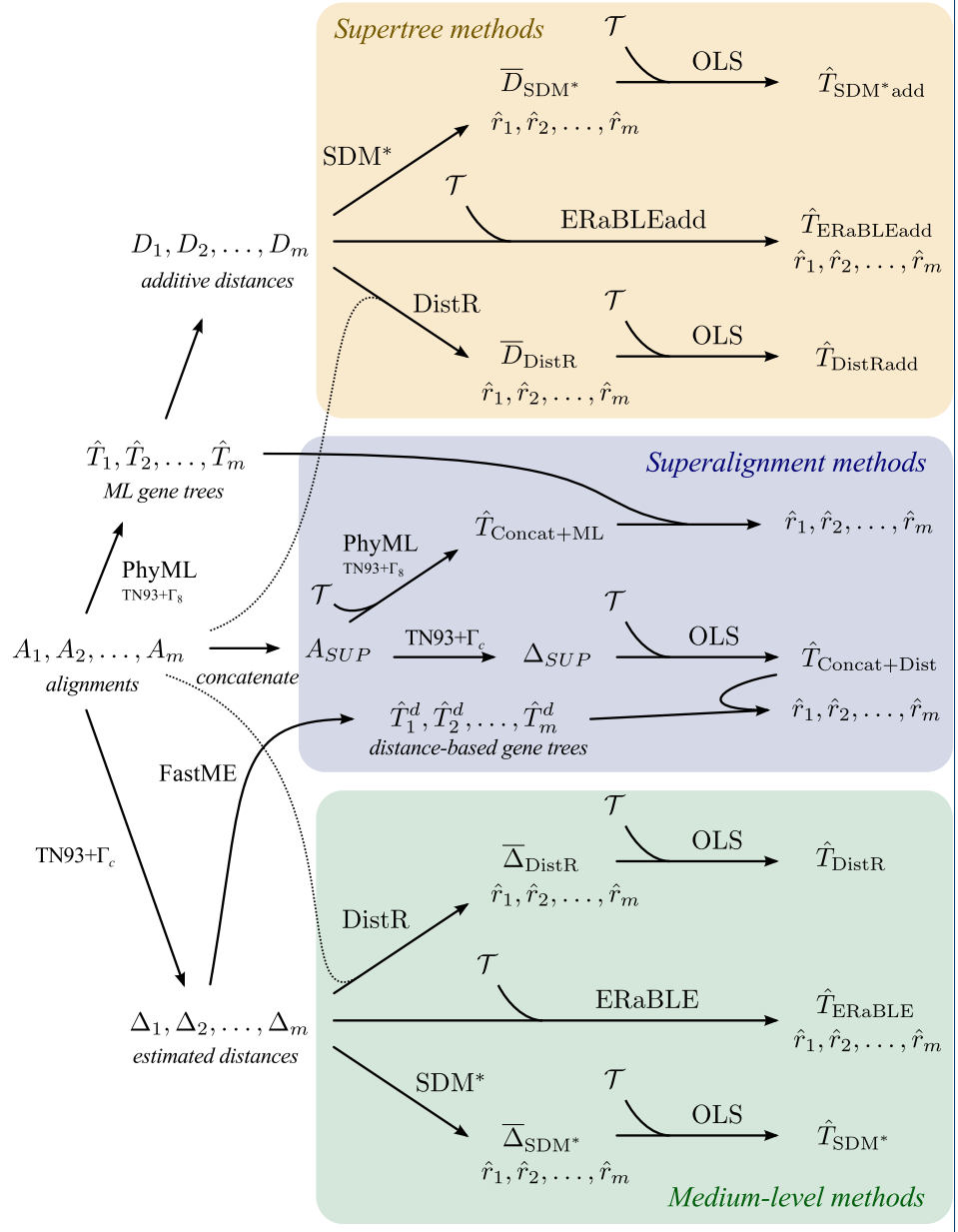

Figure 1 Pipelines of the analyses applied to both data sets, represented as flowcharts. We refer to the Analysis protocol subsection for a detailed description of each analysis method.

Supertree methods. For each alignment A_k we infer a gene tree $\hat{T}_k$ with PhyML [45, 50], using the model TN93+Γ_8. The shape parameter for the Gamma distribution is set to 1 for the simulated data sets (that is, the value used to generate the data), and left free to estimate for the OrthoMaM data set. Unless otherwise stated, in the following experiments PhyML is free to estimate the topology of $\hat{T}_k$, which

is realistic when gene trees are inferred as part of a separate analysis, for example to provide the input for supertree topology reconstruction. In other experiments, we have constrained PhyML to reconstruct gene trees of topology agreeing with $\mathcal{T}$, an approach that significantly reduces running times. (More precisely, the topology of $\hat{T}_k$ is constrained to be the restriction of $\mathcal{T}$ to L_k.) This is the correct way to proceed when the only goal is the estimation of branch lengths in a reference tree. We will come back on this second approach when comparing the computational efficiencies of the methods tested.

Standard supertree methods, such as MRP [20, 21], would then only consider the topologies of the inferred gene trees $\hat{T}_1, \hat{T}_2, \ldots, \hat{T}_m$, but this makes it impossible to estimate branch lengths for the phylogenomic tree. In order to conserve branch length information, we construct the additive distance matrices $D_1, D_2, \ldots, D_m$ corresponding to these gene trees — that is, the distance between taxa i and j in D_k equals the sum of the lengths of the branches between i and j in $\hat{T}_k$. Note that, as additive distances uniquely determine a tree [51], D_k can just be interpreted as a different representation for $\hat{T}_k$. We test three methods based on these additive matrices (hence "add" in their names).

- *SDM*add.* We run SDM* on $D_1, D_2, \ldots, D_m$, with D_k weighted by the alignment length N_k. The average matrix and scale factors thus obtained are then multiplied by the scaling factor c in equation (7), thus giving a scaled average matrix $\overline{D}_{\mathrm{SDM}^*}$, and gene rate estimates (the inverses of the resulting scale factors). Finally, on the basis of $\overline{D}_{\mathrm{SDM}^*}$ we assign OLS branch lengths to the reference topology $\mathcal{T}$, using FastME [40].
- *DistRadd.* We run DistR on $D_1, D_2, \ldots, D_m$ (and $A_1, A_2, \ldots, A_m$), thus obtaining gene rate estimates and an average matrix $\overline{D}_{\mathrm{DistR}}$. The latter is then used to assign OLS branch lengths to $\mathcal{T}$, with FastME.
- *ERaBLEadd.* We run ERaBLE on $D_1, D_2, \ldots, D_m$ and $\mathcal{T}$, with the weighings for $w_{ij}^{(k)}$ and Z_k described in the Methods section. ERaBLE directly provides gene rate estimates and branch length estimates for $\mathcal{T}$.

Note that it is problematic to evaluate the variances of the distances computed by SDM* and DistR (those in $\overline{D}_{\mathrm{SDM}^*}$ and $\overline{D}_{\mathrm{DistR}}$, respectively). This is why we used OLS branch length estimation for the last step in SDM*add and DistRadd.

Medium-level methods. From each alignment A_k, we estimate a distance matrix Δ_k, using FastME [40] with the model TN93+Γ_c. Note that estimation of the shape parameter for the Gamma distribution would require joint comparison of multiple sequences [47], but here we only use pairwise comparisons. Thus, we set the shape parameter to 1 for the simulated data sets (that is, the value used to generate the data), and to 0.5 for the OrthoMaM data set, as we consider this as a realistic estimate for mammals. (E.g., the median shape parameter estimated by PhyML when inferring the OrthoMaM gene trees is 0.493.) We test three methods identical to those described above for supertree methods, except that they use the estimated matrices $\Delta_1, \Delta_2, \ldots, \Delta_m$ instead of the additive matrices deriving from the ML gene trees. We call these methods *SDM**, *DistR* and *ERaBLE*. (See again Fig. 1.)

Superalignment methods. Let A_{SUP} denote the alignment obtained by concatenating $A_1, \ldots, A_m$. We test two methods based on A_{SUP}.

- *Concat+ML*. We assign branch lengths to the reference topology $\mathcal{T}$ by running topology-constrained PhyML on A_{SUP}, with the model TN93+Γ_8. We call the resulting tree $\hat{T}_{Concat+ML}$. Here the shape parameter for the Gamma distribution is left free to estimate. In fact, even though for each gene alignment A_k taken separately we may set this parameter to 1 for the simulated data, or to 0.5 for OrthoMaM, these values cannot be used on the concatenation A_{SUP}. This is because the alignments $A_1, A_2, \ldots, A_m$ derive from trees at different scales, meaning that rate variation in A_{SUP} will be larger than that on a single A_k, and the shape parameters smaller (PhyML estimates 0.487 for OrthoMaM, and 0.7 on average for the simulated data). As to gene rate estimates, $\hat{r}_k$ is then obtained as the ratio between the total length of the ML gene tree $\hat{T}_k$ (a source tree for supertree methods) and the total length of the tree that is obtained from $\hat{T}_{Concat+ML}$ by taking its restriction to L_k. For OrthoMaM, which, unlike the simulated data set, does not have model gene rates and a model tree, we take the outputs of this method as reference. The choice of PhyML over more computationally efficient alternatives is due to its greater availability of models, which may entail better accuracy. (See also Additional file 7, where we report about the effects of using alternative ML methods in our experiments.)
- *Concat+Dist*. From A_{SUP}, we estimate a distance matrix Δ_{SUP}, using FastME with the model TN93+Γ_c. The shape parameter for the Gamma distribution is set to the value estimated above by PhyML on A_{SUP}. Then, on the basis of Δ_{SUP}, we assign OLS branch lengths to the reference topology $\mathcal{T}$, using FastME. Call the resulting tree $\hat{T}_{Concat+Dist}$. Finally, in order to estimate gene rates, we use the same procedure as that for Concat+ML, but in a distance-based context: $\hat{r}_k$ is obtained as the ratio between the total length of a distance-based gene tree $\hat{T}^d_k$ and the length of the restriction of $\hat{T}_{Concat+Dist}$ to L_k. Distance-based gene trees $\hat{T}^d_1, \ldots, \hat{T}^d_m$ are obtained from the estimated distance matrices $\Delta_1, \ldots, \Delta_m$ using FastME with default options.

Table 1 Names and short descriptions of the methods tested.

Name	Brief description
Concat+Dist	Distance-based analysis of the concatenated alignment
Concat+ML	ML analysis of the concatenated alignment
SDM*add	SDM* run on the gene tree distance matrices (+ post-processing)
DistRadd	DistR run on the gene tree distance matrices (+ post-processing)
ERaBLEadd	ERaBLE run on the gene tree distance matrices
SDM*	SDM* run on the estimated distance matrices (+ post-processing)
DistR	DistR run on the estimated distance matrices (+ post-processing)
ERaBLE	ERaBLE run on the estimated distance matrices

NOTE.— The methods are divided in three groups, corresponding to the three frameworks considered here: superalignment, supertree and medium-level.

Results and discussion for the simulated data

Given the large number of replicates, the simulated data set is especially useful to compare the estimation accuracy of the methods tested. For each method we have plotted estimation errors against the correct values of the parameters to estimate (branch lengths and gene rates), which are known for the simulated data. Figures 2

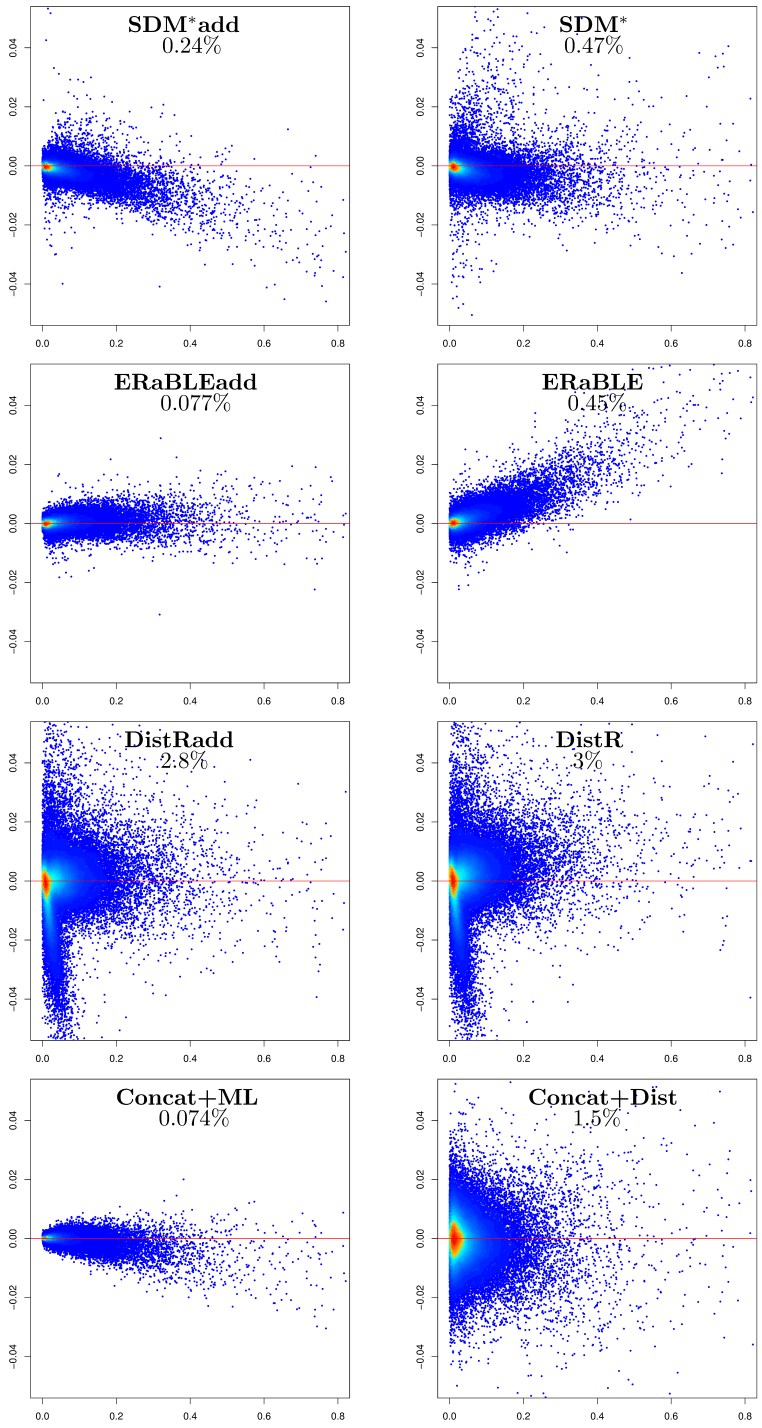

Figure 2 Accuracy of branch length estimates in the simulated data set. For each method, model branch lengths b_e (x-axis) are plotted against estimation errors $\hat{b}_e - b_e$ (y-axis) for all branches in all 500 model trees ($500 \times 77 = 38,500$ points per plot). Colors (from blue to red) indicate increased density of points. The horizontal red line corresponds to no estimation error. Method names are shown at the top of each plot, followed by the mean (over 500 values) of the fraction of variance unexplained of (b_e) relative to $(\hat{b}_e)$ (see Additional file 3).

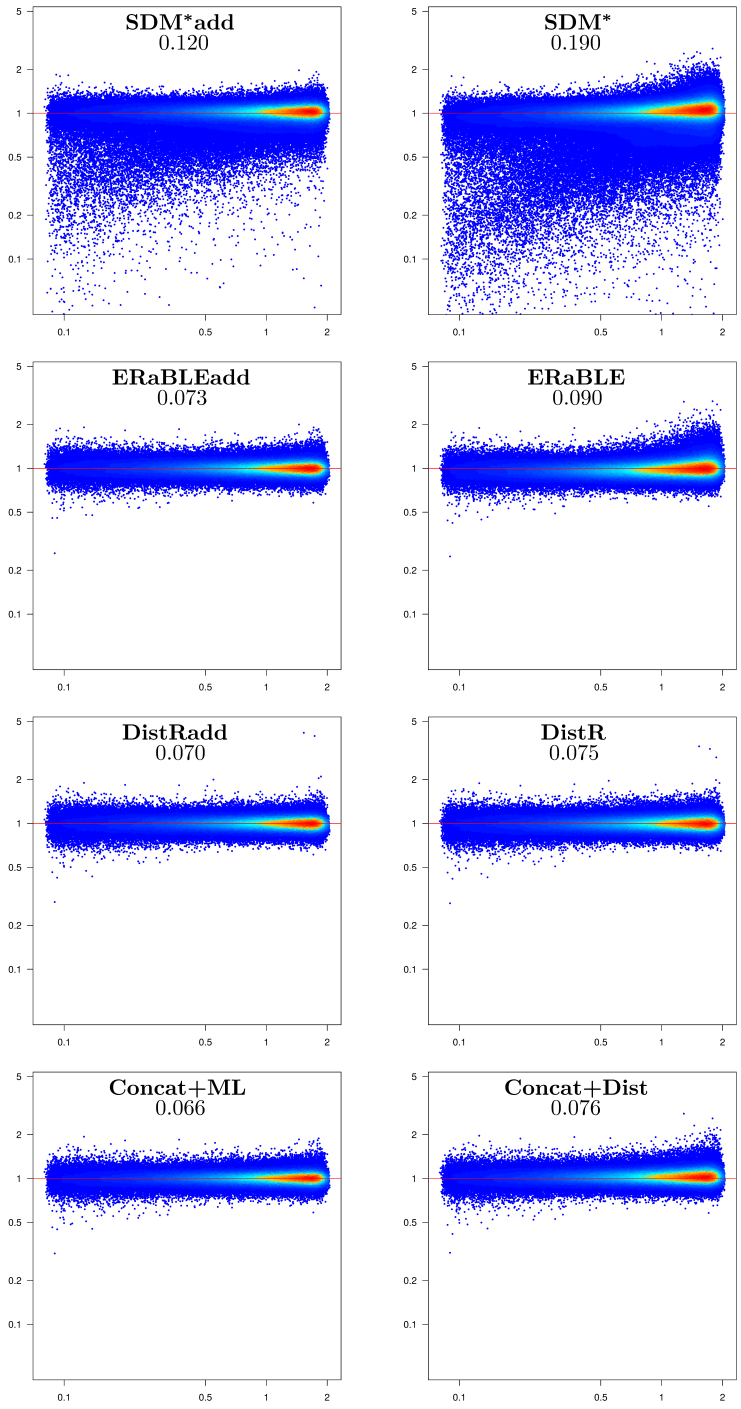

Figure 3 Estimation accuracy for gene rates in the simulated data set. Log-log scatterplots showing model gene rates r_k (x-axis) against error ratios $\hat{r}_k/r_k$ (y-axis) for all genes in all 500 replicates ($500 \times 500 = 250,000$ points per plot). Note that errors are measured with ratios, instead of differences. Colors (from blue to red) indicate increased density of points. The horizontal red line corresponds to no estimation error. Method names are shown at the top of each plot, followed by the mean absolute log-ratio between estimated and model gene rates (see Additional file 3).

and 3 show the accuracies of all tested methods in the estimation of branch lengths and gene rates, respectively. For gene rates, the scatterplots are logarithmic, as rates are inherently ratios (e.g. rates x and $1/x$, with $x > 0$, should be depicted as equally distant from rate 1).

Branch length estimation. The most accurate estimates of branch lengths are produced by Concat+ML and ERaBLEadd (see Fig. 2). Then, intermediate results are obtained by SDM*add, ERaBLE and SDM*, where some biases are observed: SDM*add seems to underestimate branch lengths, whereas ERaBLE appears to overestimate them, proportionally to the branch length. Currently, we do not have an explanation for these biases, which however are small ($\leq 5\%$) relative to the length of a branch (cf. the slope of the point cloud). Finally, DistRadd, DistR and Concat+Dist are all affected by relatively strong estimation problems for branch lengths: consider, for example, the mean fraction of variance unexplained, which for these methods is tens of times that of Concat+ML and ERaBLEadd. For DistR this is not surprising, as this method was only conceived to estimate gene rates (and not for species tree estimation) [5]. As for Concat+Dist, it is clear that the construction of a distance matrix from the superalignment entails a significant loss of information.

Gene rate estimation. With the exception of SDM-based methods (SDM* and SDM*add), all methods are approximately equally accurate in the estimation of gene rates (see Fig. 3), the best method being, as expected, Concat+ML. As apparent in the two scatterplots at the top of Fig. 3, SDM-based methods often strongly underestimate (by a factor of 2 or more) the rates of some genes. Typically these are genes that are only present in a small subset of closely related taxa. Moreover, for the other genes where this problem is not present, rate estimates tend to be slightly overestimated (see the red core of the point cloud, which lies *above* the horizontal red line). The reasons for this issue lie in the constraint used in the optimization problem solved by SDM*, which causes the same phenomena as those described in Additional file 2. The constraint used by ERaBLE avoids these issues.

Discussion. A common feature of the experiments on branch length and gene rate estimation above is that medium-level methods are generally less accurate than their supertree counterparts (compare SDM* to SDM*add, ERaBLE to ERaBLEadd, DistR to DistRadd). Again, this is not surprising, as supertree methods are based on additive distance matrices $(D_1, D_2, \ldots, D_m)$, which are expected to be more accurate estimates of the correct distances than the distances estimated directly from the alignments $(\Delta_1, \Delta_2, \ldots, \Delta_m)$. However, inferring additive distance matrices comes at a (computational) cost, as we shall show on the experiments on the OrthoMaM data set in the next section.

We conclude noting that the only methods that do not incur in any major accuracy problem on the simulated data set are ERaBLE, ERaBLEadd and Concat+ML. However, their running times and memory requirements are very different: on this data set, Concat+ML is five hundred times slower than ERaBLE (about 3h30m vs. 25s on average for a single replicate on a cluster machine with 200 GB RAM and 2.66 GHz CPU) and requires far more memory (4.2 GB vs. 70 MB). In this case the computational effort to analyse the simulated data sets is clearly not problematic for Concat+ML. This is because a simulated replicate data set is relatively small

($m = 500$). We look in more detail at running times and memory usage in the experiments in the next section, which are on a computationally more challenging data set.

Results and discussion for the OrthoMaM data set

Assessing estimation accuracy on the OrthoMaM data set is more problematic than on the simulated data set, first, because the correct values for the branch lengths and gene rates are not known and, second, because statistical noise may play an important role here, as no replicates are available. We address the former issue by adopting the estimates obtained by Concat+ML as reference values. This is justified by the observation that Concat+ML provides the most accurate branch length and gene rate estimates on the simulated data set.

On the other hand, the OrthoMaM data set allows us to observe the robustness of the methods tested to violations of the proportional model, whose assumptions are not expected to hold in real data sets. Moreover, given the relatively large number of genes, this phylogenomic data set is particularly appropriate to assess the computational feasibility of the approaches we implemented.

Computational efficiencies. Running times and memory usages of the tested methods are reported in Table 2. We decomposed running times in two parts: first ($\mathbf{T_1}$), we look at the times necessary for preprocessing steps (essentially gene tree estimation for supertree methods and distance estimation for medium-level methods); second ($\mathbf{T_2}$), we show the remaining running times, to actually produce branch length and gene rate estimates.

Preprocessing times ($\mathbf{T_1}$ in Table 2) show an advantage of medium-level methods ($\mathbf{T_1}$ in the order of the minutes), over supertree methods and Concat+ML, and the advantage of constraining PhyML to only optimize branch lengths and model parameters ($\mathbf{T_1}$ in hours), rather than also seeking an ML topology ($\mathbf{T_1}$ in days). Note however that running times in preprocessing steps is highly and easily parallelizable, meaning that waiting times on parallel architectures will be much lower than the running times indicated here.

Table 2 Computational efficiencies on the OrthoMaM data set for the tested methods.

	Concat+Dist	Concat+ML	SDM*add	DistRadd	ERaBLEadd	SDM*	DistR	ERaBLE
$\boldsymbol{T_1}$	≈ 0	3h20m/39h28m				2m46s		
$\boldsymbol{T_2}$	5m41s	41h16m	8h2m	2h9m	7s	8h33m	2h6m	7s
$\boldsymbol{M}$	889 MB	117 GB	1.2 GB	2.8 GB	222 MB	1.2 GB	3.0 GB	221 MB

NOTE.— The first row gives ($\mathbf{T_1}$) the running time to obtain the data on which subsequent computations are based: the superalignment and the distance-based gene trees for Concat+Dist, the superalignment and ML gene trees for Concat+ML, the ML gene trees and resulting additive distances for the three supertree methods, and the estimated distances for the three medium-level methods. When ML gene trees are used, two alternative approaches are possible and therefore two running times are provided: first that to infer trees with fixed topology, and then that to infer trees where the topology is also estimated. The second row gives ($\mathbf{T_2}$) the remaining running time to obtain estimates for branch lengths and gene rates. The third row (**M**) gives the maximum amount of memory allocated. All the experiments were conducted on a PC with 4 GB RAM and a 2.7 GHz CPU, except branch length estimation ($\mathbf{T_2}$ and **M**) for Concat+ML, which, because of the large memory requirements, was run on a cluster machine with 200 GB RAM and a 2.66 GHz CPU.

Actual processing times ($\mathbf{T_2}$) and memory requirements ($\mathbf{M}$) in Table 2 illustrate the main strength of the new methods we propose here: while for most methods the running times are in the order of the hours (up to about 41h for branch length estimation in Concat+ML) and memory usage in the order of the gigabytes, ERaBLE and ERaBLEadd only require a few seconds and a few hundred megabytes on the OrthoMaM data set. Particularly heavy are the memory requirements for Concat+ML: only users with access to large memory machines may use this method on a large data set (with several thousands of genes) such as OrthoMaM. As for the difference between ERaBLE and the other distance-based methods (SDM-based and DistR-based), this is consistent with the differences in computational complexities of these methods, which only for ERaBLE is linear in m. The only method with computational costs comparable to those of ERaBLE is Concat+Dist, which however on the simulated data leads to inaccurate branch length estimates.

Branch length estimation. Fig. 4 shows the accuracy of all tested methods in the estimation of branch lengths. These experiments confirm that, not surprisingly, DistR-based methods are inaccurate at this task — as already observed in the simulated data set. Moreover, it is clear that the tested methods provide branch length estimates at slightly different scales, as their scatterplots tend to be distributed along non-horizontal lines. SDM-based and ERaBLE-based methods produce branches that are on average 5-20% longer than those estimated by Concat+ML (the same holds for DistR-based methods, although it is harder to observe, because of the large variance of the estimates), whereas Concat+Dist tends to produce shorter branches.

The main reason for these discrepancies is the presence in OrthoMaM of an inverse correlation between the rate of a gene and the depth of its alignment: whereas superalignment methods are sensitive to gene alignment depths – with branch lengths estimates more influenced by genes with many aligned sequences, and thus evolving less rapidly – this is not true for the other tested methods. This observation explains the scale differences observed, as we explain in more detail in Additional file 4.

Gene rate estimation. Fig. 5 shows the accuracy of all tested methods in the estimation of gene rates. Two observations can be made: (1) the main difference in accuracy is now between supertree methods and all other methods (whereas on simulated data, the main difference was between SDM-based methods and the others); (2) again estimates are at slightly different scales, with supertree and medium-level methods having a tendency to estimate lower rates than Concat+ML.

Observation (1) is due to the use of a unique Gamma shape parameter (0.5), common to all genes, to estimate all matrices $\Delta_1, \Delta_2, \ldots, \Delta_m$. Although this is common practice in distance-based analyses, for many genes this is far from the biological reality, as the shape parameters are themselves very different from gene to gene (the distribution of shape parameters inferred by PhyML has a 5% quantile of 0.21, a median of 0.493, and a 95% quantile of 1.73). Because distance estimates are monotonically decreasing functions of the shape parameter [47], underestimating (or overestimating) the shape parameter for gene G_k results in overestimating (respectively, underestimating) all the distances in Δ_k, and therefore the rate r_k. This explains the poor accuracy in gene rate estimation for all methods that use $\Delta_1, \Delta_2, \ldots, \Delta_m$ (i.e., the medium-level methods and Concat+Dist).

It is possible to confirm this explanation by inspecting the genes corresponding to dots that significantly deviate from the red line in Fig. 5, which as expected tend to have ML Gamma shape parameters strongly deviating from 0.5 (not shown). Alternatively, Fig. 5 bis in Additional file 6 shows that if we use gene-specific Gamma shape parameters in the estimation of $\Delta_1, \Delta_2, \ldots, \Delta_m$, then rate estimates become much more accurate for all methods that use these matrices. (However, note that this information is not available from pairwise sequence comparisons only.) Fig. 5 bis also shows that, once the effect described above is taken away, SDM-based methods become again the least accurate — consistent with our results for the simulated data set.

As for observation (2) — the fact that gene rate estimates tend to be lower than those of Concat+ML for all methods except Concat+Dist (see Fig. 5) — it is easy to understand that this is strictly linked to the fact that the estimated branches tend to be longer than those of Concat+ML for all methods except Concat+Dist (see Fig. 4).

Discussion. One of the main differences with the results on the simulated data is the difficulty of setting a scale for branch lengths and gene rates. We expect this observation to extend to most real data sets, where inferring absolute estimates, rather than relative, may be very challenging.

Apart from this scaling issue, the results on the OrthoMaM data set are largely in line with those obtained on the simulated data set: DistR-based methods (DistR and DistRadd) lead to inaccurate branch length estimates, and SDM-based methods (SDM* and SDM*add) lead to inaccurate gene rate estimates — which is not surprising, given that neither of these methods was originally designed for both these tasks (in fact SDM* was designed for neither of these tasks). As for Concat+Dist, the high variance in branch length estimates observed on the simulated data seems to not be present for OrthoMaM. This is surprising, but we recall that it is hard to draw firm conclusions on estimation accuracy from this data set, for the reasons explained above.

That leaves us with ERaBLE-based methods and Concat+ML. It would seem that the choice among ERaBLE, ERaBLEadd and Concat+ML should largely be done based on their tradeoff between accuracy and computational cost (the first method being the fastest and the last the most accurate). One important lesson that the experiments on OrthoMaM highlight, however, is that unless we adopt gene-specific parameters modelling rates-across-sites heterogeneity (e.g., gene-specific Gamma shape parameters), medium-level methods such as ERaBLE may produce inaccurate gene rate estimates.

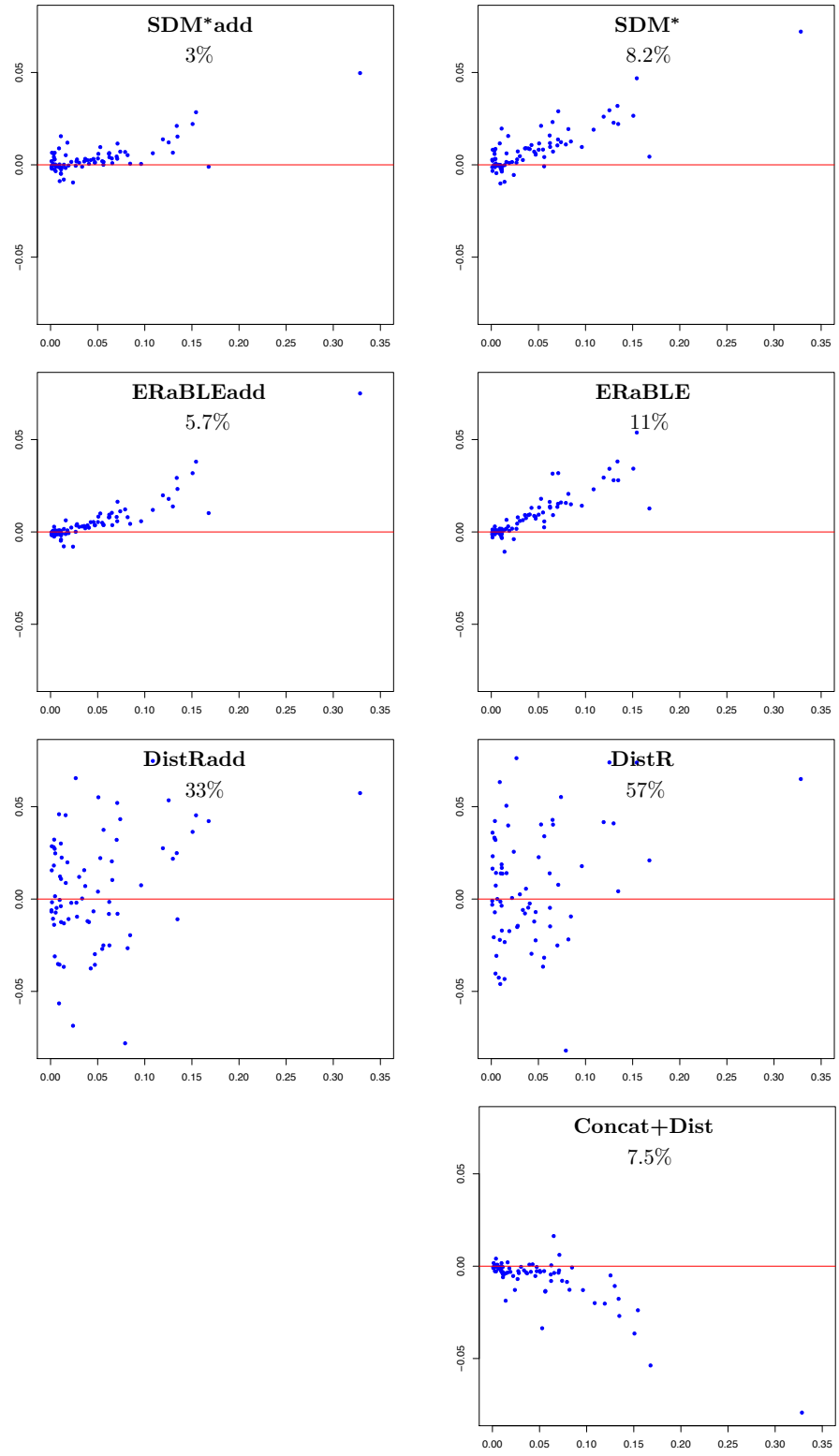

Figure 4 Accuracy of branch length estimates in the OrthoMaM data set. For each method, the 77 branch lengths $\hat{b}_e^{ML}$ estimated by Concat+ML (x-axis) are plotted against the differences $\hat{b}_e - \hat{b}_e^{ML}$ (y-axis) (where $\hat{b}_e$ is the estimate for the length of e obtained by the method at the top of the plot). The horizontal red line corresponds to no difference between the two estimates. Method names are shown at the top of each plot, followed by the fraction of variance unexplained of $(\hat{b}_e^{ML})$ relative to $(\hat{b}_e)$ (see Additional file 3).

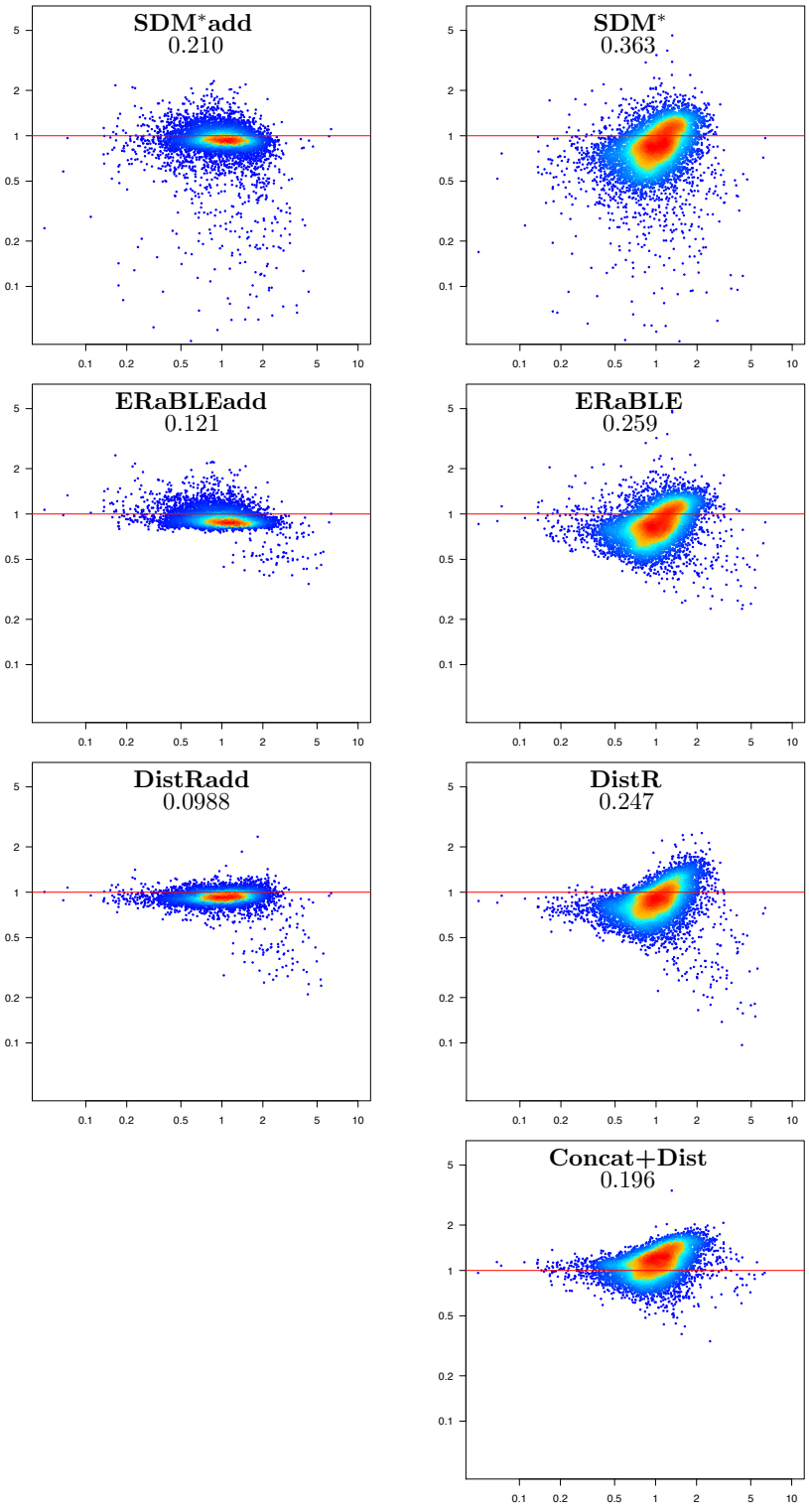

Figure 5 Estimation accuracy for gene rates in the OrthoMaM data set. Logarithmic scatterplots showing the 6,953 "reference" gene rates $\hat{r}_k^{ML}$ estimated by Concat+ML (x-axis), against ratios $\hat{r}_k/\hat{r}_k^{ML}$ (y-axis). Note that errors relative to the reference gene rates are measured with ratios, instead of differences. Colors (from blue to red) indicate increased density of points. The horizontal red line corresponds to no difference between the two estimates. Method names are shown at the top of each plot, followed by the mean absolute log-ratio between estimated and reference gene rates (see Additional file 3).

Conclusion

In this paper we have examined the notions of branch lengths in a species tree and of gene rates in a phylogenomic context. We have presented ERaBLE, a novel and efficient method for the estimation of these quantities, which are often overlooked in phylogenomic analyses – for example by classical supertree methods – or whose estimation requires computationally-demanding methodologies – usually likelihood-based analysis of a concatenated superalignment. Note that for large phylogenomic data sets such as OrthoMaM – where the concatenated alignment consists of more than 6 million sites – the application of likelihood is very onerous, especially in terms of memory requirements, which may be problematic for some users. Recall that in our experiments we have constrained the tree topology; a full likelihood analysis would further increase computational costs. Moreover, in Additional file 7, we show that ML methods more efficient than PhyML – namely ExaML [52] and FastTree 2 [53] – are still very inefficient relative to ERaBLE.

Methodologically, ERaBLE represents the fastest available method to estimate the branch lengths of a given topology from a collection of distance matrices – one matrix per gene under consideration. It generalises and reduces to (when only one matrix is provided) classical WLS branch length estimation. The most important difference with single-gene WLS is that ERaBLE also estimates gene rate parameters, modelling the different "speeds" of evolution of different genes — with little computational overhead.

ERaBLE's limitations are its reliance on a tree topology – either a well-accepted phylogeny or a tree reconstructed prior to its execution – and its seemingly strong assumptions about the data (orthology of the genes under analysis, and the proportional model). However, we stress that these hypotheses represent an ideal scenario. As shown by the experiments on the OrthoMaM data set, ERaBLE can perform well on real-world data sets where these assumptions will probably be violated to some degree, namely because of phenomena such as heterotachy [37] or limited topological incompatibilities due to incomplete lineage sorting (ILS), gene duplication and/or lateral gene transfer [35]. In order to investigate the robustness of ERaBLE and competing methods to these violations, it would be interesting to simulate data following more realistic assumptions, for example those of the multispecies coalescent [33] to study the effects of ILS. Furthermore, it would be useful to model alignment errors which are undoubtedly present in real data.

ERaBLE can be used in two ways, which differ in the way the input distance matrices are obtained: they can either be directly estimated from gene alignments, or they can be based on phylogenetic trees inferred for each gene. Our experiments (Results and discussion section) show that both these approaches provide valid alternatives to existing methodologies: the alternative methods are either only accurate for one of the two tasks that ERaBLE carries out — SDM-based methods provide branch lengths estimates comparable to those of ERaBLE, while DistR-based methods provides marginally better gene rates estimates — or computationally very demanding — as in the case of ML analysis of a concatenated superalignment.

A possible use of ERaBLE is as a complement to classical supertree methods (e.g., MRP [20, 21]), which often disregard branch length information, yet present in the input trees. In this context, ERaBLE would allow to rapidly assign meaningful branch lengths to the tree topologies reconstructed by these methods.

Alternatively, when (most of) the evolutionary relationships between the species under consideration are relatively well-known, ERaBLE can be used as a standalone, using a reference topology as input. This is the scenario that we have assumed in our experiments on the OrthoMaM data set. An interesting question for future research is the robustness of ERaBLE's estimates to errors in the reference topology.

Furthermore, it would be interesting to investigate the possibility of combining the assignment of branch lengths made by ERaBLE with a criterion for topological inference, allowing to score different tree topologies for their fit with the data. This is analogous to what is done in classical distance-based phylogenetics, where least squares branch lengths can be used in combination with criteria such as minimum evolution [54]. However, this would probably need more methodological advances, first, to reduce further the time needed to evaluate a topology — if possible, by extending the approaches shown for particular cases of WLS in recent studies [40, 43, 55] — and, second, to avoid issues of statistical inconsistency, which are known to affect single-matrix WLS in combination with minimum evolution [56].

Availability of supporting data
The simulated data set and the OrthoMaM data set supporting the results of this article can be found online at: http://www.atgc-montpellier.fr/erable

List of abbreviations
ACS: Average Consensus Supertree
BWD: Build With Distances
ERaBLE: Evolutionary Rates and Branch Length Estimation
ML: Maximum Likelihood
MRP: Matrix Representation with Parsimony
SDM: Super Distance Matrix
OLS: Ordinary Least Squares
WLS: Weighted Least Squares
GLS: Generalized Least Square

Competing interests
The authors declare that they have no competing interests.

Author's contributions
FP and MB designed the algorithm. MB implemented it and performed the analyses. MB, OG, CS, EJPD and FP participated in the design of the experiments and in the interpretation of the results. MB and FP wrote the manuscript and all authors read and approved it.

Acknowledgements
We wish to thank Alexis Criscuolo, Pascal Giorgi, Vincent Lefort, Vincent Ranwez and Bastien Vialla for useful discussions and assistance. The PhD of MB is funded by the Labex NUMEV. MB, OG and FP are also funded by the EC H2020 project VIROGENESIS (grant number 634650). This publication is contribution No 2015-173 of the Institut des Sciences de l'Evolution de Montpellier (UMR5544 - UM + CNRS + IRD).

Author details
[1]Laboratoire d'Informatique de Robotique et de Microélectronique de Montpellier (LIRMM), CNRS, Université de Montpellier, France. [2]Institut de Biologie Computationnelle, Montpellier, France. [3]Institut des Sciences de l'Evolution de Montpellier, CNRS, IRD, EPHE, Université de Montpellier, France.

References

1. Burleigh, J.G., Bansal, M.S., Eulenstein, O., Hartmann, S., Wehe, A., Vision, T.J.: Genome-scale phylogenetics: inferring the plant tree of life from 18,896 gene trees. Systematic Biology **60**(2), 117–125 (2011)
2. Criscuolo, A., Gribaldo, S.: Large-scale phylogenomic analyses indicate a deep origin of primary plastids within cyanobacteria. Molecular Biology and Evolution **28**(11), 3019–3032 (2011)
3. Baker, A.J., Haddrath, O., McPherson, J.D., Cloutier, A.: Genomic support for a moa–tinamou clade and adaptive morphological convergence in flightless ratites. Molecular Biology and Evolution **31**(7), 1686–1696 (2014)
4. Pupko, T., Huchon, D., Cao, Y., Okada, N., Hasegawa, M.: Combining multiple data sets in a likelihood analysis: which models are the best? Molecular Biology and Evolution **19**(12), 2294–2307 (2002)
5. Bevan, R.B., Lang, B.F., Bryant, D.: Calculating the evolutionary rates of different genes: a fast, accurate estimator with applications to maximum likelihood phylogenetic analysis. Systematic Biology **54**(6), 900–915 (2005)

6. Lartillot, N., Philippe, H.: A bayesian mixture model for across-site heterogeneities in the amino-acid replacement process. Molecular Biology and Evolution **21**(6), 1095–1109 (2004)
7. Pagel, M., Meade, A.: A phylogenetic mixture model for detecting pattern-heterogeneity in gene sequence or character-state data. Systematic Biology **53**(4), 571–581 (2004)
8. Fan, Y., Wu, R., Chen, M.-H., Kuo, L., Lewis, P.O.: Choosing among partition models in bayesian phylogenetics. Molecular Biology and Evolution **28**(1), 523–532 (2011)
9. Lanfear, R., Calcott, B., Ho, S.Y., Guindon, S.: PartitionFinder: combined selection of partitioning schemes and substitution models for phylogenetic analyses. Molecular Biology and Evolution **29**(6), 1695–1701 (2012)
10. Wiens, J., Morrill, M.: Missing data in phylogenetic analysis: reconciling results from simulations and empirical data. Systematic Biology **60**(5), 719–731 (2011)
11. Roure, B., Baurain, D., Philippe, H.: Impact of missing data on phylogenies inferred from empirical phylogenomic data sets. Molecular Biology and Evolution **30**(1), 197–214 (2013)
12. Zuckerkandl, E., Pauling, L.: Molecules as documents of evolutionary history. Journal of Theoretical Biology **8**(2), 357–366 (1965)
13. Douzery, E.J., Snell, E.A., Bapteste, E., Delsuc, F., Philippe, H.: The timing of eukaryotic evolution: does a relaxed molecular clock reconcile proteins and fossils? Proceedings of the National Academy of Sciences USA **101**(43), 15386–15391 (2004)
14. Merkle, D., Middendorf, M.: Reconstruction of the cophylogenetic history of related phylogenetic trees with divergence timing information. Theory in Biosciences **123**(4), 277–299 (2005)
15. Faith, D.P.: Conservation evaluation and phylogenetic diversity. Biological Conservation **61**(1), 1–10 (1992)
16. Margulies, E.H., Blanchette, M., Haussler, D., NISC Comparative Sequencing Program, E.D. Green: Identification and characterization of multi-species conserved sequences. Genome Research **13**(12), 2507–2518 (2003)
17. Wolf, Y.I., Novichkov, P.S., Karev, G.P., Koonin, E.V., Lipman, D.J.: The universal distribution of evolutionary rates of genes and distinct characteristics of eukaryotic genes of different apparent ages. Proceedings of the National Academy of Sciences USA **106**(18), 7273–7280 (2009)
18. Bininda-Emonds, O.R.: The evolution of supertrees. Trends in Ecology & Evolution **19**(6), 315–322 (2004)
19. Scornavacca, C.: Supertree methods for phylogenomics. PhD thesis, Université de Montpellier II Sciences et Techniques du Languedoc, Montpellier, France (2009)
20. Baum, B.R.: Combining trees as a way of combining data sets for phylogenetic inference, and the desirability of combining gene trees. Taxon **41**(2), 3–10 (1992)
21. Ragan, M.A.: Phylogenetic inference based on matrix representation of trees. Molecular Phylogenetics and Evolution **1**(1), 53–58 (1992)
22. Swenson, M.S., Suri, R., Linder, C.R., Warnow, T.: SuperFine: fast and accurate supertree estimation. Systematic Biology **61**(2), 214–227 (2012)
23. Willson, S.J.: Constructing rooted supertrees using distances. Bulletin of Mathematical Biology **66**(6), 1755–1783 (2004)
24. Lapointe, F.-J., Cucumel, G.: The average consensus procedure: combination of weighted trees containing identical or overlapping sets of taxa. Systematic Biology **46**(2), 306–312 (1997)
25. Criscuolo, A., Berry, V., Douzery, E.J., Gascuel, O.: SDM: a fast distance-based approach for (super) tree building in phylogenomics. Systematic Biology **55**(5), 740–755 (2006)
26. de Queiroz, A., Gatesy, J.: The supermatrix approach to systematics. Trends in Ecology & Evolution **22**(1), 34–41 (2007)
27. Schmidt, H.A.: Phylogenetic trees from large datasets. PhD thesis, Universität Düsseldorf, Düsseldorf, Germany (2003)
28. Kupczok, A., Schmidt, H.A., von Haeseler, A.: Accuracy of phylogeny reconstruction methods combining overlapping gene data sets. Algorithms for Molecular Biology **5**(1), 1–17 (2010)
29. Strimmer, K., Von Haeseler, A.: Quartet puzzling: a quartet maximum-likelihood method for reconstructing tree topologies. Molecular Biology and Evolution **13**(7), 964–969 (1996)
30. Schmidt, H.A., Strimmer, K., Vingron, M., von Haeseler, A.: TREE-PUZZLE: maximum likelihood phylogenetic analysis using quartets and parallel computing. Bioinformatics **18**(3), 502–504 (2002)
31. Semple, C., Steel, M.: Phylogenetics. Oxford University Press, New York, USA (2003)
32. Tajima, F.: Evolutionary relationship of DNA sequences in finite populations. Genetics **105**(2), 437–460 (1983)
33. Degnan, J.H., Rosenberg, N.A.: Gene tree discordance, phylogenetic inference and the multispecies coalescent. Trends in Ecology & Evolution **24**(6), 332–340 (2009)
34. Goodman, M., Czelusniak, J., Moore, G.W., Romero-Herrera, A., Matsuda, G.: Fitting the gene lineage into its species lineage, a parsimony strategy illustrated by cladograms constructed from globin sequences. Systematic Zoology **28**, 132–163 (1979)
35. Maddison, W.P.: Gene trees in species trees. Systematic Biology **46**(3), 523–536 (1997)
36. Yang, Z.: Maximum-likelihood models for combined analyses of multiple sequence data. Journal of Molecular Evolution **42**(5), 587–596 (1996)
37. Lopez, P., Casane, D., Philippe, H.: Heterotachy, an important process of protein evolution. Molecular Biology and Evolution **19**(1), 1–7 (2002)
38. Bulmer, M.: Use of the method of generalized least squares in reconstructing phylogenies from sequence data. Molecular Biology and Evolution **8**(6), 868–883 (1991)
39. Fitch, W.M., Margoliash, E.: Construction of phylogenetic trees. Science **155**(3760), 279–284 (1967)
40. Desper, R., Gascuel, O.: Fast and accurate phylogeny reconstruction algorithms based on the minimum-evolution principle. Journal of Computational Biology **9**(5), 687–705 (2002)
41. Luenberger, D.G., Ye, Y.: Linear and Nonlinear Programming vol. 116. Springer, New York, USA (2008)
42. Bryant, D., Waddell, P.: Rapid evaluation of least-squares and minimum-evolution criteria on phylogenetic trees. Molecular Biology and Evolution **15**(10), 1346–1359 (1998)
43. Mihaescu, R., Pachter, L.: Combinatorics of least-squares trees. Proceedings of the National Academy of

Sciences USA **105**(36), 13206–13211 (2008)
44. Douzery, E.J., Scornavacca, C., Romiguier, J., Belkhir, K., Galtier, N., Delsuc, F., Ranwez, V.: Orthomam v8: a database of orthologous exons and coding sequences for comparative genomics in mammals. Molecular Biology and Evolution **31**(7), 1923–1928 (2014)
45. Guindon, S., Gascuel, O.: A simple, fast, and accurate algorithm to estimate large phylogenies by maximum likelihood. Systematic Biology **52**(5), 696–704 (2003)
46. Rambaut, A., Grass, N.C.: Seq-Gen: an application for the Monte Carlo simulation of DNA sequence evolution along phylogenetic trees. Computer Applications in the Biosciences: CABIOS **13**(3), 235–238 (1997)
47. Yang, Z.: Computational Molecular Evolution vol. 21. Oxford University Press Oxford, Oxford, UK (2006)
48. Flicek, P., Amode, M.R., Barrell, D., Beal, K., Billis, K., Brent, S., Carvalho-Silva, D., Clapham, P., Coates, G., Fitzgerald, S., Gil, L., Girón, C.G., Gordon, L., Hourlier, T., Hunt, S., Johnson, N., Juettemann, T., Kähäri, A.K., Keenan, S., Kulesha, E., Martin, F.J., Maurel, T., McLaren, W.M., Murphy, D.N., Nag, R., Overduin, B., Pignatelli, M., Pritchard, B., Pritchard, E., Riat, H.S., Ruffier, M., Sheppard, D., Taylor, K., Thormann, A., Trevanion, S.J., Vullo, A., Wilder, S.P., Wilson, M., Zadissa, A., Aken, B.L., Birney, E., Cunningham, F., Harrow, J., Herrero, J., Hubbard, T.J.P., Kinsella, R., Muffato, M., Parker, A., Spudich, G., Yates, A., Zerbino, D.R., Searle, S.M.J.: Ensembl 2014. Nucleic Acids Research (2013). doi:10.1093/nar/gkt1196. http://nar.oxfordjournals.org/content/early/2013/12/06/nar.gkt1196.full.pdf+html
49. Capella-Gutiérrez, S., Silla-Martínez, J.M., Gabaldón, T.: trimAl: a tool for automated alignment trimming in large-scale phylogenetic analyses. Bioinformatics **25**(15), 1972–1973 (2009)
50. Guindon, S., Dufayard, J.-F., Lefort, V., Anisimova, M., Hordijk, W., Gascuel, O.: New algorithms and methods to estimate maximum-likelihood phylogenies: assessing the performance of PhyML 3.0. Systematic Biology **59**(3), 307–321 (2010)
51. Buneman, P.: The Recovery of Trees from Measures of Dissimilarity. In: Kendall, D.G., Tautu, P. (eds.) Mathematics the the Archeological and Historical Sciences, pp. 387–395. Edinburgh University Press, UK (1971)
52. Stamatakis, A., Aberer, A.J.: Novel parallelization schemes for large-scale likelihood-based phylogenetic inference. In: Parallel & Distributed Processing (IPDPS), 2013 IEEE 27th International Symposium On, Boston, USA, pp. 1195–1204 (2013). IEEE
53. Price, M.N., Dehal, P.S., Arkin, A.P.: FastTree 2–approximately maximum-likelihood trees for large alignments. PLOS ONE **5**(3), 9490 (2010)
54. Kidd, K.K., Sgaramella-Zonta, L.A.: Phylogenetic analysis: concepts and methods. American Journal of Human Genetics **23**(3), 235 (1971)
55. Pardi, F., Gascuel, O.: Combinatorics of distance-based tree inference. Proceedings of the National Academy of Sciences USA **109**(41), 16443–16448 (2012)
56. Gascuel, O., Bryant, D., Denis, F.: Strengths and limitations of the minimum evolution principle. Systematic Biology **50**(5), 621–627 (2001)

Additional Files

Additional file 1 (.pdf) — ERaBLE, in detail
Additional file 2 (.pdf) — The choice of Z_k in the constraint
Additional file 3 (.pdf) — Estimation accuracy measures
Additional file 4 (.pdf) — The scales of branch lengths for the OrthoMaM data set
Additional file 5 (.pdf) — Reference topology for the OrthoMaM data set
Additional file 6 (.pdf) — Usefulness of gene-specific Gamma shape parameters
Additional file 7 (.pdf) — Alternative ML methods for Concat+ML

Additional file 1 — ERaBLE, in detail

In this additional file we show how ERaBLE computes the solution for problem (6). We start by introducing some notation that allows us to rewrite the problem in matrix form.

Inputs and outputs. First, let $\hat{\alpha} = (\hat{\alpha}_1, \hat{\alpha}_2, \ldots, \hat{\alpha}_m)^t$ and $\hat{b} = (\hat{b}_1, \hat{b}_2, \ldots, \hat{b}_\tau)^t$ designate the unknowns of the problem in column vector form, where $\tau = |E(\mathcal{T})|$ is the number of branches of the topology $\mathcal{T}$ and the superscript t denotes the transpose operator. Let then δ_k be a vector listing all the input distances $\delta_{ij}^{(k)}$ for gene G_k in lexicographic order with respect to the taxon indices. For example, if $L_1 = \{1, 2, 4, 6\}$, then $\delta_1 = (\delta_{12}^{(1)}, \delta_{14}^{(1)}, \delta_{16}^{(1)}, \delta_{24}^{(1)}, \delta_{26}^{(1)}, \delta_{46}^{(1)})^t$. Similarly, let $\hat{d}$ denote the vector containing the additive distances $\hat{d}_{ij}$ resulting from the branch lengths in $\hat{b}$, again ordered lexicographically. Finally, let $\hat{d}_k$ be the vector that is obtained from $\hat{d}$ by removing all the distances involving taxa not in L_k. Informally, problem (6) requires the vectors $\hat{\alpha}_k\delta_k$ and $\hat{d}_k$ to be as close as possible, for all $k \in \{1, 2, \ldots, m\}$.

Topological matrices. Now, let A be the topological matrix representing $\mathcal{T}$. This is a $n(n-1)/2 \times \tau$ binary matrix, whose τ columns correspond to branches of $\mathcal{T}$, and whose $n(n-1)/2$ rows correspond to pairs of taxa in L, in lexicographical order (see, e.g., [42]). $A = (a_{ij,e})$ is defined by setting $a_{ij,e} = 1$, if e is on the path between i and j, and 0 otherwise. Moreover, let A_k be the $|L_k|(|L_k|-1)/2 \times \tau$ binary matrix that is obtained from A by removing all the rows corresponding to taxa not in L_k. Using these notations, we can write $\hat{d} = A\hat{b}$, and $\hat{d}_k = A_k\hat{b}$.

Weight matrices and vectors. Let W_k be the square matrix of order $|L_k|(|L_k|-1)/2$ whose diagonal entries are the weights $w_{ij}^{(k)}$, and whose every other element is zero. Finally, let $z = (Z_1, Z_2, \ldots, Z_m)^t$ and Z denote the sum of all Z_k, for $k = 1, 2, \ldots, m$.

The problem and its resolution via Lagrange multipliers. Using the notation above, problem (6) can be expressed as follows:

$$\begin{cases} \text{minimize}_{\hat{\alpha},\hat{b}} & Q(\hat{\alpha}, \hat{b}) = \displaystyle\sum_{k=1}^{m} (\hat{\alpha}_k\delta_k - A_k\hat{b})^t W_k (\hat{\alpha}_k\delta_k - A_k\hat{b}), \\ \text{subject to} & z^t\hat{\alpha} = Z. \end{cases} \tag{8}$$

We solve problem (8) using the method of Lagrange multipliers [41]. This method relies on the Lagrangian function, which here is given by $\mathcal{L}(\hat{\alpha}, \hat{b}, \lambda) = Q(\hat{\alpha}, \hat{b}) + \lambda(z^t\hat{\alpha} - Z)$. A necessary condition for $(\hat{\alpha}, \hat{b})$ to be a solution of problem (8) is that all the partial derivatives of $\mathcal{L}$ be zero:

$$\nabla_{\hat{\alpha},\hat{b},\lambda}\mathcal{L}(\hat{\alpha}, \hat{b}, \lambda) = 0. \tag{9}$$

Although in general (9) is only a necessary condition for a minimum, here it is also sufficient, as the function to minimize $Q(\hat{\alpha}, \hat{b})$ is a sum of squares, thus convex, and the constraint is linear. Solving problem (8) is thus equivalent to solving equation (9), which can be written as follows:

$$\begin{cases} \dfrac{\partial\mathcal{L}}{\partial\hat{\alpha}_k} = 0 & \Leftrightarrow & \hat{\alpha}_k\delta_k^t W_k\delta_k - \delta_k^t W_k A_k\hat{b} + Z_k\lambda/2 = 0 & m \text{ equations } (k = 1, \ldots, m). \\ \nabla_{\hat{b}}\mathcal{L} = 0 & \Leftrightarrow & \displaystyle\sum_{k=1}^{m} \left(A_k^t W_k A_k\hat{b} - \hat{\alpha}_k A_k^t W_k\delta_k\right) = 0 & \tau \text{ equations.} \\ \dfrac{\partial\mathcal{L}}{\partial\lambda} = 0 & \Leftrightarrow & z^t\hat{\alpha} = Z & 1 \text{ equation.} \end{cases} \tag{10}$$

To simplify system (10) we define the following matrices: D is the $m \times m$ matrix whose diagonal entries are the scalars $\delta_k^t W_k\delta_k$, and whose every other element is zero; B is the $\tau \times m$ matrix whose columns are the vectors $-A_k^t W_k\delta_k$; C is the $\tau \times \tau$ matrix defined by $C = \sum_{k=1}^{m} A_k^t W_k A_k$. After dropping the $1/2$ coefficient for λ, as we are not interested in the value of the multiplier, system (10) can be written

as:

$$\begin{cases} D\hat{\alpha} + B^t\hat{b} + \lambda z = 0, & (11) \\ B\hat{\alpha} + C\hat{b} = 0, & (12) \\ z^t\hat{\alpha} = Z. & (13) \end{cases}$$

Naïve matrix multiplication allows to calculate the coefficients of this system in $\mathcal{O}(mn^4)$ time, as this is dominated by the computation of $C = \sum_k A_k^t W_k A_k$, where each $A_k^t W_k A_k$ can be obtained in $\mathcal{O}(\tau^2 n^2) = \mathcal{O}(n^4)$ time (using the fact that W_k is diagonal). Adding to this the time taken by standard algorithms for the resolution of this system in $\mathcal{O}(m+n)$ equations and unknowns, we get to a total complexity of $\mathcal{O}(mn^4 + (n+m)^3)$ time for the naïve algorithm. For the data sets typical in phylogenomics this would be unfeasible. Below, we show how to bring this down to $\mathcal{O}(mn^2 + n^3)$ time.

Efficient solution of the linear system. First isolate $\hat{\alpha}$ in (11):

$$\hat{\alpha} = -D^{-1}\big(B^t\hat{b} + \lambda z\big). \tag{14}$$

Then substitute $\hat{\alpha}$ in (13) and isolate λ:

$$\lambda = -\frac{Z + z^t D^{-1} B^t \hat{b}}{z^t D^{-1} z} = -\frac{Z + u^t\hat{b}}{\omega},$$

where we define the vector $u = BD^{-1}z$ and the scalar $\omega = z^t D^{-1} z$. Replace then λ in (14) with the expression just obtained:

$$\hat{\alpha} = D^{-1}\Big(-B^t\hat{b} + z\Big(\frac{Z + u^t\hat{b}}{\omega}\Big)\Big) = D^{-1}\Big(\frac{zu^t}{\omega} - B^t\Big)\hat{b} + \frac{Z}{\omega}D^{-1}z. \tag{15}$$

Then replace $\hat{\alpha}$ with expression (15) in equation (12):

$$0 = B\hat{\alpha} + C\hat{b} = \Big(C + \frac{uu^t}{\omega} - BD^{-1}B^t\Big)\hat{b} + \frac{Z}{\omega}u.$$

If we let

$$M = \Big(C + \frac{uu^t}{\omega} - BD^{-1}B^t\Big),$$

then $\hat{b}$ can be found by solving the following system:

$$M\hat{b} = -\frac{Z}{\omega}u. \tag{16}$$

Finally $\hat{\alpha}$ can be obtained by using the value found for $\hat{b}$ in (15).

Remark on the scale of the results. Equations (16) and (15) — whose right-hand-sides are directly proportional to Z — show that the solutions $\hat{\alpha}$ and $\hat{b}$ scale proportionally with Z, the right-hand side of the constraint in our problem (8). Since ERaBLE subsequently resets the scale of $\hat{\alpha}$ and $\hat{b}$, by multiplying them by the correction factor in equation (7), this shows that the value of Z is irrelevant to the end results.

Uniqueness of the solution. If M is not invertible, then our optimization problem has multiple solutions. This happens when the sequence coverage is insufficient, with pairs of taxa i, j in crucial positions within $\mathcal{T}$, such that $\delta_{ij}^{(k)}$ is undefined for all $k \in \{1, 2, \ldots, m\}$. We note however that for the data sets that we consider here — with at least hundreds of genes — it is very unlikely to encounter this problem, unless the sequence coverage is extremely low. For example, the solution is unique for all the data sets we used in our experiments (simulated or real, and whose matrices cover from 4 to 40 taxa). A precise mathematical characterisation of the data sets guaranteeing the uniqueness of solutions is possible, but beyond the scope of this paper.

Computational complexity. M is a square matrix of order $\tau = \mathcal{O}(n)$. The resolution of the linear system in equation (16) can be carried out using standard algorithms in $\mathcal{O}(n^3)$ time and $\mathcal{O}(n^2)$ memory.

Taking into account all the other operations involved — most notably calculating all the coefficients of this linear system — the total complexity of ERaBLE is then of $\mathcal{O}(n^3 + mn^2)$ time and $\mathcal{O}(n^2 + mn)$ memory (in addition to that used to store the inputs). This is dominated by the computation of the entries in the matrices B, C, D, as we now show.

Computing matrix B. Bryant and Waddell [42] showed that it is possible to calculate the product $A^t v$, where A is a topological matrix for a tree with n leaves, and v is any vector with $n(n-1)/2$ entries, with a time complexity of $\mathcal{O}(n^2)$. This algorithm is trivially generalized to a partial topological matrix A_k, meaning that we can calculate each column of B, that is $-A_k^t(W_k\delta_k)$, in $\mathcal{O}(n^2)$ time, leading to a total time complexity of $\mathcal{O}(mn^2)$ for calculating B. Note that storing B requires $\mathcal{O}(mn)$ memory.

Computing matrix C. Recall that $C = \sum_{k=1}^{m} A_k^t W_k A_k$. We show that each each $A_k^t W_k A_k$ can be computed in $\mathcal{O}(n^2)$ time, leading to a time complexity of $\mathcal{O}(mn^2)$ for calculating C.

$A_k^t W_k A_k$ is a $\tau \times \tau$ square matrix where each entry corresponds to a pair of branches e, f in $\mathcal{T}$. It is easy to see that the entries of this matrix can be expressed as follows:

$$(A_k^t W_k A_k)_{ef} = \sum_{\substack{i \in X \cap L_k \\ j \in Y \cap L_k}} w_{ij}^{(k)}, \tag{17}$$

where X and Y denote the disjoint sets of taxa separated by both e and f, as shown in Fig. 6. (Formally, X and Y are the disjoint sets of taxa such that any path from an element of X to an element of Y must pass via both e and f.)

Figure 6 – X and Y are the disjoint sets of taxa separated by both e and f.

Now let $C_{XY}^{(k)}$ denote the right-hand side of Eqn. (17). We can calculate all the $C_{XY}^{(k)}$ values recursively:

– If X and Y are singletons with $X = \{i\}$ and $Y = \{j\}$, then :

$$C_{XY}^{(k)} = C_{ij}^{(k)} = \begin{cases} w_{ij}^{(k)} & \text{if } \{i,j\} \subset L_k, \\ 0 & \text{otherwise.} \end{cases}$$

– Otherwise, one of the two taxon sets, say, Y can be decomposed in a number of disjoint subsets $Y = \bigcup_{i=1}^{d} Y_i$, corresponding to the subtrees of the tree rooted in f and having Y as leaf set. Then:

$$C_{XY}^{(k)} = \sum_{i=1}^{d} C_{XY_i}^{(k)}.$$

Since there are $\mathcal{O}(n^2)$ $C_{XY}^{(k)}$ values to calculate, the entire matrix $A_k^t W_k A_k$ can be filled in $\mathcal{O}(n^2)$ time and requires $\mathcal{O}(n^2)$ memory to be stored.

Computing matrix D. Since D is a diagonal matrix, we only need to calculate and store the elements on its diagonal, each of which can be trivially obtained in $O(n^2)$ time, leading to a total of $O(mn^2)$ time and $O(m)$ memory.

Other computations. All the remaining calculations can be done within complexities that are of the same order as, or inferior to those detailed above. Thus $\hat{b}$ and $\hat{\alpha}$ can be obtained in $\mathcal{O}(n^3 + mn^2)$ time and $\mathcal{O}(n^2 + mn)$ (auxiliary) memory.

Additional file 2 — The choice of Z_k in the constraint

While testing ERaBLE we have realised that setting $Z_k = 1$ or $Z_k = N_k$, for all $k \in \{1, 2, \ldots, m\}$ can cause important overestimations of the scale factors $\hat{\alpha}_k$ for genes only present in a small group of closely related taxa. This phenomenon is strictly linked to the strong underestimation of a minority of gene rates — and the slight overestimation of the majority of gene rates — observed for SDM-based methods, which also use the constraint with $Z_k = 1$. In our experiments we have set $Z_k = N_k \sum_{i,j \in L_k} \delta_{ij}^{(k)}$, which largely solves this problem, despite being rather heuristic. In this additional file, we show the importance of the constraint used by ERaBLE with a very simple example.

We construct a small data set consisting of just two nucleotide alignments, those of exons ENSG00000066654_THUMPD1_000 and ENSG00000127423_AUNIP_000 obtained from OrthoMaM after trimAl filtering. We call them G_1 and G_2, respectively. For simplicity we only keep the sequences of six species, those in the set $L = \{$*Gorilla, Homo, Pan, Bos, Erinaceus, Sorex*$\}$. Since G_1 is only sampled in primates, we have $L_1 = \{$*Gorilla, Homo, Pan*$\}$, and $L_2 = L$. Alignment lengths are $N_1 = 489$ for G_1, and $N_2 = 855$ for G_2. Figure 7 shows a phylogenetic tree for these data.

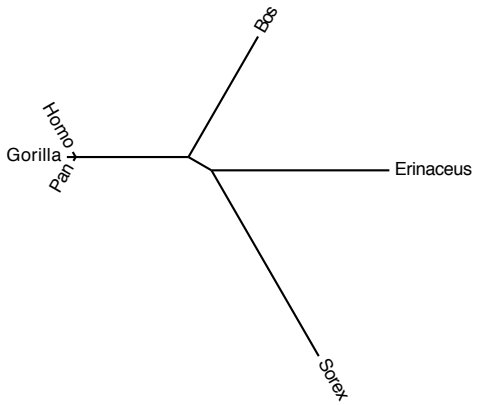

Figure 7 – Maximum likelihood tree (PhyML with model TN93+Γ_8) obtained on the concatenation of the two alignments in the example.

Fig. 8 shows the distance matrices estimated for G_1 and G_2 (left column), and the different behaviours of ERaBLE with $Z_k = 1$ and $Z_k = N_k \sum_{i,j \in L_k} \delta_{ij}^{(k)}$ (middle and right column, respectively). The behaviour for $Z_k = N_k$ is similar to that for $Z_k = 1$, and not shown here for brevity. A quick comparison of Δ_1 and Δ_2 suggests that the rate of G_1 is higher than that of G_2 (note that, in two cases out of three, $\delta_{ij}^{(1)}$ is more than the double than $\delta_{ij}^{(2)}$). We then expect that $\hat{\alpha}_1 < \hat{\alpha}_2$. However, when $Z_k = 1$, the opposite happens: solving problem (6) leads to $\hat{\alpha}_1 = 1.73$ and $\hat{\alpha}_2 = 0.274$.

The key observation to understand why this happens is that Δ_1 only contains very closely related species (all great apes, see also Fig. 7), so its distances are very small relative to many of those in Δ_2, which can be up to about 30 times larger. As a consequence, if $\hat{\alpha}_1 \leq \hat{\alpha}_2$ the value of the objective function $Q(\hat{\alpha}, \hat{b})$ is dominated by the differences $\hat{\alpha}_2 \delta_{ij}^{(2)} - \hat{d}_{ij}$, that is, the difference between $\hat{\alpha}_2 \Delta_2$ and $\widehat{D} = (\hat{d}_{ij})$. It is then intuitive that a way to reduce $Q(\hat{\alpha}, \hat{b})$ is to simultaneously reduce the scale of $\hat{\alpha}_2 \Delta_2$ and $\widehat{D} = (\hat{d}_{ij})$, which can be achieved by decreasing the value of $\hat{\alpha}_2$ (and consequently increasing that of $\hat{\alpha}_1$, given that for $Z_k = 1$ their mean is constrained to be 1).

This is precisely what is happening when setting $Z_k = 1$ in our example (middle column in Fig. 8): instead of having $\hat{\alpha}_1 < \hat{\alpha}_2$, ERaBLE produces a small $\hat{\alpha}_2 = 0.274$ and a large $\hat{\alpha}_1 = 1.73$. Compared to the alternative constraint (right column), where we have $\hat{\alpha}_1 < \hat{\alpha}_2$ as expected, it is clear that this results

Input distances

	Homo	Pan	Bos	Erinac.	Sorex
Gorilla	.0203	.0135			
Homo		.0148	Δ_1		
Gorilla	.0087	.0099	.278	.354	.432
Homo		.0062	.279	.342	.432
Pan			.286	.345	.429
Bos				.419	.422
Erinac.	Δ_2				.446

Trivial constraint $Z_k = 1$

$\hat{\alpha}_1$ **1.73** $\hat{\alpha}_2$ **.274**

	Homo	Pan	Bos	Erinac.	Sorex
Gorilla	.0351	.0233			
Homo		.0256	$\hat{\alpha}_1\Delta_1$		
Gorilla	.0024	.0028	.0763	.0970	.118
Homo		.0017	.0766	.0937	.118
Pan			.0783	.0945	.118
Bos				.115	.116
Erinac.	$\hat{\alpha}_2\Delta_2$				.122
Gorilla	.0127	.0117	.0777	.0985	.116
Homo		.0104	.0774	.0983	.116
Pan			.0764	.0973	.115
Bos				.107	.124
Erinac.	$\widehat{D}$				.122

Our constraint $Z_k = N_k \sum_{i,j \in L_k} \delta_{ij}^{(k)}$

$\hat{\alpha}_1$ **0.538** $\hat{\alpha}_2$ **1.00**

	Homo	Pan	Bos	Erinac.	Sorex
Gorilla	.0109	.0073			
Homo		.0080	$\hat{\alpha}_1\Delta_1$		
Gorilla	.0087	.0099	.278	.354	.432
Homo		.0062	.279	.342	.432
Pan			.286	.345	.429
Bos				.419	.422
Erinac.	$\hat{\alpha}_2\Delta_2$				.446
Gorilla	.0087	.0099	.284	.360	.424
Homo		.0069	.280	.357	.420
Pan			.282	.358	.422
Bos				.390	.454
Erinac.	$\widehat{D}$				.447

Figure 8 – **Changing behaviour of ERaBLE with different contraints.** In this example, ERaBLE is run on the two distance matrices Δ_1 and Δ_2 on the left. Setting $Z_k = 1$ results in the $\hat{\alpha}_k$ values and matrices $\hat{\alpha}_1\Delta_1, \hat{\alpha}_2\Delta_2$ and $\widehat{D} = (\hat{d}_{ij})$ in the middle column. Our chosen setting for Z_k results in more reasonable values for these quantities (right column), as explained in the text.

in significantly smaller differences $\hat{\alpha}_2\delta_{ij}^{(2)} - \hat{d}_{ij}$ for most distances, the only exceptions being the three distances between primates. Note that the scale of $\hat{\alpha}_2\Delta_2$ cannot be reduced indefinitely, as then the scale of $\hat{\alpha}_1\Delta_1$ becomes too large, and the fit of $\widehat{D}$ with the distances between primates in the two rescaled distance matrices becomes too loose (note that for $Z_k = 1$ the distances between primates in $\hat{\alpha}_1\Delta_1$ and $\hat{\alpha}_2\Delta_2$ already differ by an order of magnitude and the fit with $\widehat{D}$ is very poor).

The, admittedly heuristic, approach that we have adopted in our experiments, that is, setting $Z_k = N_k \sum_{i,j \in L_k} \delta_{ij}^{(k)}$, essentially prevents the genes only appearing in few and closely related taxa from having an influence on the constraint. Thus, it is the $\hat{\alpha}_k$ for the remaining genes that are constrained to have a weighted average of 1 (where the weight depends on the length of their sequence, as in equation (5)). As a consequence, these $\hat{\alpha}_k$ cannot be reduced together with $\widehat{D}$, as we showed for $Z_k = 1$. In our example, the new constraint is $24 \cdot \hat{\alpha}_1 + 3838 \cdot \hat{\alpha}_2 = 3862$, which is roughly equivalent to imposing $\hat{\alpha}_2 = 1$. The results are then much more realistic than with $Z_k = 1$: for example the rescaled matrices $\hat{\alpha}_1\Delta_1$ and $\hat{\alpha}_2\Delta_2$ are now much closer on their common entries (right column in Fig. 8).

Additional file 3 — Estimation accuracy measures

We adopted two different measures of estimation accuracy, one for branch lengths and the other for gene rates.

- The fraction of variance unexplained is a classical measure in regression analysis which can be seen as a normalised form of the sum of squared errors of prediction of an estimation model. Here, we adapt it to measure the discrepancy between a vector of branch lengths (b_e) and their estimates $(\hat{b}_e)$. We define the *fraction of variance unexplained* of (b_e) relative to $(\hat{b}_e)$ as:

$$\frac{\sum_e (b_e - \hat{b}_e)^2}{\sum_e (b_e - \bar{b})^2}$$

  where sums are over all branches in a tree and $\bar{b}$ is the arithmetic mean of the branch lengths in (b_e).
- In order to measure the discrepancy between a collection of m gene rates (r_k) and their estimates $(\hat{r}_k)$, we take the *mean absolute log-ratio* between r_k and $\hat{r}_k$, that is:

$$\frac{1}{m} \sum_{k=1}^{m} \left| \log \frac{\hat{r}_k}{r_k} \right| .$$

Additional file 4 — The scales of branch lengths for the OrthoMaM data set

The OrthoMaM data set displays an inverse correlation between the rate r_k of a gene and the depth of its alignment (its "coverage", i.e., $|L_k|$ in our notation), as is clearly shown in Fig. 9. This is not surprising — it is expected that genes evolving more slowly are easier to sample and annotate in many taxa — and we thus expect most real data sets to display the same correlation, to varying degrees.

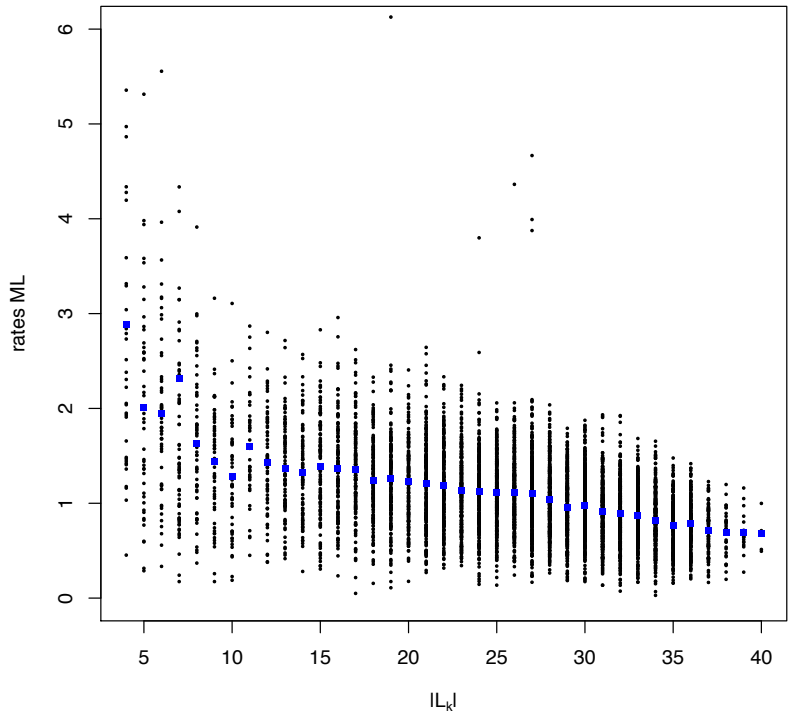

Figure 9 – **Correlation between the estimated rate of a gene and its alignment depth in the OrthoMaM data set.** For each of the 6,953 genes in the OrthoMaM data set, the number of sequences in its alignment (i.e., $|L_k|$, x-axis) is plotted against the rate estimate $\hat{r}_k$ produced by Concat+ML (y-axis). The blue squares represent the means of $\hat{r}_k$, for all genes with a fixed value of $|L_k| \in \{4, 5, \ldots, 40\}$.

This, however, poses a problem regarding the scale of the results. In loose terms, the problem is the following: *all other things being equal, should genes with high coverage influence more the scale of branch length estimates than genes with low coverage?* Note that the answer to this question only becomes relevant in data sets, such as OrthoMaM, where there is a correlation between coverage and rates: if, as realistic, genes sampled in a greater number of taxa tend to have lower rates, then answering *yes* to this question will result in shorter branch length estimates, than methods which implicitly answer *no* to it.

Close inspection of the methods in our experiments reveals that the answer to this question is *no* for the supertree and medium-level methods we tested, and *yes* for the superalignment methods. For ERaBLE-based and SDM-based methods, this is caused by the rescaling they apply to their estimates, which sets a scale that is determined by all genes in proportion to their lengths (Eqn. (5)), but which is independent of gene coverage.

As a result, the superalignment methods we tested tend to produce shorter branch length estimates than the supertree and medium-level methods we tested, which is precisely what we observe in Fig. 4. This explanation can also be confirmed by simulating data with an inverse correlation between $|L_k|$ and r_k, where similar differences in scales between the methods tested can be observed (not shown).

Additional file 5 — Reference topology for the OrthoMaM data set

For the OrthoMaM data set, the reference topology follows the Atlantogenata hypothesis (e.g., [Morgan et al. 2013]) and the Laurasiatheria intra-order placements of Romiguier et al. [Romiguier et al. 2013]. In Newick format this topology is given by:

```
(Ornithorhynchus,(((((((((Mus,Rattus),Dipodomys),Cavia),Ictidomys),(Ochotona,Oryctolagus)),
(((Otolemur,Microcebus),(Tarsius,(Callithrix,(Macaca,(Nomascus,(Pongo,(Gorilla,(Pan,Homo)))
))))),Tupaia)),(((((Felis,(Canis,(Ailuropoda,Mustela))),Equus),(Myotis,Pteropus)),
(((Bos,Tursiops),Sus),Vicugna)),(Sorex,Erinaceus))),(((Loxodonta,Procavia),Echinops),
(Dasypus,Choloepus))),(Monodelphis,(Macropus,Sarcophilus))));
```

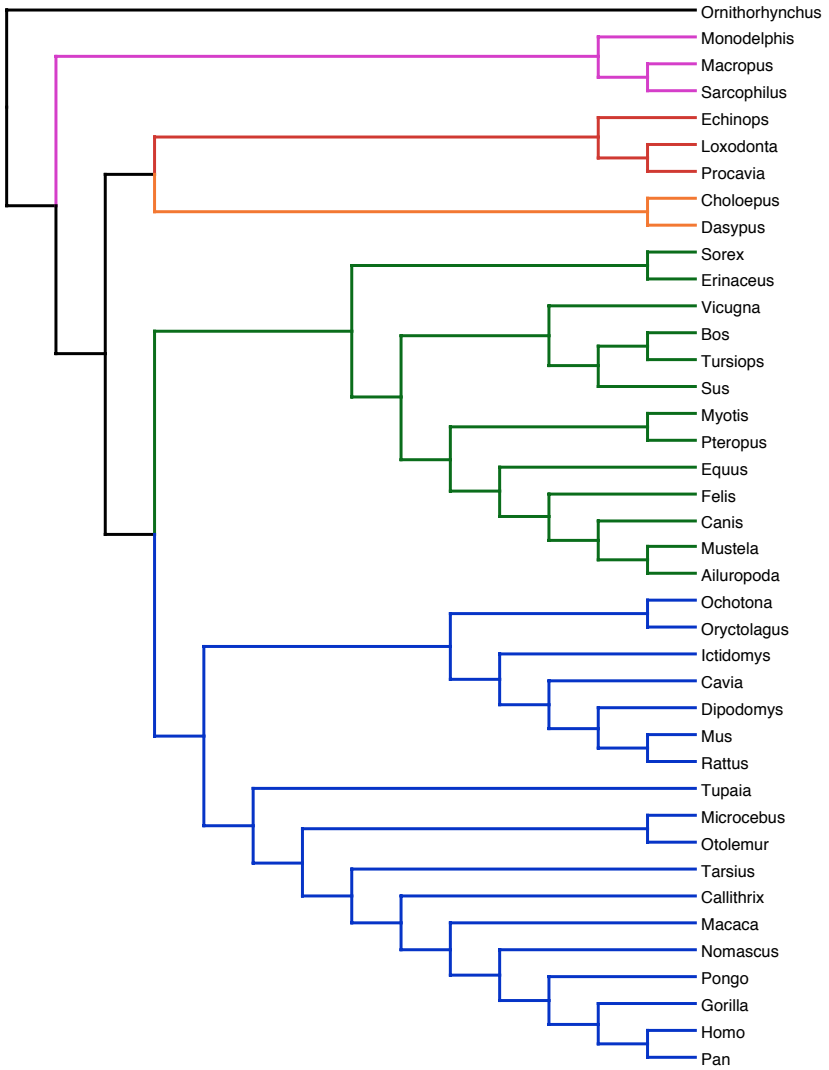

Figure 10 – **OrthoMaM data set reference topology.**

[Morgan et al. 2013] C. C. Morgan, P. G. Foster, A. E. Webb, D. Pisani, J. O. McInerney, and M. J. O'Connell, "Heterogeneous models place the root of the placental mammal phylogeny," *Molecular biology and evolution*, vol. 30, no. 9, pp. 2145–2156, 2013.

[Romiguier et al. 2013] J. Romiguier, V. Ranwez, F. Delsuc, N. Galtier, and E. J. Douzery, "Less is more in mammalian phylogenomics: At-rich genes minimize tree conflicts and unravel the root of placental mammals," *Molecular biology and evolution*, vol. 30, no. 9, pp. 2134–2144, 2013.

Additional file 6 — Usefulness of gene-specific Gamma shape parameters

Here we show the results of repeating our experiments on the OrthoMaM data set with gene-specific Gamma shape parameters in the estimation of the distance matrices $\Delta_1, \Delta_2, \ldots, \Delta_m$ (namely, for Δ_k we use the shape parameter estimated by PhyML when inferring $\hat{T}_k$). Note that doing so has a positive effect on the accuracy of medium-level methods at estimating both branch lengths (Fig. 4 bis) and gene rates (Fig. 5 bis). In general such gene-specific Gamma shape parameters will not be available to users of medium-level approaches, as they cannot be inferred from pairwise sequence comparisons only. This is why our experiments set these parameters to a fixed value.

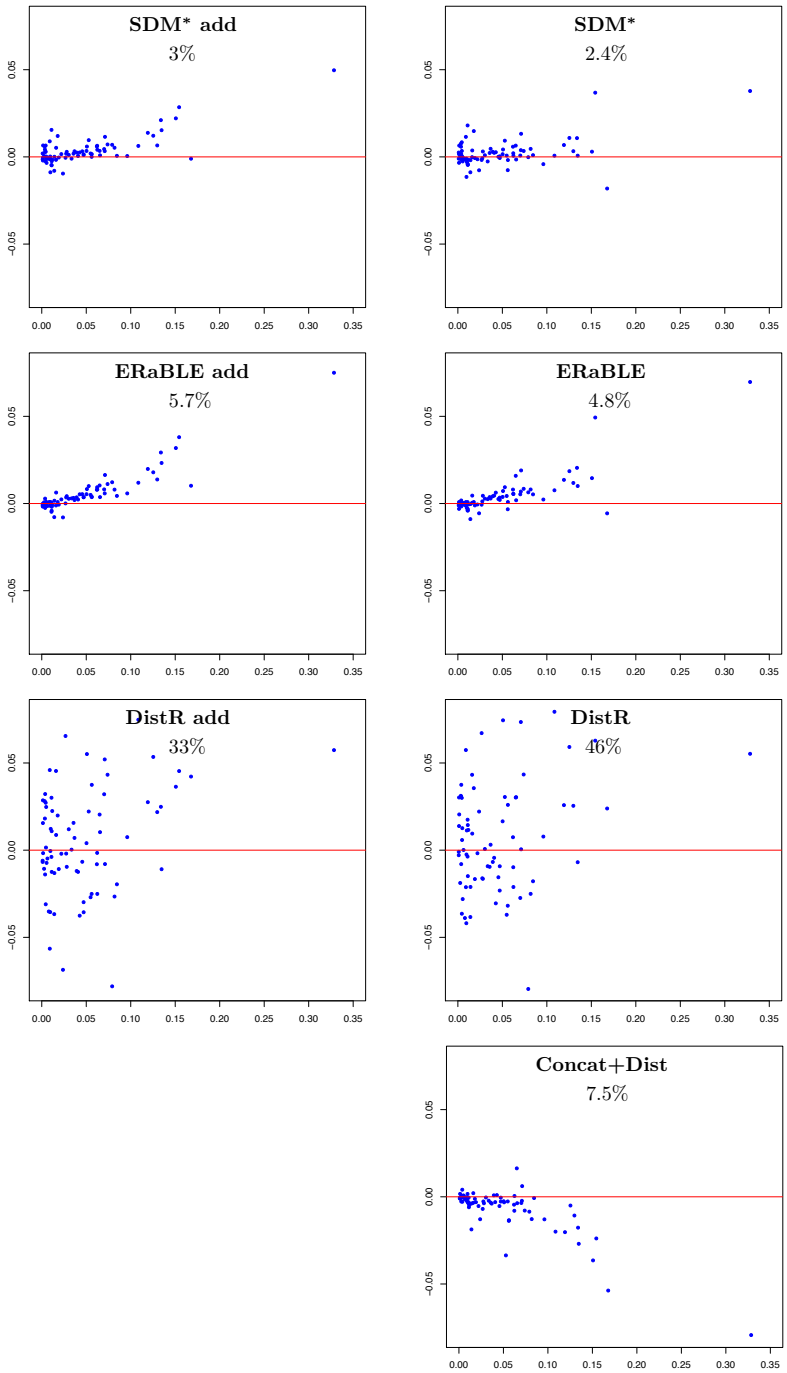

Figure 4 bis – **Accuracy of branch length estimates in the OrthoMaM data set.** Same as Fig. 4 in the main text, but here the Gamma shape parameter used for the estimation of Δ_k is set to the value estimated by PhyML when inferring $\hat{T}_k$. (Whereas in Fig. 4 the Gamma shape parameter is set to 0.5.)

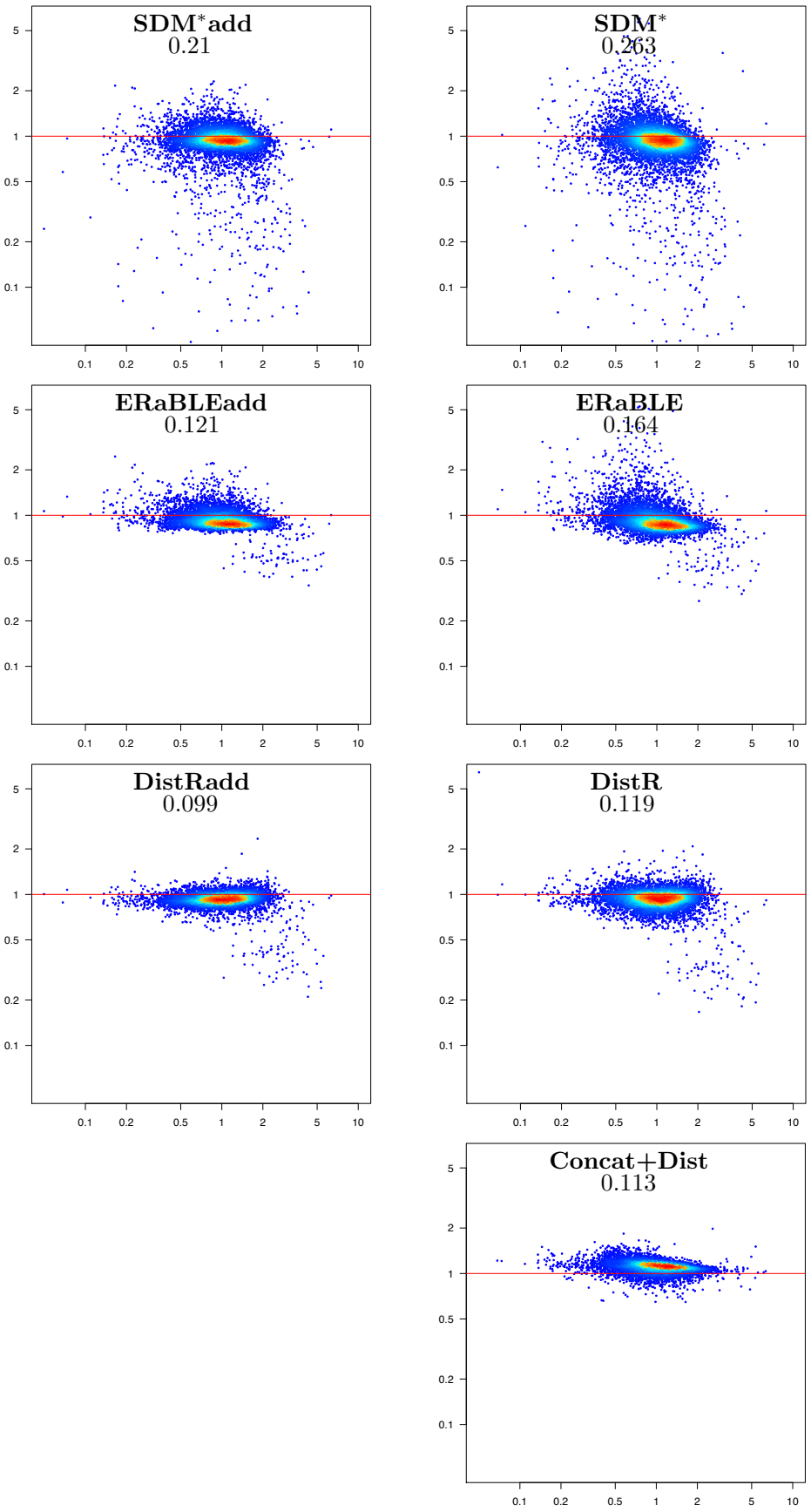

Figure 5 bis – **Estimation accuracy for gene rates in the OrthoMaM data set.** Same as Fig. 5 in the main text, but here the Gamma shape parameter used for the estimation of Δ_k is set to the value estimated by PhyML when inferring $\hat{T}_k$. (Whereas in Fig. 5 the Gamma shape parameter is set to 0.5.)

Additional file 7 — Alternative ML methods for Concat+ML

Here we show the results for the estimation of branch length in the simulated data set and in the OrthoMaM data set with alternative ML methods for the pipeline Concat+ML. We recall that Concat+ML involves assigning branch lengths to the reference topology $\mathcal{T}$ by running topology-constrained PhyML on the superalignment (concatenate), with the model TN93+Γ_8. The PhyML alternatives we have considered are ExaML [52] and FastTree 2 [53]. These methods are more computationally efficient than PhyML, but support a narrower range of models of evolution. We ran ExaML with the GTR+Γ_4 model to assign branch lengths to the reference topology. The model and number of categories in the discrete Gamma distribution are not modifiable in ExaML. We call this pipeline Concat+ExaML. We ran FastTree 2 with the GTR+CAT model with the gamma option and call this pipeline Concat+FastTree. In both cases the topology is constrained to be $\mathcal{T}$. Fig. 4 ter shows the accuracy of these pipelines in the estimation of the branch lengths in the simulated data set and in the OrthoMaM data set. Table 3 gives their running times and memory usage.

In Fig. 4 ter, we observe that Concat+FastTree tends to overestimate short branch lengths and strongly underestimate long branch lengths. We cannot explain this bias at the moment. Concat+ExaML is slightly less accurate than Concat+ML in the estimation of branch lengths for the simulated data set. This may be explained by the different substitution model employed by Concat+ExaML. As expected, Concat+ExaML and Concat+FastTree methods have a reduced computational cost in time and memory in comparison with Concat+ML, but still relatively high, when compared to ERaBLE (Table 3).

Table 3 – **Computational efficiencies on the OrthoMaM data set for the tested methods.**

	Concat+ML	**Concat+ExaML**	**Concat+FastTree**	**ERaBLE**
Time	41h16m	14h20m	3h42m	7s
Memory	117 GB	15.4 GB	41.1 GB	221 MB

NOTE.— The first row gives the time to obtain estimates for branch lengths. The second row gives the maximum amount of memory allocated. All the experiments were conducted on a cluster machine with 200 GB RAM and a 2.66 GHz CPU because of the large memory requirements, except for ERaBLE whiwh was run on a standard PC with 4 GB RAM and a 2.7 GHz CPU.

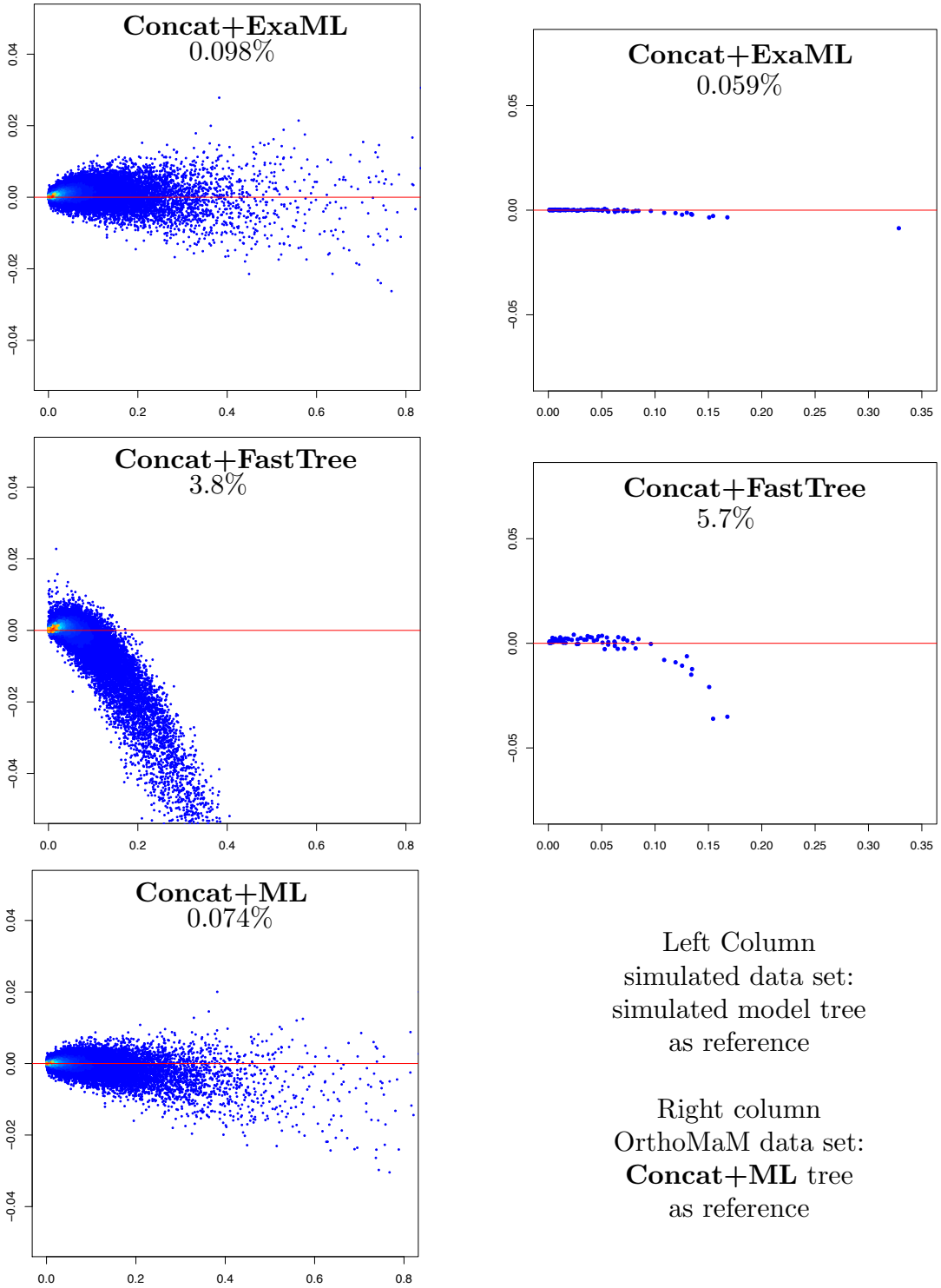

Figure 4 ter – **Accuracy of branch length estimates. Left column: accuracy for the simulated data set, right column: accuracy for the OrthoMaM data set.** For each method, the reference branch lengths b_e (x-axis) are plotted against the differences $\hat{b}_e - b_e$ (y-axis) (where $\hat{b}_e$ is the estimate for the length of e obtained by the method at the top of the plot). The horizontal red line corresponds to no difference between the two estimates. Method names are shown at the top of each plot, followed by the fraction of variance unexplained of (b_e) relative to $(\hat{b}_e)$. For the simulated data set, reference branch lengths are those of the 500 model trees. For the OrthoMaM data set, reference branch lengths are those estimated by Concat+ML on the reference topology. Colors (from blue to red) indicate increased density of points. Fore more detail, compare the left column with Fig. 2 and the right column with Fig. 4 in the main text.