



Unifying reserve design strategies with graph theory and constraint programming

Dimitri Justeau-Allaire, Philippe Birnbaum, Xavier Lorca

► To cite this version:

Dimitri Justeau-Allaire, Philippe Birnbaum, Xavier Lorca. Unifying reserve design strategies with graph theory and constraint programming. CP 2018 - International conference on principles and practice of Constraint Programming, Aug 2018, Lille, France. p.507-523, 10.1007/978-3-319-98334-9_33 . hal-01885111

HAL Id: hal-01885111

<https://imt-mines-albi.hal.science/hal-01885111v1>

Submitted on 8 Nov 2018

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Unifying Reserve Design Strategies with Graph Theory and Constraint Programming

Dimitri Justeau-Allaire^{1,2,3}, Philippe Birnbaum^{1,2,3}, and Xavier Lorca⁴

¹ CIRAD, UMR AMAP, F-34398 Montpellier, France

² Institut Agronomique néo-Calédonien (IAC), 98800 Noumea, New Caledonia

³ AMAP, Univ Montpellier, CIRAD, CNRS, INRA, IRD, Montpellier, France

{dimitri.justeau-allaire,philippe.birnbaum}@cirad.fr

⁴ ORKID, Centre de Génie Industriel, IMT Mines Albi

Campus Jarlard, 81013 Albi cedex 09, France

xavier.lorca@mines-albi.fr

Abstract. The delineation of areas of high ecological or biodiversity value is a priority of any conservation program. However, the selection of optimal areas to be preserved necessarily results from a compromise between the complexity of ecological processes and managers' constraints. Current reserve design models usually focus on few criteria, which often leads to an oversimplification of the underlying conservation issues. This paper shows that Constraint Programming (CP) can be the basis of a more unified, flexible and extensible framework. First, the reserve design problem is formalized. Secondly, the problem is modeled from two different angles by using two graph-based models. Then CP is used to aggregate those models through a unique Constraint Satisfaction Problem. Our model is finally evaluated on a real use case addressing the problem of rainforest fragmentation in New Caledonia, a biodiversity hotspot. Results are promising and highlight challenging perspectives to overtake in future work.

1 Introduction

Human activities are exerting pressure on natural habitats, which generally results in a loss of surface and an increase of fragmentation. As a consequence, many species depending on those habitats are threatened, sometimes with extinction. In this context, it is essential to devote an important part of conservation efforts in the protection of natural habitats through the establishment of nature reserves [1–4]. Designing a reserve system is a difficult process involving a trade-off between the conservation targets and the socioeconomic constraints. This problem is known as the *reserve design problem*. The associated questions are at the crossroad between conservation biology, geography, mathematics, computer science, decision theory and environmental philosophy [5]. In this paper, we focus on the mathematical modeling and the computational solving of the reserve design problem. From this point of view, it is a decision and/or optimization problem. In almost all cases, the combinatorial complexity justifies the need of a systematic approach based on mathematical modeling and computational tools.

In the literature, two major aspects of the reserve design problem usually stand out: the *feature covering* and the *spatial configuration*. The first is often referred as the reserve (or site) selection problem [6–9]. In extension, we refer to the reserve design problem when spatial attributes are considered [10–14]. Current models usually focus on a few aspects of the problem because: (1) they provide an ad-hoc solution to a specific instance of the problem, or (2) they are limited by the modeling paradigm. However, there is a need for a more unified and flexible framework [15] which, in our opinion and based on our experience in New Caledonia, could help to reduce the gap between computer scientists, conservation scientists, and practitioners.

In this paper, we show how the combination of graph-based models with CP can be the basis of such a framework. After a detailed description of the reserve design problem (Section 2), we presents two graph-based models (Section 3). One model is dedicated to the constraint representation of the features covering issues (Section 3.1) and the other one is dedicated to the constraint representation of the spatial issues (Section 3.2). We then unify the models throughout a single CP model based on the Choco constraint solver [16] (Section 4). Finally, a realistic operational use case on the problem of rainforest fragmentation in New Caledonia is depicted and first results are discussed (Section 5).

2 Description of the Problem

The reserve design is a decision and/or optimization problem in the discretized geographical space. Given a set of geographical features (e.g., Fig. 2), we are looking for a reserve system satisfying several criteria, in accordance to conservation targets. In this section, we describe and formalize the problem precisely. We start by defining the characteristics of the problem and then define a set of criteria that can be required for a reserve system.

2.1 Characteristics of the Problem – Input Data

The Discretized Geographical Space. The geographical space is tessellated into n granular parcels, which are the decision variables of the problem. Several tessellation methods are possible [17, 18]. The most commonly used is the regular square grid (illustrated in Fig. 1). We choose to this method in this paper.

We denote the number of rows by r , the number of columns by c and the set of parcels by \mathcal{P} . We identify a single parcel with the letter i , and index the parcels with integers from 0 to $n - 1$: $\mathcal{P} = \{ i \mid i \in \llbracket 0, n \rrbracket \}$. While this indexing is not the most convenient for a grid, it has the advantage to be independent of the tessellation method and thus offers extensibility for future work. Finally, we use the 8-connected (cf. Fig. 1) neighborhood to define the adjacency between the parcels, in opposition to the 4-connected neighborhood.

The Environmental Features. The geographical space is characterized by a set of m environmental features. A feature can be anything that can be spatially represented (e.g. the presence of a species, a certain type of habitat, human

constructions). We denote by \mathcal{F} the set of environmental features and use the letter j to identify the features: $\mathcal{F} = \{ j \mid j \in \llbracket 0, m \rrbracket \}$.

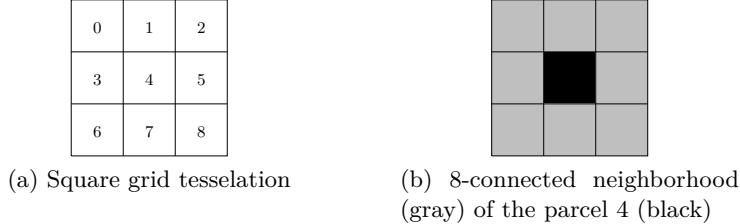


Fig. 1. Square grid tessellation and 8-connected neighborhood illustrations.

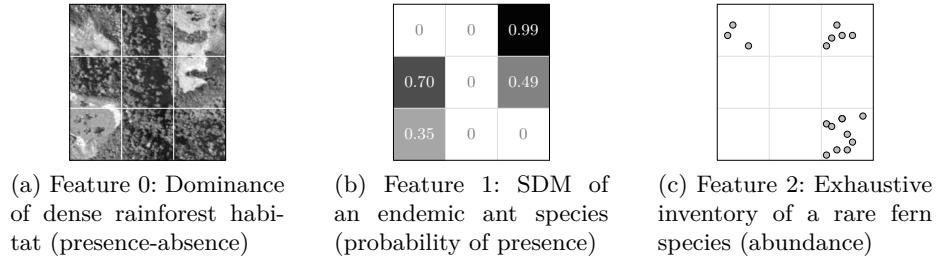


Fig. 2. Three feature examples.

The Values of the Features. To each feature j is associated a set \mathcal{V}_j , representing the available data about j among the parcels: $\mathcal{V}_j = \{ v_{ji} \in \mathbb{R}^+ \mid i \in \mathcal{P} \}$. Each $v_{ji} \in \mathcal{V}_j$ corresponds to a value describing the feature j in the parcel i . Three types of data are possible: the presence-absence data, the abundance data and the probability of presence data. An example for each data type is given in Fig. 2, and below is a short description for each of them:

- *Presence-absence*: if j is present in the parcel i , $v_{ji} = 1$, else $v_{ji} = 0$. For each $(j, i) \in \mathcal{F} \times \mathcal{P}$ we then have $v_{ji} \in \{0, 1\}$. The presence-absence data is often used to describe the occurrence distribution of a species or a particular characteristic of the landscape (e.g. forest, savanna, fields, roads, cities).
- *Abundance*: in this case, v_{ji} represents a quantitative value about the feature j in the parcel i (e.g. density of trees per parcel, average annual rainfall). For each $(j, i) \in \mathcal{F} \times \mathcal{P}$ we then have $v_{ji} \in [0, +\infty[$.
- *Probability of presence*: it can be possible to evaluate the probability of presence of a feature j for every parcel in \mathcal{P} . The most common situation is the use of Species Distribution Models (SDMs), that are able to combine observations of a species with environmental data to predict its spatial distribution [19, 20]. With probability of presence data, for each $(j, i) \in \mathcal{F} \times \mathcal{P}$ we have $v_{ji} \in [0, 1]$.

The Domains/Anti-Domains of the Features. As illustrated in Fig. 3, to each feature, is associated a set \mathcal{D}_j and its complement $\overline{\mathcal{D}}_j$. \mathcal{D}_j represents the domain of j , that is, the parcels where j is present or where the probability of presence of j is not null: $\mathcal{D}_j = \{i \in \mathcal{P} \mid v_{ji} > 0\}$. Conversely, $\overline{\mathcal{D}}_j$ represents the anti-domain of j , that is, the parcels where j is not present or where the probability of presence of j is null: $\overline{\mathcal{D}}_j = \{i \in \mathcal{P} \mid v_{ji} = 0\}$.

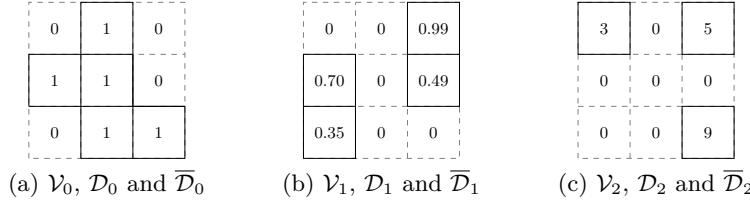


Fig. 3. The values, domains and anti-domains associated with the features in Fig. 2. The domains are represented with solid lines and the anti-domains with dashed lines.

2.2 The Reserve System – Solution of the Problem

In the first place, we define the terms “*parcel*” (sometimes called “*site*” in the literature), “*reserve*” and “*reserve system*”. As defined in the previous subsection, a parcel is a granular selection unit of the discretized geographical space. On top of that, a reserve is a set of spatially continuous selected parcels (note that a single selected isolated parcel is a reserve). Finally a reserve system is a set of spatially disjoint reserves (note that a reserve system can be composed of a single reserve). We illustrated the previous definitions in Fig. 4.

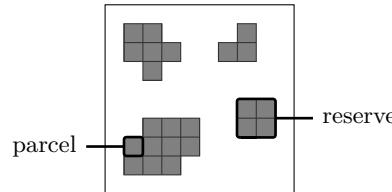


Fig. 4. Illustration of a reserve system, composed of four reserves, themselves made of several adjacent parcels.

Given that, a solution to our problem is a reserve system whose attributes are satisfying a set of criteria, themselves depending on the conservation question. We denote such a reserve system by \mathcal{S} , its number of reserves by n_r and its k^{th} reserve by \mathcal{X}_k : $\mathcal{S} = \{ \mathcal{X}_k \subseteq \mathcal{P} \mid k \in [\![0, n_r]\!] \}$ (where \mathcal{P} is the set of parcels).

2.3 Required Criteria for a Reserve System

According to the underlying conservation questions, several criteria can be required for a reserve system. We distinguish between the feature covering criteria and the spatial criteria.

Feature Covering Criteria. By providing one of the first formalization of the reserve selection problems, ReVelle et al. [7] introduced three fundamental feature covering criteria.

- *Covered Features.* Among the features that are covered (with certainty) by the reserve system \mathcal{S} , we want a set of mandatory features \mathcal{F}' to be represented (e.g. rare or endangered species).
- α -*Covered Features.* Assuming that the v_{ji} 's are pairwise independent, we want a set of features \mathcal{F}' to be covered by \mathcal{S} with a probability of at least α . This criterion is helpful when probability of presence data is available.
- k -*Redundant Features.* A feature j is k -redundant in the reserve system \mathcal{S} if and only if it is covered (with certainty) by at least k distinct parcels. We want to enforce this property for a set of features \mathcal{F}' (e.g. for increasing the chances of persistence of vulnerable species).

Spatial Criteria. A list of six geometric principles had been defined by Diamond [10] and Williams et al. [11] summarized them into six spatial attributes to take into account when designing a reserve system: the *number of reserves*, the *reserve areas* (by extension we define the *reserve system area*), the *reserve proximity*, the *reserve connectivity*, the *reserve shape* and *core areas and buffer zones*. Here we consider three of those spatial attributes (expressed as criteria) and keep the remaining ones for future work.

- *Number of reserves.* Determining if the best suited is a "single large or several small reserves" (SLOSS), or a "few large or many small reserves" (FLOMS) is a well known debate in ecology ([10, 21]). The conclusion is that the answer strongly depends on the context, and that flexibility is needed. We therefore want to set a minimum value N_{min} and/or a maximum value N_{max} for the number of reserves.
- *Reserve Areas.* Following the previous criterion, it is also essential to provide control on the reserve areas by setting a minimum area A_{min} and a maximum area A_{max} .
- *Reserve System Area.* It should also be possible to express this criterion on the whole reserve system area, by setting a minimum total area $A_{T_{min}}$ and a maximum total area $A_{T_{max}}$.

3 The Graph-Based Models

In this section, we present two graph-based models. The first one is a resource allocation model that will enable us to express the feature covering criteria in the form of constraints. In the same way, the second one is a spatial model

that will enable us to control the spatial criteria. In both models, each parcel is represented by a vertex. This common characteristic is essential since it is the one that makes the aggregation of the models possible, through a set of appropriate channeling constraints.

3.1 The Resource Allocation Graph

We consider parcels as resources that can be allocated to the conservation of features, then considered as tasks, and thus define the directed graph $G_r = (V_r, A_r)$, also called the *resource allocation graph*. The vertices of G_r are partitioned into three disjoint sets F_r , P_r and $\{s, t\}$. F_r represents the *feature* (or *task*) vertices, P_r represents the *parcel* (or *resource*) vertices, s is the source vertex and t the sink vertex.

$$\begin{aligned} V_r &= F_r \cup P_r \cup \{s, t\}; \\ F_r &= \{f_j \mid j \in \mathcal{F}\}; \\ P_r &= \{p_i \mid i \in \mathcal{P}\}. \end{aligned} \quad (1)$$

Furthermore, using $A_r(X, Y)$ as the notation for the set of all $X-Y$ arcs, we define the arcs of G_r in the following way:

$$A_r = A_r(s, F_r) \cup A_r(F_r, P_r) \cup A_r(P_r, t). \quad (2)$$

$A_r(s, F_r)$ and $A_r(P_r, t)$ are defined such that there is an arc from s to each feature vertex and an arc from each parcel vertex to t :

$$\begin{aligned} A_r(s, F_r) &= \{(s, f_j) \mid f_j \in F_r\}; \\ A_r(P_r, t) &= \{(p_i, t) \mid p_i \in P_r\}. \end{aligned} \quad (3)$$

Moreover, $A_r(F_r, P_r)$ represent the possible allocations between F_r and P_r . More precisely, there is an arc from a feature vertex f_j to a parcel vertex p_i if and only if the feature j is represented in the parcel i , that is $i \in \mathcal{D}_j$. We then have:

$$A_r(F_r, P_r) = \bigcup_{j \in \mathcal{F}} \{(f_j, p_i) \mid i \in \mathcal{D}_j\}. \quad (4)$$

On the arcs of G_r , we define a lower bound (or demand) function $l : A_r \mapsto \mathbb{R}_\infty^+$ and an upper bound (or capacity) function $u : A_r \mapsto \mathbb{R}_\infty^+$ such that if f is a flow in G_r :

$$\forall a \in A_r, \quad l(a) \leq f(a) \leq u(a). \quad (5)$$

Finally, we define $H_r : \mathcal{P}(P_r) \mapsto \mathcal{P}(V_r) \times \mathcal{P}(A_r)$ that associates to a set $X \subseteq P_r$ the subgraph of G_r induced by $\{s, t\} \cup F_r \cup X$, that is, the resource allocation graph obtained when only considering a subset of parcels. We denote by $V_r[H_r(X)]$ the vertices of $H_r(X)$ and by $A_r[H_r(X)]$ the arcs of $H_r(X)$. An example is provided in Fig. 5.

$$H_r(X) = G_r[\{s, t\} \cup F_r \cup X]. \quad (6)$$

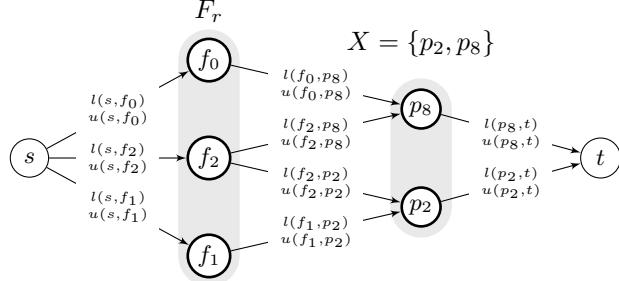


Fig. 5. $H_r(\{p_2, p_8\})$ associated with the example in Fig. 2. Lower and upper bounds are represented on the arcs.

Expressing Feature Covering Criteria as Constraints. From this point, to each feature covering criterion (as defined in the previous section) we associate a constraint that can be applied on the resource allocation model. More precisely, if \mathcal{S} is a reserve system and X_s its associated set of parcel vertices, a criterion is satisfied by \mathcal{S} if and only if its associated constraint is satisfied by $H_r(X_s)$. We express these constraints as flow constraints by defining the value of l and u on certain arcs. When the value of l is not explicitly defined, it is unconstrained and then set to 0. Similarly, u is set to $+\infty$ when its value is not explicitly defined.

Covered Features. In our resource allocation model, we can easily express this criterion as a flow constraint on $H_r(X_s)$.

Constraint 1: Covered Features.

Input parameter(s): A set of features $\mathcal{F}' \subseteq \mathcal{F}$.

The set of features \mathcal{F}' is covered by \mathcal{S} if and only if $H_r(X_s)$ admits a feasible flow f verifying (5) when:

$$\begin{cases} l(s, f_j) = 1, & \forall f_j \in \mathcal{F}' \\ l(f_j, p_i) = 1, & \forall (f_j, p_i) \in A_r(F_r, P_r) \\ u(f_j, p_i) = 1, & \forall (f_j, p_i) \in A_r(F_r, P_r) \text{ such that } v_{ji} \geq 1 \\ u(f_j, p_i) = 0, & \forall (f_j, p_i) \in A_r(F_r, P_r) \text{ such that } v_{ji} < 1. \end{cases} \quad (7)$$

α -Covered Features. To express this criterion, we assume that the probabilities of presence v_{ji} are pairwise independent. We then rely on the probability of absence $q_{ji} = (1 - v_{ji})$ and express the constraint as:

$$\forall j \in \mathcal{F}', \quad \prod_{i \in \mathcal{S}} q_{ji} \leq 1 - \alpha. \quad (8)$$

We then express the α -presence constraint in the following way:

Constraint 2: α -Covered Features.

Input parameter(s): A set of features \mathcal{F}' and a real $\alpha \in [0, 1]$.

The set of features \mathcal{F}' is covered by \mathcal{S} with a probability of at least α if and only if $H_r(X_s)$ admits a feasible flow f verifying (5) when:

$$\begin{cases} l(s, f_j) = -\log(1 - \alpha), & \forall f_j \in F'_r; \\ u(f_j, p_i) = -\log(q_{ji}), & \forall (f_j, p_i) \in A_r(F_r, P_r) \text{ such that } v_{ji} < 1. \end{cases} \quad (9)$$

k-Redundant Features. Since the k-redundancy is actually a generalization of the covering features criterion, we can also express it as a flow constraint on $H_r(X_s)$.

Constraint 3: k-Redundant Features.

Input parameter(s): A set of features \mathcal{F}' and a positive integer k .

The k-redundancy of the set of features \mathcal{F}' in the reserve \mathcal{S} is satisfied if and only if $H_r(X_s)$ admits a feasible flow f verifying (5) when:

$$\begin{cases} l(s, f_j) = k, & \forall f_j \in F'_r; \\ l(f_j, p_i) = 1, & \forall (f_j, p_i) \in A_r(F_r, P_r); \\ u(f_j, p_i) = 1, & \forall (f_j, p_i) \in A_r(F_r, P_r) \text{ such that } v_{ji} \geq 1; \\ u(f_j, p_i) = 0, & \forall (f_j, p_i) \in A_r(F_r, P_r) \text{ such that } v_{ji} < 1. \end{cases} \quad (10)$$

3.2 The Spatial Graph

We now define the undirected graph $G_s = (V_s, E_s)$, the *spatial graph*, which is a representation of the discretized geographical space \mathcal{P} (a $r \times c$ regular square grid in our case). Once again, to each parcel i of \mathcal{P} , we associate a vertex p_i , we then have:

$$V_s = \{ p_i \mid i \in \mathcal{P} \}. \quad (11)$$

Moreover, the edges of G_s are defined such that if p_u and p_v are two vertices, there is an edge between p_u to p_v if and only if the parcels u and v are spatially adjacent. The edges of G_s can be partitioned into four disjoint sets: the horizontal edges (E_H), the vertical edges (E_V), the north-west to south-east diagonal edges ($E_{N_W S_E}$) and the north-east to south-west diagonal edges ($E_{N_E S_W}$).

$$\begin{aligned} E_s &= E_H \cup E_V \cup E_{N_W S_E} \cup E_{N_E S_W}; \\ E_H &= \{ (p_i, p_{i+1}) \mid i \in \mathcal{P} \wedge \neg(i+1) \equiv 0(c) \}; \\ E_V &= \{ (p_i, p_{i+c}) \mid i \in \mathcal{P} \wedge i < c(r-1) \}; \\ E_{N_W S_E} &= \{ (p_i, p_{i+c+1}) \mid i \in \mathcal{P} \wedge i < c(r-1) \wedge \neg(i+1) \equiv 0(c) \}; \\ E_{N_E S_W} &= \{ (p_i, p_{i+c-1}) \mid i \in \mathcal{P} \wedge i < c(r-1) \wedge \neg i \equiv 0(c) \}. \end{aligned} \quad (12)$$

See Fig. 6 for an illustration of the above equation. Also note that it takes into account the extremal positions of the grid. In fact, the parcels located in

the first column are the ones whose index is a multiple of c , that is $i \equiv 0 \pmod{c}$. Moreover, the parcels located in the last column are the ones preceding those that are located in the first column, that is $(i+1) \equiv 0 \pmod{c}$. Finally, the parcels located in the last line are the ones satisfying $i < c(r-1)$.

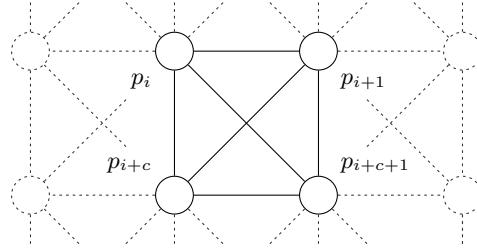


Fig. 6. Illustration of a portion of a spatial graph G_s associated with a $r \times 4$ square grid, using the 8-connectivity neighborhood definition.

Expressing Spatial Criteria as Constraints. Similarly to what had been defined for the resource allocation graph, to a solution \mathcal{S} of the problem we associate $X_s \subseteq V_s$. Moreover, to each reserve $\mathcal{X}_k \in \mathcal{S}$ we associate $X_s(k) \in X_s$, the vertices associated to the parcels of \mathcal{X}_k . We now express each spatial criterion as a constraint that can be applied on $G_s[X_s]$.

Number of Reserves. We easily express this criterion by bounding the number of connected components (NCC, [22–24]) in $G_s[X_s]$.

Constraint 4: Number of Reserves.

Input parameter(s): Two positive integer N_{min} and N_{max} .

Ensuring that the number of reserves in \mathcal{S} is bounded by N_{min} and N_{max} is equivalent to bounding the NCC of $G_s[X_s]$ with N_{min} and N_{max} .

$$N_{min} \leq \text{NCC}(G_s[X_s]) \leq N_{max}. \quad (13)$$

Reserve Areas. We express this criterion as a constraint on the number of vertices of the smallest connected component of $G_s[X_s]$ (MIN_NCC, [25, 23, 24]) and on the number of vertices of the largest connected component of $G_s[X_s]$ (MAX_NCC, [25, 23, 24]).

Constraint 5: Reserve Areas.

Input parameter(s): Two positive integer A_{min} and A_{max} .

Ensuring that the area of every reserve $\mathcal{X}_k \in \mathcal{S}$ is bounded by A_{min} and A_{max} is equivalent to constraining the lower bound of $\text{MIN_NCC}(G_s[X_s])$ to A_{min} and the upper bound of $\text{MAX_NCC}(G_s[X_s])$ to A_{max} .

$$\begin{aligned} \forall k \in [0, n_r], \quad & \text{MIN_NCC}(G_s[X_s]) \geq A_{min}; \\ & \text{MAX_NCC}(G_s[X_s]) \leq A_{max}. \end{aligned} \quad (14)$$

Reserve System Area. In the current case of a regular tessellation method, we can control the whole reserve system's area by bounding the norm of X_s .

Constraint 6: Reserve System Area.

Input parameter(s): Two positive integer $A_{T_{min}}$ and $A_{T_{max}}$.

Ensuring that the total area of the reserve system is bounded by $A_{T_{min}}$ and $A_{T_{max}}$ is equivalent to bounding $|X_s|$.

$$A_{T_{min}} \leq |X_s| \leq A_{T_{max}}. \quad (15)$$

4 The CP Model

In this section we present our CP model for the reserve design problem. For its implementation, we rely on the solver Choco [16] and its extension Choco-graph [26], which provides graph variables and constraints.

The Decision Variables. We naturally model the parcels with a boolean variable array, named `parcels`. If the parcel i is selected in the reserve system, `parcels[i] = 1`, else `parcels[i] = 0`.

```
BoolVar[] parcels = model.boolVarArray("parcels", n);
```

These decision variables are the cornerstone of our CP model because they allow us to aggregate the two models we introduced in the previous section.

The Feature Covering Constraints. Given the particular configuration of the resource allocation graph, we are able to express each feature covering constraint with several local flow conservation inequalities, one for each feature involved in the constraint. Note that we would certainly benefit from the filtering of a global flow constraint [27]. However, there is no such constraint implemented in Choco at the time we are writing this paper. We thus keep this idea for future work.

Constraint 1, Covered Features (7): with local flow conservation inequalities, (7) becomes:

$$\forall j \in \mathcal{F}', \quad \sum_{i=0}^{n-1} b_i \times (v_{ji} \geq 1) \geq 1.$$

Below is the implementation with Choco 4, using the `scalar` constraint.

```

for (int j : featuresToCover) {
    int[] coeffs = Arrays.stream(V[j])
        .mapToInt(v -> (v >= 1) ? 1 : 0)
        .toArray();
    model.scalar(parcels, coeffs, ">=", 1).post();
}

```

Constraint 2, α -Covered Features (9): the coefficients in the `scalar` constraint must be integers. We then retain only two digits of precision for the probabilities of presence. If $\alpha \in [0, 0.99]$ then $-\log(1-\alpha) \in [0, 2]$, moreover, with this precision the order of the smallest variation between two values ($\alpha = 0$ and $\alpha = 0.01$) is 10^{-3} , we thus multiply our local flow inequality by 10^3 in order to stay in the integer domain. If $v_{ji} \geq 1$, we set the flow upper bound to $-10^3 \log(1 - 0.999) = 3000$ as a replacement for $+\infty$. Consequently, we reduce (9) to:

$$\forall j \in \mathcal{F}', \quad \sum_{i=0}^{n-1} b_i \times \min(-10^3 \log(1 - v_{ji}), 3000) \geq -10^3 \log(1 - \alpha).$$

Below is the implementation with Choco 4.

```

for (int j : featuresToCover) {
    int[] coeffs = Arrays.stream(V[j])
        .mapToInt(
            v -> (v >= 1) ? 3000 : (int) (-1000 * Math.log10(1 - v)))
        .toArray();
    int scaled = (int) (-1000 * Math.log10(1 - alpha));
    model.scalar(parcels, coeffs, ">=", scaled).post();
}

```

Constraint 3, k-Redundant Features (10): similarly, we reduce (10) to:

$$\forall j \in \mathcal{F}', \quad \sum_{i=0}^{n-1} b_i \times (v_{ji} \geq 1) \geq k.$$

And implement it the following way with Choco 4:

```

for (int j : featuresToCover) {
    int[] coeffs = Arrays.stream(V[j])
        .mapToInt(v -> (v >= 1) ? 1 : 0)
        .toArray();
    model.scalar(parcels, coeffs, ">=", k).post();
}

```

The Spatial Constraints. We rely on Choco-graph to express the spatial constraints in our CP Model. First, we use a graph variable g to model the reserve system. Its kernel is the empty graph (GLB in the code), and its envelope is G_s (GUB in the code).

```

UndirectedGraph GLB = new UndirectedGraph(model, n, BIPARTITESET, false);
UndirectedGraph GUB = new UndirectedGraph(model, n, BIPARTITESET, false);
for (int i = 0; i < n; i++) {
    GUB.addNode(i);
    for (int ii : getNeighbors(i)) {
        GUB.addEdge(i, ii);
    }
}
UndirectedGraphVar g = model.graphVar("g", GLB, GUB);

```

Then, we link the graph variable `g` with the boolean variables `parcels` using the `nodesChanneling` constraint.

```
model.nodesChanneling(g, parcels).post();
```

We also force the existence of an edge between two selected adjacent parcels through an `edgeChanneling` constraint with a reified `and` constraint between each pair (i_1, i_2) of adjacent parcels. Doing so, we ensure that every existing edges between two selected vertices are also present in our graph variable.

```

BoolVar forceEdge = model.and(parcels[i1], parcels[i2]).reify();
model.edgeChanneling(g, forceEdge, i1, i2).post();

```

Constraint 4, Number of Reserves (13): we use the `nbConnectedComponents` and the `arithm` constraints.

```

IntVar nbCC = model.intVar("nbCC", Nmin, Nmax);
model.nbConnectedComponents(g, nbCC).post();

```

Constraint 5, Reserve Areas (14): at the time we are writing this paper, there is no constraint in Choco-graph for controlling the `MIN_NCC` and `MAX_NCC` graph properties. We thus implemented the `sizeConnectedComponents`⁵ constraint, which allows us to bound `MIN_NCC` and `MAX_NCC`.

```
model.sizeConnectedComponents(g, Amin, Amax).post();
```

Constraint 6, Reserve System Area (15): we can control the number of vertices of G_s (that is, the number of parcels) through the `nbNodes` graph constraint, or through the `sum` constraint over the decision variables.

```

IntVar nbParcels = model.intVar(Atmin, Atmax);
model.nbNodes(g, nbParcels).post(); // Option 1
model.sum(parcels, "=", nbParcels).post(); // Option 2

```

⁵ <https://gist.github.com/dimitri-justeau/8098af35824bbf8d52ef21282291e621>

5 Use case: Rainforest Fragmentation in New Caledonia

New Caledonia is biodiversity hotspot located in the South Pacific, slightly north of the tropic of the Capricorn. The flora of this large archipelago is distinguished by an exceptionally high rate of endemism. Like most of the world's remaining natural forests, New Caledonian rainforests are endangered with surface loss and fragmentation. A case study had been conducted in the south of New Caledonia in order to highlight "how does forest fragmentation affect tree communities" [28]. We relied on this case study and its associated dataset (up to date) for our use case, and considered the following fictive but realistic operational scenario:

"We want to establish a reserve system in which a pool of endangered species must be present. In addition, most of the other species known in the area must have a high probability to occur, or a high habitat suitability. The reserve system must be mostly covering rainforest areas. Its area and its number of reserves must be limited because of budget limitation. Moreover, each reserve must be large enough to ensure the persistence of the species."

Note. In this scenario, the objective is to protect both existing and potential rainforest areas. To do so, we relied on SDM layers that were generated with presence-only data and thus produce a score of habitat suitability rather than a standardized probability of presence. A high habitat suitability in a non-rainforest zone can then be interpreted as an adequate zone for recolonization.

5.1 Input Data, Constraints and Parameters

The original dataset consists of the mapping of a 60 km² landscape where 97 tree communities had been sampled in 88 digitized rainforest fragments (forest/non-forest). The dataset gathers 5431 identified trees belonging to 223 species. Moreover, an SDM raster layer was available for 173 of the species [29, 30]. Arbitrarily, we considered the 50 species without SDM as the endangered ones. We then prepared this dataset by tessellating the study area into a 46 × 75 regular square grid and by rasterizing the dataset according to this grid. Each parcel then has an area of about 1.7 ha. Note that we also defined a set of forbidden parcels corresponding to lakes and mining sites.

From this point, we defined a feature for each observed species in the area. When available, we relied on the SDM layer for the feature data (probability of presence data). We forced the values to 1 for the parcels where an observation is available. When no SDM was available, we only relied on the occurrence dataset (presence-absence). We represented the rainforest coverage as a presence-absence feature.

We then applied the *Covered Features* constraint for the set of endangered species, and the α -*Covered Features* constraint for the other species with $\alpha = 0.8$. In order to ensure a minimum rainforest area of 340 ha in the reserve system, we applied the *k-Redundant Features* constraint for the rainforest coverage, with

$k = 200$ parcels. Moreover, we enforced the forbidden parcels on the envelope of the graph variable \mathbf{g} . We then set the minimum area of the reserves to $A_{min} = 176$ parcels (about 300 ha) with the *Reserve Areas* constraint. In addition, we limited the reserve system area using the *Reserve System Area* constraint, with $A_{t_{max}} = 589$ parcels (about 1000 ha). According to those restrictive parameters, we allowed the number of reserve to be at most two, using the *Number of Reserves* constraint, with $N_{min} = 1$ and $N_{max} = 2$.

5.2 Questioning and Results

In the first place, the number of reserves and the number of parcels are critical parameters of our use case: the less the better. This is why we started by implementing a search strategy that starts by branching on the lower bound of the `nbCC` variable and continues by selecting the lower bound of the `parcels` variables, sorted in descending order by a score corresponding to the number of features with a value greater than 0.6 (cf. 7). The solver quickly found a solution to the decision problem (DP), as summarized in Table 1. Given that, we ran a first optimization problem (OP1) where we tried to minimize the total area of the reserve system, that is the `nbParcels` variable. We limited the computation time to 4 hours and retrieved the best solution found, which reduced the total area by 8% in comparison to DP (cf. Table 1). In order to cover more rainforest parcels, we ran a second optimization problem (OP2) in which we forced the `nbParcels` variable to be within 15% of the best value found in OP1, and tried to maximize k (the number of forest parcels), thus defined as an integer variable. After a limited run of 4 hours, we could increase the area of rainforest by 12% (cf. Table 1). A mapping of our results is provided in Fig. 7.

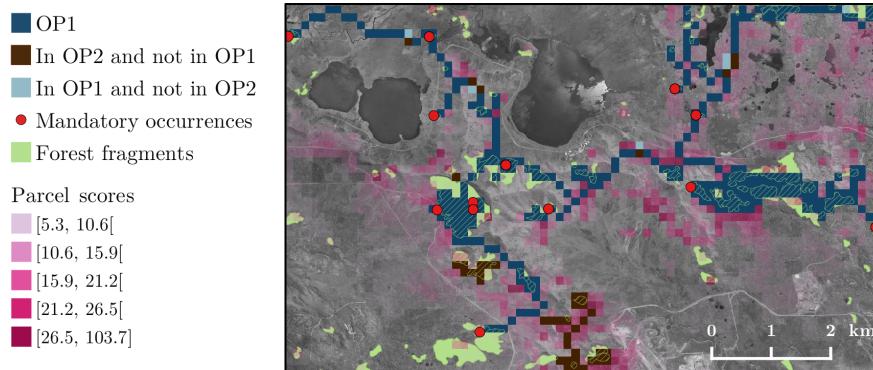


Fig. 7. Mapping of the use case best solutions. The parcel scores correspond to the heuristic score and the mandatory occurrences to rare species observed only once in the study area (they must then be covered by any solution).

Table 1. Use case results: resolution times and solution characteristics. All experiments were run on an Intel Core i5-5200U CPU (2.20GHz×4), with 7.7GB of RAM.

	DP	OP1	OP2
Resolution time	28s	3h24m	1h5m
Number of solutions found	1	8	25
Number of reserves	1	1	1
Number of parcels	318	292	328
Number of rainforest parcels	200	200	224

6 Conclusion and Challenges

To the best of our knowledge this paper tackles, for the very first time, the reserve design problem from a constraint programming point of view. It is also the first time that a reserve design model integrates such a diversity of constraints, simultaneously involving decisions based on occurrences, SDMs and spatial attributes with an exact approach. Although performance enhancements are needed, the combination of graph-based modeling and constraint programming reveals as a powerful and promising framework for dealing with the reserve design problem.

Based on a challenging use case, our model highlighted a solution compatible with the conservation strategy, namely a trend to link isolated forest patches in order to enhance the functioning of tree communities. However, in this use case we restrained to a binary landscape only composed of forest/non-forest while it is often assumed that a reserve system must include an assemblage of several landscape types. In such a mosaic, an important challenge lays in weighting and balancing the reserve system characteristics and shape in order to maintain (or restore) the functional connectivity inside and between the reserves. In fact, the functional isolation of an habitat leads to a reduction of biological flows, which tends to amplify its spatial isolation. Moreover, since the underlying processes are dynamic, robust solutions must rely both on the current state and future scenarios. It also remains to model the impacts of a reserve system on the off-reserve area, such as the creation of boundaries or enclosed areas.

These elements lead us to identify several lacks and challenges. First of all, the main lack is that Choco solver does not offer an implementation of the flow constraint [31, 32]. We will focus on its implementation in future work. Next, a bottleneck in the constraint propagation is the interaction between the constraint on the number and the size of the connected components [25]. We actually treat each one independently but we think that there is a possible enhancement of the filtering by dealing with their interaction. A first challenge for future work concerns our capacity to model constraints on the shape of the reserves by using graph properties, such as graph diameter, in order to design reserves that are compatible with the long-term persistence of species. A second challenge is more oriented to decision making aspects such as identifying key areas that have to be present in any solution. Finally, a last challenge is related to our capacity to take a dual point of view: is it possible to take into account managers' needs on the off-reserve area by adding constraints on the same graph representation?

References

1. Beier, P., Spencer, W., Baldwin, R.F., McRAE, B.H.: Toward Best Practices for Developing Regional Connectivity Maps. *Conservation Biology* **25**(5) (October 2011) 879–892
2. Baguette, M., Blanchet, S., Legrand, D., Stevens, V.M., Turlure, C.: Individual dispersal, landscape connectivity and ecological networks. *Biological Reviews* **88**(2) (May 2013) 310–326
3. Haddad, N.M., Brudvig, L.A., Clobert, J., Davies, K.F., Gonzalez, A., Holt, R.D., Lovejoy, T.E., Sexton, J.O., Austin, M.P., Collins, C.D., Cook, W.M., Damschen, E.I., Ewers, R.M., Foster, B.L., Jenkins, C.N., King, A.J., Laurance, W.F., Levey, D.J., Margules, C.R., Melbourne, B.A., Nicholls, A.O., Orrock, J.L., Song, D.X., Townshend, J.R.: Habitat fragmentation and its lasting impact on Earth's ecosystems. *Science Advances* **1**(2) (March 2015) e1500052
4. Prendergast, J.R., Quinn, R.M., Lawton, J.H., Eversham, B.C., Gibbons, D.W.: Rare species, the coincidence of diversity hotspots and conservation strategies. *Nature* **365**(6444) (September 1993) 365335a0
5. Sarkar, S.: Environmental philosophy: From theory to practice. *Studies in History and Philosophy of Science Part C: Studies in History and Philosophy of Biological and Biomedical Sciences* **45** (January 2013)
6. Pressey, R.L., Humphries, C.J., Margules, C.R., Vane-Wright, R.I., Williams, P.H.: Beyond opportunism: Key principles for systematic reserve selection. *Trends in Ecology & Evolution* **8**(4) (April 1993) 124–128
7. ReVelle, C.S., Williams, J.C., Boland, J.J.: Counterpart Models in Facility Location Science and Reserve Selection Science. *Environmental Modeling & Assessment* **7**(2) (June 2002) 71–80
8. Billionnet, A.: Solving the probabilistic reserve selection problem. *Ecological modelling*. **222** (2011) 546–554
9. Watts, M.E., Ball, I.R., Stewart, R.S., Klein, C.J., Wilson, K., Steinback, C., Lourival, R., Kircher, L., Possingham, H.P.: Marxan with Zones: Software for optimal conservation based land- and sea-use zoning. *Environmental Modelling & Software* **24**(12) (December 2009) 1513–1521
10. Diamond, J.M.: The island dilemma: Lessons of modern biogeographic studies for the design of natural reserves. *Biological Conservation* **7**(2) (February 1975) 129–146
11. Williams, J.C., ReVelle, C.S., Levin, S.A.: Spatial attributes and reserve design models: A review. *Environmental Modeling & Assessment* **10**(3) (September 2005) 163–181
12. Billionnet, A.: Designing Connected and Compact Nature Reserves. *Environmental Modeling & Assessment* **21**(2) (April 2016) 211–219
13. Dilkina, B., Houtman, R., Gomes, C.P., Montgomery, C.A., McKelvey, K.S., Kendall, K., Graves, T.A., Bernstein, R., Schwartz, M.K.: Trade-offs and efficiencies in optimal budget-constrained multispecies corridor networks. *Conservation Biology* **31**(1) (February 2017) 192–202
14. Jafari, N., Nuse, B.L., Moore, C.T., Dilkina, B., Hepinstall-Cymerman, J.: Achieving full connectivity of sites in the multiperiod reserve network design problem. *Computers & Operations Research* **81** (May 2017) 119–127
15. Rodrigues, A.S., Cerdeira, J.O., Gaston, K.J.: Flexibility, efficiency, and accountability: Adapting reserve selection algorithms to more complex conservation problems. *Ecography* **23**(5) (October 2000) 565–574

16. Prud'homme, C., Fages, J.G., Lorca, X.: Choco Documentation. (2017)
17. Sahr, K., White, D., Kimerling, A.J.: Geodesic Discrete Global Grid Systems. *Cartography and Geographic Information Science* **30**(2) (January 2003) 121–134
18. Birch, C.P.D., Oom, S.P., Beecham, J.A.: Rectangular and hexagonal grids used for observation, experiment and simulation in ecology. *Ecological Modelling* **206**(3) (August 2007) 347–359
19. Guisan, A., Zimmermann, N.E.: Predictive habitat distribution models in ecology. *Ecological Modelling* **135**(2) (December 2000) 147–186
20. Elith, J., Leathwick, J.R.: Species Distribution Models: Ecological Explanation and Prediction Across Space and Time. *Annual Review of Ecology, Evolution, and Systematics* **40**(1) (2009) 677–697
21. Etienne, R.S., Heesterbeek, J.A.: On optimal size and number of reserves for metapopulation persistence. *Journal of Theoretical Biology* **203**(1) (March 2000) 33–50
22. Dooms, G.: The CP(Graph) Computation Domain in Constraint Programming. PhD thesis, UCL - Université Catholique de Louvain (2006)
23. Beldiceanu, N., Carlsson, M., Rampon, J.X., Truchet, C.: Graph Invariants as Necessary Conditions for Global Constraints. Swedish Institute of Computer Science (2005)
24. Beldiceanu, N., Carlsson, M., Demassey, S., Petit, T.: Graph Properties Based Filtering. In: *Principles and Practice of Constraint Programming - CP 2006*. Lecture Notes in Computer Science, Springer, Berlin, Heidelberg (September 2006) 59–74
25. Beldiceanu, N., Carlsson, M., Rampon, J.X.: Global Constraint Catalog, 2nd Edition (Revision A). Swedish Institute of Computer Science (2012)
26. Fages, J.G., Prud'homme, C., Lorca, X.: Choco Graph Documentation (February 2018)
27. Bockmayr, A., Pisaruk, N., Aggoun, A.: Network Flow Problems in Constraint Programming. In: *Principles and Practice of Constraint Programming — CP 2001*. Lecture Notes in Computer Science, Springer, Berlin, Heidelberg (November 2001) 196–210
28. Ibanez, T., Hequet, V., Chambrey, C., Jaffré, T., Birnbaum, P.: How does forest fragmentation affect tree communities? A critical case study in the biodiversity hotspot of New Caledonia. *Landscape Ecology* **32**(8) (August 2017) 1671–1687
29. Pouteau, R., Bayle, É., Blanchard, É., Birnbaum, P., Cassan, J.J., Hequet, V., Ibanez, T., Vandrot, H.: Accounting for the indirect area effect in stacked species distribution models to map species richness in a montane biodiversity hotspot. *Diversity and Distributions* **21**(11) (2015) 1329–1338
30. Schmitt, S., Pouteau, R., Justeau, D., Boissieu, F., Birnbaum, P.: Ssdm: An r package to predict distribution of species richness and composition based on stacked species distribution models. *Methods in Ecology and Evolution* **8**(12) (August 2017) 1795–1803
31. Steiger, R., van Hoeve, W.J., Szymanek, R.: An Efficient Generic Network Flow Constraint. In: *Proceedings of the 2011 ACM Symposium on Applied Computing*. SAC '11, New York, NY, USA, ACM (2011) 893–900
32. Downing, N., Feydy, T., Stuckey, P.J.: Explaining Flow-Based Propagation. In: *Integration of AI and OR Techniques in Constraint Programming for Combinatorial Optimization Problems*. Lecture Notes in Computer Science, Springer, Berlin, Heidelberg (May 2012) 146–162