

Appendix D. Linkage Modeling Algorithms

Our task to map linkages for 16 focal species (some with dozens or hundreds of HCAs apiece) plus four landscape integrity models (each with >300 core areas) meant that delineating HCAs, manually identifying HCA pairs to connect, and mapping linkages one by one could easily take years of biologist and GIS analyst time. Moreover, any changes in underlying resistance or HCA layers would require re-starting the linkage modeling process. This meant that automating HCA delineation and linkage modeling would be essential to producing timely products. While existing tools (e.g., Corridor Designer, Funconn, and Circuitscape) provided valuable foundations, none met all of our needs for identifying core area pairs to connect and modeling linkages. We therefore created a set of Python scripts to accomplish this in an efficient and repeatable manner, with an eye toward releasing the scripts for future use in other regions.

The first Python script (A. Shirk, unpublished data) delineated HCAs for most species and core areas for landscape integrity modeling following steps described in section 2.5.1. Once HCA and resistance maps were available (e.g., for Sharp-tailed Grouse, Fig. D.1), a second set of scripts

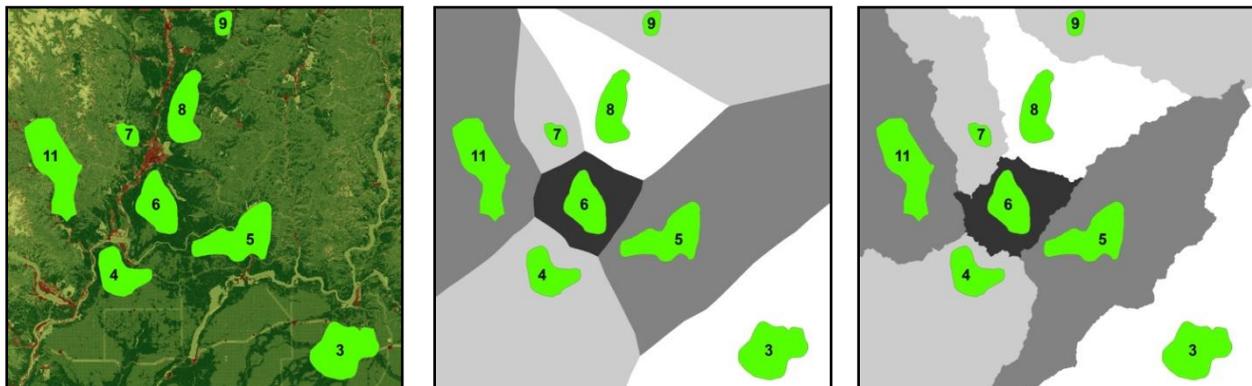


Figure D.1. HCAs, resistance surface, and allocation grids for Sharp-tailed Grouse. Left: HCAs and resistance surface. Center: Euclidean allocation zones. Right: cost-weighted distance allocation zones. In this example, HCA 11 and HCA 8 are adjacent in cost-weighted distance space, but not in Euclidean space.

(B. McRae and B. Cosentino, unpublished data) identified and mapped linkages. These scripts performed the following steps using ArcGIS and Numpy (numerical Python) functions:

- 1) Identify adjacent HCAs, i.e. all HCA pairs that are natural neighbors. These are defined as pairs that have touching allocation zone boundaries (generated using the ArcGIS Cost Allocation and Euclidean Allocation functions). A grid cell is ‘allocated’ to the nearest HCA in either Euclidean or cost-weighted distance space (Panels B and C in Fig. D.1, respectively). If a pathway from one HCA to another must pass through the allocation zone of a third, then the two HCAs are considered nonadjacent. We considered HCA pairs to be candidates for linkages if they were adjacent in either Euclidean or cost-weighted distance space.
- 2) Build a network of HCAs that are adjacent and within a maximum Euclidean distance (if specified) from one another. Euclidean distances between HCAs were derived from the

Conefor Sensinode tool developed by Jeff Jenness, and were calculated as edge-to-edge distances between HCA polygons. Adjacent HCA pairs, and distances between them, were recorded in a table to be used and modified by the scripts. The scripts also created a “stick map,” an ESRI shapefile indicating which HCA pairs were candidates for corridor mapping (Fig. D.2). Stick maps were updated at each subsequent step, and were used to evaluate effects of parameter choices such as maximum corridor lengths.

- 3) Calculate cost-weighted distances from each HCA polygon edge. The scripts minimized processing time by clipping resistance grids using bounding circles around “source” and “target” HCA pairs (Fig. D.3). For each source HCA, the scripts computed bounding circles

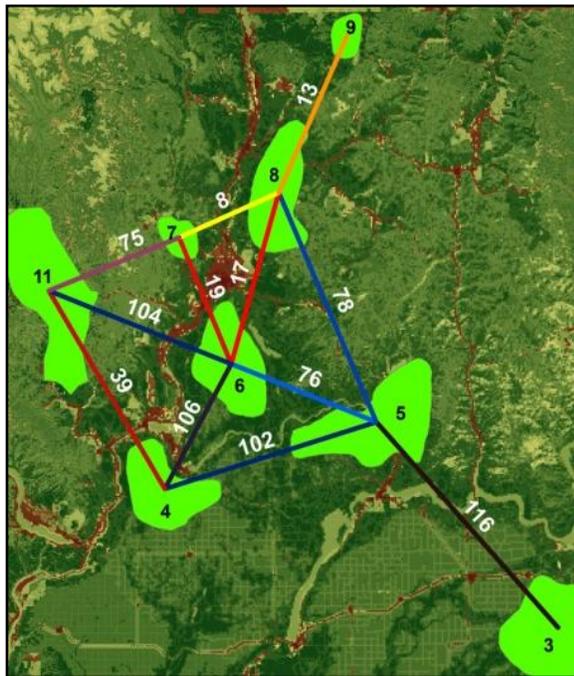


Figure D.2. “Stick Map” for Sharp-tailed Grouse. Sticks show potential links between adjacent HCAs, and include information on each. In this example, labels and colors correspond to Euclidean distances (km), with warmer colors indicating closer pairs.

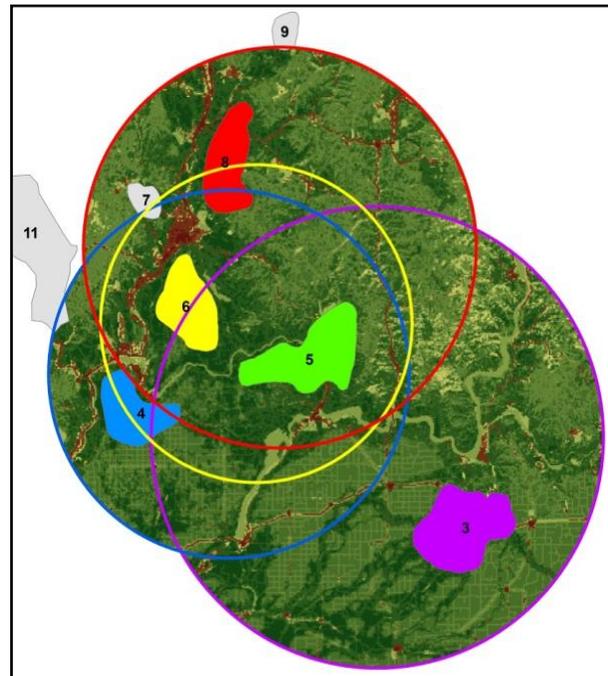
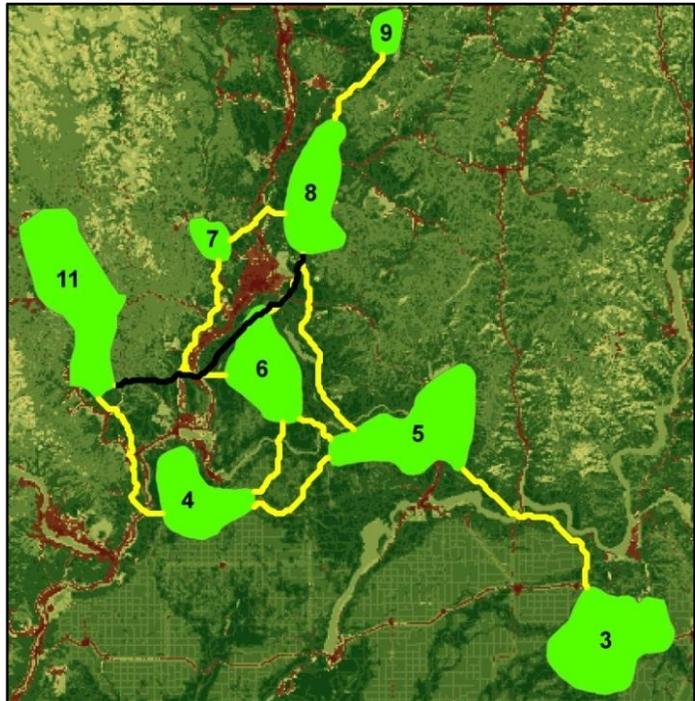


Figure D.3. Bounding circles calculated for HCA 5. Circle colors correspond to respective ‘target’ HCA colors. For cost-weighted distance calculations from any HCA, the algorithm extracted the portion of the resistance layer (background) that fell within the union of all circles containing that HCA.

encompassing the source and each target HCA to which the source was to be connected. The resistance layer was then clipped by the union of all bounding circles for the source HCA plus a buffer large enough to allow corridors sufficient room to “roam” (at least 10 km, but up to $\frac{3}{4}$ the maximum cost-weighted distance specified for corridors for each focal species). This limited cost-weighted distance calculations from each source HCA to include only the portion of the landscape likely to be relevant to connectivity between the source and target HCAs. As each cost-weighted distance surface was created, the scripts extracted minimum cost-weighted distances (representing least-cost path distances) between source and target HCA pairs. Links longer than a user-specified maximum cost-weighted distance were discarded. The scripts also calculated the least-cost path (the single-grid cell-wide route with

the lowest cumulative resistance) between HCA pairs using the ArcGIS Cost Path function (Fig. D.4). If the least-cost path for a link intersected an intermediate HCA, the link was discarded. We assume that if the lowest-resistance portion of the corridor connecting HCA A and C passes through an intermediate HCA B, then the corridor from A to B and the corridor from B to C will provide the needed connectivity from A to C.

- 4) Calculate normalized least-cost corridors between pairs for remaining links by adding cost-weighted distances from respective HCAs and subtracting the minimum least-cost path distance (the cumulative cost of moving along the ideal path between HCA pairs). This step maps all corridors in the same “currency;” grid cells in each normalized corridor raster range in value from 0 (the best or least-cost path) on up. Grid cell values are still in cost distance units, and reflect how much more costly the (locally optimal) path between the HCAs passing through each cell is relative to the (globally optimal) least-cost path connecting the HCA pair.



- 5) Create a composite corridor map in which each cell is the minimum value of all individual normalized corridor layers using the ArcGIS Mosaic function (Fig D.5).

Figure D.4. Least-cost paths between HCAs. Links whose least-cost paths intersected an intermediate HCA (as does the path shown in black connecting HCAs 8 and 11) were discarded. Least-cost path shapefiles were retained and could be queried for the same information stored in stick maps.

The scripts allowed us to rapidly map corridors using a transparent and repeatable process. A distinct advantage of the scripts is that they were relatively simple to use and fast to run (on a landscape with >44 million cells, processing time for steps 2–6 ranged from approximately 1 hour for small numbers of core areas up to 2 days for analyses with 1427 core areas). The efficiency gained through automation allowed our modeling team to work interactively with focal species experts to run analyses, obtain feedback, and re-run analyses with modified parameter values.

We will release the scripts described above on the WHCWG website following publication of this report. The Conefor Sensinode distance measuring tool is available at <http://www.conefor.org/> and http://jennessent.com/arcgis/conefor_inputs.htm.

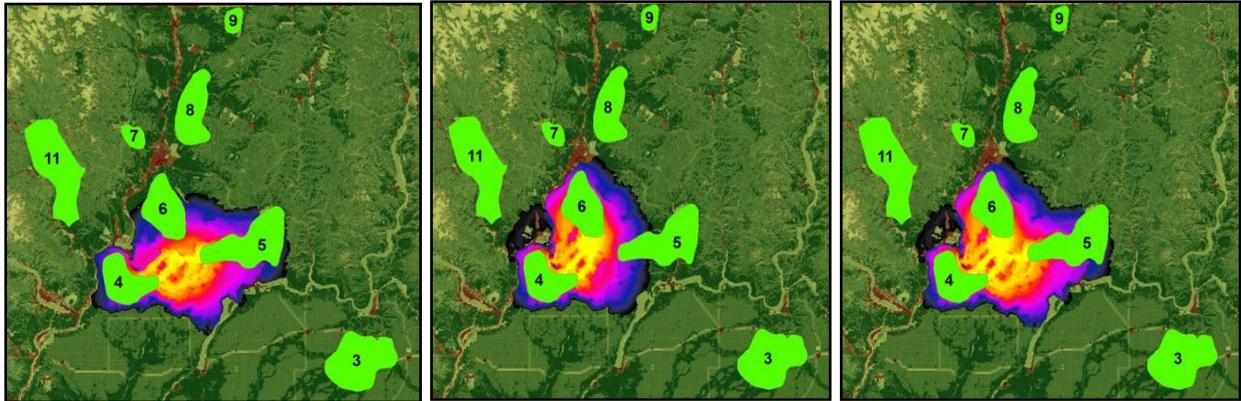


Figure D.5. Compositing process for normalized least-cost corridors. Panels show just two corridors as an example. Left and center panels show normalized least-cost corridors connecting HCA 4 to HCAs 5 and 6, respectively. Warmer colors show lower values, i.e. routes with lower cumulative resistances, for each pair. The ArcGIS Mosaic function took the minimum value of all normalized corridors to create a composite map (right panel).