Optimizing corridor placement using simulated annealing

Michael D. Catchen [1,2]

[1] McGill University; [2] Québec Centre for Biodiversity Sciences

Correspondance to:
Michael D. Catchen — `michael.catchen@mail.mcgill.ca`

how many different chapter ones will i have hmmm

Keywords:
pandoc
pandoc-crossref
github actions

1 Introduction

Human activity has rapidly reshaped the face of Earth's surface, leaving fragments of patchy habitat. Although there is no shortage of debate as to the effects of fragmentation *per se* on biodiversity and ecosystem function (**cite?**), it is generally accepted that the combination of habitat and ensuing subdivision produce negative outcomes for ecosystem function and services (**resasco?** review).

In order to mitigate the consequences of landscape change on ecosystems, developing landscape *corridors* has seen much attention in the last several decades. Bit more evidence for corridors here. But still, the spatter of fragments in a landscape, where should ecologists choose to use their limit resources to build a corridor?

Here we propose to answer that question by proposing an algorithm to estimate the landscape modification that results in optimizing a specific ecosystem process (in this paper maximizing the time until extinction of a metapopulation, although the algorithm and associated software can be generally applied to any process-based model with a quantifiable target state).

Although algorithms have been proposed for this (**peterman?** etc), they are focused on finding the where the paths of least existance for a given species is given data on that species dispersal.

2 An algorithm for optimizing corridor placement

Start with some definitions and notation.

Define the set of possible landscape modifications, $\mathfrak{M}$, in optimization language called the *search-space*. Introduce uncountability argument of this space.

Because we cannot test every possible modification in $\mathfrak{M}$, we use simulated annealing, a method for estimating the global optimum of functions with NP search-spaces.

2.1. Proposing landscape modifications This is really important. We propose (no pun intended) several algorithms for generating landscape modifications. Some of the details here might have to go in a supplement/appendix.

2.1.1 Graph-based Consider only modifications that consist of connecting nodes. TODO: only choose topologies from the minimum spanning tree of the nodes.

2.1.1.1 The two stage approach Stage-one: accept a new topology of connected nodes with probability in proportion to chain temperature (see next section).

Stage-two: modify way that the connection for a given topological structure is chosen. Because we are working in a 2D raster, all distances between points are Manhattan distances, and any link between points is composed of x horizontal steps and y vertical steps. There are thus $2^{\min(x,y)}$ ways to connect two nodes that far apart.

2.1.2 Not graph-based The reason to avoid this is because the search-space grows much faster with lattice size and budget. That being said, we can use some simply heuristics to weight proposals using "common-sense."

2.2. Simulated annealing to explore the space of landscape modifications

The transition probability function, q, which gives the probability of moving from one modification $i \in \mathbb{M}$ to a new proposed state $j \in \mathbb{M}$, as a function of a chains temperature.

Here we define $q(i, j)$ using a logistic function,

$$q(i, j, \alpha) = \frac{1}{1 + e^{\alpha(s(j) - s(i))}}$$

$s(i)$ is the function that gives the score of a proposed modification. Here, the mean time to extinction.

Simulated annealing can be written described as the following.

A markov-chain, denoted π_α

Figure 1: concept fig

2.3. Process-based optimization (using occupancy dynamics)

Here we use occupancy dynamics as the process, although we emphasize that this method works for arbitrary process models and is instead limited only by the computational demands of a given process model.

Compute new resistance surface which gives pairwise potential values for each pair of points.

2.3.1 Occupancy model This pairwise potential value becomes normalized dispersal potential in spatially-explicit metapopulation dynamics model (**ovask2003?**). Done using `MetacommunityDynamics.jl`.

There might be attempts at analytic stuff but maybe not here?

Figure 2: MTE versus epoch fig: shows the chains move toward higher *extinction times over time, i.e. it works.

3

Simulation of data for testing the algorithm

In this section we describe the generation of simulated land cover and sites for testing the algorithm.

3.1. Simulation of landscapes

3.1.1 Generation of landcover maps DiamondSquare with high autocorrelatoin (0.7). Binned into N_{cat} land cover categories.

3.1.2 *Generation of points*

Random Poisson process rounded to be integer coordinates. Padding around the edges because real data doesn't have points on the edges, 10% on each side.

3.1.3 *Resistance values assigned to each land cover type*

Each simulated land cover has N_{cat} categories. The values of resistance

Some type of performance fig vs. raster size and budget figure

4 Actual data St. Lawrence lowlands

5 Discussion