Fast computation and duality for tree sequence statistics

Peter Ralph[‡], Kevin Thornton[†], and Jerome Kelleher[§]

‡ *Mathematics and Biology, University of Oregon*
† *Ecology & Evolutionary Biology, UC Irvine*
§ *Big Data Institute, University of Oxford*

paper: (Ralph et al., 2019) **code:** `https://github.com/tskit-dev`

Tree sequences: all the genealogies

A **tree sequence** describes a correlated sequence of genealogical trees describing how a set of chromosomes are related.

- The *pedigree* plus crossover locations would give us the tree sequence for *everyone, ever*.
- Much less fully describes the history of a *sample* of genomes.
- Almost the Ancestral Recombination Graph (ARG).

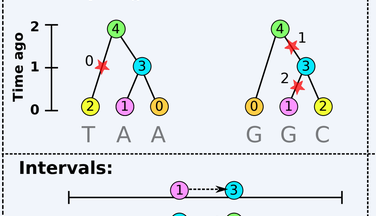
Kelleher et al. (2016) introduced the **succinct tree sequence** data structure for `msprime`; it was updated in Kelleher et al. (2018).

Tables: a data structure for tree sequences

- **Edges:** Who inherits from who.
  Records: interval (left, right); parent node; child node.
- **Nodes:** The ancestors those happen in.
  Records: time ago (of birth); individual.
- **Mutations:** When state changes along the tree.
  Records: site index; node index; derived state.
- **Sites:** Where mutations fall on the genome.
  Records: genomic position; root state.
- **Individuals:** Optional. Containers for polyploids.
  Records: metadata; pointed to by nodes.

Example

Tree topologies and mutations:



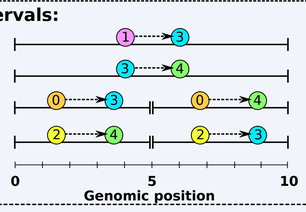
Nodes:

ID	time
0	0.0
1	0.0
2	0.0
3	1.0
4	2.0

Edges:

left	right	parent	child
0	10	3	1
0	10	4	3
0	5	3	0
0	5	4	2
5	10	3	2
5	10	4	0

Intervals:



Sites:

ID	position	ancestral state
0	2.5	A
1	7.5	G

Mutations:

ID	site	node	derived state
0	0	2	T
1	1	3	C
2	1	1	G

References

J. Kelleher, A. M. Etheridge, and G. McVean. Efficient coalescent simulation and genealogical analysis for large sample sizes. *PLoS Comput Biol*, 12(5), May 2016. URL `https://www.ncbi.nlm.nih.gov/pubmed/27145223`.

J. Kelleher, K. Thornton, J. Ashander, and P. Ralph. Efficient pedigree recording for fast population genetics simulation. *PLoS Computational Biology*, accepted, 2018. URL `https://www.biorxiv.org/content/early/2018/06/07/248500`.

P. Ralph, K. Thornton, and J. Kelleher. Efficiently summarizing relationships in large samples: a general duality between statistics of genealogies and genomes. *bioRxiv*, 2019. doi: 10.1101/779132. URL `https://www.biorxiv.org/content/early/2019/09/23/779132`.

Another example

Nodes:

id	time
0	0.00
1	0.00
2	0.00
3	0.00
4	0.00
5	0.00
6	0.00
7	0.00
8	0.09
9	0.31
10	0.39
11	0.41
12	0.43
13	0.97
14	1.04
15	1.21
16	1.36
17	1.45
18	2.41
19	2.46
20	2.85
21	3.84
22	4.45

Edges:

left	right	parent	child
0.00	1.00	8	2
0.00	1.00	8	6
0.00	1.00	9	5
0.00	1.00	9	7
0.00	1.00	10	3
0.00	1.00	10	8
0.00	1.00	11	0
0.00	1.00	11	1
0.00	1.00	12	9
0.00	1.00	12	10
0.99	1.00	13	4
0.99	1.00	13	12
0.00	0.19	14	11
0.00	0.19	14	12
0.93	0.99	15	4
0.93	1.00	15	11
0.99	1.00	15	13
0.30	0.54	16	4
0.91	0.93	16	4
0.30	0.54	16	11
0.91	0.93	16	11
0.54	0.69	17	4

cont'd

left	right	parent	child
0.54	0.69	17	12
0.69	0.91	18	4
0.69	0.91	18	11
0.00	0.30	19	4
0.69	0.69	19	11
0.19	0.46	19	12
0.69	0.99	19	12
0.00	0.19	19	14
0.93	0.99	19	15
0.30	0.46	19	16
0.91	0.93	19	16
0.69	0.69	19	17
0.69	0.91	19	18
0.19	0.30	20	11
0.19	0.30	20	19
0.60	0.69	21	11
0.60	0.69	21	17
0.54	0.60	22	11
0.46	0.54	22	12
0.46	0.54	22	16
0.54	0.60	22	17

Sample weights and summary functions

A list of **sample weights** w assigns a numeric value $w(v) \in \mathbb{R}^k$ to every sample node.
The **subtree weight** $x_T(u)$ on tree T of node u is the sum of weights of all sample nodes descended from u:

$$x_T(u) = \sum_{v \,:\, v \le_T u} w(v),$$

where $v \le_T u$ if u is on the path from v to root in the tree T.
The **total weight** is the sum of the weights over all samples: $w_{\text{total}} = \sum_v w(v)$.
A **summary function** is a real-valued function $f(w_1, \ldots, w_k)$ with the property that $f(0) = f(w_{\text{total}}) = 0$.

Site statistics

The **allele weight** for allele a at site j is the total weight of all samples inheriting this allele:

$$\bar{x}_j(a) = \sum_{v \,:\, g_j(v)=a} w(v),$$

where the sum is over all sample nodes v for which $g_j(v)$, the allele carried by node v at site j, is equal to a.
The **site statistic** at site j:

$$\text{Site}(f, w)_j = \sum_a f(\bar{x}_j(a)), \tag{1}$$

and in the *window* $[i, j)$:

$$\text{Site}(f, w)_{[i,j)} = \frac{1}{j-i} \sum_{k=i}^{j-1} \text{Site}(f, w)_k. \tag{2}$$

Branch statistics

The **Branch statistic** for a tree T is

$$\text{Branch}(f, w)_T = \sum_{u \in T} \beta_T(u) \left(f(x_T(u)) + f(w_{\text{total}} - x_T(u)) \right), \tag{3}$$

where $\beta_T(u)$ is the length of the branch ancestral to node u in tree T, and in a window $[i, j)$ is

$$\text{Branch}(f, w)_{[i,j)} = \frac{1}{j-i} \sum_{k=1}^{|\mathbb{T}|} \ell_k(i, j) \text{Branch}(f, w)_{T_k}. \tag{4}$$

Examples

For allele frequencies, use sample weights $\mathbf{1}_S$ with $\mathbf{1}_S(u) = 1$ if $u \in S$ and $\mathbf{1}_S(u) = 0$ otherwise.

Nucleotide diversity of S: Let $w = \mathbf{1}_S$, and

$$f(x) = \frac{x(n-x)}{n(n-1)}.$$

Segregating sites in S: Again, $w = \mathbf{1}_S$, and

$$f(x) = \begin{cases} 1 - \frac{x}{n} & \text{if } x > 0 \\ 0 & \text{otherwise.} \end{cases}$$

Phenotypic correlations: With normalized phenotypes $z(u)$, use $w_1(u) = z(u)$, and $w_2(u) = 1/n$, and

$$f(x_1, x_2) = x_1^2 / (2x_2(1-x_2)n(n-1)).$$

Then, $\text{Site}(w, f)_j = r_j^2$ is the squared correlation between z and the allele at site j, while $\text{Branch}(w, f)_T = r_j^2$ is the expected squared correlation between z and mutations on this tree.

A general class of statistics

Every single-site statistic is a function from genotype patterns to $\mathbb{R}$, and each SNP genotype pattern is determined by the samples below the edge it occurs on.

Ingredients: sample weights and a summary function.

1. Find the total weight of everyone below each mutation/branch.
2. Apply the summary function to this weight.
3. Add these up and divide by the genome length.

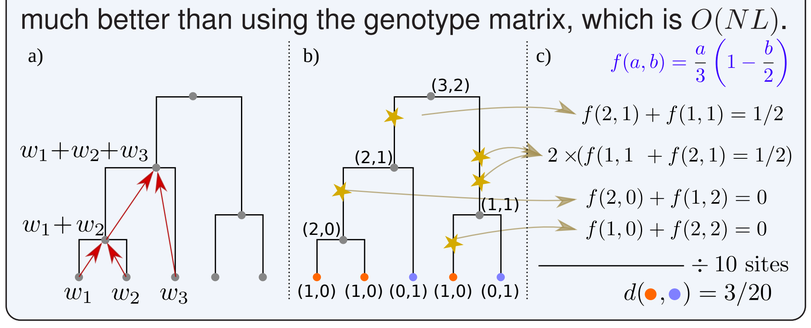
Computation

0. Find x for each node in the first tree.
1. Compute $f(x)$ for each mutation on this tree, and add these to the total.
2. Update the tree, and update n for each node in the path from changed nodes to the root.
3. Return to (1).
4. When done, divide the total by the sequence length.

Complexity for N samples at L SNPs with T trees:

$$O(N + L + T \log(N)),$$

much better than using the genotype matrix, which is $O(NL)$.



Duality

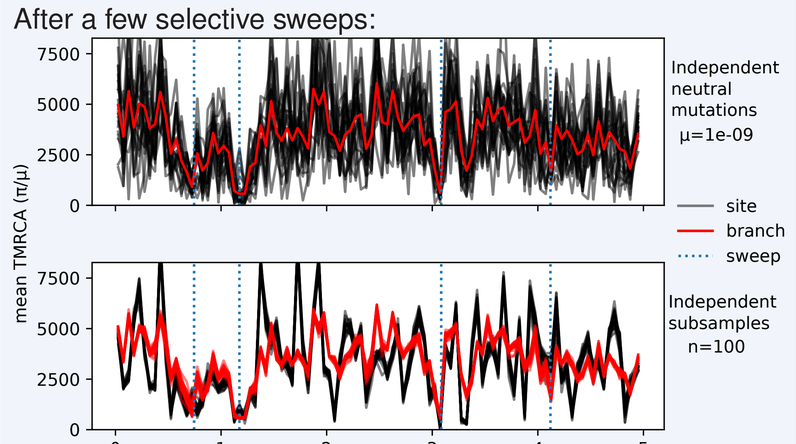
Under infinite-site mutation at rate μ,

$$\mu \text{Branch}(f, w)_{[i,j)} = \mathbb{E}\left[\text{Site}(f, w)_{[i,j)} \mid \mathbb{T}_{[i,j)}\right], \tag{5}$$

and

$$\text{Var}[\text{Site}(f, w)_{[i,j)}] = \mu^2 \text{Var}\left[\text{Branch}(f, w)_{[i,j)}\right] + \frac{\mu}{j-i} \mathbb{E}\left[\text{Branch}(f^2, w)_{[i,j)}\right].$$

After a few selective sweeps:



Real data: 1000 Genomes tree sequences from Relate (Speidel et al 2019):