

# CDFISH

## USER MANUAL

2012

Version: 0.55

Last Updated: 2013.10.08

Authors:

**Erin L. Landguth<sup>1</sup>, Clint C. Muhlfeld<sup>2</sup>, and Gordon  
Luikart<sup>3</sup>**

1 - University of Montana, Division of Biological Sciences, Missoula,  
MT, 59812, USA.

2 - U.S. Geological Survey, Northern Rocky Mountain Science Center,  
Glacier National Park, West Glacier, MT, 59936, USA.

3 - Univeristy of Montana, Flathead Lake Biological Station, Division  
of Biological Sciences, University of Montana, Polson, MT 59860, USA.

**Table of Contents**

1	Introduction.....	3
1.1	Changes from CDPOP.....	3
1.2	What can CDFISH do.....	3
1.3	How does CDFISH work.....	3
2	Getting started.....	5
2.1	Dependencies.....	5
2.1.1	Baseline requirements.....	5
2.1.2	Python on non-windows platforms.....	5
2.1.3	Python on windows.....	6
2.1.4	Obtaining NumPy.....	6
2.2	Installation.....	6
2.2.1	Installing Python and NumPy.....	6
2.2.2	Installing CDFISH.....	6
2.2.3	Description of CDFISH files.....	7
2.3	Example run.....	7
2.3.1	Command line run.....	7
2.3.2	GUI run.....	8
3	Input.....	9
3.1	Input Files.....	9
3.2	Model Parameters.....	10
3.3	Mating Parameters.....	10
3.4	Dispersal Parameters.....	11
3.5	Offspring Parameters.....	11
3.6	Genetic Parameters.....	12
4	Output.....	12
5	General issues.....	12
5.1	How to obtain CDFISH.....	12
5.2	Debugging and troubleshooting.....	12
5.3	How to cite CDFISH.....	12
5.4	Disclaimer.....	13
6	References.....	13
7	Acknowledgements.....	15

## 1 Introduction

The goal of this user manual is to explain the technical aspects of the current release of the CDFISH program (v0.5). The program is written in Python 2.7 and provided with installation instructions for most platforms, along with sample input files. CDFISH v0.5 is built on a driver-module, plug-in, docking architecture that allows for ease of future modular development. CDFISH has been debugged as carefully as possible by testing all combinations of simulation options. Information for users, including this user manual, FAQ, publications, ongoing research, developer involvement, and downloads can be found at <http://cel.dbs.umd.edu/software/CDFISH/>.

### 1.1 Changes from CDPOP v1.0

There are major innovations in CDFISH which were not included in CDPOP. We list below the new functionalities of CDFISH:

- Subpopulations of individuals. Individual (x,y) locations can now be grouped into subpopulations on the landscape that could represent home spawning grounds, for example.
- Population dynamics. Subpopulation numbers can now fluctuate as a function of vital dynamics, straying, and/or the inability of an individual to make it back to its original subpopulation.
- Offspring have 4 states: residency, migration, straying, or death. With residency, offspring do not disperse back to the source location (lake or ocean, for example), but reside in their original subpopulation. Migration is a probability of returning back to a subpopulation that is a function of the stream resistance surface (in units of cost). If an offspring does not migrate back to its original subpopulation than it can either stray to another subpopulation at some probability or die.

### 1.2 What can CDFISH do

CDFISH's realistic representation of the spatial environment and population genetic processes provide a powerful framework to investigate the impact of ecological factors on the genetic structure of aquatic species. Example simulations might include:

- How will changing stream resistant surfaces affect the genetic structure from an individual-based setting?
- What combinations of probabilities of straying, migration, residency, and vital dynamics prevent a subpopulation from becoming extinct?
- How will the incorporation of a partial barrier on the stream resistant landscape affect the genetic structure and population dynamics of individuals or populations?

### 1.3 How does CDFISH work

*Individual-based simulations in riverscapes.* CDFISH models genetic exchange for a given stream resistance surface and  $n - (x, y)$  located individuals as functions of individual-based movement through mating and dispersal, vital dynamics, and mutation. Prior to running CDFISH, users must create a stream resistance surface where each cell value (pixel) represents the unit cost of crossing each location. Pixels are given weights or 'resistance values' reflecting the presumed influence of each variable to movement or connectivity of the species in question (e.g., Dunning et al. 1992; Cushman et al. 2006; Spear et al. 2010). Stream resistance surfaces can be parameterized to reflect different costs to movement associated with water temperature and flow, habitat complexity, barriers, elevation, slope, or other stream features. From here, CDFISH requires as input a matrix of movement costs that can be computed based on shortest cost paths algorithms between all pairs of individuals (or populations). Common approaches to calculating cost distance matrices in landscape genetics include Dijkstra's algorithm (Dijkstra 1959; e.g., implemented through programs such as UNICOR (Landguth et al. submitted)), CIRCUITSCAPE (McRae and Beier 2007), PATHMATRIX (Ray 2005), and COSTDISTANCE in ArcGIS (ESRI Corp., Redlands, CA). By comparing genetic distances between individuals with ecological cost distances between them, researchers can test specific hypotheses about the influences of stream features and environmental conditions on gene flow (Cushman et al. 2006, Epps et al. 2007). The program can flexibly incorporate absolute or partial barriers, and panmixia within this cost distance modeling framework. This is critical to provide a consistent and comparable framework to evaluate the pattern-process relationships resulting from isolation by resistance, isolation by Euclidean distance, and isolation by barriers (Cushman and Landguth 2010).

*Demographics within subpopulations.* Individuals are assumed to occupy subpopulations on the landscape that is user defined by the  $(x,y)$  locations. The simulation program assumes a carrying capacity within each subpopulation and allows for varying population numbers through vital dynamics (birth and death rates), offspring straying to another subpopulation, and/or the inability of an offspring to make it back to its original subpopulation due to a high stream resistance. The genotype of each locus for each individual is initialized randomly (i.e., maximum allelic diversity). The initial age structure of the population and the sex of each initial individual is user defined.

Reproduction is heterosexual with a random mating structure. Mated pairs can be chosen based on male or female with or without replacement. Each mated pair can have a number of offspring that is a bounded random draw based on a uniform probability distribution, a Poisson draw with specified mean, or a constant number. Mendelian inheritance with  $k$ -allele mutation (rate chosen by the user) is used to generate the offspring's genotype and the sex assignment is random.

*Dispersal through residency, straying, and migration.* Offspring can either reside in each subpopulation (e.g., stream or tributary) or

disperse to a source location (e.g., lake, river, or ocean). Offspring that disperse to the source must migrate back to their respective subpopulation or stray to another subpopulation. Straying and residency probabilities are user defined for each subpopulation. Migration to subpopulations is a linear function of the stream resistant riverscape. Individual offspring move a distance from their source location based on a random draw from a probability distribution inversely proportional to a linear function; probabilities range from one at no distance from the original location to zero at the maximum dispersal distance. To realistically reflect species-specific dispersal abilities, the user specifies the maximum dispersal distance (in cost units) an individual can travel, expressed as the maximum path length for a species given its dispersal ability.

*Simulating natural selection.* Past versions of CDFISH modeled three sources of genetic variation: gene flow, genetic drift, and mutation. These versions assumed that different genotypes have an equal probability of surviving and passing on their alleles to future generations and thus, natural selection was not operating. CDFISH V 0.55 and greater implements natural selection analogously to the adaptive or fitness landscape of allele frequencies (Wright 1932). This new functionality enables extension of landscape genetic analyses to explicitly investigate the links between gene flow and selection in complex landscapes at an individual's level. The user specifies fitness landscape surfaces for each genotype of a single diallelic locus that is under selection. For example, three relative fitness surfaces must be specified for the three genotypes, AA, Aa, and aa, from the two alleles, A and a. Selection is implemented through differential survival of dispersing individuals as a function of the relative fitness at the location on that surface where the dispersing individual settles. CDFISH reads and extracts genotype and location specific fitness values for each  $n - (x, y)$  individual in the pre-processing step. The program will continue all other processes the same as above, with an additional step implementing selection during the dispersal process.

*CDFISH Input.* A user must specify the input parameters through an input script file or a graphical user interface (GUI). As the model simulates stochastic processes, most applications will quantify mean and variability of genetic structure across many runs. Thus, a Monte Carlo option is provided for the user to choose the number of runs to simulate given a single set of input parameters. In addition, a user may also frequently wish to launch several runs with different parameter values simultaneously (i.e., sensitivity analysis). This functionality is provided through batch capability. See Section 3 for program input details.

*CDFISH Output.* There are two model outputs. The first is a series of comma delimited files specifying the locations, genotypes, age, and sex of each individual at specified simulation years. The second output is a file containing yearly values for various population parameters, formatted to facilitate graphical display. See Section 4 for program

output details.

## 2 Getting started

### 2.1 Dependencies

#### 2.1.1 Baseline Requirements

CDFISH requires the Python2.7.x interpreter and NumPy package. Remember that Python modules usually require particular Python interpreters, so be sure the version ID for any external Python module or package (e.g. NumPy or others) matches the version of your Python interpreter (normally v2.7.x).

#### 2.1.2 Python on Non-Windows Platforms

Some common computer platforms come with Python installed. These include MAC OS X and most Linux distributions. To determine which Python a MAC or Linux workstation has installed, start a terminal console and enter "python." You'll see the version number on the top line (enter Control-D to exit). Replacing an older Python interpreter (pre v2.4) with a newer one (v.2.7.x) on a Linux or MAC OS X machine can be tricky, so ask a System Administrator for help if you're not sure which packages depend on the current Python installed.

#### 2.1.3 Python on Windows

Windows (7, XP, 2000, Server) does not come with Python installed, so follow the instructions below to obtain and install Python on a computer running the Windows operating system. Get a windows installation of the base Python installation (current v.2.7.x) at: <http://www.python.org/download/releases/>.

#### 2.1.4 Obtaining NumPy

A good suggestion for all-in-one installations is through the superpack Windows installer available from the SourceForge website: <http://sourceforge.net/project/>. Note that more complete information for NumPy is available at [www.scipy.org](http://www.scipy.org), where the SciPy module is also presented. Another source is

<http://www.enthought.com/products/epd.php> for a free academic and educational usage in a single downloadable installer that has everything and then some (Numpy, Scipy, Matplotlib, and 70+ modules for python). We highly recommend this version for easy installs.

### 2.2 Installation

#### 2.2.1 Install Python and NumPy

Make sure that Python and NumPy are installed, and available to you. You can test this by typing "python" at a command window. If python is available you'll get the python prompt ">>>". If it is not a recognized command, it means either that python is installed but is not in your command shell's paths, or that python is not installed. In the first case ask an administrator to add it to your command paths. If your shell locates and loads python, type, "import numpy". If python does not complain that there are no such modules, all is

well.

The following instructions assume Python and NumPy are not yet available on your computer; if they are, skip to section 2.2.2.

\* First run the Python executable installer you've chosen, accepting defaults for the installation directory. On Windows this will typically place the executables and libraries in `c:/Python2.7/bin` and the "site-packages" package tree for user installed Python modules in `c:/Python2.7/lib/site-packages`. If you are installing it on a network on which you do not have administrative privileges, you may need to ask a system administrator to install python and the NumPy packages in their default locations.

\* Next install NumPy using the supplied executable installer. This will install NumPy in your Python `./site-packages` directory. If you downloaded enthought's version, you will not need to install a separate NumPy installer.

### 2.2.2 Installing CDFISH (for Linux or Windows)

Next, install the CDPOP software itself by unpacking the zip archive supplied. Navigate to the directory on your PC where you wish to install CDFISH, and unpack the supplied zip archive file using a free archive tool like 7Zip (7z.exe), Pkunzip, Unzip, or an equivalent. Seven-Zip (7Z.exe) is highly recommended since it can handle all common formats on Windows, MAC OS X and Linux. On Windows, it is best to setup a project specific modeling subdirectory to perform your simulations outside of any folder that has spaces in its name (like "My Documents"). At this point you should be able to execute the supplied test inputs.

### 2.2.3 Description of CDFISH files

3 directories will be installed in your directory. Here is a description of each:

1. src - CDFISH source code
2. doc -
  - README.txt - a quick how to run CDFISH instructions
  - CDFISH\_user\_manual.pdf - this file
  - CDFISH\_history.txt - Notes on history and version changes.
3. data - Example input files
  - cdfish\_test.cd - example cost distance matrix file (Here, 5x5 Source and each population cost with cost between populations)
  - cdfish\_test.xy - example n-(x,y) file for individuals
  - cdfish\_test\_NAs.csv - example n-(x,y) file with sparse individuals
  - agedistribution.csv - example age distribution file
  - allelefrequency.csv - example allele frequency distribution file

- `cdfish_test.csv` - run parameters corresponding to the example files

## 2.3 Example run

### 2.3.1 Command line run

The example run is for 64-points (4 subpopulations with 16 individuals in each) representing individuals with an example cost distance matrix. To run the following example, follow these steps:

1. Double check that the 3 directories provided in the archive are in the same directory.
2. The included file `cdfish_test.csv` specifies the parameters that can be changed and used in a sample CDFISH run. Open `cdfish_test.csv` in your editor of choice. A spreadsheet program like Microsoft Excel, allows for easy editing of the tabular values.
3. There will be 3 lines of information in `cdfish_test.csv`: a header line and 2 lines of information corresponding to 2 separate CDFISH runs (batch process). See the `user_manual.pdf` that contains a breakdown for each column header and the parameters that can be changed. The 'Input' listed is for the first row in the file. Make sure you save this file in the same format - a comma delimited file - when you make changes to the parameters. Select 'Yes' or 'OK' for any Excel questions about saving in this format.
4. **Start the program with a command line:** For example, if you use python from the command line, then open a terminal window and change your shell directory to the CDFISH src home directory (i.e., `> cd C:\"homedirectorylocation"\src`).
5. **Run the program:** There are a number of ways to run this program. If you are using a command shell you can run the program by typing `"python CDFISH.py C:/"homedirectorylocation"/data cdfish_test.csv output_test"`. Note that there are 5 arguments here that must be included with spaces in between:
  - `"python"` starts python, for example from the command line. Note that other python environments may have different calls here. In PyLab (the IDE distributed with EPD), the call is `"run"`.
  - `"CDFISH.py"` runs CDFISH program.
  - `"C:/"homedirectorylocation"/data"` is the directory location of the input test files. You can point this directory to other project files, for example. We suggest not having any spaces in your directory names.
  - `"cdfish_test.csv"` is the parameter file. Note you can always rename this file, just make sure you specify the right name in the argument list!



- "output\_test" is the name of the directory that will be created with CDFISH output in the directory specified by the third argument above. This also can be named whatever you want.
6. Check for successful model run completion: The program will provide step-by-step output in the Shell window. Once completed, a simulation time will be printed out and folders batchrun0mcrun0, batchrun0mcrun1, batchrun1mcrun0, and batchrun1mcrun1, will be created in your CDFISH home directory to store output from the separate batch and/or Monte-Carlo runs. These folders are located in the data folder specified in above step. The output folder will have a unique date/time stamp proceeding in case you want to run multiple CDFISH runs in this same directory. The program will also provide a log file with program steps in your specified output directory. If parameters are such that population becomes extinct before specified generation time, then program will end.

### 2.3.2 GUI Run

Sorry, current updates are not supporting the GUI. See versions that are less than 0.52 for GUI runs. However, latest and greatest functionalities are included in versions greater than 0.52.

## 3 Input

### 3.1 Input files

The following are the general input files used in CDFISH. See examples provided for formatting your input files.

File Header	Example	Description
Xyfilename	<p>Cdfish_test.xy - example supplied for the 64 individuals in 4 subpopulations</p> <p>*cdfish_test_NAs.csv gives you an example of how to specify your initial starting population size. This example is 64 possible habitat locations that can be filled with only 47 initialized with individuals. You must specify NA values in the 'ID' and 'sex' columns, but</p>	<p>The <math>n</math>-(<math>x,y</math>) grid location values. This is a comma delimited file with 5 column headings:</p> <p>(Subpopulation)- a unique identifier for each individual corresponding to a unique subpopulation. This is an optional tracker for individuals that may be located in designated subpopulations. If individuals are just continuously distributed, then fill with arbitrary value, like '1'. If these field is used, then subpopulations must be in sequence, e.g., 1, 2, 3, ...</p> <p>(XCOORD)-x-coordinate location,</p> <p>(YCOORD)-y-coordinate location (YCOORD),</p> <p>(ID)-a string label identifier, and</p>

	still provide the xy locations that are 'OPEN' and a 'Supopulation' identifier field.	(sex)-an initial sex assignment (use 0/1 or F/M). See xyED16.csv for an example xyfilename. The column order is necessary and header file included.*See below for specifying constant versus non-constant population sizes.
agedistribution	'N' - no age distribution file is used and age is initialized at 0 for all individuals.	<p>The distribution that is used to initialize each individuals age.</p> <p>'N' - no file is not used and age initialized with 0.</p> <p>Filename - If a filename is entered, then read in the file (for example agedistribution would be entered for the example provided). See the agedistribution.csv for formatting this file and note it must be comma delimited. The file includes two columns: the 'Age class' (start initial age of offspring as 0) and 'Number in age class', which is the number of individuals in each class. This number must sum to the carrying capacity of your study or the total number of xy location specified above.</p>
CostDistance-Matrix	Cdfish_test.cd	The cost distance matrix from source to each subpopulation. Note that if you have 4 subpopulations, then this file will be a 5x5 matrix. The first entry in this matrix is considered the source and these values are used in the migration process. The rest of the values are used for the straying process. Use a .cd extension and comma delimited values.

### 3.2 Model parameters

The following lists the model parameters used for CDFISH.

File Header	Example	Description
Mcruns - Monte Carlo Replicates	'1' - replicate runs denoted in folders labeled with 'mcrun0', 'mcrun1', ...	The repeated number of simulations to be conducted for the Monte Carlo method (i.e., the number of replicates for 1 batch of parameters).
Looptime - Generation/Time	'5'	Simulation run time [generation or year]. File output indexed from 0 - (looptime-1). For
Nthfile Choice - Generations	'Sequence' - will save a specified	The choice of a specified simulation run time to write to file. If List is entered, then read nthfile_list values

of Saved Genotypes Choice	sequence of generations.	below. If Sequence is entered, then read nthfile_seq value below.
Nthfile List - List of Generations Saved	'0 3 4'	The specified simulation run time to write to file and to calculate genetic distance matrices. These values are used if nthfile_choice = 'List'. These values must be separated with a vertical bar.
Nthfile Sequence - Sequence of Generations Saved	'1'	The specified simulation run time to write to file and to calculate genetic distance matrices. This value is used if nthfile_choice = 'Sequence'. This is the 'by' value in the sequence. For example 1 would create values starting at 0, ending at looptime-1, by every 1 generation. A value of 5 would output every 5 <sup>th</sup> generation, etc.

### 3.3 Mating parameters

The following lists the parameters used for the movement of individuals due to mating.

File Header	Example	Description
Matemoveno - Mating Movement Function	'4' - random probability function of mating cost distance.	<p>Movement function answer for mating probabilitly.</p> <p>1 = Linear <math>(1 - (1/\text{Threshold}) * \text{Cost Distance})</math></p> <p>2 = Inverse Square <math>(1 / (\text{Cost Distance}^2 + 1))</math></p> <p>3 = Nearest Neighbor (Use threshold to specify Moore neighborhood)</p> <p>4 = Random Mixing: Does not consider cost distance matrix.</p> <p>5 = Negative Exponential <math>(\text{parA} * 10^{-(\text{parB} * \text{Cost Distance})})</math></p> <p>6 = Panmictic Window: Given the cost distance matrix, then movement will be panmictic within set movement threshold.</p>
Matemove-thresh - Mating Movement Threshold	'5'	A threshold option (in cost distance units) for how far an individual can search for a mate, equivalent to the cost distance kernel. You can specify 'max' to consider all individuals for mating movement. You can also place an

		integer value in front of 'max' to consider a percent cost distance movement for mating. For example '10max' would consider all mating individuals that are within 10 percent of the maximum cost distance on the surface. You can also just specify a specific cost distance value.
Freplace - Female Replacement	'N' - females mate without replacement	If you want females to mate with replacement, then specify 'Y'.  If you want females to mate without replacement, then specify 'N'.
Mreplace - Male Replacement	'Y' - males mate with replacement	If you want males to mate with replacement, then specify 'Y'.  If you want males to mate without replacement, then specify 'N'.
Reproage - Reproduction Age	'0' - individuals start reproducing at age 0.	The age at which individuals can start to reproduce. Use with overlapping generations, i.e, agemortperc not set to 100. If an age distribution file is not used, then age gets initialized with 0 at the starting generation.

### 3.4 Dispersal/Migration parameters

Here lists the parameters used for the movement of individuals with regards to offspring dispersal and migration.

File Header	Example	Description
Dispmoveno - Dispersal Movement Choice	'2' - inverse- square probability function for cost distance movement.	This is the function answer for movement for female dispersal probability.  1 = Linear $(1 - (1/\text{Threshold}) * \text{Cost Distance})$  2 = Inverse Square $(1 / (\text{Cost Distance}^2 + 1))$  3 = Nearest Neighbor (Use threshold to specify Moore neighborhood)  4 = Random Mixing: Does not consider dispersal cost distance matrix.  5 = Negative Exponential $(\text{parA} * 10^{(-\text{parB} * \text{Cost Distance})})$  6 = Min-max scaled: $(1 / \min(\text{Cost Distance}) - \max(\text{Cost Distance})) * (\text{Cost Distance} - \min(\text{Cost Distance})) + 1$ where min and max are the minimum and maximum cost distance values in the given matrix.

		7 = User defined min-max scaled: $(1 / \min(\text{Cost Distance}) - \max(\text{Cost Distance})) * (\text{Cost Distance} - \min(\text{Cost Distance})) + 1$ where min and max are the minimum and maximum cost distance values specified by the dispmovethresh option.
Dispmove-thresh - Dispersal Movement Threshold	'max' - will travel the entire landscape as an inverse square probability of cost distance on the surface.	<p>A threshold option (in cost distance units) for how far an individual male offspring can disperse. You can specify 'max' to consider all individuals for mating movement. You can also place an integer value in front of 'max' to consider a percent cost distance movement for mating. For example '10max' would consider all mating individuals that are within 10 percent of the maximum cost distance on the surface. You can also just specify a specific cost distance value.</p> <p>If 'dispermoveno' option '7' is used, then two values must be specified for the minimum and maximum threshold values and separated by a ' '.</p>
Residency	'10 10 10 10'	This is the percent of the subpopulation's open locations that take up residency from the offspring pool. There needs to be a unique value for each subpopulation separated by a ' '.
Staying	'10 10 10 10'	This is the percent for each subpopulation for the offspring that do not make it back to their own subpopulation that stray to another subpopulation. There needs to be a unique value for each subpopulation separated by a ' '.

### 3.5 Offspring parameters

The following lists the parameters to deal with offspring births and deaths.

File Header	Example	Description
Offno - Offspring Choice	'2' - Poisson distribution	<p>This is the number of offspring each mate pair can have.</p> <p>1 - for a random draw,  2 - for Poisson draw.  3 - for a constant number of offspring.  4 - for an equal clutch size for each female.</p>

		Note that option 1 - 3 will assign a clutch size to each mate event. For example, if a female mates with 4 males and offno is set to 3 with lambda of 2, then she will have 8 offspring. Option 4 will assign a clutch size to each female that mated. For example, if a female mates with 4 males and offno is set to 4 with lambda of 2, then she will have 2 offspring total and the father(s) is randomly selected from the 4 mating events.
Lmbda	'5' - Poisson distribution with mean lambda 5.	<p>The parameter value used with Offspring Number.</p> <p>If offno = 1, then lmbda is the max range value between 0 - lmbda to draw randomly from.</p> <p>If offno = 2, then lmbda is the Poisson mean for the litter size.</p> <p>If offno = 3, then lmbda is the constant litter size value.</p>
Female-percent - Female offspring	'50' - 50% random female assignment	Percent number of female born in each litter. This is a random assignment from given percentage, i.e., even if you set this to 50% some generations could have 499 female births and 501 male births, for example.
birthmortperc	'0' - 0% birth mortality.	Percent mortality in the offspring population.
Agemortperc - Age Structure Mortality	'100' - Age structured mortality here with non-overlapping generations.	<p>This is the mortality field for each age class. Each age class must be separated with a ' '.  If looking at non-overlapping generations, then specify numbers: offspring mortality (which is the same as the birth mortality in this situation) and adult mortality, where adult mortality is set to 100%.</p> <p>If looking at overlapping generations with a supplied age distribution class (e.g., age classes 0, 1, 2, 3, 4), then a mortality percentage must be entered for each age class (e.g., 50 20 20 30 100). Note that if you do not put 100 for the final age class, then the population will be allowed to go beyond your last age class specified. All age classes beyond specified in file will get lumped into last mortality category.</p>

Equalsex-ratio - Equal Sex Ratio for Offspring	'Y'	<p>This ensures an exact equal sex ratio for the following options:</p> <p>'WrightFisher' - The answer to have every generation start with equal sex ratios. CAREFUL, this parameter is not realistic for non panmictic populations and should only be used to match Wright-Fisher assumptions on equal sex ratios.</p> <p>'AtBirth' - This will ensure that each litter is equal sex ratio, but not necessary ensure that the generation will result in equal sex ratio after dispersal do to unbiased dispersal parameters and stochastic dispersal (i.e., random offspring chosen for dispersal).</p> <p>N - This option is not used and offspring sex is assigned using the Female-percent parameter above.</p>
--	-----	---

### 3.6 Genetic parameters

The following lists the parameters associated with the initialization of the genotypes, mutation rates, and selection.

File Header	Example	Description
Mutate	'0.0005'	The k-allele model mutation rate.
Loci	'10'	The number of loci.
Alleles	'10'	The number of alleles per locus. If you want a variable initial number of alleles per locus, then separate the numbers with a " ", e.g., 2 3 8 3, but make sure that you enter in the total loci amount, else an error will be thrown.
Intialize Genes Answer	'random'	<p>The choice for how to initialize the genotype for each <math>n-(x,y)</math> individuals.</p> <p>If 'random' is entered, then the genotypes get a random assignment and the population is at a maximum genetic diversity.</p> <p>If 'file' is entered, then the genetics get drawn from the allele frequency distribution file (specify file name in next column, allelfreqfilename).</p> <p>If 'known' is entered, then the genotypes are directly read from a given known file. This file is very similar to the initial xyfilename and example cdfish_test_known.csv is supplied with test data files.</p>

Allefreq- filename	'N'- allele frequency file not used.	The allele frequency distribution for each locus, used to initialize the model's $n$ individual's genotype. If you want to use a frequency distribution file, you must set Initialize Genes Answer to equal 'file' and then enter in the filename in this field. See allelefrequency.csv example file for formatting this file. It is basically a column of allele frequencies and make sure the length of the column equals your starting loci * starting alleles. Also, make sure you include the file extension, e.g., allelefrequency.csv.
cdevolveans	'N'	This is the answer for how many loci are under selection.  Use 'N' to turn off CDEVOLVE.  Use '1' for natural selection with 1 locus.  2 or more not coded in yet.
Burningen	10	This is the generation or year that the selection surface will begin operating on the locus or loci under selection, specified in previous field (cdevolveans). If 'N' is specified for 'cdevolveans', then this field is ignored.
<i>The following are the fitness surfaces for when CDEVOLVE Answer is 1. This corresponds to 1 locus that is under selection. The x,y location of an offspring is matched up with the closest x,y fitness value for the offspring's corresponding fitness surface defined by the genotype that the offspring has. That value then becomes the individual offspring mortality percentage. An offspring becomes more or less fit relative to the other offspring at that generation as a function of its genotype and where it occurs on a surface. These are ASCII formatted files with 6 lines of header information and values that are space delimited and represent percent mortality [0 - 100]. See fitvals.txt for an example fitness surface and its format. Examples for uniform and gradient spatial selection files are given.</i>		
Fitness_AA	FitvalsN100_S0.txt	When CDEVOLVE Answer is 1, then this is the offspring viability selection surface for AA. If offspring has AA, then this mortality fitness surface is used. (Gradient surface selected for in the south with completely lethal in the North.)



Fitness_Aa	Fitvals50.txt	When CDEVOLVE Answer is 1, then this is the offspring viability selection surface for Aa. If offspring has Aa, then this mortality fitness surface is used. (Uniform selection surface $s = 0.5$ )
Fitness_aa	FitvalsN0_S100.txt	When CDEVOLVE Answer is 1, then this is the offspring viability selection surface for aa. If offspring has aa, then this mortality fitness surface is used. (Gradient surface selected for in the north with completely lethal in the South.)

#### 4 Output

The following is a list of the output produced from CDFISH.

Example	Description
Grid0.csv	The genotype for each $n-(x,y)$ across specified generations.
Output.csv	A file of the various data measures at each generation: (Population, Females, Males, Migrants, Residors, Strayers, Dispersal Deaths, Offspring Deaths, Selection Deaths, Births, Adult Deaths, Alleles, $H_e$ , $H_o$ , Alleles Mutated)

Folders will be created in your project directory labeled with a unique time stamp (dos convention), e.g., 1332964297batchrun0mcrun0. Monte Carlo runs will be uniquely labeled mcrun0, mcrun1,... and each batch run will be uniquely labeled batchrun0, batchrun1, ... . In each folder you will see grid{generation}.csv files that list each individual's genotype, spatial locations, age, sex, and infection status. If you specified a grid format option, then you will additionally see the format followed by grid{generation}.csv.

In addition, an output.csv is automatically created for each batch and Monte Carlo run. These are population based metrics calculated at each generation. The following is a summary of each calculation:

- Year - This is the generation time or year if using overlapping generations.
- Population - The total population in each generation. If you specify subpopulations in the first column of your xy.csv file, then this field will be separated by '|'. The first value is the

total population size and each additional value corresponds to the subpopulation sizes in order.

- ToTFemales - The total number of females in each generation. If you specify subpopulations in the first column of your xy.csv file, then this field will be separated by '|'. The first value is the total female size and each additional value corresponds to the subpopulation sizes in order.
- ToTMales - The total number of males in each generation. If you specify subpopulations in the first column of your xy.csv file, then this field will be separated by '|'. The first value is the total male size and each additional value corresponds to the subpopulation sizes in order.
- Migrants - The number of dispersers in each generation that make it to the next generation.
- Residors - The number of residents in each generation that make it to the next generation.
- Strayers - The number of strayers in each generation that make it to the next generation.
- DisperseDeaths - The number of dispersers in each generation that do not make it to the next generation. Death due to a weighted random draw from probability based on functional cost distance choice. Or the number of individuals that could not disperse (migrate or stray) due to high cost to moving to an open locations.
- Offspring Deaths - The individuals left over from dispersal process (discarded).
- Selection Deaths - Individuals that made it back to population but did not survive do to differential mortality of genotype and spatial selection.
- Births - The number of offspring born at that generation.
- Adult Deaths - The number of deaths of the adult population, not the offspring born that year/generation. The '|' separates the age class deaths for overlapping generations for the ordered age class in that generation. Note that some years may not have all age classes and not reported.
- Alleles - This is the total number of unique alleles at each generation. This value can be calculated automatically for specified subpopulations if different subpopulations were designated in the initial xyfilename. If there are '|', then the first value corresponds to the total alleles in the population and subpopulation values follow after.
- He - This is the expected heterozygosity value at each generation. This value can be calculated automatically for specified subpopulations if different subpopulations were designated in the initial xyfilename. If there are '|', then the first value corresponds to the total He in the population and subpopulation values follow after.
- Ho - This is the observed heterozygosity value at each generation. This value can be calculated automatically for specified subpopulations if different subpopulations were

designated in the initial xyfilename. If there are '\|', then the first value corresponds to the total  $H_o$  in the population and subpopulation values follow after.

- Mutations - The total number of mutations at each generation.

## 5 General issues

### 5.1 How to obtain CDFISH

The program is freeware and can be downloaded at <http://cel.dbs.umd.edu/software/CDFISH/> with information for users, including manual instructions, FAQ, publications, ongoing research, and developer involvement.

### 5.2 Debugging and troubleshooting

For help with installation problems please check first for postings at our web site. Otherwise, please report problems including any bugs, to me at [erin.landguth@mso.umd.edu](mailto:erin.landguth@mso.umd.edu).

### 5.3 How to cite CDFISH

This program was developed by Erin Landguth and Clint Muhlfeld. The reference to cite is:

Landguth EL, Muhlfeld CC, Luikart G (2012) CDFISH: an individual-based, spatially-explicit, landscape genetics simulator for aquatic species in complex riverscapes. *Conservation Genetics Resources*, 4, 133-136.

### 5.4 Disclaimer

The software is in the public domain, and the recipient may not assert any proprietary rights thereto nor represent it to anyone as other than a University of Montana-produced program (version 1.x). CDFISH is provided "as is" without warranty of any kind, including, but not limited to, the implied warranties of merchantability and fitness for a particular purpose. The user assumes all responsibility for the accuracy and suitability of this program for a specific application. In no event will the authors or the University be liable for any damages, including lost profits, lost savings, or other incidental or consequential damages arising from the use of or the inability to use this program.

Although this program has been used by the USGS, no warranty, expressed or implied, is made by the USGS or the United States Government as to the accuracy and functioning of the program and related program material nor shall the fact of distribution constitute any such warranty, and no responsibility is assumed by the USGS in connection therewith.

We strongly urge you to read the entire documentation before ever running CDFISH. We wish to remind users that we are not in the commercial software marketing business. We are scientists who recognized the need for a tool like CDFISH to assist us in our research on land-

scape genetic issues. Therefore, we do not wish to spend a great deal of time consulting on trivial matters concerning the use of CDFISH. However, we do recognize an obligation to provide some level of information support. Of course, we welcome and encourage your criticisms and suggestions about the program at all times. We will welcome questions about how to run CDFISH or interpret the output only after you have read the entire documentation. This is only fair and will eliminate many trivial questions. Finally, we are always interested in learning about how others have applied CDFISH in ecological investigation and management application. Therefore, we encourage you to contact us and describe your application after using CDFISH.

*We hope that CDFISH is of great assistance in your work and we look forward to hearing about your applications.*

## 6 References

- Allendorf FW, Luikart G (2007) Conservation and the genetics of populations. Blackwell, Malden, MA.
- Balkenhol N, Gugerli F, Cushman SA, Waits LP, Coulon A, Arntzen JW, Holderegger R, Wagner HH (2009) Identifying future research needs in landscape genetics: where to from here? *Landscape Ecology* 24:455-463
- Balkenhol N, Landguth EL (2011) Simulation modeling in landscape genetics: on the need to go further. *Molecular Ecology* 20:667-670
- Cushman SA, McKelvey KS, Hayden J, Schwarz MK (2006) Gene flow in complex landscapes: testing multiple hypotheses with casual modeling. *The American Naturalist* 168:486-499
- Cushman SA, Landguth EL (2010) Spurious correlations and inferences in landscape genetics. *Molecular Ecology* 19:3592-3602
- Dijkstra EW (1959) A note on two problems in connexion with graphs. *Numerische Mathematik* 1:269-271
- Dunning JB, Danielson BJ, Pulliam HR (1992) Ecological processes that affect populations in complex landscapes. *OIKOS* 65:169 -175
- Epperson BK, Mcrae BH, Scribner K, Cushman SA, Rosenberg MS, Fortin MJ, James PMA, Murphy M, Manel S, Legendre P, Dale MRT (2010) Utility of computer simulations in landscape genetics. *Molecular Ecology* 19:3549-3564
- Epps CW, Wehausen JD, Bleich VC, Torres SG, Brashares JS (2007) Optimizing dispersal and corridor models using landscape genetics. *Journal of Applied Ecology* 44:714-724
- Holderegger R, Wagner HH (2006) A brief guide to Landscape Genetics. *Landscape Ecology* 21:793-796

- Landguth EL, Cushman SA (2010) CDPOP: A spatially-explicit cost distance population genetics program. *Molecular Ecology Resources* 10:156-161
- Landguth EL, et al (2010a) Quantifying the lag time to detect barriers in landscape genetics. *Molecular Ecology* 19:4179-4191
- Landguth EL, et al (2010b) Relationships between migration rates and landscape resistance assessed using individual-based simulations. *Molecular Ecology Resources* 10:854-862
- Landguth EL, Hand BK, Glassy JM, Cushman SA (2011) UNICOR: A species connectivity and corridor network simulator. *Ecography*. Submitted.
- McRae BH, Beier P (2007) Circuit theory predicts gene flow in plant and animal populations. *Proceedings of the National Academy of Science USA* 104:19885-19890
- Parmesan C (2006) Ecological and evolutionary responses to recent climate change. *Annual Review of Ecological Systems* 37:637-669
- Parmesan C, Yohe G (2003) A globally coherent fingerprint of climate change impacts across natural systems. *Nature* 421:37-42
- Ray N (2005) PATHMATRIX: a GIS tool to compute effective distances among samples. *Molecular Ecology Notes* 5:177-180
- Root TL, Price JT, Hall KR, Schneider SH, Rosenzweig C, Pounds JA (2003) Fingerprints of global warming on wild animals and plants. *Nature* 421:57-60
- Spear S, Balkenhol N, Fortin M-J, McRae B, Scribner K (2010) Use of resistance surfaces for landscape genetic studies: Considerations for parameterization and analysis. *Molecular Ecology* 19:3576-3591
- Storfer A, Murphy MA, Spear SF, Holderegger R, Waits LP (2010) Landscape genetics: where are we now? *Molecular Ecology* 19:3496-3514
- Walther G-R, Post E, Convey P, Menzel A, Parmesan C, Beebee TJC, Fromentin J-M, Hoegh-Guldberg O, Bairlein F (2002) Ecological responses to recent climate change. *Nature* 416:389-395

## 7 Acknowledgements

We are grateful to Leslie Jones and Helen Neville for their valuable input in the model development. We thank the numerous beta testers. Funding was provided by the Great Northern Landscape Conservation Cooperative (U.S. Department of Interior) through the Rocky Mountain Cooperative Ecosystem Study Unit at the University of Montana, Missoula.