Technische Universität Berlin
Electrical Engineering and Computer Science
Institute of Software Engineering and Theoretical Computer Science
Algorithmics and Computational Complexity (AKT)

The Influence of Habitat Structure on the Algorithmic Complexity of Placing Green Bridges

Maike Herkenrath

Thesis submitted in fulfillment of the requirements for the degree "Bachelor of Science" (B. Sc.) in the field of Computer Science

October 2021

Supervisor and first reviewer:	Prof. Dr. Rolf Niedermeier
Second reviewer:	Prof. Dr. Markus Brill
Co-Supervisors:	Dr. Till Fluschnik and Leon Kellerhals

Hiermit erkläre ich, dass ich die vorliegende Arbeit selbstständig und eigenhändig sowie ohne unerlaubte fremde Hilfe und ausschließlich unter Verwendung der aufgeführten Quellen und Hilfsmittel angefertigt habe.

Die selbstständige und eigenhändige Ausfertigung versichert an Eides statt

Berlin, den ____________________ ____________________________

Datum Unterschrift

Zusammenfassung

In dieser Arbeit befassen wir uns mit der Frage wo man am besten Wildtierbrücken plaziert, sodass die Tiere sich frei in ihren Lebensräumen bewegen können, ohne vom Straßenverkehr oder Zügen gefährdet zu werden. Eine Verringerung von Wildunfällen ist außerdem anzustreben, da sie für mehr Sicherheit der Autofahrer und zur Verringerung von Sachschäden beiträgt.

Wir betrachten das Problem 1-Reach Green Bridges Placement (1-Reach GBP), das von Fluschnik und Kellerhals [FK21] definiert wurde. In einem ungerichteten Graph $G = (V, E)$ gibt es eine Menge von Habitaten $\mathcal{H} \subseteq \{Z \mid Z \subseteq V\}$. Jeder Knoten steht für eine Gebiet, das durch menschengemachte Bauwerke von anderen Gebieten abgeschnitten ist. Sei es durch eine Autobahn, Bahngleise, oder Gebäude. Wenn zwei Knoten durch eine Kante verbunden sind, dann gibt es die Möglichkeit diese beiden Flächen durch eine Wildtierbrücke zu verbinden. Gesucht wird eine Kantenmenge, die einen Graphen induziert, in dem jedes Habitat eine Zusammenhangskomponente bildet. Das entspricht einer Menge von Wildtierbrücken, die es allen betrachteten Tierpopulationen ermöglicht alle Bereiche ihrer Habitate einfach zu erreichen.

Wir untersuchen, welchen Einfluss verschiedene Formen von Habitaten auf die Komplexität des Problems haben. Wir haben induzierte Pfade, Bäume und Kreise betrachtet, in planaren und nicht planaren Graphen. Es stellt sich heraus dass 1-Reach GBP schon mit Habitaten, die Dreiecke induzieren NP-schwer ist. Auch in planaren Graphen oder wenn der Maximalgrad mit drei beschränkt ist, bleibt 1-Reach GBP NP-schwer. Wir finden auch Fälle, die in polynomieller Zeit lösbar sind. Dies ist beispielsweise der Fall, wenn alle Habitate Bäume induzieren, wenn die Habitate kantendisjunkt sind, oder wenn jedes Habitat ein Face in einem planaren Graphen induziert.

Abstract

In this thesis, we consider the placement of wildlife-crossings for animals to roam freely in their habitats. When animals can cross streets or railroad tracks by traversing bridges, fewer wildlife-vehicle collisions happen, which is preferable for both animals and humans.

We analyze the 1-Reach Green Bridges Placement (1-Reach GBP) problem devised by Fluschnik and Kellerhals [FK21]. In an undirected graph there is a set of habitats $\mathcal{H} \subseteq \{Z \mid Z \subseteq V\}$. Every vertex represents an area of land that is closed off from the rest of the world by roads, railroad tracks, or

buildings. Two vertices are connected by an edge if we could build a wildlife-crossing that connects both areas. A solution to the problem is a set of edges, such that every habitat forms a connected component in the graph induced by the set. This represents a set of bridges that could be build to make sure all animals of the examined species can safely get from one part of their habitat to the next.

We examine the influence of a number of different shapes of habitats on the computational complexity of the problem. The habitats induce paths, trees, or cycles in input graphs that are planar, non-planar, or have a bounded maximum degree. It turns out that 1-Reach GBP is already NP-hard with very restricted habitats, such as triangles or with an input graph that has a maximum degree of three. We also show that 1-Reach GBP is efficiently solvable if, for example, all habitats induce trees, the habitats are all edge disjoint, or when each habitat induces a face in a plane graph.

Contents

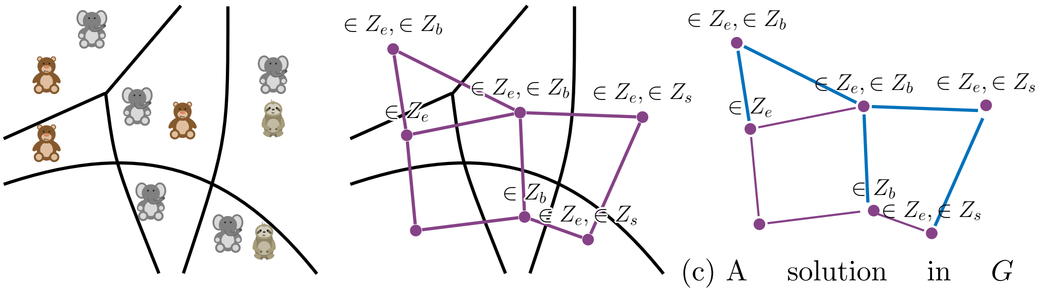
1 Introduction

Humans have claimed most of the landmass on earth. Remaining space is in part fragmented by transportation lines such as roads, railroad tracks, or canals. As a result, wildlife habitats are fragmented into smaller, isolated patches. Species affected by habitat fragmentation experience a decline in population size [And94]. Completely isolated species suffer genetic decline due to inbreeding [Poi+19]. If patches of habitats are only divided by a road or a train track, animals will still cross from one side to the other. This leads to wildlife-vehicle collisions with cars or trains that damage property as well as injures and kills humans and animals [DOR15]. Overall, this leads to diminishing biodiversity [And94].

One measure to counteract the fragmentation of habitats and reduce cases of wildlife collisions is to build wildlife crossings such as bridges or tunnels, in combination with fences along the road, to help the animals safely roam their habitats [Hui+16]. We will from now on refer to all kinds of wildlife crossings as *green bridges*. As the construction of bridges and tunnels is expensive and budgets are limited, it is crucial to find the minimum number and best possible locations to build them such that the benefit for the animals is maximized. The nature of this task allows it to be modeled as a graph problem. We represent the patches of land as vertices, where the habitat patches are subsets of vertices, and possible locations for green bridges are the edges (see Figure 1.1 for an illustration). In this thesis, we analyze the computational complexity of one such problem.

Definition of the model. There have been several approaches to modeling the problem of placing green bridges. Lai et al. [Lai+11], LeBras et al. [LeB+13] and Dilkina et al. [Dil+13] present various approaches that all model it using Steiner trees. In their models, the habitats are represented by the terminal vertices and a solution is a set of edges that connects the terminals. In these cases, however, the distance between two habitat patches is not limited. As a result, the animals may have to cross large stretches of non-habitat land before reaching a habitat patch. To take this into account, Fluschnik and Kellerhals [FK21] devised the 1-Reach Green Bridges Placement problem, which is formally defined as follows.[1]

[1]By $G[F][Z_i]$ we denote the subgraph that is induced by a vertex set Z_i in the subgraph of the graph G induced by an edge set F.



(a) Fragmented habitats. (b) Resulting graph G. (c) A solution in G marked in blue.

Figure 1.1: (a) Sketch of habitats and (b) a corresponding transition into a graph G, with habitats Z_b for the bears, Z_e for the elephants, and Z_s for the sloths. Figure (c) shows a possible solution that connects all habitats.

1-Reach Green Bridges Placement (1-Reach GBP)

Input: An undirected graph G, a set $\mathcal{H} = \{Z_1, \dots, Z_r\}$ of habitats where $Z_i \subseteq V(G)$ for all $i \in \{1, \dots, r\}$, and $k \in \mathbb{N}_0$.

Question: Is there an edge set $F \subseteq E(G)$ with $|F| \leq k$ such that for every $i \in \{1, \dots, r\}$ it holds that $G[F][Z_i]$ is connected?

An instance of 1-Reach GBP is derived from the real world by creating a graph $G := (V, E)$, where every vertex corresponds to a patch of land that is closed off by obstacles that wildlife animals cannot overcome safely by themselves. Two vertices are connected by an edge if a green bridge could connect the corresponding patches of land. The habitats $\mathcal{H}$ are subsets of the vertices and represent which areas are inhabited by a given species. The goal is to find a set F of at most k edges such that in the subgraph induced by F, each habitat forms a connected component. This way the animals can reach their whole habitat without crossing non-habitat patches of land. See Figure 1.1 for an example. Fluschnik and Kellerhals [FK21] found 1-Reach GBP to be NP-hard already on planar graphs or graphs of maximum degree four.

In this thesis, we follow a purely theoretical approach and perform a more fine-grained analysis of the computational complexity than Fluschnik and Kellerhals [FK21] did. They did not analyze specific shapes of habitats and the shapes that did emerge in their constructions seem artificial. The idea of this thesis is based on the assumption that wildlife habitats do not have completely random shapes. We investigate how different shapes of the habitats affect the computational complexity of 1-Reach GBP in the hope of finding tractable cases. In the end, we focus on planar graphs as they are very well motivated when the input graph represents an area of land.

Table 1.1: Our results for the computational complexity of 1-Reach GBP with input graph G, when every habitat induces a graph from the family $\mathcal{F}$. We denote by $\mathcal{T}$ the family of trees, by $\mathcal{C}$ the family of cycles and by $\mathcal{G}$ the family of all graphs. By P_2 we denote a path with two vertices and by C_l a cycle of l vertices. The maximum vertex degree of the input graph is $\Delta(G)$.

Habitat family $\mathcal{F}$	Further restrictions	Complexity	Ref.
$\{P_2, C_3\}$ $\{P_2, C_l\}$ for each $l \geq 4$	G is a clique $\Delta(G) = 4$	NP-hard NP-hard	Thm. 3.1
$\{P_2\} \cup \mathcal{C}$	$\Delta(G) \leq 2$	poly. time	Prop. 4.2
$\mathcal{C}$ $\{C_l\}$ for each even $l \geq 4$ $\{C_l\}$ for each odd $l \geq 7$	none G is planar G is planar	NP-hard NP-hard NP-hard	Thm. 3.2
$\{C_3\}$	G is plane and each habitat induces a face	poly. time	Thm. 4.1
$\mathcal{C}$	In every habitat-induced cycle there is one edge that is disjoint with every other habitat induced cycle.	poly. time	Prop. 2.2
$\mathcal{T} \cup \mathcal{C}$	G is planar and $\Delta(G) = 3$	NP-hard	Prop. 3.6
$\mathcal{T}$	none	poly. time	Prop. 4.1
$\mathcal{G}$	The habitat-induced graphs are all pairwise edge disjoint.	linear time	Prop. 2.1

Structure and Contributions of this Thesis. Table 1.1 summarizes all results, sorted by habitat shape, and Table 1.2 summarizes all constructions that we used to prove NP-hardness.

We start in Section 3.1 by restricting the habitats such that they induce either a P_2 or cycle of fixed length and find that 1-Reach GBP is NP-hard, even when the input graph is a clique (Section 3.1.1), or when input graph G has maximum degree $\Delta(G) = 4$ (Section 3.1.2).

Since the input graph models an area of land, our problem is best motivated on planar graphs which are our focus in Sections 3.2.2 and 3.2.3. We find that 1-Reach GBP is also NP-hard on planar and series-parallel graphs with habitats that each induce a cycle of fixed length. In contrast, if the input graph is a tree, then we can solve it in polynomial time (Section 4.1). As series-parallel graphs have a treewidth of at most two[BVS99] and trees have a treewidth of one, this results in a dichotomy.

We find another dichotomy regarding the maximum degree of the input graph. In Section 3.3, we find that 1-Reach GBP is also NP-hard with an input graph that is planar and has maximum degree $\Delta(G) = 3$. Whereas, if we restrict the maximum degree of the input graph to two, 1-Reach GBP can be solved in polynomial time (Section 4.2).

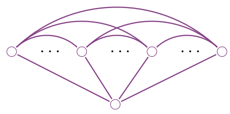
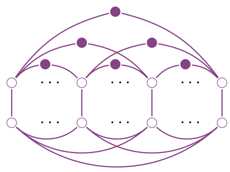
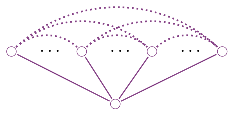
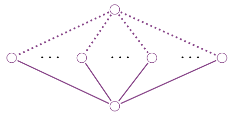
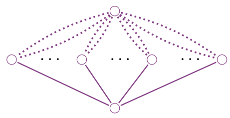
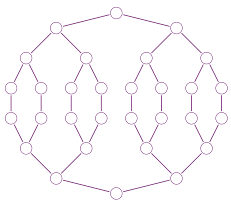
In Section 4.3 we present a polynomial-time algorithm to solve 1-Reach GBP with an (embedded) planar input graph and habitats that each induce a C_3 that is a face. To solve it we use a second graph H that is inspired by the dual graph of the input graph. It contains a vertex for each habitat, and two vertices are connected by an edge if the two habitats both induce the same edge in the original graph. Some extra vertices are added such that each edge in H represents a habitat-induced edge in the original graph. We use the fact that we can remove an edge from a cycle while still keeping the vertices connected and the fact that a matching contains at most one edge that is incident with a given vertex. By finding a maximum cardinality matching we then find the minimum cardinality solution for an instance of 1-Reach GBP.

Related work. Lai et al. [Lai+11] studied minimum-cost wildlife corridor design with a model based on the Steiner Tree problem. In their Steiner Multigraph problem they took into account the cost of land and that different species prefer habitats with different properties, for example altitude or temperature. For each species they have a set of terminals and a subset of vertices that can be used to connect the terminals. The goal is to find a subgraph that connects all terminals through their species specific set of vertices. They found that the problem is NP-hard, even for only two terminal sets with two vertices each. In the resulting solution all habitats may be connected but possibly at a great distance from another.

LeBras et al. [LeB+13] proposed the Minimum Delay Generalized Steiner Network problem, where they focus on redundant paths that connect the terminals to ensure robustness. Their goal was to find a way to connect the core habitat areas in a way that climate and man-made change of the environment will not fragment it again. The problem turned out to be NP-hard on planar graphs or when restricted to connecting only a single pair of nodes. They presented an approach formulating it as a mixed integer linear programming problem, but it does not scale well to larger problem instances from the real world.

Dilkina et al. [Dil+13] summarized different approaches for computing conservation plans that used Steiner problems. They tested them on data sets of grizzly bears, canadian lynx, and wolverines in the areas of the Yellowstone, Northern continental Divide Ecosystem, and Salmon Selway, as well as on data generated by a habitat generator that includes resistance or permeability for a given species to evaluate different solution methods.

Table 1.2: This table shows a list of constructions and sketches of the output graph for each construction we used, with information about the properties of the input graphs and the habitat shapes. The maximum degree is either bounded by the value in the table or unbounded (unb.) by a constant. (Constr. 6 has a checkmark for P_2 for the two habitats that induce trees, because they can easily be defined as a set of P_2's.)

Description	Each habitat induces		Input graph		Output graph
	P_2	C_l	$\Delta(G)$	planar	(Illustration)
G is a clique. (Construction 1)	✓	$l = 3$	unb.	×	
G with $\Delta(G) = 4$. (Construction 2)	✓	$l \geq 4$	4	×	
Parachute for C_l with crowns. (Construction 3)	×	$l \geq 3$	unb.	×	
Diamond with crowns. G is planar. (Construction 4)	×	even $l \geq 4$	unb.	✓	
Diamond with circles of uneven lengths. (Construction 5)	×	odd $l \geq 7$	unb.	✓	
Binary trees. (Construction 6)	✓	✓	3	✓	

2 Preliminaries

Notation. In this thesis, we use basic graph theory notation [Die05]. Let $\mathbb{N}$ be the natural numbers including zero and let $\mathbb{N}_0$ be the natural numbers excluding zero. Let $G = (V, E)$ be an *undirected graph* with a *vertex set* V and *edge set* $E \subseteq \{\{v, w\} \mid v, w \in V, v \neq w\}$. With $V(G)$ and $E(G)$ we denote the vertex sets and the edge sets of a graph G, respectively. For an edge $e = \{v, w\}$, the vertices v and w are its *endpoints*. The *neighborhood* of a vertex v in a graph G is defined by $N_G(v) = \{w \in V(G) \mid \{v, w\} \in E(G)\}$ and the number of neighbors of v is the *degree* $\deg_G(v)$. The *maximum degree* of a vertex in a graph is denoted by $\Delta(G) = \max_{v \in V} \deg_G(v)$. The *domination number* of a graph is the size of the smallest *dominating set*, that is a set where every vertex is either adjacent to a vertex in the set or is itself part of it. The *order* of a graph is the number $|V(G)|$ of vertices of the graph. For $x \in \mathbb{N}$, $x \geq 2$ we use the notation P_x for a *path of order* x, that is defined as a connected graph $P_x = (V, E)$ with $V = \{1, \ldots, x\}$ and $E = \{\{v, w\} \in \binom{V}{2} \mid w = v + 1\}$. A *cycle* C_x, $x \geq 3$, is a P_x with an additional edge $\{1, x\} \in E(C_x)$ connecting the first and last vertex to close the cycle. The subgraph *induced* by a subset of the vertices $V' \subseteq V$ of a graph G is $G[V'] = (V', \{\{v, w\} \in E(G) \mid v, w \in V'\})$. The subgraph induced by a subset of edges $E' \subseteq E$ in G is $G[E'] = (\bigcup_{e \in E'} e, E')$. A graph is *planar* if it can be drawn in the plane with no edges crossing except for their endpoints. A *plane* graph is a planar graph that is embedded in the plane, meaning we know how to draw it in a plane with no edges crossing. A *face* of a plane graph is an area on the plane that is bounded by edges of the graph, with no edge crossing through it. Let $F \subseteq E$ be an edge subset and $Z \subseteq V$ be a vertex subset. We call a habitat $Z \subseteq V$ in a graph G *connected* (by a solution F) if $G[F][Z]$ is a connected component, where $G[F][Z] = H[Z]$ with $H = G[F]$.

First Observations

Here we present some cases where the solutions can be easily found as well as a property of the input graph that has no effect on the computational complexity of 1-Reach GBP.

The smallest set of edges that connects a habitat induces a tree. So if a habitat induces a tree, then that tree is the only solution for said habitat.

Lemma 2.1. *Let $\mathcal{I} = (G, \mathcal{H}, k)$ be an instance of* 1-Reach GBP *and let $Z \in \mathcal{H}$ be a habitat such that $G[Z]$ is a tree T. Then for every solution F of $\mathcal{I}$, $E(T) \subseteq F$, that is, every edge of the tree induced by Z has to be included in F.*

Proof. For an edge $e = \{x, y\} \in E(T)$ suppose there is a set $F' \subseteq E(G) \setminus \{e\}$ that is a solution of $\mathcal{I}$. This means, that there is a path in $G[F'][Z]$ connecting any two vertices of Z, in particular a path from x to y.

But we know that e is not included in this path and $(\{x, y\}, \{e\})$ is already a path from x to y. This contradicts that $G[Z]$ is a tree. So there is no solution that excludes the edge e. Thus, every solution has to include all edges in $E(T)$. □

If a habitat stands alone, then all we have to do is to find a set of edges that connects all its vertices. This leads to the following proposition.

Proposition 2.1. *An instance $\mathcal{I} = (G, \mathcal{H}, k)$ of* 1-Reach GBP*, where all habitat-induced subgraphs $G[Z]$, $Z \in \mathcal{H}$, are connected and pairwise edge-disjoint, can be solved in linear time by computing a spanning tree for each habitat independently.*

Proof. A spanning tree connects all vertices of a habitat with the smallest set of edges and since the habitats are all edge disjoint, every habitat can be solved independently.

We can find a spanning tree for each habitat $Z \in \mathcal{H}$ by doing a breadth-first search in $\mathcal{O}(|Z| + |E(G[Z])|)$ time, where a maximum of $|E|$ edges will be visited in total. If no solution exists, then the BFS will stop before visiting all vertices of a habitat. □

When a habitat induces a cycle that does not completely overlap with other habitats, then we can easily find a solution for it. As a result we get the following.

Proposition 2.2. 1-Reach GBP *can be solved in $\mathcal{O}(mrn)$ time if each habitat induces a cycle and at least one edge of every habitat-induced cycle is not included in any other habitat-induced cycle.*

Proof. To connect a habitat that induces a C_l, any $l - 1$ edges of that cycle can be used. They form a path through all vertices of the habitat, that connects them with the smallest number of edges. This leaves only one edge out of the solution. We minimize the solution by making sure that for every habitat the edge that is not included in the solution is not part of another habitat-induced cycle. So for every habitat we look for an edge that is part of the cycle induced by only this one habitat.

Note that the input graph has n vertices, m edges and r habitats. To find a solution, we can iterate through all edges ($\mathcal{O}(m)$ time). For every edge we iterate through every habitat ($\mathcal{O}(rn)$ time). In a list for every habitat we save all edges that have both endpoints in the habitat as well as the number of habitats an edge is in. For every habitat choose one edge that is included in only one habitat ($\mathcal{O}(rm)$ time) and add all other edges of the habitat to the solution F. This algorithm has a running time of $\mathcal{O}(mrn + rm) = \mathcal{O}(mrn)$ time. □

Next we show that every instance of 1-Reach GBP can be augmented to have a diameter of two and domination number one by adding a vertex that is incident with all other vertices. This new instance is as hard to solve as the original one. This means that even if the input graph is restricted to have these properties we presumably cannot find an efficient way to solve it.

Proposition 2.3. *Let* $\mathcal{I} = (G, \mathcal{H}, k)$ *be an instance of* 1-Reach GBP *with* $G = (V, E)$ *and let* $\mathcal{I}' = (G', \mathcal{H}, k)$ *be the instance of* 1-Reach GBP *where* $G' := (V \cup \{s\}, E \cup \{\{s, i\} \mid i \in V\})$*. Then* $\mathcal{I}$ *is a* `yes`*-instance if and only if* $\mathcal{I}'$ *is a* `yes`*-instance.*

Proof. ($\Rightarrow$) Let F be a solution to $\mathcal{I}$. We claim that F is also a solution to $\mathcal{I}'$. As G' includedes all vertices and all edges of G, it holds that $G'[F] = G[F]$. In addition, s is not included in any habitat. Therefore every solution of $\mathcal{I}$ is also a solution of $\mathcal{I}'$.

($\Leftarrow$) Let F be a minimal solution to $\mathcal{I}'$. We claim that $F \subseteq E$. Suppose not, then there is an edge $e \in F \cap \{\{s, i\} \mid i \in V\}$ that only exists in G'. Since F is a minimal solution, there is a $Z \in \mathcal{H}$ for which $e \in G'[F][Z]$. For this to be true, both endpoints of e have to be in Z. This contradicts that $e \in \{\{s, i\} \mid i \in V\}$. Thus, $F \subseteq E$ which means $G'[F] = G[F]$, and F is a solution to $\mathcal{I}$. $\square$

3 NP-hard cases

The problem 1-Reach GBP turns out to be already NP-hard in many cases of restricted structures for habitats. We show this by reducing the NP-hard 3-regular Vertex Cover [GJS74] to different variations of 1-Reach GBP.

> 3-regular Vertex Cover
> **Input:** An undirected graph G where for every $v \in V(G)$, $\deg_G(v) = 3$, and an integer $k \in \mathbb{N}$.
> **Question:** Is there a set $S \subseteq V(G)$ of at most k vertices such that every edge in $E(G)$ has at least one endpoint in S?

In this chapter we examine cases of 1-Reach GBP where the habitats induce either a P_2 or a C_l, in Section 3.1. In Section 3.2 the habitats can only induce a C_l and in Section 3.3 we restrict the graph to be planar and have a maximum degree of three.

3.1 When habitats induce an edge or a cycle of fixed length

In this section, we consider the computational complexity of instances of 1-Reach GBP where the habitats each induce a path of order two or cycles of fixed length and comply to further restrictions.

Theorem 3.1. 1-Reach GBP *is NP-hard even if each habitat induces a P_2 or a C_l*

(i) if $l = 3$ and the input graph is a clique. (Proposition 3.1)

(ii) for every fixed $l \in \mathbb{N}$ with $l \geq 4$ and with $\Delta(G) = 4$. (Proposition 3.2)

3.1.1 Habitats induce either a P_2 or a C_3 and the input graph is a clique

In this section, we prove the following.

Proposition 3.1. 1-Reach GBP *is* NP*-hard even if each habitat induces either a P_2 or a C_3, and the input graph is a clique.*

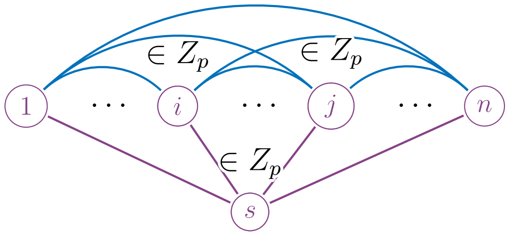


Figure 3.1: Graph G' as constructed from Construction 1, with the edges E that are included in every solution shown in blue. Herein, $e_p = \{i, j\} \in E$.

Construction 1. For an instance $\mathcal{I} = (G, k)$ of 3-REGULAR VERTEX COVER with $G = (V, E)$, $V = \{1, \ldots, n\}$ and $E = \{e_1, \ldots, e_m\}$, let $k' := |E| + k$ and construct an instance $\mathcal{I}' := (G', \mathcal{H}, k')$ of 1-REACH GBP as follows (see Figure 3.1 for an illustration). First construct the graph $G' := (V', E')$ with vertices $V' := V \cup \{s\}$ and edges $E' := \{\{i, j\} \mid i \neq j \in V\} \cup \{\{s, i\} \mid i \in V\}$. Then construct the habitats $\mathcal{H} := Z \cup E$ with $Z := \{Z_1, \ldots, Z_m\}$ and $Z_j := \{s, x, y \mid e_j = \{x, y\}\}$ for every $j \in \{1, \ldots, m\}$. ◇

The next two observations are immediate consequences from G' being a clique and Lemma 2.1.

Observation 3.1. *Every habitat induces a P_2 or a C_3.*

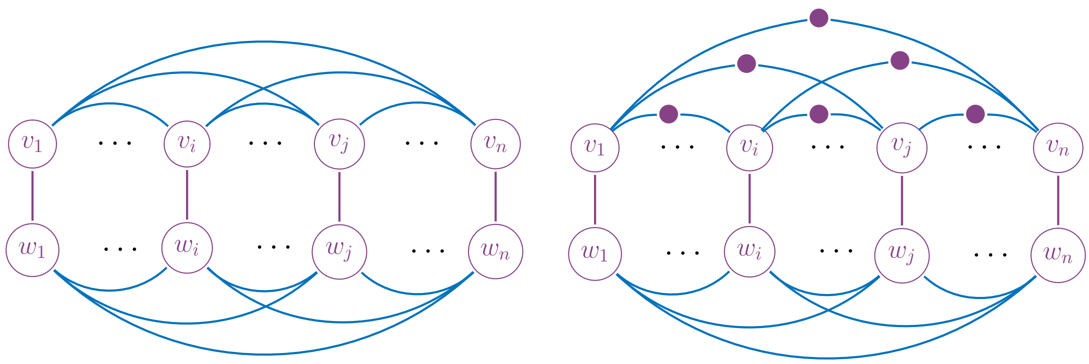
Observation 3.2. *Let $\mathcal{I}'$ be a* `yes`*-instance. Then every solution contains all edges in E.*

Next we show the equivalence of the two instances.

Lemma 3.1. *Let $\mathcal{I}'$ be the instance of* 1-REACH GBP *obtained from an instance $\mathcal{I}$ of* 3-REGULAR VERTEX COVER *using Construction 1. Then, $\mathcal{I}'$ is a* `yes`*-instance if and only if $\mathcal{I}$ is a* `yes`*-instance.*

Proof. ($\Leftarrow$) Let $S \subseteq V$ be a vertex cover of G of size at most k. We claim that $F := E \cup \{\{s, i\} \mid i \in S\}$ is a solution of $\mathcal{I}'$. Note that $|F| \leq |E| + k$. Assume towards a contradiction that there is a habitat $Z_j \in \mathcal{H}$ where $G'[F][Z_j]$ is not a connected component. As $E \subseteq F$, it must be some Z_j with $e_j = \{x, y\}$ that is not connected by the edges in F. This means that both the edges $\{s, x\}$ and $\{s, y\}$ are not included in F. This is a contradiction to S being a vertex cover because for the edge $e_j = \{x, y\} \in E(G)$ a least one of the two vertices x, y has to be in a vertex cover S. Hence, F is a solution.

($\Rightarrow$) Let F be a solution to $\mathcal{I}'$. By Observation 3.2, it holds that $E \subseteq F$. We claim that $S := \{i \in V \mid \{s, i\} \in F\}$ is a vertex cover of G. We know that $|S| \leq k$, because $|F| \leq |E| + k$ and there are at most k edges $\{s, i\} \in F$. Suppose there is an edge $e_i = \{x, y\} \in E$ with $e_i \cap S = \emptyset$. Then $\{s, x\}$ and $\{s, y\}$ are both included



(a) The graph G' if $l = 4$. (b) The graph G' if $l = 5$.

Figure 3.2: Examples of G' obtained from Construction 2. All blue edges are included in every solution.

in F and $G'[F][Z_i]$ is not a connected component. This contradicts that F is a valid solution. Hence, S contains at least one endpoint of every edge. Thus, S is a vertex cover. □

3.1.2 Habitats induce either a P_2 or a C_l and $\Delta(G) = 4$

In this section, we prove the following.

Proposition 3.2. *For every fixed $l \in \mathbb{N}$ with $l \geq 4$, 1-*Reach *GBP is NP-hard even if $\Delta(G) = 4$ and each habitat induces either a P_2 or a C_l.*

Construction 2. For an instance $\mathcal{I} = (G, k)$ of 3-Regular Vertex Cover with $G = (V, E)$, $V = \{1, \ldots, n\}$ and $E = \{e_1, \ldots, e_m\}$, construct an instance $\mathcal{I}' := (G', \mathcal{H}, k')$ of 1-Reach GBP as follows (see Figure 3.2 for illustrations).

Let $k' := (l-2)|E| + k$. Let $G' := (V' \cup E')$ and construct vertex sets $V_G := \{v_i \mid i \in V\}$, $W_G := \{w_i \mid i \in V\}$ and the edge set $E_G := \{\{v_i, w_i\} \mid i \in V\}$. Add V_G and W_G to V' and E_G to E'.

For every edge $e_i = \{x, y\} \in E$, add the edge $\{w_x, w_y\}$ and add a path P_i from v_x to v_y with $l - 4$ new vertices between. Notice that P_i in the case of $l = 4$ is just an edges between v_x and v_y.

Finally, construct the habitats $\mathcal{H} := (E' \setminus E_G) \cup Z$ with $Z := \{Z_i, \ldots, Z_m\}$ where $Z_i := V(P_i) \cup \{w_x, w_y\}$ for every edge $e_i = \{x, y\} \in E$. ◇

Observation 3.3. *Every habitat induces a P_2 or a C_l.*

Proof. The habitats $\mathcal{H}$ consist of $E' \setminus E_G$ which all induce a P_2 and of $Z_i = V(P_i) \cup \{w_x, w_y\}$ for every edge $\{\{x, y\} \in E\}$. The path P_i has $l - 3$ edges and together with the edges $\{v_x, w_x\}$, $\{w_x, w_y\}$, $\{w_y, v_y\}$ it forms a C_l. □

Due to Lemma 2.1, if a habitat induces a tree, then all of its edges have to be included in the solution. Consequently we have the following.

Observation 3.4. *Let $\mathcal{I}'$ be a* yes*-instance. Then every solution F of $\mathcal{I}'$ contains all edges $E' \setminus E_G$.*

Lemma 3.2. *Let $\mathcal{I}'$ be the instance of* 1-Reach GBP *obtained from an instance $\mathcal{I}$ of* 3-regular Vertex Cover *using Construction 2. Then, $\mathcal{I}'$ is a* yes*-instance if and only if $\mathcal{I}$ is a* yes*-instance.*

Proof. ($\Leftarrow$) Let $S \subseteq V$ be a vertex cover of G of size at most k. We claim that $F \coloneqq (E' \setminus E_G) \cup \{\{v_i, w_i\} \mid i \in S\}$ is a solution of $\mathcal{I}'$. Note that $|E' \setminus E_G| = (l-2)|E|$ and $|S| \leq k$, so $|F| \leq k'$.

Suppose there is a habitat Z_i with $e_i = \{x, y\}$ that is not connected by the edges in F, that is $G'[F][Z_i]$ is not a connected component. This means the edges $\{v_x, w_x\}$ and $\{v_y, w_y\}$, are both not in F, which contradicts that S is a vertex cover. Thus F is a solution.

($\Rightarrow$) Let $F \subseteq E(G')$ be a solution of $\mathcal{I}'$ of size k'. We claim that $S \coloneqq \{i \mid \{v_i, w_i\} \in F\}$ is a vertex cover of size at most k of G. Note that by Observation 3.4 F contains $E' \setminus E_G$ with $|E' \setminus E_G| = (l-2)|E|$, so $|S| \leq k$.

Now suppose there is an edge $e_i = \{x, y\}$ with $e_i \cap S = \emptyset$. This contradicts the fact that F is a solution and that for every habitat $Z_j \in Z$ the induced graph $G'[F][Z_j]$ is connected. Thus S is a vertex cover. □

3.2 When habitats induce cycles of fixed length

In this section, we consider the computational complexity of 1-Reach GBP where the habitats each induce a cycle of fixed length and comply to further restrictions.

Theorem 3.2. 1-Reach GBP *is NP-hard even if each habitat induces a C_l*

1. *for every fixed $l \in \mathbb{N}$ with $l \geq 3$. (Proposition 3.3)*
2. *for every fixed and even $l \in \mathbb{N}$ with $l \geq 4$, even on planar and series-parallel graphs. (Proposition 3.4)*
3. *for every fixed and odd $l \in \mathbb{N}$ with $l \geq 7$, even on planar graphs. (Proposition 3.5)*

3.2.1 Cycles of fixed length

In this section, we prove the following.

Proposition 3.3. *For every fixed $l \in \mathbb{N}$ with $l \geq 3$,* 1-Reach GBP *is NP-hard even if each habitat induces a C_l.*

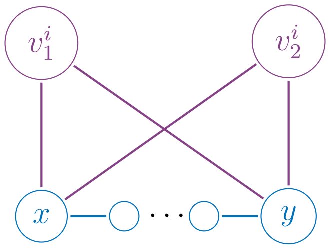


Figure 3.3: Sketch of a $(p, p+1)-$crown with the base path in blue.

We adapt Construction 1 in such a way that all habitats induce a C_3. To keep the reduction intact when removing the P_2 habitats, we replace every edge $e = \{x, y\} \in E$ with a gadget which we will call a crown. It is built as follows.

Definition 3.3 ((p, q)-Crown on x, y)**.** Let $G = (V, E)$ be a graph, $\mathcal{H}$ be a set of habitats and $x, y \in V$ be two non-adjacent vertices. Let $p, q \in \mathbb{N}_0$. Then inserting a (p, q)-crown on x and y is defined as follows (see Figure 3.3 for an illustration).

Construct the so-called *base path* P_B, a path of order p from x to y with a set $V^* := \{v_1, \ldots, v_{p-2}\}$ of new vertices between them. Next, construct the two disjoint *non-base paths* from x to y with inner vertices $V_i^C := \{v_1^i, \ldots, v_{q-p}^i\}$ for $i \in \{1, 2\}$, the *crown vertices*. Finally add the two *crown habitats* C_1 and C_2, one for every $i \in \{1, 2\}$ with $C_i := \{x, y\} \cup V^* \cup V_i^C$ to $\mathcal{H}$.

Definition 3.4 (crown solution)**.** Let $G = (V, E)$ be a graph to which a (p, q)-crown was added on two vertices $x, y \in V$. Then a *crown solution* is the subset of E that consists of the $p - 1$ edges of the base path and all edges of the two non-base paths except for one arbitrary edge for each of the two non-base paths.

Note that a crown solution consists of $2(q - p) + p - 1$ edges.

Lemma 3.3. *Let $\mathcal{I} = (G, \mathcal{H}, k)$ be an instance of* 1-Reach GBP *and let $\mathcal{I}'$ be the instance obtained by inserting a (p, q)-crown in $\mathcal{I}$, on vertices x and y of G. Then, every minimum solution of $\mathcal{I}'$ contains a crown solution.*

Proof. Note that the crown vertices are only included in the crown habitats, as they were newly added to $\mathcal{I}'$.

Let $e \in P_B$ be a base path edge. Suppose there is a minimum solution $F \subseteq E(G) \setminus \{e\}$. As the crown habitats are connected in $G[F]$, F contains $q - p + 1$ edges for every crown habitat plus the remaining $p - 2$ edges of the base path, together $2(q - p + 1) + p - 2$ edges.

Replacing one edge per crown habitat (that is not in the base path) with e reduces $|F|$ by 1. This reduces the number of edges needed to connect the crown habitats to $2(q - p) + p - 1$. Thus F was not a minimum solution and all edges of the base path have to be included in a minimum solution. □

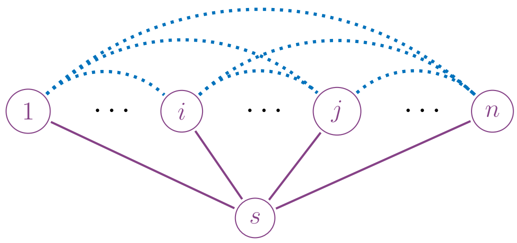


Figure 3.4: G' obtained from Construction 3. A dotted line represents an $(l-1,l)$-crown that is placed on the endpoints.

We now have everything at hand to provide the reduction from 3-REGULAR VERTEX COVER to 1-REACH GBP.

Construction 3. For an instance $\mathcal{I} = (G,k)$ of 3-REGULAR VERTEX COVER with $G = (V,E)$, $V = \{1,\ldots,n\}$ and $E = \{e_1,\ldots,e_m\}$, construct an instance $\mathcal{I}' = (G',\mathcal{H},k')$ of 1-REACH GBP as follows (see Figure 3.4 for an illustration).

Let $k' := l|E| + k$. Initially construct the graph $G' := (V',E')$, with vertices $V' := V \cup \{s\}$, edges $E' := \{\{s,i\} \mid i \in V\}$ and habitats $\mathcal{H} := \emptyset$. Then for every $e_i = \{x,y\} \in E$ insert a $(l-1,l)$-crown on x and y. Let B_i be the base path of the crown placed on e_i. Add $Z_i := \{s\} \cup V(B_i)$ as a habitat to $\mathcal{H}$. Now $\mathcal{H}$ includes the habitats Z_i and all the crown habitats. ◇

Observation 3.5. *Every habitat induces a C_l.*

Proof. The crown habitats of a (p,q)-crown per definition all induce a C_q. Here $(l-1,l)$-crowns are used, so the habitats induce a C_l. The rest of the habitats induce a cycle formed by the base path of a $(l-1,l)$-crown of length $l-2$ and two more edges from the endpoints of the base path to s. □

As a consequence of Lemma 3.3 we obtain the following.

Observation 3.6. *Every minimum solution F of instance $\mathcal{I}'$ contains a base path connecting x and y for every edge $\{x,y\} \in E$.*

Lemma 3.4. *Let $\mathcal{I}'$ be the instance of* 1-REACH GBP *obtained from an instance $\mathcal{I}$ of* 3-REGULAR VERTEX COVER *using Construction 3. Then, $\mathcal{I}'$ is a* `yes`*-instance if and only if $\mathcal{I}$ is a* `yes`*-instance.*

Proof. ($\Leftarrow$)Let $S \subseteq V$ be a vertex cover of G of size k. Let $e_i = \{x,y\} \in E$ be an edge in G and let F_i be a crown solution for the crown inserted on x and y in the construction of G'. We claim that $F := \{\{s,i\} \mid i \in S\} \cup \{F_i \mid e_i \in E\}$ is a solution of $\mathcal{I}'$. Note that the size of a crown solution is $2(q-p)+p-1$. For an $(l-1,l)$-crown that makes $2(l-(l-1))+(l-1)-1 = l$ and thus $|F| \leq l|E| + k$.

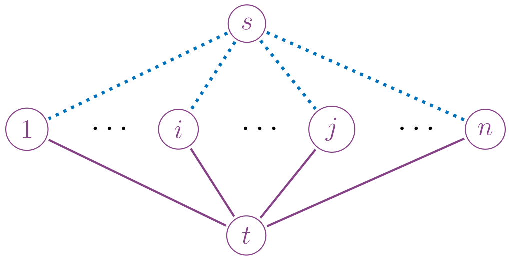


Figure 3.5: G' as constructed from Construction 4. A dotted line represents a crown that is placed on the endpoints.

Assume towards a contradiction that there is a habitat in $\mathcal{H}$ that is not connected by the edges in F. As the crown habitats are connected by the crown solutions, it must be some Z_j with $e_j = \{x, y\}$ that is not connected by the edges in F, that is, $G'[F][Z_j]$ is not a connected component. That means that both the edges $\{s, x\}$ and $\{s, y\}$ are not included in F. This is a contradiction to S being a vertex cover because for the edge $e_j = \{x, y\} \in E(G)$ at least one of the two vertices x, y has to be in a vertex cover S. Hence, F is a solution.

($\Rightarrow$) Let F be a minimum solution to $\mathcal{I}'$. We claim that $S := \{i \in V \mid \{s, i\} \in F\}$ is a vertex cover of G. We know that $|S| \leq k$, because $|F| \leq l|E| + k$ and for every edge in E there is a crown solution of size l included in F. Suppose there is an edge $e_i = \{x, y\} \in E$ with $e_i \cap S = \emptyset$. Then the corresponding habitat Z_i could not be connected by the edges in F and this contradicts that F is a valid solution. Hence S contains at least one endpoint of every edge. Thus S is a vertex cover. □

3.2.2 Cycles of even length in planar graphs

In this section, we prove the following proposition.

Proposition 3.4. *For every fixed and even $l \in \mathbb{N}$ with $l \geq 4$ 1-*Reach GBP *is NP-hard even on series-parallel graphs if each habitat induces a C_l.*

Construction 4. For an instance $\mathcal{I} = (G, k)$ of 3-regular Vertex Cover with $V = \{1, \dots, n\}$ and $E = \{e_1, \dots, e_m\}$, construct an instance $\mathcal{I}' = (G', \mathcal{H}, k')$ of 1-Reach GBP as follows (see Figure 3.5 for an illustration).

From now on we assume l is a fixed, even integer $l \geq 4$. Let $k' := (\frac{3}{2}l - 1)|V| + k$ and let $G := (V', E')$ with $V' := V \cup \{s, t\}$ and $E' := \{\{t, i\} \mid i \in V\}$. Add a $(\frac{l}{2}, l)$-crown on s and i for every $i \in V$ and let B_i be the base path of that crown.

In addition to the crown habitats let $\mathcal{H}$ include $\{Z_1, \dots, Z_m\}$, where for every $e_j = \{x, y\} \in E$ there is a habitat $Z_j := V(B_x) \cup V(B_y) \cup \{t\}$. ◇

Observation 3.7. *Every habitat induces a C_l.*

Proof. The crown habitats each induce a C_l by definition. The other habitats induce a cycle that consists of two base paths with $(\frac{l}{2} - 1)$ edges each, in addition to the 2 edges $\{s, x\}$ and $\{s, y\}$ for a habitat Z_i with $e_i = \{x, y\}$. □

Lemma 3.5. *Let $\mathcal{I}'$ be the instance of* 1-Reach GBP *obtained from an instance $\mathcal{I}$ of* 3-regular Vertex Cover *using Construction 4. Then, $\mathcal{I}'$ is a* yes*-instance if and only if $\mathcal{I}$ is a* yes*-instance.*

Proof. (⇐) Let $S \subseteq V$ be a solution to $\mathcal{I}$ of size $|S| \leq k$. For every $i \in V$ let F_i be a crown solution for the crown inserted on s and i. We claim that $F := \{\{s, i\} \mid i \in S\} \cup \bigcup_{i \in V} F_i$ is a solution to $\mathcal{I}'$. Now suppose that there is a habitat that is not connected by the edges in F. As the crown habitats are all connected by the crown solutions and the base path is part of every crown solution, there is a habitat Z_j with $e_j = \{x, y\}$ where both edges $\{s, x\}$ and $\{s, y\}$ are not in F. This contradicts the fact that S is a solution to $\mathcal{I}$. Hence if F is a solution to $\mathcal{I}'$.

(⇒) Let F be a minimum solution to $\mathcal{I}'$ of size k'. We claim that $S := \{i \mid \{s, i\} \in F\}$ is a solution to $\mathcal{I}$. Suppose there is an edge $e_j = \{x, y\}$ with $e_j \cap S = \emptyset$. That contradicts the fact that F is a solution to $\mathcal{I}'$ because $G'[F][Z_j]$ cannot be connected without $\{s, x\}, \{s, y\} \in F$. □

3.2.3 Cycles of odd length in planar graphs

This section is devoted to prove the following.

Proposition 3.5. *For every fixed odd $l \in \mathbb{N}$ with $l \geq 7$,* 1-Reach GBP *is NP-hard, even on a planar graph and if each habitat induces a C_l.*

In the next construction we use a new structure called a double-crown that is defined as follows.

Definition 3.5 ((p, q)-double-crown)**.** Let $G = (V, E)$ and $x, y \in V$. Then inserting a (p, q)-double-crown on x and y is defined as inserting a (p, q)-crown and a $(p + 1, q)$-crown on the vertices x and y, such that both crowns have an edge-disjoint base path. Let the shorter base path of the (p, q)-crown be B_S and the longer base path of the $(p + 1, q)$-crown be B_L.

Construction 5. For an instance $\mathcal{I} = (G, k)$ of 3-regular Vertex Cover with $G = (V, E)$, $V = \{1, \ldots, n\}$ and $E = \{e_1, \ldots, e_m\}$, construct an instance $\mathcal{I}' := (G', \mathcal{H}, k')$ as follows (see Figure 3.6 for an illustration). Let $k' := (3l - 2)|V| + k$ and initially let $G := (V', E')$ where $V' := V \cup \{s, t\}$ and $E' := \{\{t, i\} \mid i \in V\}$ and let $\mathcal{H} := \emptyset$. Then insert a $(\lfloor \frac{l}{2} \rfloor, l)$-double-crown on s and i for every $i \in V$. Finally, add a habitat Z_i for every edge $e_i = \{x, y\}$ with $x < y$ that consists of $\{t\}$ joined with $V(B_L)$ of the double-crown on s and x, $V(B_S)$ of the double-crown on s and y. ◇

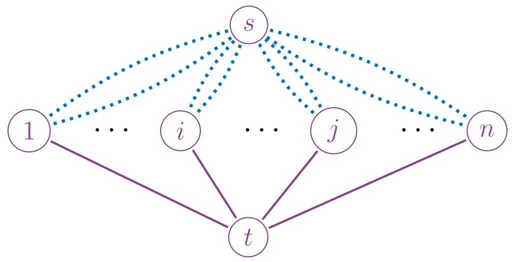


Figure 3.6: Illustration of G' obtained from Construction 5. A dotted line represents a crown that is placed on the endpoints.

Observation 3.8. *Every habitat induces a C_l.*

Proof. The crown habitats of the double crown each induce a C_l by definition. Every habitat Z_i with $e_i = \{x, y\} \in E$ induces a cycle that consists of two base paths with $\lceil \frac{l}{2} \rceil - 1$ and $\lfloor \frac{l}{2} \rfloor - 1$ edges, in addition to the 2 edges $\{t, x\}$ and $\{t, y\}$. Thus the all induce a C_l. □

Due to Lemma 3.3 we have the following.

Observation 3.9. *The base paths of all crowns inserted in G are included in every solution. All vertices in V are connected to s by the edges in the solution.*

Note that a solution for a double-crown consists of $(3l - 2)$ edges.

Lemma 3.6. *Let $\mathcal{I}'$ be the instance of* 1-Reach GBP *obtained from an instance $\mathcal{I}$ of* 3-regular Vertex Cover *using Construction 5. Then, $\mathcal{I}'$ is a* `yes`*-instance if and only if $\mathcal{I}$ is a* `yes`*-instance.*

Proof. ($\Leftarrow$) Let $S \subseteq V$ be a vertex cover of G of size at most k.

Let F_i be a crown solution for the double-crown inserted on i and s in the construction of G' where $i \in V$. We claim that $F \coloneqq \{\{s, i\} \mid i \in S\} \cup \{F_i \mid i \in V\}$ is a solution of $\mathcal{I}'$.

Assume towards a contradiction that there is a habitat in $\mathcal{H}$ that is not connected by the edges in F. As all crown habitats are connected by F, it must be Z_j with $e_j = \{x, y\}$ that is not connected by the edges in F, that is, $G'[F][Z_j]$ is not a connected component. That means that both the edges $\{t, x\}$ and $\{t, y\}$ are not included in F. This is a contradiction to S being a vertex cover because for the edge $e_j = \{x, y\} \in E(G)$ a least one of the two vertices x, y has to be in a vertex cover S. Hence, F is a solution.

($\Rightarrow$) Let F be a minimum solution to $\mathcal{I}'$. We claim that $S \coloneqq \{i \in V \mid \{t, i\} \in F\}$ is a vertex cover of G. We know that $|S| \leq k$, because $|F| \leq (3l - 2)|V| + k$, the size of a double-crown solution is $(3l - 2)$ and $|V|$ crowns were inserted in the construction. Suppose there is an edge $e_j = \{x, y\} \in E$ with $e_j \cap S = \emptyset$. Then the corresponding habitat Z_j could not be connected by the edges in F and this contradicts that F is a valid solution. Hence, S contains at least one endpoint of every edge. Thus S is a vertex cover. □

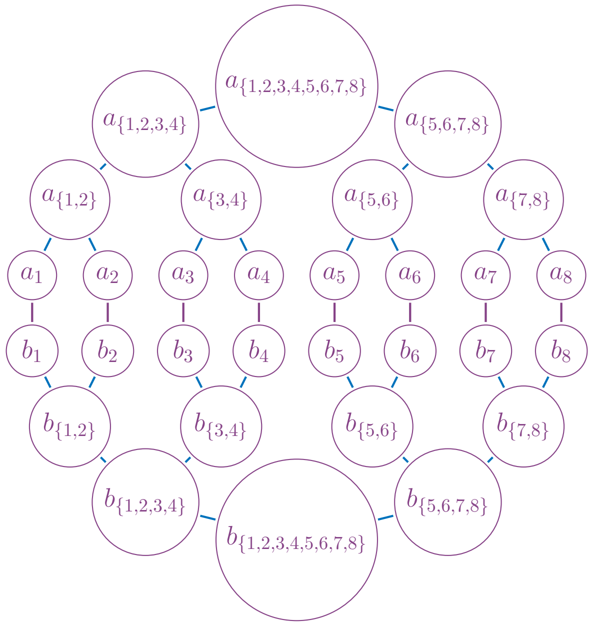


Figure 3.7: G' as constructed from Construction 6 with $5 \leq n \leq 8$.

3.3 Planar input graph that has a maximum degree of three

In this section, we show that 1-Reach GBP is NP-hard even if the input graph is planar and has maximum degree of three.

Proposition 3.6. 1-Reach GBP *is NP-hard, even on a planar graph with* $\Delta(G) = 3$ *and habitats that each induce either a tree or a cycle.*

For the reduction we need some additional notation.

Definition 3.6 (Complete binary tree of depth x)**.** Let a complete rooted binary tree of depth x be defined as follows. A binary tree is an acyclic graph where every vertex has a maximum of two child vertices. It is complete if all leaves are of the same distance to the root. A complete binary tree of depth x has 2^x leaves with the path from the root to each tree is of length x.

Definition 3.7 (Lowest Common Ancestor)**.** Let the *lowest common ancestor* ($LCA(x, y)$) of two vertices x and y in a binary tree be the vertex, of greatest distance to the root, that has both x and y as descendants.

Construction 6. For an instance $\mathcal{I} = (G, k)$ of 3-regular Vertex Cover with $G = (V, E)$, $V = \{1, \ldots, n\}$, and $E = \{e_1, \ldots, e_m\}$, construct an instance $\mathcal{I}' := (G', \mathcal{H}, k')$ of 1-Reach GBP as follows (see Figure 3.7 for an illustration).

Construct a complete binary tree T_1 of depth x where $2^{x-1} \leq n \leq 2^x$. Let the leaves be $L_1 := \{a_1, \ldots, a_{2^x}\}$ and the non-leaf vertices be a_C where $C := \{i \mid i \in V_L \text{ and } i \text{ is descendant of } a_C\}$. Let $k' := 2(2^{x+1} - 2) + k$. Now construct a second complete binary tree T_2 also of depth x, with leaves $L_2 := \{b_1, \ldots, b_{2^x}\}$. Analogous to T_1, let the non-leaf vertices of T_2 be named b_C where $C := \{i \mid i \in V_L \text{ and } i \text{ is descendant of } a_C\}$. Then add edges between the leaves of both trees such that $E' \supset E_L := \{\{a_i, b_i\} \mid i \in 2^x\}$. Add the sets $V(T_1)$ and $V(T_2)$ as habitats to $\mathcal{H}$. Finally construct a habitat Z_i for every edge $e_i = \{v, w\} \in E$ as $Z_i := \{a_v, a_w, b_v, b_w\} \cup \{a_C, b_C \mid |C \cap e_i| = 1\} \cup \{LCA(a_v, a_w), LCA(b_v, b_w)\}$. $\diamond$

Observation 3.10. *Every habitat induces a tree or a cycle.*

Proof. The graph G' was constructed by adding two trees T_1 and T_2 and then adding some edges to connect the two trees. None of the added edges connect two vertices of the same tree, so the habitats $V(T_1)$ and $V(T_2)$ still both induce a tree.

The other habitats form a cycle for every edge $\{x, y\}$ of the input graph. They consist of two paths from the LCA of a_x and a_y in one tree to the LCA of b_x and b_y in the other tree. One of which contains the edge $\{a_x, b_x\}$ and the other $\{a_y, b_y\}$. At the two LCAs they connect to form a cycle. □

Lemma 3.7. *Let $\mathcal{I}'$ be the instance of* 1-Reach GBP *obtained from an instance $\mathcal{I}$ of* 3-regular Vertex Cover *using Construction 6. Then, $\mathcal{I}'$ is a* `yes`*-instance if and only if $\mathcal{I}$ is a* `yes`*-instance.*

Proof. ($\Leftarrow$) Let $S \subseteq V$ be a vertex cover of G of size k. We claim that $F := E(T_1) \cup E(T_2) \cup \{\{a_i, b_i\} \mid i \in S\}$ is a solution of $\mathcal{I}'$.

Assume towards a contradiction that there is a habitat Z_i with $e_i = \{v, w\}$ that is not connected by the edges in F, that is, $G'[F][Z_i]$ is not a connected component. This means that the two edges $\{a_v, b_v\}$ and $\{a_w, b_w\}$ are not included in F. This is a contradiction to S being a vertex cover. Hence, F is a solution.

($\Rightarrow$) Let F be a solution to $\mathcal{I}'$. We claim that $S := \{i \in V \mid \{a_i, b_i\} \in F\}$ is a vertex cover of G. We know that $|S| \leq k$, because $|F| \leq 2(2^{x+1} - 2) + k$ and $|E(T_1)| + |E(T_2)| = 2(2^{x+1} - 2)$. Thus $|F \cap E_L| \leq k$ which means $|S| \leq k$. Suppose there is an edge $e_i = \{x, y\} \in E$ with $e_i \cap S = \emptyset$. Then the corresponding habitat Z_i could not be connected by the edges in F and this contradicts that F is a valid solution. Hence, S is a vertex cover. □

4 Tractable cases

We present some variants of 1-Reach GBP that are restricted in habitat shape or other graph properties such that we can solve them in polynomial time.

4.1 Each habitat induces a tree

In this section, we prove the following.

Proposition 4.1. *if each habitat induces a tree, then* 1-Reach GBP *can be solved in* $\mathcal{O}(m \cdot r + \sum_{Z \in \mathcal{H}} |Z|)$ *time.*

Lemma 4.1. *Let* $\mathcal{I} = (G, \mathcal{H}, k)$ *be an instance of* 1-Reach GBP *where every habitat* $Z_i \in \mathcal{H}$ *induce a tree* $T_i = G[Z_i]$*. Then* $F := \bigcup_{Z_i \in \mathcal{H}} E(T_i)$ *yields a correct solution to* $\mathcal{I}$*. As long as* $|F| \leq k$*, otherwise no solution exists.*

Proof. Due to Lemma 2.1, all edges of a tree induced by a habitat have to be included in every solution. Thus, constructing the solution as defined above is correct. □

Lemma 4.2. *The solution can be computed in* $\mathcal{O}(m \cdot r + n + \sum_{Z \in \mathcal{H}} |Z|)$ *time.*

Proof. To compute the solution we iterate over all habitats and save for every vertex which habitats it is included in ($\mathcal{O}(n + \sum_{Z \in \mathcal{H}} |Z|)$ time). Then we iterate over all edges ($\mathcal{O}(m)$ time) and check whether both endpoints of an edge are in the same habitat ($\mathcal{O}(r)$ time). If so, then said edge is added to the solution. This results in a running time of $\mathcal{O}(m \cdot r + n + \sum_{Z \in \mathcal{H}} |Z|)$. □

As a result of Proposition 4.1 we get the following observation for all instances with a tree as the input graph.

Observation 4.1. 1-Reach GBP *can be solved in* $\mathcal{O}(m \cdot r + \sum_{Z \in \mathcal{H}} |Z|)$ *time if the input graph is a tree.*

4.2 Input graph has maximum degree of at most two

The substructures of a graph G with $\Delta(G) \leq 2$ are limited to unconnected vertices, paths of order $\{2, \ldots, |V|\}$, and circles of order $\{3, \ldots, |V|\}$. The same limits hold for the structures a habitat can induce. This makes it possible to find a solution efficiently.

Proposition 4.2. *For an instance $\mathcal{I} = (G, \mathcal{H}, k)$ where $\Delta(G) \leq 2$,* 1-Reach GBP *can be solved in $\mathcal{O}(r(n+m) + rm^2)$ time.*

A solution can be constructed by applying the following steps in order.

1. For every habitat that induces a path add all edges of the path to F.
2. For every habitat that induces a cycle C, check if there is at least one edge of $E(C)$ that is not yet included in F. If so, let $e \in E(C) \setminus F$ be such an edge of C and add $C \setminus \{e\}$ to F.

   If not, then $E(C) \subseteq F$ and the habitat is already completely connected by the solution.

Lemma 4.3. *The solution constructed above is correct, as long as $|F| \leq k$. Otherwise no solution exists.*

Proof. Let $\mathcal{I} = (G, \mathcal{H}, k)$ be an instance of 1-Reach GBP and let F be a set of at most k edges, constructed according to the above rules. Suppose there is a habitat $Z \in \mathcal{H}$ that is not connected by F.

Every path is also a tree and as Lemma 2.1 shows, for every habitat that induces a tree all edges of that tree form a solution for that habitat. So Z cannot be one of the habitats inducing a tree.

Then Z induces a cycle. Since $G[F][Z]$ is not connected there are two edges $e, e' \in E(G[Z])$ that are not included in F. This contradicts that all but one edge of every cycle was added to F in the construction. Thus Z is connected by F and F is a solution of $\mathcal{I}$. □

Lemma 4.4. *The solution can be constructed in $\mathcal{O}(r(n+m) + rm^2)$ time.*

Proof. For every habitat $Z \in \mathcal{H}$ ($\mathcal{O}(r)$ time) do a breadth first search ($\mathcal{O}(n+m)$ time) that starts at any vertex of Z. To make sure only vertices of the current habitat are visited check for every vertex whether it is included in the habitat ($\mathcal{O}(n)$ time). For every habitat save all edges that are visited in a list.

If a vertex is found that has been visited before, then the habitat induces a cycle. If there are no more habitat vertices left to visit, then the habitat induces a path. Add all visited edges to the solution.

After all habitats have been searched, iterate through the lists of edges ($\mathcal{O}(rm)$ time) for every habitat that induces a cycle and check if there is an edge $e \in E$ that is not part of the solution yet ($\mathcal{O}(m)$ time). If such an edge is found add all edges in the list except e to the solution.

This yields a running time of $\mathcal{O}(r(n+m) + rm^2)$. □

4.3 Plane graph with face-inducing habitats

If the habitats all induce faces of a plane graph, then we can find a solution in polynomial time. We prove that in this section.

Theorem 4.1. 1-Reach GBP *can be solved in* $\mathcal{O}(mrn)$ *time if the input graph is a plane graph and every habitat induces a* C_3 *that is also an inner face of* G.

Let $\mathcal{I} = (G, \mathcal{H}, k)$ be an instance of 1-Reach GBP where G is a plane graph and every habitat induces a C_3 that is the boundary of a face of G. We construct a second graph $H \coloneqq (V_H, E_H)$ as follows (See Figure 4.1 for an illustration). For every habitat $Z_v \in \mathcal{H}$ create a vertex $v \in V_H$, which we will call a *habitat vertex*. For every edge $e \in E$ and two habitats $Z_v, Z_w \in \mathcal{H}$ where $e \in Z_v$ and $e \in Z_w$, add the edge $\{v, w\}$ between the two corresponding habitat vertices to H. For every edge $e \in E$ with only one habitat $Z_v \in \mathcal{H}$ where $e \in Z_v$, add a new *non-habitat vertex* to H and an edge connecting it to the habitat vertex v.

With this definition of H we get the following observation.

Observation 4.2. *For a graph* H *constructed as defined above there is a bijection* $b : E_H \to E$ *that maps each edge in* H *to an edge in* G *such that* $b(e) = e^*$ *for every edge* e *that is an endpoint of a habitat vertex* $v \in H$ *and an edge* $e^* \in G[Z_v]$.

Lemma 4.5. $\mathcal{I}$ *is a* `yes`*-instance if and only if there is a set* $E^* \subseteq E_H$ *of at most* k *edges, such that every habitat vertex is incident with at least two edges in* E^*.

Proof. ($\Rightarrow$) Let F be a solution of $\mathcal{I}$. As F is a solution, every habitat $Z_v \in \mathcal{H}$ is connected by the edges in F. That means that at least two of the three edges of the cycle induced by Z_v are included in F. All three of these edges have a dual edge in H. We claim that $E^* \coloneqq \{e \in E_h \mid b(e) \in F\}$ contains at least two edges incident with each habitat vertex. We know that $|E| \leq k$ because every solution of $\mathcal{I}$ has at most k edges.

Suppose there is a habitat vertex $v \in V_H$ which is incident with less than two edges in E^*. Then there are less than two edges in F that are included in the cycle induced by the habitat Z_v. That means the habitat Z_v cannot be connected by F, which is a contradiction to the fact that F is a solution. Hence E^* contains at least two edges incident with each habitat vertex.

($\Leftarrow$) Let $E^* \subseteq E_H$ be a set of at most k edges such that every habitat vertex is incident with at least two of the edges. We claim that $F \coloneqq \{b(e) \mid e \in E^*\}$ is a solution to $\mathcal{I}$.

Suppose there is a habitat $Z_v \in \mathcal{H}$ that is not connected by the edges in F. Then the habitat vertex v has fewer that two incident edges in E^*, which is a contradiction to the definition of E^*. Thus F is a solution. □

To find a solution to $\mathcal{I}$ we use the Maximum Cardinality Matching defined as follows.

> Maximum Cardinality Matching
> **Input:** An undirected graph G.
> **Output:** The biggest possible set of edges $M \subseteq E(G)$ such that every vertex $v \in V(G)$ is incident with at most one edge in M.

Lemma 4.6. *(i) Let M be a matching in H. Then for every habitat vertex $v \in V_H$ the set $E_H \setminus M$ contains at least two edges incident with v.*

(ii) Let $E' \subseteq E_H$ be a set of edges such that every habitat vertex $v \in V_H$ is incident with at least two edges in E'. Then $E_H \setminus E'$ is a matching.

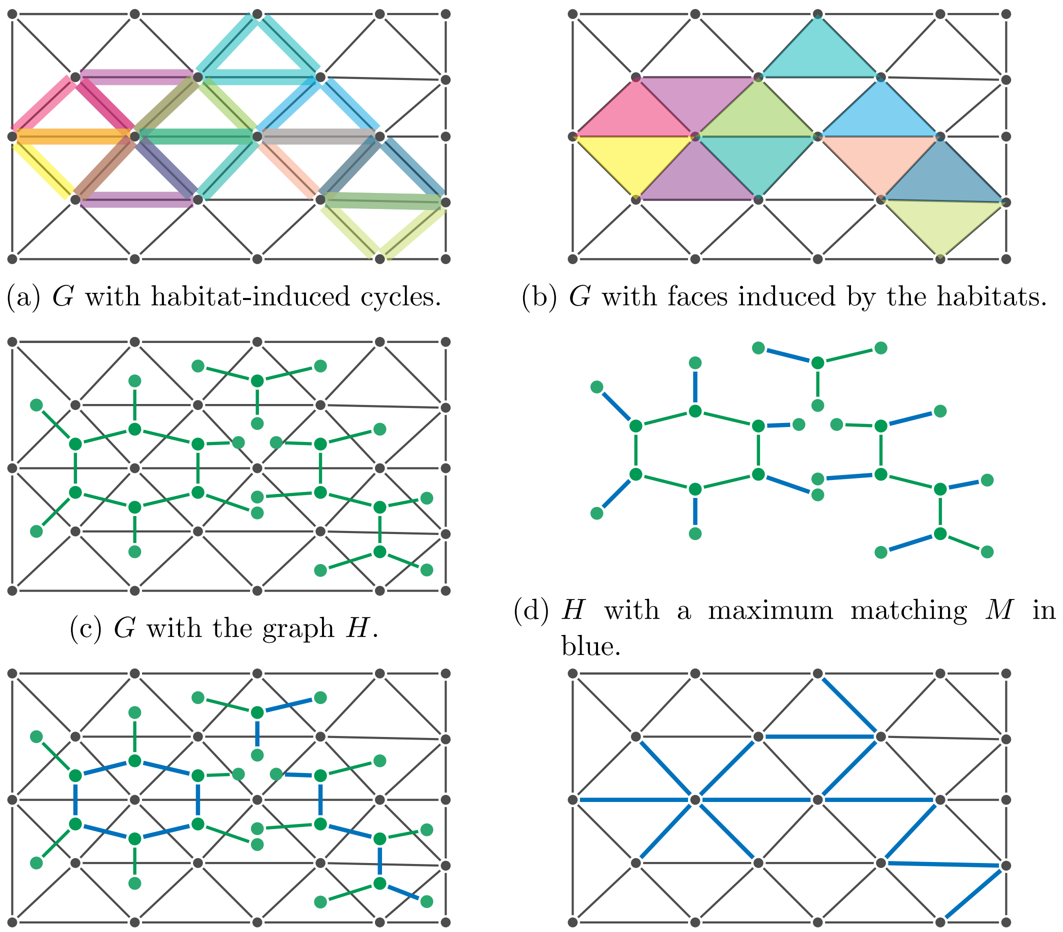
Proof. (i) The way the graph H is defined every habitat vertex $v \in V_H$ has a degree of three. By removing all edges of M at most one of the three edges incident with each habitat vertex are remover. It follows that at least two of the edges incident with each $v \in V_H$ remain.

(ii) Every habitat vertex is incident with at least two edges of E', so $E_H \setminus E'$ contains at most one edge incident with each habitat vertex. That makes $E \setminus E'$ a matching. □

Let $\mathcal{I} = (G, \mathcal{H}, k)$ be an instance of 1-Reach GBP where G is a plane graph and every habitat induces a C_3 that is the boundary of a face of G and let H be a graph constructed as defined above. Due to Lemma 4.6 we can find a set $E^* \subseteq E_H$ that contains at least two edges incident with each vertex in V_H by computing a maximum cardinality matching M. If $|E^*| > k$, then no solution exists, because the solution we find this way is a minimum solution. If it was not, then there would exist a larger matching. This contradicts that we used a maximum cardinality matching, hence we find a minimum solution.

Due to Lemma 4.5 the existence of this set E^* proves that $\mathcal{I}$ is a yes instance.

To use this algorithm we first have to construct the graph H. We can create the habitat vertices in $\mathcal{O}(r)$ time. To get the edges of H, iterate over all edges in E and then through all habitats. Creating the edges between two habitat vertices or creating an edge and a non-habitat vertex takes constant time. The whole graph H can be constructed in $\mathcal{O}(mrn)$ time. Finding a maximum matching can be done in in $\mathcal{O}(\sqrt{n}m)$ time [MV80]. Thus a solution is found in $\mathcal{O}(mrn)$ time.



(a) G with habitat-induced cycles.

(b) G with faces induced by the habitats.

(c) G with the graph H.

(d) H with a maximum matching M in blue.

(e) H with edge set $E_H \setminus M$ in blue.

(f) G with a minimum solution in blue.

Figure 4.1: Step-by-step illustration from input graph G to the graph H and the solution.

5 Conclusion

Discussion. We studied 1-Reach GBP with various restrictions to the shapes of habitats and also some restrictions to the input graph. We found that most cases are NP-hard, for example, if all habitats induce a C_3, but also that some cases can be solved efficiently, .

In the cases that proved to be NP-hard the interaction of the habitats seemed to be important. In all of them the habitats were all completely overlapping with other habitats. It made no difference to the complexity, if some of the habitats were trees or paths which can be solved trivially by choosing all edges of the induced subgraph.

In Section 4.3 we present a way to solve 1-Reach GBP if it has a plane input graph and the habitats each induce a C_3 that is a face. Here the habitats still overlap but only to a small extent. Every edge can be included in the induced cycle of at most two habitats. This leads to a clear distinction between edges that are shared by two habitats and non-shared edges, where a shared edge is always preferred.

For habitats that induce cycles with at least one edge that is not induced in any other habitat we found that it can be solved in polynomial time. This suggests that in a real-world scenario where natural habitats are not uniformly connected at least partial solutions may be found efficiently, where we select a number of edges that connect a subset of the habitats optimally.

We found that, with an input graph that has treewidth one, 1-Reach GBP can be solved in polynomial time, but with treewidth two it is already NP-hard. Moreover when the input graph has a maximum degree at most two we can solve 1-Reach GBP in polynomial time, but when the maximum degree is three it is NP-hard.

Outlook. We assume that we can use the same strategy used in Section 4.3 to also solve 1-Reach GBP on plane graph with habitats of any size, if each habitat induces a face. To do that, the graph H has to be augmented such that every habitat vertex has as many outgoing edges as the habitat has neighboring faces. Then, the matching will find a minimum solution as well. In general, all further research should focus on planar graphs as they are closest to real-world scenarios.

After focusing on 1-Reach GBP, we could continue the study of structure restrictions for d-Reach GBP, d-Closed GBP, and d--Diam GBP which were all defined in Fluschnik and Kellerhals [FK21].

The model we used is oversimplified in the way that we do not take into account the costs of buying land and building the bridges. Instead they are all treated as if they have equal costs. Hence, further research could include this information as edge costs. Also, if given some data of animal movements it may be more fitting to represent the habitats not as sets of vertices but as subgraphs instead. This way we would make sure to choose only edges that connect habitats where the animals actually cross from one patch to another.

In this thesis we did not investigate the parameterized complexity of 1-Reach GBP. In further work we could analyze the complexity regarding the number of habitats.

Literature

[And94] H. Andren. *Effects of Habitat Fragmentation on Birds and Mammals in Landscapes with Different Proportions of Suitable Habitat: A Review*. In: *Oikos* 71 (1994), pp. 355–366 (cit. on p. 7).

[BVS99] A. Brandstädt, Van Bang Le, and J. P. Spinrad. *Graph classes: a survey*. English. Vol. 3. Philadelphia, PA: SIAM, 1999, pp. xi + 304 (cit. on p. 9).

[Die05] R. Diestel. *Graph Theory*. Vol. 173. Graduate Texts in Mathematics. Springer, 2005 (cit. on p. 12).

[Dil+13] B. Dilkina, K. J. Lai, R. LeBras, Y. Xue, C. P. Gomes, A. Sabharwal, J. Suter, K. S. McKelvey, M. K. Schwartz, and C. A. Montgomery. *Large Landscape Conservation - Synthetic and Real-World Datasets*. In: *Proceedings of the Twenty-Seventh AAAI Conference on Artificial Intelligence*. AAAI Press, 2013 (cit. on pp. 7, 10).

[DOR15] B. Dorsey, M. Olsson, and L. J. Rew. *Ecological Effects of Railways on Wildlife*. In: *Handbook of road ecology* (2015), pp. 219–227 (cit. on p. 7).

[FK21] T. Fluschnik and L. Kellerhals. *Placing Green Bridges Optimally, with a Multivariate Analysis*. In: *Connecting with Computability - 17th Conference on Computability in Europe, CiE 2021*. Vol. 12813. Lecture Notes in Computer Science. Springer, 2021, pp. 204–216 (cit. on pp. 4, 7, 8, 31).

[GJS74] M. R. Garey, D. S. Johnson, and L. J. Stockmeyer. *Some Simplified NP-Complete Problems*. In: *Proceedings of the 6th Annual ACM Symposium on Theory of Computing*. ACM, 1974, pp. 47–63 (cit. on p. 15).

[Hui+16] M. P. Huijser, E. R. Fairbank, W. Camel-Means, J. Graham, V. Watson, P. Basting, and D. Becker. *Effectiveness of Short Sections of Wildlife Fencing and Crossing Structures Along Highways in Reducing Wildlife–vehicle Collisions and Providing Safe Crossing Opportunities for Large Mammals*. In: *Biological Conservation* 197 (2016), pp. 61–68 (cit. on p. 7).

[Lai+11] K. J. Lai, C. P. Gomes, M. K. Schwartz, K. S. McKelvey, D. E. Calkin, and C. A. Montgomery. *The Steiner Multigraph Problem: Wildlife Corridor Design for Multiple Species*. In: *Proceedings of the Twenty-Fifth AAAI Conference on Artificial Intelligence, AAAI 2011*. AAAI Press, 2011 (cit. on pp. 7, 10).

[LeB+13] R. LeBras, B. Dilkina, Y. Xue, C. P. Gomes, K. S. McKelvey, M. K. Schwartz, and C. A. Montgomery. *Robust Network Design For Multispecies Conservation*. In: *Proceedings of the Twenty-Seventh AAAI Conference on Artificial Intelligence*. AAAI Press, 2013 (cit. on pp. 7, 10).

[MV80] S. Micali and V. V. Vazirani. *An O(sqrt(|V|) |E|) Algorithm for Finding Maximum Matching in General Graphs*. In: *21st Annual Symposium on Foundations of Computer Science*. IEEE Computer Society, 1980, pp. 17–27 (cit. on p. 29).

[Poi+19] M. Poirier, D. Coltman, F. Pelletier, J. Jorgenson, and M. Festa-Bianchet. *Genetic Decline, Restoration and Rescue of an Isolated Ungulated Population*. In: *Evolutionary Applications* 12 (2019), pp. 1318 –1328 (cit. on p. 7).