User Manual for Mendel's Accountant

Contents:

- Welcome
- MENDEL's Basic Principles of Operation
- Computer Requirements
- Downloading Instructions
- Opening and operating MENDEL
- MENDEL Input Parameters
- Basic Parameters
- Advanced Parameters:
  - Mutation
  - Selection
  - Population Biology
  - Population Substructure
  - Computation
  - Special Applications
- MENDEL Output
- Applications of MENDEL
- MENDEL Glossary

Welcome to *Mendel's Accountant*.

This user's manual provides an introduction to the program *Mendel's Accountant*. You may download the program at http://sourceforge.net/projects/mendelsaccount.

Mendel's Accountant (MENDEL) is an advanced numerical simulation program for modeling genetic change over time and was developed collaboratively by Sanford, Baumgardner, Brewer, Gibson and ReMine.

MENDEL is a genetic accounting program that allows realistic numerical simulation of the mutation/selection process over time. MENDEL is applicable to either haploid or diploid organisms, having either sexual or clonal reproduction. Each mutation that enters the simulated population is tracked from generation to generation to the end of the experiment - or until that mutation is lost either as a result of selection or random drift. Using a standard personal computer, the MENDEL program can be used to generate and track millions of mutations within a single population.

MENDEL's input variables include such things as mutation rate, distribution specifications for mutation effects, extent of dominance, mating characteristics, selection method, average fertility, heritability, non-scaling noise, linkage block properties, chromosome number, genome size, population size, population sub-structure, and number of generations.

The MENDEL program outputs, both in tabular and graphic form, provide several types of data including: the history of deleterious and beneficial mutation counts per individual, the history of mean individual fitness, distribution of accumulating mutation effects, selection threshold history, distribution of linked mutation effects, fitness distributions before and after selection, and allele frequencies.

MENDEL provides biologists with a new tool for research and teaching, and allows for the modeling of complex biological scenarios that would have previously been impossible.

MENDEL's Basic Principles of Operation.

The actual design and code features of MENDEL are described in detail elsewhere *[Sanford et al., 2007. SCPE 8(2): 147-165 - also available on this web site]*. Following is a simple outline of how the program operates.

1. Based on user input, Mendel creates a virtual population with the specified number of individuals, mean reproduction rate, mating characteristics, and possible population sub-structure. Also specified are the genome size, number of chromosomes, and linkage dynamics.
2. Based on user input, Mendel creates a pool of potential mutations with precisely specified characteristics including: range and frequency distribution of mutation effects, ratio of recessives to dominants, and fraction of beneficial mutations.
3. Mutations are selected randomly from the pool of potential mutations and are assigned randomly to the new offspring in each generation, based upon the specified average mutation rate (Poisson distribution).
4. Individual genetic fitness values are calculated based upon each individual's total mutation inventory. Individual genetic fitness is defined as 1.0, adjusted by the positive and negative effects of all its mutations. To obtain phenotypic fitness the genetic fitness is modified using the specified heritability to account for non-heritable factors such as variations in the environment.
5. Based on phenotypic fitness values, Mendel applies selection of a specified type (probability, truncation, etc.) to eliminate a specified fraction of the individuals from the mating pool. The fraction removed is determined by the fertility of the population (average number of offspring per surviving individual). Offspring in excess of the base population size represent the population surplus that is eliminated via selection based on individual phenotypic fitness scores. Unless otherwise specified, population size is kept constant by selecting away the entire surplus population each generation.
6. Gametes are extracted from those individuals that survive the selection process and will reproduce. These gametes are generated based upon a sampling of the linkage blocks within the chromosome pairs of each reproducing individual. Reproducing individuals are paired off randomly, and their gametes are fused to create the next generation of individuals. For asexual populations, each individual remaining after selection produces one or more clonal offspring.

Steps 1-6 are repeated, for the specified number of generations. Output reports and plots are updated at regular intervals.

Computer Requirements.

MENDEL was designed to run on most computer platforms and is currently supported on Microsoft Windows, most Linux systems, and Mac OS X. A recommended hardware requirement is a minimum of 512GB of RAM. Although MENDEL can run with less RAM, the size of a run (largely determined by population size and mutation rate), is limited by the available memory. The Mac and Linux versions are currently superior versions to Windows because the Windows version has more limited capabilities. However, the Windows version should be very useful in teaching environments. For both versions, performance is significantly enhanced when a dual processor (or dual core) machine is used, because the graphical user interface does not need to compete for resources with the Mendel program itself. A stand-alone multi-core processor machine is therefore strongly recommended.

Downloading Instructions.

MENDEL can be downloaded by clicking through a series of green download buttons at http://sourceforge.net/projects/mendelsaccount. You must specify whether you want the Linux, Mac OS X, or Windows version. If "Downloading" appears on the screen but the download does not appear to be proceeding – click "this direct link" which is shown in blue. Once downloaded, carefully follow the instructions for installation.

Opening and operating MENDEL.

MENDEL is controlled through its own web-based graphical user interface. Therefore, to access MENDEL, one must open a web browser. (*Google Chrome* and *Apple Safari* are recommended. *Internet Explorer* is not supported.)

- To open the program one must click on the icon (Windows), or one can enter the following address in the browser -- http://127.0.0.1:8080/mendel/vX.Y.Z/index.html, or select the website for Linux systems (e.g. http://myservername.com/mendel/vX.Y.Z/index.html), where XYZ represents the major, minor, and patch level numbers (e.g. 2.0.2). or select the local website for Linux systems (e.g. http://myservername.com/mendel).
- To start a fresh run, simply press "Start", and begin to fill in the parameter options as desired (initially there are default input parameters in all boxes).
- To start a run using most or all of the parameters from a previous run, enter the previous run's case ID in the box (panel of options on left of screen), and then press "Start". The input options will now have the parameters from that earlier run, which can then be modified as desired prior to execution.
- Once the parameters have all been entered, press "submit". Review the selected options, and if necessary use the browser "back" button to go back and modify the parameters.

Check to see if the required memory is greater than the available memory before pressing the “execute” button. If the “required” memory is greater than the available memory, in many cases the run will still be completed successfully, but in some cases it will fail to execute or will “crash” before the run is complete.

- If the run has been successfully activated, a “Run Status” button can be activated, specifying the run’s number and it time of initiation, and its progress.
- During the run, one can periodically refresh and examine the output file (click “Output” button).
- During the run, one can periodically refresh and examine the output figures (click “Plot” button).
- During the run one can review the “Input Parameters”, or check the “Run Status”. (Run Status not available in downloadable version.)
- If desired, one can open “Run Status”, and terminate a run by selecting the current run and then selecting “Stop”. (This option requires using a PBS queueing system and thus is not available in downloadable version).
- During or after a run, one can click “Label” and add comments to a run, for future reference.

MENDEL Input Parameters.

MENDEL’s default input values are parameters that might apply to a small human population. MENDEL’s input parameters have been separated into two categories: ‘basic’ and ‘advanced’.

Basic Parameters:

The basic input parameters include the variables that most obviously affect mutation accumulation, and are most frequently changed. This parameter set is useful for initial familiarization with Mendel, and for training beginning students.

- **Case ID**. Since MENDEL allows each run to be saved, each experiment needs to be identified for the purpose of data retrieval. The ID must be alphanumeric and must have exactly 6 characters. We recommend users begin their case ID with their own initials.
- **Mutation rate**. This is the average number of new mutations per individual. In humans, this number is believed to be approximately 100. The mutation rate can be adjusted to be proportional to the size of the *functional* genome. Thus if only 10% of the human genome actually functions (assuming the rest to be biologically inert), then the biologically relevant mutation rate would be just 10. Rates of less than 1 new mutation per individual are allowed - including zero. The human default value is 10 new mutations per individual per generation.
- **Ratio of beneficial versus deleterious mutations**. While some sources suggest this number might be as high as 1:1000, most sources suggest it is more realistically about 1:1,000,000. The default setting is 1:10,000. For studying the

accumulation of only deleterious or only beneficial mutations, the number of beneficials can be set to zero or one.

- **Reproduction rate**. This is the number of offspring per reproducing individual. Since population size in Mendel is usually constant, this variable defines the maximum amount of selection. There must be an average of at least one offspring per individual (after the selection process) for the population to maintain its size and avoid rapid extinction. Except where random death is considered (see advanced parameters), the entire surplus population is removed based upon phenotypic selection. The default value for humans is two offspring per selected individual (or four offspring per reproducing female).
- **Population size**. This is the number of reproducing adults, after selection. For parallel runs, this is the population size of each sub-population. This number is normally kept constant, except where fertility is insufficient to allow replacement, or where certain advanced parameters are used. For smaller computer systems such as PCs, population size must remain small (100-1000) or the program will quickly run out of memory. The default value is 1,000, since population sizes smaller than this can be strongly affected by inbreeding and drift. We find increasing population size beyond 1000 results in rapidly diminishing selective benefit.
- **Number of generations**. The number of generations the program should run. The default is 500 generations. If there are too many generations specified, smaller computers will run out of memory because of the accumulation of large numbers of mutations, and the experiment will terminate prematurely. This problem can be mitigated by tracking only the larger-effect mutations (see advanced computation parameters). The program also terminates prematurely if fitness reaches a specified extinction threshold (default = 0.0) or if the population size shrinks to just one individual.

Advanced Parameters:

Advanced Parameters: The advanced parameter settings are for knowledgeable researchers and more advanced students, and should be set to apply to a specific species or circumstance. The advanced parameters are distributed among the general categories of: Mutation, Selection, Population Biology, Population Sub-structure, and Computation.

Advanced Mutation Parameters

- **Parameters shaping the distribution of deleterious mutations**. Deleterious mutations in the natural world typically range from a few rare lethals to a large number of nearly-neutral and neutral mutations. It is widely agreed that the distribution of mutational effects is characterized by an exponential-like function, where there are few high-impact mutations and many mutations which are nearly-neutral. Mendel uses a generalized exponential function, called the Weibull function, to generate its distribution of mutation effects ranging from 1 (lethal) down to nearly 0 (near-neutral). See Sanford et al., SCPE 8(2) p.147-165 available at http://scpe.org), for the mathematical formula that describes Mendel's mutation effect distribution, and also for an explanation of the difference between “fitness effect” and “selection coefficient”. The exact shape of the mutation effect distribution produced by MENDEL can be controlled precisely, using the variables shown below. A special-case mutation distribution is also provided, which is never seen in nature but is of theoretical interest - wherein each deleterious mutation has exactly the same effect on fitness (as specified by the user). Likewise each beneficial mutation has exactly the same effect on fitness (as specified by the user). We are also developing a special bi-modal distribution of mutation effects.

- **Functional genome size** - If we know the genome size of a species, and we subtract that fraction of the genome which is perfectly inert, we can know the size of the "functional genome". The relative abundance of near-neutrals depends, in part, upon the functional genome size. A simple viroid with a genome size of just 100 nucleotides may have very few nucleotide positions which when mutated will have only miniscule biological effects. In such a viroid, we might assume that a low-impact deleterious mutation would generally reduce genomic information by (very roughly) about one part in a hundred. Alternatively, a human genome of three billion will have very many near-neutral positions, and a typical low-impact mutation might (very roughly) reduce genomic information by only one part in three billion. Therefore, it is obvious that the distribution of deleterious mutational effects must in some way be adjusted to account for genome size. An approximate yet reasonable means for doing this is to define the minimal mutational effect as being 1 divided by the functional haploid genome size. The result of this adjustment is that smaller genomes have “flatter” distributions of deleterious mutations, while larger genomes have “steeper” distribution curves. Because we consider all entirely neutral mutations separately, we only consider the size of the functional genome, so we choose the default genome size to be 300 million (10% of the actual human genome size).
- **Fraction of mutations having “major effect”** - Most mutations have an effect on fitness that is too small to measure directly. However, mutations will have measurable effects in the far “tail” of the mutation distribution curve. By utilizing the frequency and distribution of “measurable” mutation effects, one can constrain the most significant portion of the distribution curve as it relates to the selection process. For most species, there may not yet be enough data, even for the major mutations, to accurately model the exact distribution of mutations. When such data is not yet available, we are forced to simply estimate, to the best of our ability and based on data from other organisms, the fraction of “major mutations”. The human default is 0.001.

    - **Cut-off point for defining “major effect”** - A somewhat arbitrary level must be selected for defining what constitutes a “measurable”, or “major”, mutation effect. MENDEL uses a default value for this cut-off of 0.10. This is because under realistic clinical conditions, it is questionable that we can reliably measure a single mutation’s fitness effect when it changes fitness by less than 10%.
    - **Maximal beneficial mutation effects** — Parameters shaping distribution of beneficial mutations. The distribution of beneficial mutations should generally be a mirror image of the distribution of the deleterious mutations, except that the area under the distribution curve should be adjusted to reflect the proportionately lower number of beneficial mutations compared to deleterious mutations. Since the distribution of beneficials should be affected by genome size (as with deleterious mutations), it is useful to likewise define the minimal beneficial mutation effect as 1 divided by the functional haploid genome size. In addition, beneficials should have a reduced upper range.

      A realistic upper limit must be placed upon beneficial mutations. This is because a single nucleotide change can expand total biological functionality of an organism only to a limited degree. The larger the genome and the greater the total genomic information, the less a single nucleotide is likely to increase the total. Researchers must make a judgment for themselves of what is a reasonable maximal value for a single base change. The MENDEL default value for this limit is 0.01. This limit implies that a single point mutation can increase total biological functionality by as much as 1%. In a genome such as man's, assuming only 10% of the genome is functional, such a maximal impact point mutation might be viewed as equivalent to adding three million new information-bearing base pairs each of which had the genome-wide average fitness contribution. Researchers need to honestly define the upper limit they feel is realistic for their species. However it should be obvious that, in all cases, the upper limit for beneficial mutation effects ought to correspond to a very small fraction of the total genomic information (i.e. a small number relative to one). Mutations such as antibiotic resistance need to be handled separately, as uploaded mutations.

  - **Parameters involving recessive and dominant mutations.** It is widely agreed that in diploid species, most mutations are recessive, while a small fraction are dominant. However, because modeling recessives and dominants can become computationally more intense, and because the output figures can become hard to read when plotting both dominant and recessive mutations, the default setting is co-dominance. This means that all mutations behave additively (a heterozygote will always have half the effect of a homozygote). However, for greatest realism, the majority of mutations should be made recessive, with a minority being dominant by default, as described below:

    - **Fraction of mutations recessive** – This parameter simply specifies the percentage of mutations that are recessive. The default is 0.0%. If set to 80%, then 80% of mutations are recessive, so the remaining 20% will automatically be made dominant.
    - **Recessive expression in heterozygotes** – It is widely believed that recessive mutations are not completely silent in the heterozygous condition, but are still expressed at some low level. Although the co-dominance default is 0.5 expression, a reasonable setting would be 0.05.
    - **Dominant expression in heterozygotes** - It is widely believed that dominant mutations are not completely dominant in the heterozygous condition, but are only expressed only at some very high level. Although the co-dominance default is 0.5, a reasonable setting would be 0.95.
  - **Combining mutational effects non-additively** — When there are two or more mutations within an individual, the effects of these multiple mutations must be combined. The most straightforward way to do this is additively, by just adding up the effects of all the deleterious and beneficial mutations within an individual, and adjusting original fitness (initially 1.0) by that net amount. Alternatively, one can adjust fitness by multiplying the fitness (initially 1.0) by the net effect of each mutation (the net effect of a single mutation would be one minus the fitness effect of that mutation). This multiplicative method is quite commonly used in population genetics, although in our experience it seems inadequate when modeling reality - since in this model no amount of deleterious mutation can drive fitness completely to zero. The reason for this is that in the multiplicative model, each additional deleterious mutation has less and less effect on absolute fitness. For this input parameter, the researcher can select an all additive model (0.0 multiplicative = default), or an all multiplicative model (1.0, no additive component), or a mixed model having any intermediate value between 0 and 1.0. MENDEL's default setting is the simple additive method. A third way to combine mutational effects is to use a synergistic epistasis model (see #6 below).
  - **Special case: synergistic epistasis (SE)**

    In modeling synergistic epistasis (SE) in Mendel, we distinguish between SE contributions from deleterious mutation pairs which are linked together within a linkage block on a chromosome from those which are not. Linked mutations are inherited together, and therefore the SE effects of all their mutual interactions are as well. By contrast, genetic recombination progressively tends to scramble mutations that are not linked together. Hence, the total SE contribution from non-linked mutations has a transient component. The SE effects arising from the non-linked interactions which change from one generation to the next act like a type of noise that interferes with the selection process. Therefore, realistic modeling of SE requires that linked and non-linked SE effects be treated separately. We therefore partition the SE effects on fitness into two parts, one involving interactions between mutations which occur in the same linkage block (linked interactions) and the other part involving interactions of mutations on different linkage blocks (non-linked interactions). SE effects from linked interactions are inherited, while part of those from non-linked interactions are transient and act, in effect, as a type of noise as far as the selection process is concerned.

Another difference between linked and non-linked interactions is the relative magnitude of their SE effects. Logically, the strongest SE interactions should be within the same linkage block, even as two misspellings in an encyclopedia are likely to interact more strongly if they occur within the same paragraph or sentence or word. Two mutations are most likely to interact if they occur within the same protein-coding sequence or at least the same genic region. Therefore, the treatment in Mendel includes separate scaling factors for each of these two categories of SE effects, se_linked_scaling and se_nonlinked_scaling. Normally, the scaling factor for linked interactions, se_linked_scaling, should be much larger (for example, by a factor of 1000) than the one for non-linked interactions, se_nonlinked_scaling.

Let us now consider how Mendel actually treats the linked SE interactions. We assume the amplitude of the linked SE effect of each pair-wise interaction to be proportional to the product of non-epistatic fitness effects of the two mutations in the pair. This means that if a mutation's effect on the non-mutant genome is small, then the SE contribution from its interactions with other mutations likewise is small. Whenever a new mutation is added to a given linkage block, Mendel computes and accumulates the SE contribution to linkage block fitness for that new mutation. This contribution is proportional to the non-epistatic fitness effect of the new mutation times the sum of the non-epistatic effects of each of the individual mutations already present on the block. As these SE contributions are accumulated, each of the m(m-1)/2 unique pair-wise interactions is accounted for, where m is the number of mutations in the linkage block. These contributions, as mentioned above, are scaled by a user-specified parameter se_linked_scaling. Also for these SE interactions, we assume co-dominance, which implies each haploid occurrence of a mutation gives 50% expression of the mutation's total non-epistatic value. This reduces the SE effect by a factor of 0.25. We note that, because mutations within a given linkage block are passed intact from one generation to the next, the SE effects arising from linked mutations are also passed intact from parent to offspring. Note that because linked SE interactions are inherited perfectly, they must always make the degeneration problem worse. This is because the negative SE contributions add to the negative non-epistatic fitness effects of the mutations on each linkage block, and in effect make the non-epistatic effects even more negative.

Mendel treats the non-linked SE interactions in a similar manner. Let M be the total number of mutations in the genome of a given member of the population and n be the number of equal-sized linkage blocks. The total number of unique pair-wise interactions between mutations is M(M-1)/2, the mean number of mutations per linkage block is M/n, and the approximate number of linked interactions is n(M/n)[(M/n)-1]/2 = M(M-n)/2n. With this approximation, the number of non-linked interactions becomes (1 - 1/n)M^2/2 and the ratio of the number of non-linked interactions to linked ones is n-1/(1-n/M). With n typically 1000 or greater, as M becomes much greater than n, this ratio approaches n. In other words, as the total number of mutations becomes large relative to n, the number of non-linked mutations approaches n times the number of linked mutations.

Let us denote by F the overall genotypic fitness, apart from any SE effects, of a given individual member of the population. We assume the amplitude of the non-linked SE effect of each pair-wise interaction to be proportional to the product of non-epistatic fitness effects of the two mutations in each pair. The total non-linked SE fitness contribution is then proportional to the sum of the non-epistatic fitness effects of all the individual mutations, (1-F), but scaled to account for the portion of the mutations which are linked using the factor (1 - 1/n), times the mean non-epistatic fitness effect of these mutations, (1-F)/M, times the number of unique pair-wise interactions, (1 - 1/n)M/2, that each non-linked mutation has with the others. We again assume co-dominance, which implies each haploid occurrence of a mutation gives 50% expression of the mutation's non-epistatic value. This reduces the overall contribution by a factor of 0.25. We scale this non-linked SE contribution with a user-specified input parameter se_nonlinked_scaling. As already mentioned, one expects that interaction between mutations within the same linkage block will on average have much greater SE effects than mutations which are more distant within the genome. Hence, a value for se_nonlinked_scaling should typically be much less (by a factor of 0.001 or less) than se_linked_scaling. The resulting expression for the non-linked SE contribution to individual fitness, to be subtracted from F, is therefore se_nonlinked_scaling times 0.125$(1\text{-}F)^2(1 - 1/n)^2$.

We note that the negative SE contribution to fitness from all the non-linked interactions is proportional to $(1\text{-}F)^2$. Since the number of linkage blocks is typically 1000 or greater, the factor $(1 - 1/n)^2$ can usually be approximated as unity. The SE contribution from non-linked interactions is larger for individuals in the population with lower fitness and smaller for individuals with higher fitness. It therefore tends to accentuate the spread in fitness across the population and thus to enhance selection efficiency. Since the mean mutation fitness effect is directly proportional to (1-F), the overall impact of this SE contribution from non-linked interactions is to increase the mean negative mutational fitness effect, just as is the case for the SE contribution from the linked interactions. Therefore, the net effect of SE for both linked and non-linked interactions should be a higher rate of fitness decline with time.

What sorts of absolute values for the parameters, se_linked_scaling and se_nonlinked_scaling are biologically realistic? If we assume no linkage at all, the total SE fitness contribution (all from non-linked mutations) is given by the expression se_nonlinked_scaling times 0.125$(1\text{-}F)^2$, where F is the genotypic fitness. A realistic limit on the magnitude of se_nonlinked_scaling might be the value that drives F to zero when, without SE, the fitness F is 0.5. In this case, se_nonlinked_scaling = 0.5/(0.125x0.5^2) = 16. This means that, if the accumulated mutations in a given individual reduce its fitness to 0.5 without SE, then with SE and se_nonlinked_scaling = 16, the fitness of this individual drops to zero. In our view, a biologically realistic value for se_nonlinked_scaling should therefore be on the order of 1.0 or less and for se_linked_scaling 1000 or less. In our numerical experiments we see discernable SE effects only when we use SE scaling parameters a factor of 10 or more larger when selection intensity is weak and only with parameters a factor of 100 or more when selection intensity is moderate. In other words, SE appears to play a negligible role in most biologically realistic situations.

- **Uploading Specific sets of mutations** — A specific set of mutations can be uploaded into the population before a run begins. When this option is selected a template is shown which can be used to identify mutations for uploading, or a set of mutations can be pasted into a template.
- **Allow back-mutations** — In a large genome, the rate of back mutations (mutations that arise at nucleoside sides that have already mutated), is vanishingly small and of no consequence, but in small genomes (i.e., viruses), a significant fraction of the genome can become mutated, such that this parameter becomes useful.

Advanced Selection Parameters

- **Rate of "Random Death".** A certain fraction of any population fails to reproduce, independent of phenotype. This can be expressed as the percentage of the population subject to random death. This is a useful parameter conceptually, but the same effect can be obtained by proportionately decreasing the number of offspring/female, so the default is zero.
- **Heritability.** Because a large part of phenotypic performance is affected by an individual's circumstances (the "environment"), selection in nature is less effective than would be predicted simply from genotypic fitness values. Non-heritable environmental effects on phenotypic performance must be modeled realistically. MENDEL's default value for the heritability is 0.2. This implies that on average, only 20% of an individual's phenotypic performance is passed on to the next generation, with the rest being due to non-heritable factors. For a very general character such as reproductive fitness, 0.2 is an extremely generous heritability value. In most field contexts, it is in fact usually lower than this, typically being below the limit of detection.
- **Non-scaling noise.** If a population's fitness is increasing or declining, heritability (as calculated in the normal way), tends to scale with fitness, and so the implied "environmental noise" diminishes or increases as fitness diminishes or increases. This seems counter-intuitive. Also, with truncation selection, phenotypic variance becomes un-naturally small. For these reasons, it is desirable to model a component of environmental noise that does not scale with fitness variation. The units for this non-scaling noise parameter are based upon standard

deviations from the initial fitness of 1.0. For simplicity, the default value is 0.05, but reasonable values probably exceed 0.01 and might exceed 0.1. This means that there will always be at least a population phenotypic fitness standard deviation of 0.05 even with a heritability of 1.0. The specified heritability and non-scaling noise both represent non-heritable variation, so non-scaling noise makes the effective heritability lower than the specified heritability. Mendel outputs the standard deviation for genotypic and phenotypic fitness before and after selection (in the .hst file), so effective ("realized") heritability can be calculated if the user so desires.

- **Fertility declines as fitness declines.** It is widely recognized that when fitness declines, fertility also declines. This in turn affects population surplus, which affects selection efficiency, and can eventually result in "mutational meltdown". To model this, we have included an option wherein fertility declines proportional to the square of the fitness decline. The resulting fertility decline is initially very subtle, but becomes increasingly severe as fitness approaches zero. The default value is "Yes", which means that fertility declines with fitness, especially as fitness approaches zero.
- **Type of selection.** MENDEL's default mode for type of selection is probability selection, wherein the probability of reproduction is proportional to an individual's fitness ranking within the population. Two forms of probability selection are provided—classic and unrestricted. In classic (textbook) probability selection, rather counter-intuitively, strict proportionality (relative to the most-fit individual) can combine with high average fitness and mild selection (low reproductive rates) to cause reductions in fitness and relatively rapid extinction. In unrestricted probability selection, with certain combinations of average fitness and offspring/female, a range of the highest fitness values are "guaranteed" survival in order to maintain population size. To give the researcher maximal flexibility, we also provide an option where strict truncation selection (i.e. artificial selection) is employed. An intermediate option, involving a form of "broken-line" selection which we have designated " Partial truncation" has also been added. With certain combinations of fitness distribution and offspring/female, this selection mode involves either truncation of the least fit individuals, guaranteed survival of the most fit individuals, or both. Selection then acts on the balance of the population. The partial truncation value, k, equals the fraction of the population which is truncated. If k=1, this is the same as full truncation selection. However, if k=0, this equals full probability selection. k=0.5 is the immediate blending of truncation and probability selection.

Advanced Population Parameters

- **Reproductive mode** Normal sexual reproduction is the default setting, but clonal reproduction can be specified. If clonal reproduction is selected, there is no recombination (overriding #4 below), and the genome is treated as one large non-recombining chromosome. There is no mating, and the same genome is transmitted from female to offspring, with each offspring then being assigned its own set of new mutations.

- **Ploidy Level.** Diploidy is the default setting, but haploidy can be specified. In this special case, selection occurs during the haploid phase of the reproductive cycle, which means all mutations are fully expressed (no recessive mutations).
- **Fraction self-fertilization.** Certain plants and lower animals can self-fertilize. The percentage of self-fertilization (as opposed to out-crossing) can be set to range from the default value 0%) up to 100%. As this value increases, there is a strong increase in inbreeding and in the rate of mutation fixation. Consequently, recessive loci have a much stronger effect on overall fitness than normal.
- **Dynamic linkage.** Linkage has a major effect on selection, and must be modeled as accurately as possible. MENDEL's default mode involves dynamic linkage. This requires specification of the haploid chromosome number and assumes two random crossovers within each chromosome, at random locations between linkage blocks - every generation. Because tracking every linkage block can become computationally expensive, the number of linkage blocks must be limited (default = 989, min=1, max=100,000). Furthermore, the number of linkage blocks should be an integer multiple of the number of chromosome (e.g. the default value of 989 is 43 times the default 23 chromosomes). MENDEL will automatically adjust to the nearest integer multiple (e.g. if you input 1000 and 23 chromosomes, MENDEL will use a value of 989). The number of linkage blocks is evenly distributed over a user-specified haploid number of chromosomes (default=23). We also offer the researcher the option (turn off "dynamic linkage") of a simpler model involving the specification of a fixed number of linkage blocks and fully randomized recombination between all linkage blocks each generation (no chromosome number is required).
- **Dynamic population growth** *(Linux/Mac OS X Only). This function is experimental and relatively untested.* By default Mendel uses a static population size. However, two options are provided to simulate dynamic population growth: (1) exponential growth model (e.g. Figure 1), and (2) carrying-capacity model. For the exponential growth model, two additional inputs need to be entered: *population growth rate* and *maximum population size*. The **population growth rate** parameter determines the percent growth rate per generation. A value of 1.0 represents static population size (no growth). To grow the population 2% per generation, enter the parameter 1.02 (note: one may need to manually convert published annual population growth rates to population growth per generation by using a formula such as 1.02^{20} = 1.48/generation – assuming a 20 year generation time). If the exponential growth model is selected, the "number of generations" input parameter (under *basic parameters*) is alternatively employed to input the maximum population size (the number of required generations being unknown). Number of generations is no longer necessary, because the simulation will automatically shutdown when the computer runs out of memory or the maximum population size is reached. The population growth parameter may vary between 1.0 – 1.5. The user will notice for low population

growth rates (e.g. 1.001) starting with small population sizes, the population will grow linearly until a sizeable enough population is formed, and then true exponential growth will occur. This initial linear phase is because population size in an integer number.

Mendel’s second population growth model is called “the carrying-capacity model”. Wikipedia.org *(accessed October 7, 2008)* gives the following definition for carrying capacity: “The supportable population of an organism, given the food, habitat, water and other necessities available within an environment is known as the environment's carrying capacity for that organism.” The equation describing relating population growth to the environment's carrying capacity can be given as:

$$dN/dt = rN(K-N)/K$$

[Reference: Halliburton, Richard. Introduction to Population Genetics, Benjamin Cummings, 2003]. where *N* is the population size, *r* is the maximum reproductive rate of an individual, and *K* is the carrying capacity. This equation is solved numerically as:

$$N_{i+1} = N_i\ (1 + r\Delta t\ [1 - N_i/K])$$

Where *i* represents the generation number, *rΔt* is a constant which represents the maximum reproductive rate of an individual times over a period of time (here about 20 years). Presently, this constant may only range between *0.0* to *1.0.*

- **Population bottlenecks.** Population bottlenecks can dramatically affect mutation accumulation and mutation fixation. MENDEL allows the modeling of population bottlenecks. The researcher can cause a bottleneck to automatically begin after a specified number of generations, resulting in a specified reduction in population size, and ending after a specified number of bottleneck generations. The reduction of population size occurs immediately at the beginning of the bottleneck, by selecting a random sub-sample of the population. When the bottleneck ends, the original offspring number/female does not change but half of the population excess (i.e. all offspring exceeding 2 per female) is used to increase population size, and half of the excess continues to be eliminated by selection. When the original population size is reached, normal selection is restored. To simulate recurrent bottlenecking, the negative sign can be placed before the first parameter (time to bottleneck).

Advanced Population Substructure Parameters

Population sub-structure. *(Linux Only)*. Perfectly random mating within a population probably never happens, especially in larger dispersed populations. MENDEL allows creation of multiple sub-populations (tribes), to account for this reality.

- **Homogeneous sub-populations** — If this option is selected, all Mendel parameters will be applied to each single sub-population equally, so that all sub-populations will start out the same. If this is de-selected, each tribe can have its own parameters defined separately.
- **Number of tribes** — The number of sub-populations can be specified.
- **Migration model** — One of three possible methods of migration can be selected: ring pass, stepping stone, and island models. The ring pass is the simplest mode of communications, and is typically used in testing parallel computing systems. In ring pass, the number of tribes are arranged as a circle, and each tribe sends the user-specified number of individuals to the neighbor to its right. In the stepping-stone model, the tribes are also arranged as a circle each tribe exchanges individuals with its neighbor. In the island model, every tribe exchanges individuals with every other tribe.
- **Migration rate** — The rate of migration can be specified. The rate of migration can be less than one (one migration event every X generations), and also can be zero.
- **Tribal competition** — Tribal competition can be specified (differential growth/shrinkage of tribes). Tribal competition works by first computing the global weighted average genetic fitness of all the tribes. Then, the tribal_fitness_factor is computed which is each tribes fitness relative to the global genetic fitness is computed. For example, you may have two tribes, one tribe may have a tribal_fitness_factor of 0.8 (representing 80% of global genetic fitness), and 1.2 (representing 120% of the global genetic fitness). This value is then subracted by 1 to get the reproductive_advantage_factor (e.g. the two tribes would become -0.2 and 0.2). The reproductive_advantage_factor is then multiplied by the user-specified tc_scaling_factor. Then the next generations population size is computed by multiplying the fertility_factor times the reproductive_rate. One caveat: with a reproductive_rate of 1, and a low tc_scaling_factor (e.g. 0.1), the the tribes will actually start shrinking over time. If this is undesireable, one may solve this problem by increasing the reproductive_rate.
    - A scaling factor specifies the strength of tribal competition.
    - Group heritability species the amount of environmental effect on differential tribal growth/shrinkage.
- **Average tribal data when plotting** — Check this box to plot global statistics instead of tribal statistics. Most data is averaged. However, the allele frequencies are summed instead of averaged.

Advanced Computational Parameters

- **Automatically allocate memory** — This means that the computer will automatically limit the amount of RAM allotted to to each run, so that multiple runs can done at the same time. If this feature is turned off, the users can define the maximal number of mutations per individual before a run stops, potentially allowing one experiment to use all available RAM. The **maximum deleterious mutations per individual** setting is how many deleterious mutations each individual can have. There is a similar setting for beneficial mutations per individual. During a simulation, if this number is exceed the program will shutdown with an error that this number has been exceeded.
- **Track all mutations** — Checking this box will set tracking threshold to zero, in which case all mutations will be tracked, including neutral mutations. This button must be checked if allele statistics are needed, or if neutral mutations are to be simulated.

  Tracking threshold — MENDEL can track every individual mutation. However, this may not be the best choice, especially with large populations and/or large numbers of generations. Hundreds of millions of mutations can accumulate within a virtual MENDEL population, causing computer operating speed to slow to a crawl, and eventually exceeding all available memory. In order to speed operation and allow larger experiments, MENDEL can track only those individual mutations that are "potentially meaningful". Most mutational effects are so close to zero, that they can be classified as "extremely near-neutral". Such effects are so extremely small that they have no significant impact, even after accumulating to very high numbers for many, many generations. Such mutations may more practically be dealt with as follows: (1) they can be assumed to act in a co-dominant manner (such that their dominant/recessive status can be ignored); (2) their effects can be pooled into their respective linkage block effects; and (3) their number can be monitored using a "mutation counter". For the sake of computational efficiency, the researcher only needs to make a practical decision in terms of where the cut-off value should be for defining "extreme near-neutrals". Above this threshold, all mutations are still individually tracked in the usual way. MENDEL's default for this threshold is 0.00001. Where high speed and maximal use of memory is desired, the tracking threshold can even be set at 1.0. This results in zero tracking of individual mutations - all mutation effects are dumped into the appropriate haplotype, so tracking is only by haplotype. The advantage of this option is that much larger runs can be done much faster, given the same computing resource. The disadvantage is that the information normally found in figures 2, 3, and 5 will be lost and all mutations are made co-dominant by default.

- **Extinction threshold.** Extinction can either be realized when fitness reaches the specified extinction threshold value -or- by population size dropping to a value of one. The default for fitness extinction threshold is 0.0.

- **Random number seed.** At several stages within the MENDEL program, a random number generator is required. When an experiment needs to be independently replicated, the "random number seed" must be changed. If this is not done, the second experiment will be an exact duplicate of the earlier run.
- **Poisson method.** There are two different implementations of the Poisson random number generator that Mendel uses. One is based on Numerical Recipes. The other is based on RANLIB. The Numerical Recipes seems to be a more robust implementation. However, it cannot handle mutation rates less than 1.e-5.
- **Changing parameters over time**. MENDEL allows a run to go for a specified number of generations, followed by data output, alteration of certain biological parameters, and resumption of the run. This can be done repeatedly, simply by choosing the commands **allow this run to be re-started** prior to a run, and then later **restart new phase of run** prior to subsequent runs. Most parameters can be altered at restart, but population size and the number of linkage blocks must remain unchanged. (Caution: allowing restarts of large runs will save large amounts of data, which can rapidly fill available disk storage.).
- **Restart second (third, fourth) phase of run with these new parameters**. Check this button in order to use the data from a previously run case (a case that has previously been run with the "Allow this run to be later re-started..." button checked).
- **Queuing system**. For single-user systems, this may be set to "No queue", which will only allow one Mendel job to run at a time. If any new job is submitted, qsub.pl will automatically kill any other Mendel jobs running. For multi-user systems, if PBS/Torque is installed, PBS is the best choice for managing multiple jobs. On some cluster machines, a "himem" node may be designated for submitting jobs that require an unusually high amount of memory (such as 16GB or 32GB). For example, a user may want to run very large populations, or dynamically growing populations. Setting up a high-memory node would require custom configuration of the back-end coding such as modifying qsub.pl and memory.pl.
- **Simulation engine** Currently, the Fortran version is used by default and is faster, and consumes less memory. However, due to the popularity of C-based languages, we had developed a C version of the Mendel engine. However, the development on the C-version has not continued since version 1.5 and is therefore no longer distributed with the latest Mendel releases.
- **Compute allele frequencies every ... generations.** Input the time interval (number of generations) that MENDEL will perform a polymorphism analysis of allele frequencies. Polymorphisms analysis requires cycling through all the mutations, so it is computationally expensive. Reducing this number will cause MENDEL to update the Allele Frequencies plot more often, but will also cause MENDEL to run for a longer amount of time. The default is computation of allele frequencies every 100 generations.
- **Output verbosity level.** MENDEL generates a lot of output information. However, not all of it is necessary. This `verbosity` option allows the user to limit

the amount of files that are written to disk in order to save hard disk space. A verbosity level of 0 will essentially turn most diagnostics routines off, and will output just a `.out` output file and a `.hst` history file (which will allow viewing of fitness and mutation plots). A verbosity level of 1 will write all necessary files for plotting using the default JavaScript plotting system (Flot). A verbosity level of 2 "Output everything" is required to write ancillary files, such as `.gnu` Gnuplot files, `.tim` timing information for performance benchmarking, `.pmd` polymorphism frequency table, `.acc` table of accumulated deleterious dominant mutations, etc.

Special Applications -

1. **Initial Heterozygous Alleles** – This means that the population starts with pre-existing diversity. This feature is still under development. The user must specify the number of initial contrasting alleles, and the total fitness increase - if all favored alleles go to fixation.
2. **Include Neutral mutations in the analysis** — This means that a specified fraction of all new mutations will arise within the "junk DNA" portions of the genome, and so will be perfectly neutral. If 50% of the genome is junk, then 50% of all mutations will be neutral. If the total mutation rate is 100 per generation, then the rate of neutral mutations will be 50 per generation. The remaining non-neutral mutations will have the specified beneficial-to-deleterious mutation rate. Neutral mutations will then be tracked, tallied, and plotted, just as with beneficial and deleterious mutations.
3. **Waiting Time Experiments** — MENDEL can determine the waiting time required to establish specific beneficial nucleotides or nucleotide strings. The user must specify the initialization sequence (such as AAAA), and the target sequence (such as GTCT). The user must specify the degree of benefit (a fitness benefit of 1% is designated as .01).

   Detailed instruction for waiting time experiments:

   A special version (2.5.0) of Mendel's Accountant (Mendel) allows us to determine the waiting time for establishing a new string of nucleotides within a given biological population, at a particular location within the organism's genome. At present Mendel's primary limitation in this regard is it has a lower limit for the designated mutation rate per string (mutation rate per sting must be 10^{-5} or higher). This limitation has to do with limits of current random number generators. When lower mutations are of interest, runs can be done with higher mutation rates, and then fixation times corresponding to a lower mutation rate can be approximated using proportional correction upward of the waiting times (or by extrapolation).

   When doing waiting time experiments, Mendel starts with a specified string of nucleotides. That string undergoes mutation, and the resulting mutant strings segregate and drift within the population from generation to generation. Whenever the target string arises, the individuals carrying it are given a specified reproductive advantage. Any target string that arises is protected from any further mutation. Mendel allows the mutation/selection process to continue until the target string reaches fixation.

After the target string reaches fixation, Mendel summarizes and plots the results of the experiment, and provides the following data points: a) number of generations to first instance; b) generations to final fixation; and c) total number of instances arising during the experiment.

Details on how to do a waiting time experiment –

A. Go to the Mendel control panel and select "start".
   1. Go to the yellow box and enter a case ID (exactly six characters), or click on checkbox to generate a random case ID.
   2. If desired, enter a note about the specific case.

B. Fill in "basic Parameters".
   1. Mutation rate – this is usually the mutation rate per genome, but for waiting time experiments it is the mutation rate per string (we ignore the rest of the genome). We usually use the species' known mutation rate per nucleotide per generation, multiplied by the length of the string. If a diploid population, multiply by two. The human mutation rate for a string of 5 nucleotides would be 10^{-8} times 5 times 2 (10^{-7}). Because Mendel cannot handle such a low mutation rate, we use a rate 100-10,000 fold higher, and then adjust waiting times upward by that factor. This is strictly a practical consideration.
   2. Beneficial/deleterious ratio – the default for this is .0001, but for waiting times experiments there are no deleterious mutations, and we consider all mutations neutral unless they complete the target string, so we enter 0.0 (will be auto-selected when *waiting time experiment* is selected).
   3. Reproductive rate – we recommend keeping default setting of 2, which means there will be two offspring for each surviving individual – allowing selective removal of 50% of all progeny every generation (this is very intense natural selection).
   4. Population size – Choose desired population size. We have been able to go up to a population size of 1 million – but your computer may not have enough memory for such large population sizes.
   5. Generations – Generally we see that waiting times are very long, and if you do not specify enough generations, the experiment will terminate before the desired target sequence has gone to fixation. Generally at least 50,000-150,000 generations should be specified. If you specify too many generations you may encounter memory allocation issues.

C. Click on check box for *advanced settings*, then -
   1. Go to last tab (Special Applications).
   2. Click on checkbox for "Waiting Time Experiments".
   3. Fill in initialization sequence (i.e., AAAAAA).
   4. Fill in target sequence (i.e., TCGTCG) - must be same length as above.
   5. Indicate the fitness benefit of the completed string (i.e., .001 is the same as a 0.1% increase in fitness).

D. If you click on “Waiting Time Experiments” several variables are automatically set:
    1. Immediately above, “Include neutrals in the analysis” is activated, and the “fraction genome which is non-functional junk” is auto-set to 1.0. This means all point mutations are by themselves neutral – only the completed string is going to be beneficial.
    2. Under the Population tab, the first setting “Recombination model” is auto-set to “suppressed recombination between homologous chromosomes”. This makes it so that within the string there is no recombination.
    3. Under the Mutation tab, the first setting (distribution type) is auto-set to “all mutations are equal”. This makes it so that each time the target string arises, it is assigned exactly the same fitness benefit (as designated under Special Applications tab).

E. Other optional settings.
    1. Under Mutations tab, you can specify the degree of “dominance” for the fitness benefit of the completed target string. We routinely use full dominance (set fraction recessive to zero, set expression of recessives to zero, set expression of dominants to 1.0).
    2. Under Selection tab, heritability can be modified and/or artificial truncation selection (or partial truncation selection) can be activated.
    3. Under Computation tab the random number seed can be changed, allowing replication of experiments. Memory allocation can also be modified.
    4. Mendel has many special features, many of which cannot operate in unison. We have not tested the compatibility of the Waiting Time function with the various other special functions such as bottlenecks or dynamic population size.

F. When all the inputs have been chosen, click the green button labeled “submit”(next to yellow button on top), review the settings as needed, and go back to the green button and click “execute”. The run will take a few moments before it starts to produce output.

G. Output and results.
    1. By clicking “plot” within the control panel, one can see: a) mutation count history, b) fitness history, c) current allele count, and d) allele frequencies.
    2. By clicking “output” within the control panel, one can see regular updates of the unfolding experiment. Output includes mutation counts, current fitness, sampling of the currently circulating strings in the population, number of target instances that have arisen, and the current frequency of the target string in the population.
    3. Final output (written in red) shows generations to first instance, the ID of the effective instance, generations to final fixation, and total number of instances during the experiment.
    4. If the desired mutation rate was too low for Mendel (or was too low for a timely experiment), and so an enhanced mutation rate was used, the user must increase the number of generations observed (by the same proportion that mutation rate was enhanced).

5. The actual waiting time in years must be calculated by multiplying the number of generations required times the amount of time per generation (i.e., the human generation time is 20-30 years).
6. The user is given the ID of the effective instance (the instance that eventually went to fixation), and this allows the user to examine the table of all instances, find the instance that was fixed, and then find the generation in which it arose. This allows calculation of the duration of the amplification phase of the experiment (the generations from effective instance to fixation). This amplification phase is independent of mutation rate and should not be adjusted for any possible enhancement of mutation rate.

MENDEL Output

MENDEL routinely outputs a primary summary data file and a series of figures. Raw data output files are also saved.

Output summary data file: This primary summary data file involves an on-going real-time reporting of deleterious and beneficial mutation counts, mean fitness, fitness standard deviation, as well as other data. These summary statistics are reported for generations 1, 2, 3, 10, 20, and then every 20 generations.

Raw data files: These can be accessed separately through a marked box within each output figure.

Figure 1: This figure plots the number of both deleterious and beneficial mutations per individual over time. If neutrals are enabled, these also will be plotted. The scale for deleterious mutations is on the left and the scale for the beneficial mutations is on the right. If neutrals are simulated, they share the scale on the left.

Figure 2: This figure shows average individual fitness (left scale) and the population size (right scale), plotted over time. The initial fitness is always assumed to be 1.0. For both Figures 1 and 2, the y-axis is self-scaling, since mutation counts and fitness can change dramatically over time. The fitness standard deviation is shown in gray.

Figure 3: This figure is designed to reveal how selection is altering the distribution of mutation effects in the population. Typically it shows that the frequencies in the population of mutations with larger effects are affected by the selection process much more strongly than the frequencies of mutations with small effects. Figure 3a is histogram plot that shows the mutation effect distribution for deleterious mutations. The x-axis uses a log scale to represent the magnitude of the mutation effects and ranges from lethal (-1.0) on the left to the minimal mutation effect tracked on the right. The y-axis uses a linear scale, and the height of the bar reflects the fraction of each class of mutations that was not eliminated by selection. A value of 1.0 is the expected height for all bars when there is no selection. Perfect selection for a given mutation effect interval will result in bars of zero height. Figure 3b shows the beneficial half of

this same distribution. The distribution extends from the left with the smallest mutational effects tracked (tracking threshold) to the maximal beneficial effect (on the right), and any bar significantly higher than 1.0 reflects positive selection. Note: These figures are not generated until there are a “sufficient” number of mutations. Even so, plots can initially show significant noise associated with sampling error until larger data sets have built up. This is especially true for beneficial mutations, because they are usually relatively rare. The user can know that the number of accumulated mutations has become large enough to give reliable plots when there is little random fluctuation in bar heights and the transition from selectable to near-neutral becomes smooth and unambiguous.

Figure 4: This figure plots the selection threshold for dominant alleles as a function of the generation count, beginning with generation 200. The selection threshold, described more completely in the glossary below, is the value of absolute fitness effect for which the deleterious allele frequency is 50% of what it would be, if there had been no selection. For values of absolute fitness effect smaller than this threshold, alleles frequencies are governed more by drift and less by selection, while for values larger than the threshold, the opposite is true. Data for recessive alleles is not plotted, but is available in the data file. At any given generation, one can readily estimate the selection threshold visually using Figure 3, by observing the fitness value on the horizontal scale corresponding to a bar height of 0.5.

Figure 5: This figure, like figure 3, is also designed to reveal how selection is altering the distribution of mutation effects in the population. By using a linear (but truncated) scale for the x-axis, it shows more closely just the lower impact mutations. It displays, in greater detail the differences between the theoretical (red) distribution of mutation effects (as would accumulate apart from selection) and the actual distribution of accumulating mutations (blue = recessive, green = dominant). Like figure 3, this figure reveals the variable effectiveness of selection over a range of mutation effects. Unlike Figure 3, the viewer can readily see that small-effect mutations are always much more numerous than large-effect ones, both before (red), and after (blue/green) selection. Figure 5a displays deleterious mutations, and figure 5b displays beneficial mutations. Like figure 3, the plotting of accumulated mutations is only reliable when enough mutations are present, evident when the distribution visually transitions from jagged to smooth.

Figure 6: This figure displays the frequency distribution of the fitnesses of the accumulating haplotypes versus their composite fitness effect. Those haplotypes with a net deleterious effect are plotted in red (left of zero) and those with a net beneficial effect in green (right of zero). This figure shows the distribution of net linkage blocks fitness effects (as opposed to individual mutation effects). Linkage blocks with zero mutations are ignored.

Figure 7: This figure superimposes the distribution of phenotypic fitness values of the individuals in the population before (red), and after (green), selection. A histogram format is used. The blue line plots the ratio of the number of surviving (selected) individuals within a given phenotypic class versus the number of offspring in that fitness category prior to selection (scale shown on the right). The portion of the red distribution not covered by the green represents those offspring that are selected away (those offspring that will not reproduce to create the next generation).

Figure 8: This figure is only fully accurate when all mutations are tracked. It displays the frequencies of all tracked individual mutant alleles in the population. It is plotted at user-specified intervals (by default every 100 generations), and finally at the end of the run. The figure provides the number of very rare alleles, in the population (having frequencies of less than 1%), the number of polymorphic alleles (having frequencies between 1- 99%), and the number of fixed alleles (having frequencies greater than 99%). The data button on this plot provides access to the data for the current and all the previous plots. The file with suffix .pmd provides even more detailed information on the distribution of polymorphisms at each of these plot times. Results are in table format with 500 values of polymorphism frequency (50 intervals) vs. fitness effect (10 intervals). Caution – this data represents only tracked mutations, so the tracking threshold must be set to be less than the lowest-impact mutation effect in order to account for all the data.

Applications of MENDEL

Teaching: The MENDEL program is a useful teaching tool to demonstrate to students in a concrete and visual manner the fate of mutations once they enter a population, and how they increase in frequency, are eliminated, or simply drift randomly. MENDEL shows how these dynamics play out over many generations under a wide range of conditions. The student can see how this process affects average mutation count per individual, average fitness, allelic frequencies, and mutational fixations over time. The student can experiment with the biological parameters that alter the rates of these processes. MENDEL also allows the student to see exactly what happens during a population bottleneck, what happens in a mutational meltdown scenario, how genes can circulate between sub-populations, and what happens when key biological parameters are modified during a run.

The newest Mendel capability allows students to study the “Waiting Time Problem”. A specific portion of the genome, consisting of a specific string of 1-100 nucleotides, can be caused to undergo a series of neutral mutations until a specific target string arises, which is beneficial. Whenever the target string arises, is confers a reproductive advantage, such that eventually the target string will be fixed in the population. Depending on string length and other key variables, waiting times quickly become extremely problematic.

Research: Mendel can function as a sophisticated research tool. To our knowledge, there is no simulation program comparable that provides genetic researchers with such a realistic and flexible research simulation capability. Highly specific scenarios can be run that have bearing on extinction of species, management of endangered species, germplasm preservation, epidemiology, ecology, etc. Likewise, simulations can also be run which have bearing on more basic questions, including the relative importance of the different variables that affect selection efficiency, and the nature of the waiting time problem.

MENDEL Glossary

Mendel's Accountant (MENDEL). An advanced numerical simulation program that acts as a genetic accounting system, and realistically models how genomes change over time in response to mutation and selection.

Genome. The entire genetic content of an organism. In MENDEL, the initial genome is not specified except in terms of genome size, chromosome number, and number of linkage blocks. A functional genome is simply assumed as a backdrop for the accumulating mutations, which are individually being tracked.

Functional genome. The entire physical genome, minus any portions that have no biological expression or consequences relating to biological fitness.

Population. All the individuals that constitute an inter-breeding group. In MENDEL, the population is specified in terms of number of reproducing individuals, mating pattern, fertility level, and sub-structure (tribes).

Mutation. Any heritable change in the genome that was not present in the previous generation. In MENDEL, mutations can have a range of effect from lethal to beneficial, and a range of expression from entirely dominant to entirely recessive. MENDEL tracks the effect of every mutation from the time that the mutation enters the population until it may be lost due to selection or random drift.

Mutant locus - The location of a mutation, in terms of its position within a linkage block within a chromosome. In MENDEL, all loci are assumed to be non-mutant except where a mutation has been added. Once a single mutation arises within an individual at a specific location, the same corresponding location in all the other individuals not carrying this mutation, by definition, becomes the non-mutant allele.

Mutant allele. All the derived copies of an initial mutation, which are being passed from generation to generation. A mutant allele can increase or decrease in its frequency within the population.

Mutant allele frequency - The mutant allele frequency for a given locus is determined by number of copies of a given mutation in the population, compared to how many copies there would be if every individual was homozygous for that mutation. (For diploids the total number of possible mutant copies is two times the population size.). If there are 2 copies of a mutation in a diploid population of 100, the mutant allele frequency is 1%, so the non-mutant allele frequency is 99%.

Mutation fitness effect. We refer to the biological impact of a mutation on individual fitness as the mutation's "fitness effect". In MENDEL a given mutation fitness effect can be small or large. The mutation effect is expressed as the relative change in an individual's total *biological functionality*, as reflected by a corresponding change in an individual's *genotype value* (see below). A deleterious mutation with an effect of -0.01 decreases an individual's genotype value by 1%. Crudely speaking, one percent of the genomic information is lost, or more accurately, total biological functionality is reduced by 1%. Another way of saying this is that such a mutation decreases genotypic fitness by 1%. These are all just alternative ways of describing the biological effect of a mutation. Mutation effect is independent of environmental variation (phenotypic noise), and random aspects of reproduction (reproductive noise). Mutation effect is similar but not identical to the traditional concept of a *selection coefficient*. See Sanford et al. (SCPE 8(2), 147-165) for the mathematical function we use to generate the distribution of mutation effects, and for an explanation of the difference between fitness effect and selection coefficient.

Genotype. The genotype is the specific collection of genetic alleles present in a specific individual within the population. In MENDEL, the starting genotype for all individuals is an unspecified and invariant genome for the organism. MENDEL specifies only the mutational deviations from this non-mutant starting genotype. In MENDEL, the specified genotype is simply the sum total of all the mutant alleles (including their chromosomal locations), within an individual.

Genotype value. The genotype value is that portion of an individual's total biological functionality that is derived exclusively from that individual's genetic makeup. The genotype value is different from the phenotype value because environmental factors (phenotypic noise) also contribute to an individual's biological functionality. In MENDEL, the initial genotypic value of all individuals is defined as 1.0. Beneficial mutations increase this value, and deleterious mutations decrease this value. The extent to which a mutation alters genotypic value is a function of its specific mutation effect (see above). Genotype value can be understood as being synonymous with the term *genotype fitness*. However, the concept of genotype value (genetic fitness) is distinct from what most population geneticists formally define as "fitness". For clarity, we will use the term *reproductive fitness* (see below), to refer to the traditional population geneticist's definition of fitness, as distinct from *genetic fitness*. Genetic fitness is what is actually plotted in MENDEL's figure 1b.

Linkage block (haplotype). Mutations are not inherited independently but are passed from generation to generation in clusters or blocks. These clusters are physically linked together, within "linkage blocks", which represent specific regions of linear chromosomes. Although chromosomes recombine something like cutting a deck of cards, some cards consistently stick together, due to recombinational "cold" spots. Points of frequent recombination (hot spots) separate linkage blocks (cold spots) from each other. A specific set of mutations which is physically being inherited as a single unit is called a *haplotype*.

Phenotype. The actual biological functionality of an individual. The phenotype is affected both by the genotype and by the environment in which the individual develops. The genotype and phenotype are correlated, but they are not identical. In MENDEL, the phenotype value (or fitness) is created by adding to each genotype value (or fitness) a random "environmental noise value" based upon the specified "heritability" and using a random number generator. This environmental noise value is not heritable and so is not passed on to the next generation.

Phenotype value. This is the actual biological functionality of an individual, arising due to the combination of genotypic effects and environmental effects. Phenotypic value is what selection actually acts on - it is what "Mother Nature" actually "sees". In MENDEL, the initial *mean* phenotypic value is always 1.0. In the initial first generation, all individuals will have an identical genotype, but there will still be variance around the population's mean phenotypic value - due to environmental effects. The amount of environmental noise is controlled by specifying a "heritability value", which is the input parameter that defines the ratio of genotypic variance to environmental variance. Because the phenotypic contribution from heritability scales with genotype value and therefore becomes small when the genotypic value becomes small, an additional non-scaling noise factor can also be used to modify phenotypic values. Phenotype value is synonymous with the terms *phenotypic fitness* or *biological fitness* as reflected by the common use of these terms among biologists. However, the concept of phenotype value (phenotypic fitness) is distinct from what population geneticists formally define as "fitness". For clarity, we will use the term *reproductive fitness* (see below), to refer to the traditional population geneticist's definition of fitness, which is distinct from phenotypic fitness.

Reproductive fitness. We define this term as the phenotypic value (phenotypic fitness), plus "reproductive noise". Reproductive noise arises because actual success in reproduction is not just determined by biological functionality, but also by random reproductive factors. So phenotypic fitness and reproductive fitness are correlated, but not identical. The strength of correlation between phenotype value and reproductive fitness depends upon the selection scheme employed. Artificial truncation will yield the highest possible correlation, while classical probability selection will yield the lowest correlation of the various schemes implemented in MENDEL. What we are calling "reproductive fitness" is sometimes called "Darwinian fitness" or "Wrightian fitness" after Sewell Wright, the first to formulate "Darwinian fitness" mathematically. We recognize that "Darwinian fitness" encompasses elements beyond "reproductive fitness", but have not identified a more precise term that is still understood intuitively by a broad audience.

Selection threshold. The selection process eliminates deleterious mutations with large negative fitness effect values more effectively that it does for deleterious mutations with smaller ones. Similarly, selection enhances the frequencies of favorable mutations with large positive fitness effect values more effectively than it does for favorable mutations with smaller values. For fitness effect values sufficiently small, selection plays essentially no role in altering the frequencies of such alleles. Mutations in this range, both deleterious and favorable, are generally referred to as "effectively neutral". The fate of these mutations is governed essentially entirely by drift. By contrast, deleterious mutations that have large impacts on genetic fitness are eliminated very effectively by selection such that their frequencies in the population are maintained at nearly zero. Typically, there is a very broad transitional zone (representing several orders of magnitude of fitness effect), between the zone of highly effective selection and the zone of essentially no selection. The mutations in this transition zone have been termed "nearly neutral". We define the term 'selection threshold' as the absolute value of fitness effect at the midpoint of this transition region - wherein the allele frequency is precisely 50% of what it would be if there had been no selection. This means that above this threshold value in absolute fitness effect there is more than 50% elimination of deleterious mutations as a result of the selection process, while below this threshold value there is less than 50% elimination. Typically, mutations more than an order of magnitude smaller in absolute fitness effect below the threshold are not significantly influenced by selection, while deleterious alleles more than an order of magnitude larger than the threshold are entirely eliminated by selection. For the case of favorable alleles, the same absolute threshold value tends to apply.

Mutation/selection chain. In the real biological world, this is the chain of events that links a mutational event to a selection event. A single mutation affects a linkage block, which affects a chromosome, which affects a genotype, which affects a phenotype, which affects the reproductive fitness of an individual, which affects the actual transmission of a mutation into the next generation. There is biological noise at each link in this chain, and so each link of this chain is associated with an imperfect correlation. MENDEL is designed to accurately reflect this chain of events, so that a given mutant value actually has a limited effect on a linkage value, which has a limited effect on a chromosome value, which has a limited effect on a genotype value, which has a limited effect on a phenotype value, which has a limited effect on the reproductive fitness value of a given individual - which defines the actual transmission of the mutation. The strength of the correlation at each stage depends on the values chosen by the user for the various input parameters.