

File S2 Details on the simulation of genealogies

32 The genealogies are constructed by tracing the ancestry of the sampled genetic material backwards in time. Apart from
 multiple and simultaneous mergers, another special feature is that the genealogical process takes into account explicitly
 34 that individuals are diploid and bi-parental and thus avoids logical inconsistencies that may occur when independently
 simulating the genealogies at different loci (Wakeley *et al.* 2012). However, this realism comes at a computational cost
 36 and in cases where we are only interested in average levels of genetic diversity, i.e. for the analyses underlying Figures 5,
 7, and S3 we resorted to independently simulating the genealogies at the different loci.

38 The current state of the ancestry is defined by a set of lineage packages for each population (source population and
 newly founded population). Such a lineage package contains all the genetic material that is traveling within the same
 40 individual at that time point. It has two sets of slots, one set for each genome copy. Each set has a slot for each locus.
 If the genetic material at a certain locus and genome copy is ancestral to the sample, the slot is occupied by a node,
 42 otherwise it is empty.

The ancestral history starts with $2 \cdot n_s$ lineage packages in the newly founded population. Initially all slots in the
 44 lineage packages are occupied by nodes. From there, the ancestry is modeled backwards in time until at each locus there
 is just one node left. Given the state of the ancestry in generation t , the state in generation $t - 1$ is generated as follows:
 46 Backward in time, each generation starts with a migration phase (Figure S2). All lineage packages that are currently in
 the newly founded population choose uniformly without replacement one of the Y_t migrants from the source population,
 48 or one of the $N_t - Y_t$ residents. Note that in our simulations with a single founding event, $Y_0 = N_0$ and $Y_t = 0$ for all
 $t > 0$, whereas in the case of multiple introductions, Y_t can be positive also at $t > 0$. According to their choice in this
 50 step, lineage packages either remain in the newly founded population or are transferred to the source population.

Then each lineage package splits into two because the two genome copies (sets of slots) each derive from a possibly
 52 different parent (see Figure S2). Lineage packages that do not contain ancestral material are discarded immediately. For
 each of the remaining lineage packages, recombination is implemented by independently constructing a stochastic map
 54 $R : \{1, \dots, n_l\} \rightarrow \{0, 1\}$ such that

$$R(1) = \begin{cases} 0 & \text{with probability } \frac{1}{2} \\ 1 & \text{with probability } \frac{1}{2} \end{cases} \quad (\text{S5})$$

and then

$$R(n+1) = \begin{cases} R(n) & \text{with probability } 1 - \rho_n \\ 1 - R(n) & \text{with probability } \rho_n \end{cases} \quad (\text{S6})$$

56 is drawn recursively for $n \in \{1, \dots, n_l - 1\}$. The recombination probability ρ_n between loci n and $n + 1$ was 0.5 for all
 analyses in this study. A node at locus l in the new lineage package is placed into the first genome copy if $R(l) = 0$ and

58 into the second genome copy if $R(l) = 1$.

60 After each lineage package underwent splitting and recombination, all resulting lineage packages uniformly pick one of
62 the N_{t-1} or k_0 individuals as ancestor, depending on whether they are in the newly founded or in the source population,
this time with replacement (see Figure S2). Lineage packages that chose the same ancestor are merged. If there is more
than one node at the same genome copy and slot, a coalescent event takes place.

64 Because genetic drift is strong in small populations, many pairs of lineages will already encounter their common
66 ancestor within the newly founded population. The lineages that did not coalesce until time 0 must all be in the source
population which is assumed to be of constant size k_0 at all times. To simulate the remainder of the ancestry, we could
68 proceed by going backwards generation-by-generation. However, as the source population is large it would take a long
time until all lineages find their most recent common ancestor (MRCA) and in most generations nothing would happen.
Furthermore, nodes within the same lineage package typically become separated by recombination relatively fast. For the
70 sake of computational efficiency, we therefore use an approximative algorithm to simulate the remainder of the ancestral
72 history. This efficient simulation mode excludes multiple and simultaneous mergers, events that should be very rare for a
reasonably large source population size k_0 . Independently for each locus, we determine the remaining number of nodes
 n_{total} and draw the number of generations T until the next coalescent event from a geometric distribution with success
probability

$$p_{merge} = \frac{\binom{n_{total}}{2}}{2k_0}. \quad (S7)$$

74 We merge two randomly chosen nodes, reduce n_{total} by 1, and update the current time to $t - T$. We repeat this procedure
until eventually there is only one node left, the MRCA of all sampled genetic material at the respective locus. Throughout
76 the simulation, we store all information needed to provide the topology and branch lengths (in number of generations)
for the genealogies at each locus.

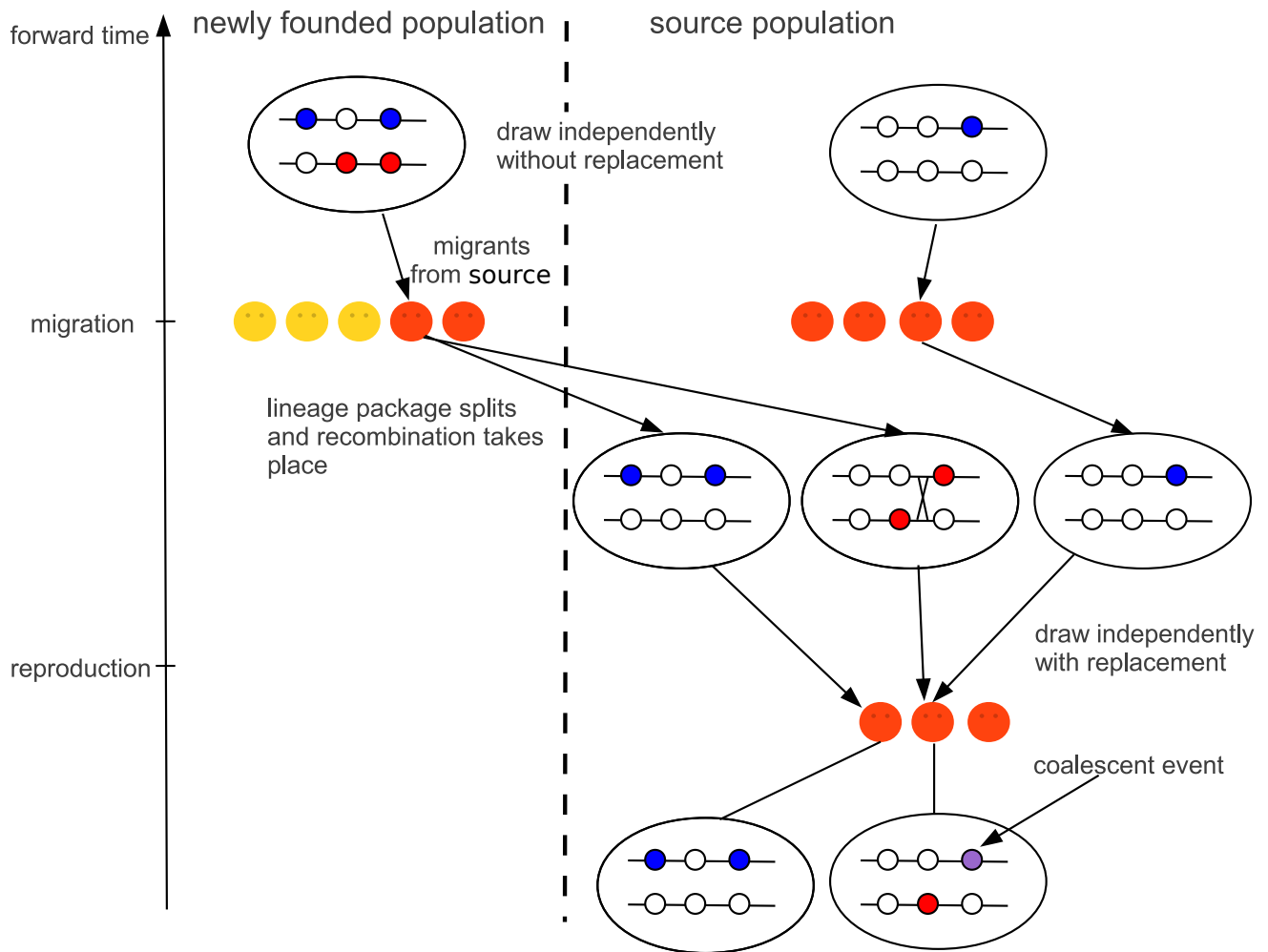


Figure S2 Illustration of the backward-in-time simulation of genealogies.