ESP

Evolution in Simulated Populations

A Monte Carlo simulation-based program to evaluate theoretical population genetic trajectories and test hypotheses using parametric bootstrapping

HTTP://WWW.BIO.SDSU.EDU/PUB/ANDY/ESP.HTML

Andrew J. Bohonak
San Diego State University

George K. Roderick
University of California, Berkeley

Citation:
Bohonak, A. J., N. Davies, F. X. Villablanca, and G. K. Roderick. 2001. Invasion genetics of New World medflies: testing alternative colonization scenarios. Biological Invasions **3:** 103-111.

v. 1.16 16 April 2003

Introduction

As with other biological disciplines, hypothesis testing in molecular ecology and population genetics requires two primary decisions. First, complex, multivariate data sets need to be reduced to simpler summary statistics or analytic frameworks. Second, null hypotheses must be evaluated using theoretical expectations that account for appropriate sources of variation. Traditionally, stochastic and/or sampling variation (sensu Slatkin and Arter 1991) have been considered when constructing population genetic null models. However, initial state variation can be as important as stochastic and sampling variation when 1) the hypothesis is designed to infer process, rather than simply describe pattern and 2) the populations are not at an equilibrium between drift, mutation and gene flow (see Figure 1). For example, suppose that very low values of heterozygosity are found in a particular population. In order to generate a null distribution for a founding event by *n* colonists, one must consider the among-population variation that results from random samples of *2n* genes from the ancestral pool (e.g., Hairston et al. 1999, Bohonak et al. *in press*). Initial state variation also consists of differences among population sets that have been randomly founded by the same process.

Over time, the influence of these founder effects decays, initial state variation becomes less important, and eventually the level of population differentiation reaches a balance between migration, mutation and random genetic drift. Natural selection will also be important for traits not evolving under neutrality. Because of these complexities, interpretations of population differentiation are context-dependent and usually open to debate. It is often difficult to determine whether populations are old enough to have reached a drift-gene flow-mutation equilibrium, and if they are not at equilibrium, multiple colonization scenarios may be consistent with the observed patterns. For very young populations (e.g., invading species) unambiguous interpretations can particularly difficult.

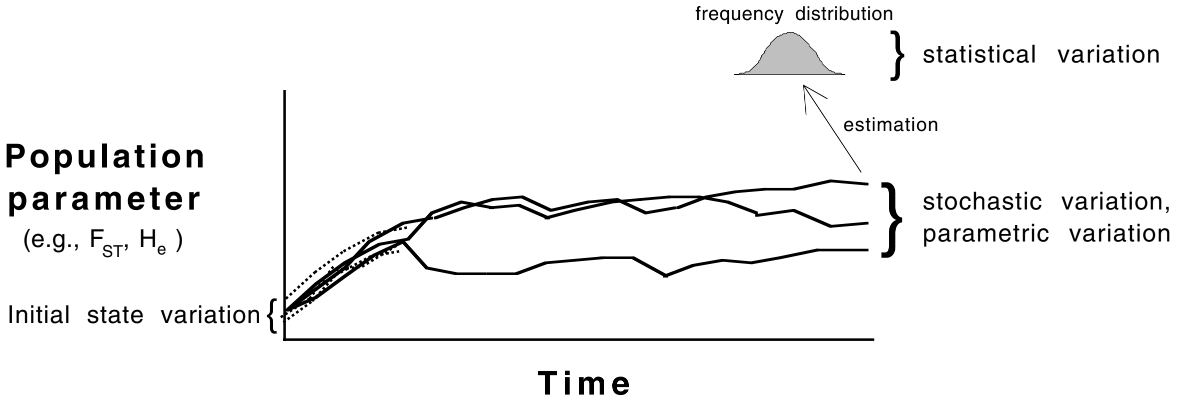


Figure 1. Four sources of variation for population genetic statistics. Each line represents one group of populations; the three solid lines all begin with the same initial state.

General coverage of these topics can be found in Hartl and Clark (1997), and in numerous reviews (e.g., Slatkin 1985, Avise 1994, Roderick 1996, Bohonak 1999).

With these motivations, we have written ESP in order to provide a framework for evaluating population differentiation over time while accommodating stochastic variation, statistical variation and variation in initial state. ESP provides a general, Monte Carlo-based algorithm to generate null distributions via parametric bootstrapping. The program has, in our view, at least four uses:

1. To evaluate the likelihood of particular values for population parameters such as rate of gene flow and effective population size.
2. To evaluate the likelihood of particular colonization scenarios.
3. To provide guidance when designing a sampling regime and choosing molecular markers.
4. To provide a teaching tool that illustrates how gene genealogies evolve over time in the context of drift, gene flow and mutation.

Style conventions

Throughout this manual, references to other sections of the manual appear in a larger font (e.g., Parameters and Settings), variables and statistics appear in *italics* (e.g., F_{ST}), program options and screen output appear in **`bold Monaco`** (e.g., **`# populations`**), and examples of ESP outfiles are represented in small Geneva (e.g., Alleles saved by ESP v0.7).

Getting started

Hardware

ESP for Power Macintosh requires from 10 MB to more than 50 MB of RAM, depending on simulation size (see Performance below). For large simulations, a G3 or G4 processor is suggested. Extensive testing as been conducted using Systems 8.5 through 9.1 on G3 (including iMacs) and G4 processors, although the application may also run well on other operating system versions.

Software installation

Installation of ESP only requires downloading and expanding the SEA (self expanding archive) available on the ESP web site. If the default RAM requirements are inappropriate for a particular set of simulations, they can be changed manually within the Macintosh finder (through the Get Info->Memory menu). Unfortunately, large simulations will dominate processing power, so that users will find it easiest *not* to run ESP in the background.

Operation

There are essentially four phases to population genetic analyses with ESP.

1. First, customize population parameters and output analyses from the opening screen.
2. Second, start ESP by entering <y>.
3. Summary statistics from one or more Monte Carlo simulations appear on the screen as the populations interact and evolve over time. More extensive output is saved as a tab-formatted text file in the ESP folder.
4. ESP outfiles can be easily viewed and manipulated in a spreadsheet such as Microsoft Excel. The outfiles are column-formatted in a manner that makes graphing the results relatively straightforward. The screen output buffer may also be sent directly to a printer.
5. When quitting ESP. the user is prompted to "save before quitting". This will save the screen buffer, which is usually not necessary, because outfiles are saved after each Monte Carlo simulation by default.

Additional options are described below.

Program overview

ESP utilizes Monte Carlo simulations to simulate the evolutionary process, and calculates summary statistics which incorporate initial state variation, stochastic variation and statistical variation (described in Bohonak and Roderick, *submitted*). ESP keeps track of each allele's* genealogical relationship to other alleles and an array of its frequency in each population. Migration, mutation and reproduction are simulated as simple Poisson processes, assuming that mating is random and that each individual contributes a large number of gametes to the gamete pool (the standard Wright-Fisher model). Mutation is treated as an infinite-allele model, and as a result, sequence convergence and saturation are not considered (i.e., the relationships among alleles are always known). Stepwise mutations (for microsatellites) are not currently incorporated.

Gene flow follows a simple island model. Under some conditions (low migration rates, very young and/or very large populations), founder effects will dominate patterns of

* For clarity, **allele** is used throughout this manual to identify unique genes in the gene pool. In contrast, **haplotype** will be used generically to refer to all members of the gene pool, whether or not they are unique. Thus, a haplotype might consist of a 1000 bp length of DNA or a "1 bp" allozyme locus. The number of haplotypes in a diploid gene pool is twice the effective population size.

differentiation, and few qualitative differences might be expected between island and spatially explicit models of migration. Future versions of the program will incorporate additional migration patterns.

A wide variety of initial conditions are permitted. Populations can be founded through an algorithm that approximates the desired initial value of F_{ST}, or through a flexible General Colonization Algorithm (GCA). The GCA permits populations to be randomly founded from randomly generated or empirical data sets in numerous ways.

Confidence intervals can be generated across multiple replicates for one or more points in time, and for one or more summary statistics. These confidence intervals can be compared to empirical data to test hypotheses via 'parametric bootstrapping'.

The program's accuracy and robustness have been tested by extensively varying parameters such as sequence length, mutation rate, number and size of populations, migration rate and initial population structure. Rates of divergence and equilibrium values for population differentiation generally agree with theoretical expectations (e.g., Cockerham and Weir 1987, Slatkin 1994, Hartl and Clark 1997). Algorithms for calculating θ and Φ_{ST} have been verified by comparisons with output from Schneider *et al.*'s (1998) Arlequin and Miller's (1998) TFPGA.

Critiques, suggestions, commentary and bug reports are welcomed. Send email to <bohonak@sciences.sdsu.edu> and include the version number and error number (if applicable). Updates will be made available periodically at <http://www.bio.sdsu.edu/pub/andy/ESP.html>.

Parameters and settings

User-defined

Upon launching the program, the values of approximately 20 parameters and settings are displayed. ESP utilizes a preferences file that is placed in the program folder, so that these settings are retained after the program is quit. If the preferences file is absent, the program will revert to defaults and create a new preferences file.

`g  # generations` Zero to 4 million.

`?  print every 50 gens` How often summary statistics will be calculated and printed

`n  2 x population size` The population size in haplotypes. (For a diploid organism, this will be equivalent to $2N_e$). Limitations are outlined in *Parameter limits* below.

`p # populations`

Number of populations in the simulation (1 to 1000). Only a subset of these will be sampled for summary statistics (see **`sample...`** below).

`m # migr./pop/gen`

The number of individuals migrating per generation, assuming the organisms are diploid. The per haplotype migration rate is displayed on the following line.

`u µ/base pair/gen`

Mutation rate per base pair per generation. Typical values might be 1 x 10^{-8} for noncoding DNA, or 1 x 10^{-6} for a "1 bp" allozyme locus. Rate heterogeneity is not permitted. An infinite allele model is used (i.e., there are no convergent alleles). The per population mutation rate is provided on the next line.

`# number steps ...`

Number of mutations between unrelated alleles. This setting defines the number of mutational steps between "unrelated" alleles (unique haplotypes present at generation 0). For colonization scenarios that begin with only one haplotype, this number will be unimportant. For colonization scenarios that utilize previously saved infiles, the value in the infile will be used. See Allele nomenclature and Summary statistics.

`s sample ...`

Calculation of summary statistics can be based on a random sample of haplotypes and populations, as defined by the user.

`b seq.length(bp)`

For simulation of allozyme or similar data sets, simply set to 1, then set the desired per locus mutation rate.

`r # replicates`

Number of replicates, 1 to 1000. Note that for calculating confidence intervals, a minimum of 40 is necessary, although the raw data will be saved in any case. (See below).

`l number of loci`

Maximum of 15. Linkage is not considered.

`x save each pop?`

With this option set to 'yes', ESP saves haplotype frequencies after the final generation for all replicates. The file format described under Colonization algorithm: *User-defined source populations*. In subsequent simulations, these files may be used as sources for colonizations, or to begin with a new set of parameters.

`f frequencies?`

The frequencies of all alleles (regardless of sampling protocol) will be saved either once (at the simulation's end), or each time the other summary statistics are calculated. Frequencies will be incorporated into the standard outfile as described below.

`d DOS file?`

The final allele frequencies may be saved as DOS files suitable for the programs Arlequin (Schneider et al. 1998) and/or TFPGA (Miller 1998).

`a calc #all./pop?`

The number of unique alleles at each locus can be saved for each population. Otherwise, only the total number of unique alleles (across all populations) is calculated.

`i initial Fst`

Initial value of F_{ST} that ESP will attempt to generate (for simulations that do not utilize a previously created file of colonists). Algorithm is described in Colonization algorithms: *Default*.

`c calc. CI?`

Confidence interval calculation. With this option enabled, mean and median values, 95% and 99% CIs will be calculated across replicate Monte Carlo simulations at each time interval for some summary statistics. If the number of replicates is less than 40, the user is prompted to increase replication. When fewer than 40 replicates are entered, the raw data will be saved in an outfile, but mean, median and CI calculation will not take place. (Data from these files can later be pooled together to manually calculate a CI in a spreadsheet application. A sample Excel spreadsheet is provided for this purpose in the ESP folder.) See Output: *Sample-based*: Confidence intervals below.

`e estimate Nm?`

This option estimates the number of migrants per generation (typically abbreviated $N_e m$) according to equilibrium expectations from a simple island model (see Output: *Sample-based*: Gene flow estimation). This permits one to determine the biases that might result from estimating gene flow in nonequilibrium conditions.

`h estimate H?`

Total and population-level expected heterozygosity will be calculated (see Output: *Sample-based*: Expected heterozygosity).

`t NEXUS?`

User has the option to save an allele network in NEXUS format (reference) for each locus. This can be done at the end of the simulation (in which case it will also be printed on the screen) or each generation that summary statistics are saved. See Output: *All haplotypes*: NEXUS tree for further details.

`z input file?`

Input from a previously saved file As described below, colonizations may take place from previously created source populations, rather than using the default algorithm. Further information about the colonization process and the name of the infile will be required after the simulation is started.

Parameter limits

Maximum number of descendants that an individual allele may have	9999

When this number is exceeded, ESP renumbers all descendants of the allele, reclaiming numbers from alleles that have gone extinct.

Maximum number of generations	4,000,000
Maximum number of populations	1000
Maximum number of loci	25
Maximum number of replicates	10000
Maximum number of mutational steps between alleles unique at generation 0 and those which appear later	60

Maximum population size

The number of haplotypes per population must be

1) ≤ 65,500

2) ≤ 4.295 x 10^9 / (# base pairs)

3) ≤ 4.295 x 10^9 / (# populations)

Colonization algorithms

The manner in which populations are founded in nature will dramatically affect their initial levels of divergence; similarly, the choice of initial conditions in an ESP simulation can dramatically affect its outcome. One of ESP's primary uses is to quantify the persistent effects of bottlenecks and founder effects on population differentiation. As a result, populations can be colonized in a wide variety of ways.

Default

When `input file?` is turned off, initial values of F_{ST} are approximated by the following algorithms. Define P to be the number of populations in the simulation, $2N_e$ to be the effective population size in haplotypes and F_0 to be the target value for F_{ST} at generation 0:

$F_0 = 0$: When `initial Fst` is set to zero, the user will be prompted for an initial number of alleles at each locus, and then for their initial frequencies. All populations begin with this same configuration.

$F_0 = 1$: Each population begins with a single, unique allele.

$0 < F_0 < 1$: At generation 0, P haplotypes are temporarily created for each locus; each is a fixed number of mutational steps apart (see Parameters and Settings: *User-defined*: `number steps…`). One of these alleles is chosen randomly, and $2N_e{}^*F_0$ haplotypes in population 1 are assigned this allele's identity. This process is repeated until all $2N_e$ haplotypes are assigned to population 1. Allele identities are assigned to each remaining population in the same manner. (Algorithm derived from discussion in Wade and McCauley 1988). Unused haplotypes are then eliminated before the simulation continues.

This algorithm provides an acceptable approximation when $2N_e$ and P are large. For more control over the colonization process, create user-defined source populations manually, or with prior simulations.

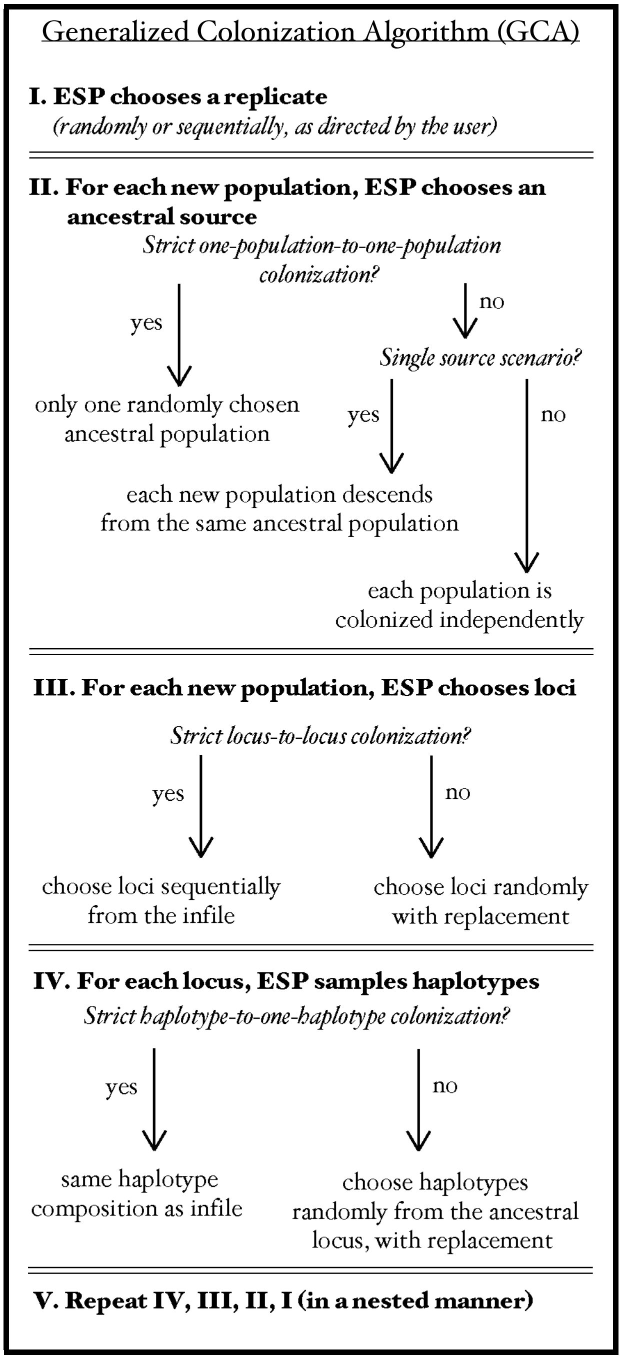
"ESP freqs" infiles

When `input file?` is turned on, ESP samples external files in order to found new populations at generation 0. These infiles may be created from an ESP simulation, or manually. ESP's Generalized Colonization Algorithm (GCA) can then be used to simulate colonization events of almost any type (see diagram below).

Creating infiles With the `save each pop?` setting activated, a file will be created for all of the replicates in a particular run. Files will be named “ESP freqs_**x**”, where **x** is the current replicate number. Frequency information, allele identities and simulation settings will be included.

Preexisting "ESP freqs" infiles can be altered by the user. Files can also be created manually in a word processing program, providing they are saved as text files and precisely follow the format described in Output: *"ESP freqs" infiles* below.

Use as input files "ESP freqs_**x**" infiles may be used as input files if they are located in the ESP folder, and if their content is consistent with the format described in Output: *"ESP freqs" infiles*.


Generalized Colonization Algorithm (GCA)
I. ESP chooses a replicate
(randomly or sequentially, as directed by the user)
II. For each new population, ESP chooses an ancestral source
Strict one-population-to-one-population colonization?
yes
no
only one randomly chosen ancestral population
Single source scenario?
yes
no
each new population descends from the same ancestral population
each population is colonized independently
III. For each new population, ESP chooses loci
Strict locus-to-locus colonization?
yes
no
choose loci sequentially from the infile
choose loci randomly with replacement
IV. For each locus, ESP samples haplotypes
Strict haplotype-to-one-haplotype colonization?
yes
no
same haplotype composition as infile
choose haplotypes randomly from the ancestral locus, with replacement
V. Repeat IV, III, II, I (in a nested manner)

When `input file?` is on, the user is able to utilize the GCA diagrammed below. This allows a great deal of flexibility in generating null hypotheses for specific colonization scenarios, or specific changes in demography. For clarity, we usually refer to the infile populations as the "ancestral populations". By sampling from these ancestral populations using different parameters, one can test many hypotheses in a common framework: for example, what are the effects of extreme vs. minor population bottlenecks? After a change in gene flow, how long does it take to reach equilibrium? Can the effects of habitat fragmentation be seen in less than 50 generations? (Fragmentation can be modeled as a change in number and size of populations, as well as gene flow. The ancestral populations receive pre-fragmentation parameters, and subsequent simulations that sample from the ancestral infile receive post-fragmentation parameters.)

Detailed information about the colonization process will be requested prior to replicate 1. Options include:

Choice of replicates

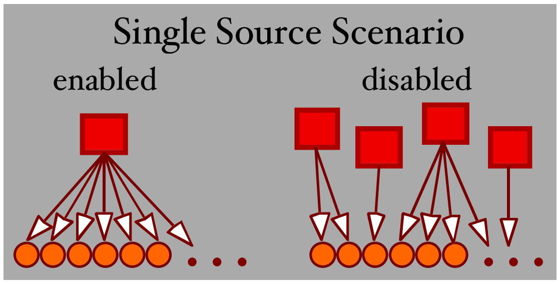
If the number of replicates in the infile meets or exceeds the number of replicates in the current run, the ancestral replicates can be used sequentially, or sampled randomly with replacement.

The random replicate option is automatically used when there is only one replicate in the current run, or the number of replicates in the infile is less than the number of replicates in the current run.

Choice of loci

If the number of loci in the infile equals the number of loci in the current run, the loci can be used sequentially in a strict ancestral locus-to-new locus fashion. Otherwise, the new loci are sampled randomly with replacement from the infile.

Choice of populations



By default, each descendant population is chosen randomly from all available ancestral populations. However, if the number of populations in the infile equals the number of populations in the current run, the populations can be used sequentially in a strict ancestral population-to-new population fashion.

A "Single Source Scenario" can also be used. This forces all new populations to be colonized from the same ancestral population. Thus, the choice of an infile population happens only once at the beginning of each replicate. The haplotype composition in each descendant population will still be unique, because the colonizing haplotypes can be randomly chosen each time.

Running multiple simulations with and without the Single Source Scenario permits the comparison of widespread, simultaneous introductions from a single source to widespread, simultaneous introductions from multiple sources.

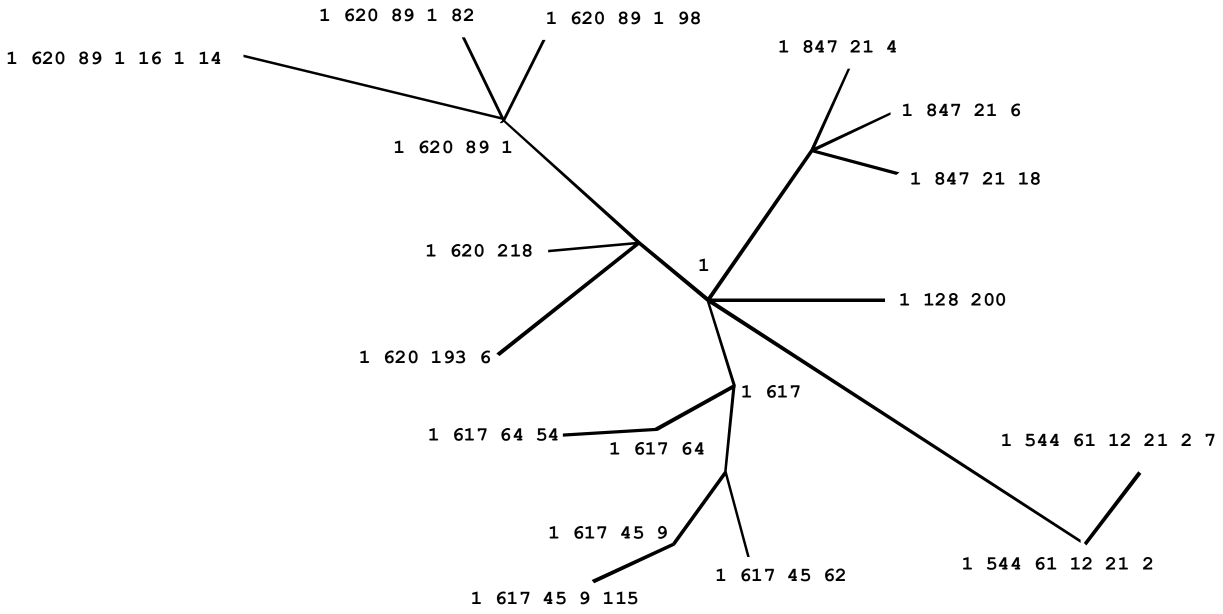
Haplotype sampling (size of the population bottleneck) Two options are available:

1. Descendant haplotypes can be chosen randomly from all available ancestral haplotypes, with a population bottleneck that is specified by the user. With this option, even if the "bottleneck size" is equal to the effective population size, the ancestral and descendant population may differ slightly because of the random sampling process. Entering 1 here means that all new haplotypes descend from a single, randomly chosen ancestral haplotype (the extreme population bottleneck!).

2. Colonizations can occur in a strict one-ancestral-haplotype to one-descendant-haplotype fashion. For this option, the population size of the infile must equal the population size of the current simulation.

Allele nomenclature

By default, ESP names alleles in a genealogically informative manner. Unique alleles present in generation zero are designated 1, 2, 3 .... New alleles one mutational step from allele 1 are designated 1_1, 1_2, 1_3 and so on. Underscore marks continue to designate the relationship between new mutations and preexisting alleles. For example, an allele network at the end of a simulation might look like:



Note that many software packages will replace underscore characters with blank spaces. This is the case with TreeView(Page 1996), which produced the unrooted network above.

Allele 1 (in the center) is the ancestor for all haplotypes in this simulation. Note that 1_620_89_1 has three extant descendants. One was the 82nd mutant to originate from this allele, the second was the 98th mutant. The leftmost allele is two steps from 1_620_89_1. The unlabeled node where its immediate ancestor should be indicates that 1_620_89_1_16_1 is now extinct. 1_620 has also disappeared.

Summary statistics

Sample-based statistics

The following statistics are calculated from a sample of the gene pool, as specified from the user (see `sample ...` above).

F-statistics Estimates of F_{ST} are calculated from the user-defined population sample of the haplotypes with the methods of Excoffier *et al.* (1992): Φ_{ST}, using the actual number of mutational steps between haplotype pairs in the distance matrix, and Weir and Cockerham (1984): θ, which ignores branch length differences among alleles. Equilibrium expectations are presented by ESP on the screen, based on the equations described in *Screen Output: Expected values for F_{ST}* below. For calculation of confidence intervals, a complete absence of variation is assigned an F_{ST} value of zero. (In many algorithms, F_{ST} is undefined when no variation exists).

Gene flow estimates Simple estimate of gene flow ($N_e m$) from the equation

$$F_{ST} = \frac{1}{4N_e m + 1}$$

which assumes an island model at equilibrium (Wright 1931, Hartl and Clark 1997). This statistic is also based on a sample from the population, unless `sample ...` has been set to include all populations and all haplotypes. "Infinite" gene flow estimates correspond to $F_{ST} = 0$.

Number of alleles The total number of extant alleles in the population sample at each locus is always counted. If the `# alleles/pop?` option is enabled, the number per population will also be enumerated for each locus. The results will appear in the outfile.

Expected heterozygosity The expected individual heterozygosity is calculated at two levels (see Hartl and Clark 1997):

1) H_T: expected heterozygosity across all individuals sampled, regardless of population affiliation.
2) H_S: expected heterozygosity calculated within each (sub)population, then averaged over (sub)populations.

Statistics based on all haplotypes

The following two statistics are calculated from all haplotypes in the gene pool, irrespective of the sampling regime:

Allele frequencies When the `frequencies?` option is toggled, the frequencies of each allele will be calculated either each time summary statistics are generated, or only at the end of the simulation. These frequencies are always based on the entire gene pool, and appear in the outfile.

NEXUS tree With the `NEXUS tree?` option enabled, ESP will generate NEXUS code (Maddison et al. 1997) so that a picture of the allele network may be generated in programs such as TreeView (Page 1996). Note that this will consist of all haplotypes in all populations at each locus. Also note that population-specific trees will not be generated and that TreeView has a maximum of 500 alleles. The NEXUS code will appear on the screen, and in a text-only outfile.

If desired, ESP will use sequential labels in this code (1,2,3 ...) rather than genealogically informative labels. This may be desired if allele names become excessively long. Correct NEXUS code will result in either case.

Screen output

Basic screen output

```
- - - - - - - - - - ->file  'ESP  OUT.2'      replicate    1 of  1

Loc.  #All.   PHIst   Theta      gen
----  -----  ------  ------  -------

   1     17   0.744   0.744        0
   2     14
   1     20   0.674   0.674       50
   2     15

  ... text omitted

   1     13   0.878   0.878     5000
   2     15

   ...  Elapsed  time  for  Rep.  1:  15  seconds
   ...  Results  saved  to  file  'ESP  OUT.2'


   Nm  =  0.10  ind./gen         EXPECTED  EQUILIBRIUM   VALUES   FOR  FST:
infinite  island     infinite  island      finite  island      finite  island
      no  µ          infinite  alleles          no  µ        infinite  alleles
---------------      ----------------      -------------      ----------------
     0.833                  0.825              0.831                0.712

Enter  's'  to  change  settings  or  'r'  for  another  group  of  runs.
```

The basic screen output begins with the title of the output file, and the current replicate number. Every 50 generations, the number of alleles at each locus and two estimates of F_{ST} are calculated from the user-defined sample size. (Note that colonization occurs at generation 0, and that the statistics calculated at generation 0 come before any gene flow, drift or mutation). After the final generation, the elapsed time is provided, and ESP verifies that the outfile has been written. The amount of gene flow (# migrating haplotypes per generation) is printed, followed by four equilibrium expectations for F_{ST} (see *Expected values for F_{ST}*). ESP prompts the user to continue.

The screen output buffer may be printed or saved as a text file. However, the outfile saved by ESP provides more information. Outfiles can also be more easily input into a spreadsheet program than copied screen text, because they are tab-delimited, rather than fixed width.

Optional settings

Screen output will also contain estimates of $N_e m$ if **`estimate Nm?`** is enabled, and expected heterozygosity (total and (sub)population) if **`estimate H?`** is enabled. (Allele frequencies and the number of alleles per population will appear in the outfile if the appropriate settings are enabled, but are not printed on the screen), If **`NEXUS tree?`** is turned on, NEXUS code will appear at the conclusion of the simulation for each locus (see *NEXUS* above). If additional DOS outfiles are created, a notification will appear on the screen.

If the GCA is used to begin the simulation and random population sampling is chosen, screen output will reflect which populations are being sampled. This list will not appear in the outfile, although a summary of the available infiles will.

Confidence interval mode

```
 Replicate      Time     Est.Finish
----------    --------    ----------
  0 of   40   10:41  AM    10:42  AM
 10 of   40   10:42  AM    10:43  AM
 20 of   40   10:42  AM    10:42  AM
 30 of   40   10:42  AM    10:42  AM
 40 of   40   10:43  AM    10:43  AM

   no gene  flow               EXPECTED  EQUILIBRIUM   VALUES   FOR  FST:
infinite  island     infinite  island      finite  island      finite  island
      no  µ          infinite  alleles         no  µ        infinite  alleles
---------------      ----------------      -------------      ----------------
     1.000                1.000              1.000               0.999

            Phi-ST            Theta
  Gen      [2.5]  [97.5]     [2.5]  [97.5]
-------    ------------     ------------
      0     0.005  0.005      0.005  0.005
     50     0.213  0.238      0.213  0.238
    100     0.374  0.417      0.374  0.417


   ...  Results  saved  to  file  'ESP  CI  1.40'


Enter  's'  to  change  settings  or  'r'  for  another  group  of  runs.
```

With `calc CI?` turned on, normal screen output is suppressed. Instead, an update of the expected finishing time will be given every 50 replicates. At the conclusion of all replicates, expected equilibrium values of F_{ST} are displayed, and 95% confidence intervals for are displayed for Φ_{ST}, θ, and $N_e m$ (if enabled). Additional summary statistics are provided in the confidence interval outfile along with means, medians, 95% CIs and 99% CIs.

Expected values for F_{ST}

Equilibrium expectations are displayed on the screen before and after each run based on four sets of assumptions. These are useful when determining how close a set of populations is to equilibrium, or what biases might be present when model assumptions are violated.

The four sets of expectations are detailed below. Here, consider P to be the number of populations, $2N_e$ to be the effective population size in haplotypes and µ to be the per haplotype mutation rate. To keep the equations below consistent with conventional presentations, m is be the per capita migration rate, equal to one-half of the per haplotype migration rate .

infinite island, no mutation: $$F_{ST} = \frac{1}{4N_e m + 1}$$

The standard Wright infinite island model, without mutation (Hartl and Clark 1997).

infinite island, mutation: $$F_{ST} = \frac{1}{4N_e(m + \mu) + 1}$$

The standard Wright infinite island model, with infinite allele mutation (Hartl and Clark 1997).

finite island, no mutation: $$F_{ST} = \frac{1}{4N_e m\left(\frac{P}{P-1}\right)^2 + 1}$$

(Takahata and Slatkin 1984, Hartl and Clark 1997).

finite island, mutation: $$F_{ST} = \frac{(1-\mu)^2\left(1-m\left(\frac{P}{P-1}\right)\right)^2}{2N_e\left[1-(1-\mu)^2\left(1-m\left(\frac{P}{P-1}\right)\right)^2\right] + (1-\mu)^2\left(1-m\left(\frac{P}{P-1}\right)\right)^2}$$

(see Weir 1996).

Outfiles

"ESP OUT" Standard outfiles

The standard ESP outfile is a tab-delimited text file named "ESP OUT.**y**", where **y** is the simulation number (beginning with 1 when the program is launched). Its general format is as follows:

locus	# alleles in sample	PHIst	Theta	gen	parameter values	run time	start time
1	2	-0.0005	-0.0005	0	listed here . . .	(see below)	
2	3						
1	8	0.02955	0.02956	50			
2	14						
1	5	0.05148	0.05137	100			
2	14						
						<1 second	Fri Jan 28…

Calculation of summary statistics is described under Summary Statistics above.

Parameter values from the simulation are listed in their own columns, and are labeled as follows:

pops	Total number of populations in the simulation.
pops sampled	Number of population sampled for summary statistics.
2N	Number of haplotypes per population.
Sample size(haps)	Number of haplotypes sampled per population for summary statistics.
bases	Number of base pairs in each haplotype.
Nm	Gene flow, in terms of individuals per generation.
m (per haplotype)	Rate of gene flow per haplotype per generation.
µ/bp/gen	Mutation rate per base pair per generation
initial conditions	Initial value of F_{ST} or information regarding infile sampling
steps(unrelated alleles)	Number of mutational steps between alleles that were unique when the simulation began
run #	Replicate number
version	ESP version number
run time	Time required to complete the simulation. Run time appears in the final row of the table, rather than the second row. The date and time the simulation was started also appear.
start time	Date and time the simulation began.

Optional settings

Output will also contain allele frequencies, estimates of N_em, expected heterozygosity (total and (sub)population), and the number of alleles per population if the appropriate settings are enabled.

"ESP NX" NEXUS files

With the `NEXUS tree?` option enabled, ESP will save NEXUS code (Maddison et al. 1997) in a text file named "ESP NX**y**.**z**", where **y** is the simulation number (beginning with 1 when the program is launched) and **z** is the locus number. See Summary statistics: *All haplotypes*: NEXUS tree.

Confidence interval files

ESP confidence intervals are saved in a tab-delimited text file named "ESP CI **y**.**z**", where **y** is the simulation number (beginning with 1 when the program is launched) and **z** is the number of replicates. This file will only be written when the number of replicates is 40 or greater. It will contain mean and median values for all enabled summary statistics, 95% CIs, 99% CIs (if the number of replicates is 200 or more) and parameter values for the simulations (as described above for *Standard "ESP OUT" outfiles*).

Note that #all.(w/in pop) reflects the number of alleles within a population averaged over loci and populations. #all. (overall) is the per-locus average number of alleles over all populations. For example, three populations with fixed differences might have one allele each (within population) but there would be three alleles (overall) at that locus.

"ESP CI data" files

In the event that a set of confidence interval simulations is prematurely terminated, some of the replicates can be recovered. Every 50 replicates, ESP saves all of the data collected to that point in time in a file labeled "ESP CI data.**z**", where **z** is the number of the current replicate. (Each CI data file is a superset of the file preceding it).

To calculate confidence intervals other than those provided, or to pool data from different CI data files, the raw data can be imported into a spreadsheet application and sorted. A sample spreadsheet for Microsoft Excel is provided in the ESP folder.

When the gene flow estimation option is on, infinite gene flow (corresponding to $F_{ST} = 0$) is coded in the ESP CI data files as -99.

"arleq" and "TFPGA" DOS outfiles

DOS outfiles for the programs Arlequin (Schneider et al. 1998) and TFPGA (Miller 1998) are formatted according to the documentation that accompanies those programs. The replicate number appears in the file name, and some information on simulation parameters appears in the remarks section of each file.

"ESP freqs" infiles

ESP infiles are text-only files named “ESP freqs_**x**”, where **x** represents the replicate. In order to sample from these infiles for colonization/fragmentation scenarios, they must remain in the same folder as the application. The file format is follows on the next page:

```
ESP version 1.1 of Mar 4 2001, simulation saved Sun Mar 4 19:30:22 2001
0 generations, 1 bp, 0.200000 Nm, 1.000000e-06 μ, sampled 5 pops (65500 haplotypes each), 2 steps
b/w unrelated alleles
2 REPS, 5 POPS, 21 LOCI, 65500 N, 30 LEVELS_MAX

REPLICATE
POP 1
LOCUS 1
ALLELE 1 0 805 13630
1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
ALLELE 2 0 876 15702
2 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0

... text omitted

POP 5

... text omitted

LOCUS 20

... text omitted

ALLELE 10_803 1 0 122
10 803 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
LOCUS 21
ALLELE 5 0 2114 24812
5 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0

... text omitted

REPLICATE
POP 1
LOCUS 1
ALLELE 1 0 861 8751
1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0

... text omitted

ALLELE 3 0 688 11295
3 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
END FILE
```

The first two lines of the infile are provided for the benefit of the user, but are not utilized by ESP. They provide information regarding the simulation that generated the allele frequencies. If the file is altered, it must still retain two lines before the line that contains REPS, POPS, LOCI, N, and LEVELS_MAX. All lines are terminated with a simple carriage return.

The third line consists of "**v** REPS, **w** POPS, **x** LOCI, **y** N, **z** LEVELS_MAX", where

- **v** is the number of replicates in the infile
- **w** is the number of populations
- **x** is the number of loci
- **y** is the number of individuals per population
- **z** is the maximum number of haplotype levels (30 in most cases)

The fourth line is blank.

Each replicate begins with the single word REPLICATE on its own line. New populations and new loci are also announced one their own lines, as POP # and LOCUS # (see e example above). New alleles are announced within each population and locus as ALLELE followed by four fields separated by a single space each. Following the standard ESP nomenclature rules (see Allele nomenclature above), the first field consist of the allele's name, and the second is its level. An allele's "level" is the number of mutational steps between it and its zero level ancestor. The third field is the number of descendants (both extant and extinct) one mutational step from the allele. Thus, in the above example, allele 10_803 is the final allele in locus 20. 10_803 is a level 1 allele, and it had given rise to no new alleles (one mutational step away) when the simulation ended. This third field is important, because it assures that ESP will retain the correct genealogical relationships among alleles in the old simulation, and those which will arise when this file is used in the future as an infile. Without this information, allele 10 could give rise to another allele named "10_803" when this file is used in future simulations. As discussed above in Parameters and Settings: *Parameter limits*, the value in the third field will not exceed 9999. The fourth field following the word "ALLELE" is the number of haplotypes for that allele (e.g., there are 122 representatives of allele 10_803 in the above infile for population 5, locus 20). Within a population and locus, the sum of field four (across alleles) should equal **y** N; in this case, there should be 65,500 alleles for each locus in each population.

The second line of each allele consist of the allele's "name", with single spaces instead of single quotes. Zeros fill the remainder of the line, so that there are **z** LEVELS_MAX items on the second line of every allele.

The infile is terminated with the command "END FILE".

Infiles

"ESP freqs" infile

Infiles for the Generalized Colonization Algorithm can be generated from simulated data sets in prior runs (see above) or separately using a word processing program. If generated by the user, the infiles must be saved as "text only", named "ESP freqs_**x**", where **x** is a number, and be in the same folder as the application. The file format must follow that described in the preceeding section, except that alleles can be given any names. (However, ESP will ignore the user-specified name, and designate alleles according to the nomenclature described under *Allele Nomenclature*.)

Performance

ESP simulates the evolution of gene genealogies relatively quickly, but (in its current implementation) is very poor at sharing processor time. Response to mouse clicks may be *extremely* slow while ESP is active. When running large simulations for considerable lengths of time, we find it easiest to quit all other applications prior to running ESP, and simply abandon the computer until the program has finished. Under confidence interval mode, a projecting finishing time will be indicated on the screen.

To stop ESP in the middle of a simulation or set of simulations, forced quits are usually necessary (i.e., command-period or option-control-escape). In preliminary trials, ESP generally responds well to forced quits unless the system is sleeping. File sharing with computers running ESP seems to operate satisfactorily.

The following table summarizes ESP running times and RAM requirements for various parameter combinations on a 400 MHz G3 running MacOS 9.1. (ESP v. 1.12, print in final generation only, sample all alleles from all populations, all optional ESP calculations off, virtual memory off, AppleTalk off).

	populations		$2N_em$		base pairs		F_0		time	
generations		$2N_e$		μ		loci		replicates		RAM
0	10	200	0	0	1*	1	0.5	1000‡	1 second	4 MB
0	100	200	0	0	1*	1	0.5	1000‡	54 seconds	5 MB
500	100	2000	0.01	1 x 10^{-6}	1*	5	0.5	500‡	1.3 hours	8 MB
1000	10	200	0.1	1 x 10^{-6}	1*	2	0.1	1	1 second	4 MB
1000	10	200	0.1	1 x 10^{-6}	1*	2	0.1	100‡	49 seconds	4 MB
1000	100	200	0.1	1 x 10^{-6}	1*	2	0.1	100‡	15.6 minutes	6 MB
1000	100	2000	0.1	1 x 10^{-6}	1*	2	0.1	1	25 seconds	5 MB
1000	100	2000	1 x 10^{-4}	1 x 10^{-8}	1000	1	≈0.18***	100‡	14.3 minutes	9 MB
10,000	100	2000	1	1 x 10^{-6}	1*	10	0**	1	2.8 minutes	4 MB
10,000	100	2000	1	1 x 10^{-6}	1*	10	0**	100‡	4.7 hours	5 MB
100,000	100	20000	0.01	1 x 10^{-8}	1000	1	0.5	1	1.6 hours	9 MB
1,600,000	100	2000	1 x 10^{-4}	1 x 10^{-8}	1000	5	1.0	1	6.1 hours	10 MB

* represents a single locus

** each locus began with two alleles at frequencies of 0.5 in all populations.

*** random colonizations from an infile with 10 replicates (100 populations and 2000 haplotypes per replicate. Average of 9.4 alleles per population. Population bottleneck of 20 colonizing haplotypes/descendant population. All "strict" sampling options for colonizations off. "Single source scenario" option off.)

‡ denotes simulations run using the Confidence Interval option.

In development

1. Windows version.
2. Locus-specific F-statistics will be saved.
3. Stepwise mutation (for microsatellites)
4. Improved background performance.
5. Automatic graphing on the screen.
6. Stepping-stone model migration and pairwise calculations of divergence. These are useful for examining geographic patterns of genetic variation in isolation by distance plots (sensu Slatkin).

Programming

ESP has been written in C and compiled using CodeWarrior for Macintosh. Output (via CodeWarrior's SIOUX module) is extremely simple, but we hope the lack of glitz is sufficiently balanced by the program's flexibility and speed. Source code will be made available upon request.

Acknowledgments

We would like to thank Neil Davies and Francis Villablanca for constructive commentary. This work was supported by grants to AJB and GKR and NSF, USDA and CDFA.

Literature cited

Avise, J. C. 1994. Molecular markers, natural history and evolution. Chapman and Hall, New York, USA.

Bohonak, A. J. 1999. Dispersal, gene flow and population structure. Quarterly Review of Biology **74:** 21-45.

Bohonak, A. J., N. Davies, F. X. Villablanca, and G. K. Roderick. *in press*. Invasion genetics of New World medflies: testing alternative colonization scenarios. Biological Invasions.

Cockerham, C. C., and B. S. Weir. 1987. Correlations, descent measures: drift with migration and mutation. Proceedings of the National Academy of Sciences of the United States of America **84:** 8512-8514.

Excoffier, L., P. E. Smouse, and J. M. Quattro. 1992. Analysis of molecular variance inferred from metric distances among DNA haplotypes: application to human mitochondrial DNA restriction data. Genetics **131:** 479-491.

Hairston, N. G., Jr., L. J. Perry, A. J. Bohonak, M. Q. Fellows, C. M. Kearns, and D. R. Engstrom. 1999. Population biology of a failed invasion: paleolimnology of *Daphnia exilis* in upstate New York. Limnology and Oceanography **44:** 477-486.

Hartl, D. L., and A. G. Clark. 1997. Principles of population genetics. 3rd edition. Sinauer Associates, Sunderland, MA, USA.

Maddison, D. R., D. L. Swofford, and W. P. Maddison. 1997. NEXUS: an extensible file format for systematic information. Systematic Biology **46:** 590-621.

Miller, M. P. 1998. TFPGA: Tools for population genetic analyses for Windows. Arizona State University, USA.

Page, R. D. M. 1996. TREEVIEW: An application to display phylogenetic trees on personal computers for Computer Applications in the Biosciences 12: 357-358., Macintosh.

Roderick, G. K. 1996. Geographic structure of insect populations: gene flow, phylogeography, and their uses. Annual Review of Entomology **41:** 325-352.

Schneider, S., J.-M. Kueffer, D. Roessli, and L. Excoffier. 1998. Arlequin: a software for population genetic data analysis for Windows. Geneva, Switzerland.

Slatkin, M. 1985. Gene flow in natural populations. Annual Review of Ecology and Systematics **16:** 393-430.

Slatkin, M. 1994. Gene flow and population structure. Pages 3-17 *in* L. A. Real, editor Ecological genetics. Princeton University Press, Princeton, NJ.

Slatkin, M., and H. E. Arter. 1991. Spatial autocorrelation methods in population genetics. American Naturalist **138:** 499-517.

Takahata, N., and M. Slatkin. 1984. Mitochondrial gene flow. Proceedings of the National Academy of Sciences of the United States of America **81:** 1764-7.

Wade, M. J., and D. E. McCauley. 1988. Extinction and recolonization: their effects on the genetic differentiation of local populations. Evolution **42:** 995-1005.

Weir, B. S. 1996. Intraspecific differentiation. Pages 385-405 *in* D. M. Hillis, C. Moritz and B. K. Mable, editors. Molecular systematics. Sinauer Associates, Sunderland, MA.

Weir, B. S., and C. C. Cockerham. 1984. Estimating F statistics for the analysis of population structure. Evolution **38:** 1358-1370.

Wright, S. 1931. Evolution in Mendelian populations. Genetics **16:** 97-159.