Appendix C from A. Moilanen, “Reserve Selection Using Nonlinear Species Distribution Models”

(Am. Nat., vol. 165, no. 6, p. 695)

Investigating Algorithm Performance Using Artificially Enlarged Problems

A special method was used for verifying algorithm performance with large landscapes. It is hard to verify the performance of stochastic optimization on large problems because the optimal result is not known in advance and excessive computation times may prevent sufficient replication of the optimization. Therefore, the algorithm was tested using artificially enlarged problems, which were constructed by replicating the data files to constitute multiples of the original landscape. Optimization over a replicated landscape is just as difficult as optimization over any other large landscape (the algorithm does not know that the landscape is an artificial multiple of a smaller region), but there is one significant advantage: the minimum suboptimality level for the replicated landscape is known from the original single-landscape optimization.

The idea is that the optimal solution for the replicated landscape can be expected to be approximately a multiple of the solution of the original problem, and if algorithm performance is good, the level of suboptimality of the best solution found should not increase when going to higher levels of landscape replication. Importantly, use of artificially enlarged problems makes it possible to investigate the scaling of algorithm performance as the problem size increases—many algorithms are suitable for the analysis of small data sets but extend poorly to large problems. This is of high relevance for any work that would apply an optimization technique to a large landscape.

Table C1 summarizes the performance of the algorithm on enlarged problems. The sizes of the best solutions for enlarged problems are essentially multiples of the single-landscape solution. A very important aspect of the proposed algorithm is its time scaling. When the landscape is replicated $25 \times$ to 46,000 selection units, the time required to run the optimization (keeping settings equal) is increased by $45 \times$. This shows that at least with this limited set of test problems, the time scaling of the algorithm is even slower than $O(N \log N)$ (for evaluating and local searching a single habitat structure), which indicates that the $O(N \log N)$ connectivity computations using fast Fourier transforms and the local search combine to a search algorithm that has excellent time scaling properties. Using the obvious simple alternative (see app. A) for either of these algorithm components resulted in a time scaling of $O(N^2)$, which precluded the use of the algorithm for large problems (results not shown). Incidentally, the $O(N^2)$ algorithm variant is approximately equally as fast as the proposed algorithm for grids of approximately 1,000 elements, but with 10,000 elements, it already runs much slower than the proposed algorithm with 46,000 elements.

The suboptimality levels for the multiplied problems show a pattern (table C1). First, the $T_j = 0.3$ solutions are apparently less optimal than solutions for $T_j = 0.5$ problems. This is likely to be caused by the smaller solution sizes of the $T_j = 0.3$ problems, which means that when using the nonlinear models, the probabilities of occurrence drop comparatively more in the $T_j = 0.3$ solutions because of larger reductions in connectivity. (The lower bound computations assumed no negative effects of connectivity loss.) Going from the single landscape to a $25 \times$ replicated landscape shows an increase in suboptimality of approximately 4%. One interpretation for this is that the single-landscape results are very close to optimal but that the $25 \times$ landscape solutions truly are 4% suboptimal.

Figure C1 shows a summary of four optimization runs over a landscape replicated to $25 \times$ the original landscape size. Visual comparison to figure 1 indicates that the optimal spatial pattern found for the large replicated landscape is essentially a multiple of the solution for the original landscape. Even when larger problems were run for some species individually, the optimizations converged consistently for a $100 \times$ landscape ($N \approx 200,000$) in about a day of computation (not shown).

Figure C1: Example of an optimization result over a landscape artificially enlarged $25 \times$ ($GA_M = 300$, $GA_G = 30$). The selection converges essentially to a multiple of the optimal result for the original landscape (cf. fig. 1), which indicates that the proposed algorithm can be used successfully even for large landscapes.

Table C1

Solution sizes and timings for artificially increased problem sizes

Size	Cells	B_2	Optimal	Suboptimal (%)	Ratio	$t(s)$ (ratio)
$T_j = 0.5$:						
1 ×	1,853	654	714–716	+9.2	1 ×	3,981 (1 ×)
4 ×	7,412	2,584	2,871–2,875	+11.1	4.02 ×	16,348 (4.11 ×)
9 ×	16,667	5,816	6,490–6,498	+11.6	9.09 ×	50,374 (12.7 ×)
25 ×	46,325	16,160	18,217–18,224	+12.7	25.5 ×	179,819 (45.2 ×)
$T_j = 0.3$:						
1 ×	1,853	358	400–403	+11.7	1 ×	3,458 (1 ×)
4 ×	7,412	1,435	1,617–1,624	+12.7	4.02 ×	15,306 (4.43 ×)
9 ×	16,667	3,228	3,679–3,687	+14.0	9.15 ×	44,164 (12.8 ×)
25 ×	46,325	8,970	10,386–10,415	+15.8	25.8 ×	145,781 (45.2 ×)

Note: Table shows where the landscape grid has been multiplied to include four, nine, or 25 copies of the original landscape (targets 0.5 and 0.3). Optimization is for all seven species simultaneously with GA parameters $GA_G = 30$, $GA_M = 300$, four replicates. Size ratio gives the size of the solution of the enlarged problem divided by the size of the solution for the original problem. Time ratio gives the respective statistic for computation time. B_2 is the lower bound for solution size made using most optimistic possible assumptions concerning effects of habitat loss (app. B); the respective maximum level of suboptimality of the best solution is given.