A community-maintained standard library of population genetic models

Jeffrey R. Adrion[1,⋆], Christopher B. Cole[2,⋆], Noah Dukler[3,⋆], Jared G. Galloway[1,⋆], Ariella L. Gladstein[4,⋆], Graham Gower[5,⋆], Christopher C. Kyriazis[6,⋆], Aaron P. Ragsdale[7,⋆], Georgia Tsambos[8,⋆], Franz Baumdicker[9], Jedidiah Carlson[10], Reed A. Cartwright[11], Arun Durvasula[12], Bernard Y. Kim[13], Patrick McKenzie[14], Philipp W. Messer[15], Ekaterina Noskova[16], Diego Ortega-Del Vecchyo[17], Fernando Racimo[5], Travis J. Struck[18], Simon Gravel[7,†], Ryan N. Gutenkunst[18,†], Kirk E. Lohmeuller[6,†], Peter L. Ralph[1,†], Daniel R. Schrider[4,†], Adam Siepel[3,†], Jerome Kelleher[19,†,ℵ], and Andrew D. Kern[1,†,ℵ]

[1]Department of Biology and Institute of Ecology and Evolution, University of Oregon, [2]Wellcome Trust Centre for Human Genetics, University of Oxford, [3]Simons Center for Quantitative Biology, Cold Spring Harbor Laboratory, [4]Department of Genetics, University of North Carolina at Chapel Hill, [5]Lundbeck GeoGenetics Centre, Globe Institute, University of Copenhagen, [6]Department of Ecology and Evolutionary Biology, University of California, Los Angeles, [7]Department of Human Genetics, McGill University, [8]Melbourne Integrative Genomics, School of Mathematics and Statistics, University of Melbourne, [9]Department of Mathematical Stochastics, University of Freiburg, [10]Department of Genome Sciences, University of Washington, [11]The Biodesign Institute and The School of Life Sciences, Arizona State University, [12]Department of Human Genetics, David Geffen School of Medicine, University of California, Los Angeles, [13]Department of Biology, Stanford University, [14]Department of Ecology, Evolution, and Environmental Biology, Columbia University, [15]Department of Computational Biology, Cornell University, [16]Computer Technologies Laboratory, ITMO University, [17]International Laboratory for Human Genome Research, National Autonomous University of Mexico, [18]Department of Molecular and Cellular Biology, University of Arizona, [19]Big Data Institute, Li Ka Shing Centre for Health Information and Discovery, University of Oxford, ⋆Denotes shared first authorship, listed alphabetically, †Denotes shared senior authorship, listed alphabetically, ℵDenotes corresponding authors, listed alphabetically

Abstract

The explosion in population genomic data demands ever more complex modes of analysis, and increasingly these analyses depend on sophisticated simulations. Recent advances in population genetic simulation have made it possible to simulate large and complex models, but specifying such models for a particular simulation engine remains a difficult and error-prone task. Computational genetics researchers currently re-implement simulation models independently, leading to duplication of effort and the possibility for error. Population genetics, as a field, also lacks standard benchmarks by which new tools for inference might be measured. Here we describe a new resource, `stdpopsim`, that attempts to rectify this situation. `Stdpopsim` is a community-driven open source project, which provides easy access to a standard catalog of published simulation models from a wide range of organisms and supports multiple simulation engine backends. We share some examples demonstrating how `stdpopsim` can be used to systematically compare demographic inference methods, and we encourage an even broader community of developers to contribute to this growing resource.

Keywords: Population genetics, Simulation, Inference, Reproducibility

Introduction

While population genetics has always used statistical methods to make inferences from data, the degree of sophistication of the questions, models, data, and computational approaches used have all increased over the past two decades. Currently there exist myriad computational methods that can infer the histories of populations (Gutenkunst et al., 2009; Li and Durbin, 2011; Excoffier et al., 2013; Schiffels and Durbin, 2014; Terhorst et al., 2017; Ragsdale and Gravel, 2019), the distribution of fitness effects (Boyko et al., 2008; Kim et al., 2017; Tataru et al., 2017; Fortier et al., 2019; Huang and Siepel, 2019; Ortega-Del Vecchyo et al., 2019), recombination rates (Chan et al., 2012; Lin et al., 2013; Adrion et al., 2019; Barroso et al., 2019), and the extent of positive selection in genome sequence data (Eyre-Walker and Keightley, 2009; Alachiotis et al., 2012; DeGiorgio et al., 2016; Kern and Schrider, 2018; Sugden et al., 2018). While these methods have increased our understanding of the impacts of genetic and evolutionary processes, very little has been done to systematically benchmark the quality of inferences gleaned from computational population genetics. As large databases of population genetic variation begin to be used to inform public health procedures, the accuracy and quality of these inferences is becoming ever more important.

Assessing the accuracy of inference methods for population genetics is challenging in large part because the "ground-truth" in question generally comes not from direct empirical observations, as the relevant historical processes can rarely be observed, but instead

from simulations. Population genetic simulations are therefore critically important to the field, yet there has been no systematic attempt to establish community standards or best practices for executing them. Instead, the general modus operandi to date has been for individual groups to validate their own methods using bespoke simulations. Often these studies focus more on showcasing a novel method than on rigorously comparing it with competing methods. Moreover, this situation results in a great deal of duplicated effort, and contributes to decreased reproducibility and transparency across the entire field. It is also a barrier to entry to the field, because new researchers can struggle with the many steps involved in implementing a state-of-the-art population genetics simulation, including identifying appropriate demographic models from the literature, translating them into input for a simulator, obtaining appropriate genetic maps, and choosing appropriate values for key population genetic parameters.

A related issue is that it has been challenging to assess the degree to which modeling assumptions and choices of data summaries can affect population genetic inferences. Yet there are clear examples of different methods yielding fundamentally different conclusions. For example, Markovian coalescent methods applied to human genomes have suggested large ancient ($> 100,000$ years ago) ancestral population sizes and bottlenecks that have not been detected by other methods based on allele frequency spectra (see Beichman et al., 2017). These distinct methods differ in how they model, summarize, and optimize fit to genetic variation data, suggesting that such design choices can greatly affect the performance of the inference. Furthermore, some methods are likely to perform better than others under certain scenarios, but researchers lack principled guidelines for selecting the best method for addressing their particular questions. The need for empirical guidance will only increase as researchers seek to apply population genetic methods to a growing collection of non-model taxa.

For these reasons, we have generated a standardized, community-driven resource for simulating published demographic models from a number of popular study systems. This resource, which we call `stdpopsim`, makes running realistic simulations for population genetic analysis a simple matter of choosing pre-implemented models from a community-maintained catalog. The `stdpopsim` catalog currently contains three organisms: humans, *Drosophila melanogaster*, and *Arabidopsis thaliana*. For each organism, the catalog contains details on the physical organization (e.g., chromosome structure) of its genome, one or more genetic maps, default population-level parameters (mutation rate, generation time) and one or more published demographic histories. Through either a command line interface or a simple Python API, users can specify which organism, genetic map, chromosome, and demographic history they are interested in simulating, and the simulation output from their chosen model is returned. In this way, `stdpopsim` will lower the barrier to high-quality simulation for exploratory analyses, enable rigorous evaluation of population genetic software, and contribute to increased reliability of population genetic

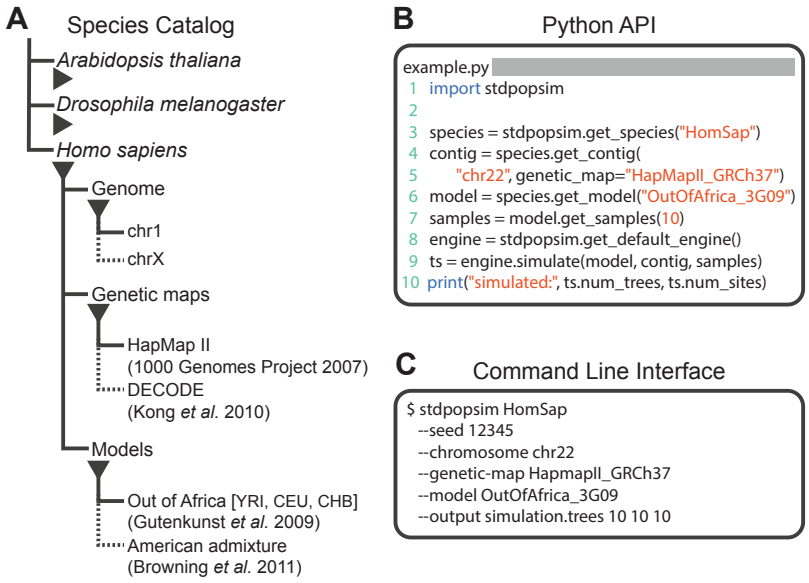

Figure 1: **Structure of `stdpopsim`.** **(A)** The hierarchical organization of the `stdpopsim` catalog contains all model simulation information within individual species (expanded information shown here for *H. sapiens* only). Each species is associated with a representation of the physical genome, and one or more genetic maps and demographic models. Dotted lines indicate that only a subset of these categories is shown. At right we show example code to specify and simulate models using **(B)** the python API or **(C)** the command line interface.

inferences.

The `stdpopsim` library has been developed by the PopSim Consortium using a distributed open source model, with strong procedures in place to continue its growth and maintain its quality. Importantly, we have rigorous quality control methods to ensure implemented models are accurate and have documented methods for others to contribute new modules. We invite new collaborators to join our community. Below we describe the resource and give examples of how it can be used to benchmark demographic inference methods.

Results

The `stdpopsim` library. The first contribution of the PopSim consortium is `stdpopsim`, a community-maintained library of empirical genome data and population genetics simulation models. Figure 1 shows a graphical representation of the structure of `stdpopsim`. The package centers on a catalog of species (Fig. 1A), initially consisting of humans, *D. melanogaster*, and *A. thaliana*. A species definition consists of two key elements. Firstly, the library defines some basic information about each species' genome, including information about chromosome lengths, average mutation rates, and generation times.

We also provide access to detailed empirical information such as genetic maps, which model observed heterogeneity in recombination rate along chromosomes. As such maps are often large, we do not distribute them directly with the software, but make them available for download in a standard format. When a simulation using such a map is requested by the user, `stdpopsim` will transparently download the map data into a local cache, where it can be quickly retrieved for subsequent simulations. In the initial version of `stdpopsim` we support the HapMapII (International HapMap Consortium et al., 2007) and deCODE (Kong et al., 2010) genetic maps for humans; the Salomé et al. (2011) map for *A. thaliana*; and the Comeron et al. (2012) map for *D. melanogaster*. Adding futher maps to the library is trivial. The second key element of a species description within `stdpopsim` is a set of carefully curated population genetic model descriptions from the literature, which allow simulation under specific historical scenarios that have been fit to present-day patterns of genetic variation. (See the Methods for a description of the community development and quality-control process for these models.)

Given the genome data and simulation model descriptions defined within the library, it is then straightforward to run accurate, standardized simulations across a range of organisms. `Stdpopsim` has a Python API and a user-friendly command line interface, allowing users with minimal experience direct access to state-of-the-art simulations. Simulations are output in the "tree sequence" format (Kelleher et al., 2016, 2018, 2019), which contains complete genealogical information about the simulated samples, is extremely compact, and can be processed efficiently using the `tskit` library (Kelleher et al., 2016, 2018). Currently, `stdpopsim` uses the `msprime` coalescent simulator (Kelleher et al., 2016) as the default simulation engine. We have implemented `SLiM` (Haller et al., 2019; Haller and Messer, 2019) as an alternative backend, to allow simulation of processes that cannot be modeled under the coalescent.

The `stdpopsim` command line interface, by default, outputs citation information for the models, genetic maps and simulation engines used in any particular run. We hope that this will encourage users to appropriately acknowledge the resources used in published work, and encourage authors publishing demographic models to contribute to our ongoing community-driven development process. Together with the `stdpopsim` version number and the long-term stable identifiers for population models and genetic maps, this citation information will result in well-documented and reproducible simulation workflows. The individual tree sequence files produced by `stdpopsim` also contain complete provenance information including the command line arguments, operating system environment and versions of key libraries used.

Model ID	Citation	CPU(s)	RAM(MB)	File(MB)
HomSap (*Homo sapiens*)				
Africa_1T12	Tennessen et al. (2012)	10.2	191.3	23.3
Zigzag_1S14	Schiffels and Durbin (2014)	3.4	103.5	7.9
OutOfAfrica_3G09	Gutenkunst et al. (2009)	11.4	181.6	21.4
OutOfAfrica_2T12	Tennessen et al. (2012)	12.4	200.4	24.7
AncientEurasia_9K19	Kamm et al. (2019)	64.8	303.1	41.2
AmericanAdmixture_4B11	Browning et al. (2018)	10.6	185.0	22.3
OutOfAfricaArchaicAdmixture_5R19	Ragsdale and Gravel (2019)	9.1	182.1	21.7
DroMel (*Drosophila melanogaster*)				
OutOfAfrica_2L06	Li and Stephan (2006)	0.6	66.7	1.6
African3Epoch_1S16	Sheehan and Song (2016)	0.5	58.8	0.2
AraTha (*Arabidopsis thaliana*)				
African2Epoch_1H18	Huber et al. (2018)	379.5	358.2	50.7
African3Epoch_1H18	Huber et al. (2018)	187.1	399.5	58.0
SouthMiddleAtlas_1D17	Durvasula et al. (2017)	141.1	315.8	43.1

Table 1: Initial set of demographic models in the Catalog and simple benchmarks. For each model we report the CPU time, maximum memory usage and the size of the output `tskit` file. In each case we simulate 100 samples drawn from the first population, for the shortest chromosome of that species and a constant chromosome-specific recombination rate. The times reported are for a single run on an Intel i5-7600 CPU. Computing resources required will vary widely depending on sample sizes, chromosome length, recombination rates and other factors.

The Species Catalog

The central feature of `stdpopsim` is the species catalog, a systematic organization of the key quantitative data needed to simulate a given species. These include a description of the assembly, information about mutation rate, recombination rate(s), and generation time in addition to a series of demographic models that are specific to that organism.

The current contents of the `stdpopsim` catalog are shown in Table 1. These range from simple, single population histories (e.g., Sheehan and Song, 2016), to complex models which include population splitting, migration, and archaic admixture (e.g., Ragsdale and Gravel, 2019). In addition to those models shown, at time or writing the PopSim Consortium has models in development for *Pongo abelii* and *Escherichia coli*.

Currently, *Homo sapiens* has the largest number of population models in `stdpopsim` (see Table 1). These models include: a simplified version of the Tennessen et al. (2012) model with only the African population specified (expansion from the ancestral population and recent growth; Africa_1T12), the three-population model of Gutenkunst et al. (2009) which specifies the out-of-Africa bottleneck as well as the subsequent divergence of the European and Asian populations (OutOfAfrica_3G09), the Tennessen et al. (2012) two-population variant of the Gutenkunst et al. model which does not include

Asian populations, but more explicitly models recent rapid human population growth (OutOfAfrica_2T12), the Browning et al. (2018) admixture model for American populations which specifies ancestral African, European, and Asian population components (AmericanAdmixture_4B11), a three-population out-of-Africa model from Ragsdale and Gravel (2019) which includes archaic admixture (OutOfAfricaArchaicAdmixture_5R19), a complex model of ancient Eurasian admixture from Kamm et al. (2019) (AncientEurasia_9K19), and a synthetic model of oscillating population size from Schiffels and Durbin (2014) (Zigzag_1S14). Together these models contain features believed to have widespread impacts in real data (e.g., bottlenecks, population growth, admixture) and are therefore highly pertinent in the context of method development.

Beyond humans we have have implemented two demographic histories for *D. melanogaster*, three from *A. thaliana*, and are currently developing models for *P. abelii* and *E. coli*. For *D. melanogaster* we have implemented the three-epoch model estimated by Sheehan and Song (2016) from an African sample (African3Epoch_1S16), as well as the out-of-Africa divergence and associated bottleneck model of Li and Stephan (2006), which jointly models African and European populations (OutOfAfrica_2L06). For *A. thaliana*, we implemented the model in Durvasula et al. (2017) inferred using `MSMC`. This model includes a continuous change in population size over time, rather than pre-specified epochs of different population sizes (SouthMiddleAtlas_1D17). We have also implemented a two-epoch and a three-epoch model estimated from African samples of *A. thaliana* in Huber et al. (2018) (African2Epoch_1H18 and African3Epoch_1H18). In addition to organism-specific models, `stdpopsim` also includes a generic piecewise constant size model and isolation with migration (IM) model which can be used with any genome and genetic map.

To guarantee reproducibility, we have standardized naming conventions for species, genetic maps, and demographic models that will enable long term stability of unique identifiers used throughout `stdpopsim`, as described in our documentation (`https://stdpopsim.readthedocs.io/en/latest/development.html#naming-conventions`).

Use case: comparing methods of demographic inference

As an example of the utility of `stdpopsim`, we demonstrate how it can be used to easily and fairly compare popular demographic inference methods. Although we present comparison of results from several methods, our aim at this stage is not to provide an exhaustive evaluation or ranking of these methods. Our hope is instead that future work built upon this resource will enable more detailed exploration of the strengths and weaknesses of the numerous inference methods that are available to the population genetics community (see Discussion).

We start by comparing popular methods for estimating population size histories ($N(t)$) of single populations and subsequently show simple examples of multi-population infer-

ence. To reproducibly evaluate and compare the performance of inference methods, we developed workflows using `snakemake` (Köster and Rahmann, 2012) that are available from `https://github.com/popsim-consortium/analysis`, that allow efficient computing in multicore or cluster environments.

For single-population population size histories, we compared `MSMC` (Schiffels and Durbin, 2014), `smc++` (Terhorst et al., 2017), and `stairway plot` (Liu and Fu, 2015) on simulated genomes sampled from a single population, in a number of the demographic models described above. Our workflow generates R replicates of C chromosomes, producing n samples in each of a total of $R \times C$ simulations for each demographic model. After simulation, the workflow prepares input files for each inference method by grouping all chromosomes, for each sample, into a single file. For each of the R simulation replicates, this step results in an input file for each of the respective inference methods and derived from the same simulated tree sequences. Each of the inference programs are then run in parallel, followed by plotting of $N(t)$ estimates from each program.

Figure 2 presents the results from simulations under OutOfAfricaArchaicAdmixture_5R19, a model of human migration out of Africa that includes archaic admixture (Ragsdale and Gravel, 2019), along with an empirical genetic map. In each column of this figure we show $N(t)$ inferred from samples taken from each of the three extant populations in the model. In each row we show comparisons among the methods (including two sample sizes for `MSMC`). Blue lines show estimates from each of three replicate whole genome simulations. There is no single "true" reference for effective population size because of model misspecification—the inference methods are fitting a single population model to data simulated from multiple populations. However, many methods work by matching coalescence time distributions, and a single-population model with varying population size can match any coalescence time distribution (in which case coalescence rate is the inverse of the effective size). For this reason, we used as our "ground-truth" (solid black lines) not historical census sizes, but rather inverse coalescence rates calculated analytically in `msprime` (see Appendix). While there is variation in accuracy among methods, populations, and individual replicates, the methods are generally accurate for this model of human history.

`Stdpopsim` allows us to readily compare relative performance on this benchmark to that based on a different model of human history. In Figure S1 we show estimates of $N(t)$ from simulations using the same physical and genetic maps, but from the OutOfAfrica_3G09 demographic model that does not include archaic admixture. Again we see that each of the methods is capturing relevant parts of the population history, although the accuracy varies across time. In comparing inferences between the models it is interesting to note that $N(t)$ estimates for the CHB and CEU simulated populations are generally better across methods than estimates from the YRI simulated population.

We can also see how well methods might do at recovering the population history

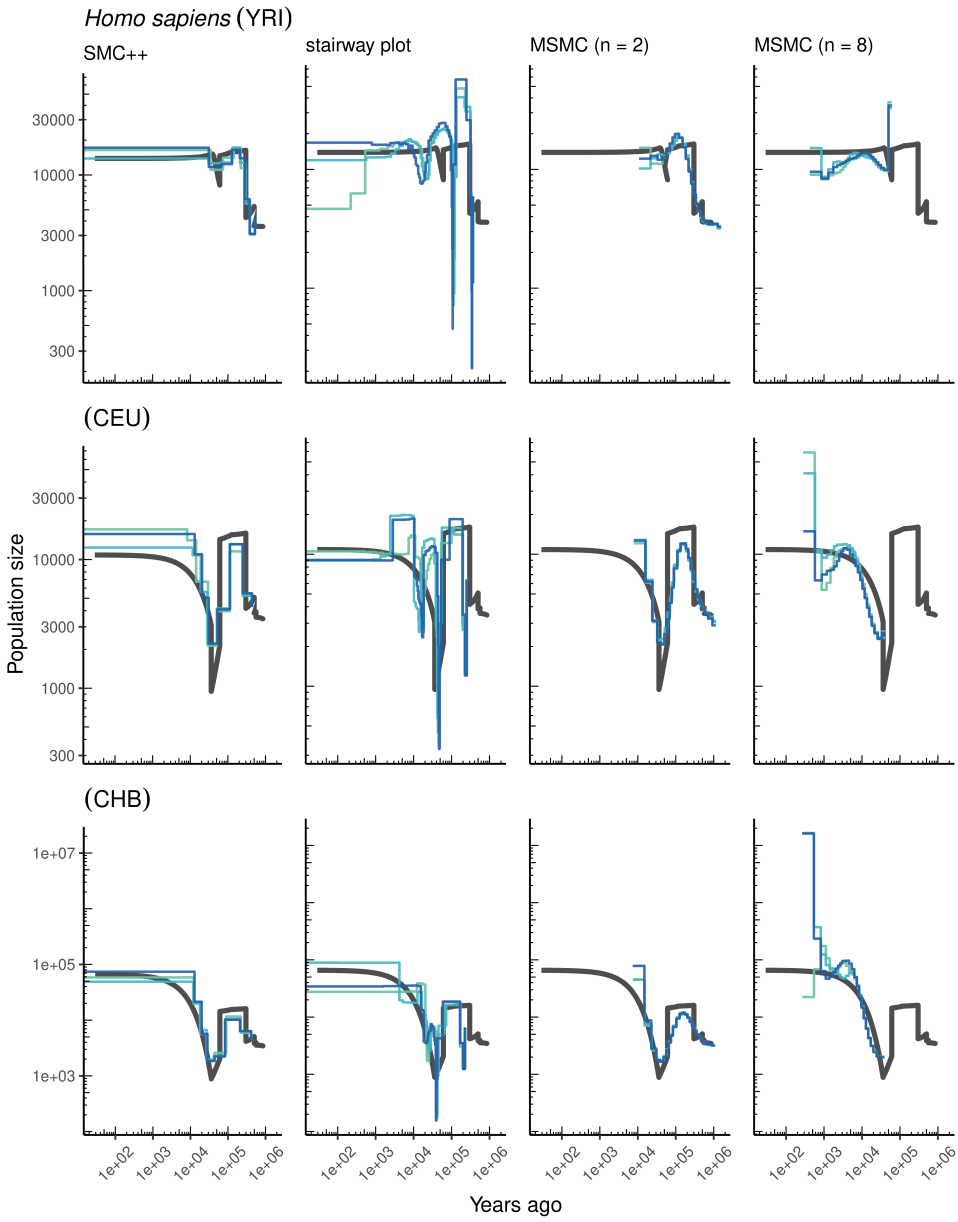

Figure 2: **Comparing estimates of $N(t)$ in humans**. Here we show estimates of population size over time ($N(t)$) inferred using 4 different methods: `smc++`, `stairway plot`, and `MSMC` with $n = 2$ and $n = 8$. Data were generated by simulating replicate human genomes under the OutOfAfricaArchaicAdmixture_5R19 model and using the HapMapII_GRCh37 genetic map (International HapMap Consortium et al., 2007). From top to bottom we show estimates for each of the three populations in the model (YRI, CEU, and CHB). In shades of blue we show the estimated $N(t)$ trajectories for each replicate. As a proxy for the truth, in black we show inverse coalescence rates as calculated from the true demographic model (see text).

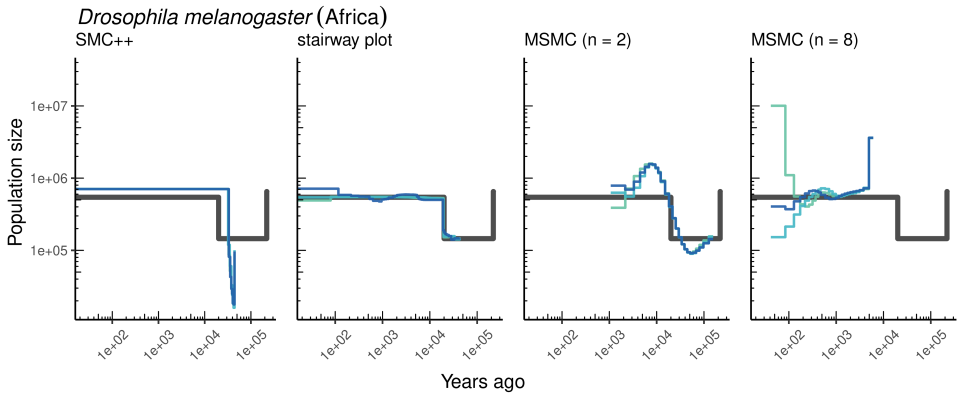

Figure 3: **Comparing estimates of $N(t)$ in *Drosophila*.** Population size over time ($N(t)$) estimated from an African population sample. Data were generated by simulating replicate *D. melanogaster* genomes under the African3Epoch_1S16 model with the genetic map of Comeron et al. (2012). In shades of blue we show the estimated $N(t)$ trajectories for each replicate. As a proxy for the truth, in black we show inverse coalescence rates as calculated from the true demographic model (see text).

of a constant-sized population, with human genome architecture and genetic map. We show results of such an experiment in Figure S2. All methods recover population size within a factor of two of the truth, however SMC-based methods, perhaps due to their regularization, tend to infer sinusoidal patterns of population size even though no such change is present.

As most method development for population genetics has been focused on human data, it is of consequence to ask how such methods might perform in non-human genomes. Figure 3 shows parameter estimates from the African3Epoch_1S16 model, originally estimated from an African sample of *D. melanogaster* (Sheehan and Song, 2016), and Figure S3 shows estimates from simulations of *A. thaliana* under the African2Epoch_1H18 model originally inferred by Huber et al. (2018). In both cases, as with humans, we use `stdpopsim` to simulate replicate genomes using an empirically derived genetic map, and try to infer back parameters of the simulation model. Accuracy is mixed among methods in this setting and generally worse than what we observe for simulations of the human genome.

Multi-population demographic models. As `stdpopsim` implements multi-population demographic models, we also explored parameter estimation of population divergence parameters. In particular, we simulated data under multi-population models for humans and *D. melanogaster* and then inferred parameters using $\partial a \partial i$, `fastsimcoal2`, and `smc++`. For simplicity, we conducted inference in $\partial a \partial i$ and `fastsimcoal2` by fitting an isolation with migration (IM) model with constant population sizes and bi-directional migration (Hey and Nielsen, 2004). Our motivation for using an IM model was to mimic the ap-

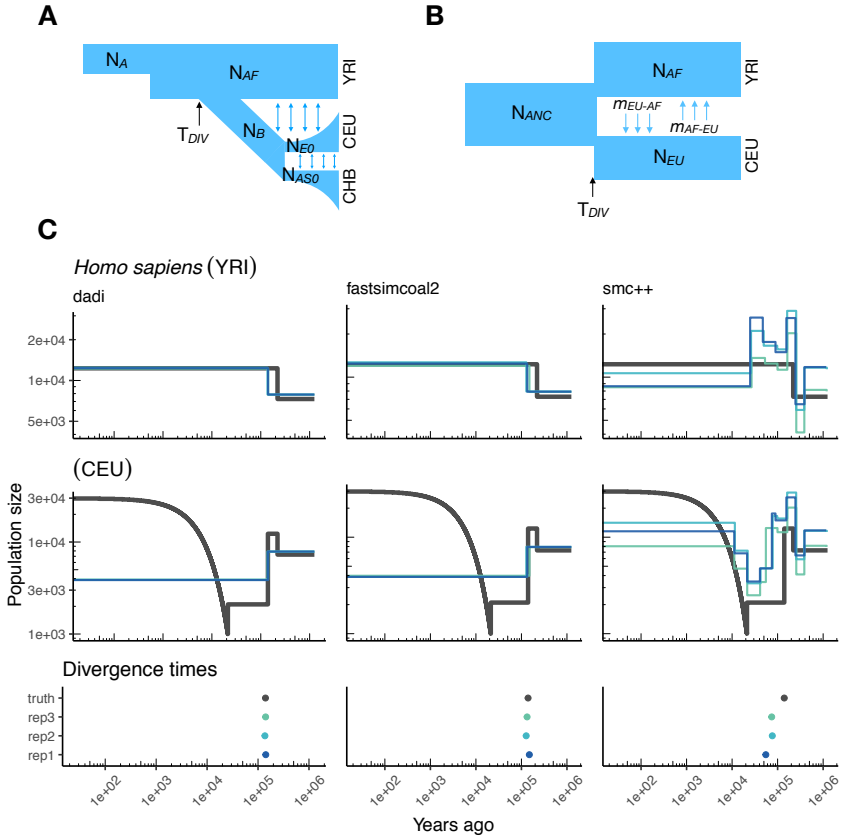

Figure 4: **Parameters estimated using a multi-population human model**. Here we show estimates of $N(t)$ inferred using $\partial a \partial i$, `fastsimcoal2`, or `smc++`. **(A)** Data were generated by simulating replicate human genomes under the OutOfAfrica_3G09 model and using the HapMapII_GRCh37 genetic map inferred in International HapMap Consortium et al. (2007). **(B)** For $\partial a \partial i$ and `fastsimcoal2` we show parameters inferred by fitting the depicted IM model, which includes population sizes, migration rates, and a split time between CEU and YRI samples. **(C)** Population size estimates for each population (rows) from $\partial a \partial i$, `fastsimcoal2`, and `smc++` (columns). In shades of blue we show $N(t)$ trajectories estimated from each simulation, and in black census sizes for the respective population. The population split date, T_{DIV}, is shown at the bottom, with a common X-axis to the population size panels.

proach often used on empirical datasets, where a relatively simple model is fit that may not reflect the true underlying demography. For human models with more than two populations (e.g., Gutenkunst et al. (2009)) this means that we are inferring parameters for a model that does not match the model from which the data were generated (Figures 4A and B). However, because the inferred models here better match the simulated models than in the single population case, here we compare our inferred population sizes directly to the census size of the simulated models (black line in Figure 4C).

In Figure 4C we show estimates of population sizes and divergence time, for each of the inference methods, using samples drawn from African and European populations simulated under the OutOfAfrica_3G09 model. Our results highlight many of the strengths and weaknesses of the different types of methods we used. For instance, the SFS-based approaches where we fit simple IM models do not capture recent exponential growth in the CEU population, but do consistently recover the simulated YRI population size history. Moreover, these approaches allow for estimating migration rates (Figure S7), also leading to more accurate inference of divergence times. However, these migration rate estimates are somewhat biased likely due to model misspecification (Figure S7). By contrast, `smc++` is much better at capturing the recent exponential growth in the CEU population, though the inferred population sizes are generally noisier. In addition, the assumption of no migration by `smc++` leads to divergence time estimates that are consistently underestimated (Figure 4C).

Again, we can compare between species and look at the performance of these methods in on a two-population model of *D. melanogaster*. Figure S4 shows parameter estimates for simulations drawn from the OutOfAfrica_2L06 model, which includes an ancestral population in Africa, then a population expansion with a population split and bottleneck into a European population with no post-divergence migration. Here again, we find that $\partial a \partial i$ and `fastsimcoal2` infer more consistent histories, but ignore the brief population bottleneck in Europe. In addition, $\partial a \partial i$ and `fastsimcoal2` both do reasonably well at correctly inferring the absence of migration (Figure S6). By contrast, the inferred demographic parameters from `smc++` are more noisy, though in some cases better capture the short bottleneck in the Europe population.

Although these results do not represent an exhaustive benchmarking, we have highlighted some of the strengths and weaknesses of these methods. Future work should build on these results and undertake more in-depth comparisons under a wider range of simulated demographic models.

Discussion

Here we have described the first major product from the PopSim Consortium: the `stdpopsim` library. We have founded the Consortium with a number of specific goals in

mind: standardization of simulation within the population genetics community, increased reproducibility and ease of use of complex simulations, community-based development and decision making guiding best practices in population genetics, and benchmarking of inference methods.

The `stdpopsim` library allows for rigorous standardization of complex population genetic simulations. Population genetics, as a field, has yet to coalesce around a set of standards for the crucial task of method evaluation, which in our discipline hinges on simulation. In contrast, other fields such as structural biology (Moult et al., 1995) and machine learning (Russakovsky et al., 2015) have a long track record of standardized method testing. We hope that our efforts represent the beginning of what will prove to be an equally longstanding and valuable tradition in population genetics.

We have illustrated in this paper how `stdpopsim` can be used for direct comparisons of inferential methods on a common set of simulations. Our benchmarking comparisons have been limited, but nevertheless reveal some informative features. For example, at the task of estimating $N(t)$ trajectories for simulated human populations, we find that the sequence-based methods (`MSMC` and `smc++`) perform somewhat better overall—at least for mid-range values of t—than the site frequency spectrum-based method (`stairway plot`) (Figures 2 and S1), which tends to over-estimate the sizes of oscillations. By contrast, `stairway plot` outperforms the sequence-based methods on simulations of *D. melanogaster* or *A. thaliana* populations, in which linkage disequilibrium is reduced (Figures 3 & S3). In simulations of two human populations (Figure 4), most methods do reasonably well at reconstructing the simulated YRI history, but struggle with the more complex simulated CEU history, in large part because of the restriction of constant population sizes. An exception is `smc++`, which does not have the same restrictions on its inferred history, and as a result does somewhat better with the CEU history but tends to overfit the YRI history. The results for the two-population *D. melanogaster* model (Figure S4) are generally similar.

Altogether, these preliminary experiments highlight the utility of `stdpopsim` for comparing a variety of inference methods on the same footing, under a variety of different demographic models. In addition, the ability of `stdpopsim` to generate data with and without significant features, such as a genetic map or population size change (e.g., Figure S2), allows investigation of the failure modes of popular methods. Moreover the comparison of methods across the various genome organizations, genetic maps, and demographic histories of different organisms, provides valuable information about how methods might perform on non-human systems. Finally, comparison of results across methods or simulation runs provides an estimate of inference uncertainty, analogous to parametric bootstrapping, especially since different methods are likely vulnerable to model misspecification in different ways.

`Stdpopsim` is intended to be a fully open, community-developed project. Our imple-

mentations of genome representations and genetic maps for the some of the most common study systems in computational genetics—humans, *Drosophila*, and *Arabidopsis* (among others)—are only intended to be a starting point for future development. In addition to other taxa, we plan to incorporate other common biological processes such as selection, gene conversion, and mutational heterogeneity. Researchers are invited to contribute to the resource by adding their organisms and models of choice. The `stdpopsim` resource is accompanied by clearly documented standard operating procedures that are intended to minimize barriers to entry for new developers. In this way, we expect the resource to expand and adapt to meet the evolving needs of the population genomics community.

Methods

Model quality control

As a consortium we have agreed to a standardized procedure for model inclusion into `stdpopsim` that allows for rigorous quality control. Imagine Developer A wants to introduce a new model into `stdpopsim`. Developer A implements the demographic model for the relevant organism along with clear documentation of the model parameters and populations. This model is submitted as a "pull request", where it is evaluated by a reviewer and then included as 'preliminary', but is not linked to the online documentation nor the command line interface. Developer A submits a quality control (QC) issue, after which a second developer, Developer B, then independently reimplements the model from the relevant primary sources and adds an automatic unit test for equality between the QC implementation and the preliminary production model. If the two implementations are equivalent, the original model is included in `stdpopsim`. If not, we move to an arbitration process whereby A and B first try to work out the details of what went wrong. If that fails, the original authors of the published model must be contacted to resolve ambiguities. Further details of our QC process can be found in our `https://stdpopsim.readthedocs.io/en/latest/development.html#`developer documentation.

Workflow for analysis of simulated data

To demonstrate the utility of `stdpopsim` we created `Snakemake` workflows (Köster and Rahmann, 2012) that perform demographic inference on tree sequence output from our package using a few common software packages. Our choice of `Snakemake` allows complete reproducibility of the analyses shown, and all code is available from `https://github.com/popsim-consortium/analysis`.

We performed two types of demographic inference. Our first task was to infer ef-

fective population size over time ($N(t)$). This was done using three software packages: `stairway plot`, which uses site frequency spectrum information only (Liu and Fu, 2015); `MSMC` (Schiffels and Durbin, 2014), which is based on the sequentially Markovian coalescent (SMC), run with two different sample sizes ($n = 2, 8$); and `smc++` (Terhorst et al., 2017), which combines information from the site frequency spectrum with recombination information as in SMC-based methods. No attempt was made at trying to optimize the analysis from any particular software package, as our goal was not to benchmark performance of methods but instead show how such benchmarking could be easily done using the `stdpopsim` resource. In this spirit we ran each software package as near to default parameters as possible. For `stairway plot` we set the parameters "numRuns=1" and "dimFactor=5000". For `smc++` we used the "estimate" run mode to infer $N(t)$ with all other parameters set to their default values. For `MSMC` we used the "–fixedRecombination" option and used the default number of iterations.

For the single-population task we ran human (HomSap) simulations using a variety of models (see Table 1): OutOfAfricaArchaicAdmixture_5R19, OutOfAfrica_3G09, a constant-sized generic model, and a two-epoch generic model where the population size instantaneously decreased from $N = 10^4$ to $N = 10^3$ five hundred generations before the present. Each HomSap simulation was run using the HapmapII_GRCh37 genetic map. For *D. melanogaster* we estimated $N(t)$ from an African sample simulated under the DroMel, African3Epoch_1S16 model using the Comeron2012_dm6 map. Finally we ran simulations of *A. thaliana* genomes using the AraTha African2Epoch_1H18 model under the Salome2012_TAIR7 map. For each model, three replicate whole genomes were simulated and the population size estimated from those data. In all cases we set the sample size of the focal population to $N = 50$ chromosomes.

Following simulation, low-recombination portions of chromosomes were masked from the analysis in a manner that reflects the "accessible" subset of sites used in empirical population genomic studies (e.g., Danecek et al., 2011; Langley et al., 2012). Specifically we masked all regions of 1 cM or greater in the lowest 5th percentile of the empirical distribution of recombination, regions which are nearly uniformly absent for empirical analysis.

Our second task was to explore inference with two-population models using some of the multi-population demographic models implemented in `stdpopsim`. For HomSap we used the OutOfAfrica_3G09 model with the HapmapII_GRCh37 genetic map, and for DroMel we used the OutOfAfrica_2L06 model with the Comeron2012_dm6 map. The HomSap model is a three population model (Africa, Europe, and Asia) including post-divergence migration and exponential growth (Figure 4C), whereas the DroMel model is a two population model (Africa and Europe) with no post-divergence migration and constant population sizes (Figure S4).

To conduct inference on these models, we applied three commonly used methods:

$\partial a \partial i$ (Gutenkunst et al., 2009), `fastsimcoal2` (Excoffier et al., 2013), and `smc++` (Terhorst et al., 2017). As above, these methods were used generally with default settings and we did not attempt to optimize their performance or fit parameter-rich demographic models.

For both $\partial a \partial i$ and `fastsimcoal2`, we fit a two population isolation-with-migration (IM) model with constant population sizes. This IM model contains six parameters: the ancestral population size, the sizes of each population 1 after the split, the divergence time, and two migration rate parameters. Importantly, this meant that for both species, the fitted model did not match the simulated model (Figures 4 and S4). In the HomSap case, we therefore performed inference solely on the Africa and Europe populations, meaning that the Asia population functioned as a "ghost" population that was ignored by our inference. Our motivation for fitting this simple IM model was to mimic the typical approach of two population inference on empirical data, where the user is not aware of the 'true' underlying demography and the inference model is often misspecified. To ground-truth our inference approach, we also conducted inference on a generic IM model that was identical to the model used for inference S5.

From HomSap simulations we took 20 whole genome samples each from the Europe and Africa populations from each replicate. Runtimes of DroMel simulations were prohibitively slow when simulating whole genomes with the Comeron2012_dm6 map due to large effective population sizes leading to high effective recombination rates. For this reason, we present only data from 50 samples of a 3 MB region of chromosome 2R from simulations under OutOfAfrica_2L06. For the generic IM simulations, we used the HomSap genome along with the HapmapII_GRCh37 genetic map and sampled 20 individuals from each population.

Following simulation, we output tree sequences and masked low-recombination regions using the same approach described for the single population workflow above. We converted tree sequences into a two-dimensional site frequency spectrum for all chromosomes in the appropriate format for $\partial a \partial i$ and `fastsimcoal2`. For each simulation replicate, we performed 10 runs of $\partial a \partial i$ and `fastsimcoal2` and checked for convergence. Detailed settings for $\partial a \partial i$ and `fastsimcoal2` can be found in the Snakefile on our git repository (`https://github.com/popsim-consortium/analysis`). Estimates from the highest log-likelihood (out of 10 runs) for each simulation replicate are shown in Figures 4C and S4C.

For `smc++`, we converted the tree sequences into VCF format and performed inference with default settings. Importantly, `smc++` assumes no migration post-divergence, deviating from the simulated model. However, because `smc++` allows for continuous population size changes, it is better equipped to capture many of the more complex aspects of the simulated demographic models (e.g., exponential growth).

To visualize our results, we plotted the inferred population size trajectories for each

simulation replicate alongside census population sizes (Figures 4C and S4C). Here, unlike the single-population workflow, we compare to census size rather than the inverse coalescence rate as the 'true' population size.

Resource availability

The version 0.1 release of `stdpopsim` is available for download on Github: `https://github.com/popsim-consortium/stdpopsim/releases`. Documentation for the project can be found here: `https://stdpopsim.readthedocs.io/en/latest/`.

Acknowledgments

We thank the Probabilistic Modeling in Genomics conference organizers with making this collaboration possible, and the Simons Center for Quantitative Biology at Cold Spring Harbor Laboratory for sponsoring the first workshop. Early on in the project we were encouraged by many people including Patrick Phillips, Richard Durbin, Dmitri Petrov, and Sohini Ramachandran. CCK and KEL were funded under NIH Award R35GM119856. JRA and ADK were funded under NIH Award R01GM117241. TJS and RNG were funded under NIH Award R01GM127348. ALG and DRS were funded under NIH award R00HG008696. ND and AS were supported in part by NIH Awards R01HG010346 and R35GM127070. FR and GG were supported by a Villum Young Investigator award (project no. 00025300). DODV is funded by a UC MEXUS-CONACYT Collaborative Grant and a DGAPA-PAPIIT grant (PAPIIT-IA200620). JK is supported by the Robertson Foundation.

References

Jeffrey R. Adrion, Jared G. Galloway, and Andrew D. Kern. Inferring the landscape of recombination using recurrent neural networks. *bioRxiv*, 2019. doi: 10.1101/662247. URL `https://www.biorxiv.org/content/early/2019/06/06/662247`.

Nikolaos Alachiotis, Alexandros Stamatakis, and Pavlos Pavlidis. Omegaplus: a scalable tool for rapid detection of selective sweeps in whole-genome datasets. *Bioinformatics*, 28(17):2274–2275, 2012.

Gustavo V. Barroso, Nataša Puzović, and Julien Y. Dutheil. Inference of recombination maps from a single pair of genomes and its application to ancient samples. *PLOS Genetics*, 15(11):e1008449, 2019. doi: 10.1371/JOURNAL.PGEN.1008449. URL `https://dx.plos.org/10.1371/journal.pgen.1008449`.

Annabel C Beichman, Tanya N Phung, and Kirk E Lohmueller. Comparison of single genome and allele frequency data reveals discordant demographic histories. *G3: Genes, Genomes, Genetics*, 7(11):3605–3620, 2017.

Adam R. Boyko, Scott H. Williamson, Amit R. Indap, Jeremiah D. Degenhardt, Ryan D. Hernandez, Kirk E. Lohmueller, Mark D. Adams, Steffen Schmidt, John J. Sninsky, Shamil R. Sunyaev, Thomas J. White, Rasmus Nielsen, Andrew G. Clark, and Carlos D. Bustamante. Assessing the evolutionary impact of amino acid mutations in the human genome. *PLoS Genetics*, 4(5):e1000083, may 2008. ISSN 15537390. doi: DOI 10.1371/journal.pgen.1000083.

Sharon R Browning, Brian L Browning, Martha L Daviglus, Ramon A Durazo-Arvizu, Neil Schneiderman, Robert C Kaplan, and Cathy C Laurie. Ancestry-specific recent effective population size in the americas. *PLoS genetics*, 14(5):e1007385, 2018.

Andrew H Chan, Paul A Jenkins, and Yun S Song. Genome-wide fine-scale recombination rate variation in drosophila melanogaster. *PLoS genetics*, 8(12):e1003090, 2012.

Josep M Comeron, Ramesh Ratnappan, and Samuel Bailin. The many landscapes of recombination in *drosophila* melanogaster. *PLoS genetics*, 8(10):e1002905, 2012.

Petr Danecek, Adam Auton, Goncalo Abecasis, Cornelis A Albers, Eric Banks, Mark A DePristo, Robert E Handsaker, Gerton Lunter, Gabor T Marth, Stephen T Sherry, et al. 1000 genomes project analysis group (2011). *The variant call format and VCFtools. Bioinformatics*, 27(15):2156–2158, 2011.

Michael DeGiorgio, Christian D Huber, Melissa J Hubisz, Ines Hellmann, and Rasmus Nielsen. Sweepfinder2: increased sensitivity, robustness and flexibility. *Bioinformatics*, 32(12):1895–1897, 2016.

Arun Durvasula, Andrea Fulgione, Rafal M Gutaker, Selen Irez Alacakaptan, Pádraic J Flood, Célia Neto, Takashi Tsuchimatsu, Hernán A Burbano, F Xavier Picó, Carlos Alonso-Blanco, et al. African genomes illuminate the early history and transition to selfing in arabidopsis thaliana. *Proceedings of the National Academy of Sciences*, 114 (20):5213–5218, 2017.

Laurent Excoffier, Isabelle Dupanloup, Emilia Huerta-Sánchez, Vitor C Sousa, and Matthieu Foll. Robust demographic inference from genomic and SNP data. *PLoS genetics*, 9:e1003905, 2013.

Adam Eyre-Walker and Peter D Keightley. Estimating the rate of adaptive molecular evolution in the presence of slightly deleterious mutations and population size change. *Molecular biology and evolution*, 26(9):2097–2108, 2009.

Alyssa Lyn Fortier, Alec J. Coffman, Travis J. Struck, Megan N. Irby, Jose E. L. Burguete, Aaron P. Ragsdale, and Ryan N. Gutenkunst. Dfenitely different: Genome-wide characterization of differences in mutation fitness effects between populations. *bioRxiv*, 2019. doi: 10.1101/703918. URL https://www.biorxiv.org/content/early/2019/07/16/703918.

Ryan N Gutenkunst, Ryan D Hernandez, Scott H Williamson, and Carlos D Bustamante. Inferring the joint demographic history of multiple populations from multidimensional snp frequency data. *PLoS genetics*, 5(10):e1000695, 2009.

Benjamin C Haller and Philipp W Messer. SLiM 3: forward genetic simulations beyond the Wright–Fisher model. *Molecular biology and evolution*, 36(3):632–637, 2019.

Benjamin C Haller, Jared Galloway, Jerome Kelleher, Philipp W Messer, and Peter L Ralph. Tree-sequence recording in slim opens new horizons for forward-time simulation of whole genomes. *Molecular ecology resources*, 19(2):552–566, 2019.

Jody Hey and Rasmus Nielsen. Multilocus methods for estimating population sizes, migration rates and divergence time, with applications to the divergence of drosophila pseudoobscura and d. persimilis. *Genetics*, 167(2):747–760, 2004. ISSN 0016-6731. doi: 10.1534/genetics.103.024182. URL https://www.genetics.org/content/167/2/747.

Yi-Fei Huang and Adam Siepel. Estimation of allele-specific fitness effects across human protein-coding sequences and implications for disease. *Genome Research*, page gr.245522.118, 2019. ISSN 1088-9051. doi: 10.1101/gr.245522.118.

Christian D. Huber, Arun Durvasula, Angela M. Hancock, and Kirk E. Lohmueller. Gene expression drives the evolution of dominance. *Nature Communications*, 9(1), Jul 2018. ISSN 2041-1723. doi: 10.1038/s41467-018-05281-7. URL http://dx.doi.org/10.1038/s41467-018-05281-7.

International HapMap Consortium et al. A second generation human haplotype map of over 3.1 million snps. *Nature*, 449(7164):851, 2007.

Jack Kamm, Jonathan Terhorst, Richard Durbin, and Yun S. Song. Efficiently inferring the demographic history of many populations with allele count data. *Journal of the American Statistical Association*, page 1–16, Jul 2019. ISSN 1537-274X. doi: 10.1080/01621459.2019.1635482. URL http://dx.doi.org/10.1080/01621459.2019.1635482.

Jerome Kelleher, Alison M Etheridge, and Gilean McVean. Efficient coalescent simulation and genealogical analysis for large sample sizes. *PLoS computational biology*, 12(5):e1004842, 2016.

Jerome Kelleher, Kevin R. Thornton, Jaime Ashander, and Peter L. Ralph. Efficient pedigree recording for fast population genetics simulation. *PLOS Computational Biology*, 14(11):1–21, 11 2018. doi: 10.1371/journal.pcbi.1006581. URL https://doi.org/10.1371/journal.pcbi.1006581.

Jerome Kelleher, Yan Wong, Anthony W. Wohns, Chaimaa Fadil, Patrick K. Albers, and Gil McVean. Inferring whole-genome histories in large population datasets. *Nature Genetics*, 51(9):1330–1338, 2019. ISSN 15461718. doi: 10.1038/s41588-019-0483-y. URL https://doi.org/10.1038/s41588-019-0483-y.

Andrew D Kern and Daniel R Schrider. diplos/hic: an updated approach to classifying selective sweeps. *G3: Genes, Genomes, Genetics*, 8(6):1959–1970, 2018.

Bernard Y Kim, Christian D Huber, and Kirk E Lohmueller. Inference of the distribution of selection coefficients for new nonsynonymous mutations using large samples. *Genetics*, 206(1):345–361, 2017.

Augustine Kong, Gudmar Thorleifsson, Daniel F Gudbjartsson, Gisli Masson, Asgeir Sigurdsson, Aslaug Jonasdottir, G Bragi Walters, Adalbjorg Jonasdottir, Arnaldur Gylfason, Kari Th Kristinsson, et al. Fine-scale recombination rate differences between sexes, populations and individuals. *Nature*, 467(7319):1099, 2010.

Johannes Köster and Sven Rahmann. Snakemake—a scalable bioinformatics workflow engine. *Bioinformatics*, 28(19):2520–2522, 2012.

Charles H Langley, Kristian Stevens, Charis Cardeno, Yuh Chwen G Lee, Daniel R Schrider, John E Pool, Sasha A Langley, Charlyn Suarez, Russell B Corbett-Detig, Bryan Kolaczkowski, et al. Genomic variation in natural populations of drosophila melanogaster. *Genetics*, 192(2):533–598, 2012.

Haipeng Li and Wolfgang Stephan. Inferring the demographic history and rate of adaptive substitution in drosophila. *PLoS genetics*, 2(10):e166, 2006.

Heng Li and Richard Durbin. Inference of human population history from individual whole-genome sequences. *Nature*, 475(7357):493, 2011.

Kao Lin, Andreas Futschik, and Haipeng Li. A fast estimate for the population recombination rate based on regression. *Genetics*, pages genetics–113, 2013.

Xiaoming Liu and Yun-Xin Fu. Exploring population size changes using snp frequency spectra. *Nature genetics*, 47(5):555, 2015.

John Moult, Jan T Pedersen, Richard Judson, and Krzysztof Fidelis. A large-scale experiment to assess protein structure prediction methods. *Proteins: Structure, Function, and Bioinformatics*, 23(3):ii–iv, 1995.

Diego Ortega-Del Vecchyo, Kirk E. Lohmueller, and John Novembre. Haplotype-based inference of the distribution of fitness effects. *bioRxiv*, 2019. doi: 10.1101/770966. URL `https://www.biorxiv.org/content/early/2019/09/16/770966`.

Aaron P Ragsdale and Simon Gravel. Models of archaic admixture and recent history from two-locus statistics. *PLoS genetics*, 15(6):e1008204, 2019.

Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, et al. Imagenet large scale visual recognition challenge. *International journal of computer vision*, 115 (3):211–252, 2015.

P A Salomé, K Bomblies, J Fitz, R A E Laitinen, N Warthmann, L Yant, and D Weigel. The recombination landscape in arabidopsis thaliana f2 populations. *Heredity*, 108(4): 447–455, Nov 2011. ISSN 1365-2540. doi: 10.1038/hdy.2011.95. URL `http://dx.doi.org/10.1038/hdy.2011.95`.

Stephan Schiffels and Richard Durbin. Inferring human population size and separation history from multiple genome sequences. *Nature genetics*, 46(8):919, 2014.

Sara Sheehan and Yun S Song. Deep learning for population genetic inference. *PLoS computational biology*, 12(3):e1004845, 2016.

Lauren Alpert Sugden, Elizabeth G Atkinson, Annie P Fischer, Stephen Rong, Brenna M Henn, and Sohini Ramachandran. Localization of adaptive variants in human genomes using averaged one-dependence estimation. *Nature communications*, 9(1):703, 2018.

Paula Tataru, Maéva Mollion, Sylvain Glémin, and Thomas Bataillon. Inference of distribution of fitness effects and proportion of adaptive substitutions from polymorphism data. *Genetics*, 207(3):1103–1119, 2017.

Jacob A Tennessen, Abigail W Bigham, Timothy D O'Connor, Wenqing Fu, Eimear E Kenny, Simon Gravel, Sean McGee, Ron Do, Xiaoming Liu, Goo Jun, et al. Evolution and functional impact of rare coding variation from deep sequencing of human exomes. *science*, 337(6090):64–69, 2012.

Jonathan Terhorst, John A Kamm, and Yun S Song. Robust and scalable inference of population history from hundreds of unphased whole genomes. *Nature genetics*, 49(2): 303, 2017.

Supplemental Figures

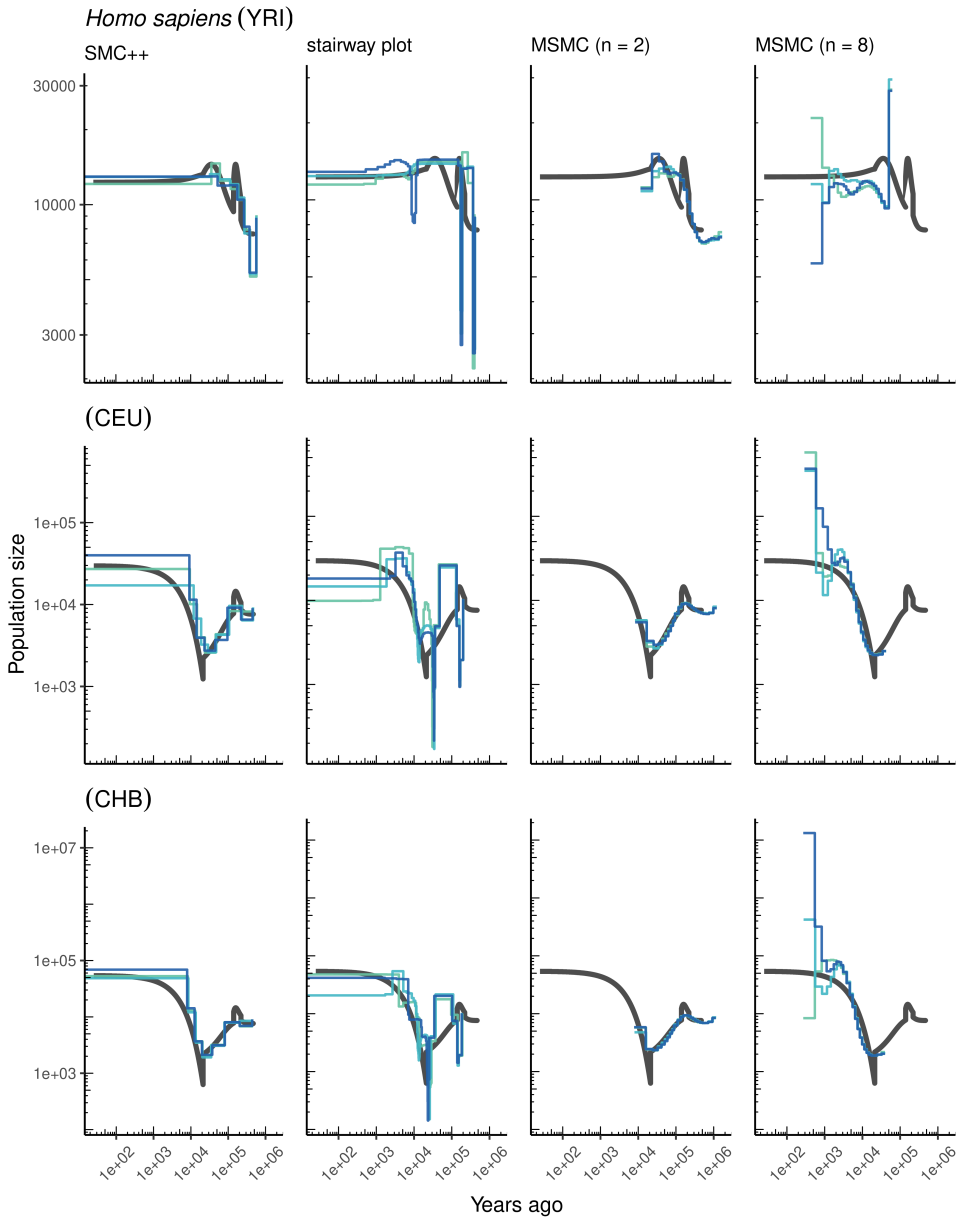

Figure S1: **Comparing estimates of $N(t)$ in humans**. Estimates of population size over time ($N(t)$) inferred using 4 different methods, `smc++`, `stairway plot`, and `MSMC` with $n = 2$ and $n = 8$. Data were generated by simulating replicate human genomes under the Gutenkunst et al. (2009) model and using the genetic map inferred in International HapMap Consortium et al. (2007). From top to bottom we show estimates for each of the three populations in the model: YRI, CEU, and CHB. In shades of blue we show the estimated $N(t)$ trajectories for each replicate. In black we show the true population size history as inferred for the rate of coalescence in the demographic model.

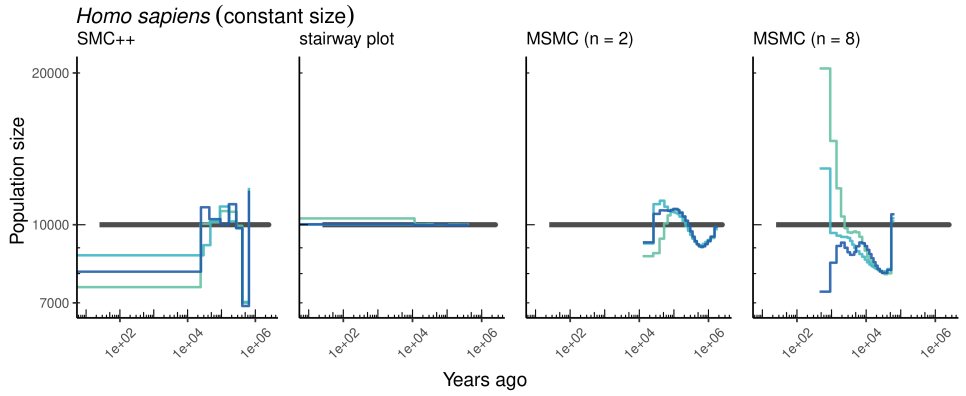

Figure S2: **Comparing estimates of $N(t)$ in humans**. Here we show estimates of population size over time ($N(t)$) inferred using 4 different methods, `smc++`, and `stairway plot`, `MSMC` with $n = 2$ and $n = 8$. Data were generated by simulating replicate human genomes under a constant sized population model with $N = 10^4$ and using the HapMapII genetic map (International HapMap Consortium et al., 2007). In black we show the true population size history of the model.

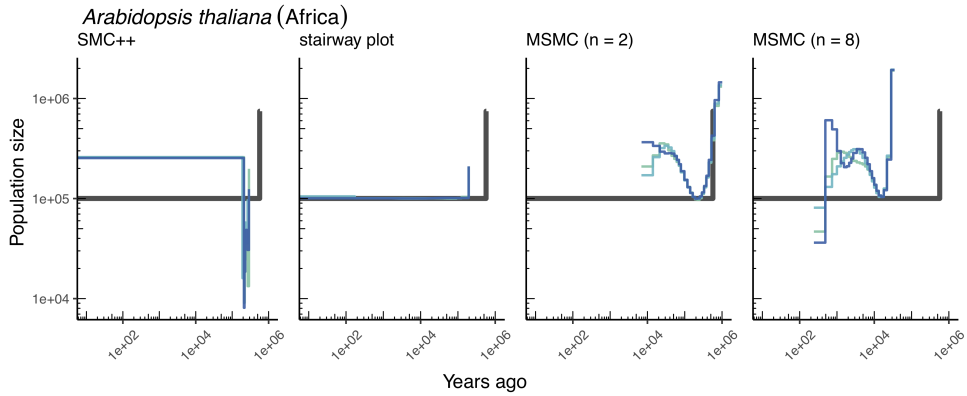

Figure S3: **Comparing estimates of $N(t)$ in *A. thaliana***. Here we show estimates of population size over time ($N(t)$) inferred using 4 different methods, `smc++`, and `stairway plot`, `MSMC` with $n = 2$ and $n = 8$. Data were generated by simulating replicate *A. thaliana* genomes under the African2Epoch_1H18 model and using the genetic map of Salomé et al. (2011). In black we show the true population size history of the model.

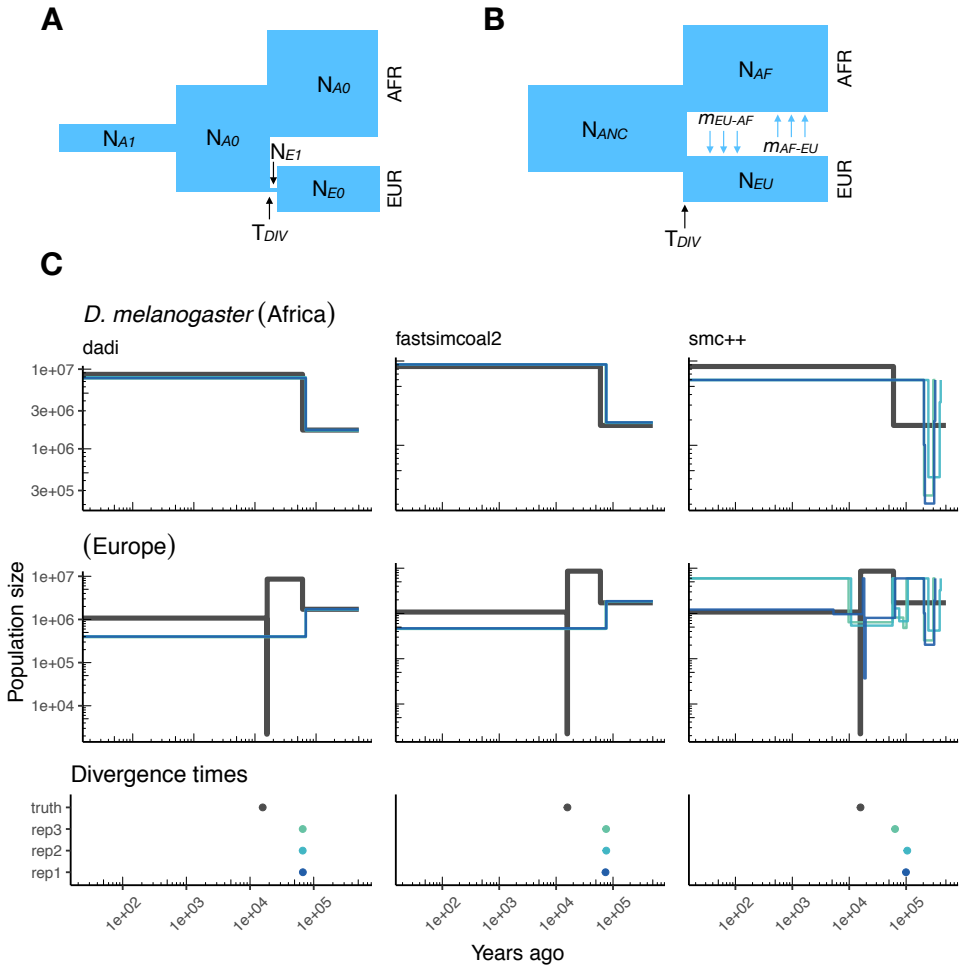

Figure S4: **Parameters estimated using a two-population *Drosophila* model.** Here we show estimates of $N(t)$ inferred using $\partial a \partial i$, `fastsimcoal2`, or `smc++`. Data were generated by simulating replicate *Drosophila* genomes under the Li and Stephan (2006) model and using the genetic map inferred in Comeron et al. (2012). See legend of Figure 4 for details. In shades of blue we show the estimated $N(t)$ trajectories for each replicate. In black we show the true population size history as given by the census size for the simulated model.

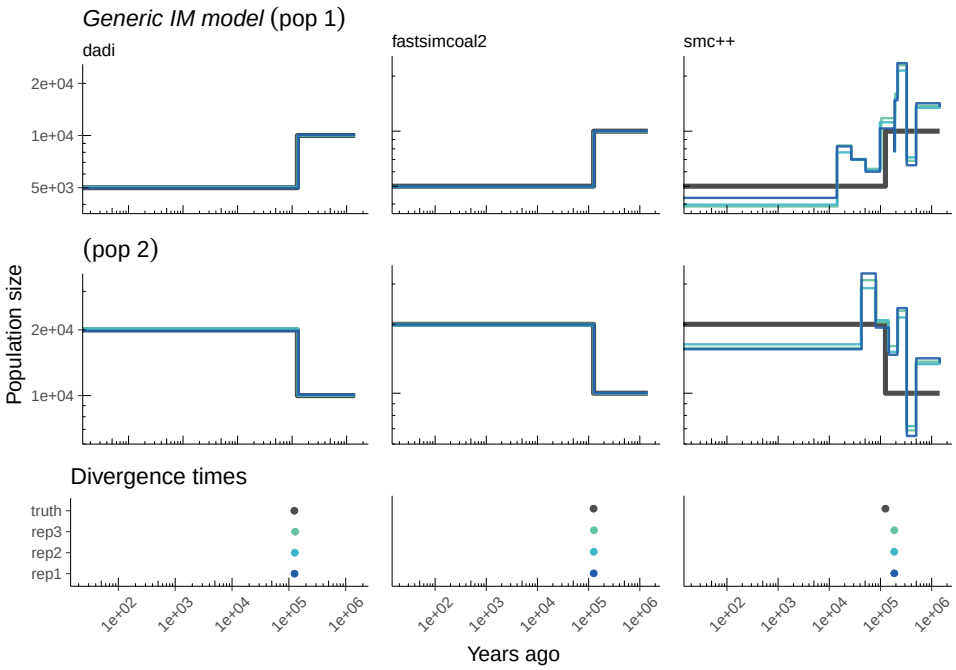

Figure S5: **Parameters estimated from a generic IM model** Here we show estimates of $N(t)$ inferred using $\partial a \partial i$, `fastsimcoal2`, or `smc++`. Data were generated by simulating under a generic IM model with a human genome and International HapMap Consortium et al. (2007) genetic map. In shades of blue we show the estimated $N(t)$ trajectories for each replicate. In black we show the true population size history as given by the census size for the simulated model.

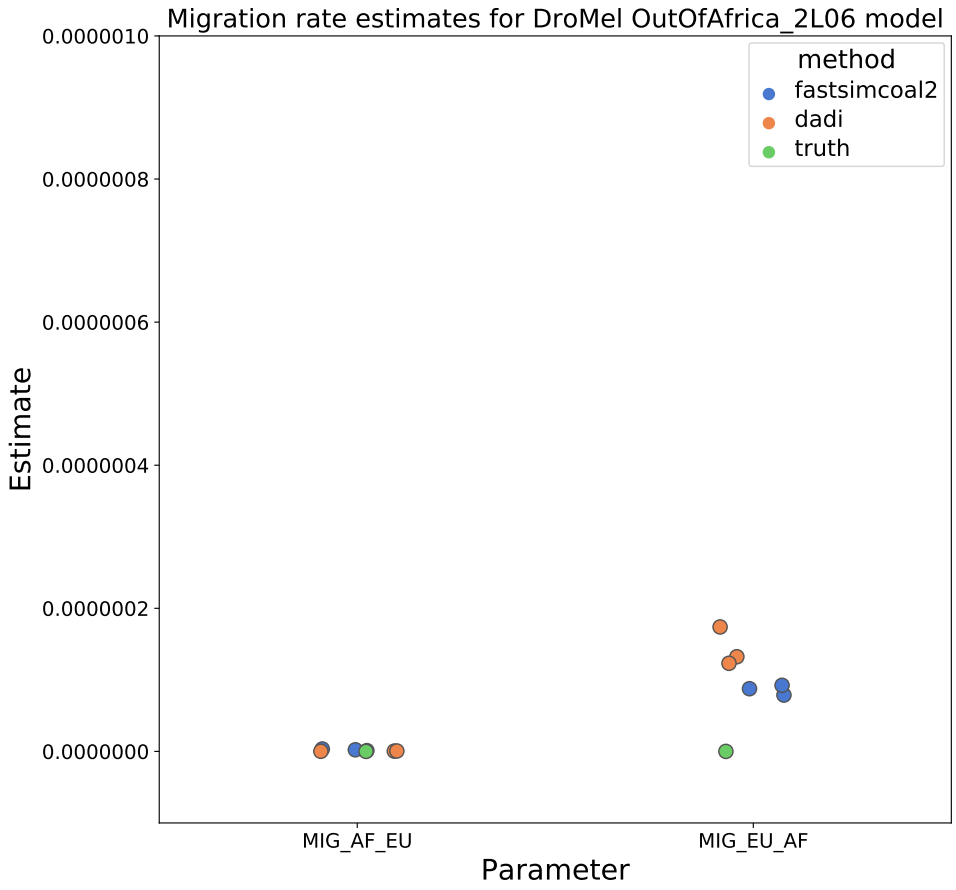

Figure S6: **Migration rate parameters estimated under a two-population *Drosophila* model**. Here we show inferred migration rates from $\partial a \partial i$ and `fastsimcoal2`. Data were generated by simulating replicate *Drosophila* genomes under the Li and Stephan (2006) model and using the genetic map inferred in Comeron et al. (2012).

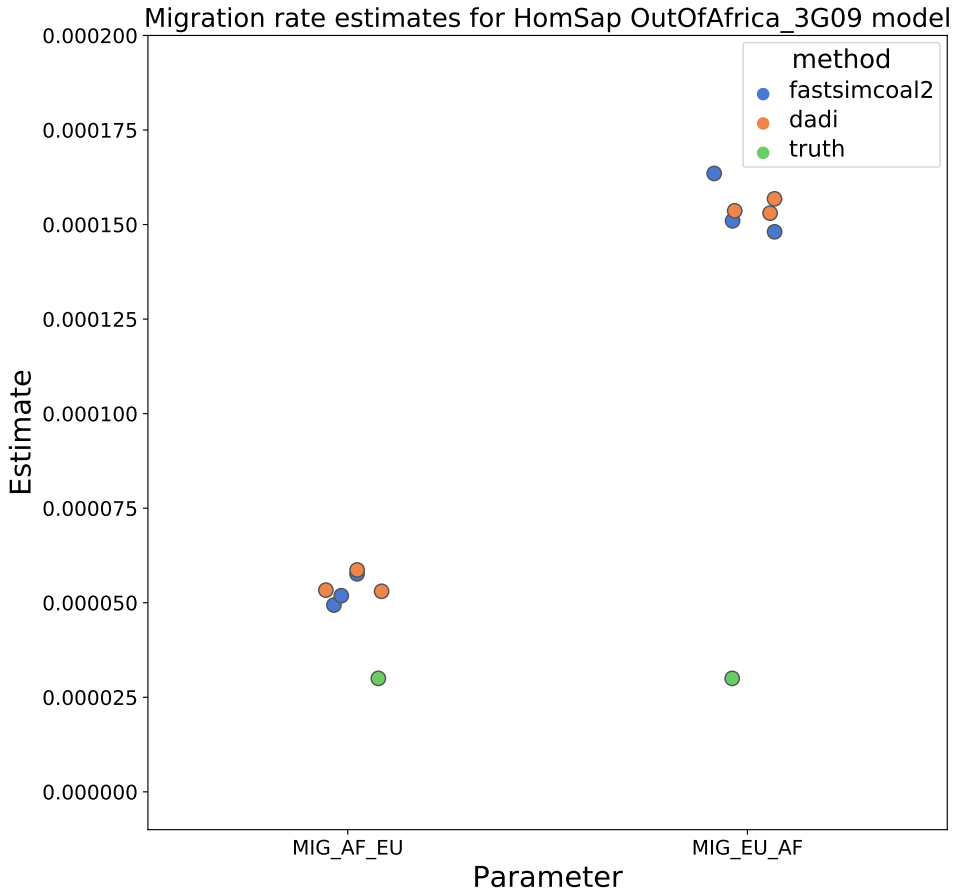

Figure S7: **Migration rate estimates for the human Gutenkunst model.** Here we show inferred migration rates from $\partial a \partial i$ and `fastsimcoal2`. Data were generated by simulating replicate human genomes under the Gutenkunst et al. (2009) model and using the genetic map inferred in International HapMap Consortium et al. (2007).

Calculating coalescence rates

We compute the coalescence rate of a collection of samples in a given demographic model at a particular point back in time as the expected number of coalescences happening at that time per unit of time and per pair of as-yet-uncoalesced lineages. More concretely, let $p(t)$ denote the probability that the lineages of a randomly chosen pair of samples have not yet coalesced t units of time ago, let $p(z,t)$ denote the probability that those lineages have not yet coalesced and are furthermore both in location z, and let $1/(2N_e(z,t))$ be the rate of coalescence in location z at the time. Then, we compute the mean coalescence rate as

$$r(t) = \frac{1}{p(t)} \sum_z \frac{p(z,t)}{2N_e(z,t)}.$$

This follows because if we have n diploid samples, and hence $\binom{2n}{2}$ lineages, the expected number of coalescences in location z between times t and $t+dt$ ago

$$\binom{2n}{2} p(z,t) \frac{dt}{2N_e(z,t)},$$

and the expected number of pairs of uncoalesced lineages at that time is

$$\binom{2n}{2} p(t).$$

The expression for $r(t)$ is a ratio of these two quantities; to obtain it we need to compute $p(t)$ and $p(z,t)$. This is relatively straightforward using the general theory of Markov chains, and is implemented in `msprime`.

Note that since these quantities are *per pair of lineages*, this definition depends on the locations of the samples. The coalescence rate also has the intuitive interpretation that it is the average between-lineage coalescence rate, averaged over where uncoalesced lineages might be. Since the local coalescence rate is the inverse of the population size, $1/r(t)$ (as shown for instance in Figure 2) is a weighted harmonic mean of the census sizes of the different populations present at that time. This is as expected: suppose that we have two populations, one big and one small, connected by migration. If all our samples are from the big population, the number of recent coalescences should be small, reflecting the large population size, while in the long run, the coalescence rate approaches an intermediate rate. On the other hand, more recent coalescences are expected if all samples are from the small population, A method that fits a single, time-varying population size to the data might be expected to find a population size trajectory to match these time-varying rates of coalescence.

We use the same computations to analytically compute *mean coalescence times*: since for any nonnegative random variable T, the mean value is $\mathbb{E}[T] = \int_0^\infty \mathbb{P}\{T > t\} dt$, we

can obtain the mean coalescence time as

$$\int_0^\infty p(t)dt,$$

where $p(t)$ is defined above.