Journal of Molecular Evolution (2022) 90:239–243
https://doi.org/10.1007/s00239-022-10058-0

MINI-REVIEW

Jump-Chain Simulation of Markov Substitution Processes Over Phylogenies

Simon Laurin-Lemay[1] · Kassandra Dickson[1] · Nicolas Rodrigue[1,2,3]

Received: 12 January 2022 / Accepted: 11 May 2022 / Published online: 2 June 2022

Abstract

We draw attention to an under-appreciated simulation method for generating artificial data in a phylogenetic context. The approach, which we refer to as jump-chain simulation, can invoke rich models of molecular evolution having intractable likelihood functions. As an example, we simulate data under a context-dependent model allowing for CpG hypermutability and show how such a feature can mislead common codon models used for detecting positive selection. We discuss more generally how this method can serve to elucidate the ways by which currently used models for inference are susceptible to violations of their underlying assumptions. Finally, we show how the method could serve as an inference engine in the Approximate Bayesian Computation framework.

Keywords Positive selection · CpG hypermutability · Substitution models · Site-interdependent models · Model violations · Likelihood ratio test · Approximate Bayesian Computation

Introduction

Model-based analyses of sets of homologous DNA and amino acid sequences have become routine practice in the study of molecular evolution. By definition, models of molecular evolution make simplifying assumptions about the underlying evolutionary process. However, relaxing the assumptions commonly adopted in model-based inferences can be technically challenging. For instance, relaxing the assumption of independence between sites can require elaborate nested Markov chain Monte Carlo (MCMC) approaches [e.g., (Robinson et al. 2003)] or Approximate Bayesian Computation (ABC) (Laurin-Lemay et al. 2018b). Indeed, we may still be years away from the development of inference-capable methods utilizing models that account for most of the understood factors at play in molecular evolution.

In the meantime, understanding the quantitative impacts of model violations on current widely adopted inference methods is crucial. For example, do codon models used to detect positive selection at the amino acid level actually detect such features, or are they being deceived by unaccounted determinants of the molecular evolutionary processes? The use of simulations, based on richer models than those used for inference, can shed light of these issues.

Traditionally, simulation of molecular evolution over phylogenetic trees is done by relying on the calculation of transition probabilities (in the stochastic process sense, rather than the biochemical sense) by exponentiation of a substitution rate matrix and drawing a state at a descendant node of a branch in proportion with these computed probabilities. This is the mode of operation of well-known simulation software, such as Seq-Gen (Rambaut and Grassly 1997) and Evolver from PAML (Yang 2007). However, conducting simulations on the basis of matrix exponentiation is limited to the models where such calculations are tractable, typically, the models that can be readily used for inference.

The traditional simulation approaches cannot be used for studying models where the rate at a particular codon site might be influenced by the codon states at other sites. In

Handling editor: **David Liberles**.

✉ Nicolas Rodrigue
nicolas.rodrigue@carleton.ca

[1] Department of Biology, Carleton University, 209 Nesbitt Biology Building, 1125 Colonel By Drive, Ottawa, ON, K1S 5B6, Canada

[2] Institute of Biochemistry, Carleton University, Ottawa, ON, Canada

[3] School of Mathematics and Statistics, Carleton University, Ottawa, ON, Canada

its most general form, such a model would operate in the sequence state space [see, e.g., (Robinson et al. 2003; Rodrigue et al. 2009)] and would require a rate matrix that is 61^N by 61^N (assuming a universal genetic code that prohibits stop codons), where N is the length of the codon sequence. With a typical protein of, say, 300 codons, it is not possible to perform any matrix algebra on the resulting 61^{300} by 61^{300} dimensional rate matrix. Thus, conducting simulations under this class of models requires an alternative approach.

The Jump-Chain Method

The jump-chain method relies on generating full realizations of the substitution process along the branches of the phylogeny by drawing dwell times as well as the nature of all events. Such simulations are used in many fields to study stochastic processes (Çinlar 1975; Gillespie 1977). The approach requires no matrix algebra on a substitution rate matrix and thus enables simulations with more complex models, such as those where the state space is at the level of the entire sequence. Ultimately, under such a model, simulating any particular substitution event—including the dwell time to that event and the nature of the substitution itself—requires knowledge of the state of the entire coding sequence. Although applicable under any model, the jump-chain method is the only approach currently known that allows one to generate artificial data under models with dependence between sites. In a phylogenetic context, it operates with the following steps:

Designate the location of the root of the tree Any point along the tree is acceptable as a root, if dealing with a time-reversible Markov substitution process as we will do here. On the other hand, the substitution model used for simulating does not need to be time reversible; there is no such constraint in the jump-chain simulation theory, although in such cases the root location becomes meaningful.

Draw a root state This step, common to both traditional and jump-chain methods, consists of a draw from the stationary distribution of the Markov process. For traditional models, such as the general time-reversible model or its special cases, this amounts to a simple draw of a nucleotide based on the nucleotide frequency parameters. When simulating under a more complex model where the stationary distribution is intractable [e.g., (Robinson et al. 2003; Rodrigue et al. 2005)], one can first simulate a long series of substitution events along an artificial branch (using the steps explained below) in order to obtain an initial state from which to simulate over the phylogeny. Doing so will be equivalent to a draw from the intractable stationary distribution. Alternatively, one could be interested in studying the substitution process starting from a real sequence, which could serve as the root state.

Draw a waiting (dwell) time The simulation process bifurcates independently from the root node, each branch using the root state drawn in the previous step. Along one of these branches, a random variable is drawn from an exponential distribution parameterized by the rate away from the root state. Under traditional models, the rate away from a state is equal to the negative of the corresponding diagonal entry in the substitution rate matrix. Otherwise, the rate away from a state is calculated as the sum of rates in the substitution matrix to all directly accessible states; assuming a point-mutation process, whereby multi-nucleotide events are assigned a rate of 0, there are at most $9N$ accessible states. Denoting the rate away from a state as R, the dwell time t is given by computing the probability integral transform of an exponential distribution of rate R and setting $t = -ln(1 - U)/R$, where U is a uniform random draw from the unit interval.

Draw the next state If the dwell time drawn in the previous step does not exceed the length of the branch along which we wish to simulate, the next state is drawn with a probability proportional to the rate to that state in the substitution matrix. This draw thus only requires the rates of the directly accessible next states (i.e., that imply a point mutation). Once drawn, this state becomes the reference for the next dwell time to be simulated.

Set the descendant node The previous two steps are repeated until the drawn dwell time brings the process beyond the end of the branch. The state at the descendant node is thus set as the last state drawn, and the procedure of the previous two steps then bifurcates independently, and so on until a dwell time is drawn beyond the length of each of the terminal branches of the tree, thereby producing the simulated alignment.

The only disadvantage is when simulating over trees with numerous long branches, which can amount to drawing numerous substitution events and thus becoming time consuming. On the other hand, detailed substitution histories can themselves become a subject of study [e.g., (Nielsen 2002; Bollback 2005)].

A Practical Example

As an example, we simulated data under a codon substitution model that allows for context-dependent hypermutability. Specifically, we consider the case of a cytosine that is followed by a guanine (CpG) along a protein-coding DNA sequence. The cytosine in a CpG context is often methylated in mammalian genomes (Tweedie et al. 1997), which gives it a high propensity to mutate to thymine through spontaneous deamination (Bird 1980). Because CpG contexts can span two adjacent codons, the widely held assumption of independence between sites becomes invalid. The first step in constructing such a model is therefore to envision the substitution process directly in the space of all possible codon sequences of a given length. Following the usual practice of traditional codon models, we focus on a point mutation-based process, which means that rates are only assigned to events in which the initial and final states differ by one nucleotide; the rates between two sequences with multiple nucleotide differences are set to zero. For a given sequence, the rate away is the sum of rates to all nearest sequences. This latter property means that the rate away from any given sequence of length N codons involves less than $9N$ terms, since there are only 9 nearest neighbor states for each codon (with stops codons excluded from the state space, there are less than 9 on average). For simplicity, one can represent the basic idea of the model in a simple 61-by-61 rate matrix, but with the understanding that a site-dependent parameter is also invoked, which requires the knowledge of the states at adjacent codons. In other words, with the site-dependent parameter, the rate specified for a given event at a particular codon site can change as the states at neighboring sites change. Thus, the rate from one codon i to another j (which differ only by one nucleotide at position c) at a particular site is given by:

$$Q_{ij} = \begin{cases} \varphi_{j_c}, & \textit{if syn. tr.,} \\ \varphi_{j_c}\kappa, & \textit{if syn. ts. non} - \textit{CpG,} \\ \varphi_{j_c}\kappa\lambda, & \textit{if syn. ts. CpG,} \\ \varphi_{j_c}\omega, & \textit{if non} - \textit{syn. tr.,} \\ \varphi_{j_c}\kappa\omega, & \textit{if non} - \textit{syn. ts. non} - \textit{CpG,} \\ \varphi_{j_c}\kappa\omega\lambda, & \textit{if non} - \textit{syn. ts. CpG,} \end{cases} \tag{1}$$

where φ_{j_c} is the frequency of the target nucleotide, κ is the transition over transversion rate ratio, λ modulates the CpG transition rate, and ω is the non-synonymous to synonymous rate ratio. For our simulations, we used the nucleotide-level parameter values, tree topology, and branch lengths obtained from running the classic version of this codon substitution model ($\lambda = 1$) on a mammalian data sets taken from Laurin-Lemay et al. (2018a). We explored three different values for ω: 0.2, 0.5, and 0.8. For each of these values, we simulated 100 replicates with $\lambda = 1$, i.e., the classic codon model, 100 replicates with $\lambda = 4$, and another 100 replicates with $\lambda = 8$, the latter being a typical value observed on mammalian data. This experiment was replicated with 9 other parameter conditions (for a total of 10; see supplementary materials).

In panels a, b, and c of Fig. 1, we report the maximum likelihood values of ω obtained under the M0 model within CodeML (Yang 2007), with the distribution of all 100 values shown as a histogram. For the simulations with $\lambda = 1$, the recovered ω values closely match those used for the simulations (marked with a dashed line), with about half of the simulations on either side of the true value. For simulations with $\lambda = 4$, the M0 model overestimates ω values in most replicates. Nearly all ω values are overestimated in simulations with $\lambda = 8$. These results suggest that applying the M0 model to data where CpGs are hypermutable is likely to lead to an overestimation of the key parameter of interest (see Supplementary Materials for details).

Panels d, e, and f of Fig. 1 show a similar set of experiments, but with data sets simulated with a mixture (equally-weighted) of ω values. For each set of 100 simulations with different λ values, the panels indicate the percentage of significant likelihood ratio tests recovered from running M7 and M8 models within CodeML. In other words, the y-axis shows the percentage of replicates that reject the null M7 model, suggesting the presence of positive selection. At $\lambda = 4$ and particularly at $\lambda = 8$, a large proportion of simulations would be considered to contain signals of positive selection, although all simulations were conducted with mixtures of ω values less than 1. Again, these results suggest that such a classic statistical test with codon models is susceptible to error in the presence of CpG hypermutability (see Supplementary Materials for details).

Future Uses

Phylogenetic simulations using the jump-chain method were used with a substitution model with dependence between sites by Robinson et al. (2003) as means of verifying their implementation. They have also been used as a means to performing posterior predictive checks [e.g., (Nielsen 2002; Rodrigue et al. 2006; Rodrigue et al. 2009; Lartillot et al. 2007; Laurin-Lemay et al. 2018b)]. In recent years, such simulations have been used to study the effect of epistasis on new models (Rodrigue and Lartillot 2017; Latrille et al. 2021). As done here, Laurin-Lemay et al. (2018a) explored the effect of CpG hypermutability on a test for detecting codon usage bias. Most of these applications remain within a small circle of researchers. We believe the method has an under-appreciated simplicity that is important to the study of much richer evolutionary models than those currently used for inference.

The method also has the potential to serve as a central kernel for the development of new inference-capable models that do not have a tractable closed-form likelihood, using ABC. The general idea of the ABC approach is to simulate a very large number of data sets, for instance, using the jump-chain method, utilizing a wide range of parameter values, and retaining the parameter values that produced data sets very similar to the true data set of interest. Indeed, Laurin-Lemay et al. (2018b) used this simulation approach in the context of an ABC implementation to

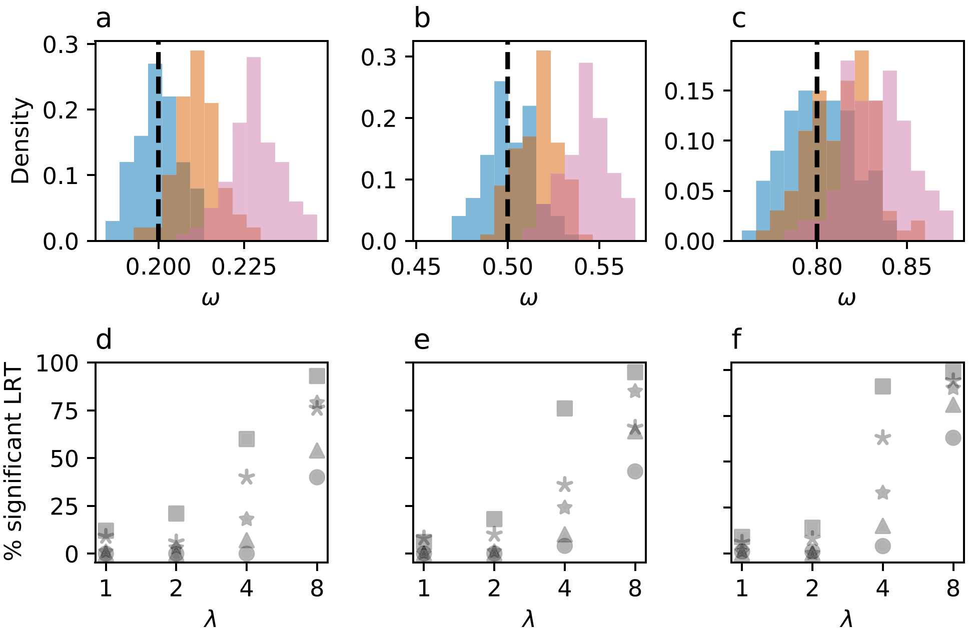

Fig. 1 Distribution of maximum likelihood ω parameter values obtained from analyzing simulated alignments with M0 model from CodeML. Simulated alignments were generated under realistic conditions, corresponding to posterior distribution of M0 obtained from analyzing a mammalian alignment of the *WDR91* gene, with different ω_0 values (black-dashed lines) and CpG transition rates (blue: $\lambda = 1$, orange: $\lambda = 4$, red: $\lambda = 8$). There were 100 replicates per condition. Details of the simulation grid are presented in supplementary materials. **a** All simulations are generated under $\omega_0 = 0.2$ (black-dashed line): 51%, 97%, and 100% of simulations had ω greater than the true value when $\lambda = 1$, $\lambda = 4$, and $\lambda = 8$, respectively. **b** All simulations are generated under $\omega = 0.5$ (black-dashed line): 50%, 90%, and 100% of simulations had ω greater than the true value when $\lambda = 1$, $\lambda = 4$, and $\lambda = 8$, respectively. **c** All simulations are generated under $\omega = 0.8$ (black-dashed line): 49%, 73%, and 95% of simulations had ω greater than the true value when $\lambda = 1$, $\lambda = 4$, and $\lambda = 8$, respectively. **d–f** Proportion of simulations (y-axis) rejecting the M7 model upon likelihood ratio test conducted with both M7 and M8 models (2 degrees of freedom). Simulated data where generated under 5 different mixtures of ω values with equally distributed values among sites from each mixture component, along with 4 levels of CpG transition rates. For realism, simulations were conducted using posterior average parameter values under M0 obtained by analyzing mammalian alignments of *STRIP1*, *GPAM*, and *WDR91* genes for panels d, e, and f, respectively. Circle, star, asterisk, triangle, and square markers correspond to ω-mixture 1 (0.1, 0.2, 0.3), mixture 2 (0.4, 0.5, 0.6), mixture 3 (0.7, 0.8, 0.9), mixture 4 (0.2, 0.5, 0.7), and mixture 5 (0.5, 0.7, 0.9), respectively (Color figure online)

study a site-dependent mutational process of CpG hypermutability within a mutation–selection framework.

We applied the methods of Laurin-Lemay et al. (2018b) to simulations with $\lambda = 8$ in the context of the more classical codon model studied here, to see if it could provide reasonable estimates of λ (see Supplementary Materials for details). As a preliminary exploration and to keep calculations manageable, we applied the method to only three simulations. Note that the simulations are stochastic processes, which will vary from one instance to the next; the actual realized CpG hypermutability can be slightly higher, or lower, than the parameter value used for simulation. All three inferences yield posterior distributions for λ that are close to the true value and include it within their 95% credibility intervals (Fig. 2). These preliminary results suggest that the method could be used not only to measure the level of CpG hypermutability but also to study the impact of accounting for this site-dependent process on other parameters, including those used to detect positive selection.

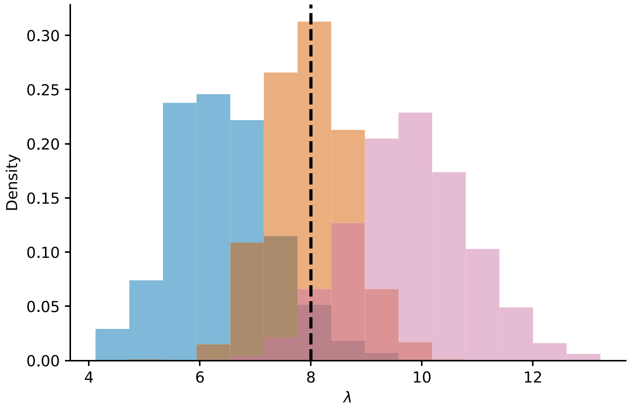

Fig. 2 Posterior distribution of λ recovered using CABC methodology when analyzing three simulated alignments (see Supplement Materials) generated with a CpG transition rate of $\lambda = 8$. For one of the simulations (blue histogram), λ has posterior mean of 6.38 with 95% credibility interval of 4.64–8.39. For a second simulation (red histogram), λ has posterior mean of 9.78 with 95% credibility interval of 7.74–11.99. For a third simulation (orange histogram), λ , and has a posterior mean of 7.99 with 95% credibility interval of 6.62–9.40 (Color figure online)

Altogether, we hope these short demonstrations will encourage other developers to explore the jump-chain simulation method within their work, either to study the robustness of their inferences to potential model violations or as a means of accounting for more of the complexities governing molecular evolution.

Supplementary Information The online version of this article (https://doi.org/10.1007/s00239-022-10058-0) contains supplementary material, which is available to authorized users.

Acknowledgements We thank three anonymous reviewers for thoughtful comments on the manuscript. This work was funded by the Natural Sciences and Engineering Research Council of Canada.

Declarations

Conflict of interest The authors declare that they have no competing interest.

References

Bird AP (1980) DNA methylation and the frequency of CpG in animal DNA. Nucleic Acids Res 8(7):1499–1504. https://doi.org/10.1093/nar/8.7.1499

Bollback JP (2005) Posterior mapping and posterior predictive distributions. Springer, New York, pp 439–462. https://doi.org/10.1007/0-387-27733-1_16

Çinlar E (1975) Introduction to stochastic processes. Prentice-Hall, Englewood Cliffs

Gillespie DT (1977) Exact stochastic simulation of coupled chemical reactions. J Chem Phys 81(25):2340–2361. https://doi.org/10.1021/j100540a008

Lartillot N, Brinkmann H, Philippe H (2007) Suppression of long-branch attraction artefacts in the animal phylogeny using a site-heterogeneous model. BMC Evol Biol 7(Suppl 1):S4. https://doi.org/10.1186/1471-2148-7-s1-s4

Latrille T, Lanore V, Lartillot N (2021) Inferring long-term effective population size with mutation-selection models. Mol Biol Evol 38(10):4573–4587. https://doi.org/10.1093/molbev/msab160

Laurin-Lemay S, Philippe H, Rodrigue N (2018a) Multiple factors confounding phylogenetic detection of selection on codon usage. Mol Biol Evol 35(6):1463–1472. https://doi.org/10.1093/molbev/msy047

Laurin-Lemay S, Rodrigue N, Lartillot N, Philippe H (2018b) Conditional approximate Bayesian computation: a new approach for across-site dependency in high-dimensional mutation-selection models. Mol Biol Evol 35(11):2819–2834. https://doi.org/10.1093/molbev/msy173

Nielsen R (2002) Mapping mutations on phylogenies. Syst Biol 51(5):729–739. https://doi.org/10.1080/10635150290102393

Rambaut A, Grassly NC (1997) Seq-Gen: an application for the Monte Carlo simulation of DNA sequence evolution along phylogenetic trees. Comput Appl Biosci 13(3):235–238. https://doi.org/10.1093/bioinformatics/13.3.235

Robinson DM, Jones DT, Kishino H, Goldman N, Thorne JL (2003) Protein evolution with dependence among codons due to tertiary structure. Mol Biol Evol 20(10):1692–1704. https://doi.org/10.1093/molbev/msg184

Rodrigue N, Lartillot N (2017) Detecting adaptation in protein-coding genes using a Bayesian site-heterogeneous mutation-selection codon substitution model. Mol Biol Evol 34(1):204–214. https://doi.org/10.1093/molbev/msw220

Rodrigue N, Lartillot N, Bryant D, Philippe H (2005) Site interdependence attributed to tertiary structure in amino acid sequence evolution. Gene 347(2):207–217. https://doi.org/10.1016/j.gene.2004.12.011

Rodrigue N, Philippe H, Lartillot N (2006) Assessing site-interdependent phylogenetic models of sequence evolution. Mol Biol Evol 23(9):1762–1775. https://doi.org/10.1093/molbev/msl041

Rodrigue N, Kleinman CL, Philippe H, Lartillot N (2009) Computational methods for evaluating phylogenetic models of coding sequence evolution with dependence between codons. Mol Biol Evol 26(7):1663–1676. https://doi.org/10.1093/molbev/msp078

Tweedie S, Charlton J, Clark V, Bird A (1997) Methylation of genomes and genes at the invertebrate-vertebrate boundary. Mol Cell Biol 17(3):1469–1475. https://doi.org/10.1128/mcb.17.3.1469

Yang Z (2007) PAML 4: phylogenetic analysis by maximum likelihood. Mol Biol Evol 24(8):1586–1591. https://doi.org/10.1093/molbev/msm088