Allele Surfing 3-11-17

1 The Simualtion:

Each diploid individaul consists of $nHybrid = 1, 2$ hybrid loci and $nOther$ other loci. Each locous is bi-allelic. I use the 0 allele as the wild type and the 1 allele as the mutationally derived copy whereever I can. The simulation has a chain (vector) of $nPop$ popualtions. Currently the carrying capacity of each popualtion is the same non-fitness dependent constnat, K. Once initalized individuals progress throguh a life cycle of migration-mating-selection. The total number of generations is given by $gmax$.

Migration: Individuals migrate to a neighboring popualtion with probability m, the direction of movement is non-dirctional. Edge popualtions have "reflecting boundaries" moving right and left with rates of $\frac{m}{2}$. From a programming standpoint over the course of migration individuals are copied from poulation 1 to popuatlion 2. I know it would probability be more efficentl to do a linked list where I can simply move individuals rather than copy their data but I am not very good at doing this so it would take me a while to program it this way so I just stuck with the copying.

Mating: After migration individuals mate. Each individual produces a poisson distributed number of offspring with mean $\lambda = 1 + R * (1 - N_p(t)/K)$ offspring, where $N_p(t)$ is the number of individuals in popualtion p at time t. Here R is the maximum growth rate, such that in the abcences of competition individudals produce $1 + R$ childern on average. I am not sure I am doing this right. For some reason, even in the absence of selection (or atleast what should be an abscence of selection) popuatlion sizes tend to remain below the carying capacitiy. To produce each offspring the other parent is chosen at random from the popualtion (and can technically include the current parent- ie selfing). Recombination is determined by three values; the rate between hybrid loci, the rate between hybrid loci and other loci and the rate between other loci. Current simuations have free recombination. In addition to recombiantion mutation at other loci occurs at a rate mu. There is no mutation at the hybrid loci.

Selection: The specifics of fitness are going to be determined by the particular scenerio. But in all cases after offspring are produced they undergo "selection". Their absolute survival fitnesses (Probabilities of survival) are calculated. To make sure that this probability remains between 0 and 1, I chose to have the fitness effects at differnt loci be mutliplicative. For the "other loci" the wildtype allele has by defintion no deleterious fitnesses effect whereas the fitness effect of a derived "1" allele has a deleterious effect s. Similarly the deleterious fitness consequences of the hybrid genotype is given by the matrix $sMtrx$ where element i, j gives the fitness effect of have haplotye combiantion ixj. Here i and$j$ denote haplotypes by their bits such that $i = 0$ is the $0, 0$ haplotype and $i = 1$ is the $1, 0$ haplotype and so forth. Regardless of whether individuals have 1 or 2 hybrid loci I still store the consequent fitnesses in a $4x4matrix$ except only a few elements have non-zero values. The resulting expression for the proabblity of survival of an individual with hybrid haplotypes i and j is given by:

$$w = \mathbf{s}^H_{i,j} \prod_{k=1}^{n_{other}} (1 - s^O_k) \tag{1}$$

1.1 Different Scenerios

1.1.1 Scenerio 1: Expansion without load

In this scenerio only the left most popualtion is initalized to have any individuals. Currently I have simply set the left most populations inital size to the carrying capcity but I can change this as I test for logistic growth (see below). All individuals have the wild-typae "0" allele at all loci, both hybrid-loci and other-loci. Although mutation can occur the fitnesses effects of mutant "1" alleles are set to 0 $s^0_i = 0$, and hybrid genotypes have no fitnesses effects either.

1.1.2 Scenerio 2: Expansion with load

The initalizaiton of this scenerio is very similar to scenerio 1 except now the there are deleterious effects of mutant "other" loci. For now I am simply drawing these effects uniformly between 0 and 1.

1.1.3 Scenerio 3: Underdominance+ no expansion load

1.1.4 Scenerio 4: Underdominance+ expansion load

1.1.5 Scenerio 5: BDMI+ no expansion load

1.1.6 Scenerio 6: BDMI+ expansion load

1.2 Recording coalescent histories

2 Checking the Simuation:

Currently the plan is to check the simulations using a variety of analytical results based on coalescent times and subsequent mesures such as F_{st} and $\pi_{i,j}$ (the distribution of pairwise differences).

2.1 Check 1: Coalescent times variable sized popuatlions:

Grifiths and Tavere 1994 provide an method for calculating the distribution of coalescent times in a popuatlion with variable size. This can be done in the following way. Let $M(t)$ be the popualation size at real time t, we will scale the coalescent by a constant rate N which we can convienently choose to equaal $N = M(0)$. Then define the relative time scaled popualtion size $v_N(\tau) = \frac{M(N\tau)}{N}$. Here $\tau = \frac{1}{N}t$ is the coalescent time. To be a true coalescent approximation we consider cases where N is large. If the coalescent times for n sampled lienages (in the coalescent time scale) be $\tau_2, \tau_3, ...\tau_n$ we define the following functions:

$$\Lambda(x) = \int_0^x \frac{1}{v_N(t)} dt \tag{2}$$

and

$$s_j = \begin{cases} \sum_{k=j}^{n} \tau_k & if j < n \\ \tau_n & if j = n \\ 0 & otherwise \end{cases} \tag{3}$$

then the joint distribution of coalescent times is given by:

$$g(\tau_2, ..\tau_n) = \prod_{j=2}^{n} \binom{j}{2} \frac{1}{v_N(s_j)} e^{-\binom{j}{2}(\Lambda(s_j) - \Lambda(s_{j+1}))} \tag{4}$$

In addition they go on to discribe the distribution of genetic variation from such a coalescent history under both the infinite sites and infinite alleles neutral models. I hope to use this equaiton to test the simulation in several ways.

2.1.1 Constant population size

First the most trivial test is to assess whether the observed distribution of mutational effects matches that expected by (4). Most recently I tried to do this by running a single simualtion of a single popaultion, recording the nearly constant population sizes throughout the simulation, and recording the full coalescent history of the popualtion. I then repeatedly sampling sets of lineages from the resulting coalescent history. This however will not work beucase the samples share a common coalescent history and hence will not represent the fully expected distribution of coalescent times. Hence I am going to redo this by running multiple simulations and sampling a single set of lineages from each resulting full coalescent history. However to do this I need to somehow constrain the simulations so that they share teh same demographic history, I can do this if I rewrite the matting function such that each individual produces a deterministic number of offspring rather than a poisson distributed number. Although I may want to do this anyway I want to check with Liz about how she is running her simulations to make sure this is lagitamate.

2.1.2 Logistically growing population

Once I get the constant popualtion size reuslt to work out I can then check a logisitcally growing popualtion. In other words check the reuslts where the popualtion does not start at carrying capacity.

2.1.3 Neutral variation at other loci

Finally, I can check the distribuiton of neutral genetic variation that arises as a result of new mutation by using the predictions of the infinite-sites model presented by Griffiths and Tavere. Although our simualtions are not truely infinite sites as was Kirkpatric and Piechal's this should be approximately close if I use a large number of loci.