BIOINFORMATICS

Vol. 00 no. 00 2006
Pages 1–2

NETGEN: Generating Phylogenetic Networks with Diploid Hybrids

M.M. Morin[*] *and B.M.E. Moret*

Department of Computer Science, University of New Mexico, Albuquerque, New Mexico, USA

ABSTRACT

Summary: NETGEN is an event-driven simulator that creates phylogenetic networks by extending the birth-death model to include diploid hybridizations. DNA sequences are evolved in conjunction with the topology, enabling hybridization decisions to be based on contemporary evolutionary distances. NETGEN supports variable rate lineages, root sequence specification, outgroup generation, and many other options. This note describes the NETGEN application and proposes an extension of the Newick format to accommodate phylogenetic networks.

Availability: NETGEN is written in C and is available in source form at `http://www.phylo.unm.edu/~morin/`.

Contact: morin@cs.unm.edu

Phylogenetic simulation and reconstruction methods usually assume a tree topology with independent events of speciation and extinction. However, reticulate events such as hybridization and lateral gene transfer are known to occur along some evolutionary paths. The presence of such events transforms a tree into an evolutionary *network*. Over the last decade, phylogenetic networks have received increased attention (Bandelt et al., 1999; Gusfield, 2005; Moret et al., 2004), yet, with the exception of `SplitsTree` (Huson and Bryant, 2006) and `T-REX` (Makarenkov, 2001), software aimed at networks remains limited; in particular, no simulator exists to generate evolutionary networks, a lack that hampers the assessment and further development of network reconstruction methods.

NETGEN (Morin, 2006) is an event-driven simulator that extends the birth-death model (Renshaw, 1991) to include diploid hybrids. Hybridization decisions are made according to the DNA sequences of contemporary lineages; thus, in contrast to tree-based simulators, sequences co-evolve with the topology. Options include variable rate lineages, hybridization limits, choice of root sequences, outgroup generation, and deviation from ultrametricity. Written in C, NETGEN is command-line driven with text files for input and output. It is composed of three modules. The first module, `NG`, contains the simulation logic; the second provides the interface between `NG` and the third, which is a sequence generator. We have chosen the popular `Seq-Gen` (Rambaut and Grassly, 1997) as our sequence generator, but any desired generator can be used with modification to the second module. Robust simulation studies can be conducted as NETGEN utilizes process identifiers to distinguish the communication files, while reproducibility is ensured by employing a pseudorandom number generator with user-specified seed.

A priority queue tracks all events. Currently three types of events are implemented: birth, death, and diploid hybridization. The queue structure enables the addition of other types of events such as lateral gene transfer, polyploid hybridizations, and mass extinction. Events are scheduled according to exponential interarrival times. At the creation of a new lineage, event times are generated for each of the event types using the corresponding user-specified rates and the earliest of these generated events is retained and placed on the queue.

The network generation process starts with two active lineages, each with a future event scheduled on the queue. The simulation advances by processing the next queued event. Once completed, events are logged and removed from the queue. This process continues until either the desired number of extant taxa is reached or the event queue is empty. In the former, normal case, we choose the *final end time* randomly between the last completed event and the first remaining event on the queue (if any); this choice avoids artificially short branch lengths and matches reality well, since the sampling of modern taxa occurs at what amounts to a randomly chosen time between two evolutionary events.

NETGEN tracks both evolutionary and clock-based branch lengths. When requesting a sequence, the evolutionary branch length is passed to the sequence generator so the appropriate amount of evolutionary change can be simulated. The clock time is required to order the queue and to identify all active lineages when searching for a potential second parent in a hybridization event.

The network aspect of NETGEN comes from the implementation of diploid hybridization. As described above, a lineage is assigned a future event when it is created. If the scheduled event is a hybridization and the maximum number of allowed hybrids has not been reached, a second contemporary lineage is sought to form a pair of parent lineages for the hybrid. First branch lengths and sequences are updated for all active lineages. Then the lineage closest (specified either in terms of sequence similarity or evolutionary distance) to the original lineage is identified; if the user-specified threshold is met, the hybridization is allowed to proceed. Each parent randomly contributes half of its chromatids in each chromosome to create a new species, while propagating its original lineage (Figure 1). The continuing lineage of the first parent and the hybridized lineage are assigned their future events based upon the specified event rates. The lineage of the second parent already had an event queued prior to the

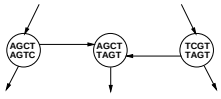

Fig. 1. Graphical representation of a diploid hybrid event. Each parent donates half of its chromatid sequences to the hybrid species.

[*] to whom correspondence should be addressed

hybridization and so proceeds without requiring the generation of a new event.

NETGEN allows the user to declare the number of chromosomes and chromatids (memory permitting) as well as the root sequence(s). If root sequence information is not provided, it is created by the sequence generator. Outgroups can be generated according to user-specified (dis)similarity bounds. NETGEN creates by default ultrametric networks (with respect to evolutionary branch lengths) and fixed event rates across all lineages—as is typical in the phylogenetic community—, but it also allows for the exploration of different model behaviors. If non-ultrametric networks are desired, the evolutionary branch lengths are deviated according to a gamma distribution. For the variable-rate lineage option, each lineage is assigned positive event rates from normal distributions specified by the user.

Under the birth-death model, the mean population size is described by $n = n_0 e^{(B-D)\cdot t}$, where n_0 is the initial population size, t is the time, and B and D are the birth and death rates, respectively (Renshaw, 1991). Like a birth, hybridization adds one new lineage to the phylogeny; therefore the new mean population size is now $n = n_0 e^{(B-D+H)\cdot t}$, where H is the hybridization rate. Thus the natural log of the population size grows linearly in time with a slope of $B - D + H$—something we verified experimentally for 25,000 extant taxa and a variety of values for B, D, and H (see Table 1).

With phylogenetic network visualization in mind, we have developed a modified version of the Newick tree format which enables us to describe networks. Figure 2 shows a tree and a network, with corresponding Newick strings. Node `3` of the network is a hybrid node, denoted by the addition of `#H`. Lateral gene transfer, another reticulate evolutionary event, can be represented in this format as well by annotating such nodes with `#LGT`. Generating this format requires knowledge about the node type and which nodes have been visited, so as to avoid traversing their subtrees more than once. This format is a depth-first traversal and reverts to Newick tree format when network nodes are not present.

Rates			Expected Growth	Measured Growth
B	D	H	$B - D + H$	Fitted Slope $\pm\sigma$
2.0	-	-	2.0	2.004 $\pm$0.013
2.0	0.5	-	1.5	1.498 $\pm$0.020
-	-	1.0	1.0	0.998 $\pm$0.008
2.0	-	1.0	3.0	2.999 $\pm$0.027
2.0	0.5	1.0	2.5	2.492 $\pm$0.031

Table 1. Predicted and observed mean population growth rates. Ten iterations per scenario, each with 25,0000 extant taxa.

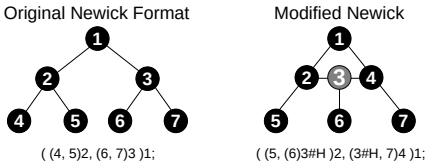

Fig. 2. Regular versus Modified Newick formats. Phylogeny on the right includes a hybrid node (3) denoted by $\#H$.

NETGEN is a novel simulator for creating phylogenetic networks with diploid hybrids. DNA sequences are evolved with the topology allowing the hybridization decisions to be based on sequence similarity. In combination with the tripartition measure of (Moret et al., 2004), NETGEN enables us to assess the performance of network reconstruction tools. Planned enhancements include adding other events such as lateral gene transfer, providing more complex models of birth (e.g. inheritability of speciation rates (Heard, 1996)), incorporating the tripartition measure of (Moret et al., 2004), and packaging the whole as a module for Mesquite (Maddison and Maddison, 2001).

ACKNOWLEDGEMENT

This work is funded by the National Science Foundation under grants IIS 01-21377, DEB 01-20709, and EF 03-31654.

REFERENCES

Bandelt, H., Forster, P., Roehl, A., (1999) Median-joining networks for inferring intraspecific phylogenies. Mol. Biol. Evol. 16 (1), 37–48.

Gusfield, D., (2005) Optimal, efficient reconstruction of root-unknown phylogenetic networks with constrained and structured recombination. J. Comput. Syst. Sci. 70 (3), 381–398.

Heard, S., (1996) Patterns in phylogenetic tree balance with variable and evolving speciation rates. Evol. 50, 2141–2148.

Huson, D., Bryant, D., (2006) Application of phylogenetic networks in evolutionary studies. Mol. Biol. Evol. 2, 254–267.

Maddison, W., Maddison, D., (2001) Mesquite: a modular system for evolutionary analyses, version 0.98. `mesquiteproject.org`.

Makarenkov, V., (2001) T-REX: Reconstructing and visualizing phylogenetic trees and reticulation networks. Bioinformatics 17 (7), 664–668.

Moret, B., Nakhleh, L., Warnow, T., Linder, C., Tholse, A., Padolina, A., Sun, J., Timme, R., (2004) Phylogenetic networks: Modeling, reconstructability, and accuracy. IEEE/ACM Transactions on Computational Biology and Bioinformatics 1 (1), 13–23.

Morin, M., (2006) NetGen Release Notes, Epsilon Version (Phylogenetic Network Generation Application). Tech. Rep. TR-CS-2006-05, Univ. of New Mexico, Albuquerque, New Mexico.

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

Renshaw, E., (1991) Modelling Biological Populations in Space and Time. Cambridge University Press.