Target-based site prioritization under climate change

Steven J Phillips
January 2017

Abstract

Network-flow methods can be used to model the survival and dispersal of species along "dispersal chains" while their suitable environmental conditions shift across a landscape in response to climate change. This note describes how this network-flow framework can be combined with the notion of target-based site prioritization to derive an index of how important each site is to ensuring a minimum number of non-overlapping dispersal chains per species.

Introduction and Statement of the Problem

A species' suitable environmental conditions shift geographically under climate change, so its survival over time may depend on repeated dispersal events from areas that are becoming unsuitable into areas that will become suitable. This process can be modeled using dispersal chains (Williams et al. 2005), which consist of a sequence of steps interleaving persistence inside a time slice and dispersal to nearby sites between time slices. Collections of dispersal chains can be represented as network flows in a suitably-defined network (Phillips et al. 2008). An example is shown in Figure 1 (similar to Figure 4 of Phillips et al. 2008), representing a landscape with six sites (labeled a through f) and two time slices ("Time 1" and "Time 2"). Edges inside a time slice (thick edges in the figure) indicate that the corresponding site is suitable in that time slice, while edges between time slices connect sites that are nearby and therefore dispersal between them is likely. In the example graph, the species can survive in site b dduringuring time 1 and then disperse to a or c or remain in b. Alternatively, it can survive in sites d, e or f in time 1, then disperse to (or remain in) site e for time 2. Edges all have unit capacity, i.e., they are limited to at most one unit of flow. A dispersal chain is just a path from the source node (S) to the sink node (T), and any integer-valued flow from S to T can be decomposed into a non-overlapping collection of dispersal chains.

Network flow has been combined with integer optimization (Phillips et al. 2008) to find a minimum number of cells whose protection would ensure that a collection of species would all have some minimumum number of dispersal chains. The optimum solution represents only one way to conserve the collection of species, however, and gives little information about alternative solutions. Furthermore, the optimization process can be time-consuming, and therefore hard to use in a context where sites are protected incrementally and/or opportunistically, so that the conservation planning requirements are continually evolving.

In this note, rather than trying to optimize a conservation plan, we seek to do site prioritization in the context of dispersal chains and network flow. Specifically, we want to define an index of how important a given site is to ensuring that a given species has at least

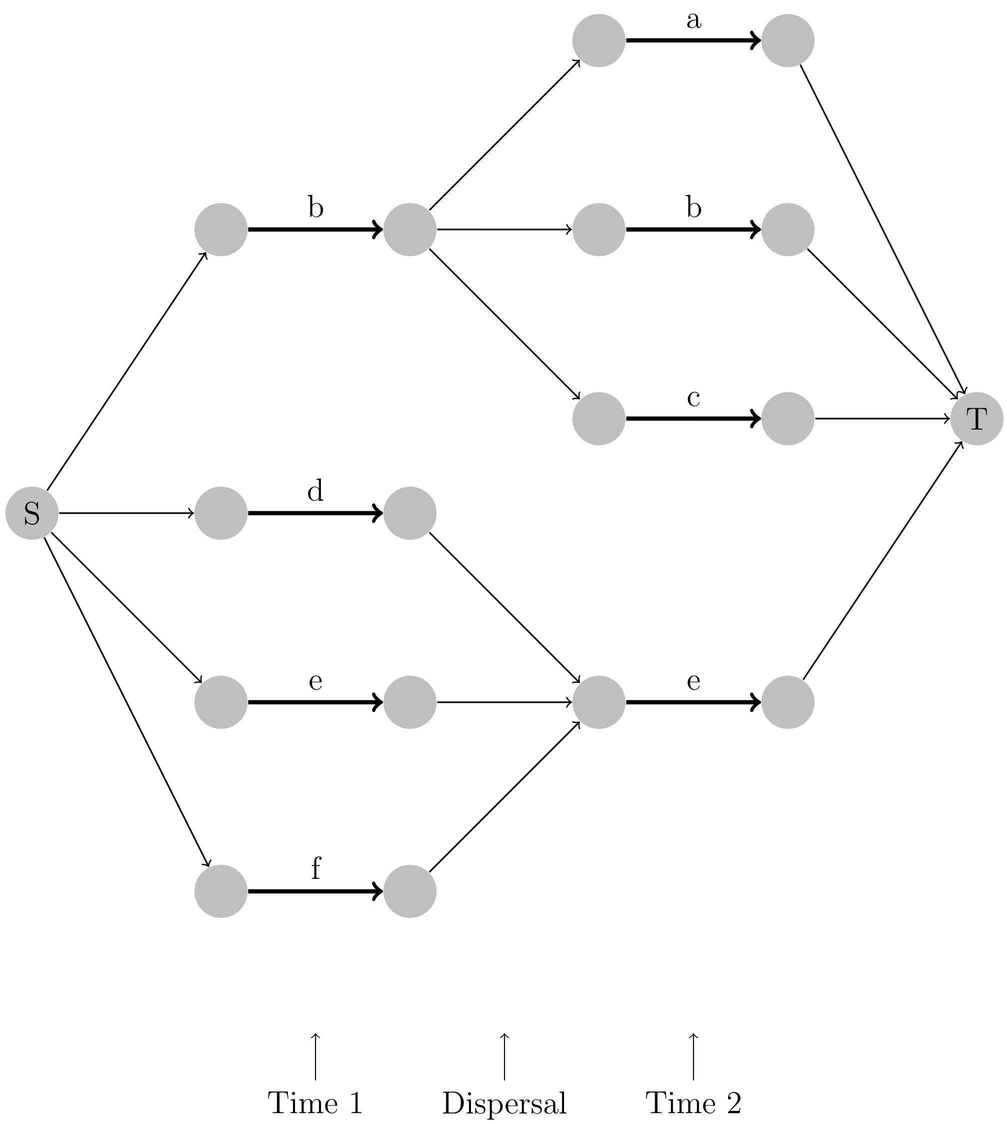

Figure 1: Graph representing persistence and dispersal possibilities for a species under climate change. Two time slices are shown (labeled "Time 1" and "Time 2") for six sites (labeled a through f). A thick edge is present for a site in time slices in which the site would support persistence of the species under climate conditions predicted for that time slice (e.g., site a in time slice 2, and site b in time slices 1 and 2). Edges between time slices represent dispersal possibilities. A path from the source (S) to the sink (T) is termed a "dispersal chain" and represents a way for the species to persist and disperse as necessary in order to survive predicted climate-change induced habitat shifts.

a minimum “target” amount of flow (i.e., number of non-overlapping dispersal chains). This can be seen as a generalization of the notion of target-based site prioritization (Phillips et al. 2010) to the climate-change context. We would like the index to satisfy some basic requirements:

1. It should equal zero if the site cannot be used in any flow that is feasible (i.e., the total flow is at least the target).
2. It should equal one if the site is essential in order to achieve the target number of non-overlapping dispersal chains, i.e., if every feasible flow uses the site.
3. It should be smoothly graduated between these two extremes.
4. It should be reasonably fast to compute, so that it can be used in incremental or collaborative conservation planning, for example by embedding it in a decision-support system.

The index can be summed across species to achieve a composite index of the importance of a site. Species that are most at risk (i.e., with few possible dispersal chains) will contribute strongly to the composite index at some sites, while species with many dispersal chains will only contribute a small amount at any site.

Proposed Solution

We propose to add a quadratic cost function $c(f) = f^2$ to the within-time-slice edges (thick edges in Figure 1), and compute a minimum-cost flow of size equal to the target. The prioritization index $I(s)$ at a site s is defined as the maximum across time slices t of the minimum-cost flow through site s in time t. The quadratic cost function (or any strictly convex increasing cost function) serves to split flow evenly across alternative paths. Therefore, the resulting flow across an edge is high only if there are few alternatives to using that edge in order to achieve the target flow.

In the example of Figure 1, if the target flow is 2, then $I(b) = I(e) = 1$, and I assigns 1/3 to all other sites. If we reduce the target flow to 1, the effect is to halve all the values of I.

Things get more interesting if we break the symmetry by removing site f from Figure 1. In this case, if the target flow is 2, then we still have $I(b) = I(e) = 1$ and $I(a) = I(c) = 1/3$, but now $I(d)$ increases to 1/2. If the target flow is 1, then some calculations show that $I(e)$ drops to 0.471 and $I(b)$ rises to 0.529 (compared to 1/2 for the graph of Figure 1), reflecting the fact that with site f missing, there are more ways to use b than e to make a feasible flow.

Computing the Proposed Index

The proposed index is most easily computed using generic optimization software – both CPLEX and GUROBI most likely have fast implementations of convex quadratic programs with linear constraints. However, if the index is to be used as part of a decision-support tool and computation time must be minimized, then fast algorithms exist specifically for minimum-cost flow with quadratic (or other convex) weights (Minoux 1984; Minoux 1986; Ahuja et al. 1993). Minoux's algorithm is not complicated, and could be implemented in the context of a library of graph algorithms such as the Boost Graph Library (`www.boost.org/libs/graph/doc`).

Limitations of and Variations on the Proposed Index

While the proposed index satisfies requirements 1, 3 and 4 above, the fact that each site appears in multiple time slices makes the index violate requirement 2 in some circumstances. Consider the situation described in Figure 1 of Phillips et al. 2010, where there are three non-overlapping chains using use the same site in different time slices. If the target flow is 2, then the min-cost flow is 2/3 along each of the three chains, so the index for all cells is at most 2/3. On the other hand, cells (2,4) and (2,5) are used (at different times) by all three chains, and are required for any feasible flow, so by requirement 2, we would like their index to be 1. This problem could be alleviated by making the index for a cell be the *sum* of flows in that cell across time slices, rather than the *maximum*, but doing so would make the index value sometimes exceed 1.

In the list of requirements above, we would prefer to have conditions 1 and 2 be necessary and not just sufficient. In other words, we would prefer for an index value of zero to imply that the site cannot be used in any feasible flow, and for an index value of 1 to imply that the site is needed for all feasible flows. However, we can construct special cases where the conditions are not necessary (details not shown here). A more complicated algorithm suggested by Aaron Archer makes condition 2 necessary (i.e. if $I(s) = 1$, then the site s is used by all feasible flows), but not condition 1. However, the extra computation time of the Archer algorithm does not seem warranted.

As described, the minimum-cost flow does not differentiate between a species remaining in the same site for multiple time slices versus needing to disperse. In contrast, we may want to consider a site to be higher priority if it allows the species to persist without needing to disperse. This could be achieved in multiple ways. First, edge costs could be used on dispersal arcs that correspond to the species truly dispersing rather than staying put, in order to encourage the minimum cost flow to avoid dispersal. Alternatively, multi-time-step arcs could be added (similar to those described in the appendix of Phillips et al. 2010) that charge the species only once (rather than multiple times) for a multi-time-step stay in a site, so that sites remain suitable across multiple time steps receive more flow and therefore higher index values.

References

Ahuja, R. K. et al. (1993). *Network Flows: Theory, Algorithms, and Applications.* Upper Saddle River, NJ, USA: Prentice-Hall, Inc.

Minoux, M. (1984). A polynomial algorithm for minimum quadratic cost flow problems. *European Journal of Operational Research 18*(3), 377–387.

Minoux, M. (1986). Solving integer minimum cost flows with separable convex cost objective polynomially. *Mathematical Programming Study 26*, 237–239.

Phillips, S. J. et al. (2010). Voting power and target-based site prioritization. *Biological Conservation 143*(9), 1989–1997.

Phillips, S. J. et al. (2008). Optimizing dispersal corridors for the cape proteaceae using network flow. *Ecological Applications 18*(5), 1200–1211.

Williams, P. et al. (2005). Planning for climate change: Identifying minimum-dispersal corridors for the Cape Proteaceae. *Conservation Biology 19*(4), 1063–1074.