RESEARCH ARTICLE

How large spatially-explicit optimal reserve design models can we solve now? An exploration of current models' computational efficiency

Yicheng Wang[1], Hayri Önal[2], Qiaoling Fang[3]

1 *College of Resources and Environment, Qingdao Agricultural University, 700 Changcheng Road, Chengyang District, Qingdao 266109, China* **2** *Department of Agricultural and Consumer Economics, University of Illinois at Urbana-Champaign, 1301 West Gregory Drive, Urbana, IL 61801, USA* **3** *College of Management, Ocean University of China, 238 Songling Road, Laoshan District, Qingdao 266100, China*

Corresponding author: *Yicheng Wang* (ywangqau@163.com)

Citation: Wang Y, Önal H, Fang W (2018) How large spatially-explicit optimal reserve design models can we solve now? An exploration of current models' computational efficiency. Nature Conservation 27: 17–34. https://doi.org/10.3897/natureconservation.27.21642

Abstract

Spatially-explicit optimal reserve design models select best sites from a set of candidate sites to assemble nature reserves to protect species (or habitats) and these reserves display certain spatial attributes which are desirable for species. These models are formulated with linear 0–1 programming and solved using standard optimisation software, but they were run on different platforms, resulting in discrepant or even conflicting messages with regard to their computational efficiency. A fair and accurate comparison of the convenience of these models would be important for conservation planners who use these models. In this article, we considered eight models presented in literature and tested their computational efficiency using randomly generated data sets containing up to 2000 sites. We focused on reserve contiguity and compactness which are considered crucial to species persistence. Our results showed that two of these models, namely Williams (2002) and Önal et al. (2016), stand out as the most efficient models. We also found that the relative efficiency of these models depends on the scope of analysis. Specifically, the Williams (2002) model solves more of the test problems when contiguity is the only spatial attribute and a large subset of the candidate sites needs to be selected. When compactness is considered also, the Önal et al. (2016) model generally performs better. Large scale models are found to be difficult to solve in a reasonable period of time. We discussed factors that may affect those models' computational efficiency, including model size, share of selected sites, model structure and input data. These results provide useful insight and guidance to conservation practitioners and researchers who focus on spatial aspects and work with large-scale data sets.

Introduction

Anthropogenic activities have caused tremendous impacts on ecosystems all over the world. These impacts include but are not limited to environmental pollution, habitat loss and fragmentation, invasion of exotic species and climate change (e.g. Aplet and McKinley 2017). These impacts led to an increased rate of extinction of species and impaired the services that ecosystems have been providing to humans. Establishing nature conservation reserves have been adopted across the globe as a direct way to restore ecosystems and protect species. However, resources devoted to this purpose such as land and budget have always been scarce and they often face conflicting demands from other sectors of society such as housing and industry. With the aim of designing ecologically effective and economically efficient nature reserves comes the science of reserve design (Kingsland 2002).

The reserve design problem is defined as selecting a subset from a larger set of candidate sites to assemble nature conservation reserves to provide adequate protection to targeted species or habitats. This problem has been studied using various approaches, including site scoring (e.g. Pressey and Nicholls 1989), gap analysis (e.g. Scott et al. 1993), heuristics (e.g. Pressey et al. 1997) and formal optimisation (e.g. Possingham et al. 2000, Williams et al. 2004). Gap analysis and heuristics have been widely used both in academic research and conservation practices, primarily because they are intuitive, easy to use and computationally convenient. Some popular nature reserve design software such as Marxan and C-plan use heuristics as their major solution algorithm (Ball et al. 2009, Sarkar 2012). However, it has long been demonstrated that these techniques find optimal solutions only occasionally and often generate solutions which significantly deviate from the optimal ones (Önal 2003, Williams et al. 2005). This means that they may not be allocating scarce conservation resources in the best possible manner. The problem can also be formulated as an optimisation model, specifically as a linear mixed integer programme (MIP), to determine the best selection of sites while meeting the conservation goals. The resulting model can be solved using MIP solvers such as CPLEX and GUROBI integrated in off-the-shelf mathematical modelling software such as GAMS and AMPL (AMPL 2016, GAMS 2016). Finding the best possible solution helps guarantee efficient allocation of conservation resources. Due to this advantage, optimal reserve design models have received substantial attention since the 1990s.

The early optimisation formulations of the reserve design problem adapted the set covering and maximal covering frameworks well known in the operations research literature (Cocks and Baird 1989, Underhill 1994, Camm et al. 1996, Church et al.

1996, Ando et al. 1998). These initial formulations neglected spatial attributes of the reserve, usually ending up with a solution where selected sites are highly scattered over the landscape and far apart from each other. A reserve with such a spatial configuration is not very meaningful in practice because both species survival and reserve management benefit from a spatially coherent reserve. The second generation optimum reserve design models incorporated spatial attributes to improve the coherence and functionality of the reserves (Williams et al. 2005). Amongst the spatial attributes discussed in literature, the most important ones are spatial contiguity (or connectivity) and compactness of the selected sites. Spatial contiguity means that any two sites in the reserve can be linked by a path of selected sites. Compactness is a more difficult concept to define, but, in general, it means that the selected sites must be close to each other (or grouped together) to the greatest possible extent. Some studies used the total distance between pairs of selected sites and the boundary length of the reserve as proxy measures for compactness. These two spatial attributes have been successfully modelled using graph theory, network flow theory and mathematical techniques (see Williams et al. 2005, Moilanen et al. 2009, Billionnet 2013 for reviews of those studies). Incorporating these spatial attributes into the optimisation frameworks requires more complex formulations, however, including a large number of additional variables and constraints. Thus, the reserve selection problem gets computationally more difficult when spatial configuration is of concern. For instance, a set covering formulation involving thousands of candidate sites could be solved in seconds (Rodrigues and Gaston 2002), but a compactness model involving about 700 sites could not be solved to optimality within 10 hours (Fischer and Church 2003).

Several modelling approaches have been introduced in the reserve design literature to incorporate spatial aspects in site selection. Due to their differences in structure and size, those models differ in their computational performance. Moreover, individual models were run on different platforms with different CPU speeds and solved using different MIP solvers and, therefore, it is not clear how those models compare to each other in terms of their computational efficiency. A fair and accurate comparison of the convenience of these models would be important for on-the-ground conservation practitioners. This is the main motivation for this article. For this purpose, we selected eight optimisation models incorporating (or could incorporate easily by proper modification) contiguity and compactness and conducted computational experiments with these models by running them on the same computer and using the same data sets that were generated randomly. We aimed to answer two questions: i) how do these models perform in solving contiguity and compactness problems with different sizes and ii) what are the likely factors affecting their computational efficiencies?

We carried out the tests on a Lenovo desktop computer with Intel Pentium (R) CPU of 2.8 gigahertz and RAM of 8.0 gigabyte and the operation system was Windows 64-bit. As the MIP solver, GUROBI 5.0 integrated in GAMS 23.9 was used.

Material and methods

Contiguity models

Table 1 summarises the eight models we tested. The 'Algebraic formula' column of the table presents the basic algebraic formula used to ensure spatial contiguity.

Some of the models were not originally proposed for reserve design purposes or did not consider the cost of site selection. We made necessary modifications to those models so that they are applicable to the reserve design problem we address here. The objective of all the models is stated as the minimisation of the total cost of site selection, namely:

Minimize $\sum_i c_i U_i$ (1)

where U_i is a binary variable which equals 1 if site i is selected and 0 otherwise and c_i is the cost of selecting site i.

Those models, not originally proposed for reserve design problems, are extended by adding the species coverage constraint described below:

$\sum_i e_i U_i \geq p$ (2)

where e_i is the population size of target species in site i; and p is the targeted (minimum viable) population of target species. This constraint is supposed to ensure a long-term survival chance for target species.

The Miller et al. (1960) model is modified as below to control the formation of arcs between selected sites.

$X_{ij} + X_{ji} \leq U_i$ for all $i<j, j \in N_i$ (3)

$X_{ij} + X_{ji} \leq U_j$ for all $i<j, j \in N_i$ (4)

$\sum_{j \in N_i} X_{ij} \leq U_i$ for all i (5)

$\sum_i \sum_{j \in N_i} X_{ij} = \sum_i U_i - 1$ (6)

where X_{ij} is a binary variable which equals 1 if an arc directs from i to neighbour j and 0 otherwise and N_i is the set of i's neighbour sites. For details, see Williams (2002).

We first generated ten hypothetical data sets. Each data set considers a 10*10 grid partition including one hundred sites. The distribution of species over these sites was generated by using a uniform distribution function with the range of [0, 5], which means that the population size (e_i) in each site takes an integer value between 0 and 5. We generated the selection cost (c_i) similarly using the range of real numbers [1, 10]. We assigned a different value to the target population size (p) in each data set, where p is calculated by $p=a*n$ where $n=100$ is the number of candidate sites and a is a popula-

Table 1. Mechanisms used to ensure contiguity and algebraic formulae of the eight models used in the computational efficiency tests.

Model number	Mechanism to ensure contiguity	Algebraic formula	References
1	An arc can direct from node i to neighbour node j only if the value assigned to i is less than the value assigned to j, so an arc directing away from a node will not return to it and therefore loops are prevented.	(1)-(6); $Y_i - Y_j + nX_{ij} \leq n - 1$ $i{\neq}j, j{\in}N_i$	Miller et al. (1960); Duque et al. (2011)'s Tree^{PRM} model.
2	Two trees are formed, one in primal graph, the other in dual graph, the interwoven structure of these two trees prevents the formation of loops.	(1)-(6); (2)-(4) in Williams (2002).	Williams (2002)
3	One selected site serves as a sink, every other selected site provides one unit of supply to the flow and all flows finally reach the sink.	(1) and (2); (1)-(3) in Shirabe (2005).	Shirabe (2005, 2009); Duque et al. (2011)'s Flow^{PRM} model
4	The tail length of a selected site equals the number of arcs linking to it and that length is strictly larger than that of the site's preceding sites.	(1)-(2); (3)-(7) in Önal and Briers (2006).	Önal and Briers (2006)
5	Solve a relaxed problem first with no contiguity constraints. If a cycle occurs, apply an inequality (a cut) to that cycle and solve again until there is no cycle in the solution.	(1)-(6); (8) in Önal and Wang (2008).	Önal and Wang (2008)
6	An amount of flow is injected into the partition from an outside node, each selected site consumes one unit of flow, residual flow is absorbed by a predefined variable.	(1) and (2); (5)-(8) in Conrad et al. (2012).	Conrad et al. (2012)
7	Capital flows from the supply node to trans-shipment nodes and for each selected site, capital can come into it from only one of its adjacent sites.	(1) and (2); (2)-(7) in Jafari and Hearne (2013).	Jafari and Hearne (2013); Jafari et al. (2017)
8	If a site is selected then at least one of its adjacent sites, which is nearer (structurally or functionally) to the centre, must also be selected.	Minimize $\sum_k \sum_i c_i X_{ki}$; (2),(4),(6) and (8) in Önal et al. (2016).	Önal et al. (2016); Wang and Önal (2016)

tion coefficient ranging from 0.25 to 2.5 with an increment of 0.25 (thus, a population between 25 and 250 individuals must be protected in the reserve and a larger a implies a larger reserve area). By designing reserves with different areas, we intend to test the impact of the number of selected sites on computational efficiency.

To eliminate the impact of input data on the test result, for each value of a we generated 50 data sets, each with a different distribution of species and selection cost across the 100 sites and used in a different run. To save time in the test and avoid tying the computer to a long computational time, in each run we imposed a one-hour maxi-

mum processing time limit. The total time limit for all runs was limited to 2 hours (i.e. the programme was terminated if it took more than 2 hours to complete the 50 runs). Average CPU times in completed runs were reported.

Contiguity and compactness models

The Önal et al. (2016) model and the Williams (2002) model were modified and used in the contiguity and compactness test because these two models were found to be the most efficient models in the above contiguity test. We modified the objective function of the Önal et al. (2016) model as below so it incorporated selection cost which is an important factor in reserve design practices:

$$\text{Minimize} \sum_k \sum_i d_{ki} X_{ki} + \sum_k \sum_i c_i X_{ki} \quad (7)$$

where X_{ki} is a binary variable which equals 1 if site i is selected into the reserve which is centred at site k and d_{ki} is the distance between centre site k and selected site i. The first part of (7) is the total distance between the cluster centres and selected sites and the second part is selection cost. These two components can be given different weights to reflect their importance in site selection; for computational purposes, we used the same weights (=1.0).

In the Williams (2002) model, we made the following modifications to incorporate compactness assuming that compactness is measured by total pair-wise distance between selected sites (Önal and Briers 2002).

$$\text{Min} \sum_i \sum_{j>i} d_{ij} W_{ij} + \sum_i c_i U_i \quad (8)$$

s.t.:

$$W_{ij} \geq U_i + U_j - 1 \text{ for all } i<j \quad (9)$$

where W_{ij} is a binary variable which equals 1 when sites i and j are both selected and 0 otherwise. The first part of (8) represents reserve compactness while the second part is the cost of site selection. Constraint (9) relates W_{ij} to the site selection variables.

Results

Contiguity: the 100-site problem

The computational results of the 100-site contiguity problem are summarised in Table 2 (for clarity reasons, we only report results with populations (p) of 50 to 250 with an increment of 50). Except Williams (2002) and Önal et al. (2016), all models encountered

Table 2. Computational efficiency of alternative contiguity models in solving a 100-site problem.

Models	Population size to be covered in the reserve				
	50	**100**	**150**	**200**	**250**
Miller et al. (1960)	136.9	>3600.0 [a]	>3600.0 [a]	>3600.0 [a]	>2400.0 [b]
Williams (2002)	9.0	50.8	20.2	3.0	0.1[c]
Shirabe (2005)	64.2	>969.1[a]	>477.1[a]	47.0	0.1[c]
Önal and Briers (2006)	>2024.3 [a]	>3600.0 [a]	>3600.0 [a]	>3600.0 [a]	>2400.6 [b]
Önal and Wang (2008)	>505.6 [a]	>944.4 [a]	>824.2 [a]	>2405.3 [a]	>1224.9 [b]
Conrad et al. (2012)	>1854.8 [a]	>2551.1 [a]	>2026.5 [a]	>123.5 [a]	0.2 [c]
Jafari and Hearne (2013)	>1905.5 [a]	>3600.0 [a]	>3600.0 [a]	>311.8 [a]	0.9 [c]
Önal et al. (2016)	12.0	5.3	3.3	4.2	6.6 [c]

a: 2 to 50 runs were completed within the total time limit (2 hours). These values are averages of the CPU times used in completed runs.

b: 3 to 11 runs were completed within the total time limit (2 hours), amongst which 1 to 5 runs were infeasible due to high coverage requirement and/or inadequate species occurrence. The reported values are averages of the CPU times used in feasible runs. The ">" indicates that at least one run could not be completed within 1 hour.

c: 22 runs were infeasible. These values are averages of the other feasible 28 runs.

computational difficulty when solving the problem. That is, at least in one run, they could not find the optimal solution within an hour. The Miller et al. (1960) model completed only two runs in 2 hours in most population settings. The Shirabe (2005) model encountered difficulty in two population settings (p=100 and 150), in which most other models also encountered computational difficulty. These results are consistent with the findings of Duque et al. (2011) who applied these methods in a districting problem and found that the model adapted from Shirabe (2005) was fastest while their own models could not solve a 49-site districting problem in 3 hours.

The models of Önal and Briers (2006) and Jafari and Hearne (2013) completed only two runs in 2 hours in, respectively, three and two population settings (for Önal and Briers (2006) model, these settings are p=100, 150 and 200 and for Jafari and Hearne (2013) model, the settings are p=100 and 150). The models of Önal and Wang (2008) and Conrad et al. (2012) are better in the sense that they completed more runs in most population settings.

The Williams (2002) and Önal et al. (2016) models stand out as the fastest amongst the eight models and completed all 50 runs within the allowed time limits (each run was completed in less than one minute of processing time). The Williams (2002) model took 0.1 second to about 50 seconds to solve those problems, while the model of Önal et al. (2016) seems a little faster, in most settings using less than 10 seconds except one case where it used an average of 12.0 seconds.

All models, except the Önal et al. (2016) model, solved the problem relatively easily when a very large or very small population was to be covered (equivalently, a very large or very small number of sites need to be selected). A similar result was

reported by Williams (2002) also. The Önal et al. (2016) model was computation-ally steady regardless of the population size specification (the run times were in the range of 3.3–12.0 seconds).

Contiguity: larger size problems

Having observed that, in the 100-site test problems, the Williams (2002) and Önal et al. (2016) models are substantially more efficient than the other models, in the remaining test runs we compared these two models against each other in solving larger problems that involved 200 to 2000 sites. We relaxed the optimality requirement by setting the relative gap (percentage difference between the incumbent solution and the best possible solution) at 1 percent level (instead of 0 percent, no gap) to avoid long processing times that might possibly be needed to solve large problems to exact optimality (this is done in GAMS by writing a simple code *option optcr=0.01*). Table 3 reports the results for different model sizes.

The Williams (2002) model solved the problem for almost all sizes when the population coefficient a was high (2.5 and 2.0), although the processing times increased as the number of candidate sites (n) was increased. For instance, when n=200 and a=2.5, the model completed all feasible runs (in some cases the model was infeasible due to high population requirement and/or inadequate population distribution in the candidate sites) with an average processing time of 0.2 second. For n=2000, the average processing time increased to 4.2 seconds. A similar pattern is observed when the population coefficient was decreased to a=2.0. For n=200, an average of 4.0 seconds was used, while for n= 2000, only 14 runs were completed within the 2 hour total time limit. For other population settings, the model encountered computational difficulty more often and could not find a solution after running 1 hour in at least one run (indicated in Table 3 with ">"). The computational difficulty has become more severe as the problem size increased. For the setting a=1.5 and n=200, for example, the model completed 14 runs (average processing time =552.3 seconds). For a=1.5 and n=500, the model completed only 4 runs (average processing time = 2124.9 seconds). The model completed only 2 runs when n=1000, 1500 and 2000. The worst performance was observed when a=1.0 and 0.5, where the model ran out of the 1 hour time limit almost in each run even with the smallest test problem (n=200).

The Önal et al. (2016) model displayed a relatively steady computational performance as in the 100-site problem. In general, the model used less time when a smaller number of sites was selected (Table 3, n=200). However, it encountered more difficulties than the Williams (2002) model in solving large problems even when the population settings were high. For instance, when n=500, the model completed only two runs for each population setting; and when n=1000, 1500 and 2000, it stopped without finding any solution within the 1 hour time limit. A more serious problem was encountered in some cases where the computer ran out of memory due to the large size of the model (millions of variables and equations, see Table 3).

Table 3. Computational efficiency of Williams (2002) and Önal et al. (2016) models in solving large-scale contiguity problems.

Models	$α_a$	n=200 (10*20)		n=500 (20*25)		n=1000 (25*40)		n=1500 (30*50)		n=2000 (40*50)	
		Model size [b]	Time (s)	Model size [b]	Time (s)	Model size [b]	Time (s)	Model size [b]	Time (s)	Model size [b]	Time (s)
Williams (2002)	2.5	2,361 1,682	0.2 [c]	6,141 4,322	0.5 [c]	12,481 8,742	1.1 [c]	18,831 13,142	2.4 [c]	25,244 17,594	4.2 [c]
	2.0		4.0		26.5		108.3		383.5 [d]		e
	1.5		>552.3 [d]		>2124.9 [d]		>3600.0 [d]		>3600.0 [d]		e
	1.0		>3600.0 [d]		>3600.0 [d]		>3600.0 [d]		>3600.0 [d]		e
	0.5		>1822.0 [d]		>3600.0 [d]		e		e		e
Önal et al. (2016)	2.5	40,001 39,263	>3278.5 [d]	250,001 248,093	>3600.0 [d]	1,000,001 996,133	e	2,250,001 2,244,163	e	4,000,001 3,992,183	e
	2.0		2851.4 [d]		>3600.0 [d]		>3600.0 [d]		e		e
	1.5		2608.3 [d]		>3600.0 [d]		>3600.0 [d]		e		e
	1.0		>2613.6 [d]		>3600.0 [d]		>3600.0 [d]		f		g
	0.5		1238.5 [d]		>3600.0 [d]		>3600.0 [d]		f		g

a: Population coefficient. The size of population to be covered in the reserve equals to *a* times the number of candidate sites (*n*).

b: The first number is the number of variables, the second number is the number of equations in the model.

c: 17 to 29 runs are infeasible due to high coverage requirement and/or inadequate species occurrence. Values are average of times used in feasible runs.

d: 2 to 19 runs were completed in 2 hours. The ">" indicates at least one run was out of time limit (1 hour).

e: 2 or more runs were completed, at least one of which terminated with no solution after running 1 hour or with integer solution but gap not available.

f: Only 1 run was completed which took more than 7 hours and found integer solution with gap over 80 percent or no solution returned.

g: Out of memory after running about 2000 seconds or integer solution found after running nearly 3 hours but with gap greater than 70 percent.

Contiguity and compactness

In this section, we compare the computational efficiency of the Williams (2002) and Önal et al. (2016) models in solving contiguity and compactness problems. We called the model based on Williams (2002) but with (8) as the objective function and (9) as an additional constraint "W-CM" (CM for contiguity and compactness) and the model based on Önal et al. (2016) but with (7) as the objective function "Ö-CM". The number of sites in the test runs was set as *n*=100, 200 and 500 and we defined the population coefficient *a* as in the contiguity tests. When *n*=500, both models encountered severe computational difficulty for most population coefficient settings. Table 4 summarises the results.

Table 4. Comparison of W-CM model and Ö-CM model in solving contiguity and compactness problems with various sizes.

Models	α[a]	n=100 (10*10)		n=200 (10*20)		n=500 (20*25)	
		Model size [a]	Time (s)	Model size [a]	Time (s)	Model size [a]	Time (s)
W-CM	2.5	6,091; 5,772	1.4[b]	22,261; 21,582	4.8 [b]	130,891; 129,072	78.3[b]
	2.0		178.2[c]		>2401.6 [d]		>3600.0 [c]
	1.5		1692.3[c]		>3600.0 [d]		>3600.0 [c]
	1.0		>3318.8[c]		>3600.0 [d]		>3600.0 [c]
	0.5		90.6		>3600.0 [d]		>3600.0 [c]
Ö-CM	2.5	10,001; 19,443	2.4 [b]	40,001; 39,263	1474.4 [b]	250,001; 248,093	>3600.0 [c]
	2.0		2.0		1360.7 [c]		>3600.0 [c]
	1.5		1.9		1135.4 [c]		>3600.0 [c]
	1.0		1.9		1073.4 [c]		>3600.0 [c]
	0.5		1.7		767.6 [c]		>3600.0 [c]

a: See Table 3 for the explanations of *a*, model size and symbol ">".
b: Some runs were infeasible due to high population requirement and/or inadequate species occurrence. Values are averages of times used in feasible runs.
c: 2 to 41 runs were completed in 2 hours. Values are averages of times used in completed runs.

The W-CM model encountered difficulty for some population settings (*a*) when solving the 100-site problem. For *a*=2.5, the solution time was only 1.4 second, but for the settings *a*=2.0 and 1.5, it jumped to 178.2 and 1692.3 seconds. For *a*=1.0, only 3 runs were completed within the 2 hour total time limit. However, for *a*=0.5, the solution time was decreased dramatically to 90.6 seconds, showing the fast-slow-fast pattern observed when solving the contiguity problems. For *n*=200 and 500, the model completed only 2 runs within the 2 hour total time limit for most population settings. In general, larger models require more solution times. However, raising the population requirement level improved the computational efficiency. For example, for *a*=2.5, the average solution times were only 4.8 and 78.3 seconds for *n*=200 and *n*=500, respectively.

The Ö-CM model used only about 2 seconds in the 100-site problem for all population coefficient settings. When *n*=200, the model used about 20 minutes or less for all population settings, again better than the W-CM model. When *n* was increased to 500, this model encountered difficulty too, where only 2 runs could be completed within the 2 hour total time limit.

It is interesting that the Ö-CM model is computationally more efficient than the W-CM model although it has a larger size both in terms of the number of variables and

the number of equations (see Table 4). This indicates that model size may not be the only factor determining the computational efficiency of MIP models.

Finally, we note that the Ö-CM model (which includes an additional distance component

$$\sum_k \sum_i d_{ki} X_{ki}$$

in the objective function, see (7)) is computationally more efficient than the Önal et al. (2016) model used in this test, although the size of these two models are exactly the same. For instance, for a=1.5 and n=200, the Önal et al. (2016) model used an average of 2608.3 seconds of processing time (see Table 3), while the Ö-CM model used only 1135.4 seconds (see Table 4). This, again, indicates that the model size is not the only factor that may affect computational efficiency of MIP models. We discuss these factors in the next section.

Discussion

Based on the computational experiments we conducted, we identified four important factors that affect the computational efficiency of MIP models in nature reserve design problems with contiguity and compactness considerations: 1) Model size, 2) The share of selected sites, 3) Model structure and 4) Input data. We discuss these factors below.

Model size is the most important factor affecting the computational efficiency of these models. As the model size (especially the number of binary variables) increases, solution time may increase substantially. Figure 1 depicts an example using the Williams (2002) model. As the number of candidate sites n increased from 100 to 1500, which gradually increased the number of variables from 6,091 to 18,831 and the number of constraints from 5,772 to 13,142, the solution time increased exponentially from 3 seconds to nearly 400 seconds. This is largely due to the number of nodes solved during the branch and bound procedure used by most MIP solvers. In general more nodes have to be solved when the model has a larger size.

Our test runs showed that the solution times also depend significantly on the share of selected sites in the optimal solution, which in turn depends on the specification of the population coefficient (a). This can be seen clearly, for instance, in the results obtained with the Shirabe (2005), Conrad et al. (2012) and Jafari and Hearne (2013) models for n=100 (see Table 2) and in the Williams (2002) model results for n=200 (see Table 3). Generally, these models follow a fast-slow-fast pattern, that is, as the share of selected sites increases, the solution time increases first, reaches a peak (where the share of selected sites is about 20–30 percent of all the candidate sites) and then decreases. This may be explained by the number of binary variables that have to be fixed at the values of 0 or 1 during the solution process. When the share of selected sites is small (or large), a large portion of binary variables will be fixed at 0 (or 1), leaving fewer combinations (a smaller branch and bound tree) that need to be searched for improving the
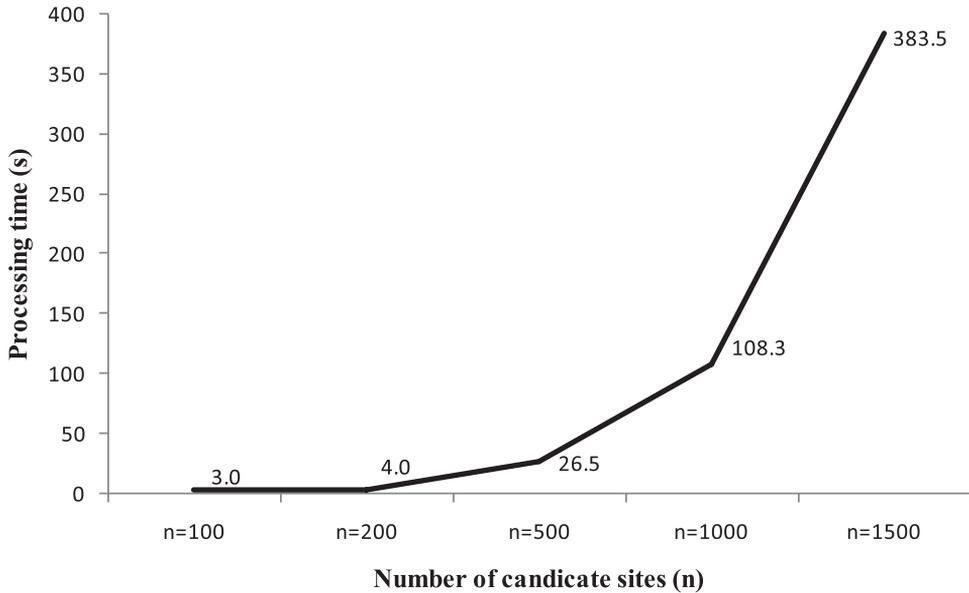
Figure 1. Relationship between processing time and the number of candidate sites (Williams (2002) model was used. population coefficient *a*=2.0. See text for detailed explanation of *a*).

objective function value (A brief description of the branch and bound algorithm can be found in Önal 2003). Similar results have been reported earlier by Williams (2002).

The Önal et al. (2016) model seems to be an exception. As noted above, it showed a relatively steady pattern in solution times for all population coefficient settings. The solution time tends to decrease only when a very small number of sites are to be selected.

Model structure is the third important factor that affects computational performance of the models we tested. Although the number of variables and constraints in a model may be substantial, due to the algebraic model structure, a large number of those variables and constraints may actually be redundant. Those redundant elements are eliminated during the preliminary solution procedure (Presolve in some GAMS solvers such as CPLEX and GUROBI) and the model sizes are much smaller after the Presolve. This is one of the main reasons for the superior computational advantage of the Önal et al. (2016) model. This situation was also observed in empirical applications of the Önal et al. (2016) model.

In addition, some algebraic forms may be integer friendly and some may not. For example, the graph-theoretic contiguity model introduced by Önal and Briers (2006) has the following constraint:

$$\sum_{i \in N_j} X_{ij} \leq 4\, U_j \text{ for all } j \ (10)$$

where X_{ij} and U_j are binary variables as defined before and N_j is the set of *j*'s neighbour sites. This constraint states that if site *j* is selected, then at most four arcs can
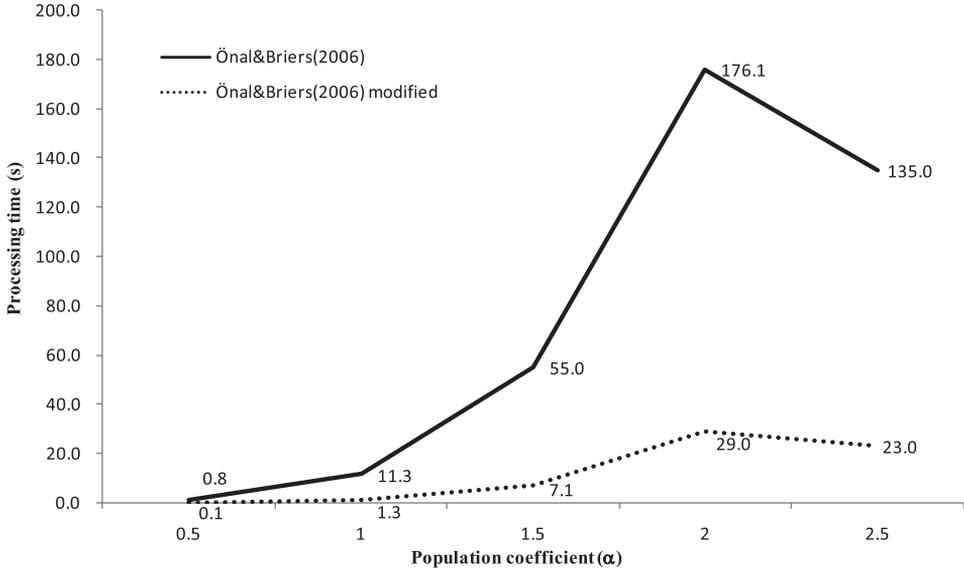
Figure 2. Significance of model structure in affecting the model's computational efficiency (number of sites *n*=36. The size of population to be covered equals to*a*times *n*.).

direct to it from its neighbour sites. The computational efficiency of the model can be improved substantially by modifying this constraint as in constraints (3) and (4). A comparison of the original Önal and Briers (2006) model and the modified model is depicted in Figure 2.

It is clear that replacing constraint (10) with constraints (3) and (4) slashes the solution time. We performed a paired samples t-test for each value of *a* to test the significance of the difference between these two models' processing times. The result reported a *p* value less than 0.05 for each *a* value, indicating that the difference between the processing times is statistically significant. This result should be attributed to the fact that a constraint like (10) is integer unfriendly while the constraints (3) and (4) improve the model's computational efficiency by helping construct a unimodular constraint matrix (ReVelle 1993, Williams 2002). Indeed, having a unimodular constraint matrix is one of the main reasons for the Williams (2002) model's good efficiency.

Input data such as land price and species distribution across candidate sites are also crucial factors that can affect the model's computational efficiency. Figure 3 depicts the processing times used by the models of Önal et al. (2016) and Williams (2002) when solving a 100-site problem with population coefficient *a*=0.5. The problem size remained unchanged across the 50 runs while species population and site costs in each run were varied.

The shortest time used by the Önal et al. (2016) model is only about 1.2 seconds, while the longest time is 131.2 seconds. For the Williams (2002) model, the shortest and longest times are 0.3 seconds and 84.5 seconds, respectively. These results
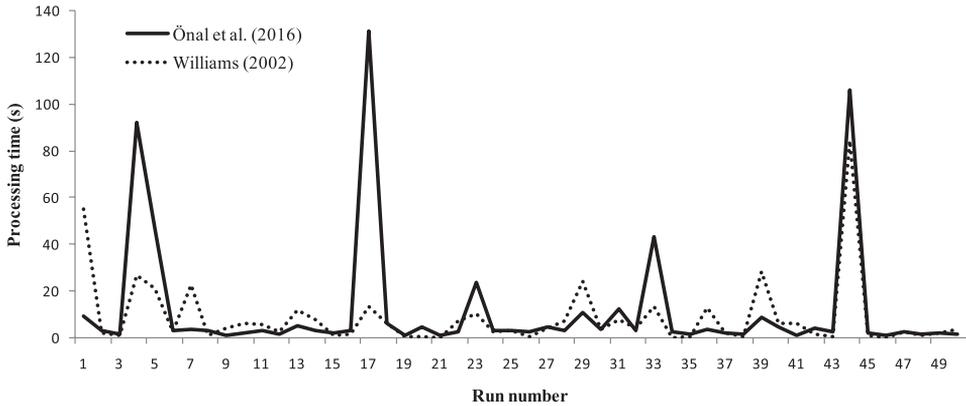
Figure 3. Computational efficiency of the Önal et al. (2016) and Williams (2002) models in solving a 100-site problem with various species distributions and site costs data.

indicate that alterations in the input data may increase or decrease the solution time substantially even if the model size remains the same. This can also be explained by the workings of the branch and bound procedure. One data set may yield a 'good' integer solution in an early iteration and eliminate a large portion of the branch and bound tree; therefore, a fewer number of nodes needs to be solved (or just the opposite may occur). This seems to have happened in runs 4, 17, 33 and 44 with the Önal et al. (2016) model and in runs 1, 4, 7, 29, 39 and 44 with the Williams (2002) model.

We should note that none of these factors alone can determine the performance of a MIP model, instead they affect the model's performance together. Solution techniques have been designed to speed up the computational process, such as supplying a cut value (Önal 2003, Cerdeira et al. 2005, Carvajal et al. 2013) or using a hybrid two-phase solution method (Conrad et al. 2012). We did not explore the effects of these techniques on the reserve design models tested here.

Conclusions

We have compared the computational efficiencies of eight representative optimal reserve design models that explicitly incorporate contiguity and compactness considerations. We identified the models presented by Williams (2002) and Önal et al. (2016) as the most efficient models. They both solved a 100-site contiguity problem easily, but encountered difficulty in solving the 200-site and larger contiguity problems. When compactness is also included, the Williams (2002) model had difficulty in solving a 100-site problem, while the Önal et al. (2016) model could solve a 200-site problem in a reasonable processing time (about 20 minutes). The Williams (2002) model could solve contiguity problems and some contiguity and compactness problems with up to 2000 sites when a large fraction of the candidate sites (such as 75 percent or more)

needs to be selected. The Önal et al. (2016) model displayed a steady performance and outperformed the Williams (2002) model when both contiguity and compactness are considered, but still the model could not be solved within 1 hour when 500 sites are involved. We identified model size, share of selected sites, model structure and input data as the four important factors affecting the computational efficiency of spatially-explicit optimal reserve design models. When developing such a model, one should try to define as few variables and equations as possible and to construct the model in such a way that its constraint matrix is unimodular (ReVelle 1993). These findings may be useful to conservation planners who select and use models in reserve design or management practices. This work may also provide guidance to academic researchers who try to develop computationally efficient optimal reserve design models.

The current nature reserves, whether terrestrial or marine, may not be adequate in providing sufficient protection to biodiversity. For instance, only over 1.89% of the marine protected areas (MPAs) worldwide is covered by exclusively no-take MPAs, far from the Convention on Biological Diversity's (CBD) Aichi Target 11 of 10% MPA coverage by 2020 and even further from the 30% target recommended by the IUCN World Parks Congress 2014 (IUCN 2018). It is almost certain that the current systems of nature reserves should be expanded. Acquiring more protected areas, however, would have to be done in a world where the human economy struggles to grow and the ecosystems and nonhuman species demands protection. Indeed, there is a conflict between economic growth and wildlife conservation (Czech 2014). Various disciplines such as ecological economics and environmental philosophy discussed this conflict and offered principles and guidelines to mediate the conflict and achieve sustainability. The science of nature reserve design or, more specifically, the optimal reserve design models, can contribute too. These models aim to solve practical conservation problems by incorporating ecological and socioeconomic considerations and help best prioritise conservation effort by guaranteeing optimal solutions. With the remarkable improvements in optimisation methods and computational power, it is reasonable to anticipate that spatial optimisation models will likely play a more significant role in the future than it does today in analysing and acquiring new protected areas.

Acknowledgments

References

Ando A, Camm JD, Polasky S, Solow A (1998) Species distribution, land values and efficient conservation. Science 279(5359): 2126–2128. https://doi.org/10.1126/science.279.5359.2126

Aplet GH, McKinley PS (2017) A portfolio approach to managing ecological risks of global change. Ecosystem Health and Sustainability 3(2): 1–15. https://doi.org/10.1002/ehs2.1261

Ball IR, Possingham HP, Watts ME (2009) Marxan and relatives: software for spatial conservation prioritization. In: Moilanen A, Wilson KA, Possingham HP (Eds) Spatial Conservation Prioritisation: Quantitative Methods and Computational Tools. Oxford University Press, Oxford, 185–195.

Billionnet A (2013) Mathematical optimization ideas for biodiversity conservation. European Journal of Operational Research 231(3): 514–534. https://doi.org/10.1016/j.ejor.2013.03.025

Camm JD, Polasky S, Solow A, Csuti B (1996) A note on optimal algorithms for reserve site selection. Biological Conservation 78(3): 353–355. https://doi.org/10.1016/0006-3207(95)00132-8

Carvajal R, Constantino M, Goycoolea M, Vielma JP, Weintraub A (2013) Imposing Connectivity Constraints in Forest Planning Models. Operations Research 61(4): 824–836. https://doi.org/10.1287/opre.2013.1183

Cerdeira JO, Gaston KJ, Pinto LS (2005) Connectivity in priority area selection for conservation. Environmental Modeling and Assessment 10(3): 183–192. https://doi.org/10.1007/s10666-005-9008-4

Church RL, Stoms DM, Davis FW (1996) Reserve selection as a maximal covering location problem. Biological Conservation 76(2): 105–112. https://doi.org/10.1016/0006-3207(95)00102-6

Cocks KD, Baird IA (1989) Using mathematical programming to address the multiple reserve selection problem: An example from the Eyre Peninsula, South Australia. Biological Conservation 49(2): 113–130. https://doi.org/10.1016/0006-3207(89)90083-9

Conrad JM, Gomes CP, vanHoeve WJ, Sabharwal A, Suter JF (2012) Wildlife corridors as a connected subgraph problem. Journal of Environmental Economics and Management 63(1): 1–18. https://doi.org/10.1016/j.jeem.2011.08.001

Czech B (2014) The conflict between economic growth and wildlife conservation, In: Gates JE, Trauger DL, Czech B (Eds) Peak Oil, Economic Growth, and Wildlife Conservation. Springer, New York, 99–117. https://doi.org/10.1007/978-1-4939-1954-3_5

Duque JC, Church RL, Middleton RS (2011) The p-regions problem. Geographical Analysis 43(1): 104–126. https://doi.org/10.1111/j.1538-4632.2010.00810.x

Fischer DT, Church RL (2003) Clustering and compactness in reserve site selection: An extension of the biodiversity management area selection model. Forest Science 49: 555–565.

Jafari N, Hearne J (2013) A new method to solve the fully connected reserve network design problem. European Journal of Operational Research 231(1): 202–209. https://doi.org/10.1016/j.ejor.2013.05.015

Jafari N, Nuse BL, Moore CT, Dilkina B, Hepinstall-Cymerman J (2017) Achieving full connectivity of sites in the multiperiod reserve network design problem. Computers & Operations Research 81: 119–127. https://doi.org/10.1016/j.cor.2016.12.017

Kingsland SE (2002) Creating a science of nature reserve design: Perspectives from history. Environmental Modeling and Assessment 7(2): 61–69. https://doi.org/10.1023/A:1015633830223

Miller CE, Tucker AW, Zemlin RA (1960) Integer Programming Formulation of Traveling Salesman Problems. Journal of the Association for Computing Machinery 7(4): 326–329. https://doi.org/10.1145/321043.321046

Moilanen A, Wilson KA, Possingham HP (2009) Spatial Conservation Prioritization: Quantitative Methods and Computational Tools.Oxford University Press, Oxford and New York.

Önal H (2003) First-best, second-best, and heuristic solutions in conservation reserve selection. Biological Conservation 115(1): 55–62. https://doi.org/10.1016/S0006-3207(03)00093-4

Önal H, Briers RA (2002) Incorporating spatial criteria in optimum reserve network selection. Proceedings. Biological Sciences 269(1508): 2437–2441. https://doi.org/10.1098/rspb.2002.2183

Önal H, Briers RA (2006) Optimum selection of a connected reserve network. Operations Research 54(2): 379–388. https://doi.org/10.1287/opre.1060.0272

Önal H, Wang Y (2008) A graph theory approach for designing conservation reserve networks with minimal fragmentation. Networks 52(2): 142–152. https://doi.org/10.1002/net.20211

Önal H, Wang Y, Dissanayake ST, Westervelt JD (2016) Optimal design of compact and functionally contiguous conservation management areas. European Journal of Operational Research 251(3): 957–968. https://doi.org/10.1016/j.ejor.2015.12.005

Possingham H, Ball I, Andelman S (2000) Mathematical methods for identifying representative reserve networks. In: Ferson S, Burgman M (Eds) Quantitative Methods for Conservation Biology. Springer, New York, 291–305. https://doi.org/10.1007/0-387-22648-6_17

Pressey RL, Nicholls AO (1989) Efficiency in conservation evaluation: Scoring versus iterative approaches. Biological Conservation 50(1–4): 199–218. https://doi.org/10.1016/0006-3207(89)90010-4

Pressey RL, Possingham HP, Day JR (1997) Effectiveness of alternative heuristic algorithms for identifying indicative minimum requirements for conservation reserves. Biological Conservation 80(2): 207–219. https://doi.org/10.1016/S0006-3207(96)00045-6

ReVelle C, ReVelle CS (1993) Facility siting and integer-friendly programming. European Journal of Operational Research 65(2): 147–158. https://doi.org/10.1016/0377-2217(93)90329-L

Rodrigues ASL, Gaston KJ (2002) Optimisation in reserve selection procedures – why not? Biological Conservation 107(1): 123–129. https://doi.org/10.1016/S0006-3207(02)00042-3

Sarkar S (2012) Complementarity and the selection of nature reserves: Algorithms and the origins of conservation planning, 1980–1995. Archive for History of Exact Sciences 66(4): 397–426. https://doi.org/10.1007/s00407-012-0097-6

Scott JM, Davis F, Csuti B, Noss R, Butterfield B, Groves C, Anderson H, Caicco S, D'Erchia F, Edwards TC, Ulliman J, Wright RG (1993) Gap Analysis: A Geographic Approach to Protection of Biological Diversity. Wildlife Monographs 123: 3–41.

Shirabe T (2005) A model of contiguity for spatial unit allocation. Geographical Analysis 37(1): 2–16. https://doi.org/10.1111/j.1538-4632.2005.00605.x

Shirabe T (2009) Districting modeling with exact contiguity constraints. Environment and Planning. B, Planning & Design 36(6): 1053–1066. https://doi.org/10.1068/b34104

Underhill LG (1994) Optimal and suboptimal reserve selection algorithms. Biological Conservation 70(1): 85–87. https://doi.org/10.1016/0006-3207(94)90302-6

Wang Y, Önal H (2016) Optimal design of compact and connected nature reserves for multiple species. Conservation Biology 30(2): 413–424. https://doi.org/10.1111/cobi.12629

Williams JC (2002) A zero-one programming model for contiguous land acquisition. Geographical Analysis 34(4): 330–349. https://doi.org/10.1111/j.1538-4632.2002.tb01093.x

Williams JC, ReVelle CS, Levin SA (2004) Using mathematical optimization models to design nature reserves. Frontiers in Ecology and the Environment 2(2): 98–105. https://doi.org/10.1890/1540-9295(2004)002[0098:UMOMTD]2.0.CO;2

Williams JC, ReVelle CS, Levin SA (2005) Spatial attributes and reserve design models: A review. Environmental Modeling and Assessment 10(3): 163–181. https://doi.org/10.1007/s10666-005-9007-5