GSpace: an exact coalescence simulator of recombining genomes under isolation by distance

Thimothée Virgoulay, François Rousset, Camille Noûs, Raphaël Leblois

Bioinformatics, 2021, 1–3
doi: 10.1093/bioinformatics/btab261
Advance Access Publication Date: 8 May 2021
Applications Note

Genetics and population analysis

GSpace: an exact coalescence simulator of recombining genomes under isolation by distance

Thimothée Virgoulay [1,2,*], François Rousset [1], Camille Noûs[3] and Raphaël Leblois [2]

[1]Institut des Sciences de l'Evolution, Univ Montpellier, CNRS, IRD, EPHE, Montpellier, France,[2]CBGP, INRA, CIRAD, IRD, Montpellier SupAgro, Univ Montpellier, Montferrier sur Lez, France and [3]Laboratoire Cogitamus, Univ Montpellier, Montpellier, France

*To whom correspondence should be addressed.

Associate Editor: Russell Schwartz
Received on December 17, 2020; revised on April 16, 2021; accepted on April 27, 2021editorial decision on April 20, 2021;

Abstract

Motivation: Simulation-based inference can bypass the limitations of statistical methods based on analytical approximations, but software allowing simulation of structured population genetic data without the classical *n*-coalescent approximations (such as those following from assuming large population size) are scarce or slow.
Results: We present GSpace, a simulator for genomic data, based on a generation-by-generation coalescence algorithm taking into account small population size, recombination and isolation by distance.
Availability and implementation: Freely available at site web INRAe (http://www1.montpellier.inra.fr/CBGP/software/gspace/download.html).
Contact: thimothee.virgoulay@umontpellier.fr

1 Introduction

GSpace is a program that can simulate neutral genomic data with recombination under a wide range of demographic models. It is based on a backward in time generation-by-generation (gen-by-gen) approach, coupled with an efficient recombination algorithm and flexible models for dispersal and subpopulation sizes. It simulates the ancestry of a sample of haploid or diploid individuals (in both cases following a standard haplo-diploid sexual life cycle), carrying one or more chromosomes.

Individual dispersal is generally restricted in space in natural populations (isolation by distance: Endler, 1979; Rousset, 1997; Wright, 1943). To represent this fact, GSpace considers a lattice of subpopulations of any size (down to single individuals) connected by limited dispersal, according to different possible dispersal distributions, including in particular fat-tailed distributions, such as the Zeta (discretized Pareto, Patil and Joshi, 1968) and the Sichel (Chesson and Lee, 2005).

The case where each subpopulation on the lattice hosts a single individual or a mating pair and dispersal is mostly restricted to a few steps apart is suitable to represent a range of territorial species inhabiting a continuous habitat (e.g. Rousset, 2000), but cannot be simulated when considering extensions of Kingman's (1982) *n*-coalescent which assume large sub-population sizes and small migration rates. To simulate small population sizes and high dispersal rates without biases (Nelson *et al.*, 2020), coalescence probabilities exact for small population size (Fu, 2006) must be used in a gen-by-gen simulation until the common ancestors of the whole simulated sample have been found. Such simulations are required to assess any inference framework which might for example allow separate estimation of sub-population size, mutation and migration probabilities, something that is not possible under *n*-coalescent approximations. In all these respects, GSpace retains and extends for genomic data some of the previous features of the IBDsim software (Leblois *et al.*, 2009). The current version considers only time-homogeneous models but time-heterogeneous models will be implemented in future versions similarly to IBDsim.

As gen-by-gen algorithms are expected to be slower than those involving *n*-coalescent approximations, we performed simulations to check the feasibility of simulating genomic data by such algorithms, and compared computation times with those of alternative software based on *n*-coalescent approximations, such as msprime (Kelleher *et al.*, 2016), FastSimcoal2 (Excoffier *et al.*, 2013), exact coalescence algorithms implemented as DTWF in msprime python package [back-in-time Wright-Fisher simulator, Nelson *et al.* (2020)], IBDSim (Leblois *et al.*, 2009) and forward algorithms, such as SimBit (Matthey-Doret, 2020).

The gen-by-gen algorithm in GSpace is slower than those involving *n*-coalescent approximations but much faster than IBDSim or forward simulators like SimBit in most cases (see Section 4).

2 Implementation

GSpace combines some parts of the modified Hudson's algorithm (Hudson, 1983) for recombination and coalescence implemented in msprime with previous features from IBDSim, in a new implementation in modern C++, as follows. At each generation going backward in time,

the program considers all possible migration, recombination and coalescence events, until all common ancestors have been found. Because neutral genetic data are simulated, genetic states do not affect genealogical trees and mutations can then be added downwards to the gene tree of each chromosome segment that did not recombine. Implementation details of such algorithm can be found in Kelleher *et al.* (2016) and Nelson *et al.* (2020) for the approximated and gen-by-gen algorithms for coalescence with recombination, respectively; and in Leblois *et al.* (2009) for gen-by-gen algorithms under isolation by distance. We only highlight below what can make GSpace different from other software.

At each generation t, the coordinates of the parent of each individual carrying ancestral lineages are randomly drawn in a 2D backward dispersal distribution of the position of a parent given the position of the lineage. The backward distributions are deduced by assuming that dispersal occurs independently in each dimension forward in time, and can automatically handle spatial heterogeneity (i.e. different forward migration rates and size of sub-populations on the lattice) as well as various edge effects. The program can consider (i) uniform, geometric and discretized Gaussian, Zeta and Sichel forward dispersal distributions, including the stepping stone and island models as special cases, as well as (ii) a custom forward migration rate matrix. Each chromosome harbors multiple discrete potentially recombining sites and the program handles multiple recombination events per chromosome, even in a single generation. When a recombination event occurs in a diploid genome in the backward simulation, the segments on each side of the recombination point originate from each of the two parental chromosomes and have a distinct coalescence history further backward in time. When a coalescence event occurs, ancestral segments of all descendant chromosomes have a unique parental segment and share a common coalescence and migration history until a recombination event occurs. The combination of such gen-by-gen diploid coalescence, migration and recombination algorithms simulates the exact patterns of linkage disequilibrium expected under a haplo-diploid life cycle.

Mutations are then added independently on each gene tree, going forward in time on each branch, from the common ancestor to the leaves. As the underlying algorithm assumes a finite number of mutable sites GSpace can handle numerous nucleotidic and allelic mutation models (e.g. IAM, KAM, JC69, see user manual for more models) but not the infinite site model.

3 Compilation, automated checks, inputs and outputs

The program is written in modern C++ (17) and can be compiled on any operating system with a modern compiler (g++ $\geq$ 7.5, clang $\geq$ 6.0) with simple command line arguments, or using the CMake build system (both the command line arguments and the CMake commands are provided in the manual). The CMake build includes unit tests for each part of the program and functional tests comparing simulation outcomes in terms of probability of identity of pairs of genes at one and two loci to theoretical results (see Rousset, 2004 and Vitalis and Couvet, 2001).

GSpace's runs can be controlled both by a settings file and by command-line arguments, which together allow the easy specification of many parameters, and quick changes of selected parameters between simulations. The settings file is exacxtread first, and allows the user to control all options of GSpace (detailed in the user manual). These options can then be altered by the command line arguments. Results can be saved in three different file formats for individual genetic data: Genepop for allelic data, Fasta and VCF (v4.3) for sequence data; as well as in the new binary treeSequence format (see tskit documentation) for efficient storage of trees with mutations.

4 Comparison

Gspace is, at the time of writing, the only gen-by-gen coalescence simulator specifically designed to handle recombination and allowing easy specification of various forward dispersal distributions. Thus, it cannot be compared in terms of computation time to other simulators not sharing such features, but it has been compared to algorithms from five other simulators in simpler cases: the n-coalescent approximations implemented in msprime v1.0.0a5 ('msprime') and FastSimcoal2 v2.6.0.3 ('fsc2'); the gen-by-gen algorithms implemented in msprime v1.0.0a5 ('DTWF') and IBDSim v2.0; and the forward simulator SimBit v3.9.13. Results for IBDSim are not detailed here because it cannot consider recombination and is not designed to handle long DNA sequences (e.g. $> 10^5$ bp). However, without recombination and for many allelic loci, GSpace is two to fifty time faster. Other simulations are detailed in Table 1 and show that although GSpace is not the fastest simulator, its speed approaches that of the approximate simulators rather than that of other generation-by generation ones.

Table 1. Comparison of computation times between GSpace and other simulators under three different demographic and mutational schemes.

	Method (see text for details)				
Case	msprime	fsc2	GSpace	DTWF	SimBit
A	0.320	3.959	6.231	7.096	56.781
B	14.507	5.664	7.471	36.335	52.121
C	0.048	0.016	0.028	2.459	39.055

Note: Mean run time in seconds over 100 (10 for SimBit) replicates for the simulation of a sample of 1000 haploid individuals carrying a single chromosome of 10^7 base pairs, with mutation and recombination rates of 10^{-8} per generation per site under: (A) a Wright–Fisher model with a population size of 10000 haploid individuals; (B and C) an island model with 20 subpopulations of 500 haploid individuals each and 50 sampled chromosomes of (B) 10^7 base pairs or (C) with 10^4 base pairs

Acknowledgements

We thank A. Dehne-Garcia, F.-D. Collin and M. Navascues for initial discussions on algorithms and code, as well as J. Kelleher and P. Ralph for constructive comments and help with tskit during the review process.

Funding

This work used the following HPC platforms: INRA MIGALE (http://migale.jouy.inra.fr) and GENOTOUL (Toulouse Midi-Pyrénées), Montpellier Bioinformatics Biodiversity supported by the LabEx CeMEB (ANR-10-LABX-04-01), and CBGP host platform. All authors were supported by the Agence Nationale de la Recherche (RL & TV: projects GENOSPACE ANR-16-CE02-0008 and Labex Cemeb ProLag; FR & RL: project INTROSPEC ANR-19-CE02-0011).

Conflict of Interest: none declared.

References

Chesson,P. and Lee,C.T. (2005) Families of discrete kernels for modeling dispersal. *Theor. Popul. Biol.*, **67**, 241–256.
Endler,J.A. (1979) Gene flow and life history patterns. *Genetics*, **93**, 263–284.
Excoffier,L. *et al.* (2013) Robust demographic inference from genomic and snp data. *PLoS Genet.*, **9**, e1003905.
Fu,Y.-X. (2006) Exact coalescent for the wright–fisher model. *Theor. Popul. Biol.*, **69**, 385–394.
Hudson,R.R. (1983) Properties of a neutral allele model with intragenic recombination. *Theor. Popul. Biol.*, **23**, 183–201.
Kelleher,J. *et al.* (2016) Efficient coalescent simulation and genealogical analysis for large sample sizes. *PLoS Comput. Biol.*, **12**, e1004842.
Kingman,J. (1982) The coalescent. *Stoch. Process. Their Appl.*, **13**, 235–248.

Leblois,R. *et al.* (2009) Ibdsim: a computer program to simulate genotypic data under isolation by distance. *Mol. Ecol. Res*., **9**, 107–109.

Matthey-Doret, R. (2021) SimBit: A high performance, flexible and easy-to-use population genetic simulator. *Mol Ecol Resour.* 10.1111/1755-0998.13372

Nelson,D. *et al.* (2020) Accounting for long-range correlations in genome-wide simulations of large cohorts. *PLoS Genet*., **16**, e1008619.

Patil,G.P. and Joshi,S.W. (1968) *A dictionary and bibliography of discrete distributions*. Published for the International Statistical Institute by Oliver and Boyd Edinburgh.

Rousset,F. (1997) Genetic differentiation and estimation of gene flow from f-statistics under isolation by distance. *Genetics*, **145**, 1219–1228.

Rousset,F. (2000) Genetic differentiation between individuals. *J. Evol. Biol*., **13**, 58–62.

Rousset,F. (2004) *Genetic Structure and Selection in Subdivided Populations*. Monographs in population biology. Princeton University Press, Princeton University, New Jersey.

Vitalis,R. and Couvet,D. (2001) Estimation of effective population size and migration rate from one- and two-locus identity measures. *Genetics*, **157**, 911–925.

Wright,S. (1943) Isolation by distance. *Genetics*, **28**, 114–138.