



HAL
open science

GECOT : Graph-based ecological connectivity optimization tool

François Hamonic, Yann Vaxès, Basile Couëtoux, Cécile H Albert

► To cite this version:

François Hamonic, Yann Vaxès, Basile Couëtoux, Cécile H Albert. GECOT: Graph-based ecological connectivity optimization tool. *Methods in Ecology and Evolution*, 2025, 16 (9), pp.1914-1922. <10.1111/2041-210X.70055>. <hal-05238813>

HAL Id: hal-05238813

<https://hal.science/hal-05238813v1>

Submitted on 4 Sep 2025

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.



Distributed under a Creative Commons CC BY 4.0 - Attribution - International License

APPLICATION

GECOT : Graph-based Ecological Connectivity Optimization Tool

François Hamonic¹ | Yann Vaxès² | Basile Couëtoux² | Cécile H. Albert¹¹Aix-Marseille Univ, CNRS, Univ Avignon, IRD, IMBE, Marseille, France²Aix-Marseille Univ, Université de Toulon, CNRS, LIS, Marseille, France**Correspondence**

François Hamonic

Email: francois.hamonic@gmail.com

Abstract

1. Habitat fragmentation is a critical threat to biodiversity, as it creates isolated patches of habitat and hinders species movement and survival. Systematic Conservation Planning (SCP) has advanced to include landscape connectivity, often using graph-based models to assess and enhance habitat networks.
2. We introduce GECOT, an open-source command line tool for solving budget-constrained multi-species landscape connectivity optimization problems when measuring landscape connectivity with the Probability of Connectivity (PC) indicator.
3. GECOT builds on a theoretical framework to model conservation and restoration planning as a connectivity optimization problem where options impact the quantity or quality of available habitat and the permeability of habitat and landscape matrix.
4. For a given case study, GECOT can be used to find the optimal solution using a mixed-integer linear solver or heuristic solutions with different sub-optimal algorithms, when the optimal solution is too expensive in computation time.

KEYWORDS

systematic conservation planning, ecological restoration, landscape connectivity, multispecies optimization, mixed integer linear programming

1 | INTRODUCTION

Habitat loss and fragmentation, mainly driven by human activities, have become major challenges for biodiversity conservation worldwide (Bronzizio, Settele, Díaz, & Ngo, 2019). To address these challenges, systematic conservation planning (SCP) has emerged as a pivotal approach, offering efficient and reproducible strategies for designing spatial conservation and restoration policies. At its core, an SCP optimization problem consists of selecting a region of interest with features to be improved, and identifying among several options which ones should be implemented (Margules & Pressey, 2000). This process inherently involves solving a combinatorial optimization problem, where numerous combinations of planning options have to be evaluated to determine the most effective set for conservation or restoration within a given budget. Given the vast number of possible combinations and constraints to be satisfied, advanced algorithms and computational techniques are often employed to find optimal or near-optimal solutions efficiently. Various software tools have been developed to assist in solving specific SCP problems, including Marxan (Ball, Possingham, & Watts, 2009), Zonation (Moilanen et al., 2005), C-Plan (Pressey, Watts, Barrett, & Ridges, 2009) and PrioritizR (Hanson et al., 2024), each of which can be used to find solutions that optimize particular landscape features. Among these tools,

only PrioritizR allows to find optimal solutions for the landscape features it targets while other ones are using heuristic methods. Although heuristic methods have been shown to return good solutions on average, they don't provide information on how far these solutions are from the optimum (Schuster, Hanson, Strimas-Mackey, & Bennett, 2020; van Mantgem, Hillebrand, Rose, & Klau, 2023). When landscape connectivity is a feature to be optimized, Marxan-Connect (Daigle et al., 2020), Zonation, C-Plan and PrioritizR use structural connectivity indicators, like the number of habitat patches, the length of their boundaries or the betweenness centrality, as opposed to functional connectivity (Taylor, Fahrig, & With, 2006). In the meantime, tools for evaluating landscape connectivity such as Conefor (Saura & Torne, 2009) and Graphab (Foltête, Clauzel, & Vuidel, 2012) have been developed and offer well established connectivity indicators like the probability of connectivity indicator (PC) (Saura & Pascual-Hortal, 2007) and the integral index of connectivity (IIC) (Pascual-Hortal & Saura, 2006), which have respectively become the first and second most commonly used measures for connectivity analyses (Hashemi & Darabi, 2022) and received encouraging empirical support (Pereira, Segurado, & Neves, 2011; Awade, Boscolo, & Metzger, 2012). However, these tools do not provide efficient methods for optimizing landscape connectivity under budgetary constraints, other than ranking landscape elements according to their relative contribution to the indicator value (Urban & Keitt, 2001). Recently, progress has been made in optimizing IIC under budget constraints with the software RestoptR

(Justeau-Allaire et al., 2023) and other studies investigated deep reinforcement learning methods (Equihua, Beckmann, & Seppelt, 2024). In this article, we present GECOT, our open source software implementing the latest methods for finding optimal solutions when maximizing the PC indicator under budget constraints (Hamon, Albert, Couëtoux, & Vaxès, 2023), available at <https://doi.org/10.5281/zenodo.15295914> (Hamon, 2025). We give more details on the PC indicator and the general optimization problem that GECOT is aimed to tackle, allowing to capture both restoration and conservation problems. Then, we show the primary features of GECOT by describing its command line interface and the format of input data. Finally, we show how to construct an instance of the problem from spatial data and use GECOT to compute the optimal solution for a given budget value.

2 | THE PROBABILITY OF CONNECTIVITY INDICATOR

In the last decades, graph theory has become a popular tool for modeling ecological landscapes (Bunn, Urban, & Keitt, 2000; Urban & Keitt, 2001) and assess their structural or functional traits with the help of indicators. Indeed, a landscape can be viewed as a directed graph $G = (V, A, w, \pi)$ whose vertices represent the habitat patches, i.e., contiguous regions of habitat, and arcs, a.k.a directed edges, represent the connections that individuals of a given species can use to travel from one patch to another. More formally, each habitat patch is represented by a vertex $u \in V$ with a weight w_u that models its ecological quality (e.g. habitat area for the species) and each route that individuals can take from a patch u to a patch v is represented by an arc $(u, v) \in A$ and a probability π_{uv} modeling the success rate for making the corresponding journey. Instead of habitat patches, vertices can also represent cells in a raster map modeling the landscape, where cells without habitat would be associated with a zero quality weight. Usually, the probability of each arc $(u, v) \in A$ of being used is calculated with a negative exponential of the distance such that

$$\pi_{uv} = e^{-\alpha \cdot l_{uv}}, \quad (1)$$

where l_{uv} is the distance between the patches u and v and $\alpha = -\ln(0.5)/\theta$ is a constant adjusted such that $\pi_{uv} = 0.5$ when $l_{uv} = \theta$, θ being the median dispersal distance of the considered species. The Probability of Connectivity (PC) indicator (Saura & Pascual-Hortal, 2007) is a measure of both habitat amount and connectivity and in this way characterize the amount of reachable habitat (Martensen, Saura, & Fortin, 2017), calculated as:

$$PC = \frac{\sum_{s \in V} \sum_{t \in V} (w_s \cdot w_t \cdot \Pi(s, t))}{\mathcal{A}^2},$$

where $\Pi(s, t)$ denotes the probability of the most reliable path from s to t , that is, the st -path whose product of the probabilities of its arcs is the highest,

$$\Pi(s, t) = \max_{P:st\text{-path}} \prod_{a \in P} \pi_a,$$

and \mathcal{A} is the area of the study region containing the landscape being analyzed, used to normalize the result, enabling to compare the PC values of very different landscapes. Thus, the numerator of the PC formula, denoted PC_{num} is sufficient for comparing PC values of different scenarios on the same landscape. Another frequently used measure is the Equivalent Connected Area (ECA) (Saura, Estreguil, Mouton, & Rodríguez-Freire, 2011) which is expressed as $ECA = \sqrt{PC_{num}}$.

2.1 | Our interpretation

We consider a stochastic process where we randomly select a habitat point p from the landscape. When a point p is randomly selected, the probability it belongs to any specific patch s is $\frac{w_s}{\mathcal{W}}$, where $\mathcal{W} = \sum_{u \in V} w_u$ is the total habitat amount in the landscape. Using linearity of expectation, we can calculate the expected amount of habitat of a given patch $t \in V$ that can be reached from a randomly selected point of habitat as

$$R(t) = w_t \cdot \left(\sum_{s \in V} \frac{w_s}{\mathcal{W}} \cdot \Pi(s, t) \right).$$

Extending this to the entire landscape, the expected amount of reachable habitat from any randomly selected habitat point becomes $\sum_{t \in V} R(t)$. Thus, PC_{num} can be reformulated as:

$$PC_{num} = \mathcal{W} \cdot \sum_{t \in V} R(t).$$

This formulation clearly shows that the PC indicator is proportional to the product of the total habitat amount (\mathcal{W}) and the expected amount of reachable habitat from a randomly selected point of habitat within the landscape.

3 | EXPRESSING CONSERVATION AND RESTORATION PROBLEMS

Ultimately, GECOT is a solver, i.e. it takes an instance of an optimization problem as input and outputs a solution to it. In the following, we describe the *Max-PC* problem that GECOT solves when optimizing the PC for a single species. Let $G = (V, A, w, \pi)$ be a directed graph representing a landscape, Φ be a set of options for improving the landscape and β be the available budget. Each option $i \in \Phi$ has a cost c_i and can improve the quality (resp. probability) of one or many vertices (resp. arcs) of the graph. To model these improvements, there is a weight $w_u^i \geq 0$ for each vertex $u \in V$ representing the quality increase of vertex u if option i is implemented, w_u^i being 0 if vertex u is not affected by option i , and a probability $\pi_{uv}^i \geq \pi_{uv}$ for each arc $(u, v) \in A$ representing the improvement in permeability of the arc (u, v) achieved by option i , π_{uv}^i being π_{uv} if arc (u, v) is not affected by option i . The *Max-PC* problem consists in finding the set of options $S \subseteq \Phi$ whose total cost remains below the budget β and which maximizes the PC value of the landscape improved by these options, i.e. $PC_{num}(G' = (V, A, w', \pi'))$. The improved weight of a patch u is calculated as $w_u' = w_u + \sum_{i \in S} w_u^i$, i.e. the improvements of different

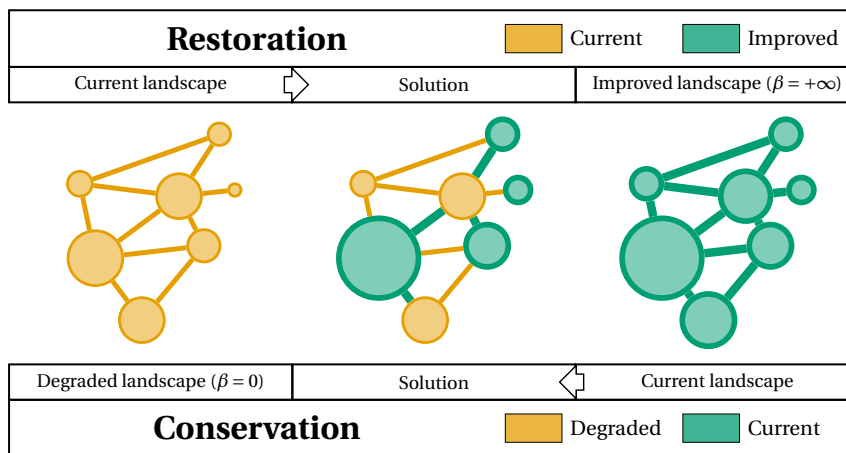


FIGURE 1 Illustration of the equivalence between restoration and conservation problems, assuming that all of the landscape elements will be degraded if no conservation options are taken, i.e. the budget β is null.

options on the same patch accumulate, and the improved probability of an arc (u, v) is calculated as $\pi'_{uv} = \max(\pi_{uv}, \max_{i \in S} \pi_{uv}^i)$, i.e. only the best allowed probability of the arc is taken. This problem captures both restoration and conservation problems. In the case of restoration, G is the current landscape, Φ is the set of restoration options, w_u^i represents the amount by which the habitat area of the patch u is increased and π_{uv}^i represents the improved probability of the arc (u, v) if the option i is chosen. In the case of conservation, G represents the landscape as it would be if no conservation action is taken, Φ is the set of conservation options, w_u^i represents the amount of habitat area of the patch u that would disappear if the option i is not chosen, and π_{uv}^i is a lower bound on the probability of the arc (u, v) guaranteed by the choice of i . In both cases (Figure 1), the *Max-PC* problem is to find a set of options S of total cost at most β that maximizes the PC of the improved (resp. protected) landscape G' .

4 | GECOT AS A TOOL

The main functionality of GECOT is solving instances of *Max-PC* which is done using a command-line with the following structure:

```
gecot_optimize <algo> <instance> <budget> [<params>]
```

- <algo> The optimization algorithm to use, see Section 4.1.
- <instance> The path to a JSON file that defines the problem instance, see Section 4.2
- <budget> The available financial budget β , limiting the total cost of selected improvement options.
- [<params>] Additional parameters that control the behavior of the tool, e.g. printing the solution in a file, see Section 4.3.

4.1 | Supported algorithms

GECOT is particularly designed for finding the optimal set of options $S \subseteq \Phi$ that maximizes the PC value of a landscape using the method introduced in (Hamonc, Albert, et al., 2023). This method, referred in GECOT by the name "prep_mip", relies on a Mixed Integer Programming (MIP) formulation whose size may vary depending on the instance landscape properties but can typically solve instances with up to 300 vertices representing habitat (more details are given in the supplementary materials). GECOT provides also four other heuristic algorithms (Table 1) that offer sub-optimal solutions but can run on much bigger instances. These algorithms, whose list may be extended in the future, are "static_incr", "static_decr", "greedy_incr" and "greedy_decr". They build a solution S by iteratively adding or removing options until no option fits into S such that the cost $\sum_{i \in S} c_i$ is below the budget β . Incremental algorithms start with $S = \emptyset$ and add options to S while decremental algorithms start with $S = \Phi$ and remove options until the budget is met. To select which option is added or removed from S , these algorithms compute the potential δ_S^i for each option i which is the amount by which the PC value varies by implementing option i if $i \notin S$, or removing it if $i \in S$. Incremental algorithms then consider adding options whose increase in PC relative to cost, i.e. δ_S^i/c_i , is greatest, while decremental algorithms consider removing options whose decrease in PC relative to cost is smallest. Static algorithms only compute these potential once, with $S = \emptyset$ for the "static_incr" or $S = \Phi$ for the "static_decr", while greedy algorithms recompute the potentials every time the most influential option is added to or removed from S . The main limitation of these heuristic algorithms is that they assess the potential effects of options independently and therefore do not take into account the potential interactions of multiple options selected together, i.e. their cumulative effects. These interactions can be: synergies when potential changes are in series and only their combination might improve the connectivity, redundancies when potential changes are in parallel and choosing only one might be sufficient, or antagonisms

TABLE 1 The available algorithms, their descriptions and the order of magnitude of the computation time obtained on a 151 habitat patches landscape described in supplementary materials (Aix/instance_70x70) on a computer with a 5Ghz CPU and limiting the number of threads to one.

Algorithm	Description	Cumulative Effects	Computation Time
static_incr	From the base landscape, add the options with the greatest gain/cost that meet the budget.	None	~ 3 seconds
static_decr	From the improved landscape, remove the options with the worst gain/cost ratio to fit the budget.	None	~ 4 seconds
greedy_incr	From the base landscape, iteratively add the option with the greatest gain/cost ratio until no more fit in the budget.	Some	~ 2 minutes
greedy_decr	From the improved landscape, iteratively remove the option with the worst gain/cost ratio until the budget is met.	Some	~ 2 minutes
mip	MIP formulation without preprocessing.	All	cancelled (>10 hours)
prep_mip	MIP formulation with preprocessing.	All	~ 40 minutes

when a potential change might make no sense without additional actions (Hamonik, Couëtoux, Vaxès, & Albert, 2023). Static algorithms compute the potentials of options only separately, meaning they don't take cumulative effects into account at all. Greedy algorithms test the interactive effects of individual options and those already in S , and by doing so they can account for some of the possible cumulative effects. Conversely, "mip" and "prep_mip" algorithms both compute the optimal solution, and by doing so they consider all possible cumulative effects.

4.2 | Instance file structure

The problem instance is defined in a JSON file that references multiple CSV files describing various aspects of the landscape graph and improvement options. An example structure for the instance file is shown in Listing 1:

Listing 1 instance.json

```
{
  "options": {
    "csv_file": "options.csv"
  },
  "cases": {
    "squirrel": {
      "vertices": {
        "csv_file": "vertices.csv"
      },
      "arcs": {
        "csv_file": "arcs.csv"
      },
      "vertices_improvements": {
        "csv_file":
          ↪ "vertices_improvements.csv"
      },
      "arcs_improvements": {
        "csv_file":
          ↪ "arcs_improvements.csv"
      }
    }
  },
  "criterion": "squirrel"
}
```

The relationships between these CSV files are summarized in Figure 2.

The "criterion" field specifies the objective function to be optimized, which aggregates the PC_{num} values of all cases. In the example of Listing 1, the optimization only concerns the species *squirrel*. For multi-species optimization, numerous cases can be specified, in addition to the *squirrel* one, describing landscape graphs for species representative of different dispersal abilities and environmental niches (Albert, Rayfield, Dumitru, & Gonzalez, 2017; Prima et al., 2024). GECOT allows specifying a custom criterion function as an arbitrary combination of weighted sums and minimums of the PC_{num} values of all graphs, as soon as all coefficients are positive. For example with *squirrel* and *hedgehog*, whose PC_{num} values of the graphs are respectively denoted S and \mathcal{H} , Listings 2 shows a criterion for optimizing the weighted sum $S + \frac{1}{2}\mathcal{H}$ and Listings 3 show a more sophisticated criterion using the minimum function to nudge the optimization in finding a solution where the PC value for *squirrel* is twice the one for *hedgehog* and prevent solutions where improving the landscape benefits to only one species.

Listing 2 Criterion for $S + \frac{1}{2}\mathcal{H}$

```
{
  "criterion": {
    "operation": "sum",
    "values": [
      [1, "squirrel"],
      [0.5, "hedgehog"]
    ]
  }
}
```

Listing 3 Criterion for $\frac{1}{5}S + \frac{1}{10}\mathcal{H} + \min(S + \frac{1}{2}\mathcal{H})$

```
{
  "criterion": {
    "operation": "sum",
    "values": [
      [0.2, "squirrel"],
      [0.1, "hedgehog"],
      {
        "operation": "min",
        "values": [
          [1, "squirrel"],
          [0.5, "hedgehog"]
        ]
      }
    ]
  }
}
```

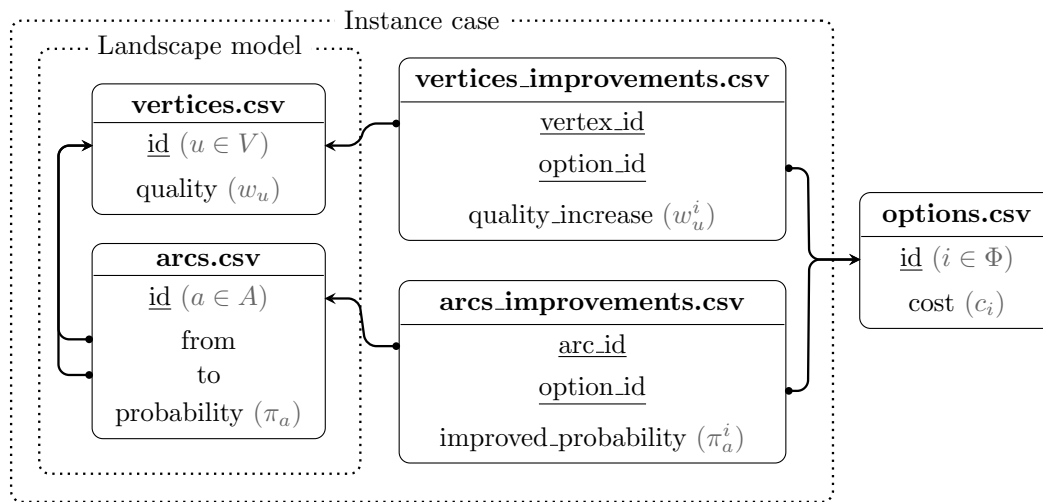


FIGURE 2 Diagram representing the structure of the input files used to define an instance of *Max-PC*. Each CSV file is represented as a box containing the list of its column names. Underlined column names correspond to unique identifiers for the rows of the corresponding file. The mathematical notation associated with each column is provided in parentheses. Arrows show how files are linked, pointing from a referencing column to the corresponding identifier of another file.

4.3 | Optional parameters

Global parameters, can be added to the command line to modify the general behavior of GECOT, like specifying the output format of the solution, enabling the trace of algorithms execution. The most common parameters are described in Table 2.

Furthermore, each algorithm provides its own specific parameters that can be added to the command line to modify its behaviour. The list of these parameters can be obtained with :

```
gecot_optimize <algorithm> --list-params
```

For example, an important parameter for the "mip" and "prep_mip" algorithms is the ability to specify the path to a commercial MIP solver, like Gurobi (Gurobi Optimization, 2022), with the parameter "-gurobi-path=<path>". Indeed, by default, GECOT includes the binaries of the open source COIN-OR Branch and Cut solver (CBC) (Forrest et al., 2022) for solving the MIP formulations, which can be greatly outperformed by commercial solvers, in terms of computation time, on large instances (examples are provided in the supplementary materials).

5 | WORKED EXAMPLE

Patch-based models are quite ubiquitous in landscape connectivity analyses for modeling landscapes as graphs. Usually in these models, each contiguous region of habitat is represented as a vertex and the connection between two patches u and v is modeled with reciprocating arcs

(u, v) and (v, u) . There is a wide range of methods for designing patch-based models, from the weights of vertices and arcs to the choice of vertex pairs to be connected by arcs. There is, for example, minimum planar graphs (Fall, Fortin, Manseau, & O'Brien, 2007), link threshold graphs and complete graphs (Galpern, Manseau, & Fall, 2011). In this section, in order to provide an example that can easily be generalized, we introduce a method for defining an instance of the *Max-PC* problem directly on raster map data.

We take a raster with 30x30 cells of 100x100 meters representing a road network surrounded by fields and forests on the outskirts of Aix-en-Provence, France. For the sake of the example, we selected arbitrarily 12 groups of road cells that we will consider suitable for installing wildlife crossings, e.g. squirrel bridges, and 6 groups of cells representing fields that we will consider for land acquisitions in order to preserve them as wilderness areas or restore them by replanting trees.



FIGURE 3 A 30x30 cell raster of a landscape on the outskirts of Aix-en-Provence (France) with roads on which wildlife crossings can be built and fields that can be acquired.

TABLE 2 Common parameters for GECOT.

Parameter	Description
-h or --help	Lists command line usages and parameters.
-v or --verbose	Enables verbosity, printing details about the input instance and the steps of algorithms.
-o or --output-json <path>	Specifies a file path for printing the solution in JSON format.
--output-csv <path>	Specifies a file path for printing the solution in CSV format.
-max-threads <num>	Limits the number of CPU cores to use.

TABLE 3 Cells properties depending on their type.

Cell type	Quality Weight	Crossing Probability	Cell Cost	Improved Quality	Improved Probability
Habitat	1	1.0	-	-	-
Field	0	0.98	-	-	-
Road	0	0.6	-	-	-
Ld. Acq.	0	0.98	0.2	1	0.995
W. Crss.	0	0.6	1	0	0.95

Ld. Acq.: Land Acquisition.

W. Crss.: Wildlife Crossing.

In our case, each raster cell is associated with one of the following cell types: "habitat", "field", "road", "wildlife crossing" and "land acquisition" where "wildlife crossing" denotes cells of roads suitable for installing one and "land acquisition" denotes cells of fields that can be acquired. Each option $i \in \Phi$ is associated with a contiguous group cells of either type "wildlife crossing" or "land acquisition". The cost c_i of option i is the sum of the costs of its cells. Then, we associate each cell type with weights representing the quality of the habitat within the cell, the probability of an individual successfully crossing the cell in either directions and, if the cell is part of a restoration option $i \in \Phi$, the cost of the cell, contributing to the total cost of the option i , and the improvements in quality and probability that implementing the option achieves, c.f. Table 3. We chose a probability of 0.98 for cells representing fields based on a mean dispersal distance for red squirrels of 3.6 km in rural areas (Hämäläinen, Fey, & Selonen, 2019) using the classical formula showed in Equation 1, and 0.6 for cells of road.

A way to build a *Max-PC* instance from this case study consists in converting the raster into a graph. Indeed, a square raster can be viewed as a graph where each vertex corresponds to a cell and adjacent cells are linked by an arc. Here, we consider 4-neighborhood, i.e. each cell is connected to its top, left, bottom and right neighbors (Figure 4). Thus, in this model, each cell of the raster corresponds to a vertex $v \in V$ of the graph and is associated to a quality weight according to its cell type, i.e. $w_v = 1$ for habitat cells and $w_v = 0$ otherwise. A straightforward modeling from that point would be to connect each pair of vertices corresponding to adjacent cells with an arc, however, this would lead to arcs whose probability depends on two cells since it represent the travel from one cell center to its boundary and then to the center of the other cell. We can add vertices to the borders of adjacent cells, so that arcs can represent the travel between the border and the center of each cell, in either direction (Figure 4a). In this way, each arc of the

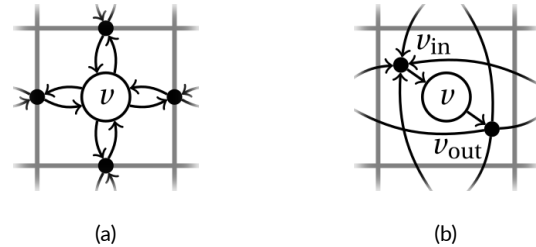


FIGURE 4 (a) A graph model for a square raster cell where each arc represent the travel between one boundary and the center of the cell. If the probability for successfully crossing the cell is independent of the direction, all arcs have the same probability of $\sqrt{\pi(v)}$. (b) An equivalent model when the probabilities of crossing the cell is the same in all four directions. In this case the arcs from v_{in} to v and from v to v_{out} have a probability of $\sqrt{\pi(v)}$ and others arcs have a probability of 1.

graph can be associated with a single cell, making it easier to model the probability improvements of a cell by increasing the probabilities of its internal arcs. However, if the probability for crossing a cell is the same for all directions, we can use only two arcs going in and out of v to represent the travel from any boundary to the center and vice versa (Figure 4b). Arcs representing the travel from the border of a cell to its center should have a probability of $\sqrt{\pi(v)}$, where $\pi(v)$ is the probability associated to the cell represented by the vertex v , since two of these arcs must be passed through to cross the cell.

Now that we have defined the structure of the graph, the CSV files describing the instance can be classified according to the format explained in Section 4.2. For example, given a cell of type "land acquisition" represented by the vertex $v \in V$ and its associated option $i \in \Phi$, the vertices improvements CSV file will contain a line representing $w_v^i = 1$ and, for each arc $a \in A$ incident to v , the arcs improvements CSV file will contain a line representing $\pi_a^i = 0.995$. GECOT can then be used to solve the instance with any budget value using the following command:

```
gecot_optimize prep_mip instance.json 4 -o sol.json
```

This command uses the `prep_mip` algorithm to find an optimal solution for the *Max-PC* instance with a budget of 4 and outputs it to the `sol.json` file. The computed solution allows to retrieve the set of options that are part of the solution, and therefore the vertices and arcs of the graph that should be improved to maximize PC with a 4 budget, c.f. Figure 5.

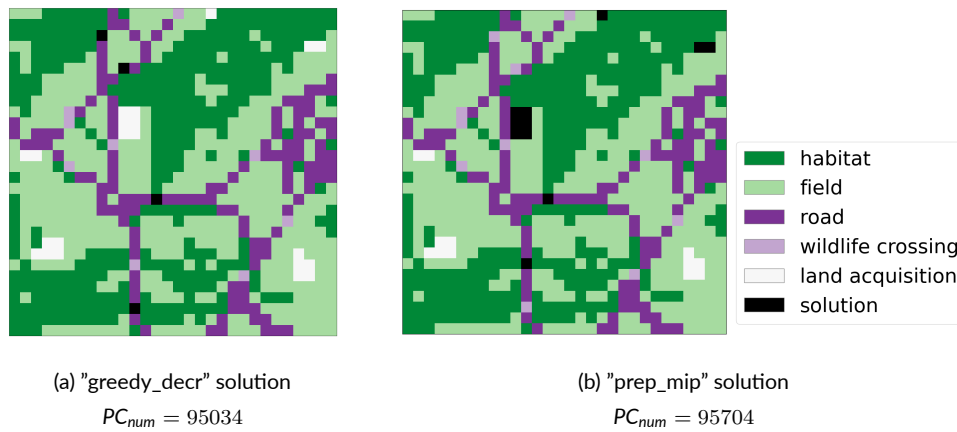


FIGURE 5 A visualization of the solutions obtained for a budget value of 4 with the "greedy_decr" (a) and "prep_mip" (b) algorithms.

In this simple case, we see that the solution provided by the "prep_mip" algorithm leads to a greater increase in PC than the one returned by the "greedy_decr" algorithm. The "prep_mip" takes also more time, which is still very reasonable here (see supplementary materials for more examples). Both solutions also differ by the type of options that are chosen with more wildlife crossings selected with the greedy algorithm and more land acquisition with the "prep_mip" algo because the greedy algorithm does not see the improvements that will come by selecting all the land acquisition cells simultaneously.

6 | DISCUSSION

Here, we aimed to provide a powerful and open-source tool to support systematic conservation planning when optimizing landscape connectivity under budget constraint using the Probability of Connectivity (PC) or the Equivalent Connected Area (ECA) indicators (both problems are identical). GECOT builds upon the *Max-PC* optimization problem that generalizes many of the problematic encountered when planning for conservation or restoration of landscapes. One of the key contributions of GECOT is its ability to find optimal solutions for landscapes with hundreds of habitat patches, a capability that, to our knowledge, has not been available in previous tools. Furthermore, GECOT allows to find optimal solutions when optimizing landscape connectivity for several species simultaneously instead of one by one, which is rarely explored (Wood et al., 2022) and likely to provide more cost-effective solutions (Dilkina et al., 2017).

A major motivation behind GECOT is to address the limitations of existing approaches, particularly the shortcomings of simple algorithms currently used in practice to optimize the PC indicator. Indeed, although static and greedy algorithms often find solutions within 95% of the value of the optimal one, they provide no theoretical guarantee on this percentage and can, in some cases, yield arbitrarily poor solutions when the cumulative effects between habitat patches are significant (Hamonnic, Albert, et al., 2023). This is particularly the case when the effectiveness of many options depends on the selection of others, for example when

different options affect the same least-cost path in series, as is the case for equipping all dams along a single river branch with fish ladders (Hamonnic, Couëtoux, et al., 2023). However, it is often necessary to make trade-offs between guaranteed solution quality and computation time, as the computational effort required to obtain the optimal solution still grows exponentially with the number of options (Hamonnic, Couëtoux, et al., 2023). We give more information in the supplementary material on how to make educated guesses about the running time of each algorithm and thus on the size of solvable instances.

We foresee two use cases for GECOT. Firstly, with the help of its command line interface, GECOT can easily be integrated in modelling frameworks and within other softwares, taking advantage of the highly efficient C++ implementations of both the optimal and the sub-optimal algorithms. Finally, GECOT can serve as a baseline for future research. The availability of a tool capable of computing optimal solutions makes it possible to benchmark and compare new heuristic and exact methods aimed at optimizing landscape connectivity through the PC indicator. We hope that this contribution will help advance the field by encouraging the development of more efficient and effective conservation planning approaches.

AUTHOR CONTRIBUTIONS

Cécile H. Albert, Basile Couëtoux, François Hamonnic and Yann Vaxès contributed to the theoretical formalism behind the software. François Hamonnic developed the software and wrote the first draft of the paper. All the authors contributed to the revision and enhancement of the draft until the final version of the manuscript.

ACKNOWLEDGEMENTS

The research on this article was supported by the ERC project SCALED (grant n°949812). For stimulating exchanges on the worked example, we

also thank Danielle Clake, Mailys Queru, Victorine Demiralp and Romain Monassier.

CONFLICT OF INTEREST STATEMENT

The authors declare no conflicts of interest.

DATA AVAILABILITY STATEMENT

The source code of GECOT and the test instances used are hosted at <https://github.com/fhamonic/gecot> and has been archived with Zenodo at <https://doi.org/10.5281/zenodo.15295914> (Hamon, 2025).

References

- Albert, C. H., Rayfield, B., Dumitru, M., & Gonzalez, A. (2017). Applying network theory to prioritize multispecies habitat networks that are robust to climate and land-use change. *Conservation Biology*, 31, 1383–1396.
- Awade, M., Boscolo, D., & Metzger, J. P. (2012). Using binary and probabilistic habitat availability indices derived from graph theory to model bird occurrence in fragmented forests. *Landscape Ecology*, 27, 185–198. doi: 10.1007/s10980-011-9667-2
- Ball, I. R., Possingham, H. P., & Watts, M. (2009). Marxan and relatives: software for spatial conservation prioritisation. *Spatial conservation prioritisation: Quantitative methods and computational tools*, 14, 185–196.
- Brondizio, E. S., Settele, J., Diaz, S., & Ngo, H. T. (2019). *Global assessment report on biodiversity and ecosystem services of the intergovernmental science-policy platform on biodiversity and ecosystem services*. IPBES, Bonn, Germany. doi: 10.5281/zenodo.3831673
- Bunn, A., Urban, D., & Keitt, T. (2000). Landscape connectivity: A conservation application of graph theory. *Journal of Environmental Management*, 59(4), 265–278. doi: 10.1006/jema.2000.0373
- Daigle, R. M., Metaxas, A., Balbar, A. C., McGowan, J., Tremblay, E. A., Kuempel, C. D., ... Beger, M. (2020). Operationalizing ecological connectivity in spatial conservation planning with marxan connect. *Methods in Ecology and Evolution*, 11(4), 570–579. doi: <https://doi.org/10.1111/2041-210X.13349>
- Dilkina, B., Houtman, R., Gomes, C. P., Montgomery, C. A., McKelvey, K. S., Kendall, K., ... Schwartz, M. K. (2017). Trade-offs and efficiencies in optimal budget-constrained multispecies corridor networks. *Conservation Biology*, 31(1), 192–202.
- Equihua, J., Beckmann, M., & Seppelt, R. (2024). Connectivity conservation planning through deep reinforcement learning. *Methods in Ecology and Evolution*, 15(4), 779–790. Retrieved from <https://besjournals.onlinelibrary.wiley.com/doi/abs/10.1111/2041-210X.14300> doi: <https://doi.org/10.1111/2041-210X.14300>
- Fall, A., Fortin, M.-J., Manseau, M., & O'Brien, D. (2007). Spatial graphs: principles and applications for habitat connectivity. *Ecosystems*, 10(3), 448–461. doi: 10.1007/s10021-007-9038-7
- Foltête, J.-C., Clauzel, C., & Vuidel, G. (2012). A software tool dedicated to the modelling of landscape networks. *Environmental Modelling & Software*, 38, 316–327. Retrieved from <https://www.sciencedirect.com/science/article/pii/S1364815212001995> doi: <https://doi.org/10.1016/j.envsoft.2012.07.002>
- Forrest, J., Ralphs, T., Santos, H. G., Vigerske, S., Hafer, L., Kristjansson, B., ... to st (2022). *coin-or/cbc: Release releases/2.10.8*. doi: 10.5281/zenodo.6522795
- Galpern, P., Manseau, M., & Fall, A. (2011). Patch-based graphs of landscape connectivity: A guide to construction, analysis and application for conservation. *Biological Conservation*, 144(1), 44–55. doi: 10.1016/j.biocon.2010.09.002
- Gurobi Optimization, L. (2022). *Gurobi optimizer reference manual*. Retrieved from <https://www.gurobi.com>
- Hämäläinen, S., Fey, K., & Selonen, V. (2019). The effect of landscape structure on dispersal distances of the eurasian red squirrel. *Ecology and Evolution*, 9(3), 1173–1181.
- Hamon, F. (2025). *Gecot release v1.0*. Retrieved from <https://github.com/fhamonic/gecot> doi: 10.5281/zenodo.15295914
- Hamon, F., Albert, C., Couëtoux, B., & Vaxès, Y. (2023). Optimizing the ecological connectivity of landscapes. *Networks*, 81(2), 278–293.
- Hamon, F., Couëtoux, B., Vaxès, Y., & Albert, C. H. (2023). Cumulative effects on habitat networks: How greedy should we be? *Biological Conservation*, 282, 110066. Retrieved from <https://www.sciencedirect.com/science/article/pii/S0006320723001672> doi: <https://doi.org/10.1016/j.biocon.2023.110066>
- Hanson, J. O., Schuster, R., Strimas-Mackey, M., Morrell, N., Edwards, B. P. M., Arcese, P., ... Possingham, H. P. (2024). Systematic conservation prioritization with the prioritizr package. *Conservation Biology*, *In press*. doi: 10.1111/cobi.14376
- Hashemi, R., & Darabi, H. (2022). The review of ecological network indicators in graph theory context: 2014–2021. *International Journal of Environmental Research*, 16. doi: 10.1007/s41742-022-00404-x
- Justeau-Allaire, D., Hanson, J. O., Lannuzel, G., Vismara, P., Lorca, X., & Birnbaum, P. (2023). : an r package for ecological restoration planning. *Restoration Ecology*, 31(5), e13910. doi: <https://doi.org/10.1111/rec.13910>
- Margules, C. R., & Pressey, R. L. (2000). Systematic conservation planning. *Nature*, 405(6783), 243–253.
- Martensen, A. C., Saura, S., & Fortin, M.-J. (2017). Spatio-temporal connectivity: assessing the amount of reachable habitat in dynamic landscapes. *Methods in Ecology and Evolution*, 8(10), 1253–1264. doi: <https://doi.org/10.1111/2041-210X.12799>
- Moilanen, A., Franco, A. M., Early, R. I., Fox, R., Wintle, B., & Thomas,

- C. D. (2005). Prioritizing multiple-use landscapes for conservation: methods for large multi-species planning problems. *Proceedings of the Royal Society B: Biological Sciences*, 272(1575), 1885–1891. doi: 10.1098/rspb.2005.3164
- Pascual-Hortal, L., & Saura, S. (2006). Comparison and development of new graph-based landscape connectivity indices: towards the prioritization of habitat patches and corridors for conservation. *Landscape ecology*, 21, 959–967.
- Pereira, M., Segurado, P., & Neves, N. (2011). Using spatial network structure in landscape management and planning: A case study with pond turtles. *Landscape and Urban Planning*, 100, 67–76.
- Pressey, R. L., Watts, M. E., Barrett, T. W., & Ridges, M. J. (2009). The c-plan conservation planning system: origins, applications, and possible futures. *Spatial conservation prioritization: quantitative methods and computational tools*, 211–34.
- Prima, M.-C., Renaud, J., Witté, I., Suarez, L., Rouveyrol, P., Fernando, M., ... Thuiller, W. (2024). A comprehensive framework to assess multi-species landscape connectivity. *Methods in Ecology and Evolution*, 15(12), 2385–2399. Retrieved from <https://besjournals.onlinelibrary.wiley.com/doi/abs/10.1111/2041-210X.14444> doi: <https://doi.org/10.1111/2041-210X.14444>
- Saura, S., Estreguil, C., Mouton, C., & Rodríguez-Freire, M. (2011). Network analysis to assess landscape connectivity trends: Application to European forests (1990–2000). *Ecological Indicators*, 11, 407–416.
- Saura, S., & Pascual-Hortal, L. (2007). A new habitat availability index to integrate connectivity in landscape conservation planning: Comparison with existing indices and application to a case study. *Landscape and Urban Planning*, 83, 91–103.
- Saura, S., & Torne, J. (2009). Conefor sensinode 2.2: a software package for quantifying the importance of habitat patches for landscape connectivity. *Environmental Modelling & Software*, 24, 135–139.
- Schuster, R., Hanson, J. O., Strimas-Mackey, M., & Bennett, J. R. (2020). Exact integer linear programming solvers outperform simulated annealing for solving conservation planning problems. *PeerJ*, 8, e9258.
- Taylor, P. D., Fahrig, L., & With, K. A. (2006). Landscape connectivity: a return to the basics. In *Connectivity conservation* (Vol. 14, pp. 29–43). Cambridge University Press. doi: 10.1017/CBO9780511754821.003
- Urban, D., & Keitt, T. (2001). Landscape connectivity: A graph-theoretic perspective. *Ecology*, 82, 1205–1218. doi: 10.1890/0012-9658(2001)082[1205:LCAGTP]2.0.CO;2
- van Mantgem, E. S., Hillebrand, J., Rose, L., & Klau, G. W. (2023). Coco: conservation design for optimal ecological connectivity. *Frontiers in Ecology and Evolution*, 11, 1149571.
- Wood, S. L., Martins, K. T., Dumais-Lalonde, V., Tanguy, O., Maure, F., St-Denis, A., ... Gonzalez, A. (2022). Missing interactions: the current state of multispecies connectivity analysis. *Frontiers in Ecology and Evolution*, 10, 830822.

SUPPORTING INFORMATION

Additional supporting information can be found online in the Supporting Information section at the end of this article.