VIRUS EVOLUTION

Virus Evolution, 2018, 4(1): vex042

doi: 10.1093/ve/vex042
Resources

TreeTime: Maximum-likelihood phylodynamic analysis

Pavel Sagulenko,[1] Vadim Puller,[1,2,3] and Richard A. Neher[1,2,3,*,†]

[1]Max Planck Institute for Developmental Biology, Spemannstrasse 35, Tübingen 72076, Germany, [2]Biozentrum, University of Basel, Klingelbergstrasse 50, 4056 Basel, Switzerland and [3]SIB Swiss Institute of Bioinformatics, Klingelbergstrasse 50, 4056 Basel, Switzerland

*Corresponding author: E-mail: richard.neher@unibas.ch

[†]http://orcid.org/0000-0003-2525-1407

Abstract

Mutations that accumulate in the genome of cells or viruses can be used to infer their evolutionary history. In the case of rapidly evolving organisms, genomes can reveal their detailed spatiotemporal spread. Such phylodynamic analyses are particularly useful to understand the epidemiology of rapidly evolving viral pathogens. As the number of genome sequences available for different pathogens has increased dramatically over the last years, phylodynamic analysis with traditional methods becomes challenging as these methods scale poorly with growing datasets. Here, we present TreeTime, a Python-based framework for phylodynamic analysis using an approximate Maximum Likelihood approach. TreeTime can estimate ancestral states, infer evolution models, reroot trees to maximize temporal signals, estimate molecular clock phylogenies and population size histories. The runtime of TreeTime scales linearly with dataset size.

Key words: molecular clock phylogenies; phylodynamics; python.

1. Introduction

Phylogenetics uses differences between homologous sequences to infer the history of the sample and learn about the evolutionary processes that gave rise to the observed diversity. In absence of recombination, this history is a tree along which sequences descend from ancestors with modification. In general, the reconstruction of phylogenetic trees is a computationally difficult problem but efficient heuristics often produce reliable reconstructions in polynomial time (Felsenstein, 2004; Price, Dehal, and Arkin, 2010; Stamatakis, 2014). Such heuristics become indispensable for large datasets of hundreds or thousands of sequences.

Beyond phylogenetic tree building, many research questions require parameter inference and hypothesis testing (Pond and Muse, 2005; Drummond et al. 2012). Again, exact inference from large datasets is computationally expensive since it requires high-dimensional optimization of complex likelihood functions or extensive sampling of the posterior distribution. Efficient heuristics are needed to cope with the growing datasets available today.

One particularly common inference problem is estimating the time of historical events from sequence data. This problem goes back to Zuckerkandl and Pauling (1965), who hypothesized that changes in protein sequences accumulate at a constant rate and that the number of differences between homologous sequences can be used as a 'molecular clock' to date the divergence between sequences. Molecular clock methods have since been used to date the divergence of ancient proteins billions of years ago as well as the spread of RNA viruses on time scales less than a year (Langley and Fitch, 1974; Rambaut, 2000; Yoder and Yang, 2000; Sanderson, 2003). Beyond dating of individual divergence events or a common ancestor algorithms have been developed to infer trees where branch lengths correspond directly to elapsed time and each node is placed such that

its position reflects its known or inferred date. Such trees are known as time trees, molecular clock phylogenies, or time stamped phylogenies. These methods have been generalized to allow for variation in substitution rates between different branches of the tree and between sites along a sequence. For a recent review of such methods, see (Kumar and Hedges, 2016).

In addition to questions regarding natural history, time trees are useful to study epidemiology and pathogen evolution (Gardy, Loman, and Rambaut, 2015). Time trees of 'measurably evolving' pathogens can be used to date cross-species transmissions, introductions into geographic regions, and the time course of pathogen population sizes. In outbreak scenarios such as the recent Ebola virus (EBOV) or Zika virus outbreaks, rapid near real-time analysis of large numbers of viral genomes has the potential to assist epidemiological analysis and containment efforts –provided sample collection, sequencing, and analysis are sufficiently rapid (Gardy, Loman, and Rambaut, 2015).

BEAST is one of most sophisticated tools for time tree estimation (Drummond et al. 2012). BEAST samples many possible histories to evaluate posterior distributions of divergence times, evolutionary rates, and many other parameters. BEAST implements a large number of different phylogenetic and phylogeographic models. The sampling of trees, however, results in run-times of days to weeks for moderately large datasets of a few hundred sequences. On the other end of the spectrum are much simpler distance based tools that infer time scaled phylogenies orders of magnitudes faster (Britton et al. 2007; Tamura et al. 2012; To et al. 2016; Volz and Frost, 2017).

We developed a new tool called TreeTime that combines efficient heuristics with probabilistic sequence evolution models. TreeTime infers maximum likelihood time trees of a few thousand tips within a few minutes. TreeTime was designed for applications in molecular epidemiology and analysis of rapidly evolving heterochronous viral sequences (Volz, Koelle, and Bedford, 2013). It is already in use as an integral component of the real-time time outbreak tracking tools nextstrain and nextflu (Neher and Bedford, 2015). The main applications of TreeTime are ancestral state inference, evolutionary model inference, and time tree estimation. We discuss the core algorithms briefly below.

2. Algorithms and implementation

TreeTime's overarching strategy is to find an approximate maximum-likelihood configuration by iterative optimization of simpler subproblems similar in spirit to 'sequential quadratic programming' or 'expectation maximization'. Iteration is used on multiple levels, for example by iterating optimization of branch lengths, ancestral sequences, parameters of the relaxed clock, or coalescent models. Such an iterative procedure typically converges quickly when the branch lengths of the tree are short such that ancestral sequence inference has little ambiguity.

Ancestral sequences or node positions can be determined to optimize the joint or marginal likelihood. A joint maximum-likelihood assignment corresponds to the global configuration with highest likelihood. In a marginal maximum-likelihood assignment, individual parameters are assigned to the most likely value after summing or integrating over all other unknown states. On a tree, both of these optimal assignments can be calculated in linear time (Pupko et al. 2000; Felsenstein, 2004) and TreeTime implements both marginal and joint ancestral reconstructions for ancestral sequences and node dates.

2.1 Iterative branch length optimization

In general, optimizing the branch lengths of a tree is a complicated computational problem with $2N-3$ free parameters and a likelihood function that requires $\mathcal{O}(N)$ steps to evaluate. However, when branch lengths are short and only a minority of sites change on a given branch, a joint optimization of branch lengths and ancestral sequences can be achieved by iteratively inferring branch length and ancestral sequences since corrections due to recurrent substitutions are neglibile. Given a tree topology and the branch length, the maximum-likelihood ancestral sequences can be inferred in linear time (Felsenstein, 2004; Pupko et al. 2000). Likewise maximum-likelihood branch length given the parent and offspring sequences are easy to optimize. We use this iterative optimization scheme to rapidly optimize branch length and ancestral sequences. For more divergent sequences, however, subleading states of internal nodes make a substantial contribution and the iterative optimization will underestimate the branch lengths. In this case, TreeTime can use branch lengths provided in the input tree.

2.2 Maximum-likelihood inference of divergence times

For a fixed tree topology, TreeTime infers ancestral sequences maximizing the joint sequence likelihood (see above). The branch lengths corresponding to the maximum-likelihood molecular clock phylogeny can be computed in linear time using dynamic programming or message passing techniques (Mézard and Montanari, 2009). This approach is similar to the approach by Rambaut (2000), but the dynamic programming technique avoids computationally expensive numerical optimization of the branch lengths.

In analogy to maximum-likelihood inference of ancestral sequences the algorithm proceeds via a post-order tree traversal propagating the maximum-likelihood assignments of subtrees towards the root, and a pre-order traversal selecting the optimal subtree given the placement of the parent node. Specifically, we calculate in post-order for each node n

$$H_n(t|\mathcal{C}_n) = E_n(t) \prod_{c\in\mathcal{C}_n} C_c(t)\,, \tag{1}$$

the likelihood that the node sits at position t given the information and constraints propagated from its children $\mathcal{C}_n$. $E_n(t)$ accounts for external contraints imposed on the date of the node (e.g. fossil dating), while the product runs over all children c of node n and multiplies the integrated messages of all subtending nodes. The time t is measured as time before present. Temporal information is propagated along the branches of the tree via

$$C_n(t_p) = \max_{\tau} b_n(\tau) H_n(t_p-\tau|\mathcal{C}_n)\,, \tag{2}$$

where $b_n(\tau)$ is the probability distribution of the branch length τ between the focal node n and its parent. This distribution is conditional on the sequences assigned to node n and its parent. Intuitively, $C_n(t_p)$ specifies the distribution of the date t_p of the parent of node n, given the constraints from the tips descending from node n and the substitutions that accumulated on the branch to the parent node. The different objects are illustrated in Fig. 1.

During the post-order traversal, the branch lengths $\tau(t_p)$ maximizing Equation (2) for a given t_p are tabulated and saved

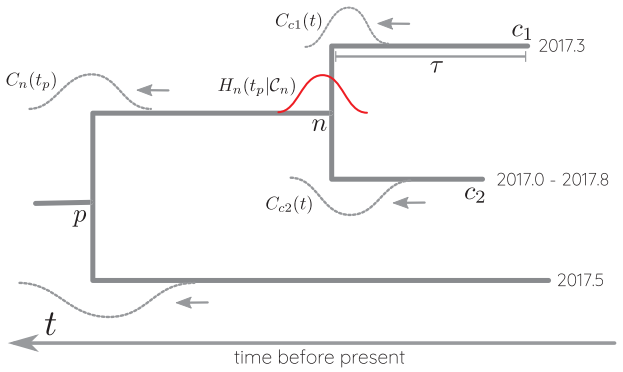

Figure 1. Illustration of TreeTime's time tree inference algorithm. Terminal nodes in the tree are either associated with exact dates or date ranges (node c_2 in this example). These temporal constraints are convolved with the distribution $b_{c_i}(\tau)$ of the branch length τ leading to node c_i to yield $C_{c_i}(t)$. At the internal node n, the messages from children c_1 and c_2 are multiplied and contribute to $H_n(t|\mathcal{C}_n)$. The latter is further passed down to the parent by convolving with $b_n(\tau)$.

for the back-trace. Once the post-order traversal arrives at the root, the root is assigned the time $t_n = \mathrm{argmax}_t E(t)\prod_c C_c(t)$.

The post-order traversal is followed by a pre-order back-trace during which the branch length of each internal node is assigned to the optimal $\tau(t_p)$ conditional on the parental position t_p. To accelerate the optimization, TreeTime tabulates the branch length likelihood function $b_n(\tau)$ and the subtree origin likelihoods $H_n(t|\mathcal{C}_n)$.

The above algorithm assigns each node to the time that maximizes the joint likelihood of all branch lengths in analogy to the ancestral state reconstruction algorithm by Pupko et al. (2000). The marginally optimal time of each internal node, that is, the time after integration over all other unconstrained nodes, can be determined in a similar manner by replacing the max in Equation (2) by a convolution integral over τ

$$C'_n(t_p) = \int_0^\infty b_n(\tau) H'_n(t_p - \tau|\mathcal{C}_n)\, d\tau\,, \tag{3}$$

where $H'_n(t|\mathcal{C}_n)$ is the analog of $H_n(t|\mathcal{C}_n)$ in Equation (1) multiplying the C'_c of all children and any external date prior.

Once the post-order transversal arrives at the root, the marginal distribution of time t of the root node r is given by

$$P_r(t) = \frac{E_r(t)}{Z_r}\prod_{c\in\mathcal{C}_r} C'_c(t) \tag{4}$$

where Z_r is a normalization factor. The corresponding marginal distributions of other nodes are then calculated during a pre-order traversal via

$$P_n(t) = \frac{1}{Z_n} H_n(t|\mathcal{C}_n)\int_0^\infty b(\tau)\frac{P_p(t+\tau)}{C'_n(t+\tau)}\, d\tau\,. \tag{5}$$

The factor $H_n(t|\mathcal{C}_n)$ accounts for the date information coming from the leaves of node n, while the integral contributes the date information from clades other than node n and its children. Note that the contribution of node n to P_p is removed by dividing $P_p(t+\tau)$ by $C'_n(t+\tau)$.

The result of the marginal reconstruction is a probability distribution of the node date given the tree, the ancestral sequence assignment, and the evolutionary model while the unknown times of other nodes are traced out. From this distribution, confidence intervals of node dates can be computed in a straight-forward manner.

TreeTime allows one to compute joint or marginal maximum-likelihood dates, but the algorithm described above can be used for any continuous character on the tree. In Equation (2), $b_n(\tau)$ can be replaced by any transmission function that depends either on the branch or properties of the child and parent node. We will use an analogous algorithm below to estimate parameters of relaxed molecular clock models.

2.3 Efficient search for the optimal root

The fraction of variance in root-to-tip (RTT) distance explained by a linear regression on sampling date is given by

$$r^2 = \frac{\left(\sum_i (t_i - \langle t\rangle)(d_i - \langle d\rangle)\right)^2}{\sum_k (t_k - \langle t\rangle)^2 \sum_l (d_l - \langle d\rangle)^2} \tag{6}$$

where the sums run over all tips of the tree and t_i and d_i are the sampling date and the distance from the root to node i, respectively. The distances d_i are measured as the sum of lengths of all branches from the root to the tip, that is, the expected number of substitutions since the root divided by the length of the sequence. The angular brackets denote the sample average. The regression and r^2 depend on the choice of root since the d_i depend on the root. In absence of an outgroup, the root is often chosen to maximize r^2 or minimize the squared residuals of a linear fit to the RTT distance. Programs such as TempEst (Rambaut et al. 2016) and LSD (To et al. 2016) allow to search for the root that maximizes this correlation and TreeTime has implemented similar functionality.

This search for the optimal root can be achieved in linear time in the number of sequences N by first calculating

$$\theta_n = \sum_{i\in\mathcal{L}_n} d_{n,i}\,, \quad \gamma_n = \sum_{i\in\mathcal{L}_n} t_i d_{n,i} \quad \text{and} \quad \delta_n = \sum_{i\in\mathcal{L}_n} d^2_{n,i} \tag{7}$$

for each internal node n. Here, the sum runs over all tips $i \in \mathcal{L}_n$ of node n while t_i and $d_{n,i}$ are the sampling date and the distance of tip i from node n, respectively. The quantities θ_n, γ_n, and δ_n can be calculated recursively from θ_c, γ_c, and δ_c of the child nodes in one post-order traversal. Once those quantities are calculated, the corresponding quantities Θ_n, Γ_n, and Δ_n that sum contributions from all tips—not just the subtending ones—can be calculated in one pre-order traversal.

With these quantities at hand, r^2 can be calculated for any choice of root on the tree as detailed in the Appendix. Hence two tree traversals are sufficient to determine the optimal root. The root position that minimizes the mean squared residual can be calculated analogously.

In general, the optimal position of the root will not be an internal node, but a position between two nodes on a branch of the tree. Such optimal position on internal branches of the tree can be determined from the quantities calculated above by solving a quadratic equation without any numerical optimization. The required algebra is described in the Appendix.

2.4 Resolving polytomies

Phylogenetic trees of many very similar sequences are often poorly resolved and contain multifurcating nodes also known as polytomies. Tree building software often randomly resolves these polytomies into a series of bifurcations. However, the order of bifurcations will often be inconsistent with the

temporal structure of the tree resulting in poor approximations. To overcome this problem, TreeTime can prune all branches of length zero and resolve the resulting polytomies in a manner consistent with the sampling dates. For each pair of nodes, TreeTime calculates by how much the likelihood would increase when grouping this pair of nodes into a clade of size two. The polytomy is then resolved iteratively by always grouping pairs corresponding to the highest gain.

2.5 Coalescent models

The likelihood of observing a particular genealogical tree depends on the size of the population, its geographic structure, and fitness variation in the population (Kingman, 1982; Nordborg, 1997; Neher, 2013). Hence parameters of models describing the ensemble of genealogies can be estimated from the data.

In the simplest case of a panmictic population without fitness variation, the ensemble of genealogies is described by a Kingman (1982) coalescent, possibly with a population size that changes over time. Within the Kingman coalescent, merger events occur at random with a rate $\lambda(t)$ that depends on the population size $N(t)$ and the current number of lineages $k(t)$.

$$\lambda(t) = \frac{k(t)(k(t)-1)}{2N(t)} \tag{8}$$

Here, the population size $N(t)$ defines a time scale measured in units of generation time and we will more generally refer to this time scale by $T_c(t)$ and measure it in units of the inverse clock rate.

The contribution of a branch between time points t_0 (child) and t_1 (parent) in the tree to the likelihood is then given by

$$p(t_0, t_1) = e^{-\int_{t_0}^{t_1} \kappa(t)dt}, \tag{9}$$

where $\kappa(t) = (k(t)-1)/2T_c(t)$ is the rate at which a given lineage merges with any of the other. A merger at time t contributes a factor $\lambda(t)$ to the coalescent likelihood.

TreeTime can estimate population sizes or coalescent time scales by maximizing the likelihood contribution of the coalescent likelihood for a fixed tree. The latter can be evaluated in one tree traversal by summing contributions from branches and merger events. In addition to a constant T_c, TreeTime can model T_c as a piecewise linear function and optimize the parameters of that function. Such piecewise functions are known as 'skyline' (Strimmer and Pybus, 2001).

As part of the iterative optimization by TreeTime, the next round of optimization of branch lengths and dates of ancestral nodes will account for the coalescent likelihood. The newly inferred dates will in turn be used to update the parameters of the coalescent model as described earlier.

2.6 Inference of time reversible substitution models

Large phylogenies typically contain 100s of substitutions and thus provide enough information to infer substitution models from the data. General time reversible (GTR) substitution models (Felsenstein, 2004) are parameterized by equilibrium state frequencies π_i and a symmetric substitution matrix W_{ij}. The substitution rate from state $j \to i$ is then $Q_{ij} = \pi_i W_{ij}$.

TreeTime infers parameters of GTR models via an iterative procedure similar to Expectation–Maximization algorithms. TreeTime first reconstructs ancestral sequences using a standard substitution model specified by the user (Jukes-Cantor by default). From this reconstruction, TreeTime calculates the time T_i spent in different states i across the tree, and the number of substitutions n_{ij} between any pair of states (i, j). Then, π and W are determined by iterating the two equations

$$W_{ij} = \frac{n_{ij} + n_{ji} + 2p_c}{\pi_i T_j + \pi_j T_i + 2p_c} \tag{10}$$

$$\pi_i = \frac{\sum_j n_{ij} + p_c + m_i}{\sum_j W_{ij} T_j + \sum_j (m_j + p_c)}, \tag{11}$$

where p_c is a small pseudo-count driving the estimate towards a flat Jukes-Cantor model in absence of data, and the m_i are the number times state i is observed in the sequence of the root. W_{ij} are evaluated at fixed π, followed by calculating π with the current W_{ij}. After each iteration, π is normalized to one, the diagonal of W_{ij} is set to $-\pi_i^{-1}\sum_{j\neq i} W_{ij}\pi_j$, and W_{ij} is rescaled such that the total expected substitution rate $-\sum \pi_i W_{ii} \pi_i$ equals one. The rescaling of π and W_{ij} can be absorbed into an overall rate μ. This algorithm typically converges in a few iterations.

2.7 Relaxed clocks

Substitution rates can vary across the tree and models that assume constant clock rates may give inaccurate inferences. Models that allow for clock rate variation have been proposed (Hasegawa, Kishino, and Yano, 1989; Yoder and Yang, 2000; Drummond et al. 2006). These models typically regularize clock rate variation through a prior and penalize rapid changes of the rate by coupling the rate along branches—known as autocorrelated or local molecular clock (Thorne, Kishino, and Painter, 1998; Aris-Brosou, Yang, and Huelsenbeck, 2002).

TreeTime implements an autocorrelated molecular with a normal prior on variation in clock rates. The choice of the normal prior allows for an exact and linear time solution for the maximum-likelihood substitution rates via the same forward/backward trace algorithm used for the inference of dates of internal nodes. Other priors could be implemented, but would require numerical optimization or approximations.

2.8. Implementation

TreeTime is implemented in Python (version 2.7) and uses the packages numpy and scipy for optimization, linear algebra, and interpolation Jones et al. (2001–2017) and van der Walt, Colbert, and Varoquaux (2011). Computationally costly operations are cast into array operations executed by numpy whenever possible.

TreeTime is organized as a hierarchy of classes. TreeAnc performs maximum-likelihood inference of ancestral sequences, ClockTree infers a time scaled phylogeny given a tree topology, and TreeTime adds an additional layer of functionality including rerooting, polytomy resolution, coalescent models, and relaxed clocks. The substitution model is implemented in the class GTR.

This structure allows TreeTime to be used in a modular fashion in Python based phylogenetic analysis pipelines. In addition, scripts can be called from the command line to perform standard tasks such as ancestral sequence inference, rerooting of trees, and time tree estimation.

2.9 Availability

TreeTime is published under an MIT license and available at github.com/neherlab/treetime. Data and scripts necessary used to validate TreeTime are available at github.com/neherlab/

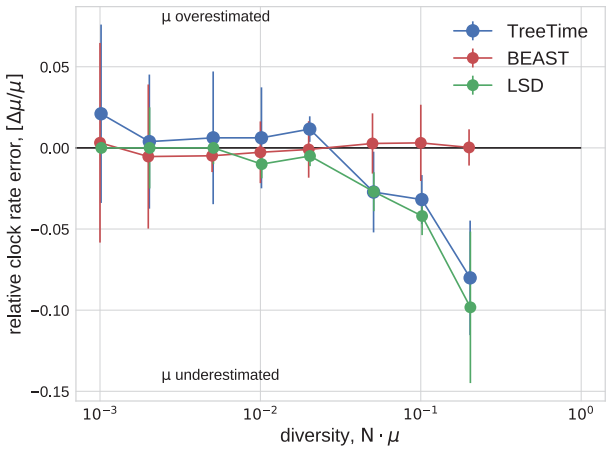

Figure 2. Estimation of the evolutionary rate from simulated data. TreeTime and LSD (following tree reconstruction with FastTree) underestimated the rate when branch lengths are long but return accurate estimates for low diversity samples. The graph shows median values, error bars indicate the inter-quartile distances.

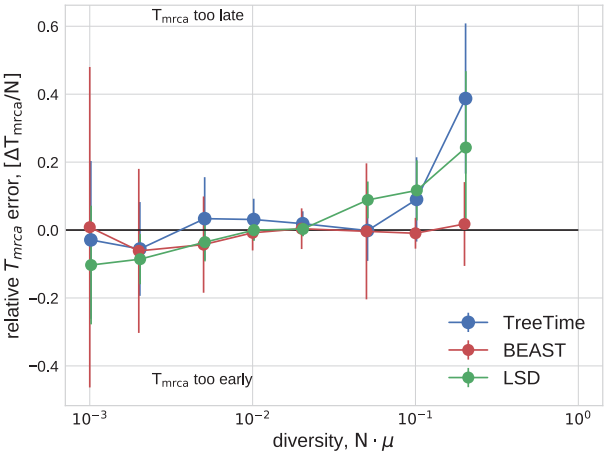

Figure 3. Estimation of the T_{MRCA} from simulated data. TreeTime, LSD, and BEAST estimated the time of the MRCA within 10% accuracy at low diversity, but TreeTime and LSD tended to overestimate the date of the root when branch lengths are long. The graph shows median values, error bars indicate the inter-quartile distances.

treetime_validation. TreeTime can be used via a web interface at treetime.ch.

3. Validation and performance

To assess the accuracy of date reconstructions of TreeTime and to compare its performance to existing tools such as Bayesian Evolutionary Analysis Sampling Trees (BEAST) and Least-square dating (LSD) (Drummond et al. 2012; To et al. 2016), we generated toy data using the FFPopSim forward simulation library (Zanini and Neher, 2012). We simulated populations of size $N = 100$ and used a range evolutionary rates $\mu = [10^{-5}, \ldots, 2 \cdot 10^{-3}]$ resulting in expected genetic diversity from 0.001 to 0.2. Sequences were sampled every 10, 20, or 50 generations. The length of the simulated sequences was $L = 1000$.

Fig. 2 shows the error in the estimates of the clock rate for TreeTime, LSD, and BEAST as a function of genetic diversity. TreeTime and LSD estimated the clock rate accurately at low diversity but tended to underestimate the rates at when diversity exceeds a few percent. This is expected in the case of TreeTime since maximum-likelihood sequence assignment can result in underestimated branch lengths. BEAST produced accurate estimates across the entire range of diversities. By sampling trees, BEAST does not suffer from the atypical maximum-likelihood assignments.

In a similar manner, TreeTime and LSD estimated the time of the most recent common ancestor to within 10% accuracy at low diversity (relative to the coalescence time) with larger deviations at diversity above 10%, see Fig. 3. BEAST returned accurate estimates across the entire range of diversities.

We also ran TreeTime on simulated data provided by To et al. (2016) and compared it to the results reported by To et al. (2016) for LSD, BEAST, and a number of other methods. Figure 4 compares the accuracy of T_{MRCA} and clock rate estimates, showing that TreeTime achieves similar or better accuracy than other methods.

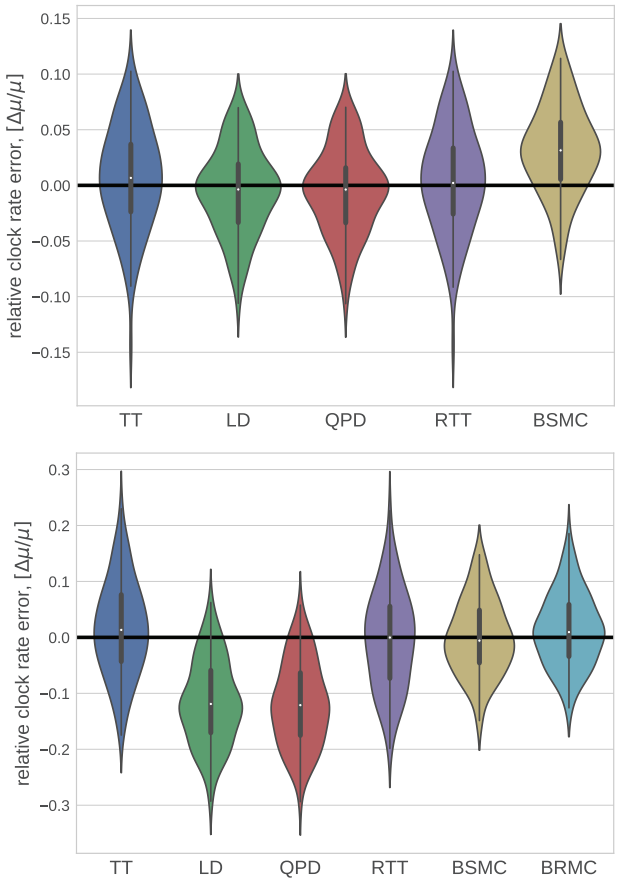

Figure 4. Method comparison on LSD test data. TreeTime (TT) showed comparable or better accuracy as BEAST (strict clock: BSMC; relaxed clock: BRMC), LSD (linear dating: LD; quadratic programming dating: QPD), or RTT regression when run on simulated data provided by (To *et al.*, 2016). Both panels use the tree set 750_11_10, the top and bottom panel show runs on alignments generated with a strict and relaxed molecular clock, respectively.

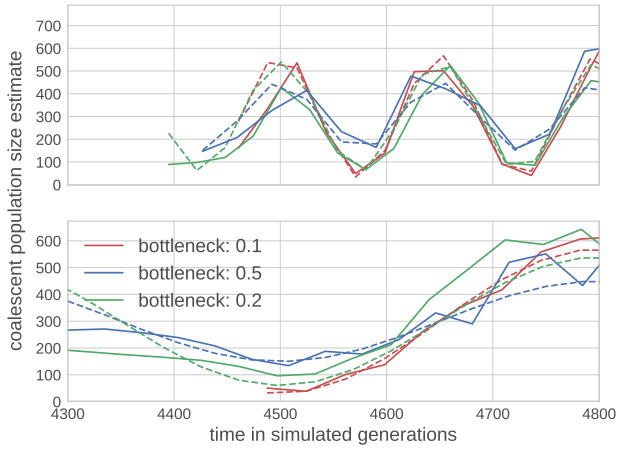

Figure 5. Reconstruction of fluctuating population sizes by TreeTime. The graph shows simulated population size trajectories (dashed lines) and the inference by TreeTime as solid lines of the same color. Different lines vary in the bottleneck sizes of 10% (red), 20% (green), and 50% (blue) of the average population size. The top panel shows data for fluctuations with period 0.5 N, the bottom panel 2 N. The average population size is $N = 300$.

3.1 Coalescent model inference

Population bottlenecks, selective sweeps, or population structure affect the rate of coalescence in a time-dependent manner. BEAST can infer a history of effective population size (inverse coalescence rate) from a tree—often known as skyline. TreeTime can perform a similar inference by maximizing the coalescence likelihood with respect to the pivots of a piecewise linear approximation of the coalescence rate history $T_c(t)$ (aka effective population size). To test the power and accuracy of this inference, we simulated sinusoidal population size histories of different amplitude and period, uniformly sampled sequences through time, and used these data to estimate the coalescent rate history. True and estimated population size histories agree well with each other as shown in Fig. 5.

3.2 Influenza phylogenies

The dense sampling of influenza A virus sequences over many decades makes this virus an ideal test case to evaluate the sensitivity of time tree estimation to sampling depth. We estimated the clock rate and the time of the most recent common ancestor of influenza A/H3N2 HA sequences sampled from 2011 to 2013 for sets of sequences varying from 30 to 3,000, see Fig. 6. TreeTime estimates are stable across this range, while estimates by LSD tend to drift with lower rates and older MRCAs for larger samples. Estimates by BEAST are generally consistent with TreeTime.

Next, we tested how accurately TreeTime infered dates of tips when only a fraction of tips have dates assigned. Every tip in TreeTime can either be assigned a precise date, an interval within which the date is assumed to be uniformly distributed, or no constraint at all. TreeTime will then determine the probability distribution of the date of the node based on the distribution of the ancestor and the substitutions that occurred since the ancestor.

Figure 7 shows the distribution of error in leaf date reconstruction as the fraction of missing dates increased from 5 to 95% of all nodes. TreeTime estimated the date of influenza sequences to an average accuracy of ≈0.5 years if >50% of dates are known.

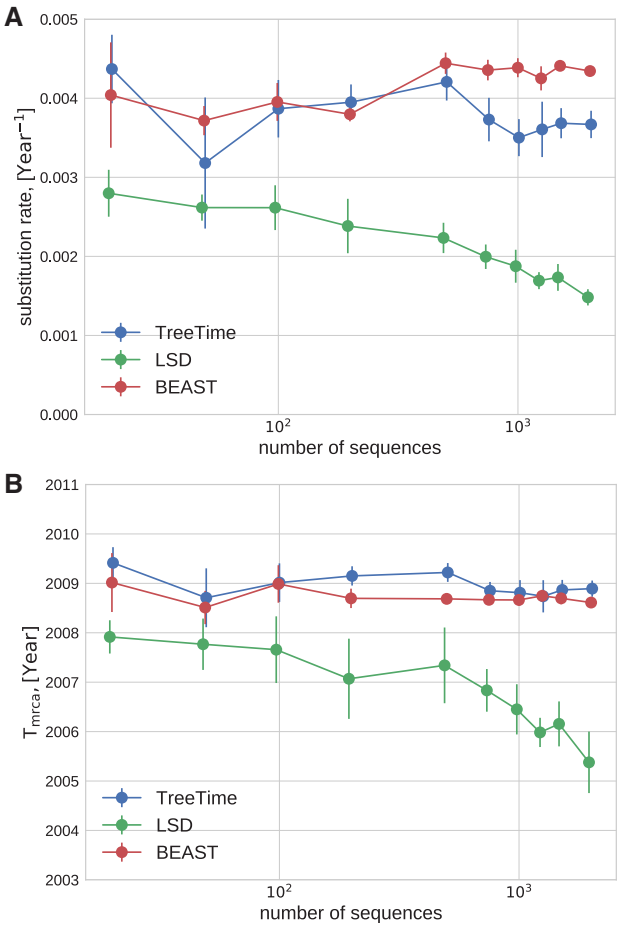

Figure 6. Sensitivity the dataset size. TreeTime and BEAST returned consistent estimates of the rate of evolution (A) and the T_{MRCA} (B) when analyzing alignments of Influenza A/H3N2 HA sequences of various size. LSD showed a systematic drift.

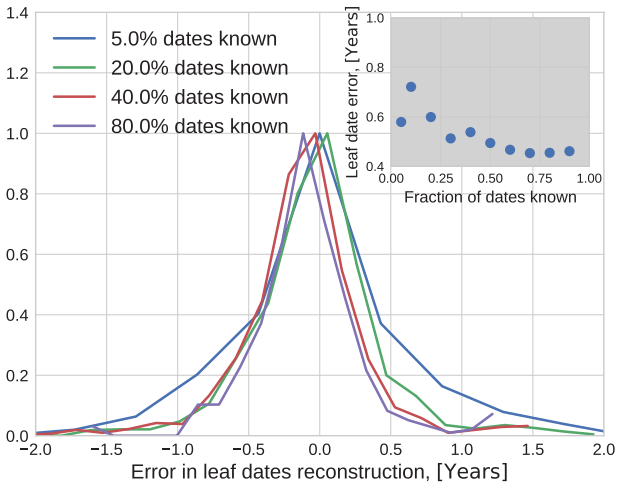

Figure 7. Sensitivity to missing information. The inter-quartile range of the error of estimated tip dates decreases from 0.7 to 0.5 years as the fraction of known dates increases from 5 to 90% (see inset).

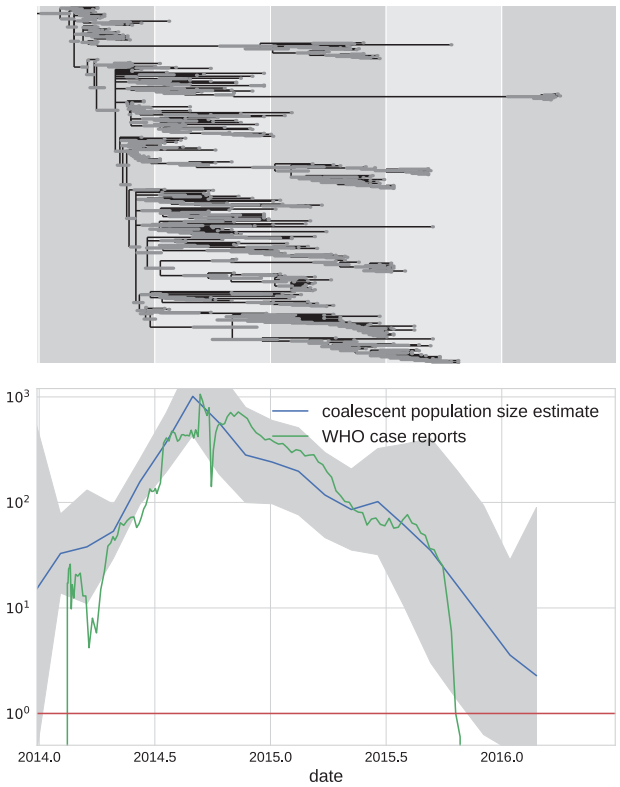

Figure 8. EBOV phylodynamic analysis. The top panel shows a molecular clock phylogeny of EBOV sequences obtained over from 2014 to 2016 in West Africa. The lower panel shows the estimate of the coalescent population size along with its confidence intervals. The estimate suggests an exponential increase until late 2014 followed by a gradual decrease leading to almost complete eradication by 2016. Ebola case counts, as reported by the WHO (2016) agree quantitatively with the estimate.

3.3 Analysis of the 2014–15 EBOV outbreak

In 2014, West Africa experienced the largest known outbreak of EBOV in humans. The genomic epidemiology has been studied intensively by multiple groups (Dudas et al. 2017). Here, we reanalyzed a subset of 350 EBOV sequences sampled throughout the outbreak from 2014–16. Due to the dense sampling, the maximum-likelihood phylogeny has many unresolved nodes and TreeTime was used to resolve polytomies using temporal information. After automatic rooting and GTR model inference, TreeTime produced the time tree shown in Fig. 8. The GTR model inferred from the tree was

$$\pi = \begin{array}{ll} A: & 0.32 \\ C: & 0.21 \\ G: & 0.195 \\ T: & 0.275 \end{array} \qquad W = \begin{array}{c|cccc} & A & C & G & T \\ \hline A & \cdot & 0.45 & 2.7 & 0.28 \\ C & 0.45 & \cdot & 0.25 & 3.7 \\ G & 2.7 & 0.25 & \cdot & 0.45 \\ T & 0.28 & 3.7 & 0.45 & \cdot \end{array} \tag{12}$$

This analysis took 4 min to complete on a 2016 laptop (Dell XPS13) with an i7 processor using a single CPU. In addition to inferring a time tree, TreeTime estimated the time course of the coalescent population size shown in the lower panel of Fig. 8. The estimated population size closely mirrors the case counts reported by the WHO throughout this period.

4. Discussion

TreeTime was developed to analyse large heterochronous viral sequence alignments and we have used TreeTime as the core component of the real-time phylogenetics pipelines nextstrain and nextflu (Neher and Bedford, 2015). TreeTime tries to strike a useful compromise between inflexible but fast heuristics and computationally expensive Bayesian approaches that require extensive sampling of treespace. The overarching algorithmic strategy is iterative optimization of efficiently solvable subproblems to arrive at a consistent approximation of the global optimum. Although this strategy is approximate and often assumes short branch length, it converges fast for many applications and trees with thousands of tips can be analyzed in a few minutes. In this paper, we presented analyses of human seasonal influenza A/H3N2 virus sequences and sequences of the recent EBOV outbreak. In both cases, average pairwise distances between strains are 10% and individual branches in the trees are much shorter still. TreeTime assumption of short branches is therefore met.

Rapid, efficient analysis phylodynamic algorithms are of increasing importance as datasets are increasing in size. For example during the recent outbreaks of EBOV and Zika virus, hundreds of sequences were generated and needed to be analyzed in near real time to inform containment efforts. Similarly, the GISRS network for surveillance of seasonal influenza virus sequences hundreds of viral genomes per month. Timely analysis of these data with Bayesian methods that require extensive tree sampling such as BEAST is difficult. Sequencing from EBOV, Zika virus outbreaks, or seasonal influenza viruses are typically very similar to each other (>90% identity) such that TreeTime assumptions and approximations are justified.

When compared with other methods recently developed for rapid estimation of time trees (Britton et al. 2007; Tamura et al. 2012; To et al. 2016), TreeTime uses probabilistic models of evolution, allows inference of ancestral characters, and coalescent models. In TreeTime, every node of the tree can be given a strict or probabilistic date constraint. This higher model complexity results in longer run times, but the scaling of run times remains linear in the size of the dataset and alignments with thousands of sequences can be analyzed routinely. The time tree inference and dating are typically faster than the estimation of the tree topology.

TreeTime was tested predominantly on sequences from viruses with a pairwise identity above 90%. The iterative optimization procedures are not expected to be accurate for trees were many sites are saturated. In scenarios with extensive uncertainty of ancestral states and tree topology, convergence of the iterative steps cannot be guaranteed. While in many cases TreeTime might still give approximate branch lengths, ancestral assignments and time tree estimates, these need to be checked for plausibility. In general global optimization and sampling of the posterior can not be avoided.

TreeTime can be used in a number of different ways. The core TreeTime algorithms and classes can be used in larger phylogenetic analysis pipelines as Python scripts. This is the most flexible way to use TreeTime and all the different analysis steps can be combined in custom ways with user specified parameters. In addition, we provide command-line scripts for typical recurring tasks such as ancestral state reconstruction, rerooting to maximize temporal order, and time tree inference. We also implemented a web-server that allows exploration and

analysis of heterochronous alignments in the browser without the need to use the command-line.

Conflict of interest: None declared.

References

Aris-Brosou, S., Yang, Z., and Huelsenbeck, J. (2002) 'Effects of Models of Rate Evolution on Estimation of Divergence Dates With Special Reference to the Metazoan 18s Ribosomal RNA Phylogeny, *Systematic Biology*, 51: 703.

Britton, T., Anderson, C. L., Jacquet, D. et al. (2007) 'Estimating Divergence Times in Large Phylogenetic Trees', *Systematic Biology*, 56: 741.

Drummond, A. J., Ho, S. Y. W., Phillips, M. J. et al. (2006) 'Relaxed Phylogenetics and Dating With Confidence', *PLOS Biology*, 4: e88.

——, Suchard, M. A., Xie, D. et al. (2012) 'Bayesian phylogenetics with BEAUti and the BEAST 1.7', *Molecular Biology and Evolution*, 29: 1969.

Dudas, G., Carvalho, L. M., Bedford, T. et al. (2017) 'Virus Genomes Reveal Factors That Spread and Sustained the Ebola Epidemic' *Nature*, 544: 309.

Felsenstein, J. 2004, *Inferring Phylogenies*. Sunderland, MA: Sinauer Associates.

Gardy, J., Loman, N. J., and Rambaut, A. (2015) 'Real-Time Digital Pathogen Surveillance - The Time is Now', *Genome Biology*, 16: 155.

Hasegawa, M., Kishino, H., and Yano, T-a. (1989) 'Estimation of branching dates among primates by molecular clocks of nuclear DNA which slowed down in Hominoidea', *Journal of Human Evolution*, 18: 461.

Jones, E., Oliphant, T., Peterson, P. et al. 2001–2017, 'SciPy: Open Source Scientific Tools for Python', <http://www.scipy.org/> accessed 12 January 2017.

Kingman, J. F. C. (1982) 'The coalescent', *Stochastic Processes and Their Applications*, 13: 235.

Kumar, S., and Hedges, S. B. (2016) 'Advances in Time Estimation Methods for Molecular Data', *Molecular Biology and Evolution*, 33: 863.

Langley, C. H., and Fitch, W. M. (1974) 'An examination of the constancy of the rate of molecular evolution', *Journal of Molecular Evolution*, 3: 161.

Mézard, M., and Montanari A., (2009), *Information, physics, and computation*. Oxford University Press.

Neher, R. A. (2013) 'Genetic Draft, Selective Interference, and Population Genetics of Rapid Adaptation', *Annual Review of Ecology, Evolution, and Systematics*, 44: 195.

—— and Bedford, T. (2015) 'Nextflu: Real-Time Tracking of Seasonal Influenza Virus Evolution in Humans', *Bioinformatics (Oxford, England)*, 31: 3546.

Nordborg, M. (1997) 'Structured coalescent processes on different time scales', *Genetics*, 146: 1501–14.

Pond, S. L. K., and Muse, S. V. (2005) 'HyPhy: hypothesis testing using phylogenies', in *Statistical methods in molecular evolution*, 125–81. New York: Springer.

Price, M. N., Dehal, P. S., and Arkin, A. P. (2010) 'FastTree 2-Approximately Maximum-Likelihood Trees for Large Alignments', *PLoS One*, 5: e9490.

Pupko, T., Pe, I., Shamir, R. et al. (2000) 'A Fast Algorithm for Joint Reconstruction of Ancestral Amino Acid Sequences', *Molecular Biology and Evolution*, 17: 890.

Rambaut, A. (2000) 'Estimating the Rate of Molecular Evolution: Incorporating Non-Contemporaneous Sequences Into Maximum Likelihood Phylogenies', *Bioinformatics*, 16: 395.

——, Lam, T. T., Carvalho, L. M. et al (2016) 'Exploring the Temporal Structure of Heterochronous Sequences Using TempEst (Formerly Path-O-Gen', *Virus Evolution*, 2: vew007.

Sanderson, M. J. (2003) 'r8s: Inferring Absolute Rates of Molecular Evolution and Divergence Times in the Absence of a Molecular Clock', *Bioinformatics*, 19: 301.

Stamatakis, A. (2014) 'RAxML Version 8: A Tool for Phylogenetic Analysis and Post-Analysis of Large Phylogenies,' *Bioinformatics*, 30: 1312.

Strimmer, K., and Pybus, O. G. (2001) 'Exploring the Demographic History of DNA Sequences Using the Generalized Skyline Plot', *Molecular Biology and Evolution*, 18: 2298.

Tamura, K., Battistuzzi, F. U., Billing-Ross, P. et al (2012) 'Estimating Divergence Times in Large Molecular Phylogenies', *Proceedings of the National Academy of Sciences of the United States of America*, 109: 19333.

Thorne, J. L., Kishino, H., and Painter, I. S. (1998) 'Estimating the Rate of Evolution of the Rate of Molecular Evolution', *Molecular Biology and Evolution*, 15: 1647.

To, T.-H., Jung, M., Lycett, S. et al. (2016) 'Fast Dating Using Least-Squares Criteria and Algorithms', *Systematic Biology*, 65: 82.

Volz, E. M., and Frost, S. D. W. (2017) 'Scalable Relaxed Clock Phylogenetic Dating', *Virus Evolution*, 3: 1–9.

——, Koelle, K., and Bedford, T. (2013) 'Viral Phylodynamics' *PLOS Computational Biology*, 9: e1002947.

van der Walt, S., Colbert, S., and Varoquaux, G. (2011). 'The NumPy array: a structure for efficient numerical computation', *Computing in Science & Engineering*, 13: 22–30.

WHO. (2016) '2014 Ebola Outbreak in West Africa - Case Counts', <https://www.cdc.gov/vhf/ebola/csv/graph1-cumulative-reported-cases-all.xlsx>.

Yoder, A. D., and Yang, Z. (2000) 'Estimation of Primate Speciation Dates Using Local Molecular Clocks', *Molecular Biology and Evolution*, 17: 1081.

Zanini, F., and Neher, R. A. (2012) 'FFPopSim: An Efficient Forward Simulation Package for the Evolution of Large Populations', *Bioinformatics*, 28: 3332.

Zuckerkandl, E., and Pauling, L. (1965) 'Evolutionary divergence and convergence in proteins', *Evolving genes and proteins*, 97: 97–166.

Appendix

To calculate the correlation between the RTT distances and tip dates via Equation (6), one first needs to calculate the means and (co)variances of tip dates and RTT distances. For a tree with N tips, this requires $\mathcal{O}(N)$ operations and calculating it for all internal nodes would therefore require $\mathcal{O}(N^2)$ operations. The same covariances are needed to calculate the regression parameters and the residuals. However, its is possible to calculate the quantities for all nodes at once, reducing the total number of operations to $\mathcal{O}(N)$.

The speed-up is possible through recursively calculating sums and averages on the tree. We denote the set of tips that descend of node n by $\mathcal{L}_n$. We will need the number tips $M_n = |\ell_n|$, the sum of their sampling times $\tau_n = \sum_{i\in\mathcal{L}_n} t$, the sum of their distances $d_{n,i}$ from node n $\theta_n = \sum_{i\in\mathcal{L}_n} d_{n,i}$, and the analogous higher order quantities $\gamma_n = \sum_{i\in\mathcal{L}_n} t_i d_{n,i}$ and $\delta_n = \sum_{i\in\mathcal{L}_n} d_{n,i}^2$.

First, assign $M_n = 1$, $\tau_n = t_n$, $\theta_n = 0$, $\gamma_n = 0$ and $\delta_n = 0$ for all tips of the tree. Then, in one post-order transversal over internal

nodes, we can calculate these quantities by summing the following expressions over the children $\mathcal{C}_n$ of node n.

$$M_n = \sum_{c \in \mathcal{C}_n} M_c \tag{13}$$

$$\begin{aligned}
\tau_n &= \sum_{c \in \mathcal{C}_n} \tau_c \\
\theta_n &= \sum_{c \in \mathcal{C}_n} M_c l_c + \theta_c \\
\gamma_n &= \sum_{c \in \mathcal{C}_n} l_c \tau_c + \gamma_c \\
\delta_n &= \sum_{c \in \mathcal{C}_n} M_n l_c^2 + 2 l_c \theta_n + \delta_c
\end{aligned} \tag{14}$$

The length of branches leading from node n to child c is denoted by l_c. To calculate the covariances at a particular node n, we need to sum over all terminal nodes rather than only tips that descend from the node. We denote the corresponding quantities by capital letters. The sums of the sampling dates and their squares are of course straightforward to evalulate, the remaining quantities that depend on the choice of the focal node n can be calculated in one pre-order transversal. Let p denote the parent node of node n

$$\begin{aligned}
\Theta_n &= \Theta_p - (N - 2M_n) l_n \\
\Gamma_n &= \Gamma_p + l_n (T - 2\tau_n) \\
\Delta_n &= \Delta_p + 2 l_n \Theta_p - 4 l_n (\theta_n + (N - M_n) l_n) + N l_n^2
\end{aligned} \tag{15}$$

Note that the order in which these calculations are performed matters. The first line, calculating Θ_n adjusts the parent value Θ_p for the fact that the branch leading to node n is transversed by $N - M_n$ path instead of M_n if the root is shifted from p to n. Similarly, Γ_n is calculated from Γ_p by adjusting with the difference of sum of times of subtending and complementary nodes. The corresponding expression for the sum of squared RTT distances is slightly more complicated but still follows from elementary algebra.

With these quantities at hand, the regression, residuals, and r^2 can be straightforwardly calculated from the means and covariances given by

$$\begin{aligned}
\langle d_{n,i} \rangle &= \frac{\Theta_n}{N} \\
\langle d_{n,i} t_i \rangle - \langle d_{n,i} \rangle \langle t_i \rangle &= \frac{\Gamma_n}{N} - \frac{\Theta_n T}{N^2} \\
\langle d_{n,i}^2 \rangle - \langle d_{n,i} \rangle^2 &= \frac{\Delta_n}{N} - \frac{\Theta_n}{N^2}
\end{aligned} \tag{16}$$

In general, the optimal root is not going to coincide with a preexisting node but will be placed somewhere along a branch. When placing the root at a position $\epsilon \in [0, 1]$ along the branch, the corresponding $\Theta_n(\epsilon)$, $\Gamma_n(\epsilon)$, $\Delta_n(\epsilon)$ are obtained by substitution ϵl_c for l_c in Equation (15). The fraction of variance explained by a RTT regression with a root placed at position ϵ on a branch then has the generic form

$$r^2 = \frac{(a + b\epsilon)^2}{r + s\epsilon + t\epsilon^2} \tag{17}$$

where the coefficients a, b, r, s and r can be obtained by substituting the expressions for $\Theta_n(\epsilon)$, $\Gamma_n(\epsilon)$, and $\Delta_n(\epsilon)$. The term $a + b\epsilon$, for example, evaluates to

$$\begin{aligned}
a + b\epsilon = &\quad \langle d_{n,i} t_i \rangle - \langle d_{n,i} \rangle \langle t_i \rangle \\
= &\quad \frac{\Gamma_n(\epsilon)}{N} - \frac{\Theta_n(\epsilon) T}{N^2} .
\end{aligned} \tag{18}$$

Substituting $\Theta_n(\epsilon)$ and $\Gamma_n(\epsilon)$ and collecting terms by powers of ϵ, the coefficients a and b can be read off. The condition for a maximum $\frac{dr^2}{d\epsilon} = 0$ results in an quadratic equation for ϵ. Hence, the optimal position of the root can be calculated with a number of operations that increases linearly in the size of the tree.

The slope of the RTT regression or clock rate in then simply $\alpha = \frac{\langle d_i t_i \rangle - \langle d_i \rangle \langle t_i \rangle}{\langle t_i^2 \rangle - \langle t_i \rangle^2}$, where d_i are evaluated with respect to the optimal root.