Reliable confidence intervals for RelTime estimates of evolutionary divergence times

Qiqing Tao[1,2,§], Koichiro Tamura[3,4,§], Beatriz Mello[5], and Sudhir Kumar[1,2,*]

[1]Institute for Genomics and Evolutionary Medicine, Temple University, Philadelphia, Pennsylvania

[2]Department of Biology, Temple University, Philadelphia, Pennsylvania

[3]Department of Biological Sciences, Tokyo Metropolitan University, Hachioji, Tokyo, Japan

[4]Research Center for Genomics and Bioinformatics, Tokyo Metropolitan University, Hachioji, Tokyo, Japan

[5]Department of Genetics, Federal University of Rio de Janeiro, RJ, Brazil

[§] Authors contributed equally to this work.

Correspondence to:
Sudhir Kumar
Temple University
Philadelphia, PA 19122, USA
E-mail: s.kumar@temple.edu

Abstract

Confidence intervals (CIs) depict the statistical uncertainty surrounding evolutionary divergence time estimates. They capture variance contributed by the finite number of sequences and sites used in the alignment, deviations of evolutionary rates from a strict molecular clock in a phylogeny, and uncertainty associated with clock calibrations. Reliable tests of biological hypotheses demand reliable CIs. However, current non-Bayesian methods may produce unreliable CIs, because they do not incorporate rate variation among lineages and interactions among clock calibrations. Here, we present a new analytical method to calculate CIs of divergence times estimated using the RelTime method, along with an approach to utilize multiple calibration uncertainty densities in these analyses. Through empirical data analysis, we show that the new methods produce CIs that overlap with Bayesian highest posterior density (HPD) intervals. These developments will encourage broader use of computationally efficient, non-Bayesian relaxed clock approaches in molecular dating analyses and biological hypothesis testing.

Introduction

Reliable inference of the confidence intervals around the estimates of divergence times is essential for testing biological hypotheses (Burbrink and Pyron 2008; Kumar and Hedges 2016). Multiple sources contribute to the uncertainty of molecular divergence time estimates (Rannala and Yang 2007; Zhu et al. 2015; Kumar and Hedges 2016). One of them is the error associated with branch length estimation in a phylogeny due to the limited number of sites and substitutions in the sequence alignment (Kumar and Hedges 2016; Warnock et al. 2017). The stochastic nature of the substitution process (e.g., Poisson process) and the uncertainty involved in accounting for the unobserved substitutions (multiple-hit correction) result in errors in branch length estimates, which lead to imprecise time estimates (Kumar and Hedges 2016). However, this error decreases with an increase in the number of sampled sites (Rannala and Yang 2007; dos Reis and Yang 2013; Zhu et al. 2015) and becomes negligible for large phylogenomic datasets. A second source of error is the variation of evolutionary rates among branches and lineages (Zhu et al. 2015; Kumar and Hedges 2016). Because rates and times are confounded, the variation of rates will naturally result in uncertainty of time estimates (Ho 2014; Zhu et al. 2015). This confounding effect cannot be eliminated by sampling more sites or genes in a dataset (Zhu et al. 2015; Kumar and Hedges 2016), so it contributes more uncertainty to time estimates than errors in branch length estimation for a large dataset. The uncertainty associated with clock calibrations due to the equivocal nature of fossil record presents a third source of error in divergence time estimation (Zhu et al. 2015; dos Reis et al. 2016; Warnock et al. 2017). The exact placement of fossil record in a phylogeny and the correct assignment of calibration constraints, especially the maximum constraint, are often difficult to justify, resulting in great uncertainty in estimation of divergence time (Bromham et al. 2018).

In Bayesian analyses, the highest posterior density (HPD) intervals usually represent the uncertainty of inferred divergence times (Drummond et al. 2006). Bayesian methods compute HPD intervals directly from the density distribution of posterior times estimated using priors for rate heterogeneity, substitution process and fossil calibrations (dos Reis et al. 2016; Bromham et al. 2018), so sources contributing to the uncertainties of time estimates are automatically incorporated in the HPD intervals. Currently, Bayesian HPDs are considered reliable estimates of uncertainties surrounding divergence time estimates. However, the large computation burden imposed by Bayesian approaches has hindered their applications to analyze many phylogenomic datasets (Pyron 2014; Mello et al. 2017; Li et al. 2019).

In contrast, non-Bayesian methods can analyze large-scale datasets quickly and generate accurate time estimates (Smith and O'Meara 2012; Tamura et al. 2012; Tamura et al. 2018). Unfortunately, the broad utility of these methods is still reduced by lack of reliable calculation of the uncertainty surrounding divergence times, which are represented by confidence intervals (CIs). Non-Bayesian approaches require the use of analytical formulations or bootstrap approaches to estimate CIs (Sanderson 2003; Xia and Yang 2011; Tamura et al. 2013). However, site-resampling bootstrap approaches do not capture the error caused by rate heterogeneity, leading to false precisions of time estimates. Recognizing the need for incorporating lineage rate variation into CIs, Tamura et al. (2013) formulated analytical equations for the RelTime method, a non-Bayesian approach that relaxes the molecular clock. However, this approach may overestimate the amount of variance and produce overly wide confidence intervals (see below), resulting in low power for statistical testing (Kumar and Hedges 2016).

Bayesian and non-Bayesian methods also use different strategies to account for the uncertainty of fossil record. Non-Bayesian methods are limited to the use of minimum boundaries only, maximum boundaries only, or minimum and maximum boundary pairs as calibration constraints (Sanderson 2003; Tamura et al. 2013), while Bayesian methods allow the usage of informative probability densities as calibrations and automatically accommodate interactions among them (Inoue et al. 2010; Ho and Duchêne 2014). Mello et al. (2017) presented a simple procedure to derive minimum and maximum boundaries from the density distributions, but this strategy does not consider interactions among calibrations and may lead to overestimates of the variance of divergence times (see below).

Here, we present an analytical approach to estimate CIs for divergence times estimated with RelTime, accounting for variance associated with the branch lengths estimation and variance due to rate heterogeneity. We also present a simple approach to derive minimum and maximum boundaries from multiple calibration densities and accommodate calibration interactions. Both methods have been implemented in MEGA X for use in graphical and command line interfaces (Kumar et al. 2012; Kumar et al. 2018). The 95% CIs produced by RelTime in empirical analyses are compared with the 95% HPD intervals produced by Bayesian methods to examine the performance of the new approaches. The approach presented here may be used to improve variance calculation of time estimates for other non-Bayesian methods, e.g., penalized likelihood methods (Sanderson 2003; Smith and O'Meara 2012).

New Methods

An analytical method to estimate confidence intervals

Considering a tree with three ingroup sequences (**Fig. 1**), relative time (t) for each node and relative rate (r) for each lineage are functions of branch lengths (b) in RelTime, e.g., r_1, r_2, r_3, r_4, t_4, and t_5 are given by the following equations when the geometric means are used (similar equations can be derived when the arithmetic mean is used) (Tamura et al. 2018):

$$r_1 = \sqrt{b_1}\sqrt{\sqrt{b_1 b_2} + b_4}/\sqrt{b_2 b_3}, \qquad\qquad \text{[eq. 1]}$$

$$r_2 = \sqrt{b_2}\sqrt{\sqrt{b_1 b_2} + b_4}/\sqrt{b_1 b_3}, \qquad\qquad \text{[eq. 2]}$$

$$r_3 = \sqrt{b_3}/\sqrt{\sqrt{b_1 b_2} + b_4}, \qquad\qquad \text{[eq. 3]}$$

$$r_4 = \sqrt{\sqrt{b_1 b_2} + b_4}/\sqrt{b_3}, \qquad\qquad \text{[eq. 4]}$$

$$t_4 = \sqrt{b_1 b_2 b_3}/\sqrt{\sqrt{b_1 b_2} + b_4}, \qquad\qquad \text{[eq. 5]}$$

$$t_5 = \sqrt{b_3}\sqrt{\sqrt{b_1 b_2} + b_4}. \qquad\qquad \text{[eq. 6]}$$

The variance of the estimated time (t_i) for node i, denoted by $v(t_i)$, can be estimated by the delta method, assuming that there is no covariance among branch lengths (b_j's):

$$v(t_i) = \sum_j \left(\frac{\partial f_{t_i}(b_1,\ldots,b_N)}{\partial b_j}\right)^2 v(b_j), \qquad\qquad \text{[eq. 7]}$$

where $f_{t_i}(b_1,\ldots,b_N)$ stands for the analytical function of b_j's to compute t_i (e.g., eq. 5 and eq. 6 for t_4 and t_5, respectively), and $v(b_j)$ stands for the variance of branch length for branch j. Therefore, $v(b_j)$ is required for computing $v(t_i)$.

As mentioned before, the uncertainty of time is related to the number of sampling sites and the degree of rate heterogeneity. We consider the total variance of branch lengths, $v(b_j)$, which is required to compute $v(t_i)$, as a summation of the variance due to site sampling, $v_S(b_j)$, and the variance due to rate heterogeneity, $v_R(b_j)$:

$$v(b_j) = v_S(b_j) + v_R(b_j). \qquad\qquad \text{[eq. 8]}$$

5

The value of $v_S(b_j)$ can be estimated by using analytical formulations or a site-resampling approach. For example, an approximate estimate of this variance can be obtained by the curvature method when the maximum likelihood method is used (Edwards 1992; Tamura et al. 2013).

However, it is more complex to estimate $v_R(b_j)$, so we do it indirectly. We first compute the variance of observed evolutionary rates for all the lineages, $V_{obs}(R)$:

$$V_{obs}(R) = (1/N) \sum_j^N (r_j - \bar{r})^2, \qquad \text{[eq. 9]}$$

where R is a random variable representing all relative rates, r_j is the relative rate for each branch j, and $\bar{r}$ is the average of r_j's. It is important to note that the relative rate for branch j is estimated as the relative rate for lineage j (Tamura et al. 2018). For example, RelTime computes the relative rate for b_4 as the geometric mean of r_1 and r_2, which is assigned to be the rate for lineage l_4 in **Figure 1**.

The variance of observed rates includes not only the variance introduced by rate heterogeneity, $R_V(R)$, but also the sampling variance associated with the branch length estimation, $S_V(R)$, because the observed relative rate r_j is calculated from branch lengths (b_j's) (e.g., equations 1 - 4). So,

$$V_{obs}(R) = R_V(R) + S_V(R), \qquad \text{[eq. 10]}$$

The value of $S_V(R)$ is obtained by summing the sampling variance of relative rate r_j for each branch j, denoted by $s_v(r_j)$:

$$S_V(R) = \sum_j^N s_v(r_j). \qquad \text{[eq. 11]}$$

$s_v(r_j)$ can be estimated by the delta method, assuming that there is no covariance among b_j's:

$$s_v(r_j) = \sum_j^N \left(\frac{\partial f_{r_j}(b_1,...,b_N)}{\partial b_j} \right)^2 v_S(b_j), \qquad \text{[eq. 12]}$$

where $f_{r_j}(b_1,...,b_N)$ stands for the analytical function of b_j's to compute r_j (e.g., equation 1, 2, 3 and 4 for r_1, r_2, r_3, and r_4, respectively).

Using equations 9 – 12, we compute the variance introduced by rate heterogeneity:

$$R_V(R) = \frac{1}{N} \sum_j^N (r_j - \bar{r})^2 - \sum_j^N \sum_j^N \left(\frac{\partial f_{r_j}(b_1,...,b_N)}{\partial b_j} \right)^2 v_S(b_j). \qquad \text{[eq. 13]}$$

Then, we can compute the rate heterogeneity variance for each branch j as being proportional to its branch length:

$$v_R(b_j) = \frac{b_j^2}{\Sigma_j^N b_j^2} R_V(R).$$

[eq. 14]

Using equations 8, 13 and 14, we can compute the total variance of branch length for branch j, denoted by $v(b_j)$. Then $v(b_j)$ can be used to compute the variance of time, $v(t_i)$, using equation 7. For example, the variance of t_4 and t_5 are given by the following equations:

$$v(t_4) = \frac{b_2 b_3 \left(\sqrt{b_1 b_2} + 2b_4\right)^2}{16 b_1 \left(\sqrt{b_1 b_2} + b_4\right)^3} v(b_1) + \frac{b_1 b_3 \left(\sqrt{b_1 b_2} + 2b_4\right)^2}{16 b_2 \left(\sqrt{b_1 b_2} + b_4\right)^3} v(b_2)$$

$$+ \frac{b_1 b_2}{4 b_3 \left(\sqrt{b_1 b_2} + b_4\right)} v(b_3) + \frac{b_1 b_2 b_3}{4 \left(\sqrt{b_1 b_2} + b_4\right)^3} v(b_4),$$

[eq. 15]

$$v(t_5) = \frac{b_2 b_3}{16 b_1 \left(\sqrt{b_1 b_2} + b_4\right)} v(b_1) + \frac{b_1 b_3}{16 b_2 \left(\sqrt{b_1 b_2} + b_4\right)} v(b_2)$$

$$+ \frac{\sqrt{b_1 b_2} + b_4}{4 b_3} v(b_3) + \frac{b_3}{4 \left(\sqrt{b_1 b_2} + b_4\right)} v(b_4).$$

[eq. 16]

For larger numbers of taxa, such analytical formulations become complicated to derive, especially for deeper nodes. Thus, we compute the variance of divergence times for deeper nodes from tips to the root recursively. For example, using equations 15 and 16, we can derive

$$v(t_5) = \frac{\left(\sqrt{b_1 b_2} + b_4\right)^2}{\left(\sqrt{b_1 b_2} + 2b_4\right)^2} \left[v(t_4) + \frac{b_4}{b_3} v(b_3) + \frac{b_3 b_4}{\left(\sqrt{b_1 b_2} + b_4\right)^2} v(b_4) \right].$$

[eq. 17]

Therefore, the calculation of $v(t_5)$ requires only $v(t_4)$, $v(b_3)$ and $v(b_4)$, which are the variance for node t_4 and branches b_3 and b_4, respectively. Variance of branches that do not directly connect to node 5, i.e., $v(b_1)$ and $v(b_2)$ in this case (**Fig. 1**) is not needed, if the value of $v(t_4)$ is computed beforehand. Thus, for any node in a phylogeny, we can compute the variance of divergence time recursively from tips to the root by using the variance of times for direct descent and ancestral nodes and the variance of direct connected branches. This procedure extraordinarily simplifies the computation of the variance of inferred time for each internal node in a tree with a large number of taxa.

It is important to note that times in the equations listed above are relative times, not absolute times because no calibrations are involved in the above equations. When one or multiple calibrations (minimum boundaries only, maximum boundaries only, or minimum and maximum boundary pairs) are given, RelTime will compute a global time factor (*f*) by altering relative times

7

such that all calibration constraints are satisfied. When a range of *f* values can satisfy all calibration constraints, RelTime selects the midpoint of the range to be the best estimate of *f*. When one or more of the absolute times computed using the *f* value falls outside the calibration constraints, then RelTime adjusts relative times and *f* such that the deviations of absolute times from the calibration constraints are minimized. This process requires local alteration of relative rates and re-optimization of all other node times in the tree recursively (Tamura et al. 2013). For example, if the minimum age constraint of a node is violated, i.e., the age estimated using *f* is younger than the minimum constraint, RelTime decrease its estimate of the evolutionary rate proportionally in that lineage to adjust the age of this node to be higher, such that the divergence time becomes the same as the minimum age constraint. The resulting slowdown is transmitted to all the descendant nodes, and it affects the ancestral rates as well.

Similarly, if the maximum age constraint of a node is violated, i.e., the age estimated using *f* is older than the maximum constraint, RelTime increases the estimated evolutionary rate proportionally in that lineage such that the divergence time matches the maximum age constraint. Effects of this rate change will be transmitted to the descendant and ancestral nodes automatically. Consequently, RelTime will ensure that the absolute times for calibrated nodes are consistent with the user-desired calibration constraints. In the final step, CIs are computed analytically using the final set of relative rates and the equations given above (e.g., equations 13-17), such that the uncertainty associated with clock calibrations can be incorporated into the CI calculation in RelTime. If the lower or upper bounds of CIs fall outside the user-specified calibration constraints, then CIs are truncated based on the imposed calibration constraints. Therefore, RelTime uses "hard" minimum and maximum bounds, as in BEAST (Bouckaert et al. 2014; Barba-Montoya et al. 2017).

A method to derive effective calibration boundaries from calibration densities

As stated above, calibration uncertainty is another critical source of uncertainties in divergence time estimates. Bayesian methods use various probability densities to accommodate the calibration uncertainty. However, the current non-Bayesian methods do not allow direct usage of probability densities and do not incorporate interactions among calibration constraints. Therefore, we developed a new procedure for use in the RelTime method to derive calibration boundaries from probability densities and account for calibration interactions.

For each calibrated node with an associated probability density, we randomly sample two dates from the given probability density. We use these two sampled dates as the minimum and

8

maximum (min-max) constraints for that node and derive such a min-max constraint for every node for which a probability density is specified. Then, we use all of these min-max boundaries in the RelTime analysis. We save the RelTime time estimates only for the calibrated nodes. This random sampling and dating process is repeated 10,000 times. Therefore, we will have generated a distribution of 10,000 inferred dates for each calibrated node. In the final step, we derive the minimum bound at 2.5% and the maximum bound at 97.5% of the distribution of inferred dates for each calibrated node. We refer to these derived bounds as "effective bounds." These effective bounds can be used together with the analytical approach described above to infer the divergence times and CIs in RelTime. It is important to note that effective bounds are used as calibration constraints, not densities. The actual distribution of 10,000 inferred dates does not impact the value of effective bounds and therefore, the final RelTime estimates of divergence times and CIs.

Our procedure is analogous to how Bayesian methods utilize calibrations, as both types of methods require resampling different sets of calibration constraints from user-specified densities, inference of divergence times using each set of sampled calibrations, and summarization of distributions of time estimates obtained from all sets of sampled calibrations. Therefore, the use of effective bounds allows RelTime to accommodate the interactions among calibration densities. However, it does not mean that RelTime and Bayesian methods are the same. Bayesian methods conduct calibration resampling and time inference steps simultaneously during the MCMC integration, whereas these steps are implemented sequentially in the RelTime method.

We compare the effective bounds to calibration bounds derived using Mello et al. (2017)'s procedure (referenced as "Mello bounds" in the following) (**Fig. 2**), in which the minimum bound was placed at 2.5% of the density age, and the maximum bound was placed at 97.5% of the density age. Effective bounds are similar to the Mello bounds when the user-specified calibration density is reliable and informative (**Fig. 2b,** see ***Materials and Methods***). When the user-specified density is uninformative, e.g., a diffused uniform distribution, Mello bounds are often diffused and match the original density (**Fig. 2c**). However, our procedure generates narrower bounds due to the accommodation of the interactions among different calibration densities and constraints. These interactions reshape the original, wider distribution to be tighter (**Fig. 2c**). Consequently, the use of effective bounds is likely to produce more precise time estimates (narrower CIs). When the user-specified calibration is unreliable, our effective bounds turn out to be better than Mello bounds. For example, if the true time is located in the user-specified density with a low probability (< 2.5%), Mello bounds may not include the true time (**Fig. 2d**), resulting in

9

incorrect time estimates. In contrast, our method does not ignore the low probability regions since it samples 10,000 times from the user-specified density to ensure that dates with very low probabilities may be sampled. Thus, effective bounds should contain the true time (**Fig. 2d**), and the use of effective bounds in RelTime should improve the accuracy and precision of time estimates.

Results and Discussion

We applied our methods to three empirical datasets, including nucleotide or protein sequences from primates, spiders, and insects (**Table 1**). We begin with the primate dataset from Barba-Montoya et al. (2017), which contains a relatively small alignment of 9,361 base pairs from nine primate species and one outgroup (**Fig. 2a**). However, all internal nodes were assigned with calibration densities. Barba-Montoya et al. (2017) used two calibration strategies in MCMCTree (Yang 2007) and BEAST (Bouckaert et al. 2014) and compared the results. We examined if the RelTime method produced estimates comparable to those obtained from Bayesian methods, when all analyses employed the same alignment, phylogeny, substitution model, and calibration uncertainty densities (e.g., uniform distributions).

For analyses of primate datasets where uniform densities were used as calibrations, we observed a high concordance between RelTime and Bayesian time estimates. The linear regression slopes were 0.97 and 1.03 when Bayesian analyses were conducted in MCMCTree and BEAST, respectively (**Fig. 3a** and **b**). This a rather small difference. Although the width of RelTime CIs was slightly smaller than the width of Bayesian HPD intervals, RelTime CIs overlapped Bayesian HPD intervals for all the nodes (**Fig. 4a** and **b**). For primate datasets where a mixture of uniform and skewed densities were used as calibrations, RelTime estimates were again similar to Bayesian estimates, with a linear regression slope of 0.95 with MCMCTree (**Fig. 3c**) and 1.00 with BEAST estimates (**Fig. 3d**). RelTime CIs overlapped with MCMCTree and BEAST HPD intervals for all the nodes (**Fig. 4c** and **d**).

We then analyzed spider and insect datasets to examine the performance of our methods for larger datasets (>40 species and >50,000 sites). These datasets consisted of protein sequences; 8 nodes had clock calibrations for the spider data, while 38 nodes of the insect phylogeny were assigned calibration values. We again observed strong concordance between RelTime and Bayesian time estimates, with a linear slope of 0.98 and 0.98 for spider and insect data, respectively (**Fig. 3e** and **f**). The high similarity between RelTime and Bayesian node times remained even after we excluded nodes on which user-specified calibrations were assigned

(slope = 0.97 and 0.98, respectively). CIs produced by RelTime were also comparable with HPD intervals produced by Bayesian methods, with more than 97% of the nodes in spider and insect datasets showing overlapping CIs and HPD intervals (**Fig. 4e** and **f**). When CIs and HPD intervals did not overlap, they were less than 5 million years apart. Therefore, we considered that RelTime CIs were similar to Bayesian HPD intervals for all datasets.

Our new methods allow RelTime to produce point time estimates and CIs similar to those inferred by Bayesian methods across all empirical analyses because our new methods effectively improve the CI calculation. We found that, on average, our analytical approach produced CIs that were 35% - 67% narrower than those produced by Tamura et al. (2013)'s method without any calibrations. However, time estimates remain very similar between these two methods with linear slopes ranging from 1.00 to 1.02. These results show that the use of the new analytical approach solely can effectively improve the precision of time estimates in RelTime while maintaining the same accuracy.

We also compared time estimates and CIs produced by the new analytical approach with effective bounds generated by our new procedure and calibration bounds generated by Mello et al. (2017)'s procedure. This comparison would illuminate how the use of effective bounds may improve the precision of divergence time estimates. Results showed that times estimated using two sets of calibration bounds were similar (linear slopes ranging from 0.99 to 1.02). However, the CI width was reduced by 7-29% when effective bounds were used. The decrease of CI widths occurred for both calibrated and uncalibrated nodes. It is because effective bounds reflect calibration interactions and reshape the original diffused calibration densities to generate narrower CIs, as discussed above. Therefore, both of our methods improve the precision of divergence time estimates produced by RelTime without sacrificing their accuracy.

It remains important to note that reliable time estimates and CIs rely strongly on the assumption that calibrations and their densities are correct. The incorrect specification of calibration constraints or densities can greatly impact the accuracy and precision of time estimates (Warnock et al. 2017). Our empirical analyses showed that RelTime and Bayesian methods generated similar estimates of divergence times and their surrounding uncertainties when the same alignment, phylogeny, and calibration densities were used. These patterns indicate that RelTime can serve as a reliable alternative to Bayesian approaches for dating the tree of life and conducting biological hypothesis testing, especially for large-scale molecular data, because RelTime is computationally efficient, requiring only a fraction of the time and resources demanded by Bayesian approaches (Tamura et al. 2012; Tamura et al. 2018). We also anticipate

that our method of deriving effective bounds from calibration densities can be applied to other non-Bayesian dating methods, e.g., penalized likelihood methods (Sanderson 2003; Smith and O'Meara 2012).

Materials and Methods

Comparisons of user-specified calibration density, Mello bounds, and effective bounds

We used the BEAST-generated primate timetree published in Barba-Montoya et al. (2017) as the true tree (**Fig. 2a**) and simulated an alignment of 9361 sites under HKY+G (Hasegawa et al. 1985) model in SeqGen with parameters derived from the empirical molecular data. Branch-specific rates were sampled from an uncorrelated lognormal distribution with a mean rate of 0.0069 substitutions per site per Ma and a standard deviation of 0.4 (log-scale). The simulated alignment was used to derive effective bounds.

We tested the performance of using effective bounds and Mello bounds under two calibration scenarios: reliable and unreliable scenarios. An informative exponential density was used at *homo sapiens – Callithrix jacchus* split (true age = 44.8Ma) and an uninformative uniform density were used at *homo sapiens – Otolemur gamettii* split (true age = 68Ma) under both scenarios. In the reliable calibration scenario, we assumed that a minimum age of 40Ma at *homo sapiens – Callithrix jacchus* split and maximum age of 130Ma at *homo sapiens – Otolemur gamettii* split are known. Therefore, we used an exponential density (mean = 4Ma and offset = 40Ma) and a uniform density (min = 40Ma, max = 130Ma) at *homo sapiens – Callithrix jacchus* split and at *homo sapiens – Otolemur gamettii* split, respectively. The true ages of both nodes located in their densities with high probabilities. Under the unreliable calibration scenario, we assumed that a minimum age of 30Ma at *homo sapiens – Callithrix jacchus* split and maximum age of 130Ma at *homo sapiens – Otolemur gamettii* split are known. Therefore, we used an exponential density (mean = 3Ma and offset = 30Ma) and a uniform density (min = 40Ma, max = 130Ma) at *homo sapiens – Callithrix jacchus* split and at *homo sapiens – Otolemur gamettii* split, respectively. This results in the true age of *homo sapiens – Callithrix jacchus* split located in its density with a low probability (< 2.5%), while the true age of *homo sapiens – Otolemur gamettii* split located in its density with a high probability.

Empirical analysis

We obtained four empirical datasets that employed different calibration strategies from three published studies (**Table 1**) (Bond et al. 2014; Tong et al. 2015; Barba-Montoya et al. 2017). Molecular data were obtained from supplementary files of original studies. Calibration densities and Bayesian timetrees (including credibility intervals) of Barba-Montoya et al. (2017) and Tong et al. (2015) were provided by authors. For Bond et al. (2014)'s data, we obtained the Bayesian timetree (including HPD intervals) from Mello et al. (2017). In RelTime analyses, we used the same alignments, substitution models, tree topologies, and calibration densities as the original studies to ensure comparability with Bayesian results. RelTime analyses were conducted in MEGA X (Kumar et al. 2018). We compared RelTime time estimates and CIs with Bayesian time estimates and HPD intervals. We did not test whether the slope between RelTime and Bayesian time estimates was one because of two reasons. First, p-value will always reject the hypothesis of slope of one when the data sample size is large. Second, the uncertainty surrounding time estimates prevents time estimates from two types of methods to be identical. We also compared the performance of our methods and the previous CI calculation method for RelTime. We first re-analyzed all empirical datasets using Tamura et al. (2013)'s method in MEGA 7 (Kumar et al. 2012; Kumar et al. 2016) and using our analytical method in MEGA X without calibrations. We then re-analyzed all empirical datasets in MEGA X using the effective calibration bounds generated by our new method and by Mello et al. (2017)'s procedure.

Acknowledgments

References

Barba-Montoya J, Dos Reis M, Yang Z. 2017. Comparison of different strategies for using fossil calibrations to generate the time prior in Bayesian molecular clock dating. Mol. Phylogenet. Evol. 114:386–400.

Bond JE, Garrison NL, Hamilton CA, Godwin RL, Hedin M, Agnarsson I. 2014. Phylogenomics resolves a spider backbone phylogeny and rejects a prevailing paradigm for orb web evolution. Curr. Biol. 24:1765–1771.

Bouckaert R, Heled J, Kühnert D, Vaughan T, Wu C-H, Xie D, Suchard MA, Rambaut A, Drummond AJ. 2014. BEAST 2: a software platform for Bayesian evolutionary analysis. PLoS Comput. Biol. 10:e1003537.

Bromham L, Duchêne S, Hua X, Ritchie AM, Duchêne DA, Ho SYW. 2018. Bayesian molecular dating: opening up the black box. Biol. Rev. Camb. Philos. Soc. 93:1165–1191.

Burbrink FT, Pyron RA. 2008. The taming of the skew: estimating proper confidence intervals for divergence dates. Syst. Biol. 57:317–328.

Dos Reis M, Donoghue PC, Yang Z. 2016. Bayesian molecular clock dating of species divergences in the genomics era. Nat. Rev. Genet. 17:71–80.

Dos Reis M, Yang Z. 2013. The unbearable uncertainty of Bayesian divergence time estimation. J. Syst. Evol. 51:30–43.

Drummond AJ, Ho SYW, Phillips MJ, Rambaut A. 2006. Relaxed phylogenetics and dating with confidence. PLoS Biol. 4:88–99.

Edwards AW. 1992. Likelihood. Baltimore: Johns Hopkins University Press

Hasegawa M, Kishino H, Yano T. 1985. Dating of the human-ape splitting by a molecular clock of mitochondrial DNA. J. Mol. Evol. 22:160–174.

Ho SY, Duchêne S. 2014. Molecular-clock methods for estimating evolutionary rates and timescales. Mol. Ecol. 23:5947–5965.

Ho SY. 2014. The changing face of the molecular evolutionary clock. Trends Ecol. Evol. 29:496–503.

Inoue J, Donoghue PCJ, Yang Z. 2010. The impact of the representation of fossil calibrations on Bayesian estimation of species divergence times. Syst. Biol. 59:74–89.

Kumar S, Hedges SB. 2016. Advances in time estimation methods for molecular data. Mol. Biol. Evol. 33:863–869.

Kumar S, Stecher G, Li M, Knyaz C, Tamura K. 2018. MEGA X: Molecular Evolutionary Genetics Analysis across Computing Platforms. Mol. Biol. Evol. 35:1547–1549.

Kumar S, Stecher G, Peterson D, Tamura K. 2012. MEGA-CC: computing core of molecular evolutionary genetics analysis program for automated and iterative data analysis.

Bioinformatics 28:2685–2686.

Kumar S, Stecher G, Tamura K. 2016. MEGA7: Molecular Evolutionary Genetics Analysis version 7.0 for bigger datasets. Mol. Biol. Evol. 33:1870–1874.

Li H-T, Yi T-S, Gao L-M, Ma P-F, Zhang T, Yang J-B, Gitzendanner MA, Fritsch PW, Cai J, Luo Y, et al. 2019. Origin of angiosperms and the puzzle of the Jurassic gap. Nat. Plants 5:461–470.

Mello B, Tao Q, Tamura K, Kumar S. 2017. Fast and Accurate Estimates of Divergence Times from Big Data. Mol. Biol. Evol. 34:45–50.

Pyron RA. 2014. Biogeographic analysis reveals ancient continental vicariance and recent oceanic dispersal in amphibians. Syst. Biol. 63:779–797.

Rannala B, Yang Z. 2007. Inferring speciation times under an episodic molecular clock. Syst. Biol. 56:453–466.

Sanderson MJ. 2003. r8s: inferring absolute rates of molecular evolution and divergence times in the absence of a molecular clock. Bioinformatics 19:301–302.

Smith SA, O'Meara BC. 2012. treePL: divergence time estimation using penalized likelihood for large phylogenies. Bioinformatics 28:2689–2690.

Tamura K, Battistuzzi FU, Billing-Ross P, Murillo O, Filipski A, Kumar S. 2012. Estimating divergence times in large molecular phylogenies. Proc. Natl. Acad. Sci. U.S.A. 109:19333–19338.

Tamura K, Stecher G, Peterson D, Filipski A, Kumar S. 2013. MEGA6: molecular evolutionary genetics analysis version 6.0. Mol. Biol. Evol. 30:2725–2729.

Tamura K, Tao Q, Kumar S. 2018. Theoretical foundation of the RelTime method for estimating divergence times from variable evolutionary rates. Mol. Biol. Evol. 35:1170–1182.

Tong KJ, Duchêne S, Ho SY, Lo N. 2015. Comment on "Phylogenomics resolves the timing and pattern of insect evolution." Science 349:487–487.

Warnock RCM, Yang Z, Donoghue PCJ. 2017. Testing the molecular clock using mechanistic models of fossil preservation and molecular evolution. Proc. R. Soc. B 284:20170227.

Xia X, Yang Q. 2011. A distance-based least-square method for dating speciation events. Mol. Phylogenet. Evol. 59:342–353.

Yang Z. 2007. PAML 4: phylogenetic analysis by maximum likelihood. Mol. Biol. Evol. 24:1586–1591.

Zhu T, dos Reis M, Yang Z. 2015. Characterization of the uncertainty of divergence time estimation under relaxed molecular clock models using multiple loci. Syst. Biol. 64:267–280.

Table 1. Empirical datasets analyzed in this article

Taxonomic Group	Data type[a]	Sequence count[b]	Sequence length	Substitution model	Calibrations[c]	Software[d]	Reference
Primate (A)	M	9	9,361	HKY + G_5	8 uniform	MCMCTree BEAST	Barba-Montoya et al. (2017)
Primate (B)	M	9	9,361	HKY + G_5	4 uniform 4 skewed	MCMCTree BEAST	Barba-Montoya et al. (2017)
Spider	A	40	55,447	WAG + G_5	8 uniform	MCMCTree	Bond et al. (2014); Mello et al. (2017)
Insect	A	143	220,091	LG + G_6	38 uniform	MCMCTree	Tong et al. (2015)

[a]M = mitochondrial DNA; A = nuclear amino acids.
[b]Sequence count excludes the outgroup.
[c]A Cauchy density distribution and an exponential density distribution are used as the skewed density in MCMCTree and BEAST, respectively.
[d]Software used in the original study for estimating divergence times.

16

Figure Legends

Figure 1. An evolutionary tree of three tips showing node times (t_is), branch lengths (b_j's), and branch rates (r_j's).

Figure 2. (a) A primate phylogeny with a user-specified uniform calibration density (gray shade) and an exponential calibration density (green shade). Red dots are the nodes shown in panels b-d. Effective bounds derived using our method (solid blue line) and bounds derived using Mello et al. (2017) procedure (solid orange line) are compared (b and c) when user-specified calibrations are reliable and (d and e) when user-specified calibration of *Homo-Callithrix* split is unreliable. The dashed red line represents the "true simulated age."

Figure 3. Comparisons of RelTime and Bayesian estimates of divergence times and the associated uncertainties. Gray bar represents the Bayesian 95% HPD intervals (x-axis) and RelTime 95% CIs (y-axis). Black dashed line represents 1:1 line. Each graph contains the slope and coefficient of determination (R^2) values of the linear regression through the origin. Calibrated nodes are shown in blue. The dataset name inside each panel refers to table 1.

Figure 4. Comparisons of RelTime 95% CIs (dark red), MCMCTree 95% HPD intervals (gray) and BEAST 95% HPD intervals (blue). Dots are point estimates of divergence times. The dataset name inside each panel refers to table 1.
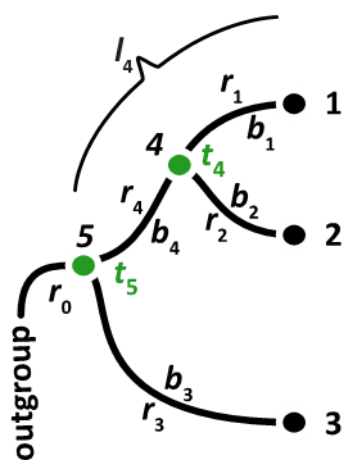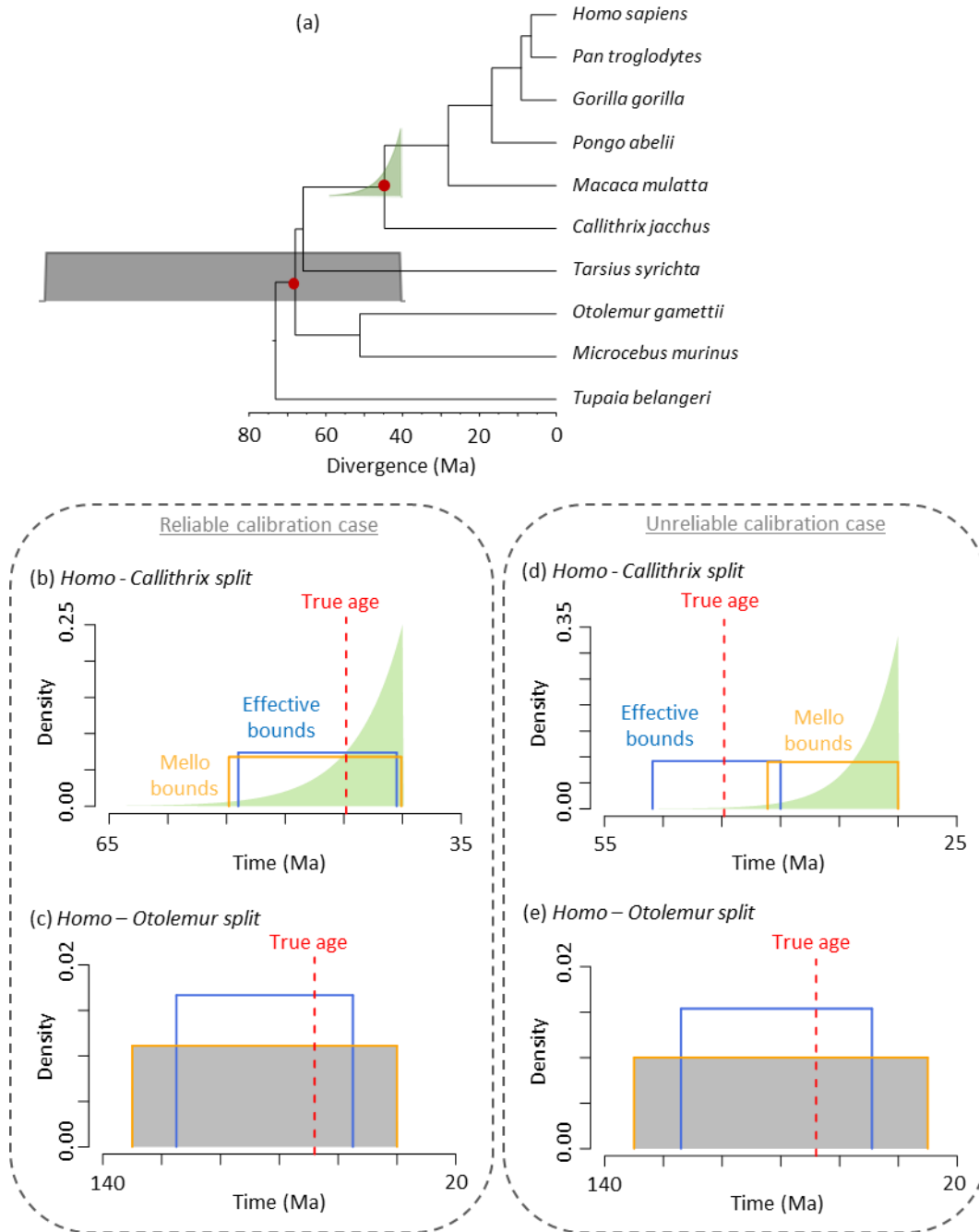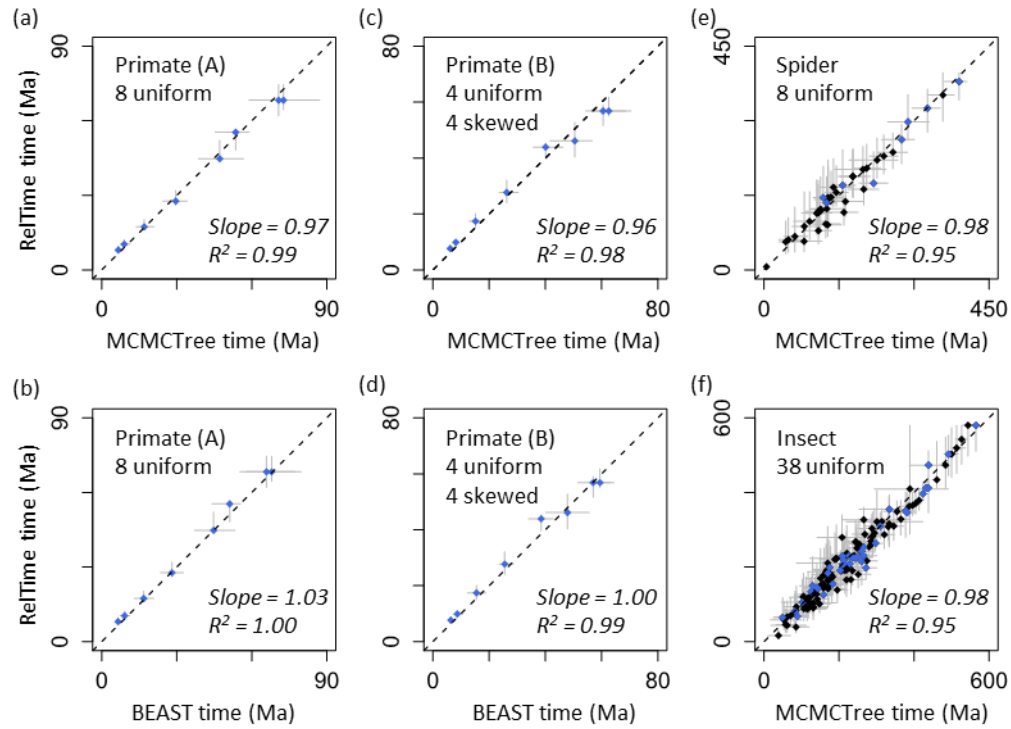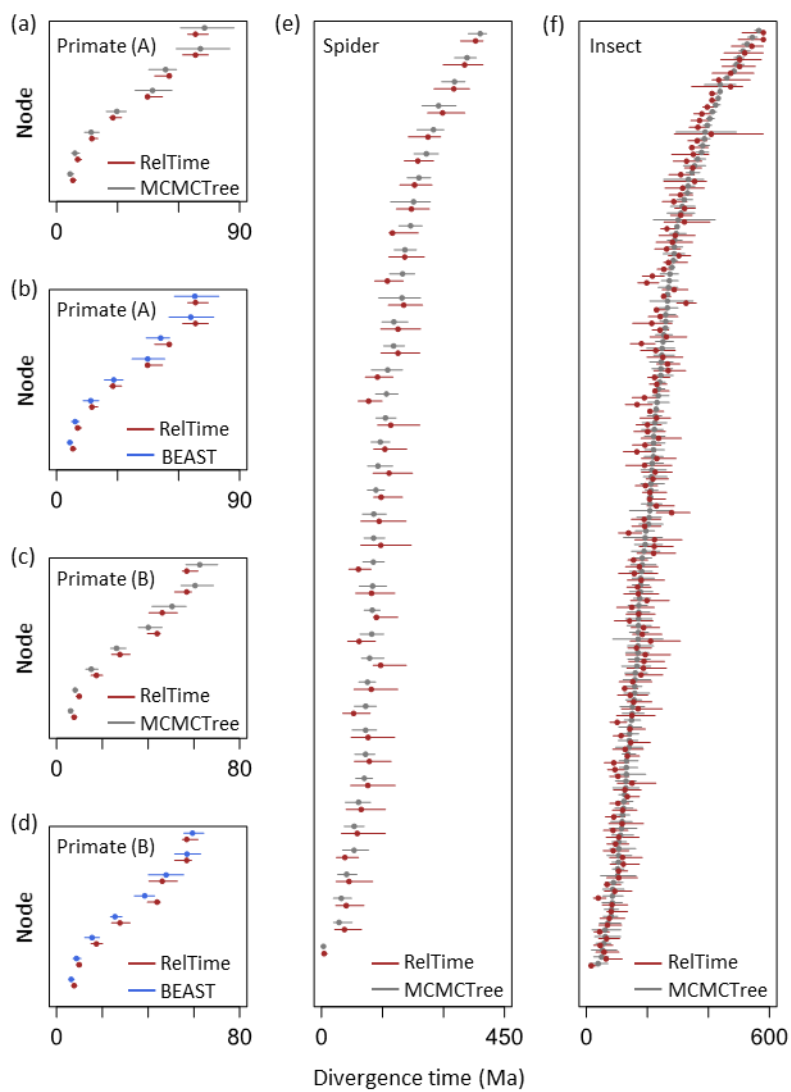
17

Figure 1

Figure 2

Figure 3

Figure 4