Inferring genomic landscapes with the integrative sequentially Markov coalescent (iSMC)

Gustavo V. Barroso[0000−0002−1943−9297] and
Julien Y. Dutheil[0000−0001−7753−4121]

Abstract The integrative Sequentially Markovian Coalescent (iSMC) is an extension of the sequentially Markovian Coalescent (SMC) model allowing for parameter heterogeneity along the genome, such as recombination and mutation rates. Heterogeneous parameters follow an autocorrelation process that modulates the genealogical process, extending the hidden state space and adding as few as two extra parameters per heterogeneous rate. Classical hidden Markov chain methodology is used to infer the posterior estimate of the rate landscape. In this chapter, we demonstrate the use of iSMC to infer both recombination and mutation landscapes, using data from *Homo sapiens* and *Homo neanderthalensis* genomes. We further indicate how to use simulations to assess statistical power and investigate possible sources of inference noise.

Keywords: sequentially Markovian coalescent, coalescent hidden Markov models, recombination landscape, mutation landscape

Running head: Inferring genomic landscapes with iSMC

Gustavo V. Barroso
University of Wisconsin-Madison, 447 Birge Hall, Madison, Wisconsin, USA e-mail: gvbarroso@gmail.com

Julien Y. Dutheil
Max Planck Institute for Evolutionary Biology, August-Thienemann-Str. 2, 24306 Plön, Germany
e-mail: dutheil@evolbio.mpg.de

1 Introduction

The ancestral recombination graph (ARG) is the collection of coalescence and recombination events in the history of a sample of haplotypes. As such, it is a complete description of the genealogical history in a sample of genomes. With knowledge of the true ARG, it would be straightforward to estimate key evolutionary parameters such as ancestral population sizes and recombination rates. However, because many events do not leave a signature in the sequence data, a large number of configurations are compatible with a given dataset. Therefore, inference methods typically rely on population genetic models to integrate over possible ARGs to account for the uncertainty in its reconstruction. In this chapter, we focus on describing one such class of models and their application to inferring molecular processes that vary along the genome, such as mutation and recombination rates, while accounting for temporal variation in population sizes.

The sequential coalescent frames the genealogical process as unfolding spatially along the genome [1]. Starting at the leftmost position and moving site by site, the coalescent tree subtending a sample of haplotypes remains the same until an ancestral recombination event separates lines of descent. The ensuing 'floating lineage' then follows its own coalescent trajectory, resulting in a new genealogy that spans until the next recombination event, and this process is iterated down to the last position in the genome, allowing the construction of the ARG. The sequentially Markovian coalescent (SMC) is an approximation of this process in which some parts of the ARG (which do not have any direct descendants in the sample) are discarded [2]. Therefore, unlike the classic (and unwieldy) backward-in-time model, the SMC enforces that recombination events always happen rightward, restricting the uncertainty to the temporal dynamics of recombination and coalescence events, for which there are accurate population genetic predictions. This allows the SMC to be formulated

as a hidden Markov model (HMM) with genealogies and nucleotide configurations as hidden and observed states, respectively [3]. As the coalescence trajectory of recombining lineages is a function of ancestral population sizes, this model forms the basis of many demographic inference tools that have been developed in the past 15 years.

Besides demography, the SMC framework has been used to model evolutionary processes that vary in magnitude along the genome. In 2018, Palamara et al. [4] used hundreds of genomes to identify regions with consistently shallow coalescence events as candidates for natural selection. In the following year, Barroso et al. [5] added further complexity to the SMC by casting it as a Markov-modulated HMM, where a prior distribution of recombination rates modulates the frequency of genealogical transitions along the genome. These local changes in the auto-correlation of the ARG are recorded in the sequence data because patterns of diversity are a function of the underlying genealogies. This flexible integrative SMC framework (ismc) was later used to incorporate spatial variation in mutation rates into the model [6].

The approach of ismc is to first estimate the distributions of recombination and/or mutation rates, then use well-established HMM techniques to reconstruct their genomic landscapes. In practice, the statistical power to perform fine-scale inferences is a function (among other things) of the level of diversity maintained in the population. In humans, where pairwise diversity is on the order of 10^{-4}, the resolution of inferred maps is low, with estimates generally noisy on scales $< 100kb$. However, with larger window sizes, the number of recombination events recorded is enough that the landscapes are well recovered, even from a single diploid genome. In this chapter, we guide the user through a typical application of the iSMC to infer recombination maps in the Altai Neanderthal. We then infer the human mutation map for chromosome 1 and compare our results with mutation rates that were recently predicted using a collection of genomic features.

2 Getting and running iSMC

The ismc source code can be obtained from GitHub https://github.com/gvbarroso/iSMC/. The package is written in C++ and depends on the BOOST and Bio++ libraries [7]. Standalone executable programs can be downloaded for Linux 64-bit systems.

ismc uses the Bio++ Option (BppO) syntax to specify models and options, such as input and output file paths. The options can be conveniently saved in one or more option files. ismc can then be run using the command line

Terminal command

```
ismc param=options.bpp
```

where options.bpp contains a list of program options. One of these options is dataset_label, which defines a prefix tag that will be used for all output files. In the following, we will refer to this suffix as [PREFIX] when mentioning the output files.

3 Input data

The core information used by ismc is a list of homozygous (coded '0') and heterozygous positions (coded '1') in at least one pair of haplotypes, possibly with missing data (coded '2'). Following the approach introduced by MSMC2 [8], ismc implements a composite likelihood approach where the likelihood of a data set is the likelihood product of multiple pairs of haplotypes. These pairs may correspond to several

(possibly unphased) diploid individuals or multiple combinations of individual haplotypes (up to $\frac{2n\times(2n-1)}{2}$ possible pairs for n phased diploid individuals). Sequence data can be provided in different formats. Individual haplotypes can be provided as complete sequences in a file in FASTA format, with missing data coded 'N'. Alternatively, variable positions for each sampled individual can be provided as a file in Variant Call Format (VCF). Genome VCF (gVCF) files provide both variable and non-variable positions, other positions being considered missing data. Conversely, simple VCF files only contain variable positions, and an additional mask file should be provided to distinguish non-variable positions from missing data.

To run ismc from a Fasta file, the following options are required:

Option file

```
input_file_type = FASTA
sequence_file_path = sequences.fasta.gz
seq_compression_type = gzip
```

Input files can be provided as compressed, either by gzip or bgzip. To use the VCF format instead, these options are required, including the specification of a mask file:

Option file

```
input_file_type = VCF //or gVCF
sequence_file_path = variants.vcf.gz
mask_file_type = BED
mask_file_path = mask.bed
seq_compression_type = bgzip
mask_compression_type = none
```

The mask file can be in BED or Fasta format and must match (in content and order) the chromosomes in the sequence file. In the BED case, ismc will mask out (that is, convert to missing data) regions that are present in this file (i.e., a negative mask). In the Fasta case, ismc will mask out sites that are not represented by either '1' or 'P' characters. When a genome VCF file is used ($\mathrm{input\_file\_type} = \mathrm{gVCF}$), no mask file is required.

Only diploid individuals are allowed in the VCF file, whose phasing, if any, will be ignored. Each diploid will be considered as a pair of haplotypes to be analyzed. Conversely, when the data are input as a Fasta file, each sequence corresponds to a single haplotype. By default, ismc will then create all possible pairs of haplotypes and fit the model to all of them. The option $\mathrm{diploid\_indices}$ allows the user to restrict the haplotype pairs that are formed. For example, $\mathrm{diploid\_indices} = (0,1,1,2)$ will create only two pairs: 0 with 1 and 1 with 2.

4 Specifying the demographic model

Similarly to other SMC methods like MSMC2 and eSMC (see Chapters 6 and 7), ismc discretizes time in intervals. The number of categories to use is specified by the option $\mathrm{number\_intervals}$, with a value typically between 30 and 40. More intervals lead to more precise inferences, but longer execution times. Setting a too large number may lead to some intervals never being observed in the data if the number of variable positions in the genome is too small, leading to optimization issues.

Unlike MSMC2, where time intervals represent free population sizes to be estimated, ismc adopts cubic spline interpolation to reduce the number of model parameters. The idea is to have a smooth, continuous curve that spans the entire coalescent history, then map TMRCAs at the prescribed time boundaries, converting to a piecewise-constant history. This ensures that we have a discrete-space Markov chain.

To allow flexible size changes, ismc can incorporate multiple spline curves that meet at each end. This is set by the number of 'knots' k in the model (default = 2), where $k+1$ curves are fit. The height of the splines at each knot (in units of scaled coalescent rates) are the demographic parameters to be estimated by ismc (labeled y_0 to y_{k+1}). ismc can estimate and compare models with different numbers of knots and select the best-fitting model according to Akaike's information criterion (AIC). For instance, setting the options $\mathrm{init\_number\_knots} = 2$ and $\mathrm{max\_number\_knots} = 5$ will compare four models with two, three, four, and five knots, respectively.

5 Estimating parameters

ismc uses maximum likelihood inference to estimate parameters. Optimization is started using the $\mathrm{optimize} = \mathrm{true}$ option. The optimizer is set with the option numerical_optimizer, with two available options: Powell or NewtonRaphson. The Powell optimizer will fit most situations. The Newton-Raphson optimizer may converge faster, but requires computationally expensive numerical derivatives. In both cases, optimization will stop when the (relative) increase in logarithmic likelihood between two iterations is less than a given threshold, specified by the option function_tolerance. A threshold of 0.1 or below is recommended. The maximum likelihood framework also enables us to compute confidence intervals for the estimated parameter values using the Fisher information matrix. This calculation can be enabled using the option $\mathrm{compute\_confidence\_interval} = \mathrm{true}$. In this case, ismc outputs ConfidenceIntervals.txt which contains lower and upper 95% intervals for each model parameter.

ismc produces two log files that allow the user to assess the proper convergence of the parameter estimation procedure: a [PREFIX]_profile.txt file containing all parameter and log-likelihood values at each iteration, and a [PREFIX]_messages.txt

file with potential warnings generated during the optimization process. In case several demographic models are tested, one pair of files will be generated per model. Furthermore, ismc produces a [PREFIX]_backup_params.txt file that contains the current parameter value at each iteration. This file is updated as the optimization progresses. Since some ismc models can take a rather long time to fit, this temporary file allows us to restart the parameter estimation in case of hardware failure or misconfiguration of the job scheduling system when using a computer grid. This can be done by setting the option resume_optim = true. The option can also be used to provide initial parameter values, for instance, obtained from a simpler, faster model.

The final parameters are outputted as a [PREFIX]_estimates.txt file. The coalescence rates for each time interval are in a separate file [PREFIX]_demography.txt, which follows the same format as MSMC2 (see Chapter 6).

6 Example 1: homogeneous model

In order to illustrate the options that we have seen so far, we fit a simple homogeneous model to the Altai Neanderthal genome, restricting the analysis to chromosome 1. The complete data preparation pipeline can be found on the companion GitHub repository.

Option file

```
DATA=chr1
NB_KNOTS=2
dataset_label = $(DATA)_homogeneous_$(NB_KNOTS)knots

sequence_file_path = ../VCFnorm/$(DATA).norm.vcf.gz
mask_file_path = ../Bed/$(DATA)_negmask.bed.gz
mask_file_type = BED
input_file_type = VCF
```

```
seq_compression_type = gzip
mask_compression_type = gzip
tab_file_path = $(DATA).tab
diploid_indices = (0,1)

splines_type = Sigmoidal
init_number_knots = $(NB_KNOTS)
max_number_knots = $(NB_KNOTS)

number_theta_categories = 1
number_rho_categories = 1
number_ne_categories = 1
number_intervals = 30

numerical_optimizer = Powell
function_tolerance = 1e-1
confidence_interval = false

optimize = true
```

The script makes use of the BppO variable syntax, defining two global variables: DATA, which specifies chromosome 1 (this allows to easily run the same model on other chromosomes with no further modification), and NB_KNOTS, which specifies the number of knots in the spline model. (For convenience, we run the model 4 times with four different numbers of knots instead of letting it run all models at once.) The options tab_file_path will be described in section 13.

The inferred demography consistently reveals a decrease in population size (Figure 1). As ismc and MSMC2 use an identical population genetic model (albeit with different implementations and estimation procedures), they produce similar results. MSMC2 displays a typical off-the-charts high population size in the recent past, while ismc provides a smoother curve due to the use of splines. However, the demography inferred by ismc has a lower resolution and does not capture intermediate peaks. Using a higher number of knots provides more nuance, but a large number

of knots (n = 5) yields a most likely artifact peak in the distant past. The lowest AIC value is obtained with the four-knot model (see Table 1). Using more data (all chromosomes) would allow us to fit more precise models.

In the next sections, we will see how to run ismc to estimate two types of landscape: recombination and mutation.

7 Specifying a heterogeneous model with variable recombination rate

The strength of ismc lies in its ability to model heterogeneous parameters along the genome. Fitting such models requires specifying (1) the a priori discrete distribution of ρ along the genome and (2) the transition probabilities between ρ categories. Property (1) is achieved with option $\mathrm{rho\_var\_model} = \mathrm{Gamma}$, specifying a (discretized) gamma distribution with mean 1. The shape of this distribution is estimated together with the demography parameters. The number of ρ categories is specified with option $\mathrm{number\_rho\_categories}$. Just like for the time intervals, there is a trade-off between accuracy (more categories) and computational efficiency (less categories). We recommend using a number of five categories. ismc currently assumes a simple auto-correlation model with a single rate of transitions between categories (Property 2).

8 Example 2: variable recombination rate

Here we extend the example from section 6 to fit a model with variable recombination rate:

Option file

```
DATA=chr1
NB_KNOTS=2
NB_RHO=5
dataset_label = $(DATA)_gamma$(NB_RHO)_$(NB_KNOTS)knots

sequence_file_path = ../VCFnorm/$(DATA).norm.vcf.gz
mask_file_path =  ../Bed/$(DATA)_negmask.bed.gz
mask_file_type = BED
input_file_type = VCF
seq_compression_type = gzip
mask_compression_type = gzip
tab_file_path = $(DATA).tab
diploid_indices = (0,1)

splines_type = Sigmoidal
init_number_knots = $(NB_KNOTS)
max_number_knots = $(NB_KNOTS)

number_theta_categories = 1
number_rho_categories = $(NB_RHO)
number_ne_categories = 1
number_intervals = 30

rho_var_model = Gamma

numerical_optimizer = Powell
function_tolerance = 1e-1
confidence_interval = false

optimize = true
resum_optim = false
```

The option file is mostly identical; we only added the options for the heterogeneous recombination rate. The inferred demographic scenarios are very similar to those inferred with the homogeneous model (Figure 2). When comparing all mod-

Table 1 Models fitted to the Altai neanderthal dataset.

Model	[a]	[b]	Log likelihood	AIC
Homogeneous	2	5	-369098.22838652721839	738206.5
Homogeneous	3	6	-369109.84457955235848	738231.7
Homogeneous	4	7	-369029.35426597035257	738072.7
Homogeneous	5	8	-369105.81665211619111	738227.6
Gamma-5	2	7	-368181.47791081061587	736377.0
Gamma-5	3	8	-368165.42031291732565	736346.8
Gamma-5	4	9	-368130.59860143717378	736279.2
Gamma-5	5	10	-368119.60966349433875	736259.2

[a] Number of knots in the demography model
[b] Total number of parameters

els, Akaike's information criterion favors the most complex one, with five knots and a heterogeneous landscape (Table 1).

9 Inference of recombination maps

Once a heterogeneous model has been fitted, it is possible to use posterior decoding to obtain site-specific estimates of recombination rates, which can be averaged in windows along the genome.

The option $\mathrm{decode} = \mathrm{true}$ will trigger the posterior decoding and output the results in multiple files for each pair of genomes. As posterior decoding may require a large amount of memory, slicing the chromosome in blocks is important for computational efficiency. The maximum length of such blocks is specified by the $\mathrm{fragment\_size}$ option, with a value set to 10,000,000 that generally enables a reasonable compromise (see section 13). In case the model was fitted but not decoded, it is possible to rerun ismc by enabling the decoding options and setting the option $\mathrm{optimize} = \mathrm{false}$; this will use the previously estimated parameters to generate the posterior decoding, without further optimization.

The ismc package contains an auxiliary program named ismc_mapper that takes the single-nucleotide landscapes output by ismc and uses the information present in the tab file to output rate maps (averaged within larger genomic windows). ismc_mapper requires its own option file, with a syntax similar to the main ismc program:

Option file

```
DATA=chr1
NB_KNOTS=2
dataset_label = $(DATA)_gamma5_$(NB_KNOTS)knots
tab_file_path = $(DATA).tab
bin_sizes = (10000,100000,1000000)
bin_rate = rho
```

The two arguments dataset_label and tab_file_path mirror ismc's options file and must be provided identically. The remaining options are specific to ismc_mapper and tell the program how to aggregate the single-nucleotide landscapes. The bin_sizes argument specifies multiple window sizes, here 10 kb, 100 kb and 1 Mb. The argument bin_rate specifies the variable of interest, in our case ρ (but see Note 1). The resulting files will contain averaged estimates of ρ per window, over all individuals and all positions in each window.

Figure 3A shows the resulting window estimates plotted along the chromosome, showing the typical ''smiling'' pattern with higher recombination rates close to the telomeres. Recombination estimates are robust to the demography model and are highly correlated when two or five knots are used (Figure 3B).

10 Power analysis using simulations

While in this example the inferred demography and recombination map appear plausible and in agreement with current knowledge about the neanderthal history and primates biology in general, it might be difficult to make sense out of parameters inferred from non-model organisms where little is known. To assess whether the estimates of ismc (or any other method) are reliable, it is recommended to conduct a power analysis. The idea is to simulate datasets with identical characteristics to the real one (alignment length, number of contigs, genetic diversity) under the inferred scenario (population size variation + recombination landscape) to assess how likely we are to recover the scenario if it were the true one.

We use msprime [9] (see Chapter 15) to simulate data, with the demography inferred with the selected model and the 10 kb-scale recombination map. The corresponding code is available in the companion repository at https://github.com/StatisticalPopulationGenomics-2ndEd/iSMC/. The resulting VCF files are indexed on the original neanderthal VCF by creating an artificial SNP at the first and last positions of the original Neanderthal chromosome. The original mask file was then applied to the simulated data during the inference.

Demography could be well recovered from the simulation, demonstrating sufficient statistical power for demography inference (Figure 4). In contrast, despite showing significant correlations (Table 2), recombination rates were largely underestimated (Figure 5A). Recombination rates are bounded by the values of the minimum and maximum categories of the discretized prior distribution. The effect is notable here, as ismc infers a shape parameter higher than 0.5, ten times higher than the simulated value. This is an artifact of binning the recombination map in windows, resulting in a normalization of the landscape.

Increasing the number of categories did not improve the estimation (Table 2, second row). Using a manual discretization scheme that allows for lower and higher categories, yet with low probabilities, improves the estimate (Figure 5B) but not the correlation with the true map (Table 2, third row). Note that the reported correlations are rank-based, which indicates that the underestimation is only a scaling issue, relative rates are largely unaffected. Custom discretization can be achieved by specifying the following argument to ismc:

Option file

```
rho_boundaries=(0.0001, 0.001, 0.01, 0.1, 0.5, 1, 10)
```

This argument supersedes the number_rho_categories argument as the number of categories is set to the number of bounds minus one (see also Note 2). ismc recovers the 1-Mb recombination landscape with a good accuracy, but this accuracy decreases at smaller scales, becoming lower than previously reported values [5]. We can use simulations to investigate possible causes.

Ancient genomes are typically of lower quality compared to genomes of extant species. This leads to an increased proportion of missing data (*i.e.*, more masked positions). To investigate how much this reduces the accuracy of the inference, we re-analized the simulated data without masking positions. The correlations increase at all scales, but only marginally (Table 2, fourth row), indicating that missing data are not the main cause of lower precision in this case.

The inferred ρ value was relatively high, with a ratio $\rho/\theta > 1$, which can lead to a reduction in statistical power [5]. Surprisingly, simulating under the same scenario

Table 2 Power analysis for the Altai neanderthal dataset. Spearman correlation coefficients between the inferred and true map at three different window sizes. All correlations are significant with P values lower than 1e-200.

Method	10 kb	100 kb	1 Mb
Gamma (5 categories)	0.5345	0.5865	0.6890
Gamma (10 categories)	0.5323	0.5857	0.6860
Gamma (custom discretization)	0.5201	0.5721	0.6587
No mask	0.5353	0.5870	0.6883
Low recombination	0.4742	0.5391	0.6447
Constant population size	0.6946	0.7498	0.7772

but with a lower recombination rate led to even lower correlations (Table 2, fifth row), indicating that the high recombination rate is not the source of lower accuracy.

Finally, we investigated the singular ''collapsing'' demography of Neanderthals as a source of power reduction. Simulating under the same recombination map but with a flat demography of $Ne = 30,000$ resulted in much higher correlations (Table 2, bottom row), identifying demography as a main cause of power reduction.

11 Inference of mutation maps

Several, if not most, evolutionary models are sensitive to variation in mutation rate along the genome. Despite their importance, the development of tools for inferring mutation maps is lagging behind other statistical models. On the one hand, identification of *de novo* mutations (e.g., in family trios or biobank-scale datasets) is particularly sensitive to sequencing artifacts, besides being impractical in most species. On the other hand, population genetic methods struggle because disentangling the effect of mutation rate variation from genetic drift and linked selection is challenging.

ismc can jointly model mutation and recombination landscapes, alongside the TMRCA distribution. Under the neutral SMC model, variation in the TMRCA represents genetic drift. Informally, the hope is that it also absorbs linked selection effects,

such that ismc learns the mutation landscape from the remaining variation in diversity along the genome. However, due to the lack of empirically validated mutation maps, this has been difficult to benchmark. Although simulations suggest that ismc recovers the mutation landscape under neutrality and that it can be robust to linked selection, questions remain about performance in real data. Fortunately, due to its importance in evolutionary and medical genetics, the human mutation map has recently been predicted from genomic features such as sequence context and epigenetic markers [10].

12 Example 3: variable mutation rate

To compare our model with this independent estimate of mutation rate variation, we fit ismc to high-quality sequence data from four individuals of the Mbuti population (chromosome 1).

Option file

```
dataset_label=Mbuti_chr1
input_file_type = VCF
sequence_file_path = mbuti_chr1.vcf.gz
mask_file_path = sgdp_chr1_mask.fa.gz
mask_file_type = FASTA
tab_file_path = Mbuti_chr1.tab

seq_compression_type = gzip
mask_compression_type = gzip

diploid_indices = (0,1,2,3,4,5,6,7)

optimize = $(OPTIM)
numerical_optimizer = Powell
confidence_interval = false
function_tolerance = 1e-3
```

```
number_threads = 4

number_rho_categories = 5
number_theta_categories = 5
number_intervals = 30

decode = $(DECODE)
decode_diploids_parallel = true
decode_breakpoints_parallel = false
fragment_size = 800000

init_number_knots = 2
max_number_knots = 2
splines_type = Sigmoidal
rho_var_model = Gamma
theta_var_model = Gamma
```

This option file specifies a model with heterogeneous mutation and recombination landscapes in a $30 \times 5 \times 5$ configuration. Since fitting such a complex model requires hefty computational resources, we run optimization and posterior decoding separately (see section 13), parallelizing each step over individuals. (Due to the large run-times, in this example we refrain from comparing models.)

Command line

```
ismc param=opt.bpp OPTIM=true DECODE=false
```

Although optimization does not require large amounts of memory, it takes about two weeks to fit a model with 750 hidden states on the largest human chromosome. (A workaround is to use the tab file to specify breakpoints around the centromere, effectively treating chromosome arms with independent genealogical processes and

accordingly doubling the number of computing threads.) The estimated parameters shaping the mutation landscape suggest an approximately bell-shaped distribution (Gamma $\alpha = \beta \approx 4$) with auto-correlation $t_{ii} \approx 0.99998$, i.e., changing mutation rates on average every 50 kb. In contrast, the inferred recombination distribution is exponential-like (Gamma $\alpha = \beta \approx 0.1$) and less auto-correlated ($r_{ii} \approx 0.99995$).

We then use the estimated parameters to reconstruct the posterior landscapes:

Command line

```
ismc param=opt.bpp OPTIM=false DECODE=true
```

To parallelize the decoding over diploids, we specified a short fragment_size (800 kb) in the options file. This option tunes peak memory use, since ismc resets the matrix of posterior probabilities after extending it over the specified length. After another two weeks and peaking at around 128 Gb of memory, ismc output single-nucleotide landscapes that were binned by ismc_mapper with the following options:

Options file

```
dataset_label = Mbuti_chr1
bin_rate = theta
bin_sizes = (10000,100000,1000000)
tab_file_path = Mbuti_chr1.tab
```

The correlations between the mutation maps predicted by ismc and Roulette increase dramatically with window size (Figure 6). This has at least two potential ex-

planations. First, noisy estimates at smaller scales are smoothed out when averaging over larger genomic bins. This effect should be observed in both methods, but is especially strong in ismc, which extracts the signal from a small sample of genomes with average pairwise diversity $< 10^{-4}$ (see fluctuations in mutation maps as a function of scale, Figure 7). Second, different biological factors shape mutation rate variation at different scales, with triplet context in one extreme and others (e.g. replication timing) likely making a difference in the order of dozens of kilobases. Because it ultimately relies on sparse polymorphisms as its source of information, ismc is more sensitive to capture larger-scale effects.

Overall, the correlations between ismc and Roulette mutation maps are encouraging. They indicate that although fitting these models is computationally expensive, ismc is capable of inferring mutation maps from a small sample with fairly low diversity, especially in large genomic windows. To this end, high-quality sequence data is critical. The model accumulates information as it traverses the genome and, therefore, relies on long blocks of informative sites (which are interrupted if missing data are widespread). However, further benchmarking is required if strong model violations are suspected. We suggest caution in the inference and analyses of mutation maps, but we are optimistic that ismc opens new avenues in the population genetics of non-model organisms.

13 Practical considerations

13.1 Computer resource management

Depending on genome length, sample size, and model complexity (i.e., the total number of hidden states), ismc can use a large amount of computer resources. To mitigate execution time, ismc uses multi-threading to compute the likelihood, spreading the

calculation over several cores (set by number_threads option). This parallelization is performed on the individual and their chromosomes. That is, if data consist of, e.g. two diploids with three chromosomes each, ismc will benefit from using up to six threads. While likelihood calculation does not require extensive live memory (intermediate results are not stored), posterior decoding requires recording conditional likelihoods for every hidden state and position in the genome. As this can be quite large, ismc offers the ability to decode the genome in windows. This saves memory, but implies resetting the Markov chain at every window boundary. The size of the window can be set using the fragment_size argument. The larger the value, the more accurate the decoding is, but the more memory is required. The minimal value depends on the recombination rate and genetic diversity; as a rule of thumb, we recommend not getting below 1 Mb. Similarly to parameter estimation by maximum likelihood, posterior decoding can also be sped up with multi-threading. Since this comes at the cost of a linear increase in memory, ismc provides two boolean options to specify axes of parallelization and fine-tune resource management: decode_diploids_parallel and decode_breakpoints_parallel (see Example 3: variable mutation rate), both of which default to false.

13.2 Two-pass estimation

ismc employs a two-step approach to reconstruct the rate maps. The first step is the maximum likelihood estimation of the parameters governing the prior distributions of mutation, recombination, and TMRCAs along the genome. (This includes fitting a demographic model.) The second step is to use the inferred distributions to reconstruct the posterior average μ and r for every site in the genome.

In practice, we can exploit this two-step procedure to better allocate computational resources. The optimization step employs several evaluations of the likelihood

function (the compressed ‘‘forward’’ algorithm of the HMM) and therefore consumes substantial amounts of time but not memory. On the other hand, computing the posterior average requires a single execution of the full ‘‘forward-backward’’ HMM algorithm, which not only can be time-consuming but also requires considerable memory. It is therefore desirable to run these steps separately, first by specifying

Option file

```
optimize = true
decode = false
```

in the option file, subsequently inverting both booleans before running iSMC again. The reconstructed rate landscapes are scaled by the effective population size, $\rho = 4 \times N_e \times r$ and $\theta = 4 \times N_e \times \mu$ (but see Note 3).

13.3 Tab files

The options file specifies the various inputs for ismc. Among these is a tab-separated text file that summarizes the boundaries of contiguous ”blocks” (e.g., chromosomes or scaffolds) that are labeled in the sequence data. ismc treats each block independently, i.e., assumes free recombination among them. They are listed one per line, with seven columns in total (coordinates are 1-indexed):

1. Block ID (e.g. ’chr1’)
2. The start coordinate of the block, mapped to the reference genome
3. The end coordinate of the block, mapped to the reference genome
4. 0 for all blocks

5. The difference between the third and second columns
6. Bottom cut-off position from where ismc_mapper should bin single-nucleotide landscapes into larger genomic windows
7. Top cut-off position until where ismc_mapper should bin single-nucleotide landscapes into larger genomic windows.

The last two columns are used by ismc_mapper and are convenient for trimming the final maps into a range of sites that match that of another file. For example, if the input sequence data for chromosome 1 spans positions 5,000–10,000,000, and we want to align it with an experimental genetic map that ranges from 10,000 to 8,000,000, columns 6-7 should be 10,000 and 8,000,000. Similarly, if one wishes to align rate maps from different VCFs files, column 6 should have the largest of the starting positions, whereas column 7 should have the smallest of the end positions among the VCFs. If such trimming is not desirable, columns 6 and 7 should be the same as columns 2 and 3. The Python script generate_tab.py, provided with the ismc package, generates a default tab file from a VCF file.

Competing Interests The authors have no conflicts of interest to declare that are relevant to the content of this chapter.

14 Notes

Note 1: multiple decoding

If multiple variables (e.g. ρ and θ) have been decoded in the same model, ismc_mapper must be run separately for each of them.

Note 2: discretization of prior distribution

The default behavior is to consider categories with equal mass, which is usually the best option for estimating the hyper-parameters (parameters of the prior distribution). The number of categories is specified by the argument number_rho_categories. A drawback of this approach is that some category values might become very close, if not identical, in the case of an extremely skewed distribution (low alpha). It is also possible to use an equally distributed grid by specifying max_rho_value, which will create *n* equally spaced categories between 0 and the specified value.

Note 3: definition of ρ

Although the population recombination rate is typically defined as $\rho = 4 \cdot Ne \cdot r$, simulation revealed that ρ in ismc is off by a factor of 2, which means that it is defined as $\rho = 2 \cdot Ne \cdot r$. This implementation was kept for the sake of backward compatibility. Estimates should be corrected accordingly when converting ρ to r:

$$\frac{\rho}{\theta} = \frac{2 \cdot Ne \cdot r}{4 \cdot Ne \cdot u} \tag{1}$$

$$= \frac{r}{2u} \tag{2}$$

$$r = 2u\frac{\rho}{\theta} \tag{3}$$

References

1. Carsten Wiuf and Jotun Hein. Recombination as a point process along sequences. *Theoretical population biology*, 55(3):248–259, 1999.
2. Gilean AT McVean and Niall J Cardin. Approximating the coalescent with recombination. *Philosophical Transactions of the Royal Society B: Biological Sciences*, 360(1459):1387–1393, 2005.
3. Jeffrey P Spence, Matthias Steinrücken, Jonathan Terhorst, and Yun S Song. Inference of population history using coalescent hmms: review and outlook. *Current opinion in genetics & development*, 53:70–76, 2018.
4. Pier Francesco Palamara, Jonathan Terhorst, Yun S. Song, and Alkes L. Price. High-throughput inference of pairwise coalescence times identifies signals of selection and enriched disease heritability. *Nat. Genet.*, 50(9):1311–1317, September 2018.
5. 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, November 2019.
6. Gustavo V. Barroso and Julien Y. Dutheil. The landscape of nucleotide diversity in *Drosophila melanogaster* is shaped by mutation rate variation. *Peer Community Journal*, 3, 2023.
7. Laurent Guéguen, Sylvain Gaillard, Bastien Boussau, Manolo Gouy, Mathieu Groussin, Nicolas C Rochette, Thomas Bigot, David Fournier, Fanny Pouyet, Vincent Cahais, Aurélien Bernard, Céline Scornavacca, Benoît Nabholz, Annabelle Haudry, Loïc Dachary, Nicolas Galtier, Khalid Belkhir, and Julien Y Dutheil. Bio++: Efficient Extensible Libraries and Tools for Computational Molecular Evolution. *Mol. Biol. Evol.*, June 2013.
8. Anna-Sapfo Malaspinas, Michael C Westaway, Craig Muller, Vitor C Sousa, Oscar Lao, Isabel Alves, Anders Bergström, Georgios Athanasiadis, Jade Y Cheng, Jacob E Crawford, et al. A genomic history of aboriginal australia. *Nature*, 538(7624):207–214, 2016.
9. Franz Baumdicker, Gertjan Bisschop, Daniel Goldstein, Graham Gower, Aaron P. Ragsdale, Georgia Tsambos, Sha Zhu, Bjarki Eldon, E. Castedo Ellerman, Jared G. Galloway, Ariella L. Gladstein, Gregor Gorjanc, Bing Guo, Ben Jeffery, Warren W. Kretzschumar, Konrad Lohse, Michael Matschiner, Dominic Nelson, Nathaniel S. Pope, Consuelo D. Quinto-Cortés, Murillo F. Rodrigues, Kumar Saunack, Thibaut Sellinger, Kevin Thornton, Hugo van Kemenade, Anthony W. Wohns, Yan Wong, Simon Gravel, Andrew D. Kern, Jere Koskela, Peter L.

Ralph, and Jerome Kelleher. Efficient ancestry and mutation simulation with msprime 1.0. *Genetics*, 220(3):iyab229, March 2022.

10. Vladimir Seplyarskiy, Evan M Koch, Daniel J Lee, Joshua S Lichtman, Harding H Luan, and Shamil R Sunyaev. A mutation rate model at the basepair resolution identifies the mutagenic effect of polymerase iii transcription. *Nature genetics*, 55(12):2235–2242, 2023.

Figures

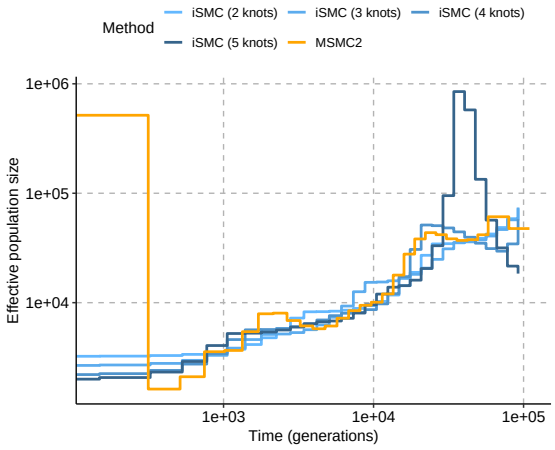


Fig. 1 Inferred demography from Altai Neandertal's chromosome 1. A mutation rate of 1.25e-8 bp^{-1} was used to scale the axes.

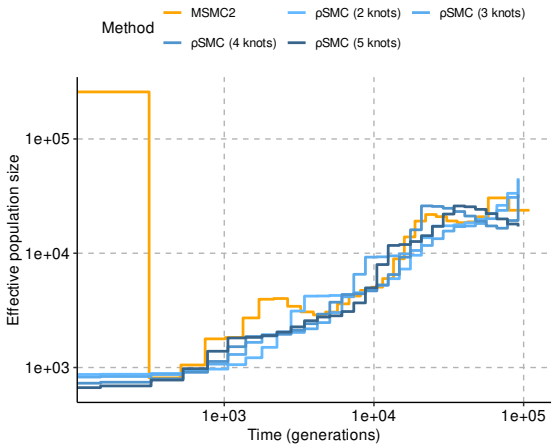


Fig. 2 Inferred demography from Altai Neanderthal chromosome 1, with a heterogeneous recombination rate. A mutation rate of 1.25e-8 bp^{-1} was used to scale the axes.

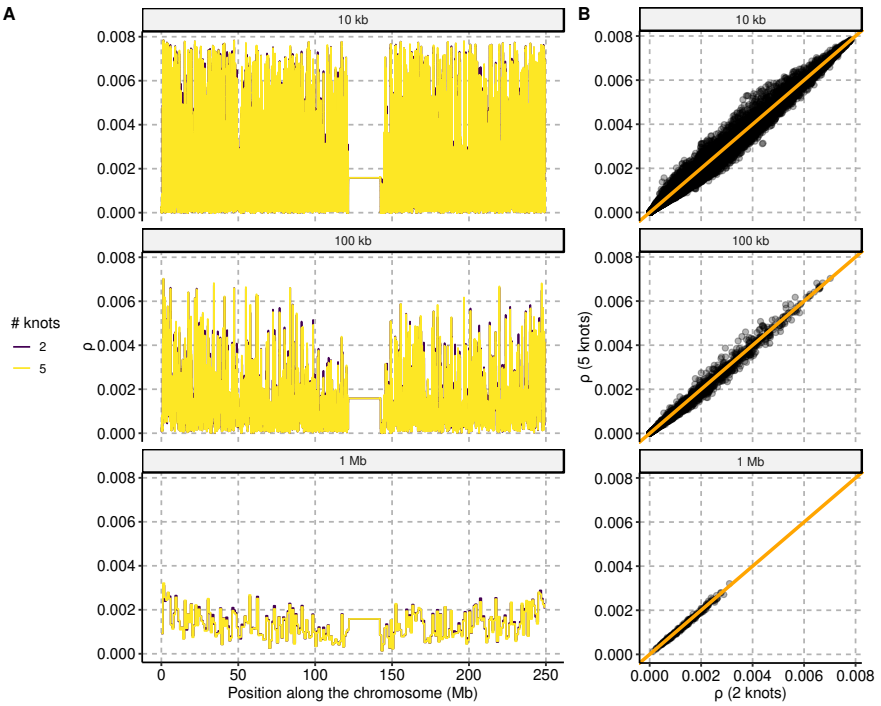
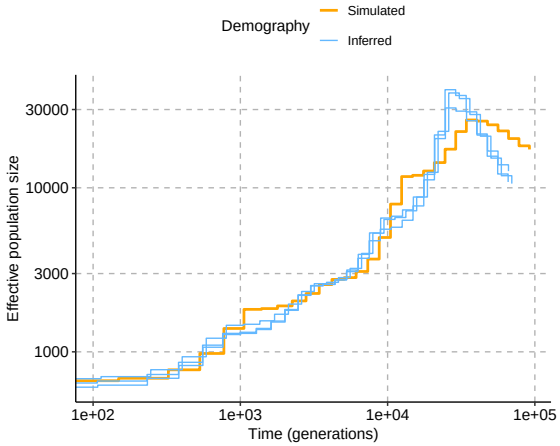


Fig. 3 Inferred recombination landscape of Altai Neanderthal chromosome 1, with distinct demography models and averaged over three different window sizes. A) Landscapes plotted along the chromosome, for two-node and five-node spline models. B) Comparison of window estimates under the two spline models. The orange line indicates the 1:1 ratio.

Fig. 4 Re-inferred demography from three simulated data sets using the model (demography + recombination map averaged in 10 kb windows) inferred from Altai Neanderthal's chromosome 1, with a heterogeneous recombination rate. A mutation rate of 1.25e-8 bp^{-1} was used in the simulations and to scale the axes.



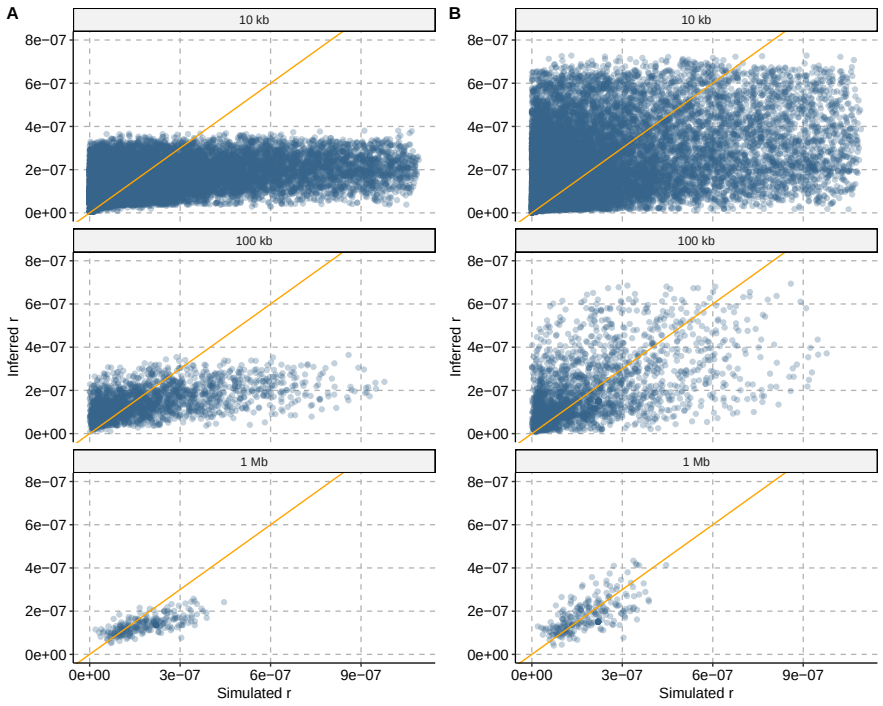


Fig. 5 Inferred recombination landscape from one simulation output (see Figure 4), plotted against the simulated landscape. A) Posterior decoding using the default discretization scheme. B) Posterior decoding using a custom discretization scheme with additional point mass at the tails of the distribution. The orange lines indicate the 1:1 ratio.

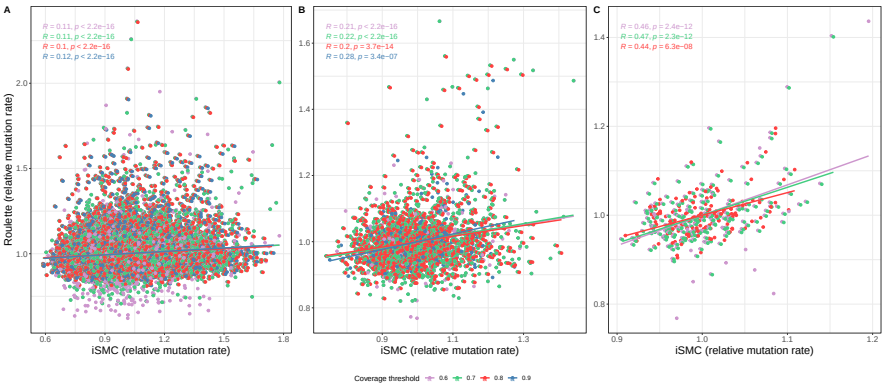


Fig. 6 Pearson correlations between iSMC-inferred and Roulette-predicted mutation rates at 10 kb (A), 100 kb (B) and 1 Mb (C) scales. Colors denote filtering thresholds, i.e., the minimum coverage required for keeping each window in the analysis.

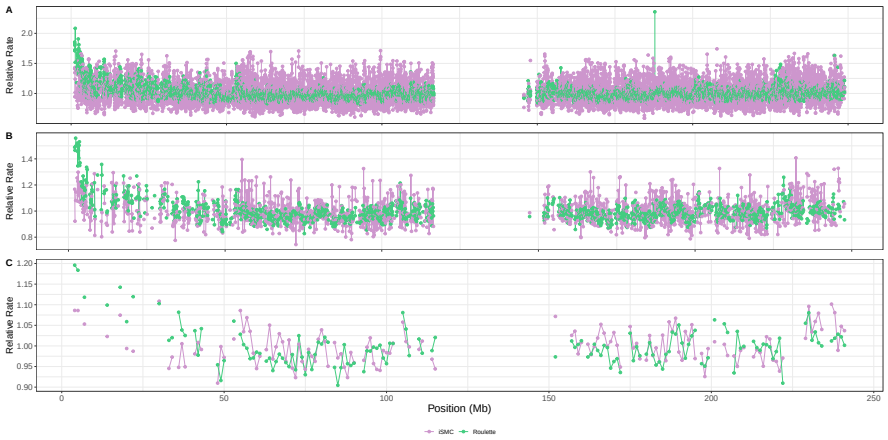


Fig. 7 Mutation maps for human chromosome 1, inferred with iSMC (purple) or predicted by Roulette (green) at 10 kb (A), 100 kb (B) and 1 Mb (C) scales. A coverage threshold of 80% is applied in each. Shown are relative rates for each method. iSMC displays larger variance than Roulette in A and B, likely due to estimation noise in these scales.