Minimizing Landscape Resistance for Habitat Conservation

Diego de Uña[1], Graeme Gange[1], Peter Schachte[1], and Peter J. Stuckey[1,2]
{d.deunagomez@student.,gkgange@,schachte@,pstuckey@}unimelb.edu.au

[1] Department of Computing and Information Systems,
The University of Melbourne, Melbourne, Australia
[2] Data 61, CSIRO, Melbourne, Australia

Abstract. Modeling ecological connectivity is an area of increasing interest amongst biologists and conservation agencies. In the past few years, different modeling approaches have been used by experts in the field to understand the state of wildlife distribution. One of these approaches is based on modeling land as a resistive network. The analysis of electric current in such networks allows biologists to understand how random walkers (animals) move across the landscape. In this paper we present a MIP model and a Local Search approach to tackle the problem of minimizing the effective resistance in an electrical network. This is then mapped onto landscapes in order to decide which areas need restoration to facilitate the movement of wildlife.

1 Introduction

In the past decades, the natural habitat of different species across the globe have become disrupted and fragmented. This is considered to be a major threat to the conservation of biodiversity by biologists and conservation experts [24]. As pointed out by Rosenberg *et al.* [24] and Pimm *et al.* [22] among other experts, the isolation of groups of animals can easily lead to extinction. For this reason, restoration needs to be undertaken in order to maintain a suitable environment where wildlife can move, feed and breed. Landscape restoration is part of Computational Sustainability, which has been increasingly attractive for researchers in our community as it presents interesting computational problems that happen to be NP-hard [10, 14, 17, 26, 27].

In ecology, patches of landscape are characterized by their *resistance*. This corresponds to how hard it is for wildlife to traverse a land patch. For instance, a low resistance value might be caused by soil of good quality that allows plants to spread more easily, or an open field might have high resistance for small rodents (that may be targets for birds of prey). Of course, this measure depends on the species being studied: some animals may be able to cross rivers more easily than others. Lower resistance means the land patch is more suitable for the species studied. Therefore, a suitable environment for wildlife would be one where their core habitats are connected by low resistance patches, so that animals can freely travel between core habitats to breed and feed.

To improve landscape connectivity, ecologists have used the idea of *corridors* [7, 15]. A corridor connects core habitats of animals using uninterrupted paths through which the animals can move. A common measure of the quality of a corridor is know as the Least-Cost Corridor (LCC) [1, 6]. LCCs are chosen to minimize the total sum of the resistance in the corridor. There has been extensive work in the Constraint Programming, AI and OR communities in helping identify the best corridors to be built (e.g. [10, 17, 27]).

Other work has addressed the problem of habitat conservation without enforcing connectivity. For instance, Crossman *et al.* [9] wrote a MIP model for habitat conservation. In their case, the intention was to minimize the number of sites to be restored while keeping a desired area for the animals and maintaining safe distances to roads and other dangers.

Although LCC is a valid model that is broadly used, it has been criticized for over simplifying the actual movement of species [20]. McRae [19, 21] proposed the use of electric circuit theory to measure the total connectivity of a landscape. This model is called Isolation By Resistance (IBR). In particular, it was shown [20] how the IBR model better matches the empirical evidence of genetic distance amongst a distributed species measured with two standard statistical models (fixation index and a step-wise mutation model). The IBR model has since been used to study the effects of habitat loss in birds [3, 4], for instance.

In the IBR model, the land patches are modeled as nodes in an electric circuit. The transition between contiguous patches is modeled by a branch of the electric circuit carrying a resistor. The resistance of the resistor in Ohms gives the resistance of moving between adjacent patches. The circuit is then connected to a 1A current source in a core habitat, while another core habitat is connected to the ground (0V). The *effective resistance* between two habitats in the electric circuit is the measure used for connectivity in the IBR model. It physically corresponds to the real resistance between two points of the circuit. An example of such circuit with 16 land patches (i.e. 16 nodes) can be seen on the left of Figure 2. Experts use the tool Circuitscape [25] for this model. Its task is to compute the currents, voltages and effective resistance that are then viewable by experts in geographic visualization software. Nevertheless, Circuitscape does not make conservation decisions, it only builds a linear system and solves it for the experts.

The model is justified by the fact that the commute time between two nodes s and t is given by $2mR_{st}$ where m is the number of edges in the graph, and R_{st} is the effective resistance between s and t in the underlying electrical network (Theorem 4.1 in [18]). The goal is therefore to lower the effective resistance between core habitats, as this directly translates into decreasing the commute time between habitats. Previous work by Doyle and Snell [12] also proved this property, and gave an interpretation of current i_{xy} through a branch (x, y) of the underlying electrical network. The current gives the net number of times a random walker would walk from x to y using that branch. This is exactly what is used by biologists to detect areas where wildlife concentrates most in the IBR model [21]. As an example, consider Figure 1. On the right side the landscape is plotted showing the conductance values in Ω^{-1} (inverse of resistance). On the left, a heatmap of the current at each patch of land. We can observe that, when moving out of their habitats, animals tend to walk using high conductance areas: indeed the $[0.2, 1]$ areas of the heatmap tend to coincide with high conductance areas on the conductance map.

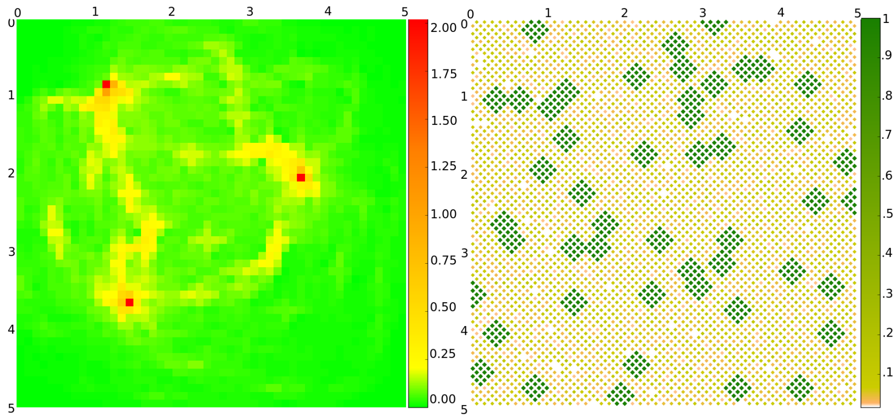

Fig. 1. Example of landscape. Left: The heatmap shows the current (in Amperes) in the electric circuit. The three darkest/red patches correspond to core habitats. This model predicts that animals will walk on the high current areas ($\gtrapprox 0.2A$, yellow/orange) often and less often in low current ($\lessapprox 0.2A$, green) areas. Right: Map of conductance (in Ω^{-1}). Darker/green areas are land where the resistance is low and easy for animals to move across.

For habitat and ecologic planning using the Isolation By Resistance model, we are interested in minimizing the effective resistance between habitats. To do so, biologists need to decide where improving the habitat is more beneficial. An improvement could mean planning for reforestation, building highway bridges for animals or improving the quality of the soil, among other actions. Our goal here will be to choose the right spots for these investments, subject to a budget. Note how the technical term *effective resistance* is often substituted by *resistance distance* in the biology literature. Also, in a non-reactive circuit with no alternating current, effective resistance and equivalent resistance are equal. For consistency we will always refer to effective resistance.

A similar problem was addressed by Ghosh *et al.* [13], but their problem was continuous. The continuous variant does not apply in our habitat conservation planning, as it is not possible to invest an infinitesimal fraction of budget in one location. In the real world, reforestation areas are typically well defined discrete interventions.

Section 2 is a brief summary on some basic electric circuit theory that we use later on. In Section 3 we discuss our Mixed-Integer Programming (MIP) model in detail. Section 4 discusses the problems of our model and presents our Local Search (LS) approach. Lastly, Section 5 presents our experimental results.

2 Preliminaries

The effective resistance corresponds to the real resistance between two nodes in the circuit. During the rest of this paper we will note R_{xy} the effective resistance between x and y.

In electric circuits, the effective resistance can be computed by solving a system of linear equations [2, 11]. Such linear system can be obtained by doing nodal analysis or mesh analysis. A more systematic way of obtaining the same system of linear equations is by using the nodal admittance matrix [5], also known by graph theoreticians as the Laplacian matrix. Let $adj(n)$ be the set of nodes adjacent to a node n in an electric circuit, and g_{ij} the conductance of branch (i, j). For an electric circuit of n nodes, the Laplacian is an $n \times n$ matrix defined as follows:

$$L_{i,j} = \begin{cases} -g_{ij} & \text{if } (i,j) \text{ is a branch of the circuit} \\ \sum_{k \in adj(i)} g_{ik} & \text{if i = j} \\ 0 & \text{otherwise} \end{cases}$$

For any two nodes s and t in an electric circuit of Laplacian L, we can calculate the effective resistance R_{st} between those nodes by simply solving the linear system given by $L_t v = f$ where L_t is L without the t^{th} row and column, and f is a vector with all its elements equal to 0 except for the s^{th}, which is equal to 1. The effective resistance we are after will be found in v_s.

This is strictly equivalent to nodal analysis. In classic nodal analysis, we would connect a source of 1A to s and connect t to the ground. Then we obtain an equation for each node of the form $\sum_{y \in adj(x)} i_{xy} = 0$ (flow conservation of the current), except for the node s where the flow of current is 1 since it's connected to the source, and the node t where the flow is -1 for being connected to the ground. Ohm's law gives us that, for any branch, $i_{xy} = g_{xy}(v_x - v_y)$ where v_x and v_y are the voltages at nodes x and y and g_{xy} is the conductance of the resistor between x and y. Then any node can be selected as *reference* or *datum* node, setting its voltage to 0. We select t as the datum node. Then by substituting with Ohm's law and the value of the datum node in the nodal analysis equations we obtain the same system as with the Laplacian. Solving the system yields the voltage of node s, among others. Since the source of current was 1A, Ohm's law gives us that $R_{st} = (v_s - v_t)/1 = v_s$. It is therefore only necessary to look at the s^{th} component of the solution vector to obtain the effective resistance. This is exactly the method implemented in Circuitscape [25], software used nowadays for landscape planning by experts.

Note how the definition of effective resistance only allows the computation of one effective resistance at a time: the current source needs to be connected to exactly one node, s, and only one node, t, must be connected to the ground. Therefore, the effective resistance between three or more nodes is undefined. Instead, we will consider the total effective resistance for three nodes a, b and c to be $R_{ab} + R_{ac} + R_{bc}$. These have to be computed by solving three different equation systems. In general, for f nodes, we will need $f(f-1)/2$ linear systems.

3 Problem Formulation

We define the problem of finding the Minimum Effective Resistance in a Circuit with Binary Investments (MERCBI) as follows.

We are given an electric network $G^{\emptyset} = (N, E, g^{\emptyset})$ where N are the nodes, E are the edges (i.e. *branches* in circuit theory) and $g^{\emptyset} : E \mapsto \mathbb{R}^+$ is a function giving the value of the original conductances that are placed on each edge. We are also given a function $g^E : E \mapsto \mathbb{R}^+$ of new improved conductances for each edge. These functions are such that $\forall e \in E, g^{\emptyset}(e) \leq g^E(e)$. Lastly, we are given a set of pairs of *focal* nodes (i.e. core habitats) $P = \{\langle s_1, t_1 \rangle, ..., \langle s_{|F|}, t_{|F|} \rangle\}$, a budget B, and a cost function $c : E \mapsto \mathbb{R}^+$. The pairs in P are the pairs of core habitats between which we want to improve the resistance. It typically contains all the possible pairs of habitats in the studied landscape, but not necessarily.

We note G^A for $A \subseteq E$ the network given by $G^A = (N, E, g^A)$ where g^A is a function defined by $g^{\emptyset}$ for all $e \notin A$ and g^E for all $e \in A$. Similarly, R^A_{xy} will refer to the effective resistance between x and y in G^A.

Our goal is to find a set $S \subseteq E$ such that we minimize $R^S = \sum_{i=1}^{|F|} R^S_{s_i t_i}$ while keeping the sum of the investment costs below B. We say that the edges in S are *investments*. The edges in $E \backslash S$ are *wild* edges. Note that it may be the case that g^S forms an open circuit (i.e. there is no way for current to go from some s_i to some t_i due to 0-conductance branches). In such case, $R^S = \infty$ by definition, which can happen if there is not enough budget to ensure connectivity.

Formally, the model is translated into a Mixed Integer Program with the following additional variables:

- b_e is a binary decision on whether the edge e of the circuit is part of the selected solution S.
- $v_x^{\langle s,t \rangle}$ is the voltage at node x when s is connected to a 1A source and t to the ground.
- $p^{\langle s,t \rangle}_{(a,b),c}$ is an intermediate variable for the product $g^S((a,b)) * v_c^{\langle s,t \rangle}$.

$$\text{Minimize} \quad \sum_{i=1}^{|F|} v_{s_i}^{\langle s_i, t_i \rangle} \tag{1}$$

$$\text{s.t.} \quad \sum_{e=0}^{e=|E|} b_e c(e) \leq B \tag{2}$$

$$\forall \langle s_i, t_i \rangle \in P, \quad \forall x \in N \backslash \{s_i, t_i\}, \sum_{y \in adj(x)} p^{\langle s_i, t_i \rangle}_{(x,y),x} - \sum_{y \in adj(x) \backslash \{t_i\}} p^{\langle s_i, t_i \rangle}_{(x,y),y} = 0 \tag{3}$$

$$\forall \langle s_i, t_i \rangle \in P, \quad \sum_{y \in adj(s_i)} p^{\langle s_i, t_i \rangle}_{(s_i,y),s_i} - \sum_{y \in adj(s_i) \backslash \{t_i\}} p^{\langle s_i, t_i \rangle}_{(s_i,y),y} = 1 \tag{4}$$

$$\forall \langle s_i, t_i \rangle \in P, \quad \forall e = (x,y) \in E, \forall z \in \{x, y\},$$
$$p^{\langle s_i, t_i \rangle}_{e,z} = b_e g^E(e) v_z^{\langle s_i, t_i \rangle} + (1 - b_e) g^{\emptyset}(e) v_z^{\langle s_i, t_i \rangle} \tag{5}$$

Equation 1 is our objective: minimize the sum of effective resistances between focal nodes. Equation 2 is our budget constraint. Equation 3 constrains the flow of current to be 0 in all nodes, except the nodes directly connected to the source. Equation 4 indicates that the flow at the source nodes has to be 1. Note how the equations at the sinks with flow -1 have been removed as they are linearly dependant from the others. Equation 5 chooses the value of the p variables based on the values of the Booleans.

In Eqs. 3 and 4, the first sum correspond to the diagonal terms in the Laplacian matrix, whereas the second sum is the non-diagonal terms of the Laplacian matrix.

Furthermore, Eqs. 3, 4 and 5 are repeated for each pair of focal nodes in P. Therefore, the equations obtained by nodal analysis are repeated $C = |P|$ times, one per pair in P. Nonetheless, these C systems are not equivalent, as the source of 1A is connected at different nodes, and the voltages are therefore different in each system. That is, it is not necessarily the case that $v_x^{\langle s_i,t_i \rangle} = v_x^{\langle s_j,t_j \rangle}$, $\forall i, j, x, i \neq j$. We say that our model contains C circuits.

3.1 Complexity

We now prove that the MERCBI problem is NP-hard by reduction from the Steiner Tree Problem on graphs (STP). The STP, as formulated by Karp [16] in his paper where he proved its NP-completeness, is: Given a graph $G = (N, E)$, a set $R \subseteq N$, weighting function w on the edges and positive integer K, is there a subtree of G weight $\leq K$ containing the set of nodes in R?

We apply the following reduction:

- The electric circuit is the graph G.
- The cost function is w.
- The original resistance of edges is infinite (i.e. $g^{\emptyset}(e) = 0, \forall e \in E$).
- The resistance upon investment of all edges is 1 (i.e. $g^{E}(e) = 1, \forall e \in E$).
- The budget is K.
- The set of pairs of focal nodes P is the set of all pairs of distinct nodes in R, which can be built in $\mathcal{O}(|R|^2)$.

Assume we have an algorithm to solve the MERCBI problem that gives a solution S. Clearly, by investing in the selected edges S we will obtain a resistance $R^S \in \mathbb{R}$ iff there is enough budget K, and $R^S = \infty$ otherwise:

1. If the resistance is ∞, then there is no Steiner Tree of cost less than K, since we could not connect focal nodes with 1Ω resistors.
2. If the resistance is 0 then we can obtain a graph G^* by restricting G^S to the edges that have been invested. Clearly, G^* is of cost $\leq K$ and a subgraph of G that connects all pairs in P. Because P is the set of all pairs of nodes from R, this means that all nodes in R are connected pairwise in G^*. Although G^* may not be a tree, we can extract a tree T from it by breaking any cycle G^* contains while maintaining its connectivity. The tree T is a Steiner tree of cost at most K.

We have therefore shown that if we have a solver for the MERCBI problem, we have one for the STP. Thus, the MERCBI problem is NP-hard.

4 Solving Approach

4.1 Greedy algorithm

We first devise a greedy algorithm for the problem based on the following observation: increasing the conductance of an edge with low current has little impact in the overall effective resistance. This intuition is easily justifiable: if the current of an edge corresponds to the net number of times a random walker uses that edge, the lower that number, the less impact improving that area would have on the total commute time. Clearly edges near focal nodes or near low resistance areas tend to concentrate more current and be used by more random walkers. Thus increasing the conductance of those edges will likely have a stronger impact in lowering the effective resistance. Our greedy algorithm is presented in Alg. 1.

Algorithm 1 Greedy algorithm

procedure $\text{GREEDY}(\mathcal{G} = (N, E), F, g^{\emptyset}, g^{E}, c, B)$
    $lp \leftarrow \text{BUILDMODEL}(G, F, g^{\emptyset}, g^{E}, c, B)$ ▷ Build the model from Section 3
    $lp.setBinariesFalse(E)$ ▷ Sets all the binaries to false
    $lp.solve()$ ▷ Solve the LP, with all binary variables fixed
    $i[e] \leftarrow 0, \quad \forall e \in E$ ▷ Array of currents
    for all $(s_i, t_i) \in P$ **do**
        for all $e = (x, y) \in E$ **do**
            $i[e] \leftarrow i[e] + |g^{\emptyset}(e) * (v_x^{\langle s_i,t_i\rangle} - v_y^{\langle s_i,t_i\rangle})|$ ▷ Ohm's law
    $se \leftarrow \text{REVERSE}(\text{SORTBY}(E, i))$ ▷ Array of edges sorted by decreasing current
    $S \leftarrow \emptyset; b \leftarrow 0; j \leftarrow 0$
    while $b < B \wedge j < |E|$ **do**
        if $g^{\emptyset}(se[j]) < g^{E}(se[j]) \wedge b + c(se[j]) \leq B$ **then**
            $S \leftarrow S \cup se[j];\ b \leftarrow b + c(se[j])$ ▷ Select edges with high current.
        $j \leftarrow j + 1$
    return S

The algorithm solves the conductance LP once to find the voltages at each node, then calculates the currents in each edge (with no investments being made), and greedily selects the edges with most current to invest in, that fit within the budget.

As we will see in the experiments, this algorithm performs surprisingly well despite not providing optimal solutions. In next sections we will try to obtain better solutions than the ones provided by this algorithm.

4.2 Performance of the Pure MIP Model

The MIP model expressed in Section 3 is a direct mapping of the nodal analysis performed in the electric circuit into a MIP model. In our implementation equation 5 was split into two *indicator constraints* which are supported by the IBM ILOG CPLEX 12.4. solver as follows:

$\forall \langle s_i, t_i \rangle \in P, \quad \forall e = (x, y) \in E, \forall z \in \{x, y\},$

$$b_e \implies p_{e,z}^{\langle s_i,t_i\rangle} = g^{E}(e) v_z^{\langle s_i,t_i\rangle} \tag{6}$$

$$\neg b_e \implies p_{e,z}^{\langle s_i,t_i\rangle} = g^{\emptyset}(e) v_z^{\langle s_i,t_i\rangle} \tag{7}$$

Finding Bounds for the Variables To help CPLEX tackle the problem, we need to compute bounds on the variables. We apply basic circuit analysis to find these bounds. To do so, we need an initial assignment for the binary variables: this could be assigning all to false, or to the value of some initial solution that respects the budget constraint (either a random solution, or obtained with Alg. 1). Without loss of generality, let us assume that all the binaries are set to false, thus $g^{\emptyset}$ gives the conductance for all edges.

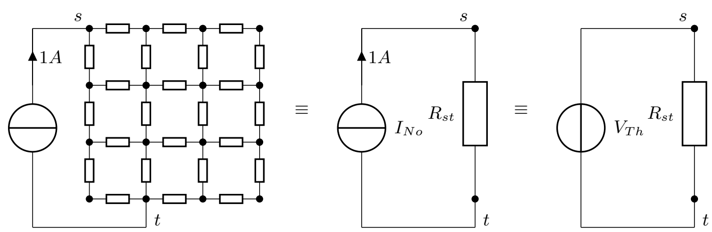

Fig. 2. Conversion from the circuit built for calculating effective resistance to an equivalent circuit with one resistor, and application of Thévenin's theorem.

When we are computing the effective resistance between nodes s and t of a circuit, we connect a current source between s and t as in Fig. 2. This can be converted into an equivalent circuit as seen in the center of the figure. The value of R_{st} is actually the value of the effective resistance when all our binary variables are fixed. To obtain bounds for the voltages at the nodes marked $\bullet$, we observe that the circuit in the center is a Norton circuit, thus we can apply Thévenin's theorem [8] to it to obtain an equivalent circuit with a voltage source instead of a current source (circuit on the right). Because the current in that Norton's circuit is 1A, the voltage of Thevenin's equivalent source will be $V_{Th} = I_{No}R_{st} = R_{st}$. Thus, the upper bound of v_s is R_{st}. Since t is connected to the ground, $v_t = 0V$. All the voltage at points between s and t are necessarily bounded by the voltages at these points, because voltage can only drop. Therefore, the bounds for all voltage variables are: $\forall \langle s_i, t_i \rangle \in P, \forall x \in N, \quad 0 \leq v_x^{\langle s_i,t_i \rangle} \leq R^{\emptyset}_{s_i t_i}$. From these bounds it is easy to derive bounds for the p variables: $\forall \langle s_i, t_i \rangle \in P, \quad \forall e = (x, y) \in E, \forall z \in \{x, y\}, \quad g^{\emptyset}(e) v_z^{\langle s_i,t_i \rangle} \leq p_{e,z}^{\langle s_i,t_i \rangle} \leq g^E(e) v_z^{\langle s_i,t_i \rangle}$.

Motivation for Local Search Approaches We attempted to solve the MIP model using IBM ILOG CPLEX 12.4. The major challenge we discovered while running our experiments was that the linear relaxation of the model performed poorly. The gap between the LP relaxation and the best integer solution found in less than 5 hours was usually bigger than 40% even for small 10×10 grid-like networks, as reported by CPLEX. As a small example, consider a 3×3 grid with $\forall e, \ g^{\emptyset}(e) = 0.01$ and $g^E(e) = 1$, a budget of 2 and 2 core habitats. The proven optimal solution gives a resistance of 49.15, whereas the linear relaxation achieves a resistance as low as 0.50. With such a big gap, the linear relaxation cannot help pruning branches. We observed these gaps both when using indicator constraints and without them (using an alternative encoding of the product of continuous and binary variables as linear inequalities).

Since the linear relaxation is so weak, we decided to try a Local Search approach.

4.3 Local Search

Although solving the MIP problem is time consuming, once all binary variables are fixed, the problem becomes a Linear Program, which can be solved efficiently. The idea of local search is to repeatedly destroy part of the solution and build a new one from the remaining solution.

Our Local Search (LS) moves are based on two stages: first choose areas where we invested but we are no longer interested in investing, then choose new areas where we would like to invest. In this section we will discuss different techniques we use to make moves in the LS.

Destroying Investments We start from a solution s where we have selected a set S of edges to be investments. We call d the *destruction rate* of investments.

To destroy investments, we select a subset S_d of S such that $|S_d| = d$. We will convert these d investments into wild edges, thus freeing part of the budget to be used elsewhere. We implemented 3 ways of selecting those d edges:

1. InvRand: Choose d invested edges randomly.
2. InvLC: Choose the d invested edges with lowest current in s.
3. InvLCP: Choose d invested edges based on a probability distribution that favors low current edges in s being selected.

The way to compute the current of an edge is by simply using Ohm's law, as in line 8 of Alg. 1. Because we are solving C linear systems for C circuits at once, the current across an edge will be the sum of the currents of that edge in the C different circuits.

As for the probability distribution, the total current across an edge is proportional to the inverse of its probability. Therefore edges with low current will have a higher tendency to be chosen than edges with high current: $\forall e = (x, y) \in E, \quad Pr(e) \propto \big(\sum_{\langle s_i,t_i\rangle \in P} |g^S(e) * (v_x^{\langle s_i,t_i\rangle} - v_y^{\langle s_i,t_i\rangle})|\big)^{-1}$.

Making New Investments Let s' be the solution s with investments $S \backslash S_d$, that is all investments of s except those selected for destruction. After destroying investments S_d, we have gotten part $b = \Sigma_{e \in S_d} c(e)$ of our budget back, so we are ready to choose new places to invest. To do so, we have 4 different strategies to choose new edges to invest in:

1. WilRand: Choose a set W of wild edges randomly.
2. WilBFS: Choose a set W of wild edges by doing a Breath-First Search (BFS) in the graph. The origin of the BFS is chosen according to a probability distribution that favors high current nodes. This ensures that the chosen edges are close to each other.

3. WilHC: Choose a set W of wild edges that have the highest current in s.
4. WilHCP: Choose a set W of wild edges based on a probability distribution that favors high-current wild edges.

For all these strategies, we ensure that $\sum_{e \in W} c(e) = b - \epsilon$ and we select those edges greedily to get the smallest slack ϵ.

The second strategy, WilBFS, requires computing the current at a node. We define it as $c_x = \sum_{(s_i,t_i) \in P} \frac{1}{2} \sum_{y \in adj(x)} |g^S((x,y)) * (v_x^{\langle s_i,t_i \rangle} - v_y^{\langle s_i,t_i \rangle})|$. That is, the sum of the accumulated current of the surrounding edges of the node.

As for the probability distributions used by both WilBFS and WilHCP, we could not use a distribution as simple as for InvLCP. The reason is that, even though the probability of choosing a particular node (or edge) of high current is much higher than the probability of choosing particular node (or edge) of low current, because there are many more nodes and edges with low current, the probability of choosing a low current node or edge would be much higher.
To overcome this, we will only allow the choice of nodes or edges that have a current higher than 10% of the current of the highest node or edge, plus a certain number γ of low current elements (i.e. below that 10%). We choose γ to be equal to the number of nodes or edges above 10% of the maximum current. The probability of choosing an element with a current below 10% of the maximum is now the same as the probability of choosing an element with a current above that threshold.

Initial solution Local Search needs to start with an initial solution. Our approach was to take the output of the greedy algorithm as the initial solution for the LS.

Using Simulated Annealing As it is well known, Local Search can get stuck into a local minimum if we are only accepting improving solutions. Therefore, we implement the simulated annealing approach introduced by Ropke *et al.* for the Pickup and Delivery Problem with Time Windows [23].

Here the idea is that we will always have 3 solutions at hand: s_{best} (for a set of invested edges S_{best}), s, and s_{new} (for S_{new}). The first is the solution found so far with lowest resistance. The second is the current solution maintained in the search (called *accepted* in [23]). The last, is the new solution which we obtained by modifying the solution s using the destruction and investment phases described earlier. If we accept a solution s_{new} only when it is better than the current solution we might get trapped in a local minimum. With simulated annealing, we accept a solution s_{new} with probability $e^{-(s_{new}-s)/T}$, where T is the temperature. The temperature decreases at each iteration by a factor called the *cooling rate*: $T_{i+1} = c_r * T_i$. This allows us to accept solutions that are worse than our currently accepted solution, thus allowing us to get out of local minima.

Initially we will have a higher tendency to accept worsening solutions, and as the temperature drops slowly over the iterations, we will only accept improving solutions. We discuss the value of the initial temperature in Section 5.

Even using simulated annealing, it is easy to see how the combined usage of InvLC and WilHC can perform the exact same destruction over and over once it hits a local minimum. To avoid that, when we are using these two destruction techniques together and we don't accept a solution, we switch to WilHCP for one iteration, which will put us in another neighborhood to explore (like a restart).

5 Experiments

All the following experiments use IBM ILOG CPLEX 12.4 on an Intel® Core™ i7-4770 CPU machine @ 3.40GHz with 15.6GB of RAM running Linux 3.16.

5.1 Instances

We could not obtain real-world instances. Nonetheless we generated artificial landscapes while trying to keep them realistic. In papers like [3, 4] where the IBR model is used on empirical data, high resistance areas have an approximate resistance between 10 and 100 times bigger than areas with low resistance.

Our instances were built in the form of grids with 4 neighbor nodes (except at the border of the landscape where nodes have 2 or 3 neighbors). The values of conductances (i.e. inverse of resistances) are chosen following a probabilistic beta-distribution with parameters $\alpha = 5$ and $\beta = 80$.

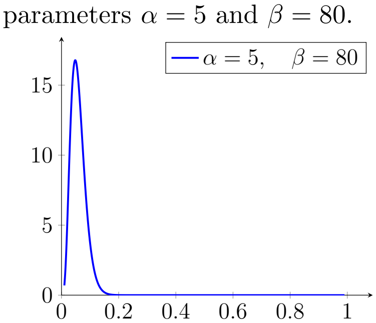

Fig. 3. Plot of Beta Distribution for $\alpha = 5, \beta = 80$.

Using this distribution, most conductances are in the desired range, still allowing the unlikely possibility of having some high conductances scattered on the map. Then, for each instance of size $n \times n$ we select n nodes, and their neighbors, to be *oases*. All the edges coming out of oasis nodes have a conductance of 1, that is the same as if we had invested. This is to account for areas that are not considered core habitats but are friendly to the animals: wild trees, fresh water lakes, etc. Figure 1 shows, on the right, an example of such map in a 50×50 grid (the scale being in Ω^{-1}).

Regarding the budget cost, we created instances with homogeneous costs (labelled Homo) where each edge costs 1. The reason for homogeneous costs is that the main application for our problem is in wild land and specifically on National Parks, where rehabilitation costs are typically fairly uniform across a large area. Typically these kind of projects are used to recover land that has been damaged by bush fires, flooding or landslides or that has been fragmented by urbanization that cannot be torn down or purchased. We also constructed instances where the cost of an investments is a uniform random number between 0.1 and 10 (labeled Rand).

Furthermore, the locations of core habitats are selected from a uniform distribution. For each size of grid and number of habitats, we generated 20 different instances, totalling 500 instances. Values reported are arithmetic means.

5.2 Improvement Over the Initial Landscape

To measure the quality of a solution, we look at the ratio between the effective resistance in the solution and the original resistance when no investment is made. Table 1 shows the average of these ratios. All the results reported on that table had a budget equal to twice the side of a grid (e.g., for a grid of 50×50, we have a budget of 100). The ratio of the budget to the number of edges where we could invest is shown in the third column. The destruction rate here is fixed to 10 for instances of size 20×20 and 25×25, and 25 for bigger instances. The number of iterations the LS does for these tests is 200. We do not show the results obtained by random selections for destruction as they perform worse than the ones shown here. The initial temperature is chosen such that a solution 50% worse than the initial solution is accepted with a probability of 10%. The cooling rate is 0.98 for all experiments.

Table 1. Averages of the ratios of updated resistance over original resistance. Budgets are always twice the side of a grid.

Cost Type	Size	Budget Ratio	Habitats	Greedy	InvLC			InvLCP		
					WilBFS	WilHC	WilHCP	WilBFS	WilHC	WilHCP
Homo	20×20	0.11	2	0.30	**0.24**	0.25	0.25	0.29	0.26	0.27
Homo	25×25	0.09	2	0.30	**0.22**	0.23	0.24	0.28	0.24	0.26
Homo	25×25	0.09	3	0.38	**0.29**	0.30	**0.29**	0.37	0.31	0.35
Homo	30×30	0.07	2	0.37	**0.26**	0.27	0.28	0.35	0.29	0.31
Homo	30×30	0.07	3	0.42	**0.33**	**0.33**	**0.33**	0.42	0.35	0.41
Homo	50×50	0.04	2	0.44	**0.34**	0.35	0.36	0.44	0.37	0.40
Homo	50×50	0.04	3	0.48	0.42	**0.40**	0.41	0.48	0.43	0.48
Homo	50×50	0.04	4	0.52	0.46	**0.44**	0.45	0.52	0.47	0.52
Homo	100×100	0.02	2	0.49	0.39	**0.38**	0.41	0.42	0.39	0.37
Homo	100×100	0.02	3	0.53	**0.42**	0.43	**0.42**	0.53	**0.42**	0.50
Homo	100×100	0.02	4	0.55	0.49	0.47	**0.46**	0.55	0.49	0.55
Homo	100×100	0.02	5	0.58	0.49	**0.46**	0.50	0.58	0.48	0.58
Rand	20×20	0.11	2	0.44	**0.34**	0.34	0.38	0.39	0.35	0.40
Rand	25×25	0.09	2	0.45	0.33	**0.32**	0.36	0.39	0.33	0.38
Rand	25×25	0.09	3	0.53	0.41	0.39	**0.38**	0.47	0.40	0.44
Rand	30×30	0.07	2	0.52	0.45	**0.42**	0.44	0.51	0.45	0.48
Rand	30×30	0.07	3	0.57	0.53	**0.51**	0.52	0.57	0.52	0.56
Rand	50×50	0.04	2	0.56	0.54	**0.51**	0.53	0.56	0.53	0.56
Rand	50×50	0.04	3	0.60	0.59	**0.55**	0.58	0.60	0.58	0.60
Rand	50×50	0.04	4	0.63	0.62	**0.60**	0.63	0.63	0.62	0.63
Rand	100×100	0.02	2	0.61	**0.51**	0.53	0.52	0.66	0.61	0.65
Rand	100×100	0.02	3	0.64	0.59	**0.57**	0.60	0.63	0.60	0.64
Rand	100×100	0.02	4	0.65	**0.58**	0.60	0.62	0.62	0.61	0.64
Rand	100×100	0.02	5	0.68	0.60	**0.59**	0.62	0.63	0.62	0.65

Our first observation is that even with a very low budget, proportionally to the mass of land, we manage to have significant drops of resistance. We infer that our approach is placing the investments in the right places. The greedy algorithm performs very well too. The Local Search is able to reduce the resistance of the greedy algorithm by up to 11%.

We can observe how in general the InvLC approach to destroy bad investments outperforms InvLCP. Regarding the selection of new investment locations it seems there is no clear winner, although WilBFS and WilHC tend to choose investments in places nearby or directly connected, and that seems to give them the advantage.

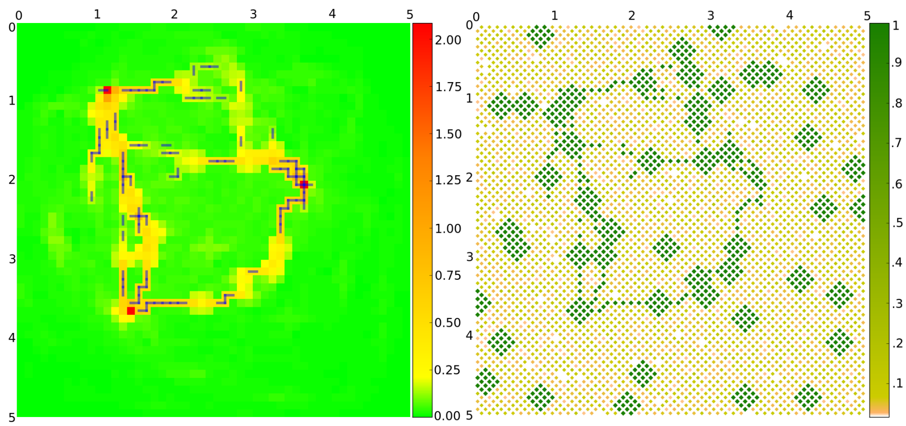

Fig. 4. Example of a solution obtained with InvLC+WilHC for the same instance as shown in Fig. 1.

As an example, Fig. 4 shows the solution found to the instance of Fig. 1 On the left, the heatmap shows the places of investment (marked lines) which redirect most of the animals, as these become more suitable for them. On the right, we see how the best investments tend to be aligned to connect low resistance areas that already existed. Notice how they do not force a complete corridor: some investments may not be used to maintain connectivity and are instead moved to areas where they might be more useful.

5.3 Optimality Gap

In this section we look at the difference between the solution obtained and the proven optimum for some instances. We could not obtain the optimal value for instances of the size given in the previous subsection. For this reason, we ran this experiment on 10×10 grids. These took, in some cases, up to 8 hours to prove optimality, even given the small size. It is clearly impractical to use the pure MIP formulation.

Table 2. Average (across 20 instances) of ratio between LS and proven optimum for 10×10 instances (200 iterations).

	InvLC			InvLCP		
	WilBFS	WilHC	WilHCP	WilBFS	WilHC	WilHCP
Average	1.06	1.07	1.07	1.11	**1.05**	1.07

As it can be seen in Table 2, the LS approach does not deviate much from the optimum. Once again we see no clear domination between the different techniques applied, although InvLCP+WilBFS seems slightly worse than others. The LS even managed to find optimal solution of two instances.

5.4 Number of Iterations

We now study the effect of the number of iterations. We are interested to know how quickly we get improvements in our solutions. To evaluate this, we looked at the 50×50 instances with homogeneous investment cost and a budget of 100 (same instances as in Table 1). We compute the ratio between the best solution found so far at the x^{th} iteration to the non-investment resistance. We average this ratio across the 60 Homo 50×50 instances (2, 3 and 4 habitats) for each iteration. The results are plotted in Figure 5. Based on the results seen in Table 1 and this plot, we can see that InvLCP+WilBFS and InvLCP+WilHCP are not only the ones that perform the worse, but we also only find better solutions very rarely. The InvLC+WilHC shows the quickest drop, but then slows down. Since the drop happens in the first 20 iterations, the temperature is still high, thus suggesting that it does not get trapped in a local minimum, instead it is most likely moving towards the global optimum (as we also saw on the optimality gap results). All other techniques have a slower slope.

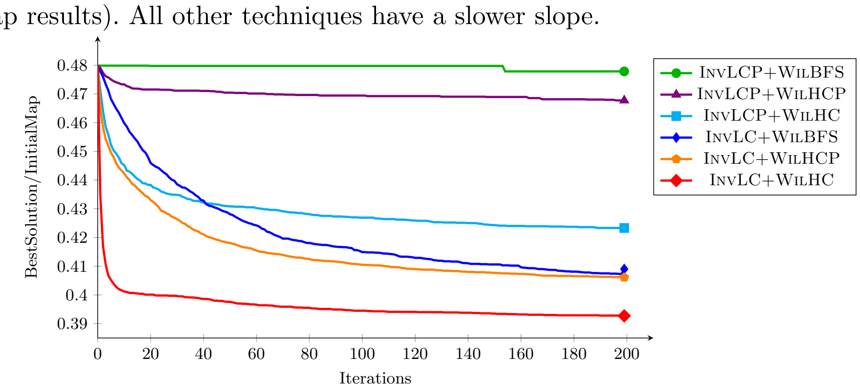

Fig. 5. Ratio between the best solution found and the value of the initial resistance over the number of iterations for 6 destruction strategies (50×50 Homo grids, 2 to 4 habitats).

Regarding the time spent to perform these 200 iterations: no instance needed more than 5 minutes in our machine. Therefore this number of iterations is also suitable for use by habitat planners during their work.

5.5 Destruction Rate

In this subsection we will look at the effects of the choice of destruction rate. Recall, the destruction rate corresponds to how many edges are converted from investments into wild edges for the next iteration of the LS. We ran these experiments on the 50×50 instances for 2, 3 and 4 habitats with a budget of 100 and homogeneous costs. Table 3 reports the average ratio between our solution and the original resistance across the 60 instances depending on the destruction rate. The number of iterations is once again 200.

Table 3. Ratio of solution found over initial landscape resistance for 50×50 grids with different destruction rates, over 200 iterations.

Destruction	InvLC			InvLCP		
Rate	WilBFS	WilHC	WilHCP	WilBFS	WilHC	WilHCP
5	0.40	**0.38**	0.40	0.47	0.39	0.45
10	0.40	**0.38**	0.39	0.48	0.39	0.46
15	0.40	**0.39**	0.39	0.48	0.40	0.46
20	0.40	**0.39**	0.40	0.48	0.41	0.47
25	0.41	**0.39**	0.41	0.48	0.42	0.47
30	0.42	**0.40**	0.41	0.48	0.43	0.47
35	0.43	**0.40**	0.42	0.48	0.44	0.47

As we can see in Table 3, the choice of destruction rate only marginally affects the quality of the solution. This shows that our local search approach is robust and no matter what parameter is used (within a reasonable range), the algorithm is likely to give similarly good solutions.

6 Conclusions and Future Work

In this paper we have introduced a new problem, the MERCBI problem, in habitat conservation. We have clearly defined it and given a MIP model. We have provided a series of heuristics used to implement a Local Search. The results show that this approach yields good quality solutions.

In future work we would like to investigate the idea of extending patches of land to more than a node: a golf course, or a farm could cover more than one node at a time and then we might consider purchasing part of them. Also, we could take into account distance for patches with different shapes where purchasing a thin part of it may be more interesting than purchasing the entire land. Also, we could look at having more than one species and take into account their predatory behavior or possibly favor endangered species.

Acknowledgements

We would like to thank Julian Di Stefano and Holly Sitters from the School of Ecosystems and Forest Sciences at the University of Melbourne as well as Nevil Amos from the Department of Environment, Land, Water and Planning of Victoria for meeting with us and introducing us to this problem.

References

1. Adriaensen, F., Chardon, J., De Blust, G., Swinnen, E., Villalba, S., Gulinck, H., Matthysen, E.: The application of least-costmodelling as a functional landscape model. Landscape and urban planning 64(4), 233–247 (2003)
2. Alexander, C.K., Sadiku, M.N.: Electric circuits (2000)
3. Amos, J.N., Bennett, A.F., Mac Nally, R., Newell, G., Pavlova, A., Radford, J.Q., Thomson, J.R., White, M., Sunnucks, P.: Predicting landscape-genetic consequences of habitat loss, fragmentation and mobility for multiple species of woodland birds. PLoS ONE 7, 1–12 (02 2012)
4. Amos, J.N., Harrisson, K.A., Radford, J.Q., White, M., Newell, G., Nally, R.M., Sunnucks, P., Pavlova, A.: Species- and sex-specific connectivity effects of habitat fragmentation in a suite of woodland birds. Ecology 95(6), 1556–1568 (2014)
5. Balabanian, N., Bickart, T.A., Seshu, S.: Electrical network theory. Wiley New York (1969)
6. Beier, P., Majka, D.R., Spencer, W.D.: Forks in the road: Choices in procedures for designing wildland linkages. Conservation Biology 22(4), 836–851 (2008), http://dx.doi.org/10.1111/j.1523-1739.2008.00942.x
7. Beier, P., Noss, R.F.: Do habitat corridors provide connectivity? Conservation Biology 12(6), 1241–1252 (1998), http://dx.doi.org/10.1111/j.1523-1739.1998.98036.x
8. Brittain, J.E.: Thevenin's theorem. IEEE Spectrum 27(3), 42– (March 1990)
9. Crossman, N.D., Bryan, B.A.: Systematic landscape restoration using integer programming. Biological Conservation 128(3), 369–383 (2006)
10. Dilkina, B., Gomes, C.P.: Solving Connected Subgraph Problems in Wildlife Conservation, pp. 102–116. Springer Berlin Heidelberg, Berlin, Heidelberg (2010)
11. Dorf, R.C., Svoboda, J.A.: Introduction to electric circuits. John Wiley & Sons (2010)
12. Doyle, P.G., Snell, J.L.: Random walks and electric networks. Mathematical Association of America, (1984)
13. Ghosh, A., Boyd, S., Saberi, A.: Minimizing effective resistance of a graph. SIAM review 50(1), 37–66 (2008)
14. Gomes, C.P.: Computational sustainability: Computational methods for a sustainable environment, economy, and society. The Bridge 39(4), 5–13 (2009)
15. Jon Aars, R.A.I.: The effect of habitat corridors on rates of transfer and interbreeding between vole demes. Ecology 80(5), 1648–1655 (1999), http://www.jstor.org/stable/176553
16. Karp, R.M.: Reducibility among combinatorial problems. In: Complexity of computer computations, pp. 85–103. Springer (1972)
17. LeBras, R., Dilkina, B.N., Xue, Y., Gomes, C.P., McKelvey, K.S., Schwartz, M.K., Montgomery, C.A., et al.: Robust network design for multispecies conservation. In: AAAI (2013)
18. Lovász, L.: Random walks on graphs. Combinatorics, Paul erdos is eighty 2, 1–46 (1993)
19. McRae, B.H.: Isolation by resistance. Evolution 60(8), 1551–1561 (2006), http://dx.doi.org/10.1111/j.0014-3820.2006.tb00500.x
20. McRae, B.H., Beier, P.: Circuit theory predicts gene flow in plant and animal populations. Proceedings of the National Academy of Sciences 104(50), 19885–19890 (2007), http://www.pnas.org/content/104/50/19885.abstract
21. McRae, B.H., Dickson, B.G., Keitt, T.H., Shah, V.B.: Using circuit theory to model connectivity in ecology, evolution, and conservation. Ecology 89(10), 2712–2724 (2008), http://dx.doi.org/10.1890/07-1861.1

22. Pimm, S.L., Jones, H.L., Diamond, J.: On the risk of extinction. The American Naturalist 132(6), 757–785 (1988), http://dx.doi.org/10.1086/284889
23. Ropke, S., Pisinger, D.: An adaptive large neighborhood search heuristic for the pickup and delivery problem with time windows. Transportation Science 40(4), 455–472 (2006), http://dx.doi.org/10.1287/trsc.1050.0135
24. Rosenberg, D.K., Noon, B.R., Meslow, E.C.: Biological corridors: Form, function, and efficacy. BioScience 47(10), 677–687 (1997), http://bioscience.oxfordjournals.org/content/47/10/677.short
25. Shah, V., McRae, B.: Circuitscape: a tool for landscape ecology. In: Proceedings of the 7th Python in Science Conference. vol. 7, pp. 62–66 (2008)
26. Urli, T., Brotánková, J., Kilby, P., Van Hentenryck, P.: Intelligent habitat restoration under uncertainty. In: Thirtieth AAAI Conference on Artificial Intelligence (2016)
27. Williams, J.C.: Delineating protected wildlife corridors with multi-objective programming. Environmental Modeling & Assessment 3(1-2), 77–86 (1998)