BIOINFORMATICS

Vol. 21 Suppl. 3 2005, pages iii31–iii38
doi:10.1093/bioinformatics/bti1200

Sequence analysis

DNA assembly with gaps (Dawg): simulating sequence evolution

Reed A. Cartwright

Department of Genetics, University of Georgia, Athens, GA 30602-7223, USA

Received on May 29, 2005; accepted on August 16, 2005

ABSTRACT

Motivation: Relationships amongst taxa are inferred from biological data using phylogenetic methods and procedures. Very few known phylogenies exist against which to test the accuracy of our inferences. Therefore, in the absence of biological data, simulated data must be used to test the accuracy of methods which produce these inferences. Researchers have limited or non-existent options for simulations useful for studying the impact of insertions, deletions, and alignments on phylogenetic accuracy.

Results: To satisfy this gap I have developed a new algorithm of indel formation and incorporated it into a new, flexible, and portable application for sequence simulation. The application, called Dawg, simulates phylogenetic evolution of DNA sequences in continuous time using the robust general time reversible model with gamma and invariant rate heterogeneity and a novel length-dependent model of indel formation. On completion, Dawg produces the true alignment of the simulated sequences. Unlike other applications, Dawg allows indel lengths to be explicitly distributed via a biologically realistic power law. Many options are available to allow users to customize their simulations and results. Because simulating with indels would be problematic if biologically realistic parameters could not be estimated, a script is provided with Dawg that can estimate the parameters of indel formation from sequence data. Dawg was applied to the sequences of four chloroplast trnK introns. It was used to parametrically bootstrap an estimation of the rate of indel formation for the phylogeny. Because Dawg can assist in parametric bootstrapping of sequence data it is useful beyond phylogenetics, such as studying alignment algorithms or parameters of molecular evolution.

Availability: Dawg 1.0.0 can be obtained at the following websites: http://www.genetics.uga.edu/sw/ or http://scit.us/dawg/. The package includes source code, example files, a brief manual and helper scripts. Binary distributions are available for Windows and Macintosh OS X. A development page for Dawg exists at http://scit.us/dawg/, with links to a Subversion repository, mailing lists and updated versions.

Contact: rac@uga.edu

1 INTRODUCTION

Many tools and procedures exist that can reconstruct sequence alignments, estimate phylogenetic relationships, or estimate evolutionary parameters from extant data. These tools and procedures are required because true phylogenies and alignments are known in only very rare, experimental instances (e.g. Hillis *et al*., 1992, 1994; Bull *et al*., 1993). 'Obvious' phylogenies and alignments, although helpful, cannot provide the same level of precision as simulated phylogenies. Because we rely on tools to infer alignments, phylogenies, and evolutionary parameters, the accuracy of these tools is an issue that biologists take seriously. In the absence of known data with true phylogenies, we are left using simulations to test the accuracy of bioinformatic procedures (e.g. Blanchette *et al*., 2004; Hillis *et al*., 1994; Kuhner and Felsenstein, 1994). Simulations can produce flawless sequence alignments derived from known (i.e. user specified) phylogenies and evolutionary parameters. Using such simulated sequences, researchers can determine the accuracy of procedures for estimating the alignment, phylogeny or evolutionary parameters. Because many of these procedures assume certain models of evolution, evolving simulated sequences using such models provide a measure of accuracy in ideal cases. Using models of evolution that deviate from ideal cases can, in turn, demonstrate the robustness of the procedures. Parametric bootstrapping using simulations can also be used to produce confidence intervals for procedures.

Sequences evolve through changing existing residues adding new residues or removing existing residues. Therefore, proper simulation of molecular evolution of DNA should involve nucleotide substitution, insertion and deletion. However, existing tools for simulating sequence evolution (Table 1) either do not include indels, like Seq-gen (Rambaut and Grassly, 1997) or evolver (Yang, 1997), include an underived model of indel formation, like Rose (Stoyle *et al*., 1998), or are designed for a particular purpose and are inflexible like EvolveAGene (Hall, 2005). I developed Dawg to fill these gaps.

Dawg is the first sequence simulation program to combine the popular general time reversible model, gamma and invariant rate heterogeneity, and a model of indel formation. Dawg has several different ways to model the distributions of the sizes of insertions and deletions. Most importantly, Dawg is the first program that can explicitly model indel sizes via a power law distribution, which has been found in both nucleotide and protein sequences (Benner *et al*., 1993; Chang and Benner, 2004; Gu and Li, 1995; Zhang and Gerstein, 2003). Additionally, Dawg can simulate recombination by using different phylogenies for different sections of the sequence. Dawg restricts recombination and indel formation to blocks of nucleotides of a constant width. Deletions remove whole blocks, while insertions and recombination occur between blocks. In most uses of the program, the blocks are 1 nt wide. Although the underlying model of evolution is neutral DNA, the evolution of coding sequences can be approximated by setting blocks to be 3 nt wide and specifying different rates of evolution for the positions in the block.

The utility of any simulation lies in its ability to generate biologically realistic data. To that end, researchers can use standard phylogenetic packages to estimate most of the parameters of the model. A helper script written in Perl, lambda.pl, is provided with Dawg to allow researchers to estimate parameters of indel evolution from biological data, facilitating biologically meaningful simulations of indel formation.

Table 1. Comparison of simulation programs

Feature	Seq-Gen 1.32	Evolver 3.14a	Rose 1.3	EvolveAGene 2.3	Dawg 1.0.0
GTR	Yes	Yes	No	No	Yes
Rate heterogeneity	$\Gamma + \mathrm{I}$	Γ	$\Gamma + \mathrm{I}$	No	$\Gamma + \mathrm{I}$
Recombination	Yes	No	No	No	Yes
Indels	No	No	Yes	Yes	Yes
Indel parameter estimation	N/A	N/A	No	No	Yes
Input format	Switch	File	File	Menu	File
Unix	Yes	Yes	Yes	No	Yes
Mac OS X	Yes	Yes	Yes	Yes	Yes
Win32	Yes	Yes	No	No	Yes

Seq-Gen (Rambaut and Grassly, 1997); Evolver (Yang, 1997); Rose (Stoyle *et al*., 1998); EvolveAGene (Hall, 2005).

```
# example.dawg
Tree = ((AY727331:0.001359,AY727330:0.001359):0.084512,
(AY727327:0.006116,AY727326:0.006116):0.079756);
Model = "GTR"
Params = {1.08031, 2.45581, 0.44452,
          1.09145, 4.06519, 1.00000}
Freqs = {0.353470, 0.143681, 0.178206, 0.324643}
Length = 300
Lambda = 0.143120
GapModel = "NB"
GapParams = {1, 0.753247}
Format = "Clustal"
File = "example.aln"
Seed = 1981
```

Fig. 1. Example.dawg, a sample input file derived from biological data. The parameters were derived from the sequences of chloroplast trnK introns of four Rosid species. The initial sequence length was shortened and a seed was added for reproducibility. The output, example.aln, can be found in Figure 2. See Sections 3.10 and 5 for more detail.

2 SYSTEMS AND METHODS

Dawg is a command line program written in standard C++, GNU Bison, and GNU Flex for portability. It is packaged using the GNU autoconf and GNU automake tools and will compile on systems that support them, including most popular derivatives of Unix like Linux, FreeBSD and Macintosh OS X. It can be compiled in Windows using the minimalist GNU for Windows (MinGW) or using Microsoft Visual Studio .Net 2003 with ports of GNU Bison and Flex. The document INSTALL explains how to compile and install the package for most systems. Development took place on a variety of machines, including a Windows XP workstation, a FreeBSD server, an Irix server, a Linux cluster and a Macintosh OS X desktop. A Perl script, lambda.pl, is distributed with Dawg and can be used to estimate parameters for the indel model from an alignment and a phylogeny with branch lengths. A few other utility scripts are also included.

Dawg is configured by an input file, an example of which is in Figure 1. One of the design goals of Dawg is to offer a robust DNA simulation package, and thus there are a wide range of options. New options and features may be added in future versions of the program. Currently options include controlling the phylogeny, substitution model, indel model and program output.

3 ALGORITHM

Dawg's algorithm for simulating evolution supports substitution, rate heterogeneity, indel formation and recombination. There is no limit on the length of sequences or size and structure of phylogenies, except those imposed by hardware and time. The complexity of the algorithm is $O(NLR)$, where N is the number of nodes in the phylogeny, L is the average sequence length, and R is the number of repetitions. Similarly, the memory requirement is $O(NL)$. Simulation of Figure 1 with 10 000 repetitions took 10.5 s on an Intel Xeon 3.06 GHz, Windows XP workstation and consumed less than a megabyte of memory.

3.1 Substitution

Dawg produces descendent sequences from ancestral sequences via a two-step evolutionary model. The first step simulates substitution via the general time reversible model with gamma and invariant rate heterogeneity (GTR + Γ + I; Felsenstein, 2004; Lanave *et al*., 1984; Rodríguez *et al*., 1990; Tavaré, 1986; Waddell and Steel, 1997; Yang, 1993, 1994). The second step simulates indel formation via a continuous-time, length-dependent model derived for Dawg.

GTR is a ten-parameter model (eight free, two dependent) which represents nucleotide substitution in continuous time. The parameters are the four stationary nucleotide frequencies, φ_i, and the six symmetric, relative, instantaneous rates of substitution, σ_{ij}. These parameters combine to form the instantaneous substitution rate matrix

$$Q = \begin{bmatrix} q_{AA} & \sigma_{AC}\varphi_C & \sigma_{AG}\varphi_G & \sigma_{AT}\varphi_T \\ \sigma_{AC}\varphi_A & q_{CC} & \sigma_{CG}\varphi_G & \sigma_{CT}\varphi_T \\ \sigma_{AG}\varphi_A & \sigma_{CG}\varphi_C & q_{GG} & \sigma_{GT}\varphi_T \\ \sigma_{AT}\varphi_A & \sigma_{CT}\varphi_C & \sigma_{GT}\varphi_G & q_{TT} \end{bmatrix},$$

where $q_{ii} = -\sum_{j \neq i} \sigma_{ij}\varphi_j$. Many common models (Felsenstein, 1981, 1984; Hasegawa *et al*., 1985; Jukes and Cantor, 1969; Kimura, 1980, 1981; Tamura and Nei, 1993) can be expressed as specializations of GTR. See Felsenstein (2004) for a recent and detailed description of the GTR model.

3.2 Heterogeneous rates

The Γ + I model of heterogeneous rates of nucleotide evolution allows for the rate of evolution to vary among sites, with a set proportion of sites remaining unchanged (Waddell and Steel, 1997). Under this model, the relative rates of substitution at each position are independent and identically distributed by the hierarchal distribution,

$$f(r|\alpha,\iota) = \begin{cases} 0 & (r < 0) \\ \iota & (r = 0) \\ (1-\iota)\dfrac{(\alpha r)^\alpha e^{-\alpha r}}{r\Gamma(\alpha)} & (r > 0) \end{cases},$$

where $0 \leq \iota \leq 1$ is the proportion of invariant sites, $\alpha > 0$ is the shape parameter, and $\Gamma(\alpha)$ is the complete gamma function. The expected value of this distribution is $1 - \iota$, and the variance is $(1 - \iota)(\gamma + \iota)$, where $\gamma = \alpha^{-1}$ is the coefficient of variance. The coefficient of variance is preferred over the shape parameter for describing the $\Gamma + \mathrm{I}$ distribution. If $\gamma = 0$, the distribution becomes discrete, and a site either evolves in step with the branch length ($r = 1$) or remains unchanged ($r = 0$). The simulation holds constant each site's relative rate of substitution, and daughter nucleotides inherit their parent's rate. Dawg extends the basic $\Gamma + \mathrm{I}$ model by allowing each position in a block to have different γ and ι parameters. Each position also has a relative scaling parameter, s, to allow some positions to evolve relatively faster than others.

3.3 Rescaling

Dawg calculates the probability that nucleotide j substitutes for nucleotide i at site n with relative rate r_n and relative scale s_n over time t as the (i, j) entry of matrix $P_n(t) = \mathrm{e}^{kQs_n r_n t}$, where k is a correction factor. The expected number of substitutions for block position w is

$$E(Y|t, w) = \sum_{i=\{A,C,G,T\}} -k\varphi_{i,w} q_{ii,w} s_w (1 - \iota_w) t,$$

where k is the rescaling constant, $\varphi_{i,w}$ is the frequency of nucleotide i for block position w, $q_{ii,w}$ is the (i, i) entry of the GTR matrix for position w, s_w is the relative scalar for position w in a block and ι_w is the proportion of invariant sites for position w. Therefore, the expected number of substitutions per site given time t is

$$E(Y|t) = \frac{1}{W} \sum_{w=1}^{W} E(Y|t, w),$$

where W is the block width. As Felsenstein (1981) and Yang (1994) suggest, Dawg rescales the substitution matrix such that the branch lengths represent the expected number of substitutions per site. Thus, since $E(Y|t) = t$,

$$k = \frac{W}{\sum_{W=1}^{W} \sum_{i=\{A,C,G,T\}} -\varphi_{i,w} q_{ii,w} s_w (1 - \iota_w)}.$$

Since the GTR parameters are the same for all block positions, the correction factor simplifies to

$$k = -W \left(\sum_{w=1}^{W} s_w (1 - 1_w) \right)^{-1} \left(\sum_{i=\{A,C,G,T\}} \varphi_i q_{ii} \right)^{-1}.$$

overleaf

This rescaling to substitution time is important for interpretation of the indel model.

Because the calculation of $P_n(t)$ involves finding the eigenvalues and eigenvectors of kQ, Dawg implements a Jacobi transformation (Press *et al.*, 1992) optimized for a four-by-four matrix. Although Jacobi transformations are simple, accurate and numerically stable, they only work on symmetric matrices, and kQ is not symmetric. Dawg utilizes a mathematical trick to find the eigensystem of a symmetric matrix related to kQ and to convert the results to the eigensystem of kQ (Yang, 1995). The matrix $S = \Phi^{1/2} k Q \Phi^{-1/2}$ is symmetric and has the same eigenvalues as kQ, where $\Phi^{1/2} = \mathrm{diag}(\varphi_A^{1/2}, \varphi_C^{1/2}, \varphi_G^{1/2}, \varphi_T^{1/2})$. If V_S are right eigenvectors of S, then $V_{kQ} = \Phi^{-1/2} V_s$ are the right eigenvectors of kQ. Once the eigensystem is found, $P_n(t)$ can be calculated and used with the state of the ancestral nucleotide to randomly draw the descendent nucleotide for the position.

Table 2. Comparisons of indel formation models

	TKF91	TKF92	Rose	McAlign	Long Indel	Dawg
Poisson process	Yes	Yes	No	No	Yes	Yes
Length dependent	Yes	Yes	Yes	Yes	Yes	Yes
Time dependent	Yes	Yes	Yes	Yes	Yes	Yes
In substitution time	No	No	No	Yes	No	Yes
Multiresidue indels	No	Yes	Yes	Yes	Yes	Yes
Overlapping ends	N/A	Yes	No	No	Yes	Yes
Time reversibility required	Yes	Yes	No	Yes	Yes	No
Immortal link	Yes	Yes	No	No	Yes	Yes
Insertion–deletion differences	Yes	Yes	Yes	No	Yes	Yes
Gaps can overlap	N/A	No	Yes	No	Yes	Yes
Alignment algorithm	Yes	Yes	No	Yes	Yes	No
Simulation algorithm	No	No	Yes	No	No	Yes

TKF91 (Thorne *et al.*, 1991); TKF92 (Thorne *et al.*, 1992); Rose (Stoyle *et al.*, 1998); McAlign (Keightley and Johnson, 2004); Long Indel (Miklós *et al.*, 2004).

3.4 Indel formation

Dawg implements a novel model of indel formation. Like substitutions, indels occur in continuous time. The model treats insertions and deletions as different processes, and each one has its own distribution of sizes and instantaneous rate of formation. The model assumes that there is a fixed, instantaneous rate of indels occurring at any site at any time; therefore, indels are more probable in longer sequences and over longer time intervals. To satisfy this assumption, Dawg uses a Poisson process that is linearly dependent on the length of the sequence. Table 2 shows some differences between Dawg's model and other indel models.

In this model indel formation is restricted to a certain block width, e.g. 1 for nucleotides and 3 for codons. Indel formation occurs in substitution time, and when the block width is 1, the instantaneous rates of formation approximately represent the ratio of insertions or deletions to substitutions. An indel is identified by two parameters: a location and a length. The location represents the place where nucleotides are inserted or the place at which a deletion begins. The length represents the number of blocks inserted or deleted.

Insertions are rather simple to model. If l is the number of blocks in the subsequence being evolved, then there are $l+1$ possible locations for an insertion to occur, including both ends of the sequence. The sequence thus has an 'immortal link' (Thorne *et al.*, 1991), ensuring that an insertion can occur if $l = 0$. If λ_I is the rate of insertion per location, then the waiting time until an insertion occurs is exponentially distributed with mean $(\lambda_I l + \lambda_I)^{-1}$. Inserted nucleotides are randomly drawn from the stationary base frequencies and heterogeneous rate distribution. Insertions are right oriented, which means that, if an insertion occurs at a recombination point, the insertion becomes associated with the rightmost section.

Deletions are more difficult to model because the ends of the subsequence have to be taken into account. A deletion that starts in a region preceding a sequence may still delete part of the sequence. To account for this, I first assume that the subsequence being modeled exists inside a larger sequence of size N blocks, such that $N \gg l$.

I also assume that the maximum size of a deletion is M blocks, such that $N \gg M$. This allows the ends of the larger sequence to be ignored and the ends of the smaller sequence to be considered. A deletion of size u that occurs in the larger sequence will delete part of the subsequence if it begins at one of the l nucleotides of the subsequence or at one of the $u-1$ nucleotides preceding the subsequence. Therefore, if deletions occur uniformly along the larger sequence, then the probability that a deletion in the larger sequence of size u removes some part of the smaller sequence is simply $(u-1+l)/N$. If $f_D(u)$ is the discrete distribution of the size of deletions, then the total probability that a deletion in the larger sequence removes some part of the smaller sequence is

$$\sum_{u=1}^{M} \frac{u-1+l}{N} f_D(u) = \frac{1}{N}\left[\sum_{u=1}^{M} u f_D(u) + (-1+l)\sum_{u=1}^{M} f_D(u)\right] = \frac{\bar{u}_D - 1 + l}{N}. \quad (1)$$

If λ_D is the rate of deletion per location, then the total rate of deletion in N is $\lambda_D N$. From this and Equation (1), the waiting time until a deletion occurs in the subsequence is exponentially distributed with mean $[\lambda_D(\bar{u}_D - 1) + \lambda_D l]^{-1}$. Because N cancels out, we can consider both it and M to be infinite, allowing more flexibility in the choice of $f_D(u)$.

3.5 Indel-size distributions

In Dawg the length of an indel is represented by the number of blocks that it covers. Dawg has three different ways to model the distribution of indel lengths. The first method allows users to specify the exact discrete distribution of indel lengths. This is referred to as the user model. The second method models indel lengths using a negative binomial distribution. This model takes two parameters, an integer (r) and a proportion (q), and has the probability mass function,

$$f(l) = \binom{r+l-2}{l-1}(1-q)^r q^{l-1},$$

where $l = 1,2,\ldots$ is the length of an indel. The mean of this distribution is $1+rq/(1-q)$, and the variance is $rq/(1-q)^2$. If $r = 1$, the distribution is geometric and is equivalent to the affine scoring model with gap-open and gap-extension penalties (Gotoh, 1982). More generally, the negative binomial distribution corresponds to the scoring model

$$s(l|\theta,r,q) = \Theta(\theta,r,q) + l\log q + \sum_{x=1}^{l+r-2}\log x$$

$$\Theta(\theta,r,q) = \log(\theta) - \log q + r\log(1-q) - \log((r-1)!),$$

where θ is the probability of an indel forming.

The third and most important method models indel lengths via a power law or Zipf distribution. Indel lengths have been found to approximately obey this distribution (Benner *et al.*, 1993; Chang and Benner, 2004; Gu and Li, 1995; Zhang and Gerstein, 2003), and some theory supports it (Benner *et al.*, 1993). In a Zipf distribution, the probability that an indel has length $l = 1,2,\ldots$ is

$$f(l) = \frac{l^{-a}}{\zeta(a)},$$

where $a > 1$ is the parameter of the distribution and $\zeta(a)$ is the Riemann Zeta function:

$$\zeta(a) = \sum_{k=1}^{\infty} k^{-a}.$$

If $a > 2$, the mean of a Zipf distribution is $\zeta(a-1)/\zeta(a)$, and if $a > 3$, the variance is $\zeta(a-2)/\zeta(a) - (\zeta(a-1)/\zeta(a))^2$. Otherwise, the mean and variance are infinite. The scoring model of a Zipf distribution is logarithmic:

$$s(l|\theta,a) = \log(\theta) - \log(\zeta(a)) - a\log(l).$$

Because the tail of a Zipf distribution is often fat, Dawg truncates the distribution to a user specified, maximum indel-size, M.

3.6 Recombination

Dawg can produce simulated sequences from phylogenies that contain recombinations. This feature is optional and is enabled when a user specifies a phylogeny of multiple trees in the input file. Other software, e.g. ms (Hudson, 2002), can be used to simulate such tree sets from demographic parameters. Dawg combines trees into a recombinant phylogeny by splitting the sequences at each node in the phylogeny into multiple sections. Each section corresponds to a separate tree and has its own ancestral node. If two or more sections have the same ancestral node, then their distance to that ancestor is specified by the last tree in the group.

Recombination occurs when different sections in the same node are descended from different ancestors. A recombinant sequence is assembled from donor sequences, and each donor sequence is associated with an ancestral node. If node A is ancestral to section N of the descendent sequence, then section N of the descendent sequence is copied from section N of donor sequence A. Donor sequences are produced by evolving ancestral sequences over the distance separating the ancestor from the descendent. Figure 2 describes this process.

3.7 Simulation

Dawg simulates this model of indel formation using a Gillespie algorithm (Gillespie, 1977). For the algorithm, the total event rate is

$$\lambda_T(l) = (\lambda_I + \lambda_D)l + \lambda_D\bar{u}_D + \lambda_I - \lambda_D,$$

and the probability that an event is an insertion is $p(l) = (l+1)\lambda_I/\lambda_T(l)$. Under the algorithm the waiting time until an event occurs is drawn from an exponential distribution with mean $\lambda_T(l)^{-1}$. When an indel is formed, it is randomly drawn as an insertion or deletion, with probabilities $p(l)$ and $1-p(l)$, respectively. The length of the indel is drawn from its indel-size distribution, and its position is drawn uniformly from the pool of all possibilities, given the indel size. The algorithm cycles, updating the length of the sequence each round, until the sum of the waiting times is greater than the branch length.

Using these models of evolution, Dawg can construct the sequence for every node in the phylogeny from the root node. The sequence of the root node can be either specified by the user or randomly constructed from the stationary distribution of nucleotide frequencies and the heterogeneous rate model.

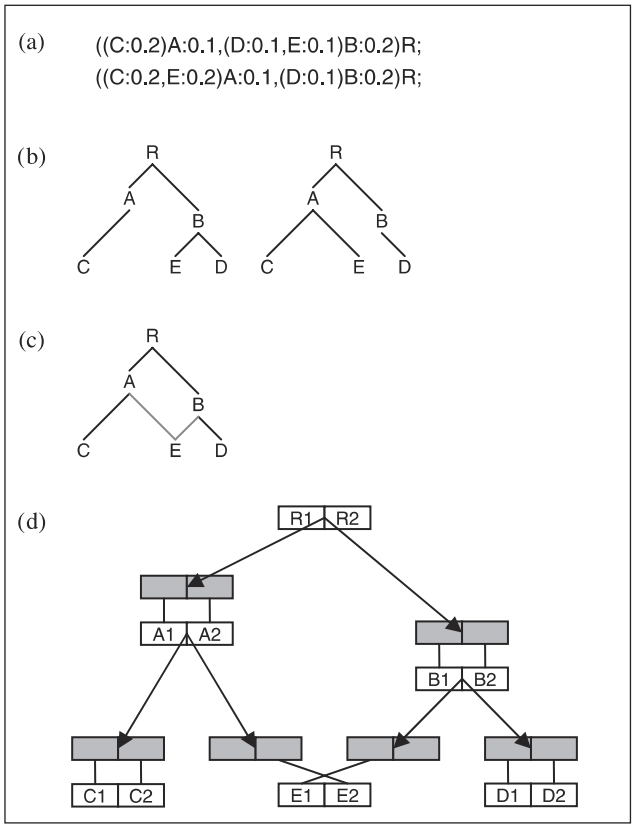

Fig. 2. (**a**) A pair of Newick trees specifying a recombinant phylogeny. (**b**) A visualization of these trees. (**c**) A visualization of the recombinant phylogeny, where black branches are found in both trees and gray branches are found in only one tree. (**d**) A graph of how Dawg simulates evolution on the recombinant tree. Boxes, pairs of boxes, and gray boxes represent sections, sequences and donors, respectively. Arrows indicate where evolution occurs, and lines indicate where donor sections are copied to descendent sections.

3.8 Alignment

Dawg maintains the indel history of each nucleotide for reconstruction of the true alignment of sequences. The indel history has one of four states: root, insertion, deletion and deleted insertion. When a descendent sequence is constructed from its ancestors, the indel states of parent nucleotides are copied to daughter nucleotides. In this way sequences at the tips of the phylogeny will contain information about their lineage from the root.

When nucleotides are inserted in the sequence, their indel state is marked as being insertions. When nucleotides are deleted in a sequence, they are not actually removed, but rather are marked as being deletions. (Dawg skips deletions when evolving sequences.) If an inserted nucleotide is subsequently deleted, it is marked to distinguish it from a deleted root nucleotide.

The alignment algorithm uses the history of sequences to construct their alignment, adding gaps to sequences opposite of insertions and deleted insertions. In the alignment, insertions are once again right-oriented, which means that if an insertion occurs at the same location as a previous deletion, then it will be to the right of that deletion in the alignment. Dawg can also translate aligned nucleotide sequences into amino acid sequences, using a newly developed and extremely efficient algorithm (Cartwright and Theobald, manuscript in preparation).

3.9 Parameter estimation

I supply a Perl script, lambda.pl, with Dawg. This script contains a simple algorithm to estimate parameters for Dawg's indel model from a nucleotide sequence alignment and a rooted phylogeny with branch-lengths that represent the expected number of substitutions per site. The script does not estimate parameters for the richest model of indel formation possible with Dawg. Instead it treats insertion and deletion as equal processes and estimates the total rate of indel formation, $\lambda_{ID} = \lambda_I + \lambda_D = 2\lambda_I = 2\lambda_D$. It can be shown that, assuming a block width of 1, the total number of unique indels in a phylogeny, N, is approximately distributed by a Poisson distribution with mean $\lambda_{ID}\bar{L}T$, where $\bar{L}$ is the average length of the sequences and T is the total branch-length of the rooted phylogeny. If $\bar{L}$ and T are known, a maximum likelihood estimate of λ_{ID} is $\hat{\lambda}_{ID} = N/(T\bar{L})$.

The script calculates T and $\bar{L}$ from the supplied alignment and rooted phylogeny. It then estimates N as the number of unique gaps in the alignment. Although $\hat{\lambda}_{ID}$ is a maximum likelihood estimator for the approximate distribution, its statistical significance to the actual distribution is unknown. I have simulated Dawg and lambda.pl together and verified that $\hat{\lambda}_{ID}$ is consistent with λ_{ID} (data not shown). The largest magnitude of deviation, -3% , occurred when $\hat{\lambda}_{ID}$ equaled 1 indel per substitution, which for non-repetitive DNA is an extremely high and biologically unrealistic value.

The Perl script goes beyond estimating the instantaneous rate of indel formation per site. It also calculates the distribution of indel sizes (user-input model) and fits this distribution to three other models: negative binomial, geometric (NB with $r = 1$) and power law. The parameters for the geometric and negative-binomial models are estimated via maximum likelihood. The power law model is fitted via linear regression of the first five indel size frequencies on a log–log scale (Jones and Handcock, 2003). This has been shown to be a very good estimator of power law distributions (Goldstein *et al.*, 2004). These models can be distinguished via maximum likelihood for probability, minimum Akaike information (Akaike, 1974) and Bayesian information (Schwarz, 1978) for parsimony, and χ^2 for goodness-of-fit.

It is worth noting that the most popular way to align sequence pairs is with affine gap penalties (Gotoh, 1982), which bias the results towards a geometric model. Alternatively, the algorithm of Miller and Myers (1988) can align sequences globally using logarithmic gap penalties, which would be appropriate for the power-law model. An implementation of this algorithm can be obtained from me.

3.10 Example usage

As an example of how to use Dawg and lambda.pl, I estimated the rate of indel formation of a set of sequences and then parametrically bootstrapped the estimate via Dawg. I used sequences from the intron of the chloroplast trnK gene of four plant species: *Hibiscus mechowii, Hibiscus cannabinus, Prunus nigra, and Prunus virginiana* (Genbank accession numbers AY727326, AY727327, AY727330, and AY727331, respectively Shaw *et al.*, 2005). The genera are both Rosids, but *Prunus* is a eurosid I and *Hibiscus* is a eurosid II. Insertions and deletions are known to be prevalent in chloroplast sequences (Clegg *et al.*, 1994), and the trnK intron almost certainly evolves neutrally and without recombination.

I first aligned these sequences using ClustalW 1.81 (Thompson *et al*., 1994) and corrected the alignment where necessary. Next, I used Paup* 4.0 (Swofford, 2002) to estimate the phylogeny and substitution parameters of the sequences for a GTR and molecular clock model. I then used lambda.pl to estimate the indel parameters from the phylogeny and aligned sequences. From the estimates, I constructed a parameter file for Dawg and simulated a thousand sequences evolving under the conditions estimated from the actual data. For each of these simulated sequence sets, I estimated phylogenetic trees and the rate of indel formation using Paup* and lambda.pl.

4 IMPLEMENTATION

Dawg is run on the command line. It is controlled through input files and a few command-line switches, which control the processing of the input files. Input files can be processed together or in succession. The structure of an input file is a series of statements in the form of '*variable* = *value*'. There are several types of values: strings, booleans, numbers, trees and vectors of values.

The default output is to stdout in Fasta format. Output can also be to a file; Phylip, Nexus and Clustal formats are also supported. Dawg can return multiple sequence sets, and a Perl script, outsplit.pl, is provided to retrieve single alignments from outputs.

5 RESULTS

The rounded, average length of the biological sequences was 741. Their estimated phylogeny was [(AY727331:0.001359, AY727330:0.001359):0.084512, (AY727327:0.006116, AY727326: 0.006116):0.079756]. The estimated stationary frequency of adenine was 0.353470, cytosine 0.143681, guanine 0.178206 and thymine 0.324643. The symmetric instantaneous rate of substitution for adenine and cytosine was estimated to be 1.08031, adenine and guanine 2.45581, adenine and thymine 0.44452, cytosine and guanine 1.09145, cytosine and thymine 4.06519, and guanine and thymine 1.0. The indel-size distribution was estimated to be geometric with a q of 0.753247.

The estimated rate of indel formation was 0.143120, and bootstrapping via Dawg gave a 95% confidence interval of 0.078530 to 0.213560. In biological terms, this is 8–21 indels per 100 substitutions. The phylogenies produced from the simulated data during bootstrapping were consistent with the biological data, having in every case the same topology as the biological phylogeny. Furthermore, the biological phylogeny had a total tree length of 0.179218, and the simulated phylogenies had an average total tree length of 0.180075 and standard deviation of 0.017993.

Figure 1 shows an input file, example.dawg, for Dawg, which was derived from the biological data mentioned above. The sequence length was shortened from 741 to 300, and a random number seed was added to make the results suitable for publication. Figure 3 shows the resulting output file, example.aln, of Dawg.

6 DISCUSSION

Although a geometric model was found to best fit the gap sizes in the trnK intron alignment, this could be an artifact induced by aligning the sequences with an affine gap model. However, the gap size distribution has little impact on bootstrapping the rate of indel formation.

```
CLUSTAL multiple sequence alignment (Created by DAWG Version 1.0.0)


AY727326        TTCGAAAATATGTTAGTACTCAATATGAATTCTTTGAGTTAAAAAAGATAAAGCAAA--A
AY727327        TTCGAAAATATGTTAGTACTCAATATGAATTCTTTGAGTTAAGAAAGATAAAGCAAA--A
AY727330        TTCAAAAATATGCTAGGACTGAATATGAATTCTTAAAGTTAAGAAAGATAAAGAAAAACA
AY727331        TTCAAAAATATGCTAGGACTGAATATGAATTCTTAAAGTTAAGAAAGATAAAGAAAAACA


AY727326        ATACATAATGTGATTTCAATATTCCAATTACCTAACAATACGGCTATCAATTAAACGATT
AY727327        ATACATAATGTGATTTCAATATTCCAATTACCTAACAATACGGCTATCAATTAAACGATT
AY727330        GTACATAATGTAAA----TTATTGCAA---------AAAACGGCTAACAATTAGACGATT
AY727331        GTACATAATGTAAA----TTATTGCAA---------AAAACGGCTAACAATTAGACGATT


AY727326        TTAGGATTACACCGACAAATATTAGGCCGATATGAATTTAACATCATGTTGTATTTAGAT
AY727327        TTAGGATTACACCGACAAATATTAGGCCGATATGAATTTACCATCATGTTGTATTTAGAT
AY727330        TTAGGATTACGCTGACAAATATTAGGATGATATTAATTTA------TCTTGTATTTAGAT
AY727331        TTAGGATTACGCTGACAAATATTAGGATGATATTAATTTA------TCTTGTATTTAGAT


AY727326        GCTGTCTTTTATTAACATTCATCATTAAAT-TTGGAACCTTTTGCATTTAAGAAGTACAT
AY727327        GCTGTCTTTTATTAACATTCATCATTAAAT-TTGGAACCTTTTGTATTTAAGAAGTACAT
AY727330        GCTGTCTTTTATCAACATTCATCACTAGATATTGGAACCTATTGCATCTAAGAAGTACAT
AY727331        GCTGTCTTTTATCAACATTCATCACTAGATATTGGAACCTATTGCATCTAAGAAGTACAT


AY727326        GTTTAATAGTGTTTAAAA-TATATATGAAATTGATCATAAGGA---TCTATAAATGCGGT
AY727327        GTTTAATAGTGTTTATAA-TATATATGAAATTGATCGTAAGGA---TCTATAAATGCAGT
AY727330        GTTTAATAGGGTT-AAAACTATATATGAAGTCGATTATAAGGAATTTCTATAAATGTAGC
AY727331        GTTTAATAGGGTT-AAAACTATATATGAAGTCGATTATAAGGAATTTCTATAAATGTAGC


AY727326        TCTTCAATTTCTTG
AY727327        TCTTCAATTTCTTG
AY727330        TCTTCAATTTCCTA
AY727331        TCTTCAATTTCCTA
```

Fig. 3. Example.aln, an example output file produced from example.dawg (Fig. 1).

Table 1 compares Dawg to three other published sequence simulation programs. However, it is important to go into some detail about the differences between Dawg and two previously published applications for simulating evolution with indels: Rose (Stoyle *et al*., 1998) and EvolveAGene (Hall, 2005).

Stoyle and colleagues do not derive Rose's model of indel formation, which differs from the model that I have derived for Dawg. In Rose, a binomial distribution with parameters TL and vp describes the number of insertions and deletions that occur along a branch. The parameter T is the branch length rounded to the nearest integer. Any branch length <0.5 will turn off insertions and deletions, which is a problem for estimates from standard nucleotide models which require branch lengths to be in substitution time. The parameter L is the length of the sequence at the bottom of the branch; unlike Dawg, Rose does not update the sequence length as new indels form along the branch. The p is the proportion of nucleotides that have a mutation rate >1; Dawg does not associate the indel model with rate heterogeneity. The parameter v is the insertion or deletion threshold, which is similar to Dawg's parameter λ. Although both models can be made to produce similar distributions, Dawg's model is derived from a simple Poisson process and thus is consistent with the derivation of models for continuous-time substitution.

For indel size distributions, Rose only provides a user-based model. Dawg provides two models in addition to a user-based model: negative-binomial and power-law. Using power-law distributions for indel sizes is advantageous for researchers because they are biologically realistic. The basic substitution model in Rose is PAM, which was developed for protein sequences. For substitutions, Dawg uses GTR $+ \Gamma +$ I, which was developed for nucleotide sequences. Perhaps the most important difference between these two models is the meaning of the branch lengths. In PAM a branch length of 1 means that sequences have 1% divergence, whereas in GTR it means that each site is expected to have had 1 substitution.

Hall (2005) developed EvolveAGene using a methodology significantly different than the one employed here to develop Dawg.

Whereas Dawg models the process of substitution, EvolveAGene models the separate processes of mutation and acceptance. EvolveAGene simulates the mutation of coding sequences based on the spontaneous mutational spectrum of *Escherichia coli*. EvolveAGene would need to be modified if another mutational spectrum is desired. Relying on mutation spectra can be restrictive for researchers studying organisms for which the mutation spectra are unknown. Furthermore, EvolveAGene does not allow users to specify their own phylogenies and restricts tree topology to balanced, bifurcating trees. EvolveAGene is rather inflexible when compared to the many options available for Dawg.

Dawg also comes with a way to estimate parameters of indel formation setting it apart from Rose and EvolveAGene. This ability may prove useful for researchers interested in studying indel formation or using gaps to aid in estimating phylogenies. The parameter estimator is not perfect. It can be improved by assigning alignment gaps to individual branches. Furthermore, it estimates a net indel rate, instead of separating insertion and deletion into separate processes. Researchers interested in additional biological realism should consider separating the indel rate into insertion and deletion rates, favoring the deletion rate. For example, Zhang and Gerstein (2003) found that deletions occurred roughly three times more often than insertions in neutral DNA. This result suggests that $\lambda_I = (1/4)\lambda_{ID}$ and $\lambda_D = (3/4)\lambda_{ID}$ would be biologically realistic parameters.

Although the model of indel formation implemented in Dawg is an improvement over previous models, it does not take into consideration several biological features of indel formation. For instance, indel formation in Dawg is content independent, whereas natural indel formation is heavily influenced by repetitive sequences. Since repetitive sequences create indel hotspots, they also violate the assumption of uniform insertion and deletion rates. Modeling indel formation with extreme biological realism is hard at this time because many of the factors influencing hotspots remain unknown. However, despite these reservations Dawg's model of indel formation offers fruitful avenues for researchers who need to model sequence evolution.

Some researchers are reconstructing extinct genomes and are using simulations to test their methodology (Blanchette *et al*., 2004; Pennisi, 2005). Dawg is not designed explicitly to simulate genomes but can be utilized in that fashion. It contains many of the features described in genome simulations used by Blanchette and colleagues. Additionally, it can simulate recombination, which may prove useful in some contexts of studying genome reconstruction. However, it does not have the ability to distinguish transposons from other indels or treat CpG regions differently than other sections of DNA. It also lacks a model of chromosomal rearrangement.

There are many possible features that can be added to Dawg. To improve realism, repetitive DNA and hotspots should be eventually included. Another important feature would be to allow separate GTR models for each block position just as separate $\Gamma + \mathrm{I}$ models are currently allowed. Another possibility is to allow each section of a sequence to evolve with a different evolutionary model. Furthermore, because Zipf distributions often have infinite means, incorporating a Lavalette distribution (Lavalette, 1996; Popescu *et al*., 1997; Popescu, 2003), which is a non-linear extension to a Zipf distribution, is probably more appropriate for indel lengths. I suspect that a Lavalette distribution may fit empirical indel size distributions better than a Zipf distribution. Incorporating inversions into Dawg's model of molecular evolution may be useful to some researchers. Other researchers might find the addition of protein models of evolution into the program to be quite useful. Other possible places for improvement are the estimation of parameters for indel formation and developing the option for Dawg to have time reversible models of indel formation.

Dawg is a portable, flexible, and robust program for simulating DNA sequence evolution with indels. It supports recombination, the general time reversible model, gamma rate heterogeneity, invariant sites and indel formation. It is an improvement over existing programs by supporting a statistically derived model of indel formation.

ACKNOWLEDGEMENTS

The author would like to thank Wyatt Anderson, Beth Dakin, Yong-Kyu Kim, Jeff Ross-Ibarra, Paul Schliekelman, Douglas Theobald, and two anonymous reviewers for their helpful comments. This work was supported by an NSF Predoctoral Fellowship and NIH Grant 5R01 GM48528-06.

Conflict of Interest: none declared.

REFERENCES

Akaike,H. (1974) A new look at the statistical model identification. *IEEE Trans. Automat. Contr.*, **19**, 716–723.

Benner,S.A. *et al*. (1993) Empirical and structural models for insertions and deletions in the divergent evolution of proteins. *J. Mol. Biol.*, **229**, 1065–1082.

Blanchette,M. *et al*. (2004) Reconstruction large regious of an ancestral mammalian genome in silico. *Genome Res.*, **14**, 2412–2423.

Bull,J.J. *et al*. (1993) Experimental molecular evolution of bacteriophage T7. *Evolution*, **47**, 993–1007.

Chang,M.S.S and Benner,S.A (2004) Empirical analysis of protein insertions and deletions determining parameters for the correct placement of gaps in protein sequence alignments. *J. Mol. Biol.*, **341**, 617–631.

Clegg,M.T. *et al*. (1994) Rates and patterns of chloroplast DNA evolution. *Proc. Natl Acad. Sci. USA*, **91**, 6795–6801.

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

Felsenstein,J. (1984) Distance methods for inferring phylogenies: a justification. *Evolution*, **38**, 16–24.

Felsenstein,J. (2004) *Inferring Phylogenies*. Sinauer Associates, Inc , Sunderland, MA.

Gillespie,D.T. (1977) Exact stochastic simulation of coupled chemical reactions. *J. Phys. Chem.*, **81**, 2340–2361.

Goldstein,M.L. *et al*. (2004) Problems with fitting to the power-law distribution. *Eur. Phys. J. B.*, **41**, 255–258.

Gotoh,O. (1982) An improved algorithm for matching biological sequences. *J. Mol. Biol.*, **162**, 705–708.

Gu,X. and Li,W.H. (1995) The size distribution of insertions and deletions in human and rodent pseudogenes suggests the logarithmic gap penalty for sequence alignment. *J. Mol. Evol.*, **40**, 464–473.

Hall,B.G. (2005) Comparison of the accuracies of several phylogenetic methods using protein and DNA sequences. *Mol. Biol. Evol.*, **22**, 792–802.

Hasegawa,M. *et al*. (1985) Dating of the human-ape splitting by a molecular clock of mitochondrial DNA. *J. Mol. Evol.*, **21**, 160–174.

Hillis,D.M. *et al*. (1992) Experimental phylogenetics: generation of a known phylogeny. *Science*, **255**, 589–592.

Hillis,D.M. *et al*. (1994) Application and accuracy of molecular phylogenies. *Science*, **264**, 671–677.

Hudson,R.R. (2002) Generating samples under a Wright-Fisher neutral model. *Bioinformatics*, **18**, 337–338.

Jones,J.H. and Handcock,M.S. (2003) An assessment of preferential attachment as a mechanism for the growth of human sexual networks. *Proc. R. Soc. Lond. B.*, **270**, 1123–1128.

Jukes,T.H. and Cantor,C.R. (1969) Evolution of protein molecules. In Munro,H.N. (ed.), *Mammalian Protein Metabolism*. Academic Press, New York, Vol. 3, pp. 21–132.

Keightley,P.D. and Johnson,T. (2004) MCALIGN: stochastic alignment of noncoding DNA sequences based on an evolutionary model of sequence evolution. *Genome Res.*, **14**, 442–450.

Kimura,M. (1980) A simple method for estimating evolutionary rate of base substitutions through comparative studies of nucleotide sequences. *J. Mol. Evol*, **16**, 111–120.

Kimura,M. (1981) Estimation of evolutionary distances between homologous nucleotide sequences. *Proc. Natl Acad. Sci. USA*, **78**, 454–458.

Kuhner,M.K. and Felsenstein,J. (1994) A simulation comparison of phylogeny algorithms under equal and unequal evolutionary rates. *Mol. Biol. Evol.*, **11**, 459–468.

Lanave,C. *et al*. (1984) A new method for calculating evolutionary substitution rates. *J. Mol. Evol.*, **20,** 86–93.

Lavalette,D. (1996) Facteur d'impact: impartialite ou impuissance? *Internal Report, Inserm U350*, Institut Curie-Recherche, Centre Universitaire, Orsay, France.

Miklós,I. *et al*. (2004) A "long indel" model for evolutionary sequence alignment. *Mol. Biol. Evol.*, **21**, 529–540.

Miller,W. and Myers,E. (1988) Sequence comparison with concave weighting functions. *Bull. Math. Biol.*, **50**, 97–120.

Pennisi,E. (2005) Extinct genome under construction. *Science*, **308**, 1401–1402.

Press,W.H., Teukolsky,S.A., Vetterling,W.T. and Flannery,B.P. (1992) *Numerical Recipes in C: The Art of Scientific Computing*, 2 Edition. Cambridge University Press, New York, pp. 463–469.

Popescu,I.I. (2003) On a Zipf's law extension to impact factors. *Glottometrics*, **6**, 83–93.

Popescu,I.I. *et al*. (1997) On the Lavalette Ranking Law. *Rom. Rep. Phys.*, **49**, 3–27.

Rambaut,A. and Grassly,N.C. (1997) Seq-Gen: an application for the Monte Carlo simulation of DNA sequence evolution along phylogenetic trees. *Comput. Appl. Biosci.*, **13**, 235–238.

Rodríguez,F. *et al*. (1990) The general stochastic model of nucleotide substitution. *J. Theor. Biol.*, **142**, 485–501.

Schwarz,G. (1978) Estimating the dimension of a model. *Ann. Stat.*, **6**, 461–464.

Shaw,J. *et al*. (2005) The tortoise and the hare II: relative utility of 21 noncoding chloroplast DNA sequences for phylogenetic analysis. *Am. J. Bot.*, **92**, 142–166.

Stoyle,J. *et al*. (1998) Rose: generating sequence families. *Bioinformatics*, **14**, 157–163.

Swofford,D.L. (2002) *PAUP*: Phylogenetic Analysis Using Parsimony (and Other Methods) 4.0 Beta*. Sinauer Associates, Inc, Sunderland, MA.

Tamura,K. and Nei,M. (1993). Estimation of the number of nucleotide substitutions in the control region of mitochondrial DNA in humans and chimpanzees. *Mol. Biol. Evol.*, **10**, 512–526.

Tavaré,S. (1986) Some probabilistic and statistical problems on the analysis of DNA sequences. *Lect. Math. life Sci.*, **17**, 57–86.

Thompson,J.D. *et al*. (1994) CLUSTAL W: improving the sensitivity of progressive multiple sequence alignment through sequence weighting, positions-specific gap penalties and weight matrix choice. *Nucleic Acids Res.*, **22**, 4673–4680.

Thorne,J.L. *et al*. (1991) An evolutionary model for the maximum likelihood alignment of DNA sequences. *J. Mol. Evol.*, **33**, 114–124.

Thorne,J.L. *et al*. (1992) Inching toward reality: an improved likelihood model of sequence evolution. *J. Mol. Evol.*, **34**, 3–16.

Waddell,P.J. and Steel,M.A. (1997) General time-reversible distances with unequal rates across sites: mixing Γ and inverse Gaussian distributions with invariant sites. *Mol. Phylogenet. Evol.*, **8**, 398–414.

Yang,Z. (1993) Maximum-likelihood estimation of phylogeny from DNA sequences when substitution rates differ over sites. *Mol. Biol. Evol.*, **10**, 1396–1401.

Yang,Z. (1994) Estimating the pattern of nucleotide substitution. *J. Mol. Evol.*, **39**, 105–111.

Yang,Z. (1995) On the general reversible Markov process model of nucleotide substitution: a reply to Saccone *et al*. *J. Mol. Evol.*, **41**, 254–255.

Yang,Z. (1997) PAML: a program package for phylogenetic analysis by maximum likelihood. *Comput. Appl. Biosci.*, **13**, 555–556.

Zhang,Z. and Gerstein,M. (2003) Patterns of nucleotide substitution, insertion and deletion in the human genome inferred from pseudogenes. *Nucleic Acids Res.*, **31**, 5338–5348.