Complexity of Graph Covering Problems*

Jan Kratochvíl
Charles University
Prague, Czech Republic

Andrzej Proskurowski
University of Oregon
Eugene, Oregon

Jan Arne Telle
University of Bergen
Bergen, Norway

Abstract

For a fixed graph H, the H-cover problem asks whether an input graph G allows a degree preserving mapping $f : V(G) \to V(H)$ such that for every $v \in V(G)$, $f(N_G(v)) = N_H(f(v))$. In this paper we design efficient algorithms for certain graph covering problems according to two basic techniques. The first is based in part on a reduction to the 2-SAT problem. The second technique exploits necessary and sufficient conditions for the partition of a graph into 1-factors and 2-factors. For other infinite classes of graph covering problems we derive $\mathcal{NP}$-completeness results by reductions from graph coloring problems. We illustrate this methodology by classifying the complexity of all H-cover problems defined by simple graphs H with at most 6 vertices.

1 Motivation and overview

For a fixed graph H, the H-cover problem admits a graph G as input and asks about the existence of a "local isomorphism": a labeling of vertices of G by vertices of H so that for every vertex $v \in V(G)$ the multiset of labels of its neighbors is equal to the neighborhood (in H) of the label of v. We trace this concept to Conway and Biggs' construction of infinite classes of highly symmetric graphs, see Chapter 19 of [3]. Graph coverings are special cases of covering spaces from algebraic topology [15] and have many applications in topological graph theory [6]. In an algorithmic framework, graph coverings have been applied by Angluin to study "local knowledge" in distributed computing environments [2] and by Courcelle and Métivier [4] to show that nontrivial minor closed classes of graphs cannot be recognized by local computations. Abello *et al.* [1] raised the question of computational complexity of H-cover problems, noting that there are both polynomial-time solvable (*easy*) and $\mathcal{NP}$-complete (*difficult*) versions of this problem for different graphs H. In [10] we initiated a general study of the computational complexity of H-cover problems. In a later paper [11]

*The preliminary version of this paper was presented at the 20th International Workshop on Graph-Theoretical Concepts in Computer Science, Herrsching, Germany, 1994 [10]

we made significant headway on the conjecture that H-cover problems are $\mathcal{NP}$-complete for k-regular graphs whenever $k \geq 3$. A very recent paper investigates the complexity of the graph covering problem for colored graphs [12].

In this paper, we extend and complete the methodology introduced in [10] to analyze the complexity of certain graph covering problems. The paper is organized as follows. First, we introduce our vocabulary by giving the necessary definitions in Section 2. In designing efficient algorithms that solve easy graph covering problems, we reduce those problems to regular factorization problems and/or to the 2-SAT problem. We present the corresponding results in Section 3. To prove $\mathcal{NP}$-completeness of the difficult graph covering problems, we use polynomial time reductions from known $\mathcal{NP}$-complete restrictions of vertex and edge coloring problems and also reductions between covering problems. These latter $\mathcal{NP}$-completeness results are based on properties of the automorphism groups of the relevant graphs. We set up a paradigm to construct such reductions and present our findings in Section 4. In the appendix, we give a catalogue of the complexity of the H-cover problem for all simple graphs H with at most 6 vertices. There are 208 such graphs, with about 100 defining covering problems with non-trivial polynomial time solution algorithms and 36 defining $\mathcal{NP}$-complete covering problems (the remaining graphs defining trivial covering problems).

2 Definitions

We use standard graph terminology [7], and consider simple, undirected graphs only. For a vertex $v \in V(G)$ of a graph G, let $N_G(v) = \{u : uv \in E(G)\}$ be the set of neighbors of v and $deg_G(v) = |N_G(v)|$ its degree. For $S \subseteq V(G)$ let $G[S]$ denote the graph induced in G by S, and let $G \setminus S = G[V(G) \setminus S]$. For $F \subseteq E(G)$, we denote by $G \setminus F$ the spanning subgraph of G with edges $\{uv \in E(G) : uv \notin F\}$. If $E(G) = \emptyset$ then G is called a *discrete* graph. The graph $\overline{G}$ has vertices $V(G)$ and edges $\{uv : uv \notin E(G)\}$. $Aut(G)$ is the automorphism group of G. By a *2-path* we mean a path whose inner vertices have degree 2 and whose end vertices (not necessarily distinct) have degrees greater than 2; the number of edges in the 2-path is its length.

A graph G is said to *cover* a graph H if there is a function (called *covering projection*) $f : V(G) \to V(H)$ which preserves the identity of the neighborhood of any vertex v of G, $\{f(u)|u \in N_G(v)\} = N_H(f(v))$ with $deg_G(v) = deg_H(f(v))$. Fixing the graph H, and allowing any graph G as the input, one can pose the question: "Does G cover H?" The computational complexity of this problem, called the *H-cover* problem for the particular graph H, is the subject of this paper.

The *degree partition* of a graph is the partition of its vertices into the minimum number of *blocks* $B_1, \ldots, B_t$ for which there are constants r_{ij} such that for each i, j $(1 \leq i, j \leq t)$ each vertex in B_i is adjacent to exactly r_{ij} vertices in B_j. The $t \times t$ matrix R $(R[i,j] = r_{ij})$ is called the *degree refinement*.

The degree partition and degree refinement of a graph are easily computed by a stepwise refinement procedure. Start with vertices partitioned by their degree values and keep refin-

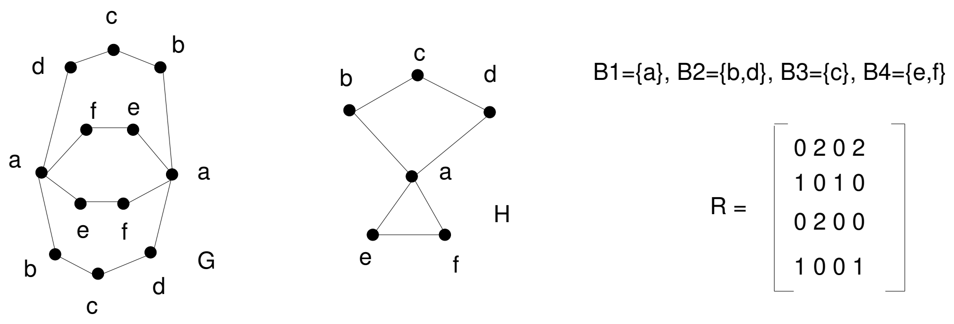


Figure 1: G labeled by a covering projection onto H, their common degree refinement R and the degree partition of H.

ing the partition until any two nodes in the same block have the same number of neighbors in any other given block. See Figure 1 for an example. Graph coverings are related to degree partitions and degree refinements (see, for instance, Leighton [13]):

Fact 2.1 *If f is a covering projection from G onto H then H and G have the same degree refinement and have degree partitions $B_1, B_2, ..., B_t$ and $B'_1, B'_2, .., B'_t$ so that for every $v \in B'_i$ we have $f(v) \in B_i$, $i = 1, 2, ..., t$.*

Without loss of generality, we will consider only connected graphs, because of the following observations (whose proofs are left to the reader.)

Fact 2.2 *Given a connected graph H, a graph G covers H if and only if every connected component of G covers H.*

Fact 2.3 *For a disconnected graph H, the H-cover problem is polynomially solvable ($\mathcal{NP}$-complete) if and only if the H_i-cover problem is polynomially solvable ($\mathcal{NP}$-complete) for every (for some) connected component H_i of H.*

3 Efficient algorithms

For a given graph G and a fixed graph H, it is easy to compare degree partitions and degree refinements in polynomial time. Surprisingly, for many graphs H, the necessary condition for the existence of a covering given by Fact 2.1 is also sufficient. For many other graphs H (including some infinite classes of graphs), for which those conditions are not sufficient, we are able to design an efficient solution algorithm paradigm by constructing an equivalent instance of the 2-SAT problem, and/or by reducing to a factorization problem in a regular graph. Before we present these results, we note that the only cover of a tree is a graph isomorphic to it, and the only covers of a cycle are cycles with lengths divisible by the length of the cycle. The following observation follows indirectly.

Fact 3.1 *For a graph H with at most one cycle, the H-cover problem is solvable in polynomial time.*

3.1 Factorization

A spanning subgraph H of a graph G is a *k-factor* if all vertices of H have degree k. When $k = 1$, the 1-factor is often referred to as a *perfect matching*. The existence of perfect matchings in bipartite graphs is a subject of the celebrated König-Hall theorem. A graph G is *k-factorable* if its edges can be partitioned into k-factors. An application of the König-Hall marriage theorem states that a regular bipartite graph is 1-factorable. A classical result of Petersen states that any $2k$-regular graph is 2-factorable. We will use these facts to show that the obvious necessary conditions are also sufficient for a class of graph covering problems.

Theorem 3.2 *Let H be a graph with all but two vertices of degree 2, and let these two vertices of higher degree be L and R. Further suppose that for every $i \geq 1$, H has l_i cycles of length i which contain L but not R, r_i cycles of length i which contain R but not L, and m_i paths of length i between L and R. Then a graph G covers H if and only if the vertices of degree > 2 in G can be partitioned into classes U and V so that for every i,*

1) every vertex of U is incident with m_i 2-paths of length i ending in V,

2) every vertex of U is incident with $2l_i$ 2-paths of length i ending in U (a cycle through one vertex of U is counted as two 2-paths),

3) every vertex of V is incident with m_i 2-paths of length i ending in U,

4) every vertex of V is incident with $2r_i$ 2-paths of length i ending in V.

Proof. Suppose f is a covering projection of G onto H. Set $U = f^{-1}(L)$ and $V = f^{-1}(R)$, conditions 1-4) then obviously hold. We will prove that these obvious necessary conditions are also sufficient.

Suppose on the other hand that (U, V) is a partition satisfying 1-4). We will set $f(x) = L$ for $x \in U$ and $f(x) = R$ for $x \in V$. It remains to show that the vertices of degree 2 can be also covered properly. Consider a particular value of i and the subgraph of G induced by U, V and the vertices of degree 2 on the 2-paths of length i. We construct an auxiliary multigraph G^i with vertex set $U \cup V$ and with an edge edge $xy \in E(G^i)$ iff x and y are connected by a 2-path of length i in G. It follows from 1) and 3) that the bipartite multigraph $G^i(U, V)$ induced in G^i by the edges between U and V is m_i-regular. Similarly, 2) implies that the subgraph $G^i(U)$ induced on U is $2l_i$-regular, and 4) implies that $G^i(V)$ is $2r_i$-regular.

The factorization theorems of Petersen yield that $G^i(U)$ can be partitioned into l_i 2-factors, $G^i(V)$ into r_i 2-factors and $G^i(U, V)$ into m_i 1-factors. The vertices of the k-th 2-factor in $G^i(U)$ (resp. $G^i(V)$) will map in the obvious way onto the k-th cycle around L (resp. R). The vertices of the k-th 1-factor of $G^i(U, V)$ will map onto the vertices of the k-th 2-path between L and R. ∎

Theorem 3.3 *Let H be a graph as in Theorem 3.2. If*

a) $l_i \neq r_i$ for some i, or

b) $m_i = 0$ for every i such that $l_i = r_i > 0$

then there is at most one partition (U, V) satisfying (1-4) of Theorem 3.2 and it can be found, if it exists, in polynomial time. Consequently, the H-cover problem is polynomially solvable.

Proof. We may assume that the input graph G has the same degree refinement as H (this can be checked in polynomial time). Each of the conditions a) and b) implies that the partition of the vertices of degree > 2 in G into U and V is unique: In case a), L and R belong to different classes of the degree partition of H and therefore are distinguishable in G as well. In case b), we may assume that $l_i = r_i$ for every i. Then every path of length i such that $l_i \neq 0$ connects vertices from the same class of the partition and every path of length j such that $m_j \neq 0$ connects vertices from different classes. The uniqueness of the (U, V) partition then follows from the connectedness of G, and this partition can be found by breadth first search through G starting in an arbitrary vertex. ∎

Corollary 3.4 *Let H be a graph as in the assumption of Theorem 3.2 and let H' be a graph obtained from H by pending trees T_L, T_R rooted in vertices L and R, respectively. If*

a) the rooted trees T_L, T_R are nonisomorphic, or

b) $l_i \neq r_i$ for some i, or

c) $m_i = 0$ for every i such that $l_i = r_i > 0$

then there is at most one partition (U, V) satisfying (1-4) of Theorem 3.2 and it can be found, if it exists, in polynomial time. Consequently, the H'-cover problem is polynomially solvable.

Proof. As noted in the beginning of Section 3, the only cover of a tree is the tree itself, and this observation holds true also for rooted trees. Thus given G, one first locates all pending subtrees isomorphic to T_L or T_R. If $T_L \ncong T_R$, L and R belong to different (and hence 1-element) blocks of the degree partition of H'. Thus the subtrees in G determine the partition into U and V which satisfies (1-4) of Theorem 3.2 (if such a partition exists at all). If $T_L \cong T_R$, their isomorphic copies in G pend on vertices that would map onto L or R anyway. We may thus forget about the pending subtrees and follow Theorem 3.3. ∎

3.2 2-satisfiability

The 2-SAT problem (where clauses have at most two variables) is solvable in polynomial time. We can solve a large class of H-covering problems by a polynomial-time reduction to an instance of the 2-SAT problem.

Theorem 3.5 [10] *The H-COVER problem is solvable in polynomial time if every block of the degree partition of H contains at most two vertices.*

Proof. Denote the vertices of the i-th block B_i of H by L_i, R_i (or L_i only, if B_i is a singleton). Suppose that G has the same degree refinement as H and its degree partition is $B'_1, B'_2, \ldots, B'_t$, where the blocks are numbered so that every covering projection sends B'_i onto B_i, $1 \le i \le t$. This structure of G can be checked in polynomial time, and G does not cover H unless it satisfies these assumptions.

The crucial part of the algorithm is to decide which vertices of B'_i should map onto L_i and which onto R_i. This can be done via 2-SAT. For every vertex u of G, introduce a variable x_u. In a truth assignment ϕ, these variables would encode

$$\phi(x_u) = \begin{cases} \mathsf{true} & \text{if } f(u) = L_i \\ \mathsf{false} & \text{if } f(u) = R_i \end{cases} \tag{1}$$

for a corresponding covering projection f (here i is such that $u \in B'_i$). We construct a formula Φ as a conjunction of the following subformulas:

1. (x_u) for every $u \in B'_i$ such that B_i is a singleton;
2. $(x_u \vee x_v) \wedge (\neg x_u \vee \neg x_v)$ for any pair of adjacent vertices u, v which belong to the same block B'_i (i.e., $L_i R_i \in E(H)$);
3. $(x_u \vee \neg x_v) \wedge (\neg x_u \vee x_v)$ if u and v belong to distinct blocks (say $u \in B'_i$ and $v \in B'_j$) and there are exactly the two edges $L_i L_j, R_i R_j$ between B_i and B_j in H;
4. $(x_u \vee x_v) \wedge (\neg x_u \vee \neg x_v)$ if u and v belong to distinct blocks (say $u \in B'_i$ and $v \in B'_j$) and there are exactly the two edges $L_i R_j, R_i L_j$ between B_i and B_j in H;
5. $(x_w \vee x_v) \wedge (\neg x_w \vee \neg x_v)$ if v and w belong to the same block (say B'_j) and are both adjacent to u which belongs to a block (say B'_i) such that $L_i L_j, L_i R_j \in E(H)$.

Note that in case 2, every $u \in B'_i$ has exactly one neighbor v in the same block, in cases 3 and 4, every $u \in B'_i$ has exactly one neighbor $v \in B'_j$, and in case 5, every $u \in B'_i$ has exactly two neighbors $v, w \in B'_j$.

It is clear that Φ is satisfiable if and only if f defined by (1) is a covering projection from G onto H. The clauses derived from 2 guarantee, if $L_i R_i \in E(H)$, that every vertex mapped on L_i has a neighbor which maps onto R_i and vice versa, the clauses from 3-5 control adjacencies to vertices from different blocks, and the technical clauses from 1 control the singletons. ∎

Using a similar technique we can actually solve a larger class of problems, as exemplified by the next theorem.

Theorem 3.6 *Let H be a graph with all but 4 vertices of degree 2. Let these four vertices be a, b, c, d. For every $i \ge 1$, let $n_i(xy)$ be the number of 2-paths of length i between x and y ($x, y \in \{a, b, c, d\}$). Suppose that for every i, one of the following holds*

1) $n_i(aa) = n_i(bb) = n_i(cc) = n_i(dd)$ and $n_i(xy) = 0$ otherwise;

2) $n_i(ab) = n_i(cd) \neq 0$ and $n_i(xy) = 0$ otherwise;
3) $n_i(ac) = n_i(bd) \neq 0$ and $n_i(xy) = 0$ otherwise;
4) $n_i(ad) = n_i(bc) \neq 0$ and $n_i(xy) = 0$ otherwise;
5) $n_i(ab) = n_i(bc) = n_i(cd) = n_i(ad) = 1$ and $n_i(xy) = 0$ otherwise.
Then the H-cover problem is solvable in polynomial time.

Proof. Given a graph G subject to the question if G covers H, we first check if G has the same degree refinement as H. In the affirmative case denote by X the set of vertices of G of degree greater than 2. Note that the vertices a, b, c, d form a block of the degree partition of H, and hence X is a block of the degree partition of G. The graph G consists of these vertices (which will map onto a, b, c or d) and of 2-paths connecting these vertices. Every vertex of X is incident with $\sum_{i \geq 1} 2n_i(aa) + n_i(ab) + n_i(ac) + n_i(ad)$ 2-paths leading to other vertices of X.

For each $x \in X$, introduce two boolean variables, $\phi(x)$ and $\psi(x)$. The encoding of the truth values will be

$$\phi(x) = \text{ true iff } f(x) \in \{a, c\}, \tag{2}$$
$$\psi(x) = \text{ true iff } f(x) \in \{a, b\}, \tag{3}$$

for a tentative covering projection f.

We construct a boolean formula as follows. For every i such that 1) holds, Φ_1 will contain the conjuction of subformulas

$$(\phi(x) \vee \neg\phi(y)) \wedge (\neg\phi(x) \vee \phi(y)) \wedge (\psi(x) \vee \neg\psi(y)) \wedge (\neg\psi(x) \vee \psi(y))$$

for all pairs of vertices $x, y \in X$ connected by a 2-path of length i in G.

For every i such that 2) holds, Φ_2 will contain the conjuction of subformulas

$$(\phi(x) \vee \phi(y)) \wedge (\neg\phi(x) \vee \neg\phi(y)) \wedge (\psi(x) \vee \neg\psi(y)) \wedge (\neg\psi(x) \vee \psi(y))$$

for all pairs of vertices $x, y \in X$ connected by a 2-path of length i in G.

For every i such that 3) holds, Φ_3 will contain the conjuction of subformulas

$$(\phi(x) \vee \neg\phi(y)) \wedge (\neg\phi(x) \vee \phi(y)) \wedge (\psi(x) \vee \psi(y)) \wedge (\neg\psi(x) \vee \neg\psi(y))$$

for all pairs of vertices $x, y \in X$ connected by a 2-path of length i in G.

For every i such that 4) holds, Φ_4 will contain the conjuction of subformulas

$$(\phi(x) \vee \phi(y)) \wedge (\neg\phi(x) \vee \neg\phi(y)) \wedge (\psi(x) \vee \psi(y)) \wedge (\neg\psi(x) \vee \neg\psi(y))$$

for all pairs of vertices $x, y \in X$ connected by a 2-path of length i in G.

For every i such that 5) holds, Φ_5 will contain the conjuction of subformulas

$$(\phi(x) \vee \phi(y)) \wedge (\neg\phi(x) \vee \neg\phi(y))$$

for pairs $x, y \in X$ connected by 2-paths of length i, and

$$(\psi(z) \vee \psi(y)) \wedge (\neg\psi(z) \vee \neg\psi(y))$$

for pairs $y, z \in X$ connected by 2-paths of length i to the same third vertex in G.

We claim that $\Phi = \Phi_1 \wedge \Phi_2 \wedge \Phi_3 \wedge \Phi_4 \wedge \Phi_5$ is satisfiable if and only if G covers H.

Given a covering projection $f : G \to H$, a truth valuation defined by (2) and (3) satisfies Φ. Indeed, Φ_1 is satisfied iff $\phi(x) \equiv \phi(y)$ and $\psi(x) \equiv \psi(y)$, i.e., if $f(x) = f(y)$, for every pair $x, y \in X$ connected by a 2-path of length i in G. One can easily check that $\Phi_2, \ldots, \Phi_5$ are satisfied as well.

Suppose on the other hand that Φ is satisfied by some truth valuation of its variables. Let $f : X \to \{a, b, c, d\}$ be a mapping derived from this valuation via (2) and (3). For the remainder of the proof let $x \in X$ be a vertex such that $f(x) = a$ (the cases $f(x) = b, c$ or d are similar). If i is such that 1) holds, it follows from Φ_1 that $f(y) = a$ for every $y \in X$ connected to x by a 2-path of length i. Hence x is incident to $2n_i(aa)$ 2-paths of length i leading to vertices mapped onto a (cycles through x are counted as two paths).

If 2) holds for i, Φ_2 guarantees that $\phi(x) \not\equiv \phi(y)$ and $\psi(x) \equiv \psi(y)$, i.e., $f(y) = b$, for every $y \in X$ connected to x via a 2-path of length i. Hence x is incident to $n_i(ab)$ 2-paths of length i leading to vertices mapped onto b. Similar arguments work for 3) and 4).

If 5) holds for i, Φ_5 guarantees that $\phi(x) \not\equiv \phi(y)$ and $\phi(x) \not\equiv \phi(z)$ for y, z connected to x via 2-paths of length i. Thus $f(y), f(z) \in \{b, d\}$. But Φ_5 also guarantees that $\psi(z) \not\equiv \psi(y)$, i.e., $f(y) \neq f(z)$. Hence x is incident to one 2-path of length i leading to a vertex mapped onto b and to one 2-path of length i leading to a vertex mapped onto d. This shows that the vertices of X are mapped properly, and an argument analogous to the argument in the proof of Theorem 3.2 shows that the vertices of degree 2 can be mapped onto the vertices of degree 2 in H so that the entire mapping is a covering projection.

It remains to note that all clauses in Φ have size 2, hence the satisfiability of Φ is an instance of 2-SAT and thus solvable in polynomial time. ∎

4 $\mathcal{NP}$-completeness

In this section, we show the inherent difficulty of H-cover problems for certain classes of graphs H. Since it is easy to verify a purported solution to the problem, the H-cover problem is in $\mathcal{NP}$, for any graph H. We first mention some earlier results.

Theorem 4.1 [9] *For every $k \geq 4$, the K_k-cover problem is $\mathcal{NP}$-complete.*

Theorem 4.2 [11] *For every fixed $k > 2$ and every k-regular graph H, the H-cover problem is NP-complete if*

H is $(1 + \lfloor \frac{k}{2} \rfloor)$-edge connected, or

H is k-edge colorable.

4.1 Reductions from coloring problems

In this section, we construct reductions to H-cover (for different classes of H) from several known $\mathcal{NP}$-complete problems: edge coloring, vertex coloring and H-coloring. The *H-coloring* problem asks for the existence of a labeling of vertices of G by vertices of H, $h : V(G) \to V(H)$, which preserves adjacencies, $uv \in E(G) \Rightarrow h(u)h(v) \in E(H)$. This problem is easy if H is bipartite but $\mathcal{NP}$-complete otherwise, as shown by Hell and Nešetřil [8]. In this setting, the *vertex k-coloring* problem is equivalent to the K_k-coloring problem. The *edge k-coloring* problem asks if each edge of a graph can be assigned one of k colors so that no two edges incident with the same vertex are assigned the same color. For $k > 2$, edge k-coloring of k-regular graphs is $\mathcal{NP}$-complete [14]. The following observation on automorphisms of H with the composition of f and π being $\pi \circ f(v) = \pi(f(v))$ is used in our reductions.

Fact 4.3 *If G covers H by $f : V(G) \to V(H)$ and $\pi \in Aut(H)$ then $\pi \circ f$ is also a covering projection from G onto H.*

For fixed H, the reductions are by vertex and edge gadget construction, yelding a graph G' which covers H if and only if a given graph G can be colored appropriately. The general outline of the construction is as follows:

1. Define vertex gadget for a vertex $v \in V(G)$ by a subgraph of a graph that covers H, with $deg_G(v)$ "ports" (interfaces with edge gadgets). Reductions from edge coloring require that a covering projection onto H distinguishes between different ports and allows for the property of automorphisms of H as per Fact 4.3. Reductions from k-vertex coloring require that a covering projection onto H acts equivalently on each port but provides k distinct mappings on the vertex gadget.

2. Define edge gadget connecting two ports of adjacent vertex gadgets in G' by a subgraph of a graph that covers H so that the "only if" property of the reduction is fulfilled. This amounts to ensuring that a covering projection onto H acts on the two ports equivalently in the case of a reduction from edge coloring, and distincly in the case of vertex coloring.

3. Neighborhoods left unspecified are completed, possibly with added vertices, so that the "if" direction of the reduction is met (this amounts to extending any partial covering projection defined in step 2 to a cover of H).

Theorem 4.4 *Let Q be a block in the degree-partition of a graph H and let there exist an $S \subseteq V(H)$ such that for every $v \in Q$, $N(v) \setminus Q = S$. Then the H-cover problem is $\mathcal{NP}$-complete in each of the following cases.*
(a) $H[Q]$ is a discrete graph, $|Q| \geq 3$ and $|S| \geq 3$;
(b) $H[Q]$ is a perfect matching, $|Q| \geq 4$ and $|S| \geq 2$;
(c) $H[Q]$ is a cycle, $|Q| \geq 3$ and $|S| \geq 1$;
(d) $H[Q]$ is a complete graph, $|Q| \geq 4$.

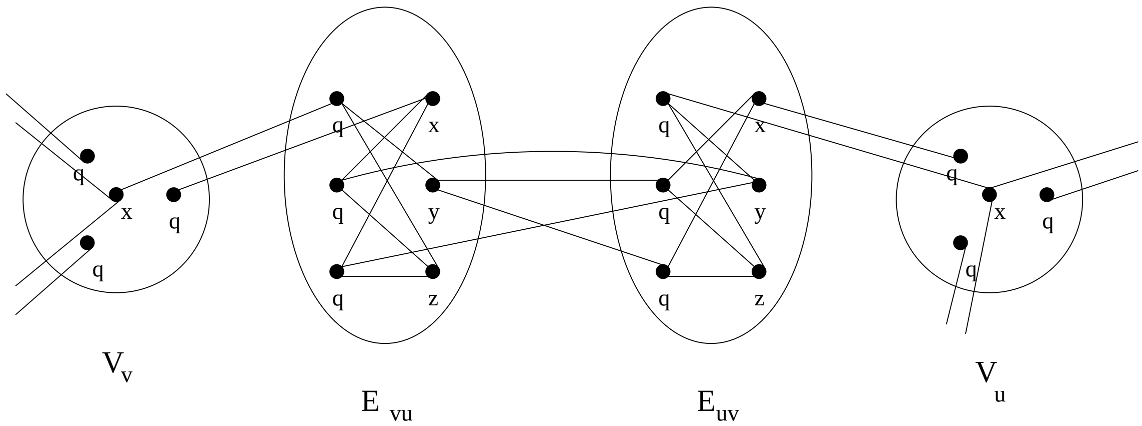


Figure 2: Gadgets for case $H[Q]$ discrete, $|Q| = 3$, with vertices of Q marked q

The proofs of cases (a),(b) and (c) appear immediately below, while the proof of case (d) appears as a corollary to Theorem 4.8 in the next section.

Proof. (a) $H[Q]$ is a discrete graph on $|Q| = k \geq 3$ vertices, each of them adjacent to the same set S of at least three vertices. For a given k-regular graph G, we construct a graph G' such that G' covers H if and only if G is k-edge colorable. Let $\{x, y, z\} \subseteq S$ and $Q = \{q_1, ..., q_k\}$. The vertex gadget V_v for a vertex $v \in V(G)$ will consist of an almost complete copy of H but lacking the edges connecting x and Q, $xq_1, ..., xq_k$. The edge gadget for an edge $uv \in E(G)$ will consist of two almost complete copies of H, call them E_{uv}, E_{vu}, each lacking the edges xq_1 and $yq_2, ..., yq_k$. The edges $yq_2, ..., yq_k$ will instead connect together E_{uv} and E_{vu} by a total of $2(k-1)$ edges, ensuring that in a successful cover of H we have both copies of q_1, in E_{uv} and E_{vu}, labeled by the same vertex (the copy of y in E_{uv} is adjacent to the copy of q_1 in E_{uv} and all its other adjacencies in Q are in E_{vu}. The vertex z of E_{vu} has all its Q-neighbors in E_{vu}, and so the Q vertices of E_{vu} map bijectively onto Q of H. Thus the only non-adjacency of $y \in E_{uv}$ among Q in E_{vu}, namely the copy of q_1, must have the same label as q_1 in E_{uv}.) The edge gadget is connected to the vertex gadgets by an edge from x of V_v to q_1 of E_{vu} and an edge from x of E_{vu} to one of the Q-vertices of V_v, say q_{a_u} (the other Q-vertices of V_v are connected to the remaining edge gadgets adjacent to V_v). Similar edges xq_1 and xq_{a_v} are added for E_{uv}, V_u, completing the construction of G'. See Figure 2 for an example where $|Q| = 3$. Note that in a covering projection from G' onto H the two copies of q_1 in the gadget of edge uv and q_{a_u}, q_{a_v} in vertex gadgets of v and u, respectively, all receive the same label, corresponding to the unique color of edge (u, v).

Assuming G is k-edge colorable with the edge uv being assigned the color $c \in \{1, ..., k\}$ we label both copies of q_1 of the edge gadget of (u, v) by q_c. Since any permutation of $V(H)$ which moves only Q is an automorphism of H, this labeling is easily extended to make a cover from G' onto H.

In the other direction, suppose G' covers H. Since Q is a block in the degree partition of H and the vertex gadget contains $y \in S$ adjacent to all Q-vertices of the gadget, all Q-vertices of V_v map onto distinct vertices of H. Thus if we color edge uv of G by the label of both copies of q_1 in the edge gadget of uv the result is a coloring of $E(G)$ such that edges incident with the same vertex receive distinct colors. ∎

Proof. (b) $H[Q]$ is a perfect matching ($|Q| \geq 4$) with each vertex of Q adjacent to the same set S of at least 2 vertices. For $|Q| \geq 6$, a reduction from edge $\frac{|Q|}{2}$-colorability similar to the one from case (a) can be easily performed. Therefore we concentrate on $|Q| = 4$. Let $Q = \{a_1, b_1, a_2, b_2\}$ with $\binom{Q}{2} \cap E(G) = \{a_1b_1, a_2b_2\}$. For a given graph G, we construct a graph G' such that G can be vertex 4-colored if and only if G' covers H. Let $v \in V(G)$ with $deg_G(v) = d$. The gadget of vertex v contains $2(d+1)$ copies of $H[Q]$ and also of an arbitrarily chosen vertex $A \in S$, call these copies $M_0, ..., M_{2d+1}$ and $A_0, A_1, ..., A_{2d+1}$, respectively. Let the four vertices of M_i be NW_i, NE_i, SW_i, SE_i, with edges NW_iNE_i and SW_iSE_i. The copies of $H[Q]$ and A will form a cycle of 6-cycles by having the vertex A_i, for every i, connected to $NE_i, SE_i, NW_{i+1}, SW_{i+1}$ (addition modulo $2d+2$). The vertex gadget contains also $d+2$ copies of $H \setminus Q$, $L_0, L_1, L_3, L_5, ..., L_{2d+1}$, where L_i, contains the previously described vertex A_i for odd i and $i = 0$. Let S_i be the copy of S in L_i. L_0 and L_1 play distinct roles and we describe their remaining connections first. Vertices $x \in S_0 \setminus A_0$ are connected to NW_0, SW_0, NW_1, SW_1, while vertices $x \in S_1 \setminus A_1$ are connected to NE_0, SE_0, NE_1, SE_1. Since Q is a block in the vertex partition of H, in a cover of H the edges of M_i, for every i, are sent to edges of $H[Q]$. Moreover, L_0 and L_1 enforce that to satisfy neighborhoods of A_0 and any $x \in S_0 \setminus A_0$ all edges in M_{2i}, for every i, must be sent to a single edge of $H[Q]$, while edges of M_{2i+1}, for every i, are sent to the other edge of $H[Q]$. Vertices $x \in S_i \setminus A_i$ for $i = 3, 5, ..., 2d-1$ are connected to $SE_{i-1}, SW_i, SE_i, SW_{i+1}$, while any $x \in S_{2d+1} \setminus A_{2d+1}$ is connected to $SE_{2d}, SW_{2d+1}, SE_{2d+1}, SW_2$. This completes the description of the vertex gadget. Figure 3 gives the construction for a special case of H a 6-vertex graph, where L_i is the graph induced by B_i and A_i. In a covering projection onto H, NW_{2i} and NW_{2i+2}, for every i, must cover the same vertex, since otherwise a vertex $x \in S_{2i+1} \setminus A_{2i+1}$ will have two neighbors with the same label. Thus we have a unique label for NW_{2i}, for every i, which will correspond to the color of $v \in V(G)$ for the instance G of the vertex 4-coloring problem. Note however, that the label of an NW_{2i+1} vertex is not fixed.

In the vertex gadget the only vertices lacking connections are A_{2i},NW_{2i}, NE_{2i}, NW_{2i+1},NE_{2i+1} for $i = 1, 2, ..., d$. The edge gadget for an edge uv consists of two copies of $H \setminus Q$, L_W and L_E, and will provide the remaining connections for some M_{2u_i}, M_{2u_i+1} from u's gadget and M_{2v_i}, M_{2v_i+1} from v's gadget. A_{2u_i} from u's gadget and A_{2v_i} from v's gadget are already contained in L_W and L_E, one in each. Additionaly, any $x \in S_W \setminus A_W$ is connected to the four NW vertices of these four copies of $H[Q]$ and any $x \in S_E \setminus A_E$ is connected to the four NE vertices. This ensures that labels of the NW vertex of M_{2u_i} and M_{2v_i} will differ while still allowing for any other combination from $\{a_1, b_1, a_2, b_2\}$.

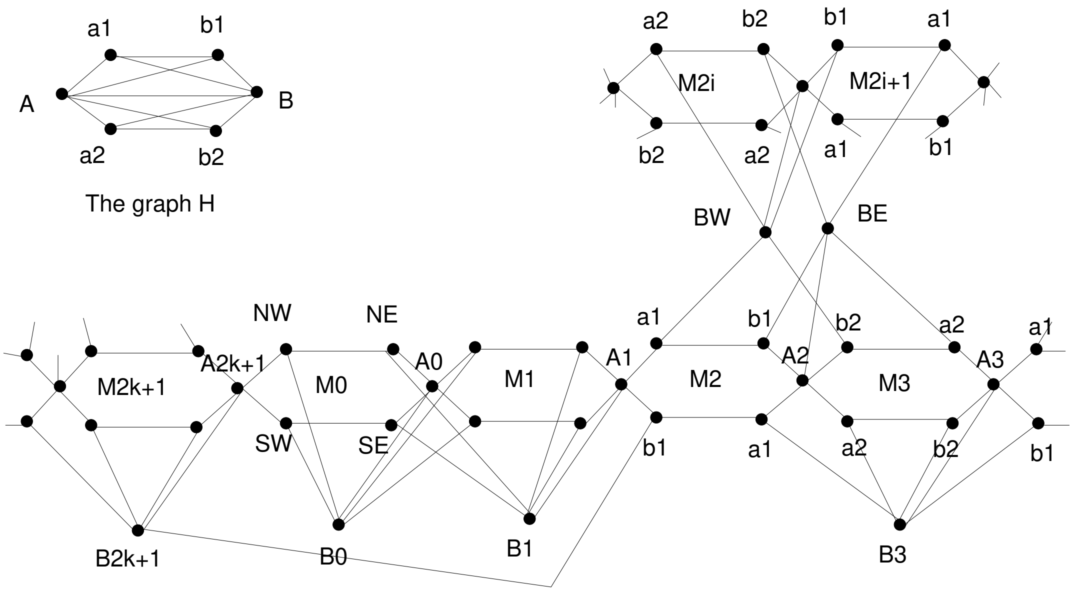


Figure 3: Case $H[Q]$ a matching. Vertex gadget for a vertex v of degree k and color a_1 at bottom, edge gadget vertices BW, BE, connected to a vertex colored a_2 at top. For $i = 1..k$, M_{2i} and M_{2i+1} are connected through an edge gadget to another vertex gadget. Expanding copies of Bx will give the general case $H[Q]$ a matching on 2 edges.

The construction of G' is completed, see Figure 3. If G' covers H then we color G using the four colors $\{a_1, b_1, a_2, b_2\}$, with vertex v receiving the same color as the label of the NW_{2i} vertices in its gadget. By the observations made above, this constitutes a vertex 4-coloring of G. Conversely, if G is vertex colorable by colors $\{a_1, b_1, a_2, b_2\}$, we label the NW_{2i} vertices of vertex gadgets accordingly. This labeling can be extended to a covering projection from G' onto H, see Figure 3 for an example where the adjacent vertices have colors a_1 and a_2. ∎

Proof. (c) $H[Q]$ is a k-cycle ($k \geq 3$) with each vertex of Q adjacent to the same set S of at least one vertex. For a given graph G, we construct a graph G' such that G can be vertex k-colored if and only if G' covers H. The gadget for a vertex v is a cycle C_v of length $deg_G(v) \times k$. In the case of a positive answer C_v will cover $H[Q]$. C_v is broken naturally into $deg_G(v)$ consecutive paths of length k, one for each edge incident with v, so that the first endpoint of each path receives the same label in any cover from C_v onto $H[Q]$. We call these endpoints the designated vertices of the gadget, with their label providing the corresponding color of vertex v. The gadget for an edge uv hooks up with one of these paths, say $c^u_1, ..., c^u_k$ from C_v and also with $c^v_1, ..., c^v_k$ from C_u. The edge gadget itself consists of k-cycles $C_1, ..., C_{k-2}$, cycle C_i having consecutive vertices $c^i_1, ..., c^i_k$, together with $(H \setminus Q)$-copies $R_1, ..., R_k$. Vertices $c^v_i, c^u_i, c^1_i, c^2_i, ..., c^{k-2}_i$ are given all their remaining adjacencies

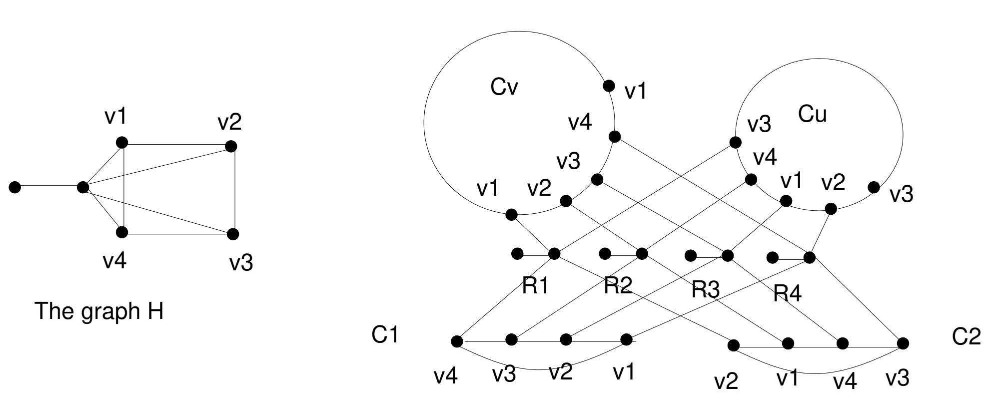


Figure 4: Case $H[Q]$ a 4-cycle. Vertex gadgets C_v and C_u and an edge gadget consisting of $R_1, R_2, R_3, R_4, C1, C2$, labeled so that vertices v and u are colored $v1$ and $v3$, respectively.

(S-neighbors) from R_i, for $i = 1, ..., k$. This also satisfies R_i locally and completes the description of G', see Figure 4 for an example.

Assume $f : V(G)' \to V(H)$ is a covering projection and let $uv \in V(G)$ as above. Since Q is a block in the degree partition of H, the vertices of vertex gadgets must be sent to Q. The designated vertices c_1^u and c_1^v share a neighbor in R_1 ($|S| \geq 1$) and hence $f(c_1^u) \neq f(c_1^v)$. Hence we use the labels of the designated vertices of vertex gadgets as a vertex k-coloring of the graph G.

Assume G can be colored with colors $1, 2, ..., k$, with $Q = \{v_1, ..., v_k\}$. A vertex v colored c has its designated vertices labeled v_c in a covering projection from its vertex gadget onto $H[Q]$. In the gadget for an edge uv the $(H \setminus Q)$-copies naturally cover $H \setminus Q$. The k-cycles in the edge gadget are labeled uniquely to cover $H[Q]$ while satisfying neighborhoods of $(H \setminus Q)$-copies. This will result in a labeling where $c_i^v, c_i^u, c_i^1, c_i^2, ..., c_i^{k-2}$, with naming conventions as above, are given k distinct labels from Q, with G' covering H. See Figure 4 for an example. ∎

The *k-starfish* graph has k vertices of degree two and k vertices of degree four with the vertices of degree four inducing a cycle and any two consecutive vertices of this cycle sharing a neighbor of degree two, see Figure 5.

Theorem 4.5 *For every $i \geq 1$ the $(2i+1)$-starfish-cover problem is $\mathcal{NP}$-complete.*

Proof. Let $k = 2i + 1$. Given a graph G, we construct a graph G' such that G is C_k-colorable if and only if G' covers the k-starfish. The vertex gadget C_v for $v \in V(G)$ consists of a cycle of length $k \times deg_G(v)$, broken naturally into $deg_G(v)$ consecutive paths of length k. The first endpoints of the paths, the designated vertices of this gadget, receive the same label in any cover of a C_k-cycle. The edge gadget for $uv \in E(G)$ hooks up with two such paths, say $F_0 = c_1^0, c_2^0, ..., c_k^0$ from C_u and $F_1 = c_1^1, c_2^1, ..., c_k^1$ from C_v. The edge gadget contains the degree-2 vertices v_j^i for $i = 0, 2, ..., k-1$ and $j = 1, 2, ..., k$ and also contains $k-2$ k-cycles, call them $F_2, F_3, ..., F_{k-1}$, with F_i having consecutive vertices $c_1^i, c_2^i, ..., c_k^i$.

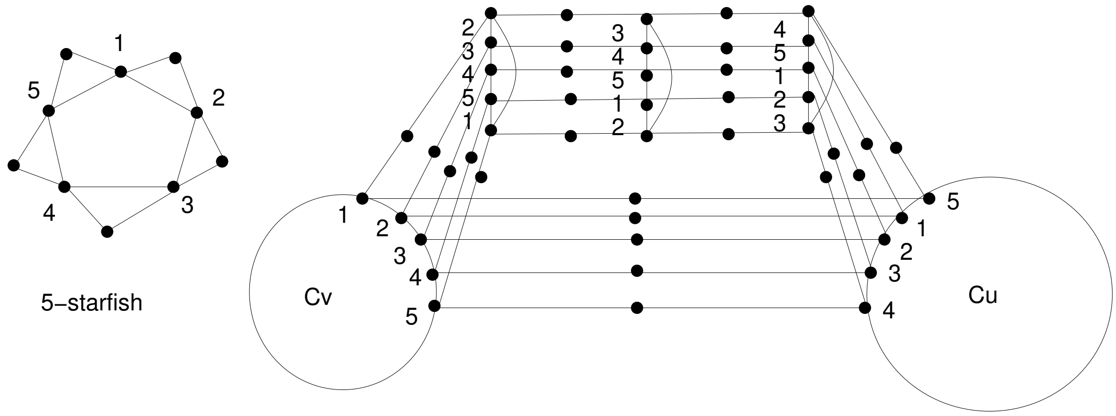


Figure 5: Case 5-starfish, with vertex gadgets C_u and C_v connected by an edge gadget.

$F_0, F_1, ..., F_{k-1}$ are hooked up by the degree-2 vertices to form a cycle, with c_j^i adjacent to v_j^i and to $v_j^{i+1 \bmod k}$. This completes the description of G', see Figure 5.

Let f be a cover of k-starfish by G', and let $uv \in E(G)$, with naming conventions as above. Since the designated vertices c_1^0 from C_u and c_1^1 from C_v have a common degree-2 neighbor, we must have $f(c_1^0)f(c_1^1)$ an edge in the k-starfish. Thus, we construct a C_k-coloring of G by focusing on the k-cycle induced by degree-4 vertices in the k-starfish, and sending $u \in V(G)$ to the f-label of the designated vertex in its vertex gadget.

For the other direction of the proof we reverse this process, labeling designated vertices by the C_k-coloring induced on the degree-4 vertices of the k-starfish, see Figure 5 for an example. ■

4.2 Reductions from covering problems

Theorem 4.6 *The $2k$-starfish-cover problem is $\mathcal{NP}$-complete whenever the k-starfish-cover problem is $\mathcal{NP}$-complete.*

Proof. Let G be a connected graph for which we test if G covers k-starfish. In particular, G contains vertices of degrees 2 and 4 only, every vertex of degree 2 is adjacent to 2 vertices of degree 4 and every vertex of degree 4 has 2 neighbors of degree 2 and 2 neighbors of degree 4. We will construct a graph H such that H covers $2k$-starfish if and only if G covers k-starfish.

Let X be the set of vertices of degree 4 in G. We first construct G' as follows:

- G' contains the vertices X,

- for every edge $xy \in E(G)$, $x, y \in X$, we add a vertex $A(xy)$ adjacent to x and y, and

- for every pair of vertices $x, y \in X$ connected in G by a common neighbor of degree 2, we add a vertex $B(xy)$ and connect it by 2-paths to x and y via two newly added vertices of degree 2.

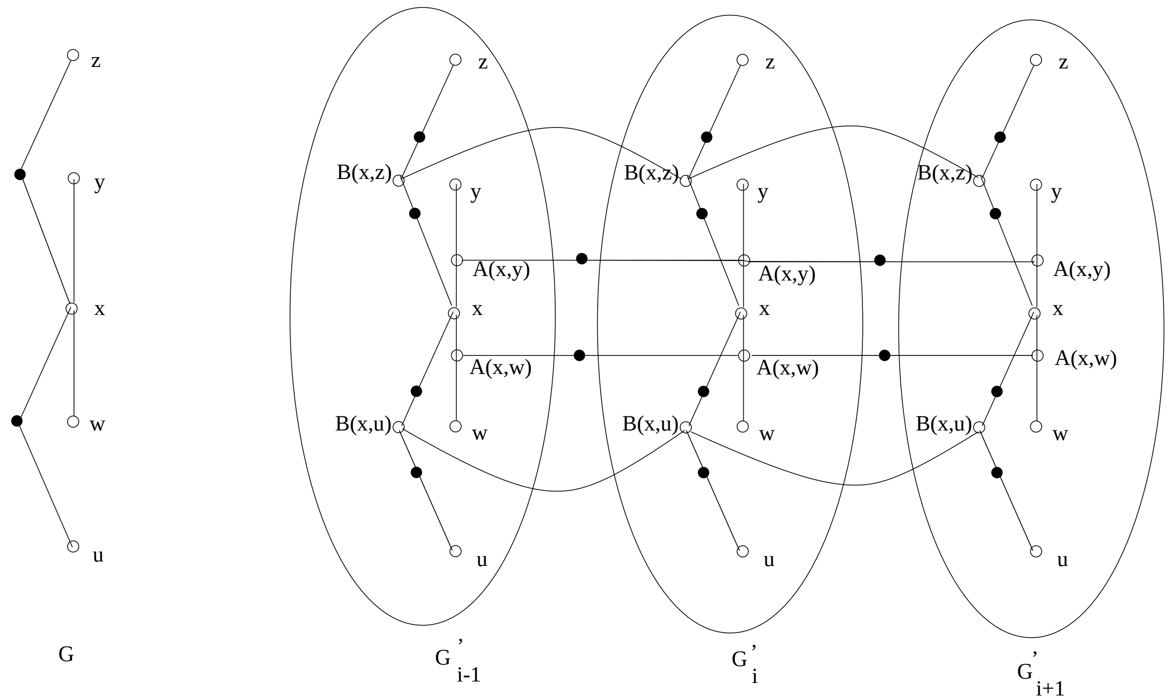


Figure 6: The construction in Theorem 4.6 with vertices of degree two (four) in black (white). Part of the input graph G on the left and the connections between subgraphs G'_{i-1}, G'_i, G'_{i+1} of the constructed graph H on the right.

Then we take $2k$ copies of G', called $G'_i, i = 1, 2, \ldots, 2k$. Vertices in G'_i will be denoted by $x_i, A_i(xy)$ and $B_i(xy)$ in the obvious way. The constructed graph H consists of these $2k$ copies of G' plus additional edges and vertices of degree 2: The vertices $A_i(xy)$ and $A_{i+1}(xy)$ will be connected by a path of length 2 via a newly added vertex of degree 2 (here and later on, the addition in subscripts is modulo $2k$, i.e., $A_{2k+1}(xy) = A_1(xy)$). Similarly, an edge between $B_i(xy)$ and $B_{i+1}(xy)$ will be added to H (thus for every such pair x, y, H contains a $2k$-cycle $B_1(xy)B_2(xy) \ldots B_{2k}(xy)$). See Figure 6. The construction of H is clearly polynomial.

Let f be a covering projection from H to the $2k$-starfish (again, we assume that the $2k$-cycle connecting the degree four vertices in the $2k$-starfish is $1, 2, \ldots, 2k$, and similarly, the k-cycle connecting the degree four vertices in the k-starfish is $1, 2, \ldots, k$). We may assume without loss of generality that for some $x \in X$, $f(x_1)$ is even. It follows from the connectedness of G that $f(x_1)$ is even for every $x \in X$ (while $f(A_1(xy))$ and $f(B_1(xy))$ are odd). Hence, the mapping g defined by $g(x) = \frac{f(x_1)}{2}$ induces a covering projection of G onto the k-starfish.

On the other hand, let g be a covering projection of G onto the k-starfish. We define f on the degree 4 vertices of H as follows

$$f(x_i) = 2g(x) + i - 1, x \in X, i = 1, 2, \ldots, 2k$$

$$f(A_i(xy)) = g(x) + g(y) + i - 1, x, y \in X, xy \in E(G), i = 1, 2, \ldots, 2k,$$

$$f(B_i(xy)) = g(x) + g(y) + i - 1, x, y \in X, i = 1, 2, \ldots, 2k.$$

This f induces a covering projection of H onto the $2k$-starfish. (Every $x \in X$ has neighbors y and z in X such that $g(y) = g(x) - 1$ and $g(z) = g(x) + 1$, and is connected via vertices of degree 2 to vertices u and v such that $g(u) = g(x) - 1$ and $g(v) = g(x) + 1$. Thus for every i, x_i in H is adjacent to $A_i(xy)$ with $f(A_i(xy)) = f(x_i) - 1$ and to $A_i(xz)$ with $f(A_i(xz)) = f(x_i) + 1$, and it is connected via vertices of degree 2 to vertices $B_i(xu)$ with $f(B_i(xu)) = f(x_i) - 1$ and to $B_i(xv)$ with $f(B_i(xv)) = f(x_i) + 1$. Vertex $A_i(xy)$ has neighbors y_i and x_i with $f(y_i) = f(A_i(xy)) - 1$ and $f(x_i) = f(A_i(xy)) + 1$ and is connected via vertices of degree 2 to vertices $A_{i-1}(xy)$ and $A_{i+1}(xy)$ with $f(A_{i-1}(xy)) = f(A_i(xy)) - 1$ and $f(A_{i+1}(xy)) = f(A_i(xy)) + 1$. Similarly, $B_i(xu)$ is connected via vertices of degree 2 to u_i and x_i with $f(u_i) = f(B_i(xu)) - 1$ and $f(x_i) = f(B_i(xu)) + 1$ and is adjacent to vertices $B_{i-1}(xu)$ and $B_{i+1}(xu)$ with $f(B_{i-1}(xu)) = f(B_i(xu)) - 1$ and $f(B_{i+1}(xu)) = f(B_i(xu)) + 1$.) ∎

Note that the 4-starfish-cover problem is solvable in polynomial time by Theorem 3.6. Our conjecture is that the k-starfish-cover problem is $\mathcal{NP}$-complete for all other $k \geq 3$. To confirm the conjecture it would suffice, in light of Theorems 4.5 and 4.6, to solve the following open problem.

Problem 4.7 *Show that the 8-starfish-cover problem is $\mathcal{NP}$-complete.*

A graph H may have an induced subgraph H' for which the H'-cover problem is $\mathcal{NP}$-complete. In general, the H-cover problem could itself be easy. Our next theorem is a non-trivial extension of a result appearing in [10], and shows $\mathcal{NP}$-completeness for the H-cover problem whenever the graph H' satisfies some general conditions

Theorem 4.8 *The H-cover problem is $\mathcal{NP}$-complete if for some block $Q = \{v_1, ..., v_k\}$ in the degree partition of H the $H[Q]$-cover problem is $\mathcal{NP}$-complete and there exists a graph F with vertices $V(F) = F_Q \cup F_R$ and a k by kh matrix ($h = |V(F)|/|V(H)|$) over Q whose columns are elements of $Aut(H[Q])$ and whose rows can be extended to h-fold covering projections of F onto $H \setminus E(H[Q])$ that send F_Q to Q.*

Proof. We reduce from the $H[Q]$-cover problem. Given a graph G, we construct a graph G' such that G covers $H[Q]$ if and only if G' covers H. Let $V(G) = \{x_1, ..., x_n\}$, $F_Q = \{y_1, ..., y_{kh}\}$ and $V(H) = Q \cup R$. G' will contain kh copies of G ($G_1, ..., G_{kh}$) and n copies of $F[F_R]$ ($R_1, ..., R_n$). A vertex $x_i \in V(G)$ thus has kh copies $x_i^1, x_i^2, ..., x_i^{kh}$ in G' ($x_i^j \in V(G_j)$), which will be used as the remaining neighbors for vertices of R_i. We let the vertex x_i^j play

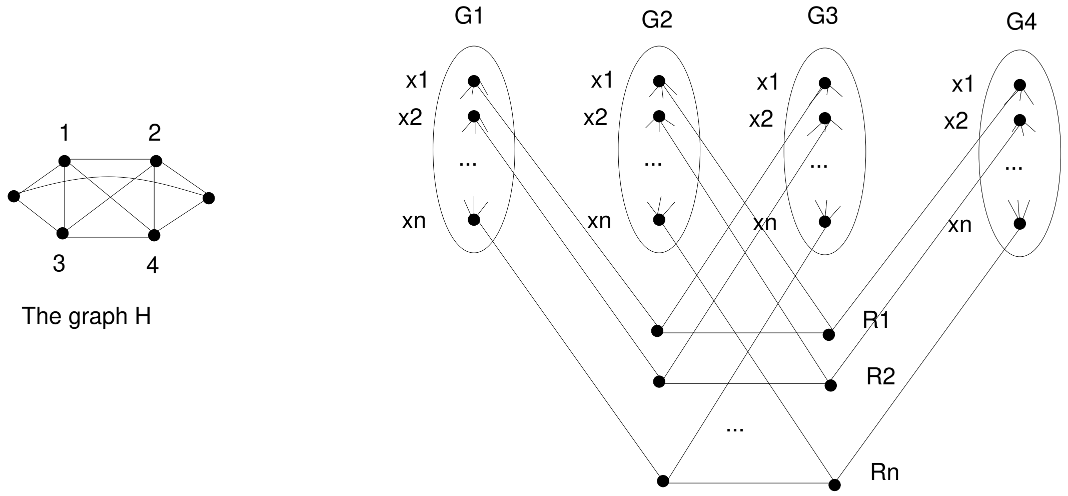


Figure 7: Case $H[Q] = K_4$, $F = H \setminus E(H[Q])$ and the graph G' constructed in Theorem 4.8

the role of vertex $y_j \in F_Q$ and connect vertices of R_i to its remaining neighbors, as specified by F, using $\{x_i^1, ..., x_i^{kh}\}$, thereby completing the construction of the graph G'. Note the construction of G' is such that G' and H have the same degree refinement. Note that since F covers $H \setminus E(H[Q]))$ with F_Q mapping to Q we have $E(F[F_Q]) = \emptyset$. See Figure 7 for a simple example.

Suppose G' covers H by a covering projection p. Let $B_1 = Q, B_2, ..., B_m$ be the blocks of H. We know there exists a covering projection f from F to $H \setminus E(H[Q])$. Let $F_1, ..., F_n$ be the n copies of F in G' (these are the components of G' after removal of edges of G). Applying f to each copy of F let $B'_1, ..., B'_m$ be the vertices sent to $B_1, ..., B_m$, respectively. Note that $V(G') = \bigcup_{1 \le i \le m} B'_i$. Since f sends F_Q to Q we have $B'_1 = \bigcup_{1 \le i \le kh} V(G_i)$. The blocks in the degree refinement of G' must be $B'_1, ..., B'_m$, since it could not be a refinement of this and it should have degree refinement identical to H. Therefore p must map each copy of G to $H[Q]$.

For the other direction, suppose $f : V(G) \to Q$ is a covering projection from G onto $H[Q]$. Let $\Delta_1, \Delta_2, ..., \Delta_{kh}$ be the columns of the matrix mentioned in the statement of the theorem and let $\pi_1, \pi_2, ..., \pi_k$ be its rows. Since $\Delta_i \in Aut(H[Q])$ for every i, we have by Fact 4.3 that $\Delta_1 \circ f, ..., \Delta_{kh} \circ f$ are also covering projections of G onto $H[Q]$ and we label the vertices of the copy G_j of G by $\Delta_j \circ f$. By construction we have that R_i is connected to vertices $x_i^1, x_i^2, ..., x_i^{kh}$. Assuming that $f(x_i) = v_j \in Q$ we label vertices of R_i by the respective labels $\Delta_1(v_j), \Delta_2(v_j), ..., \Delta_{kh}(v_j)$, corresponding to some π_r. Since π_r can be extended to a covering projection from F to $H \setminus E(H[Q])$, we can send $V(R_i)$ to R by this covering projection, locally getting a covering projection from R_i to $H[R]$ and with correct labels for remaining neighbors of R_i as well. The same is done for all n copies $R_1, ..., R_n$ of $F[F_R]$ resulting in a mapping of $V(G')$ to $V(H)$ where each copy of G covers $H[Q]$ and each copy of $F[F_R]$ covers $H[R]$ and the remaining neighbors of copies of both G and $F[F_R]$ have correct labels, hence we have a covering projection from G' to H. ∎

We give an example of the simplest kind of application of this result, namely when $h = 1$ so the fixed graph F is isomorphic to $H \setminus E(H[Q])$, see Figure 7. Consider the graph H consisting of a block $Q = \{v_1, v_2, v_3, v_4\}$ inducing a K_4 and another block on two vertices $\{v_5, v_6\}$, one of these adjacent to $\{v_1, v_2\}$ and the other adjacent to $\{v_3, v_4\}$. Note that the K_4-cover problem is $\mathcal{NP}$-complete by Theorem 4.1. Moreover, for the following 4 by 4 matrix

$$\begin{pmatrix} v_1 & v_2 & v_3 & v_4 \\ v_2 & v_1 & v_4 & v_3 \\ v_3 & v_4 & v_1 & v_2 \\ v_4 & v_3 & v_2 & v_1 \end{pmatrix}$$

both its rows and columns are projections onto Q of elements of $Aut(H)$, so by Theorem 4.8 the H-cover problem is $\mathcal{NP}$-complete.

Proof. (of case (d) Theorem 4.4) $H[Q]$ is a complete graph, with $|Q| \geq 4$. By Theorem 4.1 the $H[Q]$-cover problem is $\mathcal{NP}$-complete and $Aut(H)|_Q$ is the symmetric group on $|Q|$ points, so the conditions in Theorem 4.8 are easily satisfied. ∎

5 Conclusion

We have presented a methodology to analyze the complexity of graph covering problems. The presented results suffice to classify all simple graphs whose components have at most six vertices. However, we are still not able to give the general classification. Allthough all $\mathcal{NP}$-completeness reductions in this paper relied on nontrivial automorphisms of the fixed graph H, the result of Theorem 4.2 [11] encompasses many rigid graphs, so the classification cannot depend solely on the automorphism group of the fixed graph. Likewise, the classification cannot depend solely on the degree refinement of the fixed graph, as exemplified by the k-starfish graphs that all share the same degree refinement, but for which the covering problem is $\mathcal{NP}$-complete whenever k is not a power of 2 and solvable in polynomial time when $k = 4$. In Figure 8 see two graphs with the same degree refinement and size of degree partitions, the right graph polynomial by Theorem 3.6 while the left graph contains a subdivision of K_4 and is $\mathcal{NP}$-complete by a reduction similar to Theorem 4.8.

Resolving the complexity of graph covering problems is tightly connected to pure combinatorics. For example, Theorem 4.8 opens for the possibility of showing $\mathcal{NP}$-completeness results by simply applying existing results from graph covering theory, or developing new such results. Those results should show that for certain $H[Q]$ and supergraphs H, the covering graphs F required in Theorem 4.8 always exist. Indeed, the proof technique in [11] is of this kind.

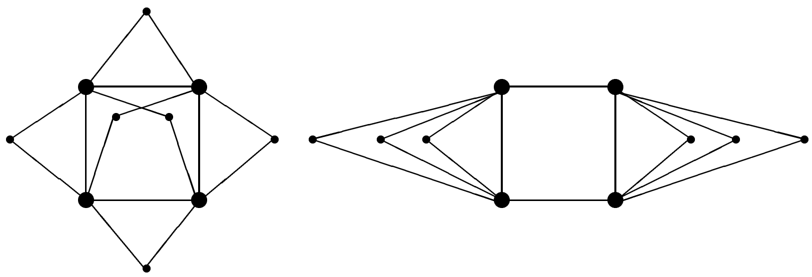


Figure 8: Two graphs with the same degree refinement but different complexity classification for the covering problem.

Acknowledgements

This research has been initiated under the auspices of the scientific exchange program between the National Research Council and the Czechoslovak Academy of Sciences and completed with partial support of the National Science Foundation through grant NSF-CCR-9213439.

Research of the first author was supported in part by Czech-US Science and Technology Program grant No. 94 051 and by Czech Research grants GAUK 194/1996 and GA ČR 201/96/0194.

References

[1] J. Abello, M.R. Fellows and J.C. Stillwell, On the complexity and combinatorics of covering finite complexes, *Australasian Journal of Combinatorics 4* (1991), 103-112;

[2] D. Angluin, Local and global properties in networks of processors, in *Proceedings of the 12th STOC* (1980), 82-93;

[3] N. Biggs, *Algebraic Graph Theory*, Cambridge University Press, 1974;

[4] B. Courcelle and Y. Métivier, Coverings and minors: Applications to local computations in graphs, *European Journal of Combinatorics* 15 (1994), 127-138.

[5] M.R. Garey and D.S. Johnson, *Computers and Intractability*, W.H.Freeman and Co., 1978;

[6] J.L. Gross and T.W. Tucker, *Topological Graph Theory*, J.Wiley and Sons, 1987.

[7] F.Harary, *Graph Theory*, Addison-Wesley, 1969;

[8] P. Hell and J. Nešetřil, On the complexity of H-colouring, *Journal of Combinatorial Theory B 48* (1990), 92-110;

[9] J.Kratochvíl, *Perfect codes in general graphs*, monograph, Academia Praha (1991);

[10] J.Kratochvíl, A.Proskurowski and J.A.Telle, Complexity of Graph Covering Problems, *Proceedings of 20th International Workshop on Graph-Theoretical Concepts in Computer Science, WG'94, Lecture Notes in Computer Science vol.903*, Springer-Verlag, 1995, pp 93-105.

[11] J.Kratochvíl, A.Proskurowski and J.A.Telle, Covering regular graphs, to appear *Journal of Combinatorial Theory - Series B.*

[12] J.Kratochvíl, A.Proskurowski and J.A.Telle, Complexity of Colored Graph Covers I, to appear *Proceedings of 22nd International Workshop on Graph-Theoretical Concepts in Computer Science, WG'97 Berlin June 18-20, Lecture Notes in Computer Science.*

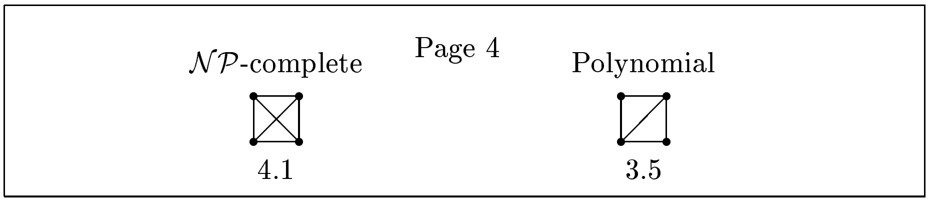
[13] F.T. Leighton, Finite common coverings of graphs, *Journal of Combinatorial Theory B 33* (1982), 231-238;

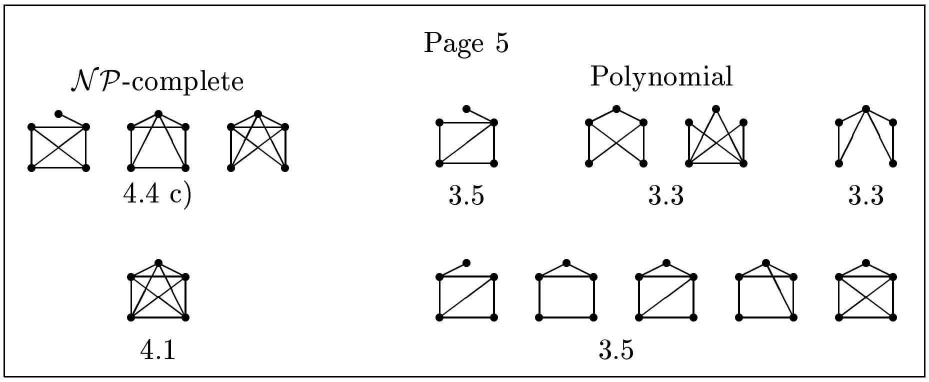
[14] D. Leven, Z. Galil, NP-completeness of finding the chromatic index of regular graphs, J. Algoritms 4 (1983) 35-44

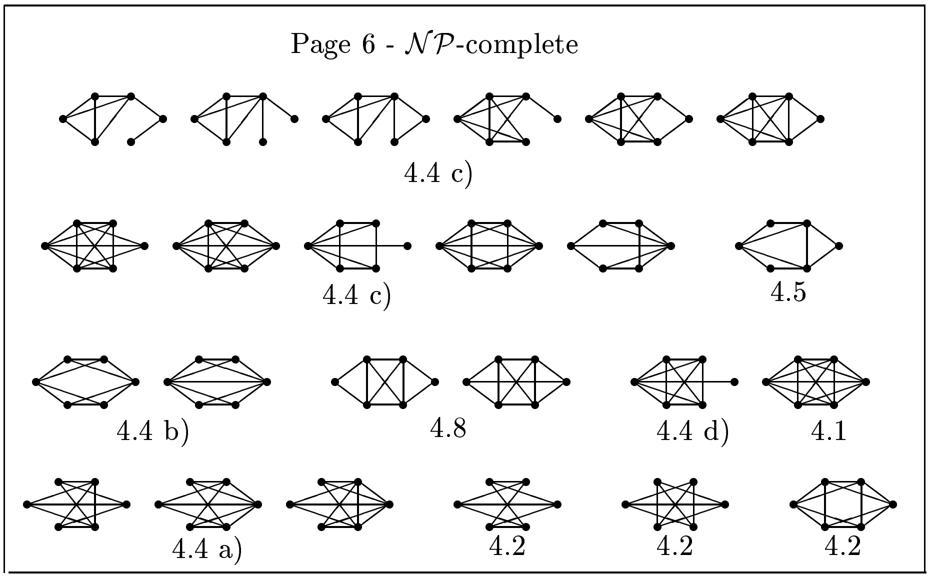
[15] W. Massey, *Algebraic Topology: An Introduction*, Harcourt, Brace and World, 1967.

Appendix

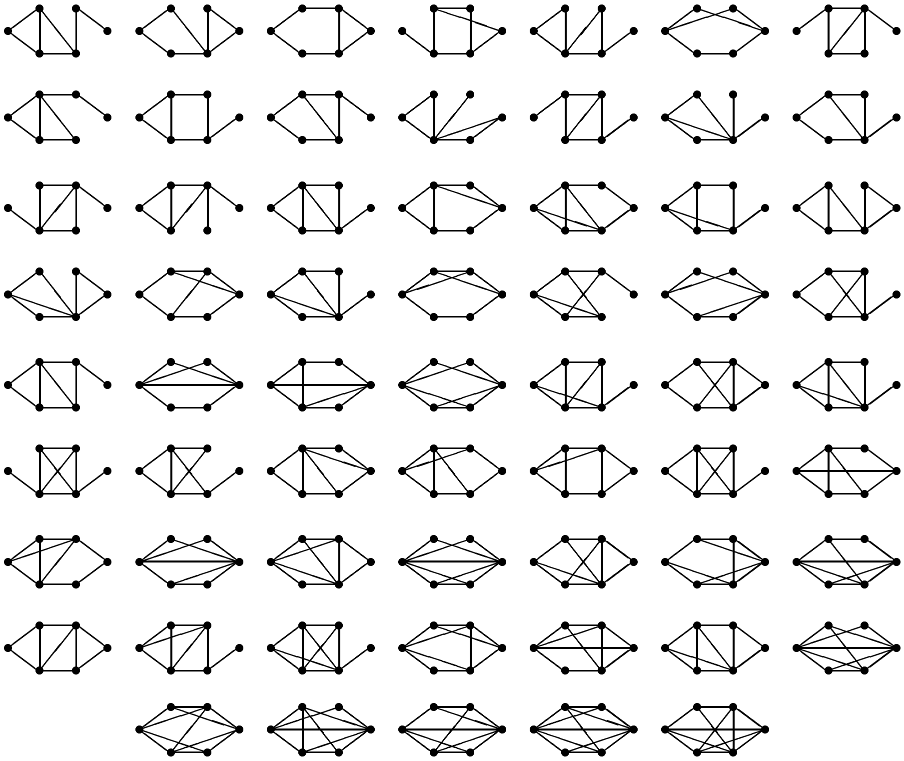
We list every connected, simple graph H on at most six vertices and at least two cycles, showing the complexity of the H-covering problem. Covering of simple graphs with at most one cycle is easy by Fact 3.1. By Fact 2.3 this resolves also the complexity of disconnected graphs having components on at most six vertices. The listing thus completes pages $1 \leq p \leq 6$ of the book on the complexity of the covering problem for simple graphs on p vertices.


Page 4
$\mathcal{NP}$-complete
Polynomial
4.1
3.5

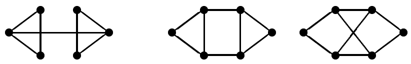

Page 5
$\mathcal{NP}$-complete
Polynomial
4.4 c)
3.5
3.3
3.3
4.1
3.5


Page 6 - $\mathcal{NP}$-complete
4.4 c)
4.4 c)
4.5
4.4 b)
4.8
4.4 d)
4.1
4.4 a)
4.2
4.2
4.2

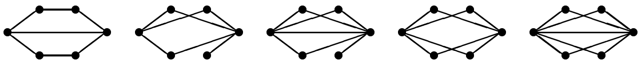
Page 6 - Polynomial



3.5



3.3 3.6



3.3 & 3.4