BIOINFORMATICS APPLICATIONS NOTE

Vol. 26 no. 17 2010, pages 2200–2201
doi:10.1093/bioinformatics/btq355

Genetics and population analysis

Advance Access publication July 29, 2010

TreesimJ: a flexible, forward time population genetic simulator

Brendan O'Fallon

Department of Genome Sciences, Foege Building S-250, Box 355065 3720 15th Ave NE,
Seattle WA 98195-5065, USA

Associate Editor: Jeffrey Barrett

ABSTRACT

Motivation: Most population genetic simulators fall into one of two classes, backward time simulators that quickly generate trees but accommodate only relatively simple selective and demographic regimes, and forward simulators that allow for a broader range of evolutionary scenarios but which cannot produce genealogies. Thus, few tools are available that allow for producing genealogies under arbitrarily complex selective and demographic models.

Results: TreesimJ is a forward time population genetic simulator that allows for sampling of genealogies, genetic data and many population parameters from populations evolving under complex evolutionary scenarios. The application provides many fitness and demographic models and new models are easy to develop. Data collection is performed by a variety of independently configurable collectors which periodically sample the population and record statistics. Output options include writing traces, histograms and summary statistics from the data collectors in addition to sampled genetic sequences and genealogies.

Summary: TreesimJ allows researchers to easily sample and analyze gene genealogies and related data from populations evolving under a wide variety of selective and demographic regimes. It is likely to be useful for population genetic researchers seeking to understand the links between evolutionary and demographic forces, genealogical structure and the resulting patterns of genetic variation.

Availability: TreesimJ home : http://staff.washington.edu/brendano/treesimj. Source and developer resources: http://code.google.com/p/treesimj

Contact: brendano@u.washington.edu

Received on May 7, 2010; revised on June 22, 2010; accepted on June 29, 2010

1 INTRODUCTION

The simulation of evolving populations plays an important role in many aspects of biological inquiry, including the validation of new inference methods, comparison of empirical data to those generated under known conditions and exploration of the dynamics of complex evolutionary processes. Many publicly available tools exist to generate nucleotide sequences under a variety of scenarios, and they are often classified by the direction of time assumed. 'Backward' time or 'coalescent' simulators track only the history of a sample of individuals, and thus can rapidly generate both trees and sequence data from populations of arbitrary size (e.g. Hudson, 2002; Laval and Excoffier, 2004; Spencer and Coop, 2004). However, these simulators require some assumptions about the nature of the evolutionary process that may not be met in every scenario, especially regarding complex selective regimes. Alternatively, 'forward time' simulators model time as flowing in the usual direction and typically track the state of the entire population (e.g. Balloux, 2001; Carvajal-Rodrguez, 2008; Guillaume and Rougemont, 2006; Hernandez, 2008; Hey, 2004). While forward time simulation is less computationally efficient than coalescent simulation, a greater range of evolutionary models can be considered. However, the burden of tracking the entire state of the population often means that the genealogical trees underlying the population are not tracked. Thus, forward simulators typically do not allow access to the genealogies that underly the data, and therefore examination of the trees generated under complex evolutionary scenarios has remained difficult using current simulation software.

TreesimJ is a simulation tool that addresses this limitation by tracking, in forward time, an evolving population as well as all ancestors of the 'current' generation, thereby combining the flexibility of a forward simulator with the genealogical capabilities typical of backward simulators. This combination allows researchers to investigate how the properties of non-recombining genealogies are affected by evolutionary forces, for instance, how variation in the strength of selection coefficients at multiple sites affects the time to most recent common ancestor (TMRCA) of a sample. Alternatively, various measures of tree shape (for instance, tree depth, skewness or the mean time for two individuals to first share a common ancestor) are easily collected and viewable as time traces or as a histogram of the aggregated values. In addition to tracking genealogies, TreesimJ can also be used to collect a variety of other statistics regarding the population, including many familiar population genetic values (nucleotide diversity, Tajima's D, haplotype diversity, etc.).

2 FEATURES

TreesimJ has several features that distinguish it from other forward population genetic simulators. First, as mentioned above, TreesimJ tracks the state of all ancestors of the present generation and allows for sampling of genealogies from the population tree. The resulting genealogies can be analyzed by a variety of methods, and may also be written to a log file or as separate files to facilitate further dissection. If the individuals in the population are endowed with DNA, fasta-formatted files with the DNA from the tree tips may be written along with each tree. With DNA and true trees in hand users may easily examine how tree shape affects resulting patterns of genetic diversity, or investigate the effectiveness of phylogenetic inference packages in reproducing the true tree.

Second, the application can be controlled through a simple and intuitive graphical user interface (GUI). The interface allows users unfamiliar with the command line or unaccustomed to editing text input files to easily control the behavior of the application.

For situations in which a GUI is not desirable, TreesimJ allows for execution in 'batch mode', with the options provided in an easily generated settings file.

Third, TreesimJ is very flexible in several respects. The application comes with a number of alternative models of fitness evolution, including neutral evolution, one-locus, two-allele models, quantitative genetic evolution and DNA evolution. The DNA models in turn support a variety of mutational (e.g. Jukes–Cantor, F84, TN93, etc.) and selective schemes. Similarly, users may choose from a variety of demographic models describing population size change. Finally, a large number of 'data collectors' are provided which periodically collect and emit information regarding the population. The collection rates and other properties of these statistics are all independently configurable.

While many models and data collectors are provided 'out of the box', a final strength of TreesimJ is the ease with which new models may be created. Written in Java, an object-oriented language, nearly all of the models are pluggable objects. Creation of a new fitness model, for instance, requires constructing a single new class that derives from the appropriate base class and implements a small number of new functions (a single line of code elsewhere is required to add the new model to the list of available models). Demographic models and new data collectors are similarly easy to construct. To facilitate the development of new models, the code is well documented and freely available for browsing and download at the project hosting site (code.google.com/p/treesimj), where a developer's guide is also available.

3 SUMMARY

TreesimJ is a flexible, easy to use population genetic simulator that tracks the ancestry of the entire population in question. Like all genetic simulators, TreesimJ has distinct strengths and weaknesses. The application's strengths are its ease of use, developer-friendly code base and the ability to track, sample and analyze the properties of ancestral trees. However, TreesimJ can be slow and memory hungry. On a Macintosh system with a 2.26 GHz Xeon ('Nahelem' architecture) CPU, a population of 1000 individuals with DNA of length 1000 runs at nearly 2000 generations per second and requires roughly 350 MB of RAM for efficient performance. However, when DNA length is increased to 50 000 the simulation speed drops to only 450 generations per second and nearly 1 GB of available RAM is required. In general, populations with more than 10 000 individuals are likely to be too slow to allow sufficient sampling of characteristics. Other limitations stem from the assumption that each individual has exactly one parent, thus diploidy, recombination and sexual selection are currently absent.

While no single simulation tool will address the needs of all users, TreesimJ is likely to be desirable for researchers interested in the connections between the structure of gene genealogies, patterns of genetic variation and the evolutionary forces affecting a population. Additionally, it is also likely to be valuable to those interested in validating inference and reconstruction methods since it can easily produce simulated datasets along with true trees under a variety of models. Finally, we hope that some of the limitations described above will be addressed as the platform continues to develop, both from contributions by the author as well as other parties.

ACKNOWLEDGEMENTS

I would like to thank Mary Kuhner, Elizabeth Walkup, Jim McGill and Bob Giansiracusa for their comments on the manuscript as well as testing early versions of the program.

Funding: National Science Foundation Postdoctoral Research Fellowship 0906018.

Conflict of Interest: none declared.

REFERENCES

Balloux,F. (2001) EASYPOP (version 1.7): a computer program for population genetics simulations. *J. Hered.*, **92**, 301–302.

Carvajal-Rodriguez,A. (2008) GENOMEPOP: a program to simulate genomes in populations. *BMC Bioinformatics*, **9**, 223.

Guillaume,F. and Rougemont,J. (2006) Nemo: an evolutionary and population genetics programming framework. *Bioinformatics*, **22**, 2556–2557.

Hernandez,R.D. (2008) A flexible forward simulator for populations subject to selection and demography. *Bioinformatics*, **24**, 2786–2787.

Hey,J. (2004) FPG: a computer program for forward population genetics simulation. Available at http://genfaculty.rutgers.edu/hey/software.

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

Laval,G. and Excoffier,L. (2004) SimCoal 2: a program to simulate genomic diversity over large recombining regions in a subdivided population with a complex history. *Bioinformatics*, **20**, 2485–2487.

Spencer,C.C.A and Coop,G. (2004) SelSim: a program to simulate population genetic data with natural selection and recombination. *Bioinformatics*, **20**, 3673–3675.