Contents

1 Wright-Fisher Exact Solver

Wright-Fisher Exact Solver (`WFES`) implements a variety of exact calculations with the Wright-Fisher model. Unlike other approaches, `WFES` does not use simulations or strong simplifying assumptions. `WFES` benefits from high-performance linear algebra techniques, making it possible to compute exact quantities for biologically realistic population sizes. The following document details the usage of the `WFES` applications.

- Executables gives a brief description of executables and types of calculations they perform.
- Computation explains what computations are performed by the `WFES` applications.
- Additional features provides extra details about some aspects of computation

and implementation of models.

- Input format gives details on the input format used by the applications.
- Output format explains the output format used by the applications.
- Usage provides detailed parameter tables and usage information.

2 Executables

The package consists of several executables, each implementing a different type of a Wright-Fisher model (WF). Each executable has flags to specify the model parameters and the calculation to be performed.

Most of the executables have additional modes, specifying what type of calculation is to be performed. These flags mostly concern the configuration of absorbing states in the model.

2.1 `wfes_single`

`wfes_single` implements the standard Wright-Fisher model of a single population. It has the following flags:

- `--absorption` mode assumes that absorption is possible at extinction *and* fixation boundaries. The calculations assume that the population starts with one or more copies of the allele (see integration for details). This calculates the following statistics:
  - P_{ext} - probability of extinction
  - P_{fix} - probability of fixation
  - T_{ext} - expected number of generations before extinction
  - T_{fix} - expected number of generations before fixation
- `--fixation` mode assumes that the extinction boundary is transient, and *only* the fixation boundary is absorbing. The calculations assume that the population starts with zero copies of the allele. Following statistics are calculated:
  - $T_{b\ fix}$ - expected number of generations between two fixation events
  - T_{std} - standard deviation of $T_{b\ fix}$
  - R - rate of substitutions ($1/T_{b\ fix}$)
- `--fundamental` mode calculates the entire fundamental matrix of the Wright-Fisher model. There is no assumption about the starting number of alleles. Note that this mode is slow for large matrices ($N > 1000$).
  - N - the fundamental matrix of the WF model. This is not produce output by default - use `--output-N` to direct output to file or stdout.
  - V - the variance of the fundamental matrix. This is not produce output by default - use `--output-V` to direct output to file or stdout.
- `--equilibrium` mode calculates the equilibrium distribution of allele frequencies. Both boundaries are non-absorbing (this is required for the existence of the equilibrium distribution).
  - E - the equilibrium distribution of allele frequencies. This is not produce output by default - use `--output-E` to direct output to file or stdout.
- `--allele-age` mode calculates moments of the allele age given a current allele frequency. Both extinction and fixation boundaries are absorbing. The calculations assume that the population starts with one or more copies of the allele (see integration for details).
  - $E(A)$ - the expectation of the allele age
  - $S(A)$ - the standard deviation of allele age

2.2 `wfes_switching`

`wfes_switching` implements a time-heterogeneous extension of the Wright-Fisher model. It is possible to switch between different parameter regimes - for example different population sizes, selection parameters, or mutation rates. We refer to each parameter regime as "component". For example, an absorbing model of oscialting population size ($N_1 = 1000, N_2 = 2000$) has two components (corresponding to each population) and $(2N_1 - 1) + (2N_2 - 1) = 7998$ states. The switching between components is parametrized with the initial probability distribution (p), and the rate of switching from one component to the next (r). The following modes are implemented:

- `--absorption` mode allows both extinction and fixation boundaries to be absorbing. The following satistics are calculated:
  - P_{ext} - probability of extinction
  - P_{fix} - probability of fixation
  - T_{ext} - expected number of generations before extinction
  - T_{fix} - expected number of generations before fixation
- `--fixation` mode assumes that the extinction boundary in non-absorbing. Following statistics are calculated:
  - $t_{b\ fix}$ - expected number of generations between two fixation events
  - r - rate of substitutions ($1/t_{b\ fix}$)

2.3 `wfes_sweep`

`wfes_sweep` implements a type of a swithching model with two parameter regimes. The first model is non-absorbing (both extinction and fixation boundaries are transient), and the second model is fixation-only. This is a model of standing genetic variation with pre-adaptive and adaptive components.

There is currently one mode:

- `--fixation` mode assumes that extinctions are non-absorbing. We output following statistics:
  - $t_{b\ fix}$ - expected number of generations between two fixation events
  - r - rate of substitutions ($1/t_{b\ fix}$)

2.4 `wfafle`

`wfafle` calculates the expected allele frequency distribution for a given piece-wise demographic history. It uses an equilibrium distribution to initiate the calculation, and then iterates forward in time by fast matrix-vector multiplications. It is also possible to start from a given allele frequency distribution. Details on calculation in [section 2](#expected-allele-frequencies-with-a-demographic), and details on usage in [section 6](#=wfafle=-usage).

2.5 `test_wfes`

This is the test harness for `wfes` models. It is based on the catch framework. Run with `test_wfes --list-tests` to see available test. These are mostly end-to-end tests, confirming optimized implementations of WF matrix building functions.

3 Computation

This section explains computations performed in `WFES` in detail.

The main feature of `WFES` is to compute rows of the fundamental matrix of the Wright-Fisher model. From the fundamental matrix, many properties of interest can be derived. We first describe the calculation applied in `wfes_single`, for a standard WF model.

3.1 Wright-Fisher model

The Wright-Fisher model describes a single bi-allelic locus in a population of fixed size. We denote a as the ancestral allele, and A as the derived, or focal allele. The organisms are diploid, so the total number of chromosomes in a population size N is $2N$. Given i copies of derived allele A at time t, the probability of having j copies in the next generation is:

$$P_{i,j}(t+1) = \binom{2N}{j} \psi_i^j (1-\psi_i)^{2N-j} \tag{1}$$

Above, ψ_i is the binomial sampling probability for the number of individuals in the next generation. In the simple case of no mutation or selection, ψ_i only depends on the current number of copies, $\psi_i = \frac{1}{2N}$. One way to parametrize the model with mutation and selection is:

$$\psi_i = \frac{[w_{AA}p^2 + w_{Aa}q](1-\mu_{A\to a}) + [w_{Aa}pq + w_{aa}q^2]\mu_{a\to A}}{w_{AA}p^2 + 2w_{Aa}pq + w_{aa}q^2} \tag{2}$$

Above, $w_{..}$ is the selection coefficient for a particular genotype, $\mu_{A\to a}$ is the backward mutation rate, $\mu_{a\to A}$ is the forward mutation rate. Variables p and q are allele frequencies of A and a respectively: $p = i/2N$, $q = 1-p$. The denominator is the average fitness of the population, $\bar{w}$.

Equation 2 can be parametrized in an arbitrary manner. We follow Kimura [1964], and assign the following selection coefficients to the genotypes:

Genotype	Fitness
AA	$1+s$
Aa	$1+sh$
aa	1

Above $h \in [0,1]$ is the dominance coefficient. With the above formulation, (2) simplifies to:

$$\psi_i = \frac{[(1+s)p^2 + (1+sh)q](1-\mu_{A\to a}) + [(1+sh)pq + q^2]\mu_{a\to A}}{(1+s)p^2 + 2(1+sh)pq + q^2} \tag{3}$$

3.2 Fundamental matrix calculation

Equation 1 yields a discrete finitie-state Markov chain, with time scale in Wright-Fisher generations. State $i = 0$ corresponds to extinction of A, and $i = 2N$ is fixation of A. The model has $2N+1$ states, where $i = 0$ and $i = 2N$ are absorbing, and the rest are transient. The transition probability matrix $\mathbf{P}$ is $(2N+1) \times (2N+1)$. The transition probability matrix can be re-ordered to group the transient-to-transient entries ($\mathbf{Q}$) and transient-to-absorbing ($\mathbf{R}$) entries:

$$\mathbf{P} = \begin{pmatrix} \mathbf{Q} & \mathbf{R} \\ \mathbf{0} & \mathbf{I_2} \end{pmatrix} \tag{4}$$

With two absorbing states, $\mathbf{0}$ is a 2×2 matrix of zeros, $\mathbf{I_2}$ is a 2×2 identity matrix, and $\mathbf{Q}$ is a $(2N-1) \times (2N-1)$ matrix. For any absorbing Markov chain, there exists a fundamental matrix $\mathbf{N}$:

$$\mathbf{N} = \sum_{k=0}^{\infty} \mathbf{Q}^k = (\mathbf{I} - \mathbf{Q})^{-1} \tag{5}$$

Each entry of $\mathbf{N}_{ij}$ is the expected number of generations spent with j copies, given that we started with i copies. Knowing the entries of $\mathbf{N}$ allows to write down many useful absorption properties of the Markov chain. For example, probability of absorbing in state k, conditional on starting with i copies is found as the $(i,k)^{\text{th}\hat{}}$ entry of:

$$\mathbf{B} = \mathbf{NR} \tag{6}$$

We can use $\mathbf{B}$ to find the expected number of generations in state j, conditional on starting in i and absorbing in k:

$$\mathbf{E}_{i,k}(j) = \frac{\mathbf{B}_{j,k}}{\mathbf{B}_{i,k}} \mathbf{N}_{ij} \tag{7}$$

The conditional time to absorption in state k is then:

$$T_{abs}(k) = \sum_{j=1}^{2N-1} \mathbf{E}_{i,k}(j) \tag{8}$$

And times to extinction and fixation are:

$$\begin{aligned} T_{ext} &= T_{abs}(0) \\ T_{fix} &= T_{abs}(2N) \end{aligned} \tag{9}$$

These are the properties calculated in `wfes_single` in the `--absorption` mode. See `wfes_single` usage for more details.

3.2.1 Example

```
./wfes_single --absorption -N 1000
```

3.3 Solving sparse systems

Solving for the entire matrix $\mathbf{N}$ is expensive for large population size. However, since $\mathbf{N}_{i,j}$ expresses number of generations spent in a state j *conditional* on starting in i, we can simplify the calculation by explicitly conditioning on i. For example if we assume that allele A start with one copy ($i = 1$), then only the first row, $\mathbf{N}_{i,\cdot}$ is of interest. We can generalize this, by assuming a finite forward mutation rate v. In this case, for small $4Nv < 1$, there is a non-zero probability that with $i \leq 1$. However, this probability vanishes quickly with increasing i, and we then only require several rows of $\mathbf{N}$. See more details in integration.

For a startring number of alleles i, we find the $i^{\text{th}\hat{}}$ row of $\mathbf{N}$:

$$(\mathbf{I} - \mathbf{Q})^T \mathbf{N}_{i,\cdot} = \mathbf{I}_{i,\cdot} \tag{10}$$

where $\mathbf{I}_{i,\cdot}$ is the $i^{\text{th}\hat{}}$ column of a $(2N-1) \times (2N-1)$ identity matrix.

This system can be solved by LU decomposition of $(\mathbf{I}-\mathbf{Q})^T$. Once the decomposition is known, we can solve for different right-hand sides of the equation, such as when $i \geq 1$.

To find matrix $\mathbf{B}$, we solve:

$$(\mathbf{I}-\mathbf{Q})\mathbf{B}_{\cdot,0} = \mathbf{I}_{\cdot,0} \tag{11}$$

where $\mathbf{B}_{\cdot,0}$ is the column of $\mathbf{B}$ corresponding to $i = 0$ extinction. Since we have tow absorbing states, we can compute:

$$B_{\cdot,2N} = \mathbf{1} - \mathbf{B}_{\cdot,0} \tag{12}$$

The LU decomposition and solution is performed with `MKL PARDISO` routines. Parameters and settings for the `MKL PARDISO` calls can be found in the source code.

3.4 Entire fundamental matrix

If the entire fundamental matrix is required, it can be calculated with `wfes_single` in the `--fundamental` mode. See `--output-N` and `--output-V` options in `wfes_single` usage.

3.4.1 Example

```
# Note - this is slow since the _entire_ fundemantal matrix is calculated
wfes_single --fundamental -N 1000 \\
--output-N fundamental.csv --output-V f_variance.csv
```

3.5 Fixation only

The calculation as stated in the previous section applies to the `--absorbing` mode of `wfes_single` - where both extinction and fixation states are absorbing. The other possible mode for the computation in `--fixation` - where only the fixation state ($i = 2N$) is absorbing, and the extinction state ($i = 0$) is transient. In this case, matrix $\mathbf{Q}$ in equation 4 is $2N \times 2N$.

If the extinction state is transient, it can be entered and left many times without terminating the Markov chain. This mode makes it easy to calculate $T_{b\ fix}$ - time between fixations - the total time it takes for a new allele to reach fixation (with the possibility of several extinctions along the way). More details on this calcualtion an applications can be found in [de Koning and de Sanctis 2018].

The time between fixations, $T_{b\ fix}$ is calculated in a similat way as T_{fix} for the model with two absorbing states (eq. 6, 7, 8). However, since there is only one ansorbing state, no re-conditioning is required (eq. 7). Then the $T_{b\ fix}$ is simply:

$$T_{b\ fix} = \sum_{j=1}^{2N-1} N_{0,j} \tag{13}$$

An advantage of this calculation is that we can safely assume that allele A starts in $i = 0$ copies. Then the integration over the starting number of copies is not necessary, since it is explicitly included in the model as transitions from $i = 0$ to $i > 0$ copies. This then means that we only need to find a single, $0^{\text{th}\hat{}}$ row of the fundamental matrix.

3.5.1 Example

```
wfes_single --fixation -N 1000
```

3.6 Variance calculation

Calculating the variance of the time spent in each state is of interest. It can be found as:

$$\mathbf{N}_{var} = \mathbf{N}(2\mathbf{N}_{dg} - \mathbf{I}) - \mathbf{N}_{sq} \tag{14}$$

where $\mathbf{N}_{dg}$ is the matrix containing the diagonal of $\mathbf{N}$, and $\mathbf{N}_{sq}$ is $\mathbf{N}$ element-wise squared.

If the `--output-V` option is used in the `--fundamental` mode, the entire $\mathbf{N}_{var}$ matrix will be calculated.

In the `--fixation` mode, the standard deviation of $T_{b\ fix}$ is calculated from the fisrt row of $\mathbf{N}$ in equation 13:

$$T_{std} = \sqrt{(2\mathbf{N}_2 - \mathbf{N}_1) - (\mathbf{N}_2)^2} \tag{15}$$

where $\mathbf{N}_1$ and $\mathbf{N}_2$ are found by solving:

$$\begin{aligned} (\mathbf{I} - \mathbf{Q})^T \mathbf{N}_1 &= \mathbf{I}_0 \\ (\mathbf{I} - \mathbf{Q})^T \mathbf{N}_2 &= \mathbf{N}_1 \end{aligned} \tag{16}$$

3.7 Equilibrium allele frequencies

The equilibrium distribution of allele frequencies is one of the key properties of the Wright-Fisher model. We use the method described by Paige, Styan, and Watcher [1975] to solve for the equilibrium distribution (see also Harrod and Plemmons 1984) of a non-absorbing Markov chain. The equilibrium distribution of the Markov chain is defined as vector π , such that $\pi\mathbf{P} = \pi$. This can be expressed in matrix form as:

$$\mathbf{\Pi P} = \mathbf{P} \tag{17}$$

where $\mathbf{\Pi}$ is a $n \times n$ matrix with π in each row. This can be re-written as:

$$\mathbf{\Pi}(\mathbf{P} - \mathbf{I_n}) = \mathbf{0_n} \tag{18}$$

We also have the constraint that $\sum_i \pi_i = 1$, which can be enforced by setting the last columns of $(\mathbf{P} - \mathbf{I_n})$ and $\mathbf{0_n}$, to $e_n = (1, 1, \ldots, 1)^T$. We use the notation $r(A)$ to denote that we set the last column of A to e_n.

$$\begin{aligned} \mathbf{\Pi} r(\mathbf{P} - \mathbf{I_n}) &= r(\mathbf{0_n}) \\ r(\mathbf{P} - \mathbf{I_n})^T \mathbf{\Pi}^T &= r(\mathbf{0_n})^T \end{aligned} \tag{19}$$

We only require a single row of $\mathbf{\Pi}$. Therefore, we can solve for any row $\Pi_{\cdot,x}$:

$$r(\mathbf{P} - \mathbf{I_n})^T (\mathbf{\Pi}^T)_{\cdot,x} = (r(\mathbf{0_n})^T)_{\cdot,x} \tag{20}$$

This equation is solved with the LU decomposition approach.

Note that the matrix P is a $2N + 1$ matrix, since the absorbing states are included. This means that we require that forward and backward mutation rates are non-zero. In case where $\mu_{A \to a} = 0$ or $\mu_{a \to A} = 0$, the matrix P becomes absorbing, and the equilibrium distribution does not exist.

This calculation is perfromed by `wfes_single` in the `--equilibrium` mode. See `wfes_single` usage for more details.

3.7.1 Example

```
wfes_single --equilibrium -N 1000 --output-E equilibrium.csv
```

3.8 Allele age

For details on the allele age calculation, the user is directed to [de Sanctis, Krukov, de Koning 2017]. Briefly, the paper describes a method to find moments of the allele age distribution given an observed number of copies in the WF model. The moments are calculated in an approach similar to those described above.

The calculation is performed by `wfes_single` in the `--allele-age` mode. The observed number of copies is set via the `--observed-copies/-x` parameter. See `wfes_single` usage for more details.

3.8.1 Example

```
wfes_single --allele-age -N 1000 -x 10
```

3.9 Switching models

`wfes_switching` implements an extended time-heterogeneous Wright-Fisher model. The classical WF model describes a single population of constant size. However, this assumption is rarely met in nature. Likewise, classical WF assumes that the rest of the parameters (selection, mutation) are time-invariant. In this section we describe an extension to the Wright-Fisher model with time-variable parameters. We combine a finite set of WF models in a joint Markov-modulated switching process. The switching process assigns a probability of switching between WF component models with different parameters. Each WF component model can have its own population size, selection coefficient, and mutation rate. Further, `wfes_sweep` combines absorbing and non-absorbing models.

Let $W_1, \ldots, W_n$ represent a finite list of distinct Wright-Fisher components with its own parameter set θ_i. Each component is a full Wright-Fisher Markov model. We also have transition probabilities $r_{x \to y}$ of switching from W_x to W_y at any time. Each component W_i has a transition probability matrix $\mathbf{P}_{(i)}$, which is written in canonical form as (eq 4):

$$\mathbf{P}_{(i)} = \begin{pmatrix} \mathbf{Q}_{(i)} & \mathbf{R}_{(i)} \\ \mathbf{0} & \mathbf{I} \end{pmatrix} \tag{21}$$

We want to describe the join process of switching between $W_1, \ldots, W_n$. We write the canonical form the switching process transition probability matrix as:

$$\mathbf{P} = \left(\begin{array}{cccc|c} \mathbf{Q}_{(1)} & \mathbf{\Gamma}_{(1,2)} & \cdots & \mathbf{\Gamma}_{(1,m)} & \mathbf{R}_{(1)} \\ \mathbf{\Gamma}_{(2,1)} & \mathbf{Q}_{(2)} & \cdots & \mathbf{\Gamma}_{(2,m)} & \mathbf{R}_{(2)} \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ \mathbf{\Gamma}_{(n,1)} & \mathbf{\Gamma}_{(n,2)} & \cdots & \mathbf{Q}_{(n)} & \mathbf{R}_{(n)} \\ \hline \mathbf{0} & \mathbf{0} & \cdots & \mathbf{0} & \mathbf{I} \end{array}\right) \tag{22}$$

The $\mathbf{\Gamma}_{x,y}$ matrix defines a matrix of switching from WF component W_x to WF component W_y. The dimensions of the matrix are $(2N_x - 1) \times (2N_y - 1)$ if both W_x and W_y have two absorbing states (or $2N_x \times 2N_y$ if W_x and W_y each have one absorbing state).

The entries of $\mathbf{\Gamma}_{x,y}$ are defined as Wright-Fisher transition probabilities given current i state is in process W_x and next state (j) is in process W_y:

$$\mathbf{\Gamma}_{x,y}(i,j) = \alpha_{x,y}\binom{2N_y}{j}(\psi_{y,i})^j(1-\psi_{y,i})^{2N_y-j} \tag{23}$$

where $\alpha_{x,y}$ is a switching rate between W_x and W_y.

This formulation essentially matches the frequencies of allele A between different component models. For example if $N_x = 100$ and $N_y = 200$ then $i_x = 10$ would correspond to $i_y = 20$. Note that $\mathbf{\Gamma}_{x,y}(i,\cdot)$ describes a full Wright-Fisher generation, so we are transofrming the entire distribution from W_x into W_y.

We additionally use a parameter $p_1,\ldots,p_n$, denoting the probability of starting in each of the components $W_1,\ldots,W_n$. This parameter is most relevant with $\alpha_{x,y}$ imposing a non-reversible switching process.

The calculations on this extended model are similar to those for the simple Wright-Fisher model, since we are still dealing with an finite state absorbing Markov chain. The main difference is that we now deal with extinction and fixation events in each of the component models. To calculate the overall statistic of a model, we weight each component with the probability of starting within each component, p_i. Consider the probability of fixation (probability of extinction is analogous):

$$P_{fix} = \sum_{i=1}^{n} p_i B_{0_i,2N} \tag{24}$$

where 0_i is the $0^{\text{th}\,\hat{}}$ state of the $i^{\text{th}\,\hat{}}$ component.

These calculations are implemented in `wfes_switching`, with `--absorption` and `--fixation` modes. Matrix parameter α is controlled by `--switching/-r` command line flag. Detailed usage information is found in section 6.

3.9.1 Example

```
# Reversible model
wfes_switching --absorption -N 100,200
# Non-reversible model
wfes_switching --absorption -N 100,200 -r '0.99,0.01;0,1' -p '1,0'
```

3.10 Model of standing genetic variation

`wfes_sweep` implements a model of selection with standing genetic variation. It is a special case of a time-heterogeneous model with two components. The first, pre-adaptive, component is a non-absorbing model with a deleterious or neutral $s_d \leq 0$. This is the model the Markov chain starts in, and the process stays in the first component for an average of τ generations. The process then switches into the second, adaptive, component with $s > 0$. The second component allows fixations with $2N$ copies. Note that the population size in both components is the same.

This model intends to capture the accumulation of standing genetic variation, followed by the onset of positive selection and eventual fixation.

The parameter τ is specified through the rate of transition out of the pre-adaptive component $\lambda = 1/\tau$. The calculations are performed in `wfes_sweep` in `--fixation` mode. See `wfes_sweep` usage section for more detail.

3.10.1 Example

```
wfes_sweep --fixation -N 1000 -s 0,0.001 -l 1e-3
```

3.11 Expected allele frequencies with a demographic

`wfafle` calculates the expected allele distribution given a piece-wise constant demographic history.

The calculation is performed according to the following procedure. Consider a piecewise constant demographic history with population sizes $N_1, N_2, \ldots, N_k$, where each population size epoch lasts for $G_1, G_2, \ldots, G_k$ generations.

1. Acquire the initial probability distribution over allele frequencies for population size N_1. This is done in one of the two ways:
   - Solve for the equilibrium allele frequency distribution using the Paige method, or
   - Read an initial allele frequency distribution from file (specified with `--initial` option)
2. Construct the Wright-Fisher transition probability matrix $\mathbf{Q}_i$ for population size N_i. Multiply the current allele frequency distribution d_i by $\mathbf{Q}_i$ exactly G_i times.
3. Construct a switching transition probability matrix $\mathbf{\Gamma}_{i \to i+1}$. This transition probability matrix incorporates the difference in population size, and other parameters (such as selection). Multiply current d_i by $\mathbf{\Gamma}_{i \to i+1}$ once.
4. Repeat steps 2 and 3 until the final epoch k is reached.

The calculation is feasible since the sparse vector-matrix multiplication in step 2 is relatively cheap.

Detailed usage information is found in `wfafle` usage.

3.11.1 Example

```
wfafle -N 1000,100,10000 -G 200,50,300
```

4 Additional features

4.1 Integration

WFES relies on assumptions about the starting number of copies of an allele in the population. By avoiding the need to calculate the entire fundamental matrix, these assumptions drastically simplify calculations.

4.1.1 Absorbing extinction boundary

Consider a model with two absorbing states - extinction and fixation (`--absorption` mode). The initial configuration can not be 0 copies of allele A, since that is an absorbing state. Thus, the starting number of copies is $i \geq 1$. In the simplest case, we can consider $i = 1$. In this situation, we only require a single row of the fundamental matrix. Alternatively, we can integrate over i by the probability of starting with each number of copies. The conditional probability of starting with i copies of the allele can be derived from the transition probability matrix $\mathbf{P}$:

$$\mathbf{P}_i = \frac{\mathbf{P}_{0,i}}{1 - \mathbf{P}_{0,0}} \quad j \geq 1 \tag{25}$$

The entries of vector $\mathbf{P}_i$ quickly approach zero, and we ignore them below some ϵ. This parameter is set by option `-c,--integration-cutoff`, which is $\epsilon = 10^{-10}$ by default. If option `--integration-cutoff` ≤ 0, no integration is performed, and we assume starting in the smallest starting state ($i = 1$).

We solve equation 10 for any row where $\mathbf{P}_i > \epsilon$, which amounts to several rows with small θ. Since the LU decomposition is the most computationally costly operation, the addition of several rows to the system has minor performance impact.

An alternative approach is to specify the number of copies explicitly. This is done with option `-p,--starting-copies`. In this case, only the p^{th} row of the fundamental matrix is found. Note that `--starting-copies\=1` and `--integration-cutoff=-1` are equivalent.

4.1.2 Non-absorbing extinction boundary

In the case where the extinction boundary is not absorbing (`--fixation` mode), the model start with $i = 0$ copies of A. In this case, it is not necessary to integrate over the starting number of copies exlplicitly - it is automatically included in the model. For the `--fixation` mode, integration flags are ignored.

4.2 Tail truncation

Each row of the WF transition probability matrix is a binomial distribution. To optimize the sparsity of the matrix as a whole, we can consider only the region that contains $1 - \alpha$ mass of the distribution on each row. This truncates the tails of the binomial distribution, significantly increasing the sparsity of the system. The truncation option `-a,--alpha` is set to 1×10^{-20} by default. Increasing the value of this parameter will result in faster run times at a sacrifice of precision. In our tests, $\alpha \leq 10^{-15}$ produced results indistinguishable from $\alpha = 0$. With $\alpha = 10^{-5}$, relative error did not exceed 0.03% with $N = 5 \times 10^4$.

4.3 Recurrent mutation

By default, all models in `WFES` allow recurrent mutation during allele segregation. However, this can be turned off with the `--no-recurrent-mu`. In this case, the mutation rates u and v describe the rates of only new mutations ($P_{0 \to i}$). No mutations are allwed once there is one or more alleles. Currently, this model is only implemented in `wfes_single`.

4.4 Parameter checks

Before the program executes, the input parameters will be checked for validity. The checks can be skipped by specifying the `--force` flag. Currently, the following checks are implemented:

- Population size must below 5×10^5 ($N \in [1, 5 \times 10^5]$) - calculations for larger population sizes require excessive amounts of time.
- Selection coefficient must be above -1 ($s \in (-1, 1)$). With the current parameterization (eq 3), selection coefficients below -1 do not make sense. Positive selection coefficients above 1 can also be problematic, but are currently allowed.
- Mutation rate between 0 and $1/4N$ ($\mu \in (0, 1/4N]$).

- If the mutation rate is above $1/4N$, then $\theta := 4N\mu > 1$. With higher values of θ, fixation have a conventional meaning. In general, we are calculating statistics concerning first hitting time, which is not the same as fixation.
- For models where both extinction and fixation boundaries are absorbing, mutation rates can be equal to 0. However, if the extinction boundary is non-absorbing (`--fixation` mode), the forward mutation rate can not be 0. Otherwise, $\mu_{a\to A} = 0$ implies an absorbing extinction boundary, which violates model assumptions. Likewise, if neither of the boundaries are absorbing (`--equilibrium` mode) both forward and mutation rates should be above 0.

5 Input format

Most of the arguments to `wfes` executables are passed on the command line. Parameters can be boolean, single numeric types (`int` or `float`), vectors of numeric types, or matrices of numeric types. There is also the `path` type, specifying a file location.

5.1 Boolean flags

Boolean flags are specified as `--flag` on the command line. They do not require an argument.

5.2 Single numeric types

Single numeric types can be `int` for integers and `float` for rationals. Internally, they are stored as `long long int` and `double`.

These types are specified on the command line as numbers, optionally with an equal sign:

```
--pop-size 10
--pop-size=10
--selection 1e-2
--selection 0.01
--selection=1e-2
--selection=0.01
```

For short option names, equal signs are not allowed:

```
-N 10
-s 1e-2
-s 0.01
```

If necessary, the values may be included in single quotes:

```
--pop-size '10'
-N '10'
```

Note that integers will not be parsed from scientific notation. Fractional notation is not supported for rationals.

5.3 Numeric vectors

Numeric vectors of lengths `k` are of type `int[k]` and `float[k]`.

On the command line, vectors are specified as comma-separated values. They can be optionally quoted:

```
--pop-sizes 100,200
--pop-sizes '100,200'
--pop-sizes=100,200
--pop-sizes='100,200'
-N 100,200
-N '100,200'
```

Parsing rules for each element of a vector are the same as for single numeric types.

5.4 Numeric matrices

Numeric matrices with `k` rows and `l` columns have types `int[k][l]` and `float[k][l]`.

On the command line, entries of the matrix on a row aew specified as a comma-separated values. Rows are divided by semi-colons. Matrix arguments *have* to be quoted in order not to clash with shell symbols. The following matrix:

$$\begin{bmatrix} 0.4 & 0.6 \\ 0.1 & 0.9 \end{bmatrix} \tag{26}$$

is represented on the command line as:

```
--switching='0.4,0.6;0.1,0.9'
--switching '0.4,0.6;0.1,0.9'
-r '0.4,0.6;0.1,0.9'
```

5.5 Path type

Paths are specified as strings on the command line. They are used to specify output paths for matrices and vectors.

```
--output-I i.csv
```

More details on the output in the output format section.

6 Output format

6.1 Common output

The default output from each executable is in the long format. The long format includes a separate named line for each input parameter and calculated statistic. The `--csv` flag can be specified to output values as a single comma-separated line. The order of output is preserved. A new file will be created if an output does not exist. Existing files will be overwritten.

6.1.1 Example

```
> wfes_single --absorption -N 1000
N = 1000
s = 0.0000000000e+00
h = 5.0000000000e-01
u = 1.0000000000e-09
v = 1.0000000000e-09
a = 1.0000000000e-20
P_ext = 9.9949998695e-01
P_fix = 5.0001305912e-04
T_ext = 1.4564579449e+01
T_fix = 3.9957557932e+03

# Lines wrapped
> wfes_single --absorption -N 1000 --csv
1000, 0.0000000000e+00, 5.0000000000e-01, 1.0000000000e-09,
1.0000000000e-09, 1.0000000000e-20, 9.9949998695e-01,
5.0001305912e-04, 1.4564579449e+01,  3.9957557932e+03
```

For vectorized outputs, each vector is printed on a single line in the long output. In `csv` format, the vector values are concatenated (in the same order).

```
> wfes_switching --absorption -N 1000,2000
N = 1000, 2000
s = 0.0000000000e+00, 0.0000000000e+00
h = 5.0000000000e-01, 5.0000000000e-01
u = 1.0000000000e-09, 1.0000000000e-09
v = 1.0000000000e-09, 1.0000000000e-09
p = 5.0000000000e-01, 5.0000000000e-01
a = 1.0000000000e-20
P_ext = 9.9962498690e-01
P_fix = 3.7501309543e-04
T_ext = 1.5099382608e+01
T_fix = 5.3286017483e+03

# Lines wrapped
> wfes_switching --absorption -N 1000,2000 --csv
1000, 2000, 0.0000000000e+00, 0.0000000000e+00,
5.0000000000e-01, 5.0000000000e-01, 1.0000000000e-09,
1.0000000000e-09, 1.0000000000e-09, 1.0000000000e-09,
5.0000000000e-01, 5.0000000000e-01, 1.0000000000e-20,
9.9962498690e-01, 3.7501309543e-04, 1.5099382608e+01,
5.3286017483e+03
```

6.2 Matrix output

There are several output options for matrices and vectors, all starting with `--output-`. These direct output of matrices and vectors into files. The files will be output in a `.csv` format. Vectors are output as line vectors.

For any such flag, specifying `--output-X stdout` will direct the output to standard output. The vector/matrix output will precede parameter output.

6.2.1 Example

```
> wfes_single --equilibrium -N 10
N = 10
s = 0.0000000000e+00
h = 5.0000000000e-01
u = 1.0000000000e-09
v = 1.0000000000e-09
a = 1.0000000000e-20

> wfes_single --equilibrium -N 10 --output-E equilibrium.csv
N = 10
s = 0.0000000000e+00
h = 5.0000000000e-01
u = 1.0000000000e-09
v = 1.0000000000e-09
a = 1.0000000000e-20

> ls
equibrium.csv

# Lines wrapped
> head equlibrium.csv
0.499999931260253, 2.31983343180376e-08, 1.05874638827596e-08,
7.82998730182177e-09, 6.20531885243497e-09, 5.2869035051936e-09,
4.72199881365079e-09, 4.3593818965314e-09, 4.13230287385339e-09,
4.00702389663565e-09, 3.96694049464916e-09, 4.00702387929014e-09,
4.13230283807624e-09, 4.35938183991734e-09, 4.72199873190492e-09,
5.28690339081455e-09, 6.20531869114697e-09, 7.82998706396576e-09,
1.05874635195799e-08, 2.31983334045496e-08, 0.499999924115377

# Lines wrapped
> wfes_single --equilibrium -N 10 --output-E stdout
0.499999931260253, 2.31983343180376e-08, 1.05874638827596e-08,
7.82998730182177e-09, 6.20531885243497e-09, 5.2869035051936e-09,
4.72199881365079e-09, 4.3593818965314e-09, 4.13230287385339e-09,
4.00702389663565e-09, 3.96694049464916e-09, 4.00702387929014e-09,
4.13230283807624e-09, 4.35938183991734e-09, 4.72199873190492e-09,
5.28690339081455e-09, 6.20531869114697e-09, 7.82998706396576e-09,
1.05874635195799e-08, 2.31983334045496e-08, 0.499999924115377
N = 10
s = 0.0000000000e+00
h = 5.0000000000e-01
u = 1.0000000000e-09
v = 1.0000000000e-09
```

```
a = 1.0000000000e-20
```

6.3 Verbose output

The `--verbose` flag will output timing and solver details. This flag is common for all executables. The majority of the output is produced by the `PARDISO` solver (specifically `msglvl=1`), see PARDISO parameter table. In addition, wall clock time to build the matrix and total wall clock time are printed.

7 Usage

7.1 `wfes_single` usage

`wfes_single` implements calculations for the standard Wright-Fisher model.

7.1.1 Modes

`wfes_single` supports two modes:

- `--absorption` mode assumes that absorption is possible at extinction and fixation boundaries.
- `--fixation` mode assumes that the extinction boundary is transient, and the fixation boundary is absorbing.
- `--fundamental` mode calculates the entire fundamental matrix of the Wright-Fisher model.
- `--equilibrium` mode calculates the equilibrium distribution of allele frequencies.
- `--allele-age` mode calculates moments of the allele age given a current allele frequency.

Table 1: Command line arguments for `wfes_sinlge`

Parameter	Option	Default	Type	Range	Description
Population size	`-N/--pop-size`	Required	`int`	$[2, 5 \times 10^5]$	Size of the population
Selection coefficient	`-s/--selection`	`0`	`float`	$[-1, 1]$	Individual selection coefficient
Dominance coefficient	`-h/--dominance`	`0.5`	`float`	$[0, 1]$	Dominance coefficient
Backward mutation rate	`-u/--backaward-mu`	`1e-9`	`float`	$(0, \frac{1}{4N}]$ *	Backward mutation rate ($A \to a$)
Forward mutation rate	`-b/--forward-mu`	`1e-9`	`float`	$(0, \frac{1}{4N}]$ *	Forward mutation rate ($a \to A$)

Continued on next page

Continued from previous page

Parameter	Option	Default	Type	Range	Description
Recurrent mutation	`-m/--no-recurrent-mu`	`true`	`bool`		Exclude recurrent mutation
Tail truncation	`-a/--alpha`	`1e-20`	`float`	$[0, 10^{-10}]$	Tail truncation cutoff
Integration cutoff	`-c/--integration-cutoff`	`1e-10`	`float`	$[0, 10^{-3}]$	Integration cutoff for initial number of copies
Initial number of copies	`-p/--starting-copies`		`int`	$[1, N]$	Initial number of copies
Observed number of copies	`-x/--observed-copies`		`int`	$[1, N]$	Observed number of copies for allele age calculation
Number of threads	`-t/--num-threads`	`n_cores`	`int`		Number of cores to be used for matrix construction and linear algebra
Q matrix	`--output-Q`		`path`	{`file`, `stdout`}	Output the transition probability matrix for transient states
R matrix	`--output-R`		`path`	{`file`, `stdout`}	Output the transition probability matrix between transient and absorbing states
N matrix	`--output-N`		`path`	{`file`, `stdout`}	Output the calculated rows of the fundemental matrix
B matrix	`--output-B`		`path`	{`file`, `stdout`}	Output the conditional absorption probability matrix
I vector	`--output-I`		`path`	{`file`, `stdout`}	Output initial probability distribution
E vector	`--output-E`		`path`	{`file`, `stdout`}	Output equilibrium distribution (`--equlibrium` only)

Continued on next page

Continued from previous page

Parameter	Option	Default	Type	Range	Description
V vector	`--output-V`		`path`	{`file`, `stdout`}	Output variance fundamental matrix (slow)
CSV output	`--csv`		`bool`		Generate all output in CSV format
Force parameters	`--force`		`bool`		Do not perform parameter validity checks
Verbose out	`--verbose`		`bool`		Output timing and statistical information
Help	`--help`		`bool`		Show executable options

7.2 `wfes_switching` usage

`wfes_switching` implements time-heterogeneous extension to the Wright-Fisher model.

7.2.1 Modes

`wfes_switching` supports two modes:

- `--absorption` - both extinction and fixation boundaries are absorbing for all component models
- `--fixation` - only fixation boundary is absorbing for all component models

Table 2: Command line arguments for `wfes_switching`

Parameter	Option	Default	Type	Range	Description
Population sizes	`-N/--pop-sizes`	Required	`int[k]`	$[2, 5 \times 10^5]$	Sizes of each of the populations
Selection coefficients	`-s/--selection`	`[0]*k`	`float[k]`	$[-1, 1]$	Individual selection coefficient
Dominance coefficient	`-h/--dominance`	`[0.5]*k`	`float[k]`	$[0, 1]$	Dominance coefficient
Backward mutation rate	`-u/--backaward-mu`	`[1e-9]*k`	`float[k]`	$(0, \frac{1}{4N}]$ *	Backward mutation rate ($A \to a$)

Continued on next page

Continued from previous page

Parameter	Option	Default	Type	Range	Description
Forward mutation rate	`-v/--forward-mu`	`[1e-9]*k`	`float[k]`	$(0, \frac{1}{4N}]$ *	Forward mutation rate ($a \to A$)
Probability of starting	`-p/--starting-prob`	`[1/k]*k`	`float[k]`	$[0, 1]$	Probabilty of starting in each of the component models
Relative probability of switching	`-r/--switching`	`[1]*[k,k]`	`float[k][k]`	$[0, 1]$	Transition probabiluity amtrix between the WF component models
Tail truncation	`-a/--alpha`	`1e-20`	`float`	$[0, 10^{-10}]$	Tail truncation cutoff)
Number of threads	`-t/--num-threads`	`n_cores`	`int`		Number of cores to be used for matrix construction and linear algebra
Q matrix	`--output-Q`		`path`	{`file`, `stdout`}	Output the transition probability matrix for transient states
R matrix	`--output-R`		`path`	{`file`, `stdout`}	Output the transition probability matrix between transient and absorbing states
N matrix	`--output-N`		`path`	{`file`, `stdout`}	Output the calculated rows of the fundemental matrix
B matrix	`--output-B`		`path`	{`file`, `stdout`}	Output the conditional absorption probability matrix
CSV output	`--csv`		`bool`		Generate all output in CSV format
Force parameters	`--force`		`bool`		Do not perform parameter validity checks

Continued on next page

Continued from previous page

Parameter	Option	Default	Type	Range	Description
Verbose out	`--verbose`		`bool`		Output timing and statistical information
Help	`--help`		`bool`		Show executable options

For vector argument defaults, `[z]*k` notation means a vector of length `k`, where each element is `z`. For example, `[z]*3` is `[z,z,z]`.

7.3 `wfes_sweep` usage

`wfes_sweep` implements a model of positive selection with standing genetic variation.

7.3.1 Modes

`wfes_sweep` supports one mode:

- `--fixation` - only fixation boundary is absorbing for the adaptive component

Table 3: Command line arguments for `wfes_sweep`

Parameter	Option	Default	Type	Range	Description
Population size	`-N/--pop-size`	Required	`int`	$[2, 5 \times 10^5]$	Size of the population
Selection coefficients	`-s/--selection`	Required	`float[2]`	$[-1, 1]$	Individual selection coefficient
Rate of switching	`-l/--lambda`	Required	`float`	$[1e-20, 1]$	Rate of switching from pre-adaptive regime into the adaptive regime
Dominance coefficient	`-h/--dominance`	`[0.5]*2`	`float[2]`	$[0, 1]$	Dominance coefficient
Backward mutation rate	`-u/--backaward-mu`	`[1e-9]*2`	`float[2]`	$(0, \frac{1}{4N}]$ *	Backward mutation rate ($A \to a$)
Forward mutation rate	`-v/--forward-mu`	`[1e-9]*2`	`float[2]`	$(0, \frac{1}{4N}]$ *	Forward mutation rate ($a \to A$)

Continued on next page

Continued from previous page

Parameter	Option	Default	Type	Range	Description
Tail truncation	`-a/--alpha`	`1e-20`	`float`	$[0, 10^{-10}]$	Tail truncation cutoff
Number of threads	`-t/--num-threads`	$=n_{\text{cores}}$	`int`		Number of cores to be used for matrix construction and linear algebra
Integration cutoff	`-c/--integration-cutoff`	`1e-10`	`float`	$[0, 10^{-3}]$	Integration cutoff for initial number of copies
Initial number of copies	`-p/--starting-copies`		`int`	$[1, N]$	Initial number of copies
Q matrix	`--output-Q`		`path`	{`file`, `stdout`}	Output the transition probability matrix for transient states
R matrix	`--output-R`		`path`	{`file`, `stdout`}	Output the transition probability matrix between transient and absorbing states
N matrix	`--output-N`		`path`	{`file`, `stdout`}	Output the calculated rows of the fundemental matrix
B matrix	`--output-B`		`path`	{`file`, `stdout`}	Output the conditional absorption probability matrix
I vector	`--output-I`		`path`	{`file`, `stdout`}	Output initial probability distribution
CSV output	`--csv`		`bool`		Generate all output in CSV format
Force parameters	`--force`		`bool`		Do not perform parameter validity checks
Verbose out	`--verbose`		`bool`		Output timing and statistical information

Continued on next page

Continued from previous page

Parameter	Option	Default	Type	Range	Description
Help	`--help`		`bool`		Show executable options

For vector argument defaults, `[z]*k` notation means a vector of length `k`, where each element is `z`. For example, `[z]*3` is `[z,z,z]`.

7.4 `wfafle` usage

7.4.1 Specifying the demographic

`wfafle` uses a piecewise-constant demograpghic history to track a single population. The demographic is specified each "epochs", each with a population size and length in generations:

```
wfafle --pop-sizes 100,200 --generations 100,50
```

Note that the length of the `--pop-sizes` and `--generations` vectors has to be the same length. In addition, each epoch can have a different selection coefficient, mutation rate, and dominance coefficient. For example, if we want to specify a negatively selected allele for both epochs:

```
wfafle --pop-sizes 100,200 --generations 100,50 --selection -1e4,-1e-4
```

Note that the length of the epoch is allowed to be `0` generations. For example, the following invocation only solves for the equilibrium distribution:

```
wfafle --pop-sizes 1000 --generations 0
```

The result is the same as for `wfes_single --equilibrium -N 1000`.

Given an initial allele frequency distribution, transform it into a different population size:

```
wfafle --initial 1000.csv --pop-sizes 1000,2000 --generations 0,0
```

Note that `1000.csv` file should contain $2 \times N + 1 = 2001$ entries, corresponding to the probability for each allele count.

In practice, performing these calcualtions with large population sizes requires a large amount of RAM. It is desirable that these are performed on large shared-memory clusters. To be able to recover the calculation if an intermediate step fails, using a single epoch per invocation is recommended.

7.4.2 Input

`wfafle` reads the `--initial` probability distribution from a file. The file should contain a single-line vector in a `.csv` format. Spaces around each number are allowed. The number of entries should be $2N + 1$, corresponding to the probability of each allele count for a given population size N.

7.4.3 Output

Unlike other applications, `wfafle` only has one type of output - the allele frequency distribution. The output is put directly into `stdin`. The output is formatted as a single-line `.csv` table.

Table 4: Command line arguments for `wfafle`

Parameter	Option	Default	Type	Range	Description
Population sizes	`-N/--pop-sizes`	Required	`int[k]`	$[2, 5 \times 10^5]$	Population size for each of the k epochs
Generations	`-G/--generations`	Required	`int[k]`	$[0, \infty]$	Number of generations each of the k epochs last
Selection coefficient	`-s/--selection`	`0`	`float[k]`	$[-1, 1]$	Individial selection coefficient
Backward mutation rate	`-u/--backaward-mu`	`1e-9`	`float[k]`	$(0, \frac{1}{4N}]$ *	Backward mutation rate ($A \to a$)
Forward mutation rate	`-v/--forward-mu`	`1e-9`	`float[k]`	$(0, \frac{1}{4N}]$ *	Forward mutation rate ($a \to A$)
Tail truncation	`-a/--alpha`	`1e-20`	`float`	$[0, 1 \times 10^{-5}]$	Tail truncation cutoff
Initial allele frequency distribution	`-i/--initial`	Equilibrium	`float[2N+1]`		Allele frequency distribution at the start of epoch 1
Number of threads	`-t/--num-threads`	`n_cores`	`int`		Number of cores to be used for matrix construction and linear algebra
Verbose output	`--verbose`		`bool`		Output timing and statistical information
Help	`--help`		`bool`		Show executable options