

# ssmc User Guide

*Bill Peterman*

*2017-01-03*

**ssmc: An R Package for assessing source-sink population status and contribution of populations to the metapopulation using demographic connectivity networks and Monte Carlo simulation**

## Background

The functions found in this package were originally developed as part of research focused on understanding source-sink populations dynamics of amphibians. This research was supported by the Department of Defense Strategic Environmental Research Development Program (SERDP; Project #RC-2155). This document is meant to serve as a supplement to the documentation found in the `help` files of `ssmc`. The math behind the model is presented below, but understanding the math is not essential to applying and the using the model. Please carefully read this User Guide as well as all documentation related to the functions in the R package.

## Modeling process

There are 5 steps in the modeling process:

1. Select values for parameters
  - population size – assessed separately for each population
  - size of metamorphs<sup>+</sup> – assessed separately for each population
  - proportion of population that is philopatric – equivalent among all populations at each iteration
  - probability of surviving to adulthood<sup>+</sup> – equivalent among all populations at each iteration
  - dispersal distance – equivalent among all populations at each iteration
2. Determine probability of dispersal (see ‘Calculating dispersal’ below)
3. Determine the number of individuals dispersing between populations
4. Determine the contribution and influence of each pond to the mean metapopulation lifetime
5. Repeat steps 1–4 for the specified number of Monte Carlo iterations

Each of the above stochastic variables is drawn from a normal distribution with a mean and standard deviation that is specified based on your system. At each Monte Carlo iteration, different values are drawn.

After all iterations are complete, the ratio of immigrant:philopatric ( $\frac{i}{p}$ ) individuals is assessed. If  $\frac{i}{p} > 1$ , the population is considered a sink, if  $\frac{i}{p} \leq 1$  the population is considered a source. This evaluation is made for each iteration, and the frequency at which a population acts as a source or sink is then determined. Additionally, the average number of emigrants, immigrants, and philopatric individuals is determined for each population, as well as the change in mean metapopulation lifetime when each population is removed.

<sup>+</sup> The size of metamorphs, if specified, is used to determine the population-specific probability of survival to adulthood using the equation:

$$\text{logit}(\text{survival}) = -1.366 + 0.87 * \text{size},$$

If metamorph size is not provided, then survival probability is drawn from a distribution and will be uniform across all populations. The above equation comes from Altwegg & Reyer (2003)

## Calculating dispersal

The probability of successful dispersal from population  $i$  to population  $j$  is calculated as:

$$m_{ij} = \exp(-\alpha d_{ij}),$$

for  $j \neq i$  and  $m_{ii} = 0$ . Average dispersal distance is  $1/\alpha$  and  $d_{ij}$  is the distance between populations  $i$  and  $j$ . The probabilities of  $m_{ij}$  are standardized so that rows  $i$  sum to one by

$$m'_{ij} = \frac{m_{ij}}{\sum_j m_{ij}}.$$

The number of individuals,  $n$ , dispersing from population  $i$  to  $j$  is then determined using a multinomial distribution such that

$$P(n_1, n_2, \dots, n_k) = \frac{N!}{(n_1! n_2! \dots n_k!)} * (p_1^{n_1} p_2^{n_2} \dots p_k^{n_k})$$

where

- $N$  is the number of emigrants from  $i$
- $k$  is the total number of populations that individuals could disperse to
- $n_{ij}$  is the number of dispersers from population  $i$  to  $j$
- $p_{ij}$  is the probability dispersing from population  $i$  to  $j$ , as determined from  $m'_{ij}$

Values  $n_{ij}$  are components of the landscape matrix  $M_{ij}$ . Summing across columns,  $j$ , of matrix  $\mathbf{M}$  gives the total number of emigrants leaving population  $i$ , while summing across rows,  $i$ , gives the total number of immigrants entering population  $j$ .

$$emigrants_i = \sum_{n=1}^j M_{ij}$$

$$immigrants_j = \sum_{n=1}^i M_{ij}$$

## Determining effect on Metapopulation Mean Lifetime (MMLT)

Metapopulation Mean Lifetime is a global measure that represents the persistence or viability of a metapopulation (Drechsler 2009), and was adapted from work by Frank & Wissel (1998, 2002). MMLT has been further adapted by Kininmonth et al. (2010) for applications with graph theory. This framework approach is sensitive to network topology, specifically spatial locations and size of populations. Calculation of MMLT is as follows: First an extinction rate  $v_i$  is calculated as:

$$v_i = \varepsilon A_i^{-\eta},$$

where  $\varepsilon$  (default = 1) and  $\eta$  (default = 0.5) are parameters relating the local extinction rate to patch area,  $A$ . Then, based on in-degree ( $immigrants_i$ ) and out-degree ( $emigrants_i$ ) edge weights from the network graph,  $C_i^{(in)}$  (colonizability) and  $C_i^{(out)}$  (colonization strength) are calculated:

$$C_i^{(in)} = immigrants_i$$

$$C_i^{(out)} = emigrants_i$$

Colonizability and colonization strength are then multiplied by the expected lifetime of a patch,  $1/v_i$ .

$$u_i^{(out)} = \frac{C_i^{(out)}}{v_i}$$

$$u_i^{(in)} = \frac{C_i^{(in)}}{v_i}.$$

The colonization-extinction ratio of patch  $i$  is the harmonic mean of  $u_i^{(out)}$  and  $u_i^{(in)}$ :

$$U_i = \left( \frac{1}{2} (u_i^{(in)})^{-2} + \frac{1}{2} (u_i^{(out)})^{-2} \right)^{-\frac{1}{2}},$$

which is then aggregated for the entire network:

$$q = \prod_{i=1}^N \{\max(U_i, \sqrt{2})\}^{1/N}.$$

The local extinction rates,  $v_i$ , are also aggregated as a geometric mean:

$$v = \prod_{i=1}^N v_i^{1/N}.$$

The metapopulation mean lifetime is then:

$$\text{MMLT} = \frac{1}{v} \sum_{i=1}^N \sum_{k=1}^N \frac{1}{k} \frac{(N-i)!}{(N-k)!} \frac{1}{(N-1)^{k-1}} q^{k-1}.$$

However, because of the large number of patches likely to be used with this approach, the approximating equation of Drechsler (2009)

$$\ln(vT) \approx N \left( \frac{1}{q} + \ln(q) - 1 \right).$$

The importance of population  $i$  to the metapopulation mean lifetime is then determined as the amount of change ( $\Delta\text{MMLT}$ ) in MMLT when population  $i$  is removed:

$$\Delta\text{MMLT} = \frac{(\log_{10}(\text{MMLT.new} + 1) - \log_{10}(\text{MMLT.old} + 1))}{\log_{10}(\text{MMLT.old} + 1)}$$

The assessment of population contribution to the metapopulation as measured through  $\Delta\text{MMLT}$  is implemented through the `site.analysis` functions. In addition to calculating  $\Delta\text{MMLT}$  for existing populations, it is possible to assess the impact of creating new habitat or restoring existing habitat. This approach calculates MMLT when potential population  $i$  is created or restored, and the average MMLT over all Monte-Carlo iterations is then multiplied by the frequency that population  $i$  is colonized by 2 or more individuals. This adjustment ensures that populations that are infrequently colonized are given lower priority for creation or restoration. The `best.locale` function allows for these assessments.

## Setup

### Install necessary software and packages

You will also need to have R  $\geq$  v3.0 installed. I would highly recommend installing R studio when working with R.

### Getting the R Package

First, install `ssmc` from GitHub. This will require the `devtools` package

```
# Install 'devtools' package, if needed
if(!("devtools" %in% list.files(.libPaths())) {
  install.packages("devtools",
    repo = "http://cran.rstudio.com",
    dep = TRUE)
}

library(devtools) # Loads devtools

install_github("wpeterman/ssmc") # Download package
```

Load `ssmc` and clear your workspace.

```
library(ssmc)
rm(list = ls())
```

## Assessing the source-sink status and importance of populations to the metapopulation

The `ssmc` package comes with a simulated sample data set called `site.dat`. After loading the `ssmc` package, you can explore the structure and format of the data file.

```
str(site.dat)

## 'data.frame': 25 obs. of 8 variables:
## $ site : int 1 2 3 4 5 6 7 8 9 10 ...
## $ x : num 17.1 13.1 82 15.5 53.3 ...
## $ y : num 0.0656 0.1824 0.4946 0.38 0.0204 ...
## $ A : num 74.5 15.9 48.9 74.2 68.4 ...
## $ ab.mean : num 0 67.6 0 0 0 ...
## $ ab.sd : num 0 15.1 0 0 0 ...
## $ svl.mean: num -0.19 -0.783 -0.551 1.381 -0.132 ...
## $ svl.sd : num 0.188 0.012 0.905 0.185 0.847 ...
```

There are seven variables included in the data frame. This format is not essential to using the functions of `ssmc`, as parameter values are individually specified for each function.

To assess source-sink status of existing populations, use the `site.analysis` function. Following the example provided in the help documentation, run the function.

```
site_results <- site.analysis(sites = site.dat[,1:4],
  pop.abun = site.dat[,5:6],
  met.size = site.dat[,7:8],
  prop.philo = 0.95,
  sd.philo = 0.05,
  lower.upper_philo=c(lower=0, upper=1),
  lower.upper_survive = c(lower=0, upper=1),
  dispersal = 25,
  sd.dispersal = 10,
  lower.upper_dispersal = c(lower=10, upper=Inf),
  eps = 1,
  mu = 2,
  eta = 0.5,
  iterations = 10,
  seed = 123)
```

The results of `site.analysis` contain both a summary data frame as well as the raw data from each Monte Carlo iteration. The summary data frame can be exported and sorted based on the site/population rank

using the `ssmc_summary` function.

```
site_summary <- ssmc_summary(ssmc_results = site_results)
str(site_summary)
```

```
## 'data.frame':   25 obs. of  13 variables:
## $ site      : int  20 16 14 12 13 7 24 15 25 10 ...
## $ x         : num  28.69 38.95 42.08 4.33 58.1 ...
## $ y         : num  0.949 0.462 0.457 0.744 0.246 ...
## $ immig     : num  1.1 0.8 1 1.2 0.9 1.2 1.2 1.1 0.9 0.8 ...
## $ philo     : num  58.6 86.3 8.4 13.3 17.2 5 8.6 79.6 6.4 8.8 ...
## $ emig      : num  3.6 4.7 1.1 0.9 1 0.4 0.4 5.4 0.1 0.2 ...
## $ in_deg    : num  0.9 0.8 0.7 1.1 0.9 1.2 1 1 0.8 0.8 ...
## $ out_deg   : num  3.3 4 1 0.9 0.9 0.4 0.4 4.6 0.1 0.2 ...
## $ pct_immig : num  0.0209 0.0169 0.0996 0.0787 0.0573 ...
## $ pct_sink  : num  0 0 0 0 0 0.2 0.1 0 0 0.3 ...
## $ pct_src   : num  1 1 1 1 1 0.8 0.9 1 1 0.7 ...
## $ delt_mmlt : num  -0.0867 -0.0623 -0.0492 -0.0439 -0.0426 ...
## $ rank      : int   1 2 3 4 5 6 7 8 9 10 ...
```

It is possible to sort the summary data frame by using any of the columns of the output data frame.

Elements of the summary data frame include:

1. **immig**: The average number of immigrants entering each population
2. **philo**: The average number of philopatric individuals
3. **emig**: The average number of emigrants dispersing from a population
4. **pct\_immig**: Average percentage of a population that is comprised of immigrants
5. **pct\_sink**: The frequency that a population acted as a sink (i.e.  $\frac{i}{p} > 1$ ) across all simulations
6. **pct\_src**: The frequency that a population acted as a source (i.e.  $\frac{i}{p} \leq 1$ ) across all simulations
7. **delta\_mmlt**: The change in metapopulation mean lifetime when population  $i$  is removed. This is calculated as:

$$\Delta\text{MMLT} = (\log_{10}(\text{MMLT.new} + 1) - \log_{10}(\text{MMLT.old} + 1)) / \log_{10}(\text{MMLT.old} + 1)$$

8. **rank**: The rank order importance of each location as determined by  $\Delta\text{MMLT}$

Finally, the distance between connected sites is monitored throughout the analysis and is summarized in the results output `$connect.df`. The mean and standard deviation of the distance between connected sites can be reviewed by

```
site_results$connect.df
```

```
##   avg_dist sd_dist  avg_max   sd_max
## 1 13.82469 10.4314 30.15437 6.363448
```

### Comments on using `site.analysis`

1. This is a rather computationally intensive process, so the number of Monte Carlo iterations has been kept to a minimum for example purposes. It is highly recommended that 1,000–10,000 iterations be used when conducting real analyses.
2. Because `met.size` was specified, there is no need to specify `prop.survive` and `sd.survive`.
3. `met.size`, if provided, must be in standard units with a mean of zero and standard deviation of one
4. The units for dispersal distance are arbitrary, but must correspond with the distances that will be calculated between populations based on the xy coordinates. The model was originally applied to populations with xy coordinates in UTM, and dispersal distances in meters.

5. The `lower.upper_` parameters are used to specify lower and upper threshold values when sampling from a normal distribution.

### Assessing the potential of a new or restored population to contribute to the metapopulation

The use of the `best.locale` function is identical to `site.analysis` except that a data frame of potential sites is provided to be evaluated as part of the metapopulation network. `potential.dat` is provided with the package as an example of a potential location data frame.

```
best_results <- best.locale(sites = site.dat[,1:4],
                           potential.sites = potential.dat,
                           pop.abun = site.dat[,5:6],
                           met.size = site.dat[,7:8],
                           prop.philo = 0.95,
                           sd.philo = 0.05,
                           lower.upper_philo=c(lower=0, upper=1),
                           lower.upper_survive = c(lower=0, upper=1),
                           dispersal = 25,
                           sd.dispersal = 10,
                           lower.upper_dispersal = c(lower=10, upper=Inf),
                           eps = 1,
                           mu = 2,
                           eta = 0.5,
                           iterations = 10,
                           seed = 123)
```

At each Monte Carlo iteration, the function evaluates the MMLT of the network with the population included. Because these are hypothetical populations with no abundance, size, or survival parameters, these parameters are set to the mean value of all non-zero populations for that iteration of the analysis. As such, each potential location is evaluated as if it were an ‘average’ population on the landscape.

The output from the `best.locale` analysis is similar to that of `site.analysis` and the summary data frame can again be extracted and sorted using `ssmc_summary`

```
best_summary <- ssmc_summary(ssmc_results = best_results, 'rank_adj')
str(best_summary)
```

```
## 'data.frame': 25 obs. of 8 variables:
## $ site : int 5 3 13 11 24 9 21 4 8 12 ...
## $ x : num 16.3 55.2 76.8 61.4 36.3 ...
## $ y : num 15.02 10.38 8.55 43.19 40.44 ...
## $ mmlt : num 2.73 2.68 2.64 2.64 2.63 ...
## $ rank : num 1 2 4 3 6 7 8 9 10 13 ...
## $ frq_col : num 0.4 0.3 0.3 0.2 0.2 0.2 0.2 0.2 0.2 0.2 ...
## $ mmlt_col: num 1.092 0.804 0.793 0.529 0.526 ...
## $ rank_adj: num 1 2 3 4 5 6 7 8 9 10 ...
```

By specifying ‘rank\_adj’, the summary output is sorted by the adjusted rank column (adjusted for the frequency that a population is colonized by 2 or more individuals across all Monte-Carlo simulations).

Assessment of existing sites for restoration can be done by specifying `restore = TRUE` when running `best.locale`. In doing so, the function will identify all sites that currently have an abundance of zero, and will then assess MMLT when each is ‘restored’ to be an average population.

```
restore_results <- best.locale(sites = site.dat[,1:4],
                              restore = TRUE,
                              potential.sites = potential.dat,
```

```

pop.abun = site.dat[,5:6],
met.size = site.dat[,7:8],
prop.philo = 0.95,
sd.philo = 0.05,
lower.upper_philo=c(lower=0, upper=1),
lower.upper_survive = c(lower=0, upper=1),
dispersal = 25,
sd.dispersal = 10,
lower.upper_dispersal = c(lower=10, upper=Inf),
eps = 1,
mu = 2,
eta = 0.5,
iterations = 10,
seed = 123)

```

Summarize the output based on the adjusted rank:

```

restore_summary <- ssmc_summary(ssmc_results = restore_results, 'rank_adj')
str(restore_summary)

```

```

## 'data.frame':  9 obs. of  8 variables:
## $ site      : int  8 22 23 5 4 1 18 6 3
## $ x         : num  20.9 60.2 17.4 53.3 15.5 ...
## $ y         : num  0.2067 0.2905 0.5194 0.0204 0.38 ...
## $ mmlt      : num  3.3 3.89 3.68 2.69 2.66 ...
## $ rank      : num  3 1 2 6 7 8 4 5 9
## $ frq_col   : num  0.4 0.3 0.3 0.4 0.3 0.3 0.1 0.1 0.1
## $ mmlt_col  : num  1.322 1.167 1.104 1.076 0.799 ...
## $ rank_adj  : num  1 2 3 4 5 6 7 8 9

```

## Citations

Altwegg, R., and H.-U. Reyer. 2003. Patterns of natural selection on size at metamorphosis in water frogs. *Evolution* 57:872–882.

Drechsler, M. 2009. Predicting metapopulation lifetime from macroscopic network properties. *Mathematical Biosciences* 218:59–71.

Frank, K., and C. Wissel. 1998. Spatial aspects of metapopulation survival – from model results to rules of thumb for landscape management. *Landscape Ecology* 13:363–379.

Frank, K., and C. Wissel. 2002. A formula for the mean lifetime of metapopulations in heterogeneous landscapes. *The American Naturalist* 159:530–552.

Kininmonth, S., M. Drechsler, K. Johst, and H. P. Possingham. 2010. Metapopulation mean life time within complex networks. *Marine Ecology Progress Series* 417:139–149.