FacSexCoalescent

A coalescent simulator that accounts for partial frequencies of sex

Version 0.95

Matthew Hartfield, Stephen I. Wright, and Aneil F. Agrawal

http://github.com/MattHartfield/FacSexCoalescent

Email: matthew.hartfield@birc.au.dk

Last Updated: 9th August 2018

Contents

1 Introduction

FacSexCoalescent, also called **FSC**, is a program written in C, geared towards simulating genealogies from 'facultative' sexual organisms, which alternate between sexual and (parthenogenetic) asexual reproduction. The program reproduces how diversity is partitioned between samples, in order to determine the neutral (i.e., non-selective) forces that underlie these data.

The program uses coalescent theory to rebuild genealogies for each sample, and can produce many independent replicates in a short period of time. Mutations are then added to the genealogy using an infinite-sites model (i.e., it is assumed that neither multiple mutations at a site or back mutation do not arise). The program is run using Unix, or a Unix-type command line (such as Linux, or the Terminal in Mac OS X).

The main features of this simulation package are:

- **Facultative frequencies of sex**, where each offspring has a set probability of being born via sexual reproduction, otherwise they are born asexually. This is in contrast to other coalescent simulations, which assume the population is obligately sexual.

- **Tracking of paired and unpaired samples**, depending on whether one or both haplotypes are sampled from diploid individuals. The coalescent process differs for these types of samples when the frequency of sex is low [1].

- **Meiotic and mitotic crossover recombination**; the former acts on offspring undergoing sexual reproduction, while the latter acts on organisms irrespective of their reproductive mode.

- Implementation of **meiotic and mitotic gene conversion**. The former only acts during sexual reproduction, while the latter acts during both asexual and sexual reproduction.

- **Population structure** can be implemented via an island model (i.e., where the population is split into several subpopulations, each of equal size and with equal migration rates between them).

- **Heterogeneous frequencies of sex**, which can change either over time or over space.

1.1 Citing the program

The full program that considered multi-locus genealogies was published in:

Hartfield M., Wright S.I, and Agrawal A.F. (2018) Coalescence and linkage disequilibrium in facultatively sexual diploids. *Preprint available on bioRxiv, doi: 10.1101/232405.*

Single-locus theory that formed the initial program was outlined in:

Hartfield M., Wright S.I, and Agrawal A.F. (2016) Coalescent Times and Patterns of Genetic Diversity in Species with Facultative Sex: Effects of Gene Conversion, Population Structure, and Heterogeneity. *Genetics* **202**: 297–312.

2 Installing the program

The source code can be downloaded from GitHub (`https://github.com/MattHartfield/FacSexCoalescent`). The program uses routines found within the GNU Scientific Library (GSL) (`http://www.gnu.org/software/gsl/`) Since GSL is distributed under the GNU General Public License (`http://www.gnu.org/copyleft/gpl.html`), you must download and install it separately.

After placing the code in the directory of your choice, it can then be compiled using a C compiler such as GCC. An example compiler command would be:

```
gcc -o FacSexCoalescent -lm -lgsl -lgslcblas -I/usr/local/include
                -L/usr/local/lib FacSexCoalescent.c
```

Note the `-lm -lgsl -lgslcblas -I/usr/local/include -L/usr/local/lib` switches are used to link GSL routines to the source code. The `-I` and `-L` commands should be updated to point to where GSL is installed on the local machine.

3 Getting started

The basic simulation setup needs five parameters:

```
FacSexCoalescent <Population Size> <Paired Samples> <Single
        Samples> <Frequency of Sex> <Reps>
```

- **Population size** is the number of individuals in the entire diploid population. Hence if there are N individuals, then there are $2N$ haplotypes present. Also note that if the population is divided according to an island model, then N is the total population size *across all subpopulations*.
- **Paired Samples, Single Samples** are the number of sampled haplotypes where *both* or *only one* haplotype was taken from an individual. Note that each paired sample counts as two haplotype samples.
- **Frequency of sex** is a value that has to be greater than zero, and goes up to (and including) 1. It represents the probability that an offspring will be created via sexual reproduction. Note that if you switch on heterogeneity in frequencies of sex, the input values will be determined using other parameters, so a dummy variable is set here. See the section "Heterogeneous frequencies of sex" for more information.
- **Reps** is the number of independent replicates of the simulation.

For example, the command:

```
FacSexCoalescent 10000 5 0 0.1 100
```

will simulate genealogies in a population of $10,000$ individuals, for 5 paired samples (10 haplotypes in total), with a 10% frequency of sex, repeated 100 times.

In order for meaningful output to be produced, it is mandatory to specify either a mutation rate (using $-\mathbf{t}$), or indicate that gene trees are to be printed (using $-\mathbf{T}$). These options are described below.

4 Interpreting output

4.1 Adding mutations

Mutations can be added to each genealogy using the **−t** flag.

`FacSexCoalescent ... -t` θ

Theta, $\theta = 4N\mu$ is the scaled neutral mutation rate that determines how polymorphisms are placed along the genealogy. Note that this is the *only* input parameter that is scaled to $4N$; other parameters are scaled to $2N$.

If mutations are added to the genealogy, two sets of outputs will be produced:

1. A **seed file** ('Seed.dat'), containing the random seed used in the program. This can be set manually by prefacing the program with `GSL_RNG_SEED=X`, where X is the seed.

2. A **polymorphism file** ('Mutations'), containing 'reps' number of files for the polymorphism data per run. Each file is labelled '`Muts_X.dat`' where X is the run number.

If the 'Mutation' folder does not exist, it will be created by the program. These files are overwritten every time the program is executed, so make sure to back up any specific simulation data you wish to keep.

Each mutation file consists of a table, listing each polymorphism and which individuals carry it. Here's an example for a simulation consisting of 10 haplotypes, with six polymorphisms present.

```
0.033513 0 0 1 1 1 0 0 0 0 1
0.068214 0 0 0 0 0 0 0 1 0 0
0.153827 0 1 0 0 0 1 0 0 0 0
0.416249 0 0 0 0 0 0 1 0 0 0
0.551314 0 1 1 1 1 1 0 0 0 1
0.987333 1 0 0 0 0 0 0 1 1 0
```

The first column outlines the relative position of the polymorphism along the sampled genome, so it is a value between 0 and 1. The next ten columns denote which samples carry the ancestral ('0') or derived allele ('1') at that site. Haplotypes from paired alleles are listed first, followed by unpaired haplotypes. If there are multiple demes, then paired and unpaired samples for deme 1 are listed first, then paired/unpaired samples from deme 2, and so on.

4.2 Printing gene trees

Adding –**T** to the command line following the basic commands will print out the gene trees for each run. These are printed in Newick format [2] in time units of $2N$ generations. Note this is different from `ms` [3], which prints out trees in units of $4N_0$ generations (N_0 being the *sub*population size).

For non-recombining samples, trees are printed to a single file, 'Trees.dat'. Each line lists the Newick tree for each simulation run. As with other files, it is overwritten every time the program is run, so make sure you back up any particular outputs that you wish to keep. With crossover recombination and gene conversion, the output will be slightly different: more details are provided in the section "Gene trees with crossing-over or gene conversion".

4.3 Printing `ms`–style output to screen

It is also possible to print out results to `STDOUT`, in the same format as `ms` [3], by adding the –**P** flag. Here, for each simulation run, the tree is outputted (if the –T flag is added); the number of segregating sites, along with positions; and a table of polymorphisms.

Caution: the table here is transposed compared to those printed to the 'Mutation' folders. That is, the rows represent each polymorphism, and the columns are results per individual.

5 Advanced commands

The program also includes four other major options. These are (i) crossing over, both meiotic and mitotic; (ii) gene conversion, both meiotic and mitotic; (iii) population subdivision; and (iv) heterogeneity in frequencies of sex. These options can be added using the appropriate flags after setting the basic options. Be aware that some of the options require preceding options to be set. For example, any type of gene conversion first needs meiotic recombination to be set (even if the meiotic recombination rate is zero), and heterogeneity requires population structure to be specified (even if there is only one deme).

Caution: most of the compound parameters defined here are scaled by $2N$, for N the total population size. This can contrast with other simulations, such as `ms`, which instead scale parameters by $4N_0$.

5.1 Recombination via meiotic crossing-over

This is added using the –**r** flag:

`FacSexCoalescent ... -r` $2N\tilde{c}(L-1)$ L

In this scenario, if a sample was produced via sexual reproduction, then it can also be split into two samples via a finite-sites crossover recombination model [4, 5]. These samples remain paired; if sex is frequent then this pairing will quickly segregate into two unpaired samples [6].

Two options are specified. The first is the scaled crossover rate $2N\tilde{c}(L-1)$, for c the total probability that one of the breakpoints in the sample undergoes crossover recombination *during sexual reproduction*, and L the number of sites. Hence it is an 'absolute' rate; the net crossover probability across the population will also be dependent on the frequency of sex, σ [6]. The second input is the number of sites L. Note that mutations are still added via an infinite-sites model (i.e., mutation locations are specified on an absolute scale).

For example, say you are modelling a tract consisting of $2,001$ sites, with an average meiotic crossover recombination probability 10^{-8} between them, and the population is of size $N = 20,000$. Then the input $2N\tilde{c}(L-1) = 2 \times 20,000 \times 10^{-8} \times 2,000 = 0.8$.

5.2 Recombination via mitotic crossing-over

Mitotic crossing-over can be specified with the –**x** flag. It is necessary to first use the –r flag, even if $2N\tilde{c}(L-1)=0$ (so L can be specified first).

`FacSexCoalescent` ... `-r` $2N\tilde{c}(L-1)$ L `-x` $2N\tilde{c}_A(L-1)$

Mitotic crossing-over can act in all organisms, even if they have not undergone sexual reproduction.

5.3 Meiotic and mitotic gene conversion

These are defined using the –**c**, –**m** flags respectively. As with mitotic crossing over, it is necessary to use –r to first define crossover parameters and number of sites. Only one of these options can be set if the user wishes to consider just one type of conversion event.

`FacSexCoalescent` ... `-r` $2N\tilde{c}(L-1)$ L
`-c` $2Ng_S(L-1)$ λ_S `-m` $2Ng(L-1)$ λ

Two types of conversion events are considered, since they act during different reproductive stages:

- **Meiotic** gene conversion acts *only acts during sexual reproduction*, hence it is a weak force in primarily asexual organisms.
- **Mitotic** gene conversion acts during any cell division, so can occur during both sexual and asexual stages of reproduction.

g_S, g are the probabilities that either gene conversion event initiates at any breakpoint within the length of sampled genome; as with recombination, g_S is the probability of meiotic gene conversion *given sexual reproduction has occurred*. Hence the net meiotic gene conversion rate across the population each generation is also dependent on σ. λ_S, λ is the mean length of a meiotic, mitotic conversion event in number of sites.

For example, if a tract consists of 2,001 sites and a mitotic gene conversion event initiates with probability 5×10^{-7} in a population of size $N = 20,000$. Then the input $2Ng = 2 \times 20,000 \times 2 \times 10^{-7} \times 2,000 = 16$.

Gene conversion is resolved in different ways, depending on whether only one or multiple sites are defined.

- *Single-site gene conversion:* If only one site is defined, then only mitotic gene conversion affects genealogies. It acts on paired samples, causing one haplotype to coalesce into the other [1]. In this case g is the total probability of mitotic gene conversion *affecting* the site, as opposed to it being the probability of gene conversion initiation. It is not possible to define meiotic gene conversion on samples covering a single site. **Note:** in this case λ is not used but still needs to be inputted, hence a dummy value should be set.

- *Multi-site gene conversion:* If gene conversion (either meiotic or mitotic) acts on unpaired samples, then the sample is split into two that remain paired, as if it were undergoing a double recombination. Only mitotic conversion can act on paired samples. If it does act, then part of the 'recipient' sample coalesces into the 'donor' sample across the length of the conversion tract. If the mitotic gene conversion event is long enough to cover all extant material on the 'recipient' sample, then the paired sample is converted into a single sample, as is the case when a single site undergoes gene conversion. Multi-site gene conversion routines are based on those outlined in Wiuf and Hein (2000) [7].

Note 1: With obligate sex and multiple sites, the simulation can replicate previous gene conversion simulations *if only one type of conversion is simulated*. That is, one of g_S or g is set to zero.

Note 2: A key parameter with gene conversion across multiple sites is $(L - 1)/\lambda_i$, the number of breakpoints in units of average gene conversion length (for $\lambda_i = \lambda_S$ or λ) [7]. If this ratio is very low (i.e., if $\lambda_i \gg L$) then the assumption that no more than one conversion event per generation breaks down, and simulations

may become inaccurate. If this ratio is low for either meiotic or mitotic gene conversion, a warning is produced. See [6] for a fuller discussion of the probability and rate of different gene conversion events.

Gene trees with crossing-over or gene conversion

If either recombination or gene conversion is included, the printout of gene trees (using –T) is altered. In this case a gene tree is outputted for each segment of the genome that did not undergo recombination. Each tree is also prefaced with a number, identifying the number of sites that the tree covers.

For example, output for an obligately sexual population of 4 unpaired samples, with $L = 1001$ and meiotic crossover recombination rate $2N\tilde{c}(L-1) = 0.1$ might give the following output:

```
//
[34](0:3.514,(2:2.546,(3:0.562,1:0.562):1.984):0.968);
[967](0:4.057,(2:2.546,(3:0.562,1:0.562):1.984):1.511);
```

This output represents two different trees, the first covering the initial 34 sites, the second representing the rest of the sample.

In addition, trees are no longer outputted to a single file. Instead, a folder 'Trees' is created, containing files '`Trees_X.dat`' of the genealogies for each simulation run.

5.4 Island Model

A subdivided population can be modelled using the –**I** flag.

`FacSexCoalescent ... -I` d `[Paired 1, Single 1] ...`
`[Paired` d`, Single` d`]` $2Nm$

The first parameter d is the number of subpopulations. Each subpopulation is the same population size, so N has to be a multiple of d. $2d$ parameters are then listed, denoting the number of paired and unpaired samples in each deme. These need to add up to the total number of paired, unpaired samples listed in

the initial parameter set. The final parameter is the population-scaled migration rate between demes.

For example, the command line:

```
FacSexCoalescent 10000 8 4 0.01 5 1000 -I 4 2 1 3 2 2 0 1 1 1
```

splits a population of $N = 10,000$ into four demes, each of size 2,500. The first contains 2 paired samples and 1 unpaired sample; the second 3 paired sample and 2 unpaired samples; the third 2 paired samples and 0 unpaired samples; and the fourth 1 paired and 1 unpaired sample. This leads to 8 paired samples and 4 unpaired samples present across all demes. The scaled migration rate is $2Nm = 1$, with a frequency of sex equal to 0.01 in each deme.

5.5 Heterogeneous frequencies of sex

The simulation can also account for cases where the frequency of sex changes in either time or space. Options can be set using the **−H** flag, after first using the **-I** flag to set the number of subpopulations (this is necessary, even if $d = 1$).

`FacSexCoalescent ... -I ... -H <Type>` $[P_1, P_2]$
$[\sigma_{1,1}, \sigma_{1,2}] \ \ldots \ [\sigma_{d,1}, \sigma_{d,2}]$

Type is a parameter defining the kind of heterogeneity to be modelled. It takes one of three values:

- Type = 0 denotes **temporal heterogeneity**, where frequencies of sex fluctuate between two values over time.
- Type = 1 denotes **a stepwise change in sex**, where the frequency of sex instantaneously changes value at a specific time in the past.
- Type = 2 denotes **spatial changes in sex**. The frequencies of sex per deme are fixed over time, but differ between subpopulations.

It is possible to combine both spatial and temporal changes in sex by setting Type = 0, and then defining two frequencies of sex per deme as defined below.

If a different input value is used for 'Type' then the program exits with an error. Note that when the probabilities of sexual reproduction change, they do so in *all* demes simultaneously.

$[P_1,\ P_2]$ are two parameters for how the frequencies of sex change over time. They are differently defined depending on the type of heterogeneity being invoked:

- For Type = 0 (temporally heterogeneous frequencies of sex), $[P_1,\ P_2]$ are equal to the probabilities that (1) the population changes from low-sex to high-sex frequencies per generation, and (2) the population changes from high-sex to low-sex frequencies. Once the frequency of sex changes, the time until the next change (from e.g., low frequency to high frequency) is drawn from a Geometric distribution with mean $1/P$, then rescaled to $2N$ generations.

- For Type = 1 (stepwise change in sex), P_1 is the time in the past (in $2N$ generations) when the frequency of sex changes from one value to the other. P_2 is not used and should be set to zero.

- For Type = 2, $[P_1,\ P_2]$ are not used and should be set to zero.

Following this are $2d$ parameters (for d the number of subpopulations) defining the different rates of sex over time in each deme. Again, the exact input depends on the type of heterogeneity used:

- For Type = 0, each pair represents the *low* and *high* probabilities of sex per deme respectively. The first value always represent the low probability, so has to be the lesser of the two. It is possible here to set the low probability of sex to zero; if so then the high probability *has to be greater than zero.*

- For Type = 1, the pairs represent the probabilities of sex at the *current time* and *after the stepwise change* respectively. The first value does not need to be higher than the second in this case.

- For Type = 2, only the first of each pair needs to be defined as the constant probability of sex in that deme (the second value should be set to zero).

Examples of heterogeneous sex simulations

For a single population that changes sex over time, a typical input might be:

```
FacSexCoalescent 10000 5 0 1 1000 -I 1 5 0 0 -H 0 0.0001 0.0001
0.0001 1
```

This sets up a population of 5 paired samples, with the frequency of sex changing between 0.0001 and 1 every $1/0.0001 = 10,000$ generations on average. Note that the frequencies of sex defined following the –H flag overrides the baseline value, so it acts as a dummy input in this case.

If the frequency of sex is different over two demes, a setup might be:

```
FacSexCoalescent 10000 5 0 1 1000 -I 2 3 0 2 0 1 -H 2 0 0 0.0001 0
1 0
```

Here there are two demes, with 3 paired samples in one and 2 in the other. In the first deme the frequency of sex is equal to 0.0001, in the second it is equal to 1 (obligate sex).

Next is a command line for a stepwise change in sex at $0.25 \times (2N) = 5,000$ generations in the past. The current frequency of sex is 0.0001, which then switches to obligate sex following the single change.

```
FacSexCoalescent 10000 5 0 1 1000 -I 1 5 0 0 -H 1 0.25 0 0.0001 1
```

Finally, a complicated setup that combines temporal *and* spatial changes in sex over three demes. We set $N = 9,000$ here so the population is cleanly split between demes:

```
FacSexCoalescent 9000 5 0 1 1000 -I 3 2 0 2 0 0 1 1 -H 0 0.001
0.0001 0.5 1 0.0001 0.001 0.0001 1
```

There are three demes, two with two paired samples and a third with an unpaired sample. The migration rate $2Nm$ is set to 1. In deme 1 the probabilities of sex switch between 0.5 and 1; in the second deme the probabilities are 0.0001 and 0.001; in the third, the probabilities are 0.0001 and 1. On average, low-sex states switch to high-sex states every 1000 generations; high-sex switch to low-sex states every 10,000 generations.

6 Summary of options

These can also be displayed by running `FacSexCoalescent` without any input parameters. Note that it is mandatory to use one of the –t or –T flags.

- **–t** θ: Adds neutral mutations to the genealogy, occurring with rate $\theta = 4N\mu$.
- **–T**: Prints out genealogies (in Newick format).
- **–P**: Prints out results to `STDOUT` in a similar format to `ms`.
- **–r** $2N\tilde{c}(L-1)$ L: Introduces meiotic crossing over, with probability $\tilde{c}$ between sites, over a tract of length L.
- **–x** $2N\tilde{c}_A(L-1)$: Introduces mitotic crossing over, with probability $\tilde{c}_A$ between sites, over a tract of length L. Need to first use –r.
- **–c** $2Ng_S(L-1)$ λ_S: *Meiotic* gene conversion, with probability g_S that it affects the sample. Average conversion length λ_S. Need to first use –r.
- **–m** $2Ng(L-1)$ λ: *Mitotic* gene conversion, with probability g that it affects the sample. Average conversion length λ. Need to first use –r.
- **–I** d [Paired 1, Single 1] ... [Paired d, Single d] $2Nm$: Population is setup as an island model. d demes, then specify the samples taken from each deme. The scaled migration rate is $2Nm$.
- **–H** (Type) $[P_1,\ P_2]$ $[\sigma_{1,1},\ \sigma_{1,2}]$ ... $[\sigma_{d,1},\ \sigma_{d,2}]$: Introduce heterogeneity in the frequencies of sex over time or space. Need to first use –I to define population structure. 'Type' defines what heterogeneity is present: $[P_1,\ P_2]$ are parameters defining how frequencies of sex change over time: $[\sigma_{j,1},\ \sigma_{j,2}]$ are the frequencies of sex that the population switches between over time.

7 Acknowledgements and Contact Information

MH was supported by a Marie Curie International Outgoing Fellowship, grant number MC-IOF-622936 (project SEXSEL), and an European Research Council

grant (FP7/20072013, ERC Grant 311341) awarded to Thomas Bataillon, during the creation of this software. SIW, AFA were supported by Discovery Grants from the Natural Sciences and Engineering Research Council of Canada.

Comments, feedback, bug reports and suggestions for future versions should be sent to matthew.hartfield@birc.au.dk.

References

[1] M. Hartfield, S. I. Wright, and A. F. Agrawal, "Coalescent times and patterns of genetic diversity in species with facultative sex: Effects of gene conversion, population structure, and heterogeneity," *Genetics*, vol. 202, no. 1, pp. 297–312, 2016. doi: 10.1534/genetics.115.178004.

[2] J. Felsenstein, *Inferring phylogenies*, vol. 2. Sinauer Associates Sunderland, 2004.

[3] R. R. Hudson, "Generating samples under a Wright–Fisher neutral model of genetic variation," *Bioinformatics*, vol. 18, no. 2, pp. 337–338, 2002. doi: 10.1093/bioinformatics/18.2.337.

[4] R. R. Hudson, "Properties of a neutral allele model with intragenic recombination," *Theor. Popul. Biol.*, vol. 23, no. 2, pp. 183–201, 1983. doi: 10.1016/0040-5809(83)90013-8.

[5] J. Hein, M. H. Schierup, and C. Wiuf, *Gene Genealogies, Variation and Evolution: A Primer in Coalescent Theory*. Oxford: Oxford University Press, 2005.

[6] M. Hartfield, S. Wright, and A. Agrawal, "Coalescence and linkage disequilibrium in facultatively sexual diploids," *bioRxiv*, 2018. doi: 10.1101/232405.

[7] C. Wiuf and J. Hein, "The coalescent with gene conversion," *Genetics*, vol. 155, no. 1, pp. 451–462, 2000.