Efficient Recycled Algorithms for Quantitative Trait Models on Phylogenies

Gordon Hiscott[1], Colin Fox[2], Matthew Parry[1], and David Bryant[1,*]

[1]Department of Mathematics and Statistics, University of Otago, Dunedin, New Zealand

[2]Department of Physics, University of Otago, Dunedin, New Zealand

*Corresponding author: E-mail: david.bryant@otago.ac.nz.

Accepted: March 15, 2016

Abstract

We present an efficient and flexible method for computing likelihoods for phenotypic traits on a phylogeny. The method does not resort to Monte Carlo computation but instead blends Felsenstein's discrete character pruning algorithm with methods for numerical quadrature. It is not limited to Gaussian models and adapts readily to model uncertainty in the observed trait values. We demonstrate the framework by developing efficient algorithms for likelihood calculation and ancestral state reconstruction under Wright's threshold model, applying our methods to a data set of trait data for extrafloral nectaries across a phylogeny of 839 Fabales species.

Key words: likelihood algorithm, quantitative traits, continuous traits, comparative method, numerical quadrature, numerical integration.

Introduction

Statistical models for nucleotide or amino acid mutations and substitutions, and the algorithms for computing with them, are fundamental to the study of molecular evolution and biology. As we widen our focus from the evolution of genes to the evolution of genomes, individuals, and populations, a whole new class of modeling challenges present themselves. These include the development of realistic quantitative models for traits which vary over a continuous range of values (O'Meara 2012). Of course, the usefulness of any new model is contingent on the tools available to compute with them. The main contribution of this article is to show how, by combining ideas from statistical phylogenetics and numerical mathematics, we can compute efficiently with a far larger range of evolutionary models.

The algorithms we develop are for computation of the likelihood, that is the probability of the data given the phylogeny, evolutionary model and parameters. If we are working with an evolutionary model with only a small (finite) number of states, then likelihoods can be computed using the dynamic programming algorithm of Felsenstein (1981a). We will show how to extend this algorithm to also compute likelihoods for (essentially) arbitrary continuous trait models.

There is already a wide range of evolutionary phenomena that are studied using continuous trait models. Much of comparative genomics relies on implicit or explicit models for the evolution of morphology (Stevens 1991; Felsenstein 2002; Ronquist 2004; Harmon et al. 2010; O'Meara 2012), many of which make gross simplifying assumptions about how traits vary over time. Continuous evolutionary models have been used in comparative transcriptomics to study heritable aspects of gene expression levels (Khaitovich et al. 2005, 2006), an area with exceptional promise given recent improvements in accuracy and the ability to sample in situ (Voelckel et al. 2002).

Continuous trait models will be of growing importance in evolutionary studies of whole-genome single nucleotide polymorphism-databases. Inference methods based on the coalescent such as SNAPP (Bryant et al. 2012) do not scale well as the number of individuals grows, while those based on continuous models of gene frequencies (Cavalli-Sforza and Edwards 1967; Felsenstein 1981b; Sirén et al. 2011) depend only on proportions of populations with each allele, so scale extremely well. In addition, it is often easier to model the effect of selection on continuous gene frequency models than with the coalescent. Continuous evolutionary models have also been applied successfully to the study of ancestral geography distributions (Lemey et al. 2010).

Our interest is in developing techniques used to compute with these models, and to expand the range of models we can work with. Early work of Felsenstein (1968, 1973), revisited by Freckleton (2012) and FitzJohn (2012), demonstrated that if traits are evolving according to Brownian motion then we can compute likelihoods quickly and (up to numerical precision) exactly. Felsenstein's approach extends to other Gaussian processes, notably the Ornstein–Uhlenbeck (OU) process (Lande 1976; Felsenstein 1988; Hansen 1997), and for several decades, Gaussian models were used almost exclusively to model the evolution of quantitative traits. Ho and Ané (2014) used clever algebraic techniques to develop an alternative algorithm for computing the likelihood and related quantities. They survey several other models which can be handled using the same approach.

These methods are very efficient, and when they can be used, they should be used. The drawback of these methods is that they are fundamentally restricted to models which are Gaussian processes or transforms of Gaussian processes, where the computational bottleneck lies in the computation of a quadratic form involving the covariance matrix of Ho and Ané (2014). Many evolutionary models cannot be handled within this framework (e.g., Ronquist 2004; Landis et al. 2013). Some of the properties of Gaussian processes are quite restrictive: Gaussian processes have single modes, so can only model adaptive landscapes with single peaks; Brownian motion has independent increments, so the rate of change is independent of the value of a trait. The standard strategy for computing with non-Gaussian models is to resort to Monte-Carlo strategies. Even when we are working with a model satisfying the assumptions of Ho and Ané (2014), the algorithms they describe do not give an efficient method for integrating over sets of trait values at the tips, as in the threshold models we discuss below.

Computing the probability of quantitative character evolution may be framed as a numerical integration (quadrature) problem. For most models, if we know the value of the trait at each ancestral node in the phylogeny, we can quickly compute the various transition probabilities. Because we do not usually know these ancestral trait values we integrate them out. This is a multidimensional integration problem with one dimension for each ancestral node (or two dimensions for each node if we are modeling covarying traits) see Felsenstein (2004).

Methods for estimating or approximating integrals are usually judged by their "rate of convergence": how quickly the error of approximation decreases as the amount of work (function evaluations) increases. Consider the problem of computing a one-dimensional integral

$$\int_0^1 f(x)\mathrm{d}x \tag{1}$$

where f is a "nice" function with continuous and bounded derivatives. Simpson's rule, a simple textbook method reviewed below, can be shown to have an $O(N^{-4})$ rate of convergence, meaning that, asymptotically in N, evaluating ten times more points reduces the error by a factor of 10^4. In contrast, a standard Monte Carlo method has a rate of convergence of $O(N^{-\frac{1}{2}})$, meaning that evaluating ten times more points will only reduce the error by a factor of around 3. For this reason, numerical analysis texts often refer to Monte Carlo approaches as "methods of last resort."

Despite this apparently lacklustre performance guarantee, Monte Carlo methods have revolutionized phylogenetics in general and the analysis of quantitative characters in particular. The reason is their partial immunity to the curse of dimensionality. Methods like Simpson's rule are not practical for a high number of dimensions as the asymptotic convergence rate, quoted above, is only achieved for an infeasibly large number of function evaluations N. The effective convergence rate for small N can be very poor, and typically worse than Monte Carlo. In contrast, there are Monte Carlo approaches which achieve close to $O(N^{-\frac{1}{2}})$ convergence irrespective of dimension. This has been critical when computing the likelihoods of complex evolutionary models with as many dimensions as there are nodes in the phylogeny.

The main contribution of our article is to demonstrate how to efficiently and accurately compute likelihoods on a phylogeny using a sequence of one-dimensional integrations. We obtain a fast algorithm with convergence guarantees that far exceed what can be obtained by Monte Carlo integration. Our approach combines two standard tools: classical numerical integrators and Felsenstein's pruning algorithm for discrete characters (Felsenstein 1981a). Indeed, the only real difference between our approach and Felsenstein's discrete character algorithm is that we use numerical integration techniques to integrate states at ancestral nodes, instead of just carrying out a summation.

The running time of the algorithm is $O(N^2n)$, where N is the number of points used in the numerical integration at each node and n is the number of taxa (leaves) in the tree. Using Simpson's method, we obtain a convergence rate of $O(nN^{-4})$, meaning that if we increase N by a factor of 10, we will obtain an estimate which is accurate to four more decimal places.

To illustrate the application of our general framework, we develop an efficient algorithm for computing the likelihood of a tree under the threshold model of Wright (1934) and Felsenstein (2005, 2012). We also show how to infer marginal trait densities at ancestral nodes. We have implemented these algorithms and used them to study evolution of extrafloral nectaries (EFN) on an 839-taxon phylogeny of Marazzi et al. (2012). MATLAB code for computing the threshold likelihood has been posted on MATLAB Central and complete MATLAB code for all analyses and simulations can be found in supplementary material, Supplementary Material online.

The combination of numerical integrators and the pruning algorithm opens up a large range of potential models and approaches which we have only just begun to explore. It may well be that Gaussian type models provide good approximations in many contexts, however the extent to which this is true will be unknown until we have computational tools for handling richer models.

Materials and Methods

Models for Continuous Trait Evolution

Phylogenetic models for continuous trait evolution, like those for discrete traits, are specified by the density of trait values at the root and the transition densities along the branches. We use $f(x_r|\theta_r)$ to denote the density for the trait value at the root, where θ_r is a set of relevant model parameters. We use $f(x_i|x_j, \theta_i)$ to denote the transitional density for the value at node *i*, conditional on the trait value at its parent node *j*. Here, θ_i represents a bundle of parameters related to node *i* such as branch length, population size, and mutation rate. All of these parameters can vary throughout the tree.

To see how the model works, consider how continuous traits might be simulated. A state X_r is sampled from the root density $f(X_r|\theta_r)$. We now proceed through the phylogeny from the root to the tips, each time visiting a node only after its parent has already been visited. For each node *i*, we generate the value at that node from the density $f(X_i|x_j, \theta_v)$, where x_j is the simulated trait value at node *j*, the parent of node *i*. In this way, we will eventually generate trait values for the tips.

We use $X_1, \ldots, X_n$ to denote the random trait values at the tips and $X_{n+1}, \ldots, X_{2n-1}$ to denote the random trait values at the internal nodes, ordered so that children come before parents. Hence, X_{2n-1} is the state assigned to the root. Let

$$\mathcal{E}(T) = \{(i, j) : \text{node } i \text{ is a child of node } j\} \qquad (2)$$

denote the set of branches in the tree. The joint density for all trait values, observed and ancestral, is given by multiplying the root density with all of the transition densities

$$\begin{aligned} &f(x_1, \ldots, x_n, x_{n+1}, \ldots, x_{2n-1}|\theta) \\ &\quad = f(x_{2n-1}|\theta) \prod_{(i,j)\in\mathcal{E}(T)} f(x_i|x_j, \theta_i). \end{aligned} \qquad (3)$$

The probability of the observed trait values $x_1, \ldots, x_n$ is now determined by integrating out all of the ancestral trait values:

$$\begin{aligned} \mathcal{L}(T) = f(x_1, \ldots, x_n|\theta) = \int\int \cdots \int f(x_{2n-1}|\theta_r) \\ \prod_{(i,j)\in\mathcal{E}(T)} f(x_i|x_j, \theta_i)\mathrm{d}x_{n+1}, \ldots, \mathrm{d}x_{2n-1}. \end{aligned} \qquad (4)$$

In these integrals, the bounds of integration will vary according to the model.

The oldest, and most widely used, continuous trait models assume that traits (or transformed gene frequencies) evolve like Brownian motion (Cavalli-Sforza and Edwards 1967; Felsenstein 1973). For these models, the root density $f(x_r|\theta)$ is Gaussian (normal) with mean 0 and unknown variance σ_r^2. The transition densities $f(x_i|x_j, \theta_v)$ are also Gaussian, with mean x_j (the trait value of the parent) and variance proportional to branch length. Note that there are identifiability issues which arise with the inference of the root position under this model, necessitating a few tweaks in practice (see the discussion in Chapter 23 of Felsenstein 2004).

It can be shown that when the root density and transitional densities are all Gaussian, the joint density (4) is multivariate Gaussian. Furthermore, the covariance matrix for this density has a special structure which methods such as the pruning techniques of Felsenstein (1968, 1973), Freckleton (2012), and FitzJohn (2012) exploit, as does the top-down approach of Ho and Ané (2014). This general approach continues to work when Brownian motion is replaced by an OU process (Lande 1976; Felsenstein 1988; Hansen 1997), or indeed to many linear or generalized linear models.

Gaussian models, and their relatives, are mathematically and computationally convenient, but rely on assumptions which are unrealistic and inappropriate in many contexts. Numerous researchers have implemented models which do not fit into the general Gaussian framework; most have resorted to Monte Carlo computation to carry out their analyses.

Landis et al. (2013) discuss a class of continuous trait models which are based on *Lévy processes* and include jumps. At particular times, as governed by a Poisson process, the trait value jumps to a value drawn from a given density. Examples include a *compound Poisson process* with Gaussian jumps and a *Variance Gamma* model given by Brownian motion with time varying according to a gamma process. Both of these processes have analytical transition probabilities in some special cases.

Lepage et al. (2006) use the Cox–Ingersoll–Ross (CIR) process to model rate variation across a phylogeny. Like the OU process (but unlike Brownian motion), the CIR process is ergodic. It has a stationary Gamma density which can be used for the root density. The transition density is a particular noncentral chi-squared density and the process only assumes positive values.

Kutsukake and Innan (2013) examine a family of compound Poisson models, focusing particularly on a model where the trait values make exponentially distributed jumps upwards or downwards. In the case that the rates of upward and downward jumps are the same, the model has jumps that follow a double exponential distribution. Kutsukake and Innan

(2013) use approximate Bayesian computation to carry out inference.

Sirén et al. (2011) propose a simple and elegant model for gene frequencies whereby the root value is drawn from a Beta distribution and each transitional density is Beta with appropriately chosen parameters.

Trait values at the tips are not always observed directly. A simple, but important, example of this is the threshold model of Wright (1934), explored by Felsenstein (2005). Under this model, the trait value itself is censored and we only observe whether or not the value is positive or negative. A similar complication arises when dealing with gene frequency data as we typically do not observe the actual gene frequency but instead a binomially distributed sample based on that frequency (Sirén et al. 2011).

If the trait values at the tip are not directly observed we integrate over these values as well. Let $\pi(z_i|x_i)$ denote the probability of observing z_i given the trait value x_i. The marginalized likelihood is then

$$\mathcal{L}(T|z_1,\ldots,z_n) = \int\int\cdots\int f(x_r|\theta) \prod_{(i,j)\in\mathcal{E}(T)} f(x_i|x_j,\theta_v) \prod_{i=1}^{n} \pi(z_i|x_i) \mathrm{d}x_1,\ldots,\mathrm{d}x_{2n-1}. \tag{5}$$

Numerical Integration

Analytical integration can be difficult or impossible. For the most part, it is unusual for an integral to have an analytical solution and there is no general method for finding it when it does exist. In contrast, numerical integration techniques (also known as numerical quadrature) are remarkably effective and are often easy to implement. A numerical integration method computes an approximation of the integral from function values at a finite number of points. Hence, we can obtain approximate integrals of functions even when we do not have an equation for the function itself. See Cheney and Kincaid (2012) for an introduction to numerical integration, and Dahlquist and Björck (2008) and Davis and Rabinowitz (1984) for more comprehensive technical surveys.

The idea behind most numerical integration techniques is to approximate the target function using a function which is easy to integrate. In this article, we will restrict our attention to Simpson's method which approximates the original function using piecewise quadratic functions. To approximate an integral $\int_a^b f(x)\mathrm{d}x$ we first determine $N+1$ equally spaced points (N even)

$$\begin{aligned} &x_0 = a,\ x_1 = a + \frac{b-a}{N},\ x_2 = a + 2\frac{b-a}{N},\ldots, \\ &x_k = a + k\frac{b-a}{N},\ldots,x_N = b. \end{aligned} \tag{6}$$

We now divide the integration into $N/2$ intervals

$$\int_a^b f(x)\mathrm{d}x = \sum_{\ell=1}^{N/2} \int_{x_{2\ell-2}}^{x_{2\ell}} f(x)\mathrm{d}x. \tag{7}$$

Within each interval $[x_{2\ell-2}, x_{2\ell}]$, there is a unique quadratic function which equals $f(x)$ at each the three points $x = x_{2\ell-2}, x = x_{2\ell-1},$ and $x = x_{2\ell}$. The integral of this quadratic on the interval $[x_{2\ell-2}, x_{2\ell}]$ is

$$\frac{(b-a)}{3N}(f(x_{2\ell-2}) + 4f(x_{2\ell-1}) + f(x_{2\ell})) \tag{8}$$

Summing over ℓ, we obtain the approximation

$$\int_a^b f(x)\mathrm{d}x \approx \sum_{\ell=1}^{N/2} \frac{(b-a)}{3N}(f(x_{2\ell-2}) + 4f(x_{2\ell-1}) + f(x_{2\ell})). \tag{9}$$

With a little rearrangement, the approximation can be written in the form

$$\int_a^b f(x)\mathrm{d}x \approx \frac{(b-a)}{N}\sum_{k=0}^{N} w_k f(x_k) \tag{10}$$

where $w_k = 4/3$ when k is odd and $w_k = 2/3$ when k is even, with the exception of w_0 and w_N which both equal 1/3. Simpson's method is easy to implement and has a convergence rate of $O(N^{-4})$. Increasing the number of intervals by a factor of 10 decreases the error by a factor of 10^{-4}. See Dahlquist and Björck (2008) and Davis and Rabinowitz (1984) for further details.

It should be remembered, however, that the convergence rate is still only an asymptotic bound, and gives no guarantees on how well the method performs for a specific function and choice of N. Simpson's method, for example, can perform quite poorly when the function being integrated has rapid changes or sharp peaks. We observed this behavior when implementing threshold models, as described below. Our response was to better tailor the integration method for the functions appearing. We noted that the numerical integrations we carried out all had the form

$$\int_a^b e^{-\frac{(x-\mu)^2}{2\sigma^2}} f(x)\mathrm{d}x \tag{11}$$

where μ and σ varied. Using the same general approach as Simpson's rule, we approximated $f(x)$, rather than the whole function $e^{-\frac{(x-\mu)^2}{2\sigma^2}} f(x)$, by a piecewise quadratic function $p(x)$. We could then use standard techniques and tools to evaluate $\int_a^b e^{-\frac{(x-\mu)^2}{2\sigma^2}} p(x)\mathrm{d}x$ numerically. The resulting integration formula, which we call the "Gaussian kernel method," gives a significant improvement in numerical accuracy.

A further complication is that, in models of continuous traits, the trait value often ranges over the whole real line, or at least over the set of positive reals. Hence, we need to approximate integrals of the form

$$\int_{-\infty}^{\infty} f(x)\mathrm{d}x \text{ or } \int_{0}^{\infty} f(x)\mathrm{d}x \tag{12}$$

though the methods discussed above only apply to integrals on finite intervals. We truncate these integrals, determining values U and L such that the difference

$$\int_{-\infty}^{\infty} f(x)\mathrm{d}x - \int_{L}^{U} f(x)\mathrm{d}x \tag{13}$$

between the full integral $\int_{-\infty}^{\infty} f(x)\mathrm{d}x$ and the truncated integral $\int_{L}^{U} f(x)\mathrm{d}x$ can be bounded analytically. Other strategies are possible; see Dahlquist and Björck (2008) for a comprehensive review.

A Pruning Algorithm for Integrating Continuous Traits

Felsenstein has developed pruning algorithms for both continuous and discrete characters (Felsenstein 1981a,b). His algorithm for continuous characters works only for Gaussian processes. Our approach is to take his algorithm for discrete characters and adapt it to continuous characters.

The (discrete character) pruning algorithm is an application of dynamic programming. For each node i, and each state x, we compute the probability of observing the states for all tips which are descendants of node i, conditional on node i having ancestral state x. This probability is called the partial likelihood at node i given state x. Our algorithm follows the same scheme, with one major difference. Since traits are continuous, we cannot store all possible partial likelihoods. Instead, we store likelihoods for a finite set of values and plug these values into a numerical integration routine.

Let i be the index of a node in the tree not equal to the root, let node j be its parent node. We define the partial likelihood, $\mathcal{F}_i(x_j)$, to be the likelihood for the observed trait values at the tips which are descendants of node i, conditional on the parent node j having trait value x_j. If node i is a tip with observed trait value x_i we have

$$\mathcal{F}_i(x_j) = f(x_i|x_j, \theta_i) \tag{14}$$

recalling that $f(x_i|x_j, \theta_i)$ is the density for the value of the trait at node i conditional on the value of the trait for its parent. More generally, we may only observe some value z_i for which we have the conditional probability $\pi(z_i|x_i)$ conditional on the trait value x_i. In this case, the partial likelihood is given by

$$\mathcal{F}_i(x_j) = \int f(\tilde{x}_i|x_j, \theta_i)\pi(z_i|\tilde{x}_i)\mathrm{d}\tilde{x}_i. \tag{15}$$

Suppose node i is not the root and that it has two children u and v. Since trait evolution is conditionally independent on disjoint subtrees, we obtain the recursive formula

$$\mathcal{F}_i(x_j) = \int f(\tilde{x}_i|x_j, \theta_i)\mathcal{F}_u(\tilde{x}_i)\mathcal{F}_v(\tilde{x}_i)d\tilde{x}_i. \tag{16}$$

Finally, suppose that node i is the root and has two children u and v. We evaluate the complete tree likelihood using the density of the trait value at the root,

$$\mathcal{L}(T) = \int f(x|\theta_r)\mathcal{F}_u(x)\mathcal{F}_v(x)\mathrm{d}x. \tag{17}$$

The bounds of integration in (15)–(17) will vary according to the model.

We use numerical integration techniques to approximate (15)–(17) and dynamic programming to avoid an exponential explosion in the computation time. Let N denote the number of function evaluations for each node. In practice, this might vary over the tree, but for simplicity we assume that it is constant. For each node i, we select $N+1$ trait values

$$X_i[0] < X_i[1] < \cdots < X_i[N]. \tag{18}$$

How we do this will depend on the trait model and the numerical integration technique. If, for example, the trait values vary between a and b and we are applying Simpson's method with N intervals we would use $X_i[k] = a + \frac{b-a}{N}k$ for $k = 0, 1, 2, \ldots, N$.

We traverse the tree starting at the tips and working toward the root. For each nonroot node i and $k = 0, 1, \ldots, N$ we compute and store an approximation $F_i[k]$ of $\mathcal{F}_i(X_j[k])$, where node j is the parent of node i. Note that this is an approximation of $\mathcal{F}_i(X_j[k])$ rather than of $\mathcal{F}_i(X_i[k])$ since $\mathcal{F}_i(x)$ is the partial likelihood conditional on the trait value for the parent of node i. The value approximation $F_v[i]$ is computed by applying the numerical integration method to the appropriate integral (15)–(17), where we replace function evaluations with approximations previously computed. See below for a worked example of this general approach.

The numerical integration methods we use run in time linear in the number of points being evaluated. Hence, if n is the number of tips in the tree, the algorithm will run in time $O(nN^2)$. For the integration techniques described above, the convergence rate (in N) for the likelihood on the entire tree had the same order as the convergence rate for the individual one-dimensional integrations (see below for a formal proof of a specific model). We have therefore avoided the computational blow-out typically associated with such high-dimensional integrations, and achieve this without sacrificing accuracy.

Posterior Densities for Ancestral States

The algorithms we have described compute the joint density of the states at the tips, given the tree, the branch lengths, and other parameters. As with discrete traits, the algorithms can be modified to infer ancestral states for internal nodes in the tree. Here, we show how to carry out reconstruction of the marginal posterior density of a state at a particular node. The differences between marginal and joint reconstructions are reviewed in Yang (2006, p. 121).

First consider marginal reconstruction of ancestral states at the root. Let u and v be the children of the root. The product $\mathcal{F}_u(x)\mathcal{F}_v(x)$ equals the probability of the observed character conditional on the tree, branch lengths, parameters, and a state of x at the root. The marginal probability of x, ignoring the data, is given by the root density $f(x|\theta_r)$. Integrating the product of $\mathcal{F}_u(x)\mathcal{F}_v(x)$ and $f(x|\theta_r)$ gives the likelihood $\mathcal{L}(T)$, as in (17). Plugging these into Bayes' rule, we obtain the posterior density of the state at the root:

$$f(x_r|z_1,\ldots,z_n) = \frac{\mathcal{F}_u(x_r)\mathcal{F}_v(x_r)f(x_r|\theta_r)}{\mathcal{L}(T)}. \tag{19}$$

With general time reversible models used in phylogenetics, the posterior distributions at other nodes can be found by changing the root of the tree. Unfortunately, the same trick does not work for many quantitative trait models. Furthermore, recomputing likelihoods for each possible root entails a large amount of unnecessary computation.

Instead, we derive a second recursion, this one starting at the root and working toward the tips. A similar trick is used to compute derivatives of the likelihood function in Felsenstein and Churchill (1996). For a node i and state x we let $\mathcal{G}_i(x)$ denote the likelihood for the trait values at tips which are not descendants of node i, conditional on node i having trait value x. If node i is the root r, then $\mathcal{G}_r(x)$ is 1 for all x.

Let node i be any node apart from the root, let node j be its parent and let node u be the other child of j (that is, the sibling of node i). We let $\tilde{x}$ denote the trait value at node j. Then $\mathcal{G}_i(x)$ can be written

$$\mathcal{G}_i(x) = \int f(\tilde{x}|x,\theta_i)\mathcal{G}_j(\tilde{x})\mathcal{F}_u(\tilde{x})d\tilde{x}. \tag{20}$$

This integral can be evaluated using the same numerical integrators used when computing likelihoods. Note that $f(\tilde{x}|x,\theta_i)$ is the conditional density of the parent state given the child state, which is the reverse of the transition densities used to formulate the model. It should be noted that while Brownian motion has reversible transition probabilities, the OU process does not. How $\mathcal{G}_i(x)$ is computed will depend on the model and its properties; see below for an implementation of this calculation in the threshold model.

Once $\mathcal{G}_i(x)$ has been computed for all nodes, the actual (marginal) posterior densities are computed from Bayes' rule. Letting u, v be the children of node i,

$$f(x_i|z_1,\ldots,z_n) = \frac{\mathcal{G}_i(x_i)\mathcal{F}_u(x_i)\mathcal{F}_v(x_i)f(x_i)}{\mathcal{L}(T)}. \tag{21}$$

Case study: threshold models

In this section, we show how the general framework can be applied to the threshold model of Wright (1934) and Felsenstein (2005, 2012). Each trait is modeled by a continuously varying "liability" which evolves along branches according to a Brownian motion process. While the underlying liability is continuous, the observed data are discrete: at each tip we observe only whether the liability is above or below some threshold.

We will use standard notation for Gaussian densities. Let $\phi(x|\mu,\sigma^2)$ denote the density of a Gaussian random variable x with mean μ and variance σ^2; let

$$\Phi(y|\mu,\sigma^2) = \int_{-\infty}^{y} \phi(x|\mu,\sigma^2) \tag{22}$$

denote its cumulative density function, with inverse $\Phi^{-1}(\alpha|\mu,\sigma^2)$.

Let $X_1,\ldots,X_{2n-1}$ denote the (unobserved) liability values at the n tips and $n-1$ internal nodes. As above we assume that the $i<j$ whenever node i is a child of node j, so that the root has index $2n-1$.

The liability value at the root has a Gaussian density with mean μ_r and variance σ_r^2:

$$f(x_{2n-1}|\theta_r) = \phi(x_{2n-1}|\mu_r,\sigma_r^2). \tag{23}$$

Consider any nonroot node i and let j be the index of its parent. Let t_i denote the length of the branch connecting nodes i and j. Then X_i has a Gaussian density with mean x_j and variance $\sigma^2 t_v$:

$$f(x_i|x_j,\theta_i) = \phi(x_i|x_j,\sigma^2 t_i). \tag{24}$$

Following Felsenstein (2005), we assume thresholds for the tips are all set at zero. We observe 1 if the liability is positive, 0 if the liability is negative, and ? if data are missing. We can include the threshold step into our earlier framework by defining

$$\pi(z_i|x_i) = \begin{cases} 1 & \text{if } z_i = 1 \text{ and } x_i > 0, \text{or } z_i = 0 \text{ and } x_i \leq 0, \text{or } z_i = ? \\ 0 & \text{otherwise.} \end{cases} \tag{25}$$

The likelihood function for observed discrete values $z_1,\ldots,z_n$ is then given by integrating over liability values for all nodes on the tree:

$$\mathcal{L}(T|z_1,\ldots,z_n) = \int_{-\infty}^{\infty}\cdots\int_{-\infty}^{\infty} \phi(x_{2n-1}|\mu_r,\sigma_r^2) \prod_{(i,j)} \phi(x_i|x_j,\sigma^2 t_i) \prod_{i=1}^{n} \pi(z_i|x_i)\, dx_1 \ldots dx_{2n-1}. \tag{26}$$

The first step toward computing $\mathcal{L}(T|z_1, \dots, z_n)$ is to bound the domain of integration so that we can apply Simpson's method. Ideally, we would like these bounds to be as tight as possible, for improved efficiency. For the moment we will just outline a general procedure which can be adapted to a wide range of evolutionary models.

The marginal (prior) density of a single liability or trait value at a single node is the density for that liability value marginalizing over all other values and data. With the threshold model, the marginal density for the liability at node i is Gaussian with mean μ_r (like the root) and variance v_i equal to the sum of the variance at the root and the transition variances on the path from the root to node i. If P_i is the set of nodes from the root to node i, then

$$v_i = \sigma_r^2 + \sigma^2 \sum_{j \in P_i} t_j. \tag{27}$$

The goal is to constrain the error introduced by truncating the integrals with infinite domain. Let ϵ be the desired bound on this truncation error. Recall that the number of internal nodes in the tree is $n{-}1$. Define

$$L_i = \Phi^{-1}\left(\frac{\epsilon}{2(n-1)}|\mu_r, v_i\right) \tag{28}$$

and

$$U_i = \Phi^{-1}\left(1 - \frac{\epsilon}{2(n-1)}|\mu_r, v_i\right). \tag{29}$$

The bounds L_i and U_i are chosen so that the (marginal) probability X_i lies outside the interval $[L_i, U_i]$ is at most $\epsilon/(n-1)$. For this model, these are given by the inverse distribution function of a Gaussian; other models would involved different transition densities. By the inclusion–exclusion principle, the joint probability $X_i \in [L_i, U_i]$ for any internal node i is at most ϵ. We use this fact to bound the contribution of the regions outside these bounds.

$$\int_{-\infty}^{\infty} \cdots \int_{-\infty}^{\infty} f(x_{2n-1}|\mu_r, \sigma_r^2) \prod_{(u,v)} f(x_v|x_u, \theta_v) \prod_{i=1}^{n} \pi(z_i|x_i) \mathrm{d}x_1 \dots \mathrm{d}x_{2n-1} - \int_{a_{2n-1}}^{b_{2n-1}} \cdots \int_{a_{n+1}}^{b_{n+1}} \int_{-\infty}^{\infty} \cdots \int_{-\infty}^{\infty} f(x_{2n-1}|\mu_r, \sigma_r^2) \prod_{(u,v)} f(x_v|x_u, \theta_v) \prod_{i=1}^{n} \pi(z_i|x_i) \mathrm{d}x_1 \dots \mathrm{d}x_{2n-1} \tag{30}$$

$$\le \int_{-\infty}^{\infty} \cdots \int_{-\infty}^{\infty} f(x_{2n-1}|\mu_r, \sigma_r^2) \prod_{(u,v)} f(x_v|x_u, \theta_v) \mathrm{d}x_1 \dots \mathrm{d}x_{2n-1} - \int_{a_{2n-1}}^{b_{2n-1}} \cdots \int_{a_{n+1}}^{b_{n+1}} \int_{-\infty}^{\infty} \cdots \int_{-\infty}^{\infty} f(x_{2n-1}|\mu_r, \sigma_r^2) \prod_{(u,v)} f(x_v|x_u, \theta_v) \mathrm{d}x_1 \dots \mathrm{d}x_{2n-1} \tag{31}$$

$$\le P(X_{n+1} \notin [L_{n+1}, U_{n+1}] \text{ or } X_{n+2} \notin [L_{n+2}, U_{n+2}] \text{ or } \cdots \text{or} X_{2n-1} \notin [L_{2n-1}, U_{2n-1}]) \tag{32}$$

$$< \epsilon. \tag{33}$$

We therefore compute values L_i, U_i for $n+1 \le i \le 2n-1$ using (28) and (29) repeatedly, and use these bounds when carrying out integration at the internal nodes. We define

$$X_i[k] = L_i + \frac{U_i - L_i}{N} k \tag{34}$$

for $k = 0, 1, \dots, N$ and each internal node i.

The next step is to use dynamic programming and numerical integration to compute the approximate likelihood. Let node i be a tip of the tree, let node j be its parent and let z_i be the binary trait value at this tip. For each $k = 0, 1, \dots, N$ we use standard error functions to compute

$$F_i[k] = \mathcal{F}_i(X_j[k]) \tag{35}$$

$$= \begin{cases} \int_0^{\infty} \phi(\tilde{x}|X_j[k], \sigma^2 t_i) \mathrm{d}\tilde{x} & \text{if } z_i = 1 \\ \int_{-\infty}^{0} \phi(\tilde{x}|X_j[k], \sigma^2 t_i) \mathrm{d}\tilde{x} & \text{if } z_i = 0 \\ 1 & \text{if } z_i = ?. \end{cases} \tag{36}$$

Here, $\phi(x|\mu, \sigma^2)$ is the density of a Gaussian with mean μ and variance σ^2.

Now suppose that node i is an internal node with parent node j and children u and v. Applying Simpson's rule to the bounds L_i, U_i to (16) we have for each $k = 0, 1, \dots, N$:

$$F_i[k] = \frac{U_i - L_i}{N} \sum_{\ell=0}^{N} w_\ell \phi(X_i[\ell]|X_j[k], \sigma^2 t_i) F_u[\ell] F_v[\ell] \tag{37}$$

$$\approx \mathcal{F}_i(X_j[k]). \tag{38}$$

Suppose node i is the root, and u, v are its children. Applying Simpson's rule to (17) gives an approximate likelihood of

$$\frac{U_{2n-1} - L_{n-1}}{N} \sum_{\ell=0}^{N} w_\ell \phi(X_i[\ell]|\mu_r, \sigma_r^2) F_u[\ell] F_v[\ell]. \tag{39}$$

Pseudocode for the algorithm appears in Algorithm 1.

Algorithm 1: Compute probability of a threshold character.

```
Input:
    $N$: Number of intervals in numerical integration.
    $t_1, \ldots, t_{2n-2}$: branch lengths in tree.
    $\mu_r, \sigma_r^2$: mean and variance of root density
    $\sigma^2$: variance of transition densities (per unit branch length)
    $z_1, \ldots, z_n$ observed character ($z_i \in \{+1, 0, ?\}$)
Output:
    Probability $L$ of observed character under the threshold model.

Construct the vector $\mathbf{x} = [0, 1, 2, \ldots, N]/N$.
Construct the vector $\mathbf{w} = [1, 4, 2, 4, 2, \ldots, 4, 2, 1]$ as in (??)
Compute the path length $p_i$ from the root to each node $i$.
Initialize $F_i[k] \leftarrow 1$ for all nodes $i$ and $0 \leq k \leq N$.
For all $i = n+1, n+2, \ldots, 2n-1$
    $L_i \leftarrow \Phi^{-1}(\frac{nN^{-4}}{2(n-1)} | \mu_r, \sigma_r^2 + \sigma^2 p_i)$
    $U_i \leftarrow \Phi^{-1}(1 - \frac{nN^{-4}}{2(n-1)} | \mu_r, \sigma_r^2 + \sigma^2 p_i)$
    $X_i \leftarrow (U_i - L_i)\mathbf{x} + L_i$
For all tip nodes $i = 1, 2, \ldots, n$
    Let $j$ be the index of the parent of node $i$
    For $k = 0, \ldots, N$
        If $z_i = 1$
            $F_i[k] = 1 - \Phi(0; X_j[k], \sigma^2 t_i)$
        else if $z_i = 0$
            $F_i[k] = \Phi(0; X_j[k], \sigma^2 t_i)$
For all internal nodes $i = n+1, \ldots, 2n-2$, excluding the root
    Let $j$ be the index of the parent of node $i$
    Let $u, v$ be the indices of the children of node $i$
    For $k = 0, 1, \ldots, N$
        $F_i[k] \leftarrow \frac{U_i - L_i}{N} \sum_{\ell=0}^{N} \mathbf{w}_\ell \phi(X_i[\ell]; X_j[k], \sigma^2 t_i) F_u[\ell] F_v[\ell]$
Let $u, v$ be indices of the the children of the root.
$L \leftarrow \frac{U_{2n-1} - L_{n-1}}{N} \sum_{\ell=0}^{N} \mathbf{w}_\ell \phi(X_i[\ell]; \mu_r, \sigma_r^2) F_u[\ell] F_v[\ell]$
```

Algorithm 1 Pseudo-code of the likelihood approximation algorithm for a single character, under the threshold model. The nodes are numbered in increasing order from tips to the root.

Regarding efficiency and convergence we have:

Theorem 1 Algorithm 1 runs in $O(nN^2)$ time and approximates L(T) with $O(nN^{-4})$ error.

Proof

The running time follows from the fact that for each of the $O(n)$ nodes in the tree we carry out $O(N)$ applications of Simpson's method.

Simpson's rule has $O(N^{-4})$ convergence on functions with bounded fourth derivatives (Dahlquist and Björck 2008). The root density and each of the transition densities are Gaussians, so individually have bounded fourth derivatives. For each node i, let n_i denote the number of tips which are descendants of the node. Using induction on (16), we see that for all nodes i, the fourth derivative of $\mathcal{F}_i(x)$ is $O(n_i)$.

If we use $\epsilon = nN^{-4}$ in (28) and (29) then replacing the infinite domain integrals with integrals on $[L_i, U_i]$ introduces at most nN^{-4} error. Using a second induction proof on (16) and (37) together with the bound on fourth derivatives, we have that $|\mathcal{F}_i(X_j[k]) - F_i[k]|$ is at most $O(n_i N^{-4})$ for all nodes i, where node j is the parent of node i. In this way we obtain error bound of $O(n_{2n-1}N^{-4}) = O(nN^{-4})$ on the approximation of $\mathcal{L}(T|z_1, \ldots, z_n, \theta)$. □

We can estimate posterior densities using the recursion (20) followed by equation (21). The conditional density

$$f(\tilde{x}|x, \theta_i) = \phi\left(\tilde{x}|\mu_r + \frac{v_j}{v_i}(x - \mu_r), \frac{\sigma^2 t_i v_j}{v_i}\right) \tag{40}$$

can be obtained by plugging the transitional density

$$f(x|\tilde{x}, \theta_i) = \phi(x|\tilde{x}, \sigma^2 t_i) \tag{41}$$

and the two marginal densities (27)

$$f(\tilde{x}) = \phi(\tilde{x}, v_j), f(x) = \phi(x, v_i) \tag{42}$$

into the identity $f(\tilde{x}|x, \theta_i) = f(x|\tilde{x}, \theta_i)\frac{f(\tilde{x})}{f(x)}$. We thereby obtain the recursion

$$\mathcal{G}_i(x) = \int \phi\left(\tilde{x}|\mu_r + \frac{v_j}{v_i}(x - \mu_r), \frac{\sigma^2 t_i v_j}{v_i}\right)\mathcal{G}_j(\tilde{x})\mathcal{F}_u(\tilde{x})d\tilde{x} \tag{43}$$

which we estimate using Simpson's method. Algorithm estimates values of the posterior densities at each node, evaluated using the same set of grid points as used in Algorithm 1. An additional round of numerical integration can be used to obtain posterior means and variances.

Evolutionary Precursors of Plant Extrafloral Nectaries

To study the methods in practice, we reanalyze trait data published by Marazzi et al. (2012), using a fixed phylogeny. Marazzi et al. (2012) introduce and apply a new discrete state model for morphological traits which, in addition to states for presence and absence, incorporates an intermediate "precursor" state. Whenever the intermediate state is observed at the tips it is coded as "absent." The motivation behind the model is that the intermediate state represents evolutionary precursors, changes which are necessary for the evolution of a new state but which may not be directly observed. These precursors could explain repeated parallel evolution of a trait in closely related traits (Marazzi et al. 2012). They compiled a data set recording presence or absence of plant EFNs across a phylogeny of 839 species of Fabales, fitting their models to these data.

The threshold model also involves evolutionary precursors in terms of changes in ancestral liabilities. We use these models, and our new algorithms to analyze the EFN data set. Our analysis also makes use of the time-calibrated phylogeny inferred by Simon et al. (2009), although unlike Marazzi et al. (2012) we ignore phylogenetic uncertainty.

Experimental Protocol

We conduct three separate experiments. For the first experiment, we examine the rate of convergence of the likelihood algorithm as we increase N. This is done for the "All" EFN character (Character 1 in Marazzi et al. [2012]) for a range of

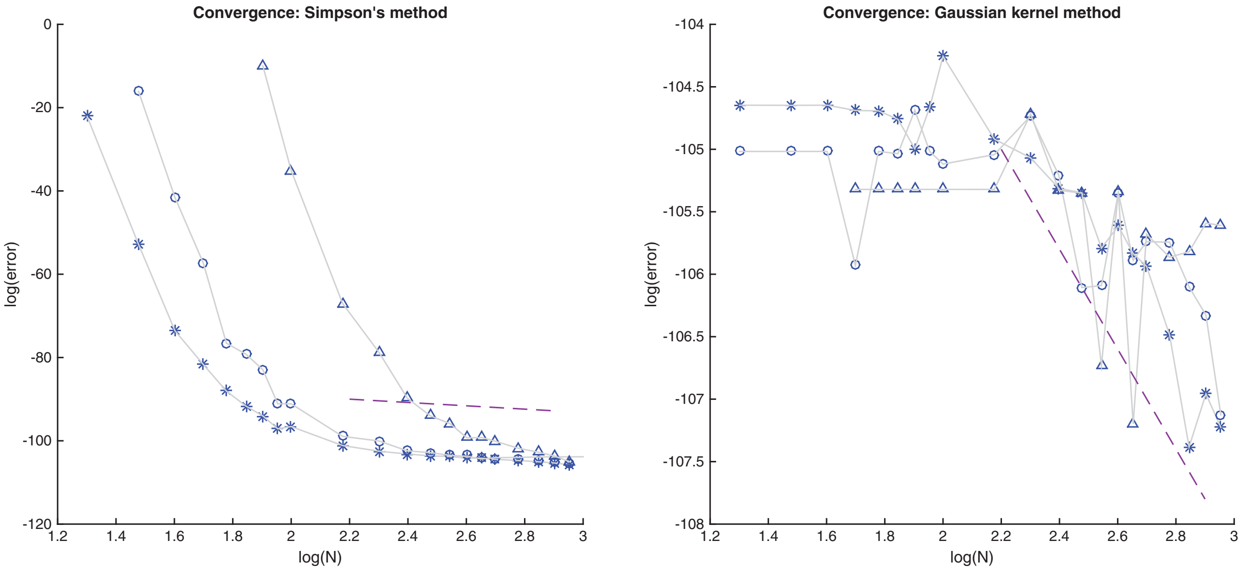

Fig. 1.—Log-log plots of error as a function of N for the dynamic programming algorithm with Simpson's method (left) and with the Gaussian kernel method (right). The likelihoods were computed under the threshold model on EFN trait data for an 839 taxon tree. Dotted lines have slope −4 (corresponding to convergence rate of N^{-4}. Note the difference in scale for the two methods.). Logarithms computed to base 10. Letting h be the height of the tree, the circles in both plots represent errors when $\sigma_r^2 = h$, the asterisks represent errors when $\sigma_r^2 = 0.1h$, and the triangles represent errors when $\sigma_r^2 = 10h$.

estimates for the liability variance at the root, σ_r^2. The interest in σ_r^2 stems from its use in determining bounds L_i, U_i for each node, with the expectation that as σ_r^2 increases, the convergence of the integration algorithm will slow. The mean liability at the root, μ_r, was determined from the data using Maximum Likelihood estimation.

We also examined convergence of the algorithm on randomly generated characters. We first evolved liabilities according to the threshold model, using the parameter settings obtained above. To examine the difference in performance for non-phylogenetic characters, we also simulated binary characters by simulated coin flipping. Twenty replicates were carried out for each case.

The second experiment extends the model comparisons carried out in Marazzi et al. (2012) to include the threshold models. For this comparison we fix the transitional variance σ^2 at one, since changing this values corresponds to a rescaling of the Brownian process, with no change in likelihood. With only one character, the maximum likelihood estimate of the root variance σ_r^2 is zero, irrespective of the data. This leaves a single parameter to infer: the value of the liability at the root state. We computed a maximum likelihood estimate for the state at the root, then applied our algorithm with a sufficiently large value of N to be sure of convergence. The Akaike Information Criterion (AIC) was determined and compared with those obtained for the model of Marazzi et al. (2012).

For the third experiment, we determine the marginal posterior densities for the liabilities at internal nodes, using Algorithm 2.

```
Algorithm 2: Compute posterior densities

Input:
    $N$, $t_1, \ldots 2n-2$, $\mu_r$, $\sigma_r^2$, and $\sigma^2$ as in Algorithm 1
    Vector $p$, likelihood $L$ and arrays $F_i$ computed in Algorithm 1.
Output:
    Arrays $H_i$ for each internal node $i$.
Construct the vectors $\mathbf{x}$, $\mathbf{w}$, $\{L_i : i \in \{n+1, \ldots, 2n-2\}\}$,
    $\{U_i : i \in \{n+1, \ldots, 2n-2\}\}$, and path lengths $p_i$ as in Algorithm 1.
$G_{2n-1}[k] \leftarrow 1$ for all $k$.
For all $i = 2n-2, 2n-3, \ldots, n+1$
    Let $j$ be the index of the parent of node $i$.
    Let $v$ be the index of the sibling of node $i$.
    For $k = 0, 1, \ldots, N$
        $\mu \leftarrow \mu_r + \frac{\sigma_r^2+\sigma^2 p_j}{\sigma_r^2+\sigma^2 p_i}(X_i[k] - \mu_r)$
        $V \leftarrow \frac{\sigma^2 t_i(\sigma_r^2+\sigma^2 p_j)}{\sigma_r^2+\sigma^2 p_i}$
        $G_i[k] \leftarrow \frac{U_j - L_j}{N} \sum_{\ell=0}^{N} \mathbf{w}_\ell \phi(X_j[\ell]; \mu, V) G_j[\ell] F_v[\ell]$
For all $i = n+1, \ldots, 2n-1$
    Let $u, v$ be the children of node $i$.
    For all $k = 0, 1, \ldots, N$
        $H_i[k] \leftarrow \frac{1}{L} G_i[k] F_u[k] F_v[k] \phi(X_i[k] | \mu_r, \sigma_r^2 + \sigma^2 p_i)$
```

Algorithm 2 Pseudocode for the algorithm to efficiently compute ancestral posterior densities under the threshold model. At the termination of the algorithm, $H_i[k]$ is an estimate of the posterior density at internal node i, evaluated at $x = X_i[k]$.

These posterior probabilities are then mapped onto the phylogeny, using shading to denote the (marginal) posterior probability that a liability is larger than zero. We therefore obtain a figure analogous to supplementary figure S7, Supplementary Material online, of Marazzi et al. (2012).

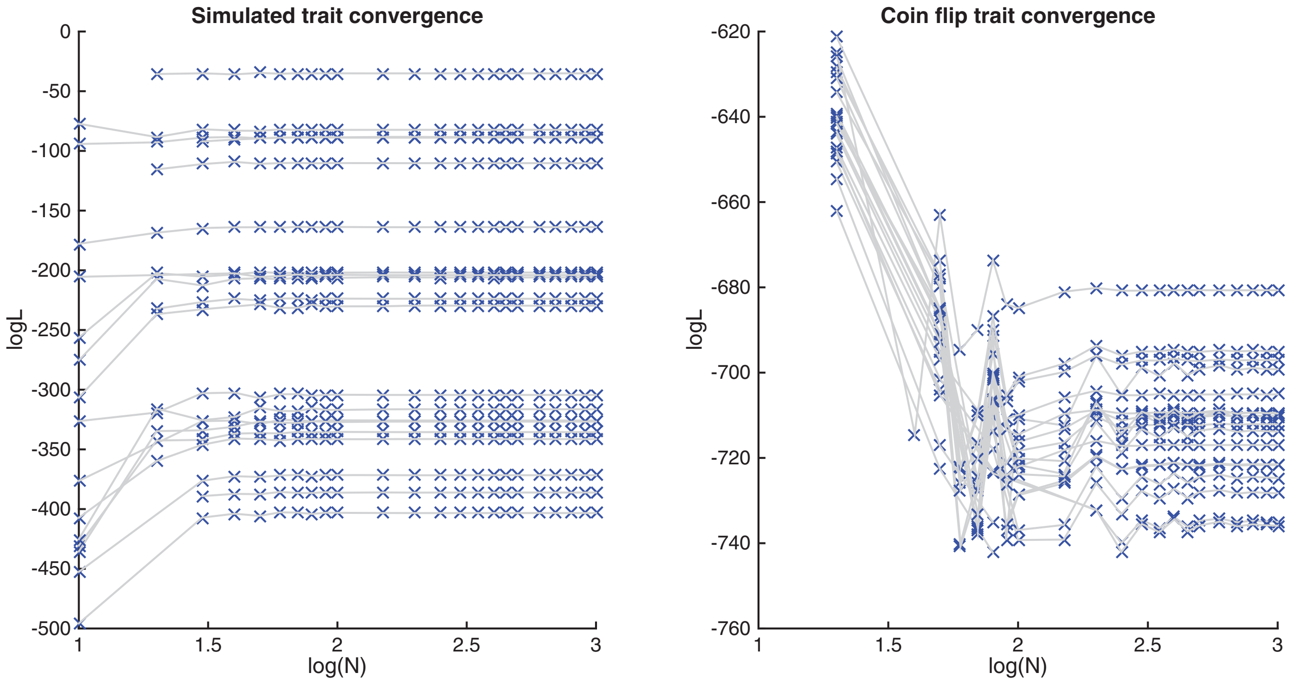

Fig. 2.—Plots of log-likelihood values as a function of *log* (*N*) for the two types of data simulated from the fixed EFN tree, computed using our algorithm together with the Gaussian kernel method. Logarithms computed to base 10.

Results

Convergence of the Algorithm

To examine convergence, we compute the absolute error of each likelihood approximation because the actual likelihood is not available we use the approximation when $N = 1{,}000$. Plots of error versus N are given in figure 1, both for Simpson's method (left) and for the modified Gaussian kernel method (right). For larger N, the error in a log-log plot decreases with slope at most -4 (as indicated), corresponding to N^{-4} convergence of the method. Log-log plots of error versus N for the simulated data are given in figure 2. In each case, the method converges for by $N \approx 30$.

While the level of convergence for both algorithms is correct, the accuracy of the method based on Simpson's method is far worse. When a branch length is short, the transition density becomes highly peaked, as does the function being integrated. Such functions are difficult to approximate with piecewise quadratics, and Simpson's method can fail miserably. Indeed, for $N < 50$, we would often observe estimated probabilities equal to 0, or estimates greater than 1! (These were omitted from the plots). Although we can always bound estimates computed by the algorithm, a sounder approach is to improve the integration technique. This we did using the Gaussian kernel method, and the result was far improved accuracy for little additional computation. For the remainder of the experiments with this model we used the Gaussian kernel method when carrying out numerical integration.

Table 1

Table of Log-Likelihood and AIC Values for the Binary Character, Precursor, and Threshold Models on Six EFN Traits

Trait	Model	*k*	*log L*	AIC
1 (All)	Binary	2	−251.7	507.4
	Precursor	1	−246.7	495.4
	Threshold	1	−240.6	**483.2**
2 (Leaves)	Binary	2	−240.3	484.6
	Precursor	1	−234.5	470.9
	Threshold	1	−230.6	**463.1**
3 (Inflorescence)	Binary	2	−108.3	220.5
	Precursor	1	−110.9	223.9
	Threshold	1	−108.3	**218.5**
4 (Trichomes)	Binary	2	−86.7	177.3
	Precursor	1	−86.9	175.9
	Threshold	1	−85.8	**173.5**
5 (Substitutive)	Binary	2	−163.0	330.1
	Precursor	1	−161.6	325.3
	Threshold	1	−161.3	**324.6**
6 (True)	Binary	2	−132.3.1	268.7
	Precursor	1	−131.1	264.3
	Precursor	2	−126.7	257.3
	Threshold	1	−125.3	**252.6**

Note.—Column *k* indicates numbers of parameters for each model. Data for the binary and precursor models copied from table 1 in Marazzi et al. (2012). All likelihoods and AIC values rounded to 1 d.p. Boldface indicates the best fitting model for each trait. A pre-cursor model with one parameter was used for all experiments, except for trait 6 where a two-parameter model gave a better AIC than the one-parameter model (see discussion in Marazzi et al. (2012).

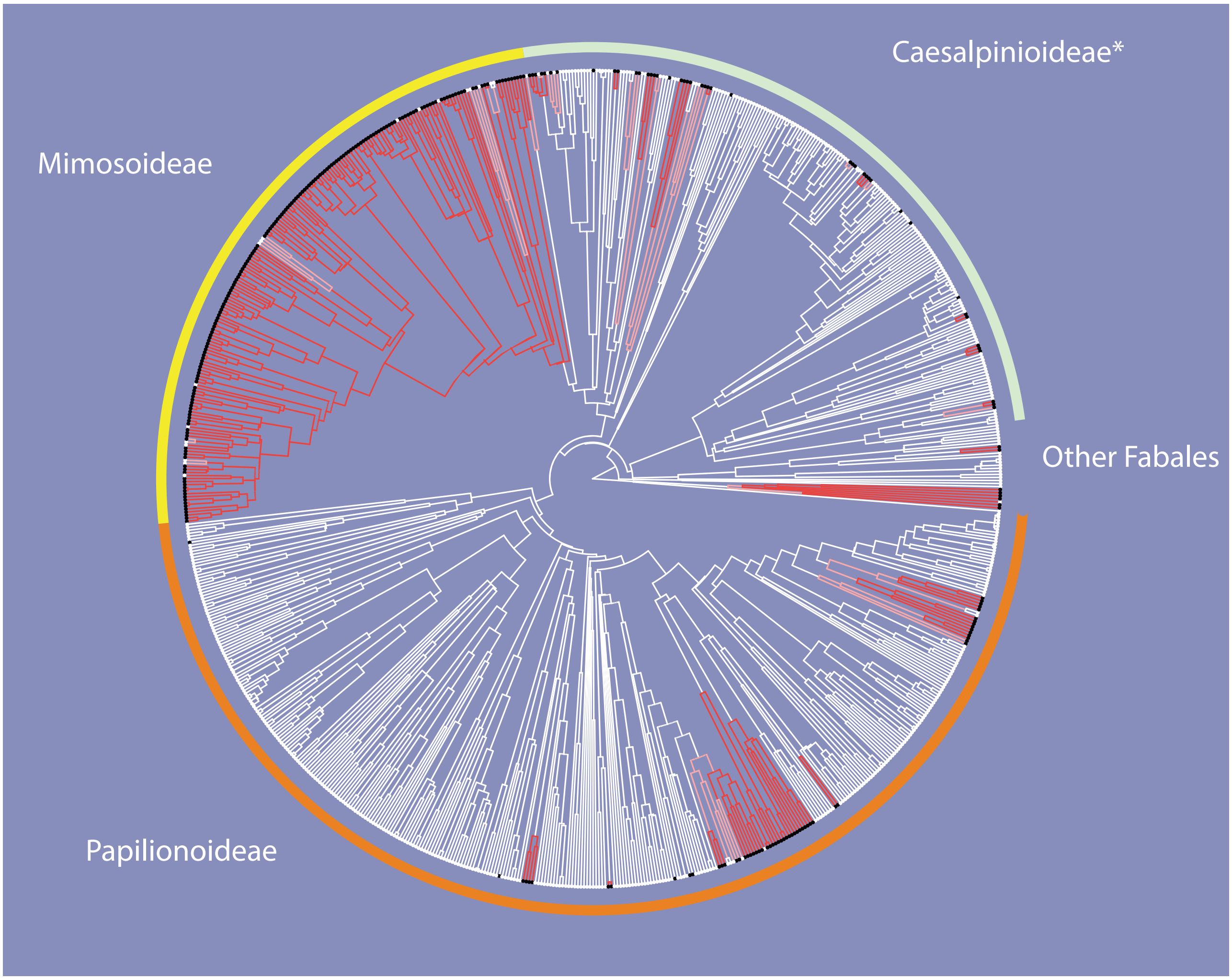

Fig. 3.—Marginal posterior probabilities for the liabilities, for EFN trait 1 of Marazzi et al. (2012) on the phylogeny inferred by Simon et al. (2009). Lineages with posterior probability > 0.7 colored red, lineages with posterior probability < 0.3 colored white, and remaining lineages colored pink.

Model Comparison

Marazzi et al. (2012) describe AIC comparisons between their precursor model and a conventional binary trait model. We extend this comparison to include the threshold model. This is a one parameter model, the parameter being the value of the liability at the root. We used the MATLAB command fminsearch with multiple starting points to compute the maximum likelihood estimate for this value. The resulting log-likelihood was $log?L = -240.6$, giving an AIC of 483.2. This compares to an AIC of 507.4 for the (two parameter) binary character model and an AIC of 495.4 for the (one parameter) precursor model of Marazzi et al. (2012).

We analyzed the five other EFN traits in the same way, and present the computed AIC values in table 1, together with AIC values for the two parameter binary state model and one parameter precursor model computed by Marazzi et al. (2012) (and the two parameter precursor model for trait 6). We see that the threshold model fits better than either the binary or precursor models for all of the six traits.

It is not clear, a priori, why the threshold model would appear to fit some data better than the precursor model because they appear to capture similar evolutionary phenomena. It would be useful to explore this observation more thoroughly, given the new computational tools, perhaps incorporating phylogenetic error in a manner similar to Marazzi et al. (2012).

Inferring Ancestral Liabilities

Figure 3 gives a representation of how the (marginal) posterior liabilities change over the tree. Branches are divided into three classes according to the posterior probability that the liability is positive, with lineages with posterior probability > 0.7 colored red, lineages with posterior probability < 0.3 colored white, and remaining lineages colored pink.

This diagram can be compared with Marazzi et al. (2012), figure S7. The representations are, on the whole, directly comparable. A positive liability corresponds, roughly, to an ancestral precursor state. Both analyses suggest multiple origins of a precursor state, for example for a large clade of Mimosoidae. Interestingly, there are several clades where the analysis of Marazzi et al. (2012) suggests widespread ancestral distribution of the precursor state whereas our analysis indicates a negative liability at the same nodes.

Once again, our analysis is only preliminary, our goal here simply being to demonstrate what calculations can now be carried out.

Discussion

We have introduced a new framework for the computation of likelihoods from continuous characters, and illustrated the framework using an efficient algorithm for evaluating (approximate) likelihoods under Wright and Felsenstein's threshold model.

This framework opens up possibilities in several directions. The numerical integration, or numerical quadrature, literature is vast. In this article, we have focused in on a popular and simple numerical integration method, and our algorithm should be seen as a proof of principle rather than a definitive threshold likelihood method. There is no question that the numerical efficiency of Algorithm 1 could be improved significantly through the use of more sophisticated techniques: better basis functions or adaptive quadrature methods for a start.

The connection with Felsenstein's (discrete character) pruning algorithm also opens up opportunities for efficiency gains. Techniques such as storing partial likelihoods, or approximating local neighborhoods, are fundamental to efficient phylogenetic computations on sequence data (Felsenstein 1981a; Larget and Simon 1998; Swofford 2002; Pond and Muse 2004; Stamatakis 2006). These tricks could all be now applied to the calculation of likelihoods from continuous traits.

Finally, we stress that the algorithm does not depend on special characteristics of the continuous trait model, beyond conditional independence of separate lineages. Felsenstein's pruning algorithm for continuous characters is limited to Gaussian processes and breaks down if, for example, the transition probabilities are governed by Levy processes (Landis et al. 2013). In contrast, our approach works whenever we can numerically evaluation transition densities, an indeed only a few minor changes would transform our Algorithm 1 to one implementing on a far more complex evolutionary process.

Supplementary Material

Supplementary material is available at *Genome Biology and Evolution* online (http://www.gbe.oxfordjournals.org/).

Acknowledgments

This research was supported by an Allan Wilson Centre Doctoral Scholarship to G.H., financial support to D.B. from the Allan Wilson Centre, a Marsden grant to D.B., and financial support to all authors from the University of Otago.

Literature Cited

Bryant D, Bouckaert R, Felsenstein J, Rosenberg NA, RoyChoudhury A. 2012. Inferring species trees directly from biallelic genetic markers: bypassing gene trees in a full coalescent analysis. Mol Biol Evol. 29(8):1917–1932.

Cavalli-Sforza LL, Edwards AW. 1967. Phylogenetic analysis. models and estimation procedures. Am J Hum Genet. 19(3 Pt 1):233.

Cheney E, Kincaid D. 2012. Numerical mathematics and computing. Boston (MA): Cengage Learning.

Dahlquist G, Björck Å. 2008. Numerical Integration. chap. 5 in Numerical Methods in Scientific Computing (Vol. 1) SIAM, Philadelphia, PA.

Davis PJ, Rabinowitz P. 1984. Methods of numerical integration. Orlando (FL): Academic Press.

Felsenstein J. 1968. *Statistical inference and the estimation of phylogenies*. Ph.D. Thesis, Department of Zoology, University of Chicago, Chicago, IL.

Felsenstein J. 1973. Maximum likelihood and minimum-steps methods for estimating evolutionary trees from data on discrete characters. Syst Zool. 22(3):240–249.

Felsenstein J. 1981a. Evolutionary trees from DNA sequences: a maximum likelihood approach. J Mol Evol. 17(6):368–376.

Felsenstein J. 1981b. Evolutionary trees from gene frequencies and quantitative characters: finding maximum likelihood estimates. Evolution 35(6):1229–1242.

Felsenstein J. 1988. Phylogenies and quantitative characters. Annu Rev Ecol Syst. 19:445–471.

Felsenstein J. 2002. Quantitative characters, phylogenies, and morphometrics. In: Macleod N., editor. Morphology, Shape and Phylogeny. London: Taylor and Francis. p. 27–44.

Felsenstein J. 2005. Using the quantitative genetic threshold model for inferences between and within species. Philos Trans R Soc B Biol Sci. 360(1459):1427–1434.

Felsenstein J. 2012. A comparative method for both discrete and continuous characters using the threshold model. Am Nat. 179(2):145–156.

Felsenstein J, Churchill GA. 1996. A hidden Markov model approach to variation among sites in rate of evolution. Mol Biol Evol. 13(1):93–104.

FitzJohn RG. 2012. Diversitree: comparative phylogenetic analyses of diversification in R. Methods Ecol Evol. 3(6):1084–1092.

Freckleton RP. 2012. Fast likelihood calculations for comparative analyses. Methods Ecol Evol. 3(5):940–947.

Hansen TF. 1997. Stabilizing selection and the comparative analysis of adaptation. Evolution 51(5):1341–1351.

Harmon LJ, et al. 2010. Early bursts of body size and shape evolution are rare in comparative data. Evolution 64(8):2385–2396.

Ho L, Ané C. 2014. A linear-time algorithm for Gaussian and non-Gaussian trait evolution models. Syst Biol. 63(3):397.

Khaitovich P, Enard W, Lachmann M, Pääbo S. 2006. Evolution of primate gene expression. Nat Rev Genet. 7(9):693–702.

Khaitovich P, Pääbo S, Weiss G. 2005. Toward a neutral evolutionary model of gene expression. Genetics 170(2):929–939.

Kutsukake N, Innan H. 2013. Simulation-based likelihood approach for evolutionary models of phenotypic traits on phylogeny. Evolution 67(2):355–367.

Lande R. 1976. Natural selection and random genetic drift in phenotypic evolution. Evolution 314–334.

Landis MJ, Schraiber JG, Liang M. 2013. Phylogenetic analysis using Lévy processes: finding jumps in the evolution of continuous traits. Syst Biol. 62(2):193–204.

Larget B, Simon D. 1998. *Faster likelihood calculations on trees. Technical Report 98-02. Department of Mathematics and Computer Science, Duquesne University, Pittsburgh, Pa.*

Lemey P, Rambaut A, Welch JJ, Suchard MA. 2010. Phylogeography takes a relaxed random walk in continuous space and time. Mol Biol Evol. 27(8):1877–1885.

Lepage T, Lawi S, Tupper P, Bryant D. 2006. Continuous and tractable models for the variation of evolutionary rates. Math Biosci. 199(2):216–233.

Marazzi B, et al. 2012. Locating evolutionary precursors on a phylogenetic tree. Evolution 66(12):3918–3930.

O'Meara BC, et al. 2012. Evolutionary inferences from phylogenies: a review of methods. Ann Rev Ecol Evol Syst. 43:267–285.

Pond SLK, Muse SV. 2004. Column sorting: Rapid calculation of the phylogenetic likelihood function. Syst Biol. 53(5):685–692.

Ronquist F. 2004. Bayesian inference of character evolution. Trends Ecol Evol. 19(9):475–481.

Simon MF, et al. 2009. Recent assembly of the Cerrado, a neotropical plant diversity hotspot, by in situ evolution of adaptations to fire. Proc Natl Acad Sci U S A. 106(48):20359–20364.

Sirén J, Marttinen P, Corander J. 2011. Reconstructing population histories from single nucleotide polymorphism data. Mol Biol Evol. 28(1):673–683.

Stamatakis A. 2006. Raxml-vi-hpc: maximum likelihood-based phylogenetic analyses with thousands of taxa and mixed models. Bioinformatics 22(21):2688–2690.

Stevens PF. 1991. Character states, morphological variation, and phylogenetic analysis: a review. Syst Bot. 16(3):553–583.

Swofford DL. 2002. *PAUP*; Phylogenetic analysis using parsimony. Sunderland, Massachusetts: Sinauer Associates; 2002. Version 4.0b.*

Voelckel C, Gruenheit N, Biggs P, Deusch O, Lockhart P. 2012. Chips and tags suggest plant-environment interactions differ for two alpine *Pachycladon* species. BMC Genomics 13(1):322.

Wright S. 1934. An analysis of variability in number of digits in an inbred strain of Guinea pigs. Genetics 19(6):506.

Yang Z. 2006. Computational molecular evolution. Vol. 21. Oxford: Oxford University Press.

Associate editor: Laura Katz