Discussiones Mathematicae
Graph Theory xx (xxxx) 1–31
https://doi.org/10.7151/dmgt.2540

A NOVEL APPROACH TO COVERS OF MULTIGRAPHS WITH SEMI-EDGES

Jiří Fiala and Michaela Seifrtová

Charles University, Department of Applied Mathematics
Malostranské nám. 25, Prague, Czech Republic

e-mail: fiala@kam.mff.cuni.cz
michaela.seifrtova@mff.cuni.cz

Abstract

We present three equivalent models for graphs and their covers, two of which are applicable to more general structures such as mixed colored directed multigraphs with semi-edges.

In this context, we extend the concept of equitable and degree partitions, and provide efficient algorithms for their calculation.

We demonstrate that the dart model introduces simpler concepts. Leveraging this model, we introduce a novel notion of the degree matrix for the general graph model. Additionally, we reassert and expand upon Leighton's theorem regarding the existence of a finite common cover for graphs containing semi-edges.

Keywords: graph coverings, topological graph theory.

2020 Mathematics Subject Classification: 05C60.

1. Introduction

In a topological context, a covering projection refers to a mapping that is both continuous and bijective within the neighborhood of any point in the domain. When we view a graph as the set of points of its crossing-free drawing, then the concept of topological covering projection naturally translates into graph homomorphism that acts as an isomorphism on the neighborhood of every vertex (up to vertices of degree two, for details see e.g. the survey [14]). This discrete variant of the topological covering space is established as *graph cover*.

Graph covers already appeared in the classical monograph of Reidemeister in 1932 [32, pages 109–114], called there *"Isomorphismus von Streckenkomplex* $\mathfrak{C}$

zu Streckenkomplex $\mathfrak{C}^*$*"*. Since then they appeared several times in constructions in algebraic and structural graph theory, sometimes indeed rediscovered [4, 7, 14–18, 30, 33]. Covers have applications in computer science within distributed computing environments [2] that led to the concept of common covers [3, 26]. Computational complexity of the decision problem whether a covering exists has also been thoroughly studied [1,5,12,13,22–25], but so far the full characterization is not known. These studies often involved the concept of *equitable partition* well-known from practical and theoretical approaches to the graph isomorphism problem [8] and graph spectra [10].

In the standard graph terminology developed in 1936 by König [21], a graph consists of a set of vertices and a set of edges that are pairs of vertices. (Although it is well known that the concepts of his predecessors such as Euler in the Seven bridges of Königsberg problem (1736) or Kirchhoff in electronics (1847) or Cayley (1874) and Sylvester (1878) in chemistry allow multiple edges.) For this model of König we use the term a *simple graph*.

More complex graph concepts involving multiple edges, loops, orientations, and edge colors emerge naturally for graph covers as the result of the degree reduction procedure proposed by Kratochvíl, Proskurowski and Telle [22]. They call these objects colored mixed directed multigraphs. Here, edges and vertices are matched by an incidence relation, and we call this concept the *incidence model*.

Semi-edges also naturally arose in a discrete model of branched coverings well known in algebraic topology [19]. The dart description of a graph with possible semi-edges was involved in studies of lifting automorphisms on discrete structures [27,28]. Recently, computational complexity of graph covering problems involving semi-edges was partially examined [6].

The *dart model* mentioned in the previous paragraph uses a different approach to the construction of graphs, where vertices and edges are built from even more elementary objects called darts. In this model, vertices establish one equivalence relation on darts, and edges another. This concept stems from the inherent operation of splitting of an edge into two parts (often called arcs) frequently used in topological and algebraic graph theory [4,17]. The initial formalization of a graph embedding in combinatorial terms dates back to Edmonds in 1960 [11]. He employed a concept that aligns with what could be perceived as the dart model of a graph for this purpose. The term *dart* was coined by Jones and Singerman in their 1978 paper on orientable maps [20].

The primary contribution of this paper lies in offering a cohesive perspective on the aforementioned graph models and delving into the respective advantages and disadvantages, particularly regarding their relationship to graph cover theory.

We showcase the advantages of the dart model through a concise and elementary proof of Leighton's common cover theorem [2, 26]. Our approach extends the scope of previous versions [31] in accommodating semi-edges, rendering our proof more comprehensive.

2. Graph Models

This section provides an overview of various approaches to defining a graph. We start with the conventional definition of a graph as a set of vertices and edges. Subsequently, we introduce an extended definition of a graph involving loops, directed edges and loops and semi-edges, each allowing for multiple occurrences and possible colors. From the variety of possible definitions, we present two. Both of these new definitions are mutually equivalent, and consistent also with the standard one, when restricted to simple graphs without isolated vertices. Hence all statements and properties of graphs remain consistent regardless of the specific definition obviating the need for constant specification. Nonetheless, utilizing a particular definition occasionally simplifies significantly the proof of these statements. To adhere to convention, we keep the concept of the graph in mind and, when necessary, explicitly emphasize the definition we are employing.

2.1. Simple graph

Definition. A (*simple*) *graph* is a pair (V, E), where V is a set whose elements are called vertices and E is a set of unordered pairs $\{u, v\}$ of distinct vertices.

In the following we will use a standard shorter notation for edges, an undirected edge $e = \{u, v\}$ will be denoted by uv. We call the edge uv incidental with vertices u and v. The degree of a vertex is the number of edges incident with this vertex. We say that a graph is *regular* if all its vertices have the same degree and we say it is *k-regular* if all its vertices have degree k.

2.2. Incidence graph model

The first generalization towards multigraphs with loops and semi-edges retains vertices as the fundamental building blocks.

Definition. A *graph* is a quadruple $(V, \Lambda, \iota, \kappa)$, with a set of *vertices* V and a set of *links* $\Lambda = E \cup L \cup S \cup \overrightarrow{E}$, where E is the set of undirected edges, L is the set of undirected loops, S is the set of semi-edges and $\overrightarrow{E}$ is the set of directed edges and loops. The function ι is an *incidence mapping* $\iota : \Lambda \longrightarrow \binom{V}{2} \cup V \cup (V \times V)$, such that $\iota(e) \in V$ for $e \in L \cup S$, $\iota(e) \in \binom{V}{2}$ for $e \in E$ and $\iota(e) \in V \times V$ is the ordered pair of vertices for $e \in \overrightarrow{E}$.

The mapping $\kappa : \Lambda \to C$ is a link *coloring*, where C is a set of *colors*. *Color classes* of κ are inclusion-wise maximal sets of links of the same color. This coloring need not be proper in the sense that adjacent edges may receive the same color.

In other words, since we allow multiple links of the same type incident with the same vertex (or with the same pair of vertices), the links are given by their names and the incidence mapping ι expresses which vertex (or vertices) 'belong' to a particular link. Observe, that a semi-edge has to be non-oriented.

The incidence definition allows us to consider various classical and some new concepts of structures related to graphs, including:

- simple graphs, where $\Lambda = E$ and ι is injective;
- multigraphs, where $\Lambda = E \cup L$;
- directed graph, where $\Lambda = \overrightarrow{E} \setminus \{vv\colon v \in V\}$ and ι is injective;
- semi-simple graphs, where $\Lambda = E \cup S$ and ι is injective;
- uncolored graphs, where κ is constant, i.e., it has only one color class.

The degree of a vertex is then defined as follows.

Definition. For a color $c \in C$, the *c-degree* of a vertex $v \in V$ is defined as

$$\deg^c(v) = |\{e\colon v \in \iota(e), e \in E \cup S, \kappa(e) = c\}| + 2|\{e\colon v = \iota(e), e \in L, \kappa(e) = c\}|$$

if $c \in \kappa(E \cup L \cup S)$, while for $c \in \kappa(\overrightarrow{E})$ we define

$$\deg^c(v) = (\deg^{c-}(v), \deg^{c+}(v))$$

where the *c-outdegree* is

$$\deg^{c-}(v) = |\{e\colon \iota(e) = (v, u) \text{ for some } (u, e \in \overrightarrow{E}, \kappa(e) = c\}|$$

and the *c-indegree* is

$$\deg^{c+}(v) = |\{e\colon \iota(e) = (u, v) \text{ for some } (u, e \in \overrightarrow{E}, \kappa(e) = c\}|.$$

The *total degree* of v is

$$\deg(v) = \sum_{c \in \kappa(E \cup L \cup S)} \deg^c(u) + \sum_{c \in \kappa(\overrightarrow{E})} (\deg^{c-}(u) + \deg^{c+}(u)).$$

The degree of a vertex is the number of edges incident with this vertex. A loop adds 2 to the degree of its vertex. This may not seem immediately intuitive, but it aligns naturally with considerations of graph embeddings on surfaces and becomes evident when exploring the definition of a covering projection. Moreover, this notion remains coherent with the concept of edges being the amalgamation of two semi-edges, a concept detailed in the subsequent sections.

2.3. Dart graph model

This section presents a different approach to graphs with semi-edges, using the set of so called darts as the fundamental set and defining both vertices and edges with the help of darts.

Definition. A *graph* is a quadruple (D, V, Λ, κ), where D is a set of *darts*, and where V and Λ are each a partition of D into disjoint sets and $\kappa : D \to C$ is a *dart coloring*. Moreover, all sets in Λ have size one or two.

Vertices are here the sets of darts forming the partition V, i.e., V is the set system on D, whose elements are disjoint subsets of D. The set of *links* Λ consists of three disjoint sets $\Lambda = E \cup L \cup S$, where E represents the *edges*, i.e., those links of Λ that have non-empty intersection with two distinct vertices from V, L are the *loops*, i.e., those 2-element sets of Λ that are subsets of some set from V, and S are the *semi-edges*, i.e., the 1-element sets from Λ.

For an illustration and comparison to the incidence graph model of Definition 2.2 see Figure 1.

The usual terminology that a vertex $v \in V$ is *incident* with an edge $e \in E$ or that distinct vertices u and v are *adjacent* can be expressed as $v \cap e \neq \emptyset$ and there exists $e \in E$ such that $u \cap e \neq \emptyset$ and $v \cap e \neq \emptyset$, respectively.

For a dart color $c \in C$, the *c-degree* of a vertex $v \in V$ is $\deg^c(v) = |\{d \in v \colon \kappa(d) = c\}|$. The *degree* of a vertex $v \in V$ is simply $\deg(v) = |v|$. Observe that the degree of a vertex is always positive, hence isolated vertices cannot be expressed in this model.

2.4. Terminology related to the dart model

The *multiedge* between u and v is an inclusion-wise maximal subset of links that are incident with both u and v, i.e., $\{e \in E \colon e \cap v \neq \emptyset \wedge v \cap e \neq \emptyset\}$ and the cardinality of this set is the *multiplicity* of the (multi)edge uv. In the same way we define the multiplicity of a loop or of a semi-edge.

As the concept of a graph built from darts might seem unusual, we review few further concepts that are well established for the classical definition.

A graph $H = (D_H, V_H, \Lambda_H, \kappa_H)$ is a *subgraph* of a graph $G = (D_G, V_G, \Lambda_G, \kappa_G)$ if their sets of darts satisfy $D_H \subseteq D_G$ and their partitions fulfill $V_H = V_G|_{D_H}$, $\Lambda_H = \Lambda_G|_{D_H}$ and $\kappa_H = \kappa_G|_{D_H}$.

A *path* is a graph formed by a sequence of distinct darts such that consecutive darts constitute either an edge or a vertex of degree 2. A path is *closed* if both the first pair and the last pair constitute edges. A closed path corresponds to the usual definition of a simple graph, which is a path. We also say that a closed path connects the vertex forming the first dart to the vertex of the last one. Our concept of a path is more general, as not all paths need to be closed, and such paths do not have counterpart in simple and incidence graph models. If the first pair and the last pair are vertices, then the path is *open*. In all other cases (including a sequence of length 1) the path is *half-way*.

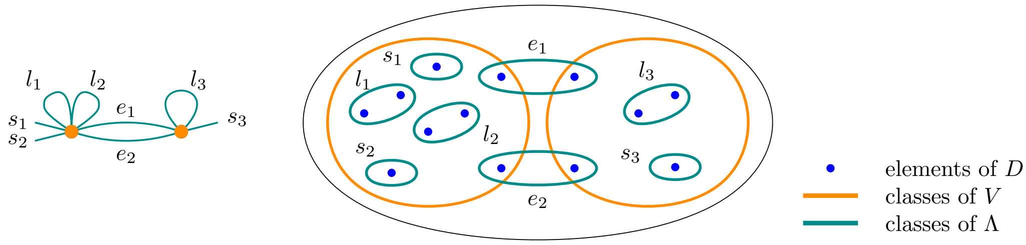


Figure 1. An example of a graph presented in the usual graph-theoretical way (left) and using the dart model by Definition 2.3 (right).

By a *component* of a graph we mean an inclusion-wise maximal induced subgraph such that each two of its vertices are connected by a subgraph isomorphic to a path. We say that a graph is *connected* if it has only a single component.

Semi-simple graphs have the property that whenever two darts are in the same equivalence class of Λ, they belong to different vertices, which are distinct equivalence classes of V, because there are no loops. For any two vertices u, v exists at most one link l such that $l \cap u \neq \emptyset$ and also $l \cap v \neq \emptyset$ (there are no multiple links). A *simple* graph then is a semi-simple graph where all equivalence classes of Λ have cardinality 2.

Table 1 depicts and summarizes the differences and relationships between the two graph models.

2.5. Transitions between the three models

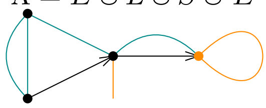
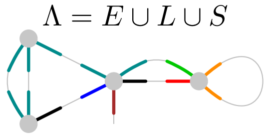
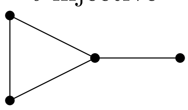
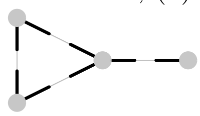
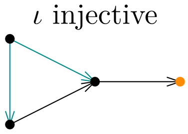
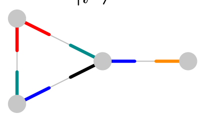
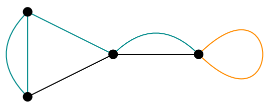
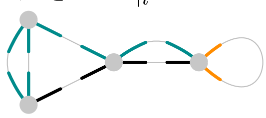
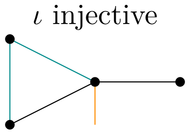
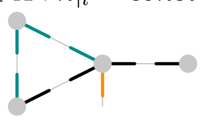
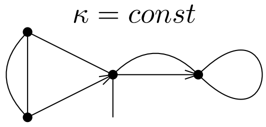
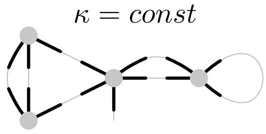
Upon closer examination, it is evident that Definitions 2.1, 2.2 and 2.3 are equivalent when restricted to simple graphs without isolated vertices.

To see that both our approaches to the general definition of a graph are equivalent (up to isolated vertices), we demonstrate the reciprocal conversion between the dart representation of a graph and the incidence representation.

To encode the orientations, we incorporate dart colors, as the darts corresponding to a directed edge will be differentiated by distinct colors.

Note that regardless of the specific color chosen, directed/bichromatic links can always be distinguished from undirected edges, loops, and semi-edges. To simplify the technical intricacies of our arguments while maintaining the necessary expressive power for graph covers, we make the following assumptions about colorings.

- In the incidence model, the directed edges and loops are assigned colors

Considered models of a graph	Incidence model $(V, \Lambda, \iota, \kappa)$	Dart model (D, V, Λ, κ)
	$\Lambda = E \cup L \cup S \cup \overrightarrow{E}$ 	$\Lambda = E \cup L \cup S$ 
Classical concepts: simple graph	$\Lambda = E$, $\kappa = const$, ι injective 	$\Lambda = E$, $\kappa = const$, $(*)$ 
(colored) directed graph	$\Lambda = \overrightarrow{E} \setminus \{vv \colon v \in V\}$, ι injective 	$\Lambda = E$, $\forall l \in \Lambda : \kappa\|_l \neq const$, $(*)$ 
(colored undirected) multigraph	$\Lambda = E \cup L$ 	$\Lambda = E \cup L$, $\forall l \in \Lambda : \kappa\|_l = const$ 
(colored) semi-simple graph	$\Lambda = E \cup S$, ι injective 	$\Lambda = E \cup S$, $\forall l \in \Lambda : \kappa\|_l = const$, $(*)$ 
uncolored graph	$\kappa = const$ 	$\kappa = const$ 

$(*)$ no link multiplicities, i.e., $\forall u, v \in V : |\{l \in \Lambda : l \cap u \neq \emptyset \wedge l \cap v \neq \emptyset\}| \leq 1$

Both columns display identical graphs, except for the uncolored graphs, where a distinction arises due to the requirement for at least two dart colors to interpret directions. In the case of undirected graphs, vertices remain uncolored. The assignment of distinct colors to adjacent vertices would implicitly imply directions on the incident edges. Vertices and links in the dart model are represented by gray disks and curves, respectively.

Table 1. Comparison of various graph models.

distinct from those used for undirected edges, loops and semi-edges, formally $\kappa(\overrightarrow{E}) \cap \kappa(E \cup L \cup S) = \emptyset$.

- In the dart model, darts within bichromatic links are assigned colors from a different subset than those within monochromatic links. Formally, the partition of Λ into $\Lambda_1 = \{l \in \Lambda \colon \exists c \in C \colon \kappa(l) = c\}$ and $\Lambda_2 = \Lambda \setminus \Lambda_1$ satisfies $\kappa(\Lambda_1) \cap \kappa(\Lambda_2) = \emptyset$.

To fulfill these assumptions, we can modify the colorings by introducing previously unused colors, ensuring the specified conditions are met. Specifically, for each $c \in \kappa(\overrightarrow{E}) \cap \kappa(E \cup L \cup S)$, we introduce a new color c' into the set of colors. We retain c for the directed links, while for every undirected link previously colored by c, we replace its color with c'. We follow a similar procedure for the dart model.

We demonstrate that within graphs without isolated vertices, both the incidence and dart models have identical expressive power.

Theorem 1. *There exists a surjective map from the class of incidence graph models without isolated vertices to the class of dart graph models, such that every dart graph model is isomorphic to an element of the image, and if $G = (V_G, \Lambda_G, \iota_G, \kappa_G)$ is mapped on $H = (D_H, V_H, \Lambda_H, \kappa_H)$, then there is a bijection $h_V : V_G \to V_H$, a bijection $h_\Lambda : \Lambda_G \to \Lambda_H$ and a bijection $h_\kappa : \kappa_G(E \cup L \cup S) \cup \kappa_G(\overrightarrow{E}) \times \{0,1\} \to \kappa_H(D_H)$ that preserve incidences.*

- $\forall v \in V_G, \forall e \in \Lambda : v \in \iota(e) \Rightarrow h_V(v) \cap h_\Lambda(e) \neq \emptyset$,

as well as each c-degree of each vertex, formally:

- $\forall v \in V_G, \forall c \in \kappa_G(E \cup L \cup S) \colon \deg(v)^c = \deg(h_V(v))^{h_\kappa(c)}$,
- $\forall v \in V_G, \forall c \in \kappa_G(\overrightarrow{E}) \colon \deg(v)^{c-} = \deg(h_V(v))^{h_\kappa(c,0)} \wedge \deg(v)^{c+} = \deg(h_V(v))^{h_\kappa(c,1)}$.

Proof. Given $(V_G, \Lambda_G, \iota_G, \kappa_G)$, we define the set of darts as the set of triples

$$\begin{aligned} D_H &= \{(\iota(e), e, 0) \colon e \in S\} \\ &\cup \{(\iota(e), e, 0), (\iota(e), e, 1) \colon e \in L\} \\ &\cup \{(u, e, 0), (v, e, 0) \colon \iota(e) = \{u, v\}, e \in E\} \\ &\cup \{(u, e, 0), (v, e, 1) \colon \iota(e) = (u, v), e \in \overrightarrow{E}\}. \end{aligned}$$

Observe that in the second row each undirected loop gives rise to two darts.

The partition V_H is given by the equivalence relation $\sim_{V_H}$: $(v_1, e_1, i) \sim_{V_H} (v_2, e_2, j)$ if $v_1 = v_2$, and the bijection h_V between the vertex sets V_G and V_H by $h_V(v) = \{(v, e, i) \in D_H\}$, i.e., we choose all triples whose first component is v.

The partition Λ_H and the bijection h_Λ is given by $\sim_{\Lambda_H}$: $(v_1, e_1, i) \sim_{\Lambda_H} (v_2, e_2, j)$ if $e_1 = e_2$ and thus $h_\Lambda(e) = \{(u, e, i) \in D_H\}$.

The coloring κ_H and the bijection h_κ is obtained from the coloring κ_G as follows:

- for undirected links, i.e., $e \in E \cup L \cup S$, $\kappa_H((v,e,i)) = \kappa_G(e)$ and thus h_κ is the identity,
- for directed links, i.e., if $e \in \overrightarrow{E}$, $\kappa_H((v,e,i)) = (\kappa_G(e), i)$.

The construction of H alongside the bijections h_V, h_Λ, and h_κ aligns well with both the definition of an edge incident with a vertex and the definition of the c-degree (as per Definition 2.2). Notably, we introduced two darts for each loop to contribute to the degree by two.

To complete our argument, we need to establish that this construction yields a surjective mapping between the two classes of structures. We demonstrate that the reverse construction is also viable and, in fact, simpler.

Given $H = (D_H, V_H, \Lambda_H, \kappa_H)$, where $V_G = V_H$, we define $\iota_G(e) = \{v\colon e \cap v \neq \emptyset\}$.

Uncolored vertices are represented by undirected links meeting the following conditions.

- $S = \{e \in \Lambda_H : |e| = 1\}$,
- $L = \{e \in \Lambda_H : e = \{d_1, d_2\} \wedge d_1 \neq d_2 \wedge \kappa_H(d_1) = \kappa_H(d_2) \wedge \exists v \in V_H\colon e \subseteq v\}$,
- $E = \{e \in \Lambda_H : e = \{d_1, d_2\} \wedge d_1 \neq d_2 \wedge \kappa_H(d_1) = \kappa_H(d_2) \wedge \neg\exists v \in V_H\colon e \subseteq v\}$.

The coloring of these undirected links is determined by $\kappa_G(e) = \kappa_H(e)$ — note that $\kappa_H(e)$ itself is a single color even if e consists of two darts.

Dealing with the directed links is a bit more complicated, as the direction cannot be uniquely derived from the bichromatic coloring of a link. To resolve this issue we choose a complete ordering $\prec$ of the color classes of κ_H and define directed links as follows.

- $\overrightarrow{E} = \{(d_1, d_2) : e = \{d_1, d_2\} \in \Lambda_H \wedge \kappa_H(d_1) \prec \kappa_H(d_2)\}$.

  The coloring on G is completed by setting $\kappa_G((d_1, d_2)) = (\kappa_H(d_1), \kappa_H(d_2))$.
  This step also concludes the construction of $\Lambda_G = E \cup L \cup S \cup \overrightarrow{E}$.
  Bijections h_V, h_Λ and h_κ are obtained as follows.

- $h_V(v) = v$, i.e., h_V is the identity on $V_G = V_H$;
- $h_\Lambda(e) = e$ for undirected edges and $h_\Lambda((u,v)) = \{u, v\}$ for directed ones;
- colors of undirected links are mapped onto colors of monochromatic links, i.e., $h_\kappa(c) = c$ when $c \in \kappa_G(E \cup L \cup S)$; and finally
- for every color $c \in \kappa_G(\overrightarrow{E})$ used on directed links we know by the construction of the coloring κ_G that the color c is in fact a pair $c = (c_0, c_1)$. Thus we may define $h_\kappa((c,i)) = c_i$ for both choices of $i \in \{0, 1\}$.

Finally, given a dart graph model H, we first select an ordering $\prec$, followed by the construction of an incidence graph model G. From this intermediate graph, we derive a dart graph model denoted as H'. Through a tedious yet straightforward analysis, one can confirm the isomorphism between H and H'. Consequently, the mapping between these two classes of models is surjective. ■

3. Graph Covers

3.1. Simple and semi-simple graph covers

Definition. We say that a graph $G = (V_G, E_G)$ *covers* a connected graph $H = (V_H, E_H)$ (denoted as $G \longrightarrow H$) if there exists a map $f\colon V_G \to V_H$ such that the map f is a local isomorphism, i.e., f maps the neighborhood of each vertex u bijectively on the neighborhood of $f(u)$.

As a consequence of the assumption that H is connected, the map f is surjective. This follows from the well known path lifting theorem [32].

From now on we will follow the convention that G (*guest*) is a source graph and H (*host*) is the target of the map f, which we will often call a *covering projection.*

3.2. Graph covers on incidence models

To extend the concept of graph covers onto multigraphs we have to consider also a mapping of links. While in the case of simple graphs, this mapping is uniquely determined by the vertex mapping, such definitiveness no longer holds when loops and multiple edges are present. Additionally, it is crucial to differentiate between various types of links and respect the coloring.

Definition. A graph $G = (V_G, \Lambda_G, \iota_G, \kappa_G)$ covers a graph $H = (V_H, \Lambda_H, \iota_H, \kappa_H)$ if and only if G allows a pair of mappings $f_V : V_G \longrightarrow V_H$ and $f_\Lambda : \Lambda_G \longrightarrow \Lambda_H$ such that the following conditions are satisfied.

1. For every link $e \in \Lambda_G$ we have $\iota_H(f_\Lambda(e)) = f_V(\iota_G(e))$, i.e., the pair of maps f_V and f_Λ encode a homomorphism from G to H, with respect to the incidence relations ι_G and ι_H.
   (We use the usual convention that $f_V(\{u, v\}) = \{f_V(u), f_V(v)\}$ and also $f_V((u, v)) = (f_V(u), f_V(v))$ to be able to apply f_V also onto unordered and ordered pairs.)
2. Each link $e \in \Lambda_G$ satisfies $\kappa_G(e) = \kappa_H(f_\Lambda(e))$, i.e., this homomorphism is color-preserving.

3. For every semi-edge $e \in S_H$, $f_\Lambda^{-1}(e)$ is a disjoint union of edges and semi-edges spanning all vertices $u \in V_G$ such that $f_V(u) = \iota_H(e)$, i.e., the preimage of a semi-edge e is a 1-factor on $f_V^{-1}(\iota_H(e))$.
4. For every loop $e \in L_H$, $f_\Lambda^{-1}(e)$ is a disjoint union of loops and cycles spanning all vertices $u \in V_G$ such that $f_V(u) = \iota_H(e)$, i.e., the preimage of a loop e is a 2-factor on $f_V^{-1}(\iota_H(e))$.
5. For every edge $e \in E_H$, $f_\Lambda^{-1}(e)$ is a disjoint union of edges spanning all vertices $u \in V_G$ such that $f_V(u) \in \iota_H(e)$, i.e., the preimage of an edge e is a perfect matching between the two sets forming $f_V^{-1}(\iota_H(e))$.
6. For every directed loop $e \in \overrightarrow{E_H}$, $f_\Lambda^{-1}(e)$ is a disjoint union of directed loops and cycles spanning all vertices $u \in V_G$ such that $f_V(u) \in \iota_H(e)$, i.e., the preimage of a directed loop e is a directed 2-factor on $f_V^{-1}(\iota_H(e))$.
(Here we use the convention that a directed link viewed as an ordered pair (x, y) contains both of its elements $x, y \in (x, y)$.)
7. For every directed edge $e \in \overrightarrow{E_H}$, $f_\Lambda^{-1}(e)$ is a disjoint union of directed edges spanning all vertices $u \in V_G$ such that $f_V(u) \in \iota_H(e)$, i.e., the preimage of a directed edge e is a directed perfect matching between the two sets forming $f_V^{-1}(\iota_H(e))$.
(Note that the direction is uniquely determined when using the convention stated in the first condition.)

See an example of a covering projection depicted in Figure 2(a).

Conditions 1 and 3 imply that semi-edges are mapped onto semi-edges. Similarly, we can infer that (directed) loops are also correspondingly mapped to (directed) loops.

Conversely, undirected edges can be mapped onto undirected edges, undirected loops and semi-edges, while directed edges only allow mapping onto directed edges and directed loops.

Conditions 3–7 express that the cover is locally bijective. For every undirected edge or semi-edge e incident with $f_V(u)$ in H, there is exactly one ordinary edge or semi-edge (but not both) of G which is incident with u and mapped to e by f_Λ. For every undirected loop e incident with $f_V(u)$ in H, there is exactly one loop or exactly two ordinary edges (but not both) of G which are incident with u and mapped to e by f_Λ. Analogously for directed links.

Proposition 2. *For simple graphs, Definition* 3.1 *and Definition* 3.2 *are equivalent.*

Proof. For both graphs G and H we first identify the sets V in the standard and incidence models, as well as we identify E in the first model with $\Lambda = E$ in the other. Thus for an edge $e = \{u, v\} \in \Lambda$, the incidence mapping ι has in both graphs the form $\iota(e) = \iota(\{u, v\}) = \{u, v\}$.

Let $G \longrightarrow H$ according to Definition 3.1, i.e., there exists a map $f\colon V_G \to V_H$, which is a local isomorphism, mapping the neighborhood of each vertex u bijectively on the neighborhood of the vertex $f(u)$. We set $f_V = f$. This vertex mapping naturally defines the mapping of edges $f_\Lambda\colon \Lambda_G \to \Lambda_H$ as $f_\Lambda(\{u,v\}) = \{f(u), f(v)\}$. Note that f_Λ is well defined as f is a homomorphism.

Now we show that these maps fulfill the first and the fifth condition of Definition 3.2. The other conditions need not be verified as they relate to sorts of links that do not appear in G or in H.

By the definition of f_Λ, the incidence relations ι_G and ι_H commute with f_V and f_Λ. Formally, for any $e = \{u,v\} \in \Lambda_G$ we have $\iota_H(f_\Lambda(e)) = \iota_H(f_\Lambda(\{u,v\})) = \iota_H(\{f(u), f(v)\}) = \{f(u), f(v)\}$ as well as we have $f_V(\iota_G(e)) = f_V(\iota_G(\{u,v\})) = f_V(\{u,v\}) = f(\{u,v\}) = \{f(u), f(v)\}$.

When we choose any $e = \{u,v\} \in \Lambda_H$, then f is bijective on the neighborhood of any $f_V^{-1}(u)$, i.e., $f_\Lambda^{-1}(e)$ is a perfect matching between $f_V^{-1}(u)$ and $f_V^{-1}(v)$.

For the reverse implication, suppose that G covers H according to Definition 3.2, i.e., consider mappings $f_V : V_G \longrightarrow V_H$ and $f_\Lambda : \Lambda_G \longrightarrow \Lambda_H$. The map f_V is a homomorphism due to condition 1. As for any edge $e \in \Lambda_H$ the preimage $f^{-1}(e)$ is a perfect matching by condition 5. It follows that for any vertex $u \in V_G$ each edge incident with $f(u)$ has a single preimage incident with u. Therefore f is a local bijection, as required by the standard Definition 3.1. ■

3.3. Graph covers on dart models

The lengthy and technical Definition 3.2 of covers using the incidence relation stands in contrast to the concise and elegant Definition 3.3 formulated in terms of darts.

Definition. A graph $G = (D_G, V_G, \Lambda_G, \kappa_G)$ *covers* a connected graph $H = (D_H, V_H, \Lambda_H, \kappa_H)$ if there exists a map $f_D\colon D_G \to D_H$ such that the following conditions are satisfied.

1. For every $u \in V_G$, there exists a $u' \in V_H$ such that the restriction of f_D onto u forms a bijection between u and u'.
2. For every $d \in D_G$ we have $\kappa_G(d) = \kappa_H(f_D(d))$.
3. For every $e \in \Lambda_G$, there is an $e' \in \Lambda_H$ such that $f_D(e) = e'$.

For an example see Figure 2(b).

We now show that Definition 3.2 of a graph cover in the incidence model is equivalent to Definition 3.3 in the dart model.

To simplify our expressions, instead of using bijections h_V and h_Λ we simply do not distinguish between V_G in the dart definition and $V_{G'}$ in the incidence definition and use in both the same symbol V_G. Analogously for the sets of links Λ.

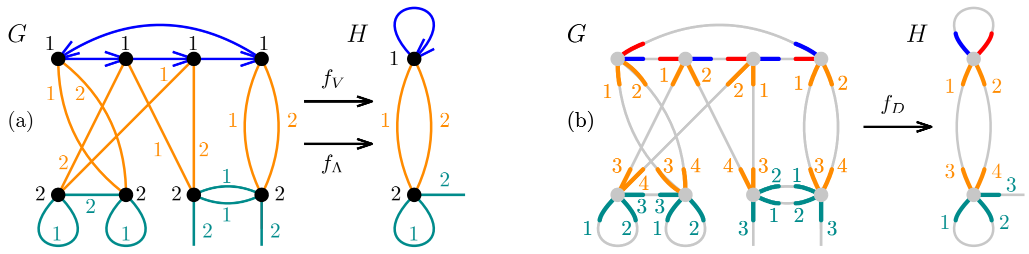


Figure 2. Covering between colored graphs G and H. Mappings f_V, f_Λ and f_D are indicated by colors and if more objects of the same color are in H, then also by numbers. (a) The incidence model. (b) The dart model. Vertices and links are indicated by gray disks and curves.

Theorem 3. *Let $G = (D_G, V_G, \Lambda_G, \kappa_G)$ and $H = (D_H, V_H, \Lambda_H, \kappa_H)$ be dart graph models and let $G' = (V_G, \Lambda_G, \iota_{G'}, \kappa_{G'})$ and $H' = (V_H, \Lambda_H, \iota_{H'}, \kappa_{H'})$ be their corresponding incidence models obtained through the surjective map described in Theorem* 1. *Then G covers H if and only if G' covers H'.*

Proof. Let $f_D\colon D_G \to D_H$ be a covering from G to H. As f_D respects the partitions V_H and Λ_H, mappings $f_V\colon V_G \to V_H$ and $f_\Lambda\colon \Lambda_G \to \Lambda_H$ defined by

$$f_V(u) = f_D(u) \quad \text{and} \quad f_\Lambda(e) = f_D(e)$$

are in fact well defined and are homomorphisms respecting the incidence functions $\iota_{G'}$, $\iota_{H'}$, and colorings $\kappa_{G'}$, $\kappa_{H'}$ as required in conditions 1 and 2 of Definition 3.2.

Formally, for any $e \in \Lambda_G$, examining the left-hand side of condition 1, we can expand it as $\iota_{H'}(f_\Lambda(e)) = \{v \in V_H : f_\Lambda(e) \cap v \neq \emptyset\} = \{v \in V_H : f_D(e) \cap v \neq \emptyset\}$, while the right-hand side yields the same expression via $f_V(\iota_{G'}(e)) = f_V(\{u \in V_G : e \cap u \neq \emptyset\}) = f_D(\{u \in V_G : e \cap u \neq \emptyset\}) = \{v \in V_H : f_D(e) \cap v \neq \emptyset\}$.

For condition 2 consider any $e \in \Lambda_G$. We have $\kappa_{G'}(e) = h_\kappa(\kappa_G(e)) = h_\kappa(\kappa_H(f_D(e))) = h_\kappa(\kappa_H(f_\Lambda(e))) = \kappa_{H'}(f_\Lambda(e))$.

Conditions 3–7 come from the fact that for every vertex $u \in V_G$ the restriction of $f_D|_u$ is a bijection between u and $f_D(u) = f_V(u)$.

In particular for undirected links (conditions 3–5) we argue as follows. If $e' = \{d_1, d_2\} \in L_H$ is a loop and $d_1, d_2 \in u'$, it follows from the fact that f_D is a bijection between each vertex u and its image $f_D(u)$, that $f_D^{-1}(e')$ contributes two darts to each vertex of $f_V^{-1}(u')$, i.e., $f_\Lambda^{-1}(e')$ induces a graph where each vertex is of degree two. Similarly, if e is an edge, its preimage is a set of edges. The intersection of each of the edges of $f_\Lambda^{-1}(e)$ with each vertex is at most one, hence $f_\Lambda^{-1}(e)$ induces a subgraph with degrees one — a matching in G. Finally, if $e' = \{d'\}$ with $d' \in u'$, then $f_D^{-1}(d')$ is a set of darts such that each dart contributes one to the degree of each vertex of $f_V^{-1}(u')$. In accordance with the third condition of Definition 3.3, the set $f_\Lambda^{-1}(e')$ consists of semi-edges and edges that form a 1-factor on $f_V^{-1}(u')$.

The case for directed links (conditions 6 and 7) follows a similar pattern. When $e' = (d_1, d_2)$ is a directed loop and $d_1, d_2 \in u'$, then $f_D^{-1}(e')$ contributes one dart to the indegree of each vertex of $f_V^{-1}(u')$ as well as one dart to the outdegree, thus $f_\Lambda^{-1}(e')$ is a directed 2-factor. Finally, for a directed edge $e' = (d_1, d_2)$ with $d_1 \in u'$, $d_2 \in v'$ we get analogously that $f_D^{-1}(e')$ contributes one dart to the outdegree of each vertex of $f_V^{-1}(u')$ and one dart to the indegree of $f_V^{-1}(v')$, yielding a directed perfect matching from the set $f_V^{-1}(u')$ to $f_V^{-1}(v')$.

For the reverse implication, suppose $G' = (V_G, \Lambda_G, \iota_{G'}, \kappa_{G'})$ covers $H' = (V_H, \Lambda_H, \iota_{H'}, \kappa_{H'})$ and we are given the covering projections $f_V \colon V_G \to V_H$ and $f_\Lambda \colon \Lambda_G \to \Lambda_H$ fulfilling all the conditions of Definition 3.2. Recall the definition of the set of darts as previously described in the proof of Theorem 1 as the set of triples. We hence define the covering $f_D : D_G \to D_H$ by $f_D((u,e,i)) = (f_V(u), f_\Lambda(e), i)$.

This mapping is well defined due to condition 1 of Definition 3.2, see the condition $u \in \iota_{G'}(e) \Rightarrow f_V(u) \in f_V(\iota_{G'}(e)) = \iota_{H'}(f_\Lambda(e))$.

Colors are preserved as well, since for a dart $d = (u,e,i) \in D_G$ belonging to an undirected link e we have $\kappa_G(d) = \kappa_G((u,e,i)) = \kappa_{G'}(e) = \kappa_{H'}(f_\Lambda(e)) = \kappa_H((f_V(u), f_\Lambda(e), j)) = \kappa_H(f_D(d))$.

For directed links we have the same identity with an additional constraint $i = j$. (Note that the third component of the triple is irrelevant for undirected links.)

Regarding a link e in the graph G, our formal argument suggests that $f_D(e) = \{f_D(d) \colon d \in e\} = \{f_D((u,e,i)) \colon u \in \iota_G(e)\} = \{(f_V(u), f_\Lambda(e), i) \colon u \in \iota_G(e)\} = f_\Lambda(e) = e' \in \Lambda_H$ to get the third condition of Definition 3.3.

Finally, to argue the first condition of Definition 3.3 we have to combine conditions 3–7 of Definition 3.2 to see that f_D preserves vertex degrees and thus it is a local bijection. So for $u \in V_G$, $u' = f_V(u)$, any color $c \in \kappa_G(\Lambda_G)$ and undirected degree we have

$$
\begin{aligned}
\deg_G^c(u) &= |\{d \in u \colon \kappa_G(d) = c\}| \\
&= |\{(u,e,i) \colon \iota_{G'}(e) \ni u, e \in E_{G'} \cup S_{G'} \cup L_{G'}, h_\kappa(\kappa_{G'}(e)) = c\}| \\
&= |\{e \colon \iota_{G'}(e) \ni u, e \in E_{G'} \cup S_{G'}, h_\kappa(\kappa_{G'}(e)) = c\}| \\
&\quad + 2|\{e \colon \iota_{G'}(e) = u, e \in L_{G'}, h_\kappa(\kappa_{G'}(e)) = c\}| \\
&= |\{e' \colon e' = f_\Lambda(e), \iota_{G'}(e) \ni u, e \in E_{G'} \cup S_{G'}, h_\kappa(\kappa_{G'}(e)) = c\}| \\
&\quad + 2|\{e' \colon e' = f_\Lambda(e), \iota_{G'}(e) = u, e \in L_{G'}, h_\kappa(\kappa_{G'}(e)) = c\}| \\
&= |\{e' \colon \iota_{H'}(e') \ni u', e' \in E_{H'} \cup S_{H'}, h_\kappa(\kappa_{H'}(e')) = c\}| \\
&\quad + 2|\{e' \colon \iota_{H'}(e') = u', e' \in L_{H'}, h_\kappa(\kappa_{H'}(e')) = c\}|
\end{aligned}
$$

$$
\begin{aligned}
&= |\{(u', e', i) \colon \iota_{H'}(e') \ni u', e' \in E_{H'} \cup S_{H'} \cup L_{H'}, h_\kappa(\kappa_{H'}(e')) = c\}| \\
&= |\{d \in u' \colon \kappa_H(d) = c\}| \ = \ \deg^c_H(u').
\end{aligned}
$$

For colors of directed links we maintain outdegrees as

$$
\begin{aligned}
\deg^{c-}_G(u) \ &= \ |\{d \in u \colon \kappa_G(d) = c\}| \\
&= |\{(u, e, 0) \colon \iota_{G'}(e) \ni u, e \in \overrightarrow{E}_{G'}, h_\kappa(\kappa_{G'}(e)) = c\}| \\
&= |\{e \colon (u, \cdot) = \iota_{G'}(e), e \in \overrightarrow{E}_{G'}, h_\kappa(\kappa_{G'}(e)) = c\}| \\
&= |\{e' \colon e' = f_\Lambda(e), (u, \cdot) = \iota_{G'}(e), e \in \overrightarrow{E}_{G'}, h_\kappa(\kappa_{G'}(e)) = c\}| \\
&= |\{e' \colon (u', \cdot) = \iota_{H'}(e'), e' \in \overrightarrow{E}_{H'}, h_\kappa(\kappa_{H'}(e')) = c\}| \\
&= |\{(u', e', 0) \colon \iota_{H'}(e') \ni u', e' \in \overrightarrow{E}_{H'}, h_\kappa(\kappa_{H'}(e')) = c\}| \\
&= |\{d \in u' \colon \kappa_H(d) = c\}| \ = \ \deg^{c-}_H(u').
\end{aligned}
$$

Likewise, when considering the indegree $\deg^{c+}$ where the third component of the triple is 1, we use the second entry of pairs representing directed edges. ■

4. Degree Reduction and Vertex Colors

For the problem of determining the existence of a graph covering projection between two graphs, Kratochvíl *et al.* designed degree reduction procedures. These procedures allow us to focus solely on graphs with a degree of at least three [22]. They are based on the idea that the existence of a leaf can be represented by the color of its neighbor and that a path between two vertices, each with a degree of at least 3, where all intermediate vertices have a degree of 2, can be substituted with a potentially colored and directed edge. We present their approach, which reduces vertices of degrees 1 and 2 in a dart graph model, as the algorithm DegreeReduction. The pseudocode for this algorithm is detailed in Algorithm 1. (The terminology we utilize was introduced in Section 2.4.)

The goal of first **while** loop is to eliminate all vertices of degree 1.

In the second **while** loop, our goal is to first identify a maximal subgraph P in G with all internal vertices having a degree of at least two. Subsequently, we replace this subgraph with an appropriate link. Such P induces in G either a path or a cycle with a single vertex of degree at least 3. In the latter case $u = v$, and to replace the cycle is by a loop we need to distinguish darts d_u and d'_v we utilize the prime notation.

It is important to note that when $G' = \text{DegreeReduction}(G)$, G' has at most as many darts as G. Additionally, the algorithm runs in quadratic time; both **while** loops could execute at most $|V_G|$ times, and each iteration requires linear time.

Input: a graph $G = (D, V, \Lambda, \kappa)$
while *G contains a vertex $v \in V$ of degree 1 with a neighbor u of degree at least 2* **do**
  let $e = \{d, d'\} \in \Lambda$ be the link connecting u to v with $v = \{d\}$, $d' \in u$;
  foreach *dart $d_u \in u, d_u \neq d'$* **do**
    insert the triple $(1, \kappa(d), \kappa(d'))$ into $\kappa(d_u)$;
    order $\kappa(d_u)$ by $\preceq$
  end
  remove d, d' from D, v from V, e from Λ and d' from u;
end
while *a component of G distinct from a cycle and from a path contains a vertex $v \in V$ of degree 2* **do**
  let $P = (d_1, \ldots, d_k)$ be the inclusion-wise maximal subgraph of G containing v that is isomorphic to a path or a cycle, and for which all vertices of G that intersect $\{d_2, \ldots, d_{k-1}\}$ have degree 2;
  if *P is half-way* **then**
    assume without loss of generality that $\{d_1, d_2\} \in \Lambda$, as otherwise we may reverse P;
    let u be the vertex of G containing d_1;
    add a new link containing only one dart d_u;
    insert d_u into u and color d_u by $(2, \kappa(d_1), \ldots, \kappa(d_k))$
  else if *P is closed* **then**
    let u, v be the (not necessarily distinct) vertices of G, where $d_1 \in u$ and $d_k \in v$;
    add a new link containing two new darts d_u and d'_v;
    insert d_u into u and d'_v into v;
    color d_u by $\left(2, \kappa(d_1), \ldots, \kappa\left(d_{\frac{k}{2}}\right)\right)$ and d'_v by $\left(2, \kappa(d_k), \ldots, \kappa\left(d_{\frac{k}{2}+1}\right)\right)$
  end
  remove $d_1, \ldots, d_k$ from D and restrict all links and vertices to the new set D;
end
return G

Algorithm 1: DEGREEREDUCTION

The result G' of DegreeReduction(G) is unique and encodes the original graph G up to the following two exceptions.

- When G is a simple tree, then a single edge remains. Its coloring encodes the original tree, but it is not unique as it depends on the choice of the remaining edge.
- When we eliminate all vertices of degree one in a tree G with two semi-edges incident with distinct vertices, then we are left with an open path which contains two maximal half-way paths.

However, these cases are simple from the perspective of computational complexity. To maintain clarity in our presentation, we exclude such cases from our reasoning as they could complicate the algorithm.

The following proposition characterizes outcomes of the Algorithm DegreeReduction.

Proposition 4. *If a connected graph G is neither a simple tree, nor a tree with two semi-edges incident with distinct vertices, then* DegreeReduction(G) *is a unique*

(i) *single vertex graph consisting of only one or two semi-edges, if G is a tree with one semi-edge or with two semi-edges incident with the same vertex,*

(ii) *cycle, if G is a simple cactus,*

(iii) *graph of minimum degree* 3 *otherwise.*

Moreover, from DegreeReduction(G) *we may reconstruct G if the coloring of G does not use symbols* 1 *or* 2*.*

Proof. We first describe the two reverse operations.

- If d_u is a dart colored by $(2, c_1, \ldots, c_k)$, then we replace it by a path $d_1, \ldots, d_k$ such that each d_i is colored by c_i;
- if u is a vertex such that all darts $d \in u$ have color $(1, c_u, c_v)$, then we add a link (d_u, d_v) such that $d_u \in u$, d_v forms a new vertex v of degree 1, the new links are colored by $\kappa(d_u) = c_u$ and $\kappa(d_v) = c_v$, and the triple $(1, c_u, c_v)$ is removed from the colors of all other darts in u.

Note that the assumption that 1 and 2 are not used on the coloring of G allows us to distinguish between the original colors and those introduced by the algorithm.

The order of processing vertices of degree one in the first **while** loop is irrelevant. Once this loop is completed, the colors added to links incident with any resulting vertex u of degree at least two encode the entire tree stemming from u, akin to the well-known tree isomorphism algorithm.

Note that in a tree with a single semi-edge, the leaves are sequentially pruned until only one semi-edge remains. In the case of a tree with two semi-edges incident with the same vertex, the result is a one-vertex graph with two semi-edges. ■

Theorem 5. *All connected non-trees G and H of maximum degree at least* 3 *satisfy that G covers H if and only if* DEGREEREDUCTION(G) *covers* DEGREE-REDUCTION(H).

Proof. For the forward direction assume that $f : D_G \to D_H$ is a covering. We synchronize the execution of DEGREEREDUCTION on G with that on H as follows.

When a vertex $v = \{d\}$ in H is eliminated, we identify $f^{-1}(d)$ and eliminate from G all vertices formed from the darts of $f^{-1}(d)$. By local injectivity these are of degree one, and as f preserves colors, it will also preserve the new colors after this step.

Similarly, when a path P is removed from H, we identify its preimage in G and eliminate these so that after this step the mapping f is still a covering.

In the opposite direction, assume that some graph G' obtained as DEGREE-REDUCTION(G) covers the graph H' = DEGREEREDUCTION(H). Once more, we synchronize the reconstruction of G with that of H. When we reconstruct a vertex v of degree 1 that is a neighbor of u, we identify the set $f^{-1}(u)$ and for each its vertex we reconstruct one neighbor of degree one. The coloring of new links in G' is extended by using the same colors as on the new link in H'.

Likewise, when a path P in H' is reconstructed from a link e, we identify $f^{-1}(e)$ and reconstruct the corresponding collection of paths with an adequate coloring. ■

Some papers on this topic utilize colored edges and vertices. However, as demonstrated by our approach, a graph with colored vertices and edges may be replaced by another graph with refined edge coloring, which also depends on the colors of the vertices incident with those edges.

5. EQUITABLE AND DEGREE PARTITIONS AND THEIR MATRICES

As graph covers are local isomorphisms, they must preserve the degree of any vertex as well as the degrees of its neighbors, the degrees of neighbors of neighbors, etc. In this context, we define an equivalence relation on vertices in simple graphs or incidence models. This relation ensures that vertices within the same class cannot be distinguished based on their degrees, the degrees of their neighbors, and so on. We then show that such an equivalence can also be generalized to the dart model with a simple and natural matrix description.

5.1. Partitions of simple graphs

The concept of vertex partition that must be preserved under any graph isomorphism was introduced by Corneil [8, 9]. For our purposes we rephrase it as follows.

Definition. An *equitable partition* of a simple graph $G = (V, E)$ is a partition $\mathcal{B}$ of the set V into disjoint *blocks* $B_1, B_2, \ldots$ such that whenever vertices u and v belong to the same block B_i, then they have the same number of neighbors in every block, formally $\forall i, j : u, v \in B_i \Rightarrow |N(u) \cap B_j| = |N(v) \cap B_j|$.

The matrix M whose entries are $m_{i,j} = |N(u) \cap B_j|$ for $u \in B_i$ is called the *matrix of the equitable partition.*

Definition. A *degree partition* is the equitable partition that minimizes the number of blocks.

Observe that the degree partition of a graph always exists as in the worst case, each vertex defines its own block. Degree partition can be obtained by an iterative procedure DegreePartition whose pseudocode is shown in Algorithm 2. As in each iteration the new partition is sorted we get a unique *canonical* ordering of the final degree partition $\mathcal{B}$.

Definition. The matrix of the canonically ordered degree partition of a graph G is called the *degree matrix* of G.

Input: a graph $G = (V, E)$
Initialize: set $\mathcal{B}^0 = \{B_1^0\}$, $B_1^0 = V$ and $i = 0$.
repeat
- For each vertex $u \in V$ set a vector $\bar{u}^i$ having on the jth position the number of neighbors of u in the jth block, i.e., $\bar{u}^i = (u_1^i, u_2^i, \ldots)$, where $u_j^i = |N(u) \cap B_j^i|$ for all $j \in \{1, 2, \ldots, |\mathcal{B}^i|\}$;
- Refine the partition into blocks containing vertices with the same vectors, i.e., create a new set system $\mathcal{B}^{i+1} = \{B_1^{i+1}, B_2^{i+1}, \ldots\}$ so that whenever $u, v \in B_j^{i+1}$, then $\bar{u}^i = \bar{v}^i$;
- Sort $\mathcal{B}^{i+1}$ by the lexicographic ordering of the corresponding vectors $\bar{u}^i$;
- Set $i = i + 1$

until *until no refinement of blocks is executed, i.e.,* $\mathcal{B}^{i+1} = \mathcal{B}^i$;
return $\mathcal{B}^i$

Algorithm 2: DegreePartition

The following statement is well known [15].

Theorem 6. *If a graph G covers a connected graph H, then they have identical degree matrices.*

5.2. Partitions in incidence graph models

Definition. An *equitable partition* of a graph $G = (V, \Lambda, \iota)$, for $\Lambda = E \cup L \cup S \cup \overrightarrow{E}$ is a partition $\mathcal{B}$ of the set V into disjoint blocks $B_1, B_2, \ldots$ such that whenever vertices u and u' belong to the same block B_i, then

1. they have the same number of undirected edges towards any other block B_j, formally $\forall j \neq i : |\{e \in E : u \in \iota(e) \wedge \iota(e) \cap B_j \neq \emptyset\}| = |\{e \in E : u' \in \iota(e) \wedge \iota(e) \cap B_j \neq \emptyset\}|$,
2. they have the same number of directed edges towards any block B_j, for incoming edges, formally $\forall j : |\{e \in \overrightarrow{E} : u = \iota(e)_1 \wedge \iota(e)_2 \in B_j\}| = |\{e \in \overrightarrow{E} : u' = \iota(e)_1 \wedge \iota(e)_2 \in B_j\}|$ as well as for outgoing edges, formally $\forall j : |\{e \in \overrightarrow{E} : u = \iota(e)_2 \wedge \iota(e)_1 \in B_j\}| = |\{e \in \overrightarrow{E} : u' = \iota(e)_2 \wedge \iota(e)_1 \in B_j\}|$,
3. within the block B_i they have the same number of incident edges, loops and semi-edges, where loops are counted twice, formally $|\{e \in E \cup S : u \in \iota(e) \wedge \iota(e) \subseteq B_i\}| + 2|\{e \in L : u = \iota(e)\}| = |\{e \in E \cup S : u' \in \iota(e) \wedge \iota(e) \subseteq B_i\}| + 2|\{e \in L : u' = \iota(e)\}|$.

Definition. An *equitable partition* of a colored graph $G = (V, \Lambda, \iota, \kappa)$, for $\Lambda = E \cup L \cup S \cup \overrightarrow{E}$ is a partition $\mathcal{B}$ of the set V into disjoint blocks $B_1, B_2, \ldots$ such that whenever vertices u and u' belong to the same block B_i, then

1. they have the same number of undirected edges of the same color c towards any other block B_j, formally $\forall c, \forall j \neq i : |\{e \in E : u \in \iota(e) \wedge \iota(e) \cap B_j \neq \emptyset \wedge \kappa(e) = c\}| = |\{e \in E : u' \in \iota(e) \wedge \iota(e) \cap B_j \neq \emptyset \wedge \kappa(e) = c\}|$,
2. they have the same number of directed edges of the same color c towards any block B_j, for incoming edges, formally $\forall c, \forall j : |\{e \in \overrightarrow{E} : u = \iota(e)_1 \wedge \iota(e)_2 \in B_j \wedge \kappa(e) = c\}| = |\{e \in \overrightarrow{E} : u' = \iota(e)_1 \wedge \iota(e)_2 \in B_j \wedge \kappa(e) = c\}|$as well as for outgoing edges, formally $\forall c, \forall j : |\{e \in \overrightarrow{E} : u = \iota(e)_2 \wedge \iota(e)_1 \in B_j \wedge \kappa(e) = c\}| = |\{e \in \overrightarrow{E} : u' = \iota(e)_2 \wedge \iota(e)_1 \in B_j \wedge \kappa(e) = c\}|$,
3. within the block B_i they have the same number of incident edges, loops and semi-edges of the same color c, where loops are counted twice, formally $\forall c : |\{e \in E \cup S : u \in \iota(e) \wedge \iota(e) \subseteq B_i \wedge \kappa(e) = c\}| + 2|\{e \in L : u = \iota(e) \wedge \kappa(e) = c\}| = |\{e \in E \cup S : u' \in \iota(e) \wedge \iota(e) \subseteq B_i \wedge \kappa(e) = c\}| + 2|\{e \in L : u' = \iota(e) \wedge \kappa(e) = c\}|$.

The degree partition of incidence graph models is a generalization of the standard degree partition. The partition always exists, the utmost case is again covered by one-vertex blocks. The algorithm also is similar to the previous DEGREEPARTITION, we only need to capture in the vector corresponding to each vertex the number of incident undirected edges, and also the number of in-going and out-going directed edges with the other end-vertex in particular blocks. A pseudocode for the algorithm for finding a degree partition of a graph in the incidence model is given in Algorithm 3.

Input: a graph $G = (V, \Lambda, \iota)$
Initialize: Set the initial partition $\mathcal{B}^0 = \{B_1^0, B_2^0, \dots B_{k_0}^0\}$ of V, k_0 being the number of distinct colors of V, B_i^0 containing vertices of the same color and blocks being ordered by $\preceq$ of the corresponding colors, i.e., $c^{-1}(B_i^0) \preceq c^{-1}(B_j^0)$ whenever $i \leq j$.
Set $i = 0$.
repeat

For each vertex $u \in V_G$ belonging to B_l^i, set a vector $\bar{u}^i$ of length $3k_i$ having on the j-th position:

- $\bar{u}_j^i = |\{e \in E : u \in \iota(e) \wedge \iota(e) \cap B_j^i \neq \emptyset\}|$,
  i.e., the number of undirected edges incident with u and the vertices of block B_j for $j = 1 \dots l-1, l+1 \dots k_0$;
- $\bar{u}_l^i = |\{e \in E \cup S : u \in \iota(e) \wedge \iota(e) \subseteq B_l^i \wedge \kappa(e) = c\}|$
  $+ 2|\{e \in L : u = \iota(e) \wedge \kappa(e) = c\}|$,
  i.e., the undirected degree of vertex u within block B_l^i;
- $\bar{u}_{k_0+j}^i = |\{e \in \overrightarrow{E} : u \in \iota(e) \wedge \iota(e) \cap B_j^i \neq \emptyset \wedge \kappa(e) = c\}|$,
  i.e., the number of outgoing edges of u with end-vertex in block B_j^i;
- $\bar{u}_{2k_0+j}^i = |\{e \in \overrightarrow{E} : u = \iota(e)_2 \wedge \iota(e)_1 \in B_j^i\}|$,
  i.e., the number of incoming edges of u with the initial vertex in block B_j^i.

Subdivide the blocks of the set system $\mathcal{B}^i$ so that only the vertices with identical vectors remain in each of the blocks of the new system $\mathcal{B}^{i+1} = \{B_1^{i+1}, B_2^{i+1}, \dots, B_{3k_i}^{i+1}\}$, i.e., whenever $u, u' \in B_j^{i+1}$, then $\bar{u}^i = \bar{u'}^i$.
Set k_i the number of blocks of $\mathcal{B}^{i+1}$.
Sort $\mathcal{B}^{i+1}$ by the lexicographic ordering of the corresponding vectors $\bar{u}^i$.
Set $i = i + 1$.

until *no subdivision is executed, i.e.,* $\mathcal{B}^{i+1} = \mathcal{B}^i$;
return $\mathcal{B}^i$

Algorithm 3: IncDegreePartition

Generalizations of the degree matrix need to consider the presence of multiple edges with diverse directions, multiplicities, and colors within a block of the degree partition or between two blocks. In the next section, we demonstrate an elegant resolution to this complexity within the dart graph model.

5.3. Partitions in dart graph models

Definition. An *equitable partition* of a graph $G = (D, V, \Lambda, \kappa)$ is a partition $\mathcal{B}$ of the set D into disjoint blocks $B_1, B_2, \ldots, B_k$ such that whenever darts d and d' belong to the same block, then

1. both vertices u and u' such that $d \in u$ and $d' \in u'$ have the same number of incident darts within each block: $\forall i \in \{1, \ldots, k\}\colon |u \cap B_i| = |u' \cap B_i|$, and
2. the links containing darts from the same block connect identical blocks, i.e., for $l, l' \in \Lambda$ such that $d \in l$ and $d' \in l'$ and $\forall i \in \{1, \ldots, k\}\colon l \cap B_i \neq \emptyset \iff l' \cap B_i \neq \emptyset$;
3. they have identical color: $\kappa(d) = \kappa(d')$.

Note that the degree partition on darts is in fact a refinement of the standard degree partition of a graph defined on the set of vertices (the latter emerges from the union of the blocks of darts belonging to a given vertex).

A pseudocode for the algorithm for finding a canonically ordered degree partition of a graph $G = (D, V, \Lambda, \kappa)$ is given in Algorithm 4. As in the preceding section we assume that the set of colors C is linearly ordered by $\preceq$.

Note that the last vector component distinguishes darts of monochromatic links from bicolored ones. Technically, whenever $d \in B^i_j$ it always follows that $j \in \bar{d}^i_{k_i}$. Although removing j from two-element sets could streamline sorting, this optimization might complicate other arguments unnecessarily. An example of getting the equitable partition is depicted in Figure 3.

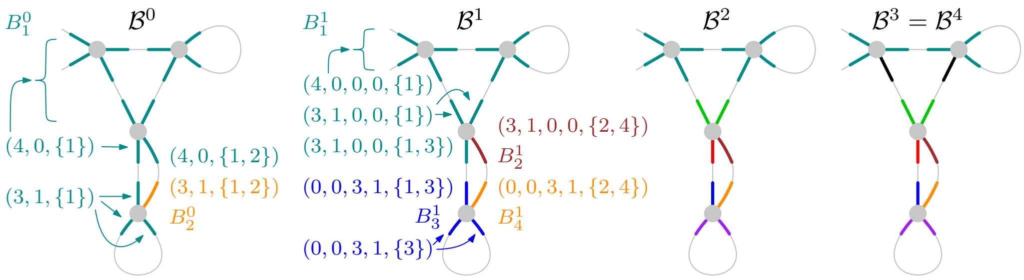


Figure 3. Example of the execution of DARTDEGREEPARTITION. The initial graph is 4-regular, but has darts of two distinct colors corresponding to initial blocks B^0_1 and B^0_2. The final degree partition has eight blocks and is obtained by three refining iterations. For clarity, in $\mathcal{B}^1$, etc. we distinguish the new blocks of the partition by dart colors. In $\mathcal{B}^2$, etc. the vectors are omitted.

Theorem 7. *Let G and H be graphs such that G covers H. Then the covering projection maps each block of the degree partition of G onto a block with the same index in the degree partition of H.*

Input: a graph $G = (D, V, \Lambda, \kappa)$
Initialize:
Set the initial partition $\mathcal{B}^0 = \{B_1^0, B_2^0, \ldots, B_{k_0}^0\}$ of D, k_0 being the number of distinct colors of D, each block B_i^0 containing darts of the same color and blocks being ordered by $\preceq$ of the corresponding colors, i.e., $c^{-1}(B_i^0) \preceq c^{-1}(B_j^0)$ whenever $i \leq j$;
Set $i = 0$.
repeat
For each dart $d \in D$ set a vector $\bar{d^i}$ of length $k_i + 1$ having on the j-th position the number of darts from the jth block belonging to the same vertex as d when $j \leq k_i$, and on the $(k_i + 1)$-st position we put the set of indices of blocks that contain darts of the link $l \ni d$. Formally $\bar{d^i} = (\bar{d_1^i}, \bar{d_2^i}, \ldots \bar{d_{k_i+1}^i})$, where

$$\bar{d_j^i} = \begin{cases} |u \cap B_j^i| & \text{for } u \in V : d \in u \text{ and } 1 \leq j \leq k_i, \\ \{k : B_k^i \cap l \neq \emptyset\} & \text{for } l \in \Lambda : d \in l \text{ and } j = k_i + 1. \end{cases}$$

Subdivide the blocks of the set system $\mathcal{B}^i$ so that only the darts with identical vectors remain in each of the blocks of the new system $\mathcal{B}^{i+1} = \{B_1^{i+1}, B_2^{i+1}, \ldots, B_{k_{i+1}}^{i+1}\}$.
Sort $\mathcal{B}^{i+1}$ by the lexicographic ordering of the corresponding vectors $\bar{d^i}$;
Set $i = i + 1$
until *no subdivision is executed, i.e.,* $\mathcal{B}^{i+1} = \mathcal{B}^i$;
return $\mathcal{B}^i$

Algorithm 4: DartDegreePartition

Proof. Let $f\colon D_G \to D_H$ be a covering and let $B_1, B_2, \ldots, B_k$ be the blocks of the degree partition of G. Let us define $B_i' = f(B_i)$ for $i \in 1, \ldots, k$. We argue that this mapping is well defined, i.e., whenever darts d and d' have the same image, they are in the same block. Let us first prove that $B_1', B_2' \ldots, B_k'$ is a degree partition of H. The correctness of the definition will come as a special case of the proof.

In the following, let δ and δ' be two darts and let d and d' be their preimages (guaranteed by surjectivity of f), i.e., $f(d) = \delta, f(d') = \delta'$. Let u, u' be vertices such that $d \in u, d' \in u'$. When δ, δ' are in the same block B_i', then so are d and d' and we have $|u \cap B_j| = |u' \cap B_j|$. Note that $f(u \cap B_j) = f|_u (u \cap B_j)$ for any $u \in D_G$, and because f is a cover, $f|_u$ is a bijection, and it holds that $|f(u \cap B_j)| = |f(u' \cap B_j)|$ or, equivalently, $|f(u) \cap B_j')| = |f(u') \cap B_j)'|$.

Now consider links e, e' such that $\delta \in e$ and $\delta' \in e'$. Because f is a covering, there exist links l, l' such that $f(l) = e$ and $f(l') = e'$. Furthermore, we know that $d \in l$ and $d' \in l'$. Knowing that d, d' are in the same block, we have $l \cap B_j \neq \emptyset \iff l' \cap B_j \neq \emptyset$. As f is a nonempty mapping, $f(l \cap B_j) \neq \emptyset \iff l \cap B_j \neq \emptyset$ and we are done.

The correctness of the definition of blocks B'_i can be established by setting $\delta = \delta'$. ∎

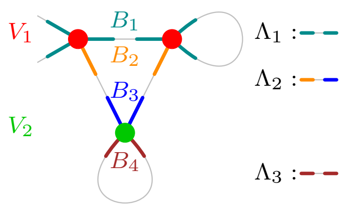
Notice that any equitable partition inherently induces a partition of vertices $V = V_1 \cup \cdots \cup V_r$ and links $\Lambda = \Lambda_1 \cup \cdots \cup \Lambda_s$ akin to the incidence models. When the blocks of an equitable partition are linearly ordered, it implicitly establishes a linear order on both the blocks of vertices and links.

This arrangement permits the definition of its matrix in the following manner. It is essential to note that instead of an adjacency-like matrix typically defined for simple graphs, we employ an analogy of the incidence matrix.

Definition. We say that M is the *matrix of an equitable partition* $B_1, \ldots, B_k$ with induced vertex blocks $V_1, \ldots, V_r$ and link blocks $\Lambda_1, \ldots, \Lambda_s$ if M has k rows and $r + s$ columns, where

$$m_{i,j} = \begin{cases} |B_i \cap u| & \text{for } j \in \{1, \ldots, r\} \text{ and any } u \in V_j, \\ 1 & \text{for } j - r \in \{1, \ldots, s\} \text{ and if } l \cap B_i \neq \emptyset \text{ for some } l \in \Lambda_{j-r}, \\ 0 & \text{for } j - r \in \{1, \ldots, s\} \text{ and if } l \cap B_i = \emptyset \text{ for each } l \in \Lambda_{j-r}. \end{cases}$$

Analogously to Definition 5.1, the matrix associated to canonically ordered degree partition is called the *degree matrix* of a graph.



$$M = \begin{array}{c c} \begin{array}{ccccc} V_1 & V_2 & \Lambda_1 & \Lambda_2 & \Lambda_3 \end{array} & \\ \begin{pmatrix} 3 & 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 1 & 0 \\ 0 & 2 & 0 & 1 & 0 \\ 0 & 2 & 0 & 0 & 1 \end{pmatrix} & \begin{array}{c} B_1 \\ B_2 \\ B_3 \\ B_4 \end{array} \end{array}$$

Figure 4. Example of a degree matrix in the dart model.

An example of a degree matrix is depicted in Figure 4. As a direct consequence of Theorem 7, we assert that an analogous version of Theorem 6 remains valid, even though the dart model employs a different concept of the degree matrix.

Theorem 8. *In the dart model, if a graph G covers a connected graph H, then they have identical degree matrices.*

Proof. First, by Theorem 7, we can assume that the row indices in both degree matrices align with corresponding blocks in the degree partitions of G and H. This implies a one-to-one correspondence between vertex and link blocks, and consequently, between the columns of the two matrices.

Identical values in the columns associated with vertices align with the first condition of Definition 3.3. Similarly, the third condition corresponds to values in the columns related to links. For instance, a value of 1 indicates a dart block incident with a link block, irrespective of whether a single dart connects to a semi-edge or an edge joining two distinct blocks, or if two darts are incident with an edge inside a block. ∎

6. UNIVERSAL AND FINITE COMMON COVERS

The *universal cover* of a graph G is a simple, possibly infinite tree T that allows a covering projection to G. When G is colored, then T is colored accordingly as well.

It is well known that the universal cover T can be built on the elements of the fundamental group of G and that it also could be constructed directly from the relationships between blocks of an equitable partition that are described by its matrix [2, 15, 26, 31].

Leighton proved that isomorphic universal covers of two finite uncolored multigraphs yield the existence of a common *finite* cover.

Theorem 9 [26]**.** *Given any two finite, undirected, and connected multigraphs G and G', the following are equivalent.*

1. *G and G' share a common finite cover,*
2. *G and G' have the same universal cover,*
3. *G and G' have the same degree matrix.*

Having proved implications $1 \Rightarrow 2$ and $2 \Rightarrow 3$, Leighton wrote that "the final implication $3 \Rightarrow 1$ is substantially more difficult." Neumann rephrased Leighton's proof and extended it for the use of colors [31]. We claim that the statement remains valid also when semi-edges are present.

Theorem 10. *In the dart model, two finite connected graphs G and G' have identical degree matrices if and only if they have a finite common cover.*

While a possible workaround would be to first build simple graphs covering G and G' and apply Theorem 9, our objective is to present a direct construction. Despite the resemblance to the proofs by Neumann and Leighton, we include our proof here for both completeness and to exhibit the benefits of employing the dart model in this construction.

Proof. The necessity has been proved as Theorem 8. The sufficiency is proved as follows.

Denote by β_i for $i \in \{1,\dots,k\}$ the size of a (dart) block B_i of the degree partition of the graph G and by $\delta_j, j \in \{1,\dots,r\}$ the number of vertices in the class V_j of the corresponding partition of vertices. Let t be the least common multiple of all numbers $\beta_1,\dots,\beta_k$. As a consequence t is also multiple of each $\delta_1,\dots,\delta_r$.

Denote by $a_i = \frac{t}{\beta_i}$ for $i \in \{1,\dots,k\}$ and by $c_j = \frac{t}{\delta_j}$ for $j \in \{1,\dots,r\}$. Since $\beta_i = m_{i,j}\delta_j$ we get that $a_i m_{i,j} = c_j$. We may routinely derive that values a_i, c_j could be directly derived from the matrix M and hence are independent on further structure of G and G'.

For $i \in \{1,\dots,k\}$ and $j \in \{1,\dots,r\}$ choose arbitrary sets $A_i\colon |A_i| = a_i$, $C_j\colon |C_j| = c_j$, groups $\Pi_{i,j}\colon |\Pi_{i,j}| = m_{i,j}$, and bijections $\varphi_{i,j} : \Pi_{i,j} \times A_i \to C_j$.

For $i, i^* \in \{1,\dots,k\}$ such that $m_{i,j} = m_{i^*,j} = 1$ for some $j \in \{r+1,\dots,r+s\}$ we write $i \sim i^*$ and define a bijection $\sim_{i,i^*}$ between A_i and A_{i^*}. Note that i and i^* need not to be distinct.

In the graph G, choose for each vertex $v \in V_j$ and block $B_i, i \in \{1,\dots,k\}$ a bijection $\psi_{v,i}\colon v \cap B_i \to \Pi_{i,j}$. Obtain $\psi'_{v',i}$ by the same process on the vertices of G'.

The graph H is composed as follows.

$$
\begin{aligned}
D_H &= \Big\{(i,d,d',\alpha)\colon i \in \{1,\dots,k\}, d \in B_i, d' \in B'_i, \alpha \in A_i\Big\},\\
V_H &= \Bigg\{\Big\{(i,d,d',\alpha)\colon i \in \{1,\dots,k\}, d \in v \cap B_i, d' \in v' \cap B'_i,\\
&\qquad \alpha \in A_i, \varphi_{i,j}\Big(\psi_{v,i}(d)\psi_{v',i}(d'),\alpha\Big) = \gamma\Big\}\colon\\
&\qquad\quad j \in \{1,\dots,r\}, v \in V_G, v' \in V_{G'}, \gamma \in C_j\Bigg\},\\
\Lambda_H &= \Big\{\Big\{(i,d,d',\alpha),(i^*,d^*,d'^*,\alpha^*)\Big\}\colon\\
&\qquad\quad i, i^* \in \{1,\dots,k\}, i \sim i^*, \{d,d^*\} \in \Lambda_G, \{d',d'^*\} \in \Lambda_{G'},\\
&\qquad\quad \alpha \in A_i, \alpha^* \in A_{i^*}, \alpha \sim_{i,i^*} \alpha^*\Big\}.
\end{aligned}
$$

An example of the construction is depicted in Figure 5.

Projections $(i,d,d',\alpha) \to d$ and $(i,d,d',\alpha) \to d'$ are the desired covering projections $H \to G$ and $H \to G'$ according to Definition 3.3. Both projections are by the definition dart preserving. Both preserve colors, as the darts d and d' have the same color i as the dart (i,d,d',α). They also map links onto links due to the conditions $\{d,d^*\} \in \Lambda_G, \{d',d'^*\} \in \Lambda_{G'}$ in the construction of Λ_H. Also, as $\sim_{i,i^*}$ is a bijection, every dart is matched with at most one other dart to form a link. Note that an edge can be mapped on a semi-edge e.g. when $d = d^*$ and $d' \neq d'^*$.

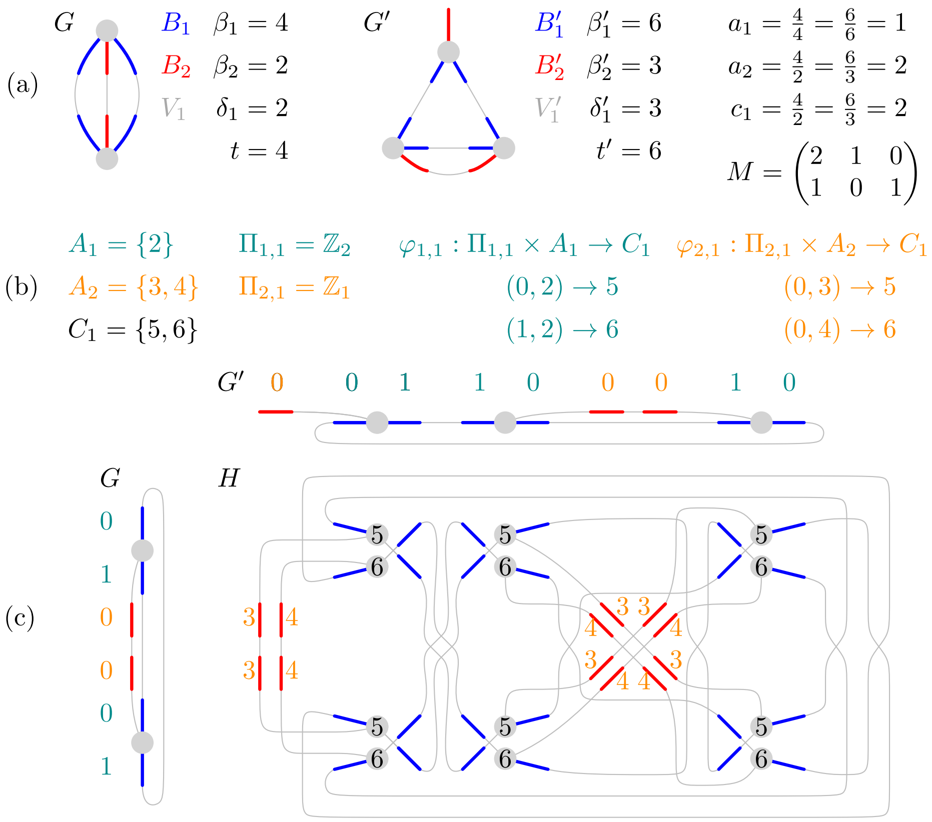


Figure 5. Example of the construction of the common cover H of G and G'. (a) The graphs G and G', their parameters and their common degree matrix. (b) The auxiliary sets and mappings. (c) The color and position of darts of H correspond to i, d and d'. All blue darts have $\alpha = 2$. Red darts have α indicated, as well as vertices their γ. Bijections $\psi_{v,i}$ and $\psi'_{v',i}$ are indicated on darts of G and G'. Both $\sim_{1,1} = \sim_{2,2} = id$.

It remains to argue that both projections are bijective when restricted to the neighborhood of any vertex. Consider without loss of generality the projection $H \to G$ and a vertex u of H determined by $v \in V_G, v' \in V_{G'}, \gamma \in C_j$. Then, since $\varphi_{i,j}$ is a bijection, there is are unique $\tau \in \Pi_{i,j}$ and $\alpha \in A_i$ such that $\varphi_{i,j}(\tau, \alpha) = \gamma$. Now, as $\psi_{v,i}$ and $\psi_{v',i}(d')$ are bijections and $\Pi_{i,j}$ is a group, for any dart $d \in v \cap B_i$ there exists a unique dart $d' \in v' \cap B'_i$ such that $\psi_{v,i}(d)\psi_{v',i}(d') = \tau$, namely $d' = \psi^{-1}_{v',i}((\psi_{v,i}(d))^{-1}\tau)$. Thus d' and α are unique satisfying $\varphi_{i,j}\Big(\psi_{v,i}(d)\psi_{v',i}(d'), \alpha\Big) = \gamma$ for fixed d and γ. After integration along all colors i we get a bijection between darts incident with u and the darts incident with v. For the projection to any vertex of G' we argue analogously. ∎

Moreover, it is worth noting that universal covers correspond one-to-one with degree matrices in the dart model. Consequently, Theorem 10 can be extended to encompass universal covers as well.

7. Conclusion

We have demonstrated that the dart model is suitable and often simpler for various concepts related to graph covers surveyed in this paper. Our generalization includes possible presence of semi-edges, which are related to more general covering concepts such as wrapped quasicoverings [19].

An interesting question arises: from the perspective of graph covers, is the expressive power of graphs with semi-edges strictly stronger than that of simple graphs? For instance, when H is formed by two semi-edges stemming from a single vertex, the simple graphs that cover H are all cycles. Therefore, to preserve the existence of a covering, one must transform H into a graph containing a cycle. Note that classical constructions like Cartesian product with K_2 or taking two copies of a graph and matching corresponding semi-edges into edges do not guarantee that edges obtained from semi-edges are only mapped by a covering onto edges obtained from semi-edges in the target graph.

There are other natural directions for further explorations, e.g. structural relationships of degree matrices suggested in [15] or other kinds of locally constrained homomorphisms [14].

Moreover, our model could be refined to graph embeddings where the vertex embedding is modelled combinatorially with help of permutations [29]. Cycles of the permutation yield the equivalence classes of the underlying graph, and the particular order along a cycle yields the rotational scheme of links stemming from a vertex. It is likely that our approach and results could be extended to this situation.

Acknowledgements

We extend our gratitude to Jan Kratochvíl and Roman Nedela for their invaluable discussions and continuous encouragement throughout this project. We thank anonymous referees for suggestions improving our paper. We also utilized ChatGPT ver 3.5 to refine several explanatory sentences during the review process. Both authors were supported by research grant GAČR 20-15576S of the Czech Science Foundation. M. Seifrtová was also supported by the grant SVV-2023-260699 of GAUK.

References

[1] J. Abello, M.R. Fellows and J.C. Stillwell, *On the complexity and combinatorics of covering finite complexes*, Australas. J. Combin. **4** (1991) 103–112.

[2] D. Angluin, *Local and global properties in networks of processors*, in: Proc. 12th ACM Symposium on Theory of Computing (1980) 82–93.
https://doi.org/10.1145/800141.804655

[3] D. Angluin and A. Gardiner, *Finite common coverings of pairs of regular graphs*, J. Combin. Theory Ser. B **30** (1981) 184–187.
https://doi.org/10.1016/0095-8956(81)90062-9

[4] N. Biggs, Algebraic Graph Theory (Cambridge University Press, 1974).
https://doi.org/10.1017/CBO9780511608704

[5] H.L. Bodlaender, *The classification of coverings of processor networks*, J. Parallel Distrib. Comput. **6** (1989) 166–182.
https://doi.org/10.1016/0743-7315(89)90048-8

[6] J. Bok, J. Fiala, P. Hliněný, N. Jedličková and J. Kratochvíl, *Computational complexity of covering multigraphs with semi-edges: Small cases*, in: 46th International Symposium on Mathematical Foundations of Computer Science, LIPIcs, Leibniz Int. Proc. Inform. **202**, F. Bonchi and S.J. Puglisi (Ed(s)), (Dagstuhl Publishing, Germany, 2021) 21:1–21:15.
https://doi.org/10.4230/LIPIcs.MFCS.2021.21

[7] P. Boldi and S. Vigna, *Fibrations of graphs*, Discrete Math. **243** (2002) 21–66.
https://doi.org/10.1016/S0012-365X(00)00455-6

[8] D.G. Corneil, Graph Isomorphism, PhD Thesis (University of Toronto, 1968).

[9] D.G. Corneil and C.C. Gotlieb, *An efficient algorithm for graph isomorphism*, J. ACM **17** (1970) 51–64.
https://doi.org/10.1145/321556.321562

[10] D.M. Cvetković, M. Doob and H. Sachs, Spectra of Graphs, Theory and Applications, Revised Version (Huthing Pub. Ltd, 1995).

[11] J.R. Edmonds, A Combinatorial Representation for Oriented Polyhedral Surfaces, PhD Thesis (University of Maryland, 1960).
https://doi.org/10.13016/daw5-mvla

[12] J. Fiala, P. Klavík, J. Kratochvíl and R. Nedela, *Algorithmic aspects of regular graph covers with applications to planar graphs*, in: Automata, Languages, and Programming, Lecture Notes in Comput. Sci. **8572**, J. Esparza, P. Fraigniaud, T. Husfeldt and E. Koutsoupias (Ed(s)), (Springer, Berlin, Heidelberg, 2014) 489–501.
https://doi.org/10.1007/978-3-662-43948-7_41

[13] J. Fiala and J. Kratochvíl, *Partial covers of graphs*, Discuss. Math. Graph Theory **22** (2002) 89–99.
https://doi.org/10.7151/dmgt.1159

[14] J. Fiala and J. Kratochvíl, *Locally constrained graph homomorphisms—structure complexity, and applications*, Comput. Sci. Rev. **2** (2008) 97–111.
https://doi.org/10.1016/j.cosrev.2008.06.001

[15] J. Fiala, D. Paulusma and J.A. Telle, *Locally constrained graph homomorphisms and equitable partitions*, European J. Combin. **29** (2008) 850–880. https://doi.org/10.1016/j.ejc.2007.11.006

[16] J.L. Gross and T.W. Tucker, *Generating all graph coverings by permutation voltage assignments*, Discrete Math. **18** (1977) 273–283. https://doi.org/10.1016/0012-365X(77)90131-5

[17] J.L. Gross and T.W. Tucker, Topological Graph Theory (Dover Publications, INC, Mineola, New York, 1987).

[18] A. Grothendieck, *Technique de descente et théorèmes d'existence en géométrie algébrique.* I: *Généralités. Descente par morphismes fidèlement plats*, Séminaire N. Bourbaki **5** (1960) 299–327. http://www.numdam.org/item?id=SB_1958-1960__5__299_0

[19] B. Jackson, T.D. Parsons and T. Pisanski, *A duality theorem for graph embeddings*, J. Graph Theory **5** (1981) 55–77. https://doi.org/10.1002/jgt.3190050104

[20] G.A. Jones and D. Singerman, *Theory of maps on orientable surfaces*, Proc. Lond. Math. Soc. **37(3)** (1978) 273–307. https://doi.org/10.1112/plms/s3-37.2.273

[21] D. König, Theorie der endlichen und unendlichen Graphen: Kombinatorische Topologie der Streckenkomplexe (Akademische Verlagsgesellschaft, Leipzig, 1936).

[22] J. Kratochvíl, A. Proskurowski and J.A. Telle, *Complexity of colored graph covers* I. *Colored directed multigraphs*, in: Graph-Theoretic Concepts in Computer Science, Proc. 23 WG'97, Lecture Notes in Comput. Sci. **1335**, (Springer, Berlin Heidelberg, 1997) 242–257. https://doi.org/10.1007/BFb0024502

[23] J. Kratochvíl, A. Proskurowski and J.A. Telle, Covering directed multigraphs II. When 2-SAT helps (Tech. Rep. 1997–354 KAM-DIMATIA Preprint Series, 1997).

[24] J. Kratochvíl, A. Proskurowski and J.A. Telle, *Covering regular graphs*, J. Combin. Theory Ser. B **71** (1997) 1–16. https://doi.org/10.1006/jctb.1996.1743

[25] J. Kratochvíl, A. Proskurowski and J.A. Telle, *Complexity of graph covering problems*, Nordic J. Comput. **5** (1998) 173–195.

[26] F.T. Leighton, *Finite common coverings of graphs*, J. Combin. Theory Ser. B **33** (1982) 231–238. https://doi.org/10.1016/0095-8956(82)90042-9

[27] A. Malnič, R. Nedela and M. Škoviera, *Lifting graph automorphisms by voltage assignments*, European J. Combin. **21** (2000) 927–947. https://doi.org/10.1006/eujc.2000.0390

[28] A.D. Mednykh and R. Nedela, Harmonic Morphisms of Graphs: Part I: Graph Coverings (Vydavatelstvo Univerzity Mateja Bela v Banskej Bystrici, 2015).

[29] R. Nedela and M. Škoviera, *Regular embeddings of canonical double coverings of graphs*, J. Combin. Theory Ser. B **67** (1996) 249–277.
https://doi.org/10.1006/jctb.1996.0044

[30] S. Negami, *Graphs which have no planar covering*, Bull. Inst. Math. Acad. Sin. **16** (1988) 377–384.

[31] W.D. Neumann, *On Leighton's graph covering theorem*, Groups Geom. Dyn. **4** (2010) 863–872.
https://doi.org/10.4171/ggd/111

[32] K. Reidemeister, *Einführung in die kombinatorische Topologie*, Monats. Math. **40** (1933) A9.
https://doi.org/10.1007/BF01708901

[33] H. Sachs, *Simultane Überlagerung gegebener Graphen*, Publ. Math. Inst. Hung. Acad. Sci. Ser. A **9** (1965) 415–427.

Received 28 February 2023
Revised 25 January 2024
Accepted 26 January 2024
Available online 13 February 2024