

Genetic and population analysis

# Admix'em: a flexible framework for forward-time simulations of hybrid populations with selection and mate choice

Rongfeng Cui<sup>1,2,3,\*</sup>, Molly Schumer<sup>2,†</sup> and Gil G. Rosenthal<sup>1,2</sup>

<sup>1</sup>Department of Biology, Texas A&M University, College Station, TX 77843, USA, <sup>2</sup>Centro De Investigaciones Científicas De Las Huastecas "Aguazarca", Calnali, Mexico and <sup>3</sup>Max Planck Institute for the Biology of Aging, Cologne, Germany

\*To whom correspondence should be addressed.

†Present address: Department of Ecology and Evolutionary Biology, Princeton University, Princeton, NJ, USA

Associate Editor: Janet Kelso

Received on 28 April 2015; revised on 18 October 2015; accepted on 25 November 2015

## Abstract

**Summary:** We introduce a new forward-time simulator, Admix'em, that allows for rapid and realistic simulations of admixed populations with selection. Complex selection can be achieved through user-defined fitness and mating-preference probability functions. Users can specify realistic genomic landscapes and model neutral SNPs in addition to sites under selection. Admix'em is designed to simulate selection in admixed populations but can also be used as a general population simulator. Usage and examples are in the supplement.

**Availability and Implementation:** C++ and OpenMP, supports 64-bit Linux/Unix-like platforms. <https://github.com/melop/admixem>.

**Contact:** [rcui@age.mpg.de](mailto:rcui@age.mpg.de)

**Supplementary information:** [Supplementary data](#) are available at *Bioinformatics* online.

## 1 Introduction

Population simulators are invaluable to empiricists, since they can be used to test alternative hypotheses and estimate population parameters (e.g. approximate Bayesian computation; [Beaumont \*et al.\*, 2002](#)). Population simulators can also be used to design experiments and estimate power. Here, we present a new population simulator called Admix'em. Though primarily designed to simulate selection during admixture, it is also a general-purpose population simulator. Admix'em differs from similar programs (e.g. *simuPop*, *Nemo*, *QuantiNemo*, *ForSim*, *FFPopSim*) because it allows users to easily simulate complex selection (e.g. epistasis, selection on a phenotype controlled by many loci) and non-random mating (e.g. frequency dependent preferences, sexual imprinting). For comparison to other simulators, see [Supplementary Table S1](#).

To be useful to researchers, population simulators should be fast, easy to use, and able to simulate scenarios of realistic complexity. Currently available coalescent simulators are computationally efficient

but have limited implementation of selection. For example, coalescent simulators such as *ms* ([Hudson, 2002](#)) and *Macs* ([Chen \*et al.\*, 2009](#)) do not implement selection, while others such as *msms* ([Ewing and Hermisson, 2010](#)) only allow for selection at a single locus. In contrast, forward time simulators can simulate more biologically realistic selection scenarios, but are inevitably less efficient. Several previously developed forward time simulators allow users to incorporate demographic processes and selection at multiple loci (e.g. [Guillaume and Rougemont, 2006](#)). However, existing forward time simulators either offer limited choices of selection scenarios or require further programming by the user (e.g. [Peng and Amos, 2008](#)), reducing usability.

In contrast, Admix'em can simulate complex selection without requiring further programming. For example, Admix'em can implement sexual selection as a function of social and genetic background (e.g. frequency dependent or learned sexual selection) and allows incorporation of indirect genetic effects ([Wolf \*et al.\*, 1998](#)). To simulate such scenarios, users need only modify selection functions in a

text file (see [Supplementary Materials](#)). Though complex selection scenarios require additional computational resources, Admix'em is easily parallelizable. The program is implemented in C++ with an object-oriented structure, is easily compiled, self-contained, and includes an interactive interface for simulation setup and scripts to export data. Recently, we used Admix'em to develop a new model of hybrid speciation (Schumer *et al.*, 2015).

## 2 Program overview

Admix'em makes the following assumptions: (i) non-overlapping generations; (ii) diploid sexual individuals; (iii) only one sex (assumed to be female) exerts sexual selection. See [Supplementary Materials A–C](#) for performance and validation.

The program runs for a user-defined number of generations. Each generation, females mate (males chosen with replacement) and produce a Poisson distributed number of offspring (default mean = 2/female) which experience selection (see below). After selection, if the population is smaller than the user-defined carrying-capacity, additional offspring from the same mating events are drawn from a Poisson distribution with a new mean defined as (carrying capacity–current population size)/number of females. This process is repeated until the population reaches carrying capacity or females deplete all gametes (number is user-defined).

### 2.1 Configuration

Admix'em tracks genotypes and ancestry at all marker loci for each individual. DNA or protein sequences are not explicitly modeled. Before running the program, users generate files through an interactive interface and specify parameters including the number and lengths of chromosomes and the number of markers. Users specify loci under selection and other simulation details in provided template files ([Supplementary Materials D](#)), which are easily modifiable by text editors and spreadsheet editors.

Admix'em accommodates two initial populations (with user-defined ancestry proportions and size) and an unlimited number of parental or hybrid populations that can be formed from these initial populations. Migration is defined by the user, allowing simulations of populations ranging from stepping stone to island models. Two options are available for migration: (i) constant number or (ii) number of migrants drawn from a binomial distribution. Migration rates can vary between generations and populations.

### 2.2 Recombination

Admix'em can simulate recombination with biologically realistic genome (~2000 cM) and population sizes (e.g. 10 000), even with many markers (>10 000; [Supplementary Materials A, B](#)). Users specify the expected number of recombination events per chromosome per meiosis. Recombination can be implemented in two ways. With uniform recombination, the location of each recombination breakpoint is randomly determined during run time. Alternatively, users can set candidate breakpoints (e.g. using a genetic map) and the cumulative recombination probability is used to determine the stochastic selection of breakpoints during recombination. Admix'em does not track recombination breakpoints but their locations can be inferred from genotypes at marker loci.

### 2.3 Mutation

Admix'em supports mutation of specified loci with locus-specific mutation probabilities. The new allelic state post-mutation is specified through a user-defined randomization function.

## 2.4 Phenotypes and selection

Users can define an arbitrary number of 'phenotypes,' ranging from the genotype at a single locus to traits determined by many loci, which are optionally under selection (see below). This flexibility makes it possible to simulate complex traits, dominance, epistasis and much more (see [Supplementary Materials D](#)). In addition, users can add noise to simulate environmental effects.

Users can specify individual fitness ( $\omega$ ) as an arbitrary function of any combination of genotypes or phenotypes, defined on a 0–1 scale. The value of  $\omega$  is the probability that the individual will survive to mate. In addition, Admix'em allows users to define phenotypes and fitness based on population-level characteristics such as phenotype frequency, and can model phenotypes that do not have a genetic basis ([Supplementary Materials D](#)).

Admix'em allows for realistic implementation of sexual selection. Users can specify mating probabilities as a function of an individual's phenotype, or the phenotype distribution of the current or previous population ([Supplementary Materials D](#)). Population statistics are supplied during run-time, allowing learned preferences dependent on phenotype distributions (Verzijden *et al.*, 2012) and frequency-dependent selection (Fitzpatrick *et al.*, 2007). Mate selection is conducted randomly with a user-defined number of mates to sample. Mate preference and fitness can be population and generation specific.

### 2.5 Output

Admix'em outputs individual genotypes, phenotypes, individual fitness and parent-offspring relationships. We provide scripts to convert to Admixmap format (McKeigue *et al.*, 2000), and to calculate  $F_{IS}$  statistics, hybrid index and phenotype distributions.

### 2.6 Parallelization

Parallelization is with OpenMP on a multi-core, shared-memory computer and performance scales linearly with number of CPUs.

## 3 Validation and performance

We validate Admix'em's performance for neutral loci, selected loci, epistatic selection, and decay in linkage disequilibrium using population genetic models ([Supplementary Materials A](#)).

## Acknowledgements

Simulations were performed on the Brazos cluster at Texas A&M and the Della cluster at Princeton. We thank Adam Jones and Alisa Sedghifar for comments.

## Funding

National Science Foundation (NSF) DDIG IOS-1210324 to R.C. NSF DDIG DEB-1405232 to M.S. NSF IOS-0923825 to G.G.R.

*Conflict of Interest:* none declared.

## References

- Beaumont, M.A. *et al.* (2002) Approximate Bayesian computation in population genetics. *Genetics*, **162**, 2025–2035.
- Chen, G.K. *et al.* (2009) Fast and flexible simulation of DNA sequence data. *Genome Res.*, **19**, 136–142.
- Ewing, G. and Hermisson, J. (2010) MSMS: a coalescent simulation program including recombination, demographic structure and selection at a single locus. *Bioinformatics*, **26**, 2064–2065.

- Fitzpatrick, M.J. *et al.* (2007) Maintaining a behaviour polymorphism by frequency-dependent selection on a single gene. *Nature*, **447**, 210–212.
- Guillaume, F. and Rougemont, J. (2006) Nemo: an evolutionary and population genetics programming framework. *Bioinformatics*, **22**, 2556–2557.
- Hudson, R.R. (2002) Generating samples under a Wright–Fisher neutral model of genetic variation. *Bioinformatics*, **18**, 337–338.
- McKeigue, P. *et al.* (2000) Estimation of admixture and detection of linkage in admixed populations by a Bayesian approach: application to African-American populations. *Ann. Hum. Genet.*, **64**, 171–186.
- Peng, B. and Amos, C.I. (2008) Forward-time simulations of non-random mating populations using simuPOP. *Bioinformatics*, **24**, 1408–1409.
- Schumer, M. *et al.* (2015) Reproductive isolation of hybrid populations driven by genetic incompatibilities. *PLoS Genet.*, **11**, e1005041.
- Verzijden, M.N. *et al.* (2012) The impact of learning on sexual selection and speciation. *Trends Ecol. Evol.*, **27**, 511–519.
- Wolf, J.B. *et al.* (1998) Evolutionary consequences of indirect genetic effects. *Trends Ecol. Evol.*, **13**, 64–69.