Graph Relations and
Constrained Homomorphism Partial Orders

Von der Fakultät für Mathematik und Informatik
der Universität Leipzig
angenommene

D I S S E R T A T I O N

zur Erlangung des akademischen Grades

DOCTOR RERUM NATURALIUM
(Dr.rer.nat.)

im Fachgebiet

Mathematik

vorgelegt

von Bachelorin Yangjing Long
geboren am 08.07.1985 in Hubei (China)

Die Annahme der Dissertation wurde empfohlen von:

1. Professor Dr. Pavol Hell (Burnaby, Kanada)
2. Professor Dr. Jürgen Jost (Leipzig)

Die Verleihung des akademischen Grades erfolgt mit Bestehen der Verteidigung am
14.10.2014 mit dem Gesamtprädikat magna cum laude.

献给我挚爱的母亲们！

Acknowledgements

Abstract

We consider constrained variants of graph homomorphisms such as embeddings, monomorphisms, full homomorphisms, surjective homomorpshims, and locally constrained homomorphisms. We also introduce a new variation on this theme which derives from relations between graphs and is related to multihomomorphisms. This gives a generalization of surjective homomorphisms and naturally leads to notions of R-retractions, R-cores, and R-cocores of graphs. Both R-cores and R-cocores of graphs are unique up to isomorphism and can be computed in polynomial time.

The theory of the graph homomorphism order is well developed, and from it we consider analogous notions defined for orders induced by constrained homomorphisms. We identify corresponding cores, prove or disprove universality, characterize gaps and dualities. We give a new and significantly easier proof of the universality of the homomorphism order by showing that even the class of oriented cycles is universal. We provide a systematic approach to simplify the proofs of several earlier results in this area. We explore in greater detail locally injective homomorphisms on connected graphs, characterize gaps and show universality. We also prove that for every $d \geq 3$ the homomorphism order on the class of line graphs of graphs with maximum degree d is universal.

Contents

List of Figures

Chapter 1

Introduction

The subject of this thesis—graph relations and constrained homomorphism orders—belongs to the field of discrete mathematics, more specifically to graph theory. It has a close relationship to several other areas of mathematics stemming from algebra and mathematical structures via category theory and logic to computer science disciplines such as computational complexity.

The term 'graph' was introduced by Sylvester in 1878 [79]. A *directed graph* G is a pair $G = (V_G, E_G)$ such that E_G is a subset of $V_G \times V_G$. We denote by V_G the set of *vertices* of G and by E_G the set of *edges* of G. The class of all finite directed graphs is denoted by DiGraphs. An *undirected graph* (or simply a *graph*) G is a directed graph such that $(u, v) \in E_G$ if and only if $(v, u) \in E_G$. We thus consider undirected graphs to be a special case of directed graphs. Unless explicitly stated otherwise we allow loops on vertices. The class of all finite undirected graphs is denoted by Graphs.

Graphs are frequently employed to model natural or artificial systems. Let us begin the introduction with two examples taken from the real world.

In social science we can use social networks to express the relationships between individuals, groups and organizations. For example, a friendship network shows the friendship relation among individuals in a community. At the same time, travelling is an important activity in modern society as people travel frequently, for business or for pleasure. If we consider all the places that people in a certain community have ever visited, we can define a network on places in the following natural way. Draw an edge between place A and place B if and only if we can find two people who are friends, one who has visited A, and the other B. In this

way we build a new network on places generated by the friendship network and the information about people and the places they visited.

The research presented in this thesis was originally motivated by a generalization of concepts from bioinformatics. Consider protein-protein interaction networks (PPIs) in biology. Proteins contain domains that determine their function. Information on the domain content of proteins, which can be seen as a relation $R \subseteq D \times P$ between domains and the proteins which contain them is readily available from biological databases. Candidates for interaction between domains can be derived from knowledge about the interacting proteins, i.e., a graph G with vertex set P, and the relation R: If $(p, p') \in E_G$, $(d, p) \in R$, and $(d', p') \in R$, then the domains (d, d') are putative interaction partners.

In both examples, we use a network and a relation to create a new network. Conversely, given two networks, can we determine some meaningful relation between them? This is the idea behind the graph relations.

More abstractly, let $G = (V_G, E_G)$ be a graph, B a finite set, and $R \subseteq V_G \times B$ a binary relation such that for every $b \in B$ there is $v \in V_G$ such that $(v, b) \in R$. Then the graph $G * R$ has vertex set B and edge set

$$E_{G*R} = \{(a, b) \in B \times B \mid \text{ there is } (u, v) \in E_G \text{ such that } (u, a), (v, b) \in R\} .$$

If there exists an R satisfying the equation $G * R = H$, we say there is a *relation* from G to H. The graphs G and H can be seen to be conjugates, i.e., $E_G = R^+ \circ E_H \circ R$, where $R^+ \subseteq B \times V_G$ is the transpose of the relation R and $\circ$ denotes the *composition* of binary relations, which is defined as follows: If $S \subseteq X \times Y$ and $T \subseteq Y \times Z$ are two binary relations, then $S \circ T = \{(x, z) \in X \times Z \mid \text{ there is } y \in Y \text{ such that } (x, y) \in S \text{ and } (y, z) \in T\}$.

The concept of a relation between graphs turns out be closely related to the well-studied graph theory notion of a graph homomorphism. For given graphs $G = (V_G, E_G)$ and $H = (V_H, E_H)$, a *homomorphism* $f : G \to H$ is a mapping $f : V_G \to V_H$ such that $(u, v) \in E_G$ implies $(f(u), f(v)) \in E_H$. If there is no homomorphism from G to H, we write $G \nrightarrow H$. Graph homomorphisms have been studied ever since the 1960s, originally as a generalization of graph colourings, with pioneering work by G. Sabidussi [74], Z. Hedrlín and A. Pultr [35] subsequently seeing rapid development by P. Hell and J. Nešetřil [40]. There is a lot of intensive research going on around this topic; for an extensive review of the subject, see [40].

A homomorphism from a graph G to a graph H naturally defines a mapping $f^1 : E_G \to E_H$ by setting $f^1((u, v)) = (f(u), f(v))$ for all $(u, v) \in E_G$. If both f and f^1 are surjective, we call f a *surjective homomorphism*. Note that all surjective homomorphisms $f : G \to H$ are (modulo representation) also relations $G * R = H$, where $R = \{(u, f(u)) \mid u \in V_G\}$.

Obviously, relations that are not functional are not homomorphisms, but correspond to multihomomorphisms, which in a sense are multifunctions. A

multifunction from a non-empty set X to a non-empty set Y is a function from X to $P(Y) \setminus \emptyset$ where $P(Y)$ is the power set of Y [57]. Multifunctions between two topological spaces have been studied since 1960s [88]. Multifunctions were first studied in an algebraic sense in [81], in which they feature as multihomomorphisms from one group to another group. To the best of our knowledge, multihomomorphisms between graphs first appeared as building blocks of Hom-complexes, introduced by Lovász, and are related to recent exciting developments in topological combinatorics [56], in particular to deep results involved in a proof of the Lovász hypothesis [2].

A relation with full domain thus can be regarded as a *surjective multihomomorphism*, a multihomomorphism such that the pre-image of every vertex in H is non-empty and for every edge (x, y) in H we can find an edge (u, v) in G satisfying $x \in f(u)$, $y \in f(v)$.

It appears that surjective multihomomorphisms between groups were first defined in [81] and studied in [57]. However, surjective multihomomorphisms between graphs have not yet received attention.

In this work we show that even though the notion of relations has not been noticed by graph theorists it has its own charm. Owing to the omission of the constraint of being a mapping, relations substantially generalize the sort of situation that can hold for graph homomorphisms; a relation can ignore some part of the original graph and/or map one vertex to multiple vertices. Even relations with full domains are a generalization of surjective homomorphisms. Every relation can be decomposed in a standard way into a surjective homomorphism and an injective relation, whose transpose can be seen as a full homomorphism. It produces many interesting notions of graph classes, such as R-cores, cores, reduced graphs and cocores. We show that core and reduced graph coincide, develop the characterization of cocore and R-core, and provide effective algorithms to compute them. We also give the relations of containment among these notions and give examples which can distinguish them. We give a perhaps surprisingly simple characterization of those graphs G for which all relations of G to itself are automorphisms. For the computational complexity of testing for the existence of a relation, we describe a reduction of this problem to the corresponding problem for surjective homomorphisms.

Inspired by relations between graphs, we widen the scope of our study to investigate the properties of *constrained homomorphisms*, i.e., homomorphisms that satisfy further conditions such as surjectivity. Partial orders induced by the existence of constrained homomorphisms and their properties form the second topic of this thesis. In particular we consider the following:

1. A homomorphism $f : G \to H$ is a *monomorphism*[1], (also called *injective homomorphism* [40] or *M-morphism*) if it is injective.

2. A homomorphism $f : G \to H$ is a *full homomorphism*, or *F-morphism*, if $(f(u), f(v)) \in E_H$ implies $(u, v) \in E_G$.

3. A homomorphism $f : G \to H$ is an *embedding*, or *E-morphism* if it is both a monomorphism and a full homomorphism.

For undirected graphs we also consider the following forms of locally constrained homomorphisms.

1. A homomorphism $f : G \to H$ is a *locally injective homomorphism*, or *LI-morphism*, if for all $v \in V_G$, the restriction of the mapping f to the domain $N_G(v)$ and range $N_H(f(v))$ is injective.

2. A homomorphism $f : G \to H$ is a *locally surjective homomorphism*, or *LS-morphism*, if for all $v \in V_G$, the restriction of the mapping f to the domain $N_G(v)$ and range $N_H(f(v))$ is surjective.

3. A homomorphism $f : G \to H$ is a *locally bijective homomorphism*, or *LB-morphism*, if it is both a locally surjective and locally bijective homomorphism, that is, if the restrictions to the neighbourhoods are bijections.

There are also other types of constrained homomorphism that have been studied. One example is that of a compaction [48]. A *compaction* of a graph G to a graph H is a homomorphism $f : G \to H$ such that for every vertex x of H, there exists a vertex v of G with $f(v) = x$, and for every edge (x, y) of H, $x \neq y$, there exists an edge (u, v) of G with $f(u) = x$ and $f(v) = y$. The computational complexity of compaction has been studied by Vikas [84, 85].

Constrained homomorphisms have been intensively studied in their structural and algorithmic aspects as well as in their applications. Each type of constrained homomorphism has its own history. For example, globally constrained homomorphisms have also been studied from the 1960s. The earliest literature we can find is what Hedetniemi published in 1966 [33]. It is however surprising to learn from the survey paper [22] that the notion of a locally bijective homomorphism was already well established in combinatorial topology as early as 1932 [72]. The other types of locally constrained homomorphism received attention much later. Locally surjective homomorphisms were introduced by Everett and Borgatti in 1991 [18] and locally injective homomorphisms by Nešetřil in 1971 [63].

---

[1]The name 'monomorphism' originates from category theory.

There are many results that link graphs and orders, such as those concerning vertex-edge-face partial orders [9], planar graphs, hypergraph colourings [16], and on-line algorithms [50]. In a similar vein, the dimension of a partial order is analogous to the chromatic number of a graph [49]. See the survey paper [82] by Trotter. We give another instance of partial orders on graphs.

For any fixed type of finite relational structure, homomorphisms induce an ordering of the set of all structures. In particular, given two (directed) graphs G and H, we write $G \leq H$, or $G \to H$, if there is a homomorphism from G to H. The relation $\leq$ is a quasi-order and so it induces an equivalence relation. For two (directed) graphs G and H such that $G \to H$ and $H \to G$, we write $G \sim H$ and say that *G and H are homomorphically equivalent*. The homomorphism orders (Graphs, $\leq$) and (DiGraphs, $\leq$) are the partially ordered sets of all equivalence classes of finite undirected and directed graphs, respectively, ordered by the relation $\leq$. We shall abuse notation and follow the convenient practice of identifying an equivalence class in either of these homomorphism quasi-orders with one of its representatives, namely that (di)graph which is the core of all other (di)graphs equivalent to it. So by (Graphs, $\leq$), respectively (DiGraphs, $\leq$), we will actually be referring to the partially ordered sets induced by the quasi-ordered sets of all graphs (digraphs) on the set of graph (digraph) cores. (The notion of a graph core requires some preparation and will be defined in its proper place in Chapter 3.)

These orders are of particular interest. For example, each of the two orders forms a distributive lattice, with the disjoint union of graphs being the supremum and the categorical product being the infimum.

The homomorphism order has been a fruitful research topic for several decades. One of the well studied properties of the homomorphism order is density. Both (Graphs, $\leq$) and (DiGraphs, $\leq$) are dense, the former shown by Welzl [87] and the latter by Nešetřil and Tardif [64].

The other property which is extensively studied is universality. Any countable partial order is isomorphic to a suborder of (DiGraphs, $\leq$). This result first appeared in the context of categories in 1969 [71]. *Antichains* are sets of elements which are pairwise incomparable. A characterization of finite maximal antichains was given in [59].

Maximal antichains are particularly relevant because of their relationship to the notion of a *homomorphism duality*, introduced by Nešetřil and Pultr [59]. An ordered pair (F, D) of graphs, or directed graphs, is a *duality pair* if $\{G \mid F \to G\} = \{G \mid G \nrightarrow D\}$. Moreover, Nešetřil and Tardif obtain a correspondence between duality pairs and gaps in the homomorphism order [60].

In the second part of this work, we consider properties of constrained homomorphism orders analogous to those for the homomorphism order. We provide a systematic method for characterizing the gaps and dualities in constrained homomorphism orders by analysing properties of future-finite or past-finite orders, which has simplified the proofs in the full homomorphism order. We also give a method of building universal orders from future-finite or past-finite universal orders. Among other things, by realizing that oriented cycles are a new universal class in the homomorphism order, we prove universality of the homomorphism order on the class of line graphs and of locally injective homomorphism order on connected graphs.

This thesis consists of five chapters, including this introduction.

In Chapter 2 we describe notions that we require from graph theory and order theory. In particular, we introduce basic graph-theoretic concepts and state their properties, and define the main operations on graphs. We also give the fundamental properties of partial orders. Chapter 3 gives a brief overview of graph homomorphisms and constrained homomorphisms. Moreover, we introduce the new notion of graph relation, and we compare it to the other well-studied types of constrained graph homomorphism. We provide the necessary background on the theory of computational complexity in Appendix A.

Further properties of relations are introduced in Chapter 4. We first show that every relation can be decomposed in a standard way into a surjective and an injective relation (Corollary 4.1.3). The equivalence classes of strong relational equivalence (Theorem 4.2.4) and weak relational equivalence (Proposition 4.2.9) are characterized. We then introduce the notion of R-cores, cores, reduced graphs and cocores. We show that core and reduced graph coincide, develop the characterization of cocore and R-core, and provide an effective algorithm to compute them. We also give a perhaps surprisingly simple characterization of those graphs G for which all relations of G to itself are automorphisms (Theorem 4.3.7). The computational complexity of testing for the existence of a relation between two graphs is briefly discussed. We describe the reduction of this problem to the corresponding problem for surjective homomorphisms. Finally, we briefly summarize the most important similarities and differences between weak and strong relational composition (Section 4.6). Some of the results from this chapter have been published by J. Hubička, J. Jost, Y. Long, P. F. Stadler, and L. Yang in [44].

In Chapter 5 we review existing results on the homomorphism order (in Section 5.3) and introduce the notion of constrained homomorphism orders. Moreover, after some discussion of universality, a new and significantly easier proof of universality of graph homomorphisms is given (Theorem 5.3.5), which simplifies the proof given in [43]. As a new result, we show that graph homomorphisms

are universal even on the class of oriented cycles. The notions of future-finite-universal and past-finite-universal partial order are introduced (in Section 5.1) for later use. We survey existing results on constrained homomorphism orders and provide several new results about cores, universality, gaps and dualities in the context of constrained homomorphisms. We derive properties of these partial orders and identify their similarities and differences. At the same time we give a simple condition for the existence of dualities in future-finite and past-finite partial orders (in Section 5.2). In the full homomorphism order (Section 5.5) we characterize the gaps and give a new and simple proof of the existence of left duals (Theorem 5.5.7) (simplifying results independently obtained by Feder and Hell [19] and Ball, Nešetřil, Pultr [3, 4]). For locally injective homomorphisms we prove universality on connected graphs (Theorem 5.7.14) and give a partial characterization of gaps (Theorem 5.7.13). We also prove the universality of all three kinds of locally constrained homomorphisms on graphs. We apply the characterization of PR-cores and FR-cores which are derived in Chapter 4 and carry over most results on the surjective homomorphism order. We prove that for every $d \geq 3$ the homomorphism order on the class of line graphs with maximum degree d is universal (Theorem 5.8.3). Some of the results from this chapter have been submitted [27] or are in preparation [28].

Chapter 2

Graphs and Orders

2.1 Graphs

Graphs, as mathematical structures used to model pairwise relations between objects from a certain collection, are the main theme of this thesis. Graphs are among the most ubiquitous models of both natural and human-made structures. They can be used to model many types of relations and process dynamics in physical, biological, social and information systems. Many problems of practical interest can be represented by graphs.

Basic concepts

We introduce several variants of the notion of 'graph'. Our situation is the same as Hell and Nešetřil described in [40]: "One challenge we face is the intermingling of the various versions of graphs, directed graphs, and more general systems. It is typical of the area to freely jump from graphs to directed graphs, allowing or disallowing loops, as it dictated by the context." The main purpose of this part is to cover all the various versions of graphs used in this thesis.

Definition 2.1.1. A *directed graph* (or *digraph*) G is a pair $G = (V_G, E_G)$ such that E_G is a subset of $V_G \times V_G$. Set V_G is called the *vertex set of G*, and each element V_G is a *vertex* (or *point*, *node*) of graph G. Set E_G is called the *edge set of G*, and every element in E_G is an *edge* (or *arc*) of the graph G.

Directed graphs are visualized by representing vertices as points and edges as arrows. There are some examples of directed graph shown in Figure 2.1.
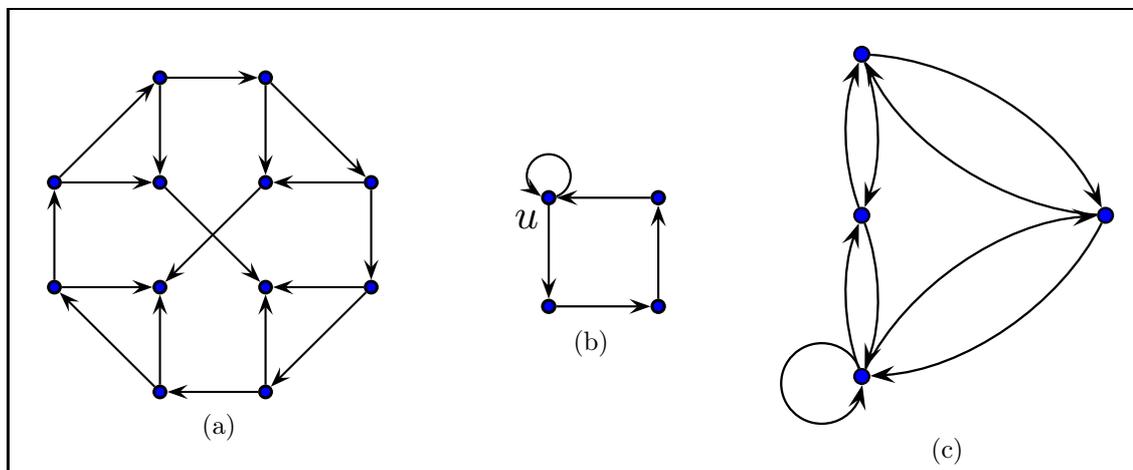
Figure 2.1: Directed graphs.

Definition 2.1.2. An *undirected graph* (or simply a *graph*), is a graph with symmetric edge set, i.e., a directed graph such that (u, v) is an edge if and only if (v, u) is an edge.

The graph shown in Figure 2.1c, for instance, is an undirected graph. On the one hand, we can consider an undirected graph to be a special case of a directed graph. On the other hand, each directed graph has a unique *underlying graph*, that is, the directed graph with the direction on edges omitted. Figure 2.2 shows an example of the underlying graph of a directed graph.

Note that undirected graphs and directed graphs can be mutually constructed. We have seen that we can create an undirected graph from a directed graph. We can also orient the edges of undirected graph to be directed. An *orientation* of an undirected graph is an assignment of a direction to each edge, making it into a directed graph. A directed graph is called an *oriented graph* if it is an orientation of an undirected graph. In other words, an oriented graph is a directed graph having no symmetric pair of directed edges.

An edge that connects a vertex to itself is called a *loop*. For instance, the graph in Figure 2.1b has a loop on vertex u. In graph theory, sometimes we only consider *simple graphs*[1], that is, undirected graphs without loops. But in this thesis, for the sake of convenience, we use the term *graph* for an undirected graph with loops allowed.

---

[1]In the literature it is often the case that the term graph refers to a simple graph.
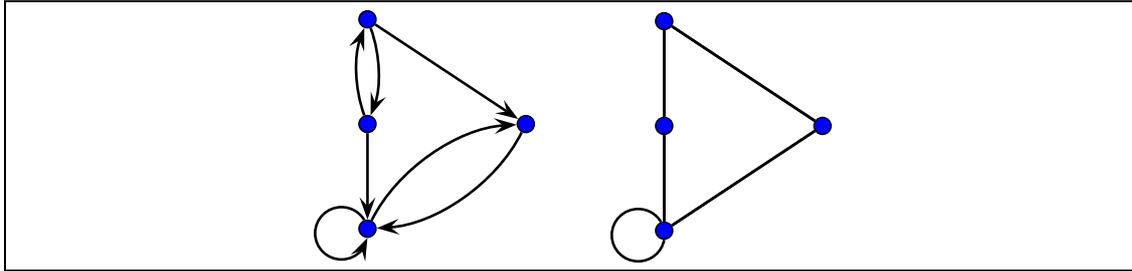
Figure 2.2: A directed graph and its underlying graph.

Sometimes we consider multiple edges between a single pair of vertices. A *multigraph* G is an ordered pair $G := (V, E)$ with V as the vertex set and E, a multiset of unordered pairs of vertices, as the edge set. Figure 2.3 shows a simple graph, a graph, a directed graph, an oriented graph, and a multigraph.

A directed graph is a *weighted directed graph* if a number (weight) is assigned to each edge. Such weights might represent, for example, costs, lengths or capacities, etc. depending on the problem at hand. Some authors call such a graph a *network* [77].
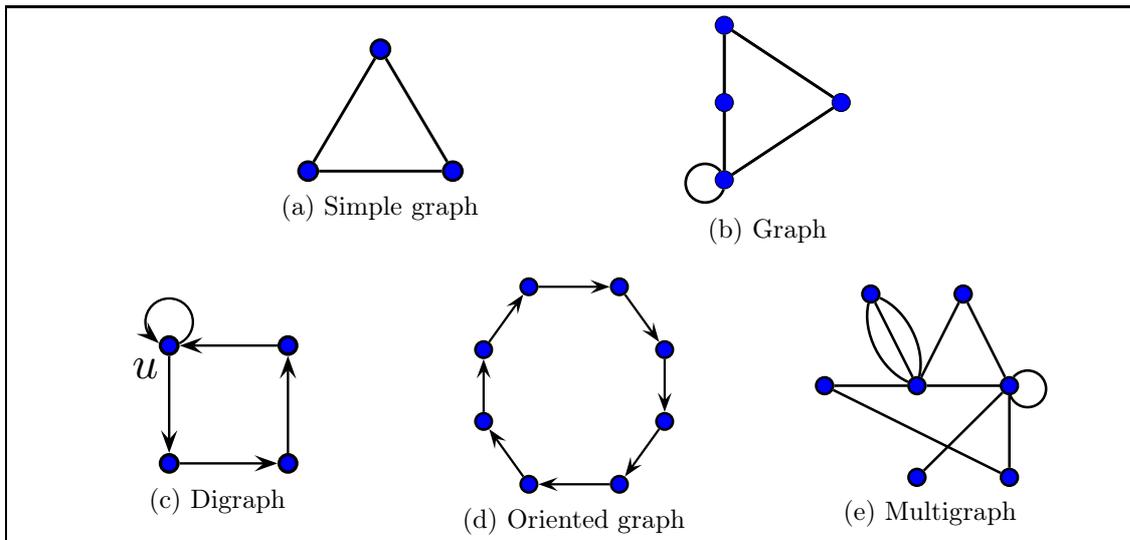
Figure 2.3: Different classes of graphs.

The *size of a graph* G, denoted by $|G|$, is the number of its vertices. It is possible to consider an *infinite graph*, that is, a graph with infinitely many vertices. However, in this thesis we consider only *finite graphs*, that is, graphs of finite size. There are different classes of graphs. In this thesis, if not explicitly stated, by a graph we mean a finite simple graph possibly with a loop on some vertices.

Adjacency and neighbourhoods

Let $G = (V_G, E_G)$ be a graph, x and y be vertices in graph G. If $(x, y) \in E_G$, then we say vertex x and vertex y are *adjacent*. The *degree* or *valency* of a vertex v in a graph G, denoted by $d(v)$, is the number of vertices which is adjacent to vertex v. An *isolated vertex* is a vertex with degree 0. Note that a simple loop is not an isolated vertex in this case. For example, in Figure 2.4, vertex i is an isolated vertex, vertex o is an isolated loop, but not an isolated vertex.
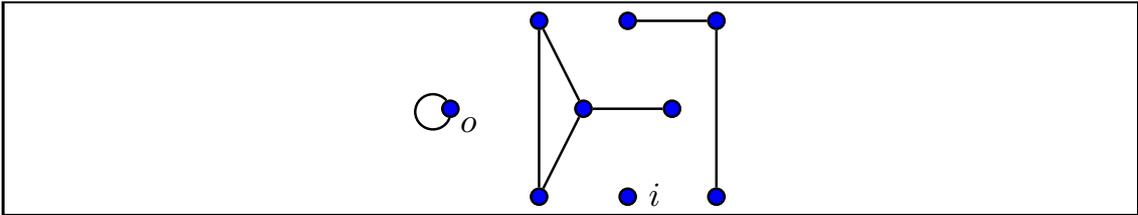
Figure 2.4: Disconnected graph with an isolated vertex and a loop.

Let G be a graph, and x be a vertex of graph G.

Definition 2.1.3. The *(open) neighbourhood* of x in G, denoted by $N_G(x)$, is the set of vertices adjacent to a vertex $x \in G$. The *closed neighbourhood* of x in G, denoted by $N_G[x]$, is the open neighbourhood and x, i.e., $N_G[x] = N_G(x) \cup \{x\}$.

For the sake of brevity we use $N(x)$ and $N[x]$ if it is clear which graph we refer to. As an example, in Figure 2.4 $N(o) = o, N[o] = o, N(i) = \emptyset, N[i] = i$.

Subgraphs

We say that $G' = (V', E')$ is a *subgraph* of $G = (V, E)$, denoted by $G' \subseteq G$, if $V' \subseteq V$ and $E' \subseteq E$. If G' contains all edges of G which join two vertices in V', then G' is an *induced subgraph* of G. Graph G' is said to be the subgraph *induced by V'* and denoted by $G[V']$. Induced subgraphs form a special class of subgraphs. Figure 2.5 shows the difference between subgraphs and induced subgraphs.

(a) Subgraph   (b) Induced subgraph
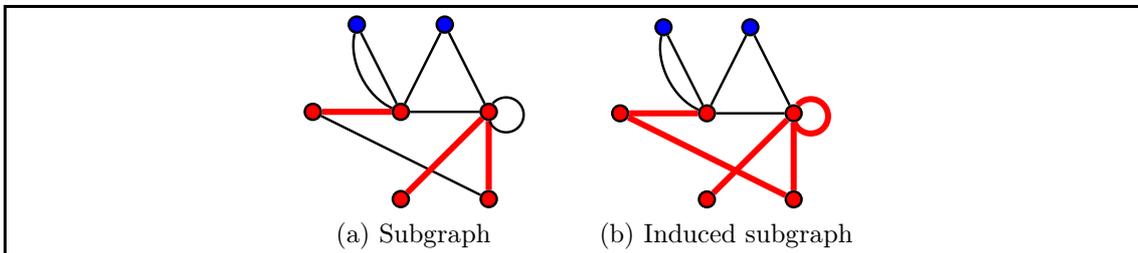
Figure 2.5: Subgraph and induced subgraph.

DIAMETER AND CONNECTIVITY

A *walk* in a graph G is a sequence of vertices $v_0, v_1, \ldots, v_k$ of G such that $(v_{i-1}, v_i) \in E_G$ for each $i = 1, 2, \ldots, k$. A *path* is a non-repeated walk, that is the vertices in a walk are distinct. A *cycle* is a walk $v_0, v_1, \ldots, v_k$ with $k \geq 2$ such that $v_0 = v_k$ and the vertices v_i for $0 < i < k$ are distinct from each other and x_0.
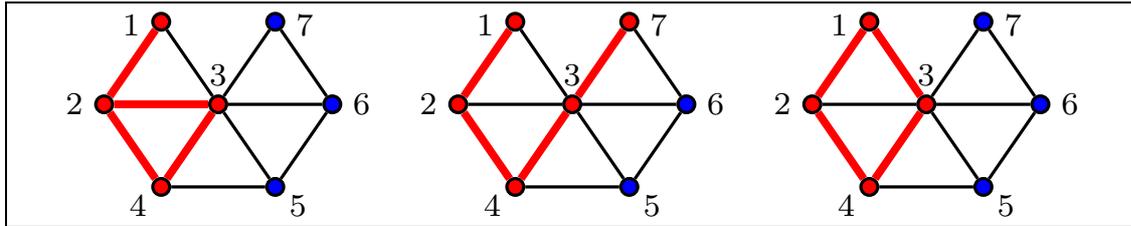
Figure 2.6: An example of walk, path and cycle.

Example 2.1.1. In Figure 2.6, sequence 1, 2, 4, 3, 2 is a walk (but not a path nor a cycle), sequence 1, 2, 4, 3, 7 is a path, and sequence 1, 2, 4, 3, 1 is a cycle.

The *distance* between two vertices u, v in a graph, denoted by $d_G(u, v)$, is the number of edges in a shortest path connecting them; if there is no path connects vertices u and v, then the distance is infinite. The *eccentricity* ϵ of a vertex v is the greatest distance between v and any other vertex. The *radius* of a graph G, denoted by $rad(G)$, is the minimum eccentricity of any vertex. The *diameter* of a graph G, denoted by $diam(G)$, is the maximum eccentricity of any vertex in the graph, that is, the greatest distance between any pair of vertices.

Graph connectivity is in the sense of topological space. A graph is *connected* if there is a path from any vertex to any other vertex in the graph. A *connected component* of an undirected graph is a subgraph in which any two vertices are connected to each other by paths, and which is connected to no additional vertices in the subgraph. For example, the graph shown in the Figure 2.4 has three connected components: an isolated vertex, a loop and a connected graph.

Graph constructions

When we need a new graph, the most convenient way is to construct it from old graphs. There are many methods of construction, for instance, building subgraphs and induced subgraphs. In this subsection we introduce some common graph construction methods.

COMPLEMENT GRAPHS

The *complement* (or *inverse*) of a simple graph G, denoted by $\overline{G}$, is a simple graph on the same vertices such that two vertices of $\overline{G}$ are adjacent if and only if they are not adjacent in G.

Figure 2.7 shows an example of Petersen graph (left) and its complement graph (right).
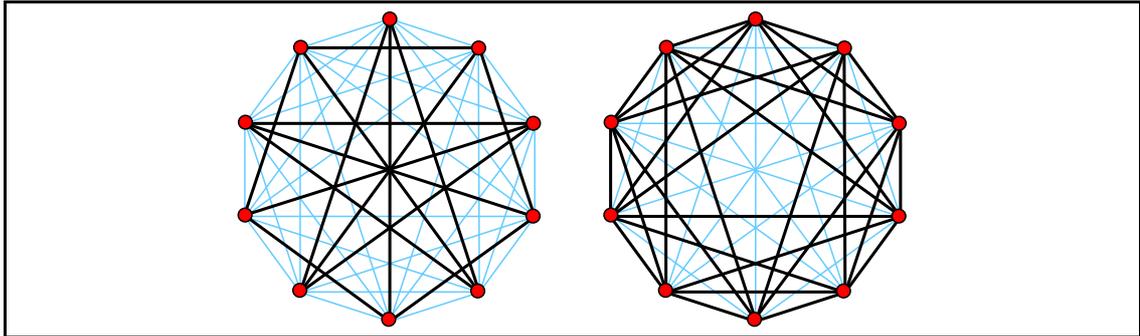
Figure 2.7: Petersen graph and its complement[1].

VERTEX DELETIONS

Deleting or adding some vertices and edges are common methods to construct graphs.

Let G be a directed graph. If $W \subset V_G$, then we define $G - W$ to be $G[V_G \setminus W]$, i.e., the subgraph of G obtained by deleting the vertices in W and all edges adjacent with them.

CLOSED NEIGHBOURHOOD DELETIONS

We define another method of vertex deletion. Let $N_G[x]$ be the closed neighbourhood of vertex x in directed graph G. We define $\overline{G_x} := G - N[x]$, i.e., the induced subgraph of G that is obtained by removing the closed neighbourhood of a vertex x. Analogously, for a subset $S \subseteq V_G$ we define $\overline{S} = G - \bigcup_{x \in S} N_G[x]$. That is, the induced subgraph obtained by removing all vertices in S and their neighbours. Figure 2.8 shows an example of $\overline{G_u}$.

Similarly we can also consider edge deleting and adding instead of vertex [14]. However, since it is not needed in the sequel, we omit it here.

---

[1]This figure is from *Complement graph. (2013, December 20). In Wikipedia, The Free Encyclopedia.* Retrieved 09:44, February 20, 2014, from *http://en.wikipedia.org/w/index.php?title=Complement_graph&oldid=587001685*.
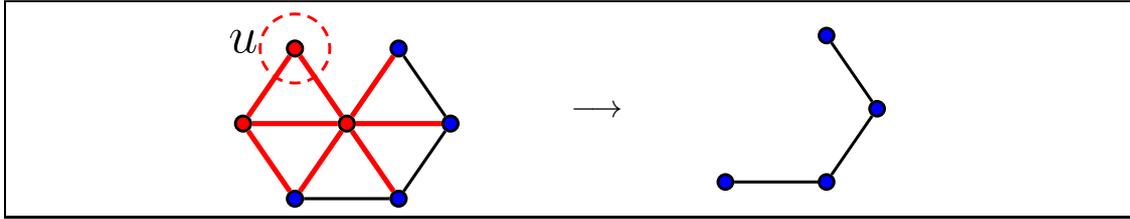
Figure 2.8: Vertex deletion $\overline{G}_u = P_4$.

DUPLICATIONS, CONTRACTIONS

Definition 2.1.4. The *vertex duplication* of a graph G by a vertex x produces a new graph $G' = G \circ x$, by adding a new vertex x' and then fastening it to the neighbourhood of x $(N_G(x') = N_G(x))$.

Obviously, this operation can be iterated.

Definition 2.1.5. The *vertex multiplication* of G by the positive integer vector $h = (h_1, \ldots, h_n)$ is the graph $H = G \circ h$ whose vertex set consists of h_i copies of x_i, and each copy of x_i is adjacent to a copy of x_j in H if and only if x_i is adjacent to x_j in G.
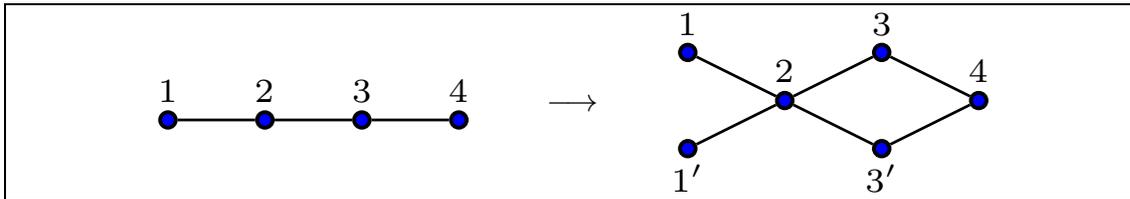
Figure 2.9: Vertex multiplication: $P_4 \circ (2, 1, 2, 1)$.

Duplication always enlarges graphs. Dually, we can also reduce a graph.

Definition 2.1.6. The *contraction* of a pair of vertices v_i and v_j of a graph produces a graph in which the two vertices v_1 and v_2 are replaced with a single vertex v such that v is adjacent to the union of the vertices to which v_1 and v_2 were originally adjacent.

Example 2.1.2. We give two examples:

1. Figure 2.9 shows a vertex multiplication of P_4 by duplication vertex 1 and 3.

2. In vertex contraction, it does not matter if v_1 and v_2 are connected by an edge or not. If they are, we produces a loop, see Figure 2.10.

Remark 2.1.1. In this thesis, contraction is referred to as vertex contraction, even though in some literature [14] contraction is referred to as edge contraction or is ambiguous about the distinction between vertex contraction and edge contraction.
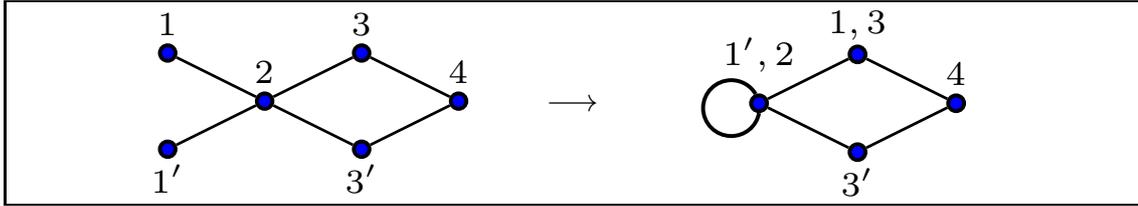
Figure 2.10: Vertex contraction $\{1, 3\}$ and $\{1', 2\}$.

Quotient graphs

Forming a quotient graph is another way of constructing new graphs from old. A *partition* $\mathcal{P}$ of the vertices set V_G is a division into disjoint subsets: $\mathcal{P} = \{S_1, \ldots, S_k\}$, i.e. $S_i \cap S_j = \emptyset$ for any i, j and $\bigcup_{1 \leq i \leq k} S_i = V_G$. *Quotient graph* induced by the partition $\mathcal{P}$, written $G/\mathcal{P}$, is the graph $\widetilde{G} = (\widetilde{V}, \widetilde{E})$ with vertex set $\widetilde{V} = \{S_1, \ldots, S_k\}$, in which $(S_i, S_j) \in \widetilde{E}$ if and only if $\exists v \in S_i, w \in S_j$ such that $(v, w) \in E$.

Remark 2.1.2. In our definition we allow loops, i.e., for the case $i = j$, if there exists $(v, w) \in E$ and $(v, w) \in S_i$, then there is a loop on S_i in the quotient graph.
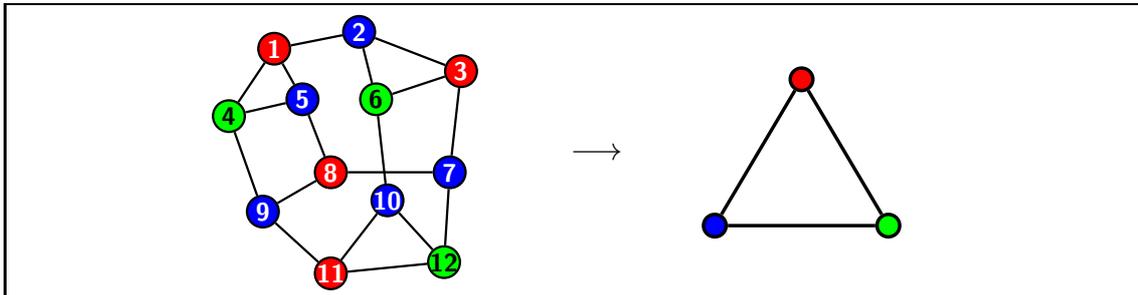
Figure 2.11: Quotient graph of Frucht graph (left) is K_3.

If we take the partition $\mathcal{P} = \{S_1, S_2, S_3\}$ by different colours of Frucht graph, i.e. $S_1 = \{1, 3, 8, 11\}$ (red vertices), $S_2 = \{2, 5, 7, 9, 10\}$ (blue vertices), $S_3 = \{4, 6, 12\}$ (green vertices), then the quotient graph of Frucht graph is K_3 (Figure 2.11).

Graph classes

In graph theory, some classes of graphs are frequently considered, due to their simple but well-behaved structural properties. For instance, connected graphs are a nice class of graphs. In this subsection we introduce some of these concepts which we will need for later use.

PATH GRAPHS

A path in a graph is formed by a sequence of distinct vertices $v_1, \ldots, v_n$ such that (v_i, v_{i+1}) is an edge for all $1 \leq i < n$. A *path graph* (or simply *path*) with n vertices, denoted by P_n, is a graph induced by a path. As in most graph theory literature, we do not make a distinction between a path and its path graph.

An *oriented path* $\overrightarrow{P}_n$ with n vertices, is a directed graph G where $V_G = \{v_1, \ldots, v_n\}$ and for every $1 \leq i < n$ either $(v_i, v_{i+1}) \in E_G$ or $(v_{i+1}, v_i) \in E_G$ (but not both), and there are no other edges. Thus an oriented path is any orientation of an undirected path. We denote by Path the class of all finite oriented paths. A *directed path* with n vertices, denoted by $\overrightarrow{P_n}$, is a path which the edges are oriented in the same direction, i.e. $(v_i, v_{i+1}) \in E_G$ or $(v_{i+1}, v_i) \in E_G$ for all $1 \leq i < n$. A path P_n can be seen as an underlying graph of an oriented path $\overrightarrow{P}_n$ and a directed path.

Figure 2.12 shows an oriented path, a directed path and a path.
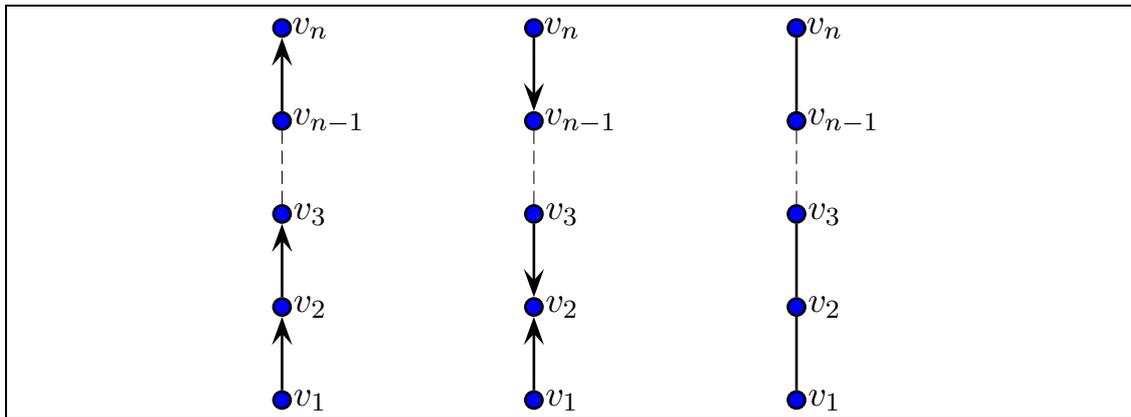
Figure 2.12: Directed path, oriented path and path.

CYCLE GRAPHS

A *(unoriented) cycle graph* (or simply *cycle*[1]) with n ($n \geq 3$) vertices $\{1, 2, \ldots, n\}$, denoted by C_n, is a simple graph with edges $(i, i+1)$ for all $1 \leq i < n$ and $(1, n)$. We denote by Cycle the class of graphs formed by all C_k, $k \geq 3$, and by Cycles the class of graphs formed by disjoint union of finitely many graphs in Cycle.

A *directed cycle* is formed by a cycle with edges oriented in one direction.

Definition 2.1.7. *Oriented cycle* with n vertices, denote by $\overrightarrow{C}_n$, is an orientation of a cycle graph, with all the edges being oriented in the same direction. We denote

---

[1]Like path, we also do not distinct cycle graph and cycle.

(a) C_3

(b) C_6
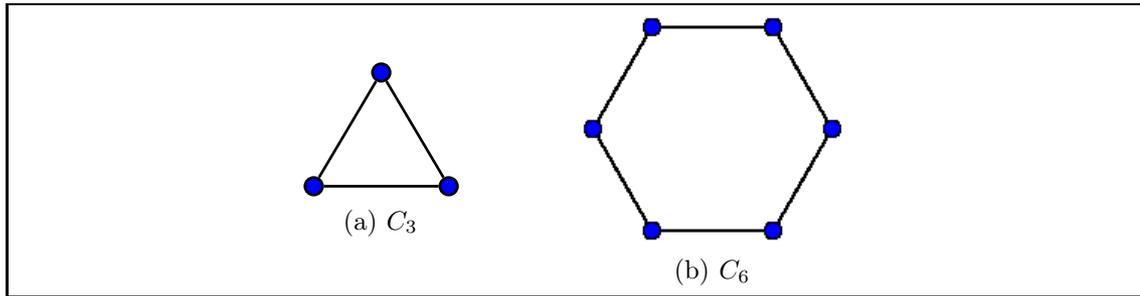
Figure 2.13: Cycle graphs.

by DiCycle the class of directed graphs formed by all $\overrightarrow{C}_k$, $k \geq 3$, and DiCycles the class of directed graphs formed by disjoint union of finitely many graphs in DiCycle.
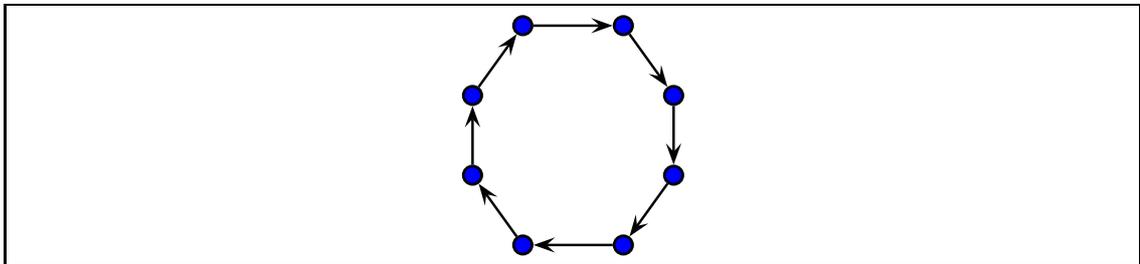
Figure 2.14: Oriented cycle with 8 vertices.

Reflexive, transitive graphs

A graph is *reflexive* if for every vertex there is a loop. A graph is *transitive* if for any pair of edges (v_i, v_j) and (v_j, v_k), there is an edge (v_i, v_k). A *directed acyclic graph* (or shortly DAG) is a directed graph with no directed cycles. Reflexive graphs, transitive graphs and directed acyclic graphs are representations of relational structure which we will introduce in the second part of this chapter.
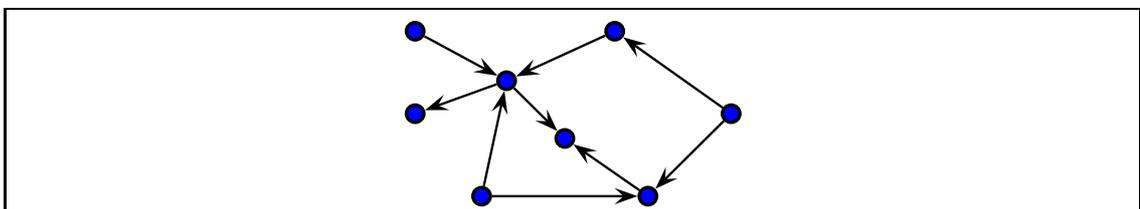
Figure 2.15: A directed acyclic graph.

COMPLETE GRAPHS

A *complete graph* with n vertices, denoted by K_n, is a simple graph in which every pair of distinct vertices is connected by a unique edge. For convenience, we denote the vertices of K_n by $1, 2, \ldots, n$.

Figure 2.16 shows the complete graphs with vertex number 1 to 4.

(a) K_1 (b) K_2 (c) K_3 (d) K_4
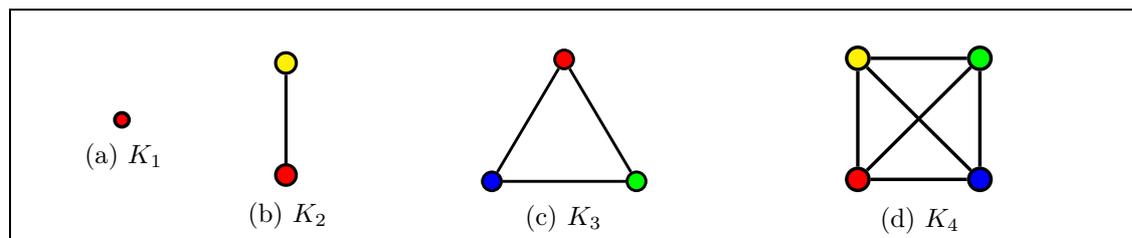
Figure 2.16: Complete graphs.

M-PARTITE GRAPHS

A *m-partite graph* is a simple graph whose vertices can be decomposed into k disjoint sets $U_1, \ldots, U_m$, such that every edge connects a vertex in U_i to one in U_j for some distinct $i, j \in \{1, 2, \ldots, m\}$. A *complete m-partite graph* is a m-partite graph where every vertex in U_i is connected to every vertex in U_j.

See Figure 2.17 for an example of complete 3-partite graph.
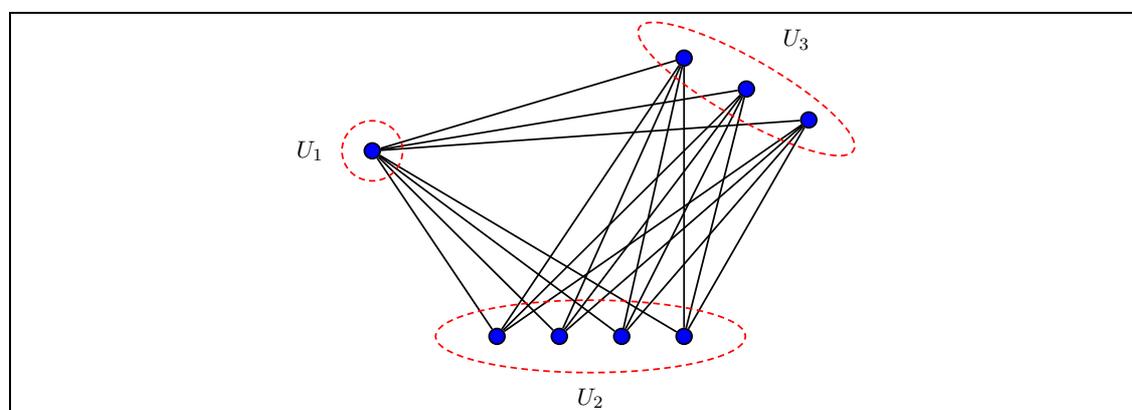
Figure 2.17: A complete 3-partite graph.

TREES

A *tree* is a connected simple graph which has no cycles. A *oriented tree* is a directed tree having no symmetric pair of directed edges. Figure 2.18 shows an

oriented tree (left) and a tree (right).
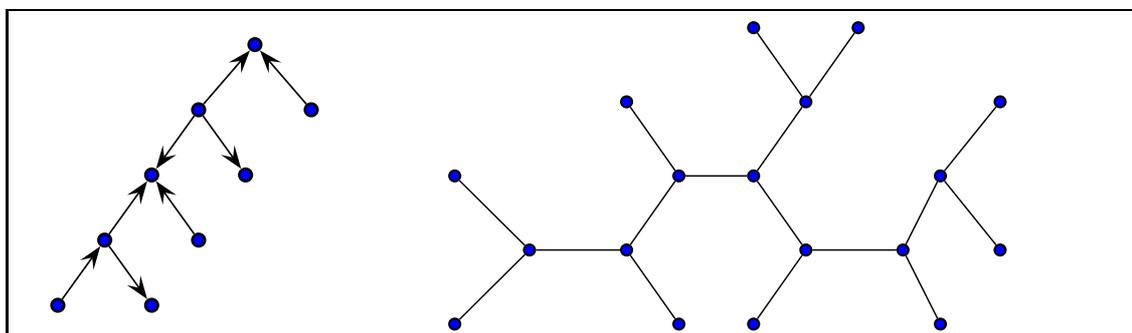
Figure 2.18: Trees.

Graph colourings

A *(vertex) colouring* is a way of colouring the vertices of a simple graph such that no two adjacent vertices share the same colour. A colouring using at most k colours is called a *(proper) k-colouring*. The *chromatic number* of a simple graph G, denoted by $\chi(G)$, is the smallest k such that G admits a k-colouring.

Figure 2.16 shows n-colourings of the graph K_n for $n = 1, \ldots, 4$.

Example 2.1.3. It is non-trivial to determine the chromatic number of an arbitrary graph, but for some certain graphs, the chromatic numbers are known:

1. $\chi(K_n) = n$.

2. $\chi(P_n) = 2$.

3. $\chi(C_{2k+1}) = 3$, $\chi(C_{2k}) = 2$.

4. The chromatic number of the Frucht graph (Figure 2.11 left) as well as the Petersen graph (Figure 2.7) is 3.

Graph isomorphisms

An *isomorphism* f of graphs G and H is a bijection between the vertex sets of G and H such that any two vertices u and v of G are adjacent in G if and only if $f(u)$ and $f(v)$ are adjacent in H. In the case when G and H are the same graph, an isomorphism is called an automorphism.

---

[1]This example is from *Graph isomorphism. (2014, February 18). In Wikipedia, The Free Encyclopedia.* Retrieved 09:48, February 20, 2014, from `http://en.wikipedia.org/w/index.php?title=Graph_isomorphism&oldid=596007728`.

Graph G	Graph H	An isomorphism between G and H
		$f(a) = 1$ $f(b) = 6$ $f(c) = 8$ $f(d) = 3$ $f(g) = 5$ $f(h) = 2$ $f(i) = 4$ $f(j) = 7$
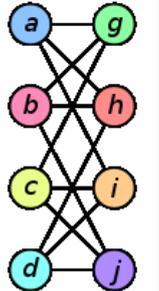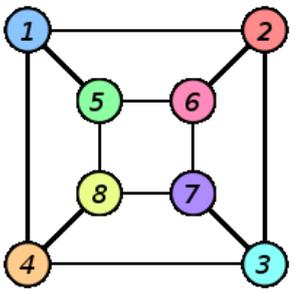
Table 2.1: Graph isomorphism [1].

We mention that the graph isomorphism problem is one of the few standard problems in computational complexity theory belonging to NP, but not known to belong to either of its well-known (and, if P $\neq$ NP, disjoint) subsets: P and NP-complete.

Matrix representations of graphs

ADJACENCY MATRICES

There are several matrix representations of a graph. The most intuitive one is its adjacency matrix. *Adjacency matrix* of a graph G, denoted by A_G, is a $|G| \times |G|$ matrix. If there is an edge from a vertex x to a vertex y then the element $a_{x,y}$ is 1, otherwise it is 0. If graph G is a multigraph or weighted graph, then the elements $a_{x,y}$ are taken to be the number of edges between the vertices or the weight of the edge (x, y), respectively. The adjacency matrix is in many cases very useful. For example, in computing, it makes finding subgraphs and reversing a directed graph easy.

Example 2.1.4. The adjacency matrices of Fig. 2.3c and Fig. 2.3e are as below:

$$\begin{pmatrix} 1 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{pmatrix} \qquad \begin{pmatrix} 1 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{pmatrix}$$

Suppose two (directed) graphs G and H whose adjacency matrices A_G and A_H are given. Then G and H are isomorphic if and only if there exists a permutation matrix P such that $PA_GP^{-1} = A_H$.

DDMs and DRMs

Let G be a connected graph. An *equitable partition* of G is a partition $B_1, \ldots, B_k$ of its vertex set such that any vertex in B_i has the same number of neighbours in B_j for any i, j. A *degree decomposition matrix* (also called *degree matrix*[1] in [26] or *degree partition matrix*) of G is a square matrix $M = \{m_{i,j}\}$ of order k such that there is an equitable partition of V_G into blocks $B_1, \ldots, B_k$ satisfies $|N_G(u) \cap B_j| = m_{i,j}$ for all $u \in B_i$ and $1 \leq i, j \leq k$.

Note that a graph G can allow several degree matrices, with an adjacency matrix itself being the largest one. An equitable partition is said to be *finer* than another one when every class of its partition is a subset of some class of the latter one, e.g., the partition into singletons is finer than any other equitable partition of the same graph. In this case a canonical ordering can be imposed on the blocks, so the corresponding degree matrix, called the *degree refinement matrix*, or *DRM*, denoted by $drm(G)$, is also defined uniquely [25].

There is an efficient algorithm for computing the degree refinement matrix of a given graph [25].
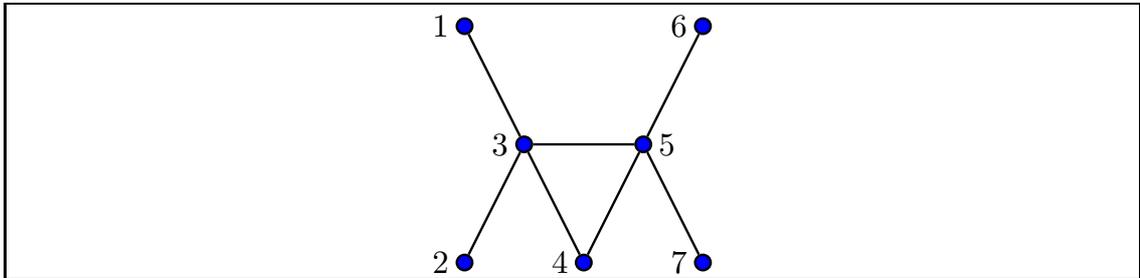
Figure 2.19: An example.

Example 2.1.5. The graph showed in Fig. 2.19 has degree decomposition

---

[1] It is different from the degree matrix which contains information about the degree of each vertex.

matrices:

$$
\begin{pmatrix}
0 & 0 & 1 & 0 & 0 & 0 & 0 \\
0 & 0 & 1 & 0 & 0 & 0 & 0 \\
1 & 1 & 0 & 1 & 1 & 0 & 0 \\
0 & 0 & 1 & 0 & 1 & 0 & 0 \\
0 & 0 & 1 & 1 & 0 & 1 & 1 \\
0 & 0 & 0 & 0 & 1 & 0 & 0 \\
0 & 0 & 0 & 0 & 1 & 0 & 0
\end{pmatrix}
\quad
\begin{pmatrix}
1 & 0 & 0 & 0 & 0 \\
2 & 0 & 1 & 1 & 0 \\
0 & 1 & 0 & 1 & 0 \\
0 & 1 & 1 & 0 & 2 \\
0 & 0 & 0 & 1 & 0
\end{pmatrix}
\quad \text{and} \quad
\begin{pmatrix}
0 & 1 & 0 \\
2 & 0 & 1 \\
0 & 2 & 0
\end{pmatrix}
$$

which are the adjacency matrix with partitions $B_1 = \{1, 2\}, B_2 = \{3\}, B_3 = \{4\}, B_4 = \{5\}, B_5 = \{6, 7\}$, and $B_1 = \{1, 2, 6, 7\}, B_2 = \{3, 5\}, B_3 = \{4\}$, respectively. The last one is the degree refinement matrix.

MATCHINGS

A *matching* M in a graph G is a set of edges which do not share endpoints. M is a *perfect matching* M if any vertex of G can be found as an endpoint in M.
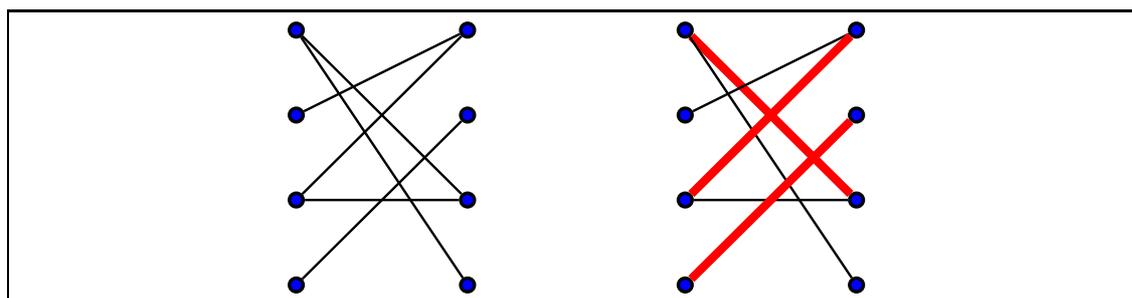
Figure 2.20: Matching.
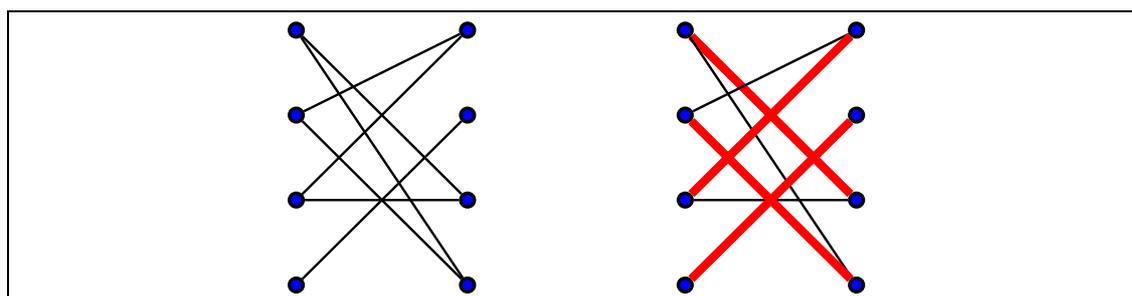
Figure 2.21: Perfect matching.

Fig. 2.20 shows a matching, but it is not a perfect matching. Actually there is no perfect matching in the graph shown in Fig. 2.20, while Fig. 2.21 shows a perfect matching.

Hall's marriage theorem, or simply Hall's theorem, gives a necessary and sufficient condition for being able to select a distinct element from each of a collection of finite sets. Here we state a simpler version.

Theorem 2.1.1 (Hall's marriage theorem). *Given a finite bipartite graph G, with bipartite sets X and Y of equal size, there is a perfect matching in G if and only if for any subset W of X, $|W| \leq |N_G(W)|$.*

Proof. If there is a perfect matching in a bipartite graph G, then for any subset W of X, $|W| \leq |N_G(W)|$. Conversely, if for any subset W of X, $|W| \leq |N_G(W)|$, then any vertex x in X has at least one neighbor. Otherwise taking $W = x$ the inequality does not hold. Similarly any vertex y in Y has at least one neighbor, otherwise taking $W = X$ the inequation does not hold. We prove by induction on $|X|$. Suppose $|X| = n$. For $n = 1$ it is trivial. Assume that the statement is true for all bipartite graphs satisfying $|X| = |Y| = k$. We want to prove that it is also true for $|X| = |Y| = k + 1$. Take vertex v in X. We consider two casses. Case 1: Suppose $\forall W \subseteq X \setminus \{v\}, |N_G(W)| \geq |W| + 1$. In this case we can match v with any of its neighbors and it is still a perfect matching. Case 2: There exists a subset $W \subseteq X \setminus \{v\}$ such that $|N(W)| = |W|$. Pick a minimum cardinality set satisfying this property. By induction and the minimality of W, we know that W can be matched to $N(W)$. Now consider $X \setminus W$, take a set $W' \subseteq X \setminus W$. Then W' muss have neighbors outside $N(W)$ otherwise the inequality does not hold for $W \cup W'$. It has to have at least $|W'|$ neighbors outside $N(W)$. Becase of the arbitrariness of W', the inequality holds for the graph induced by $X - W$ and $Y - N_G(W)$, by induction there is a perfect matching in the graph. So combining this matchings we have a perfect matching on G. $\square$

2.2   Ordered sets

In the language of graph theory, a *(partially) ordered set* is a transitive, reflexive directed acyclic graph. Orders are ubiquitous in everyday life. Order theory is related with many other mathematical branches, such as universal algebra, topology and category theory. It also has broad application in many other fields, such as the humanities and social sciences, computer science (e.g. programs, binary strings, information orderings) [13]. Several introductory books on this topic are available, such as the one by Davey and Priestley [13]. In this section we introduce the definitions and elementary facts about orders.

Basic definitions

Definition 2.2.1. Let P be a set. An *order* (or *partial order*) is a binary relation $\leq$ on P such that, for all $a, b, c \in P$, the following properties are satisfied:

1. *reflexivity*: $a \leq a$,

2. *antisymmetry*: $a \leq b$ and $b \leq a$ imply $a = b$,

3. *transitivity*: $a \leq b$ and $b \leq c$ imply $a \leq c$.

A set P equipped with an order relation $\leq_P$ is called an *ordered set* (or *partial order set*, *poset* [1]), denoted by $(P, \leq_P)$, if the relation $\leq_P$ on P is reflexive, antisymmetric and transitive. A relation $\leq_P$ on a set P which is reflexive and transitive but not necessarily antisymmetric is called a *quasi-order* or *pre-order*.

Example 2.2.1. Standard examples of orders arising in mathematics include:

1. The set of real numbers ordered by the standard less-than-or-equal relation.

2. The set of natural numbers equipped with the relation of divisibility.

3. The set of subsets of a given set (its power set) ordered by inclusion.

4. The vertex set of a directed acyclic graph ordered by reachability.

Visualization of an order

An order can be visually represented by a *Hasse diagram*, which is a graph drawing where the vertices are the elements of the order and the order relation is indicated by both the edges and the relative positioning of the vertices: if two elements $x \leq y$, then there exists a path from x to y and y sits up of x.

Example 2.2.2. Figure 2.22 shows two Hasse diagrams:

1. The Hasse diagram of the divisor order (right): the set of all divisors of 60 ordered by divisibility,

2. the Hasse diagram of the inclusion order (left): the set of the power set of $\{1, 2, 3\}$ equipped with subset relation $\subseteq$. Note that even though $\{3\} < \{1, 2, 3\}$ (since $\{3\} \subset \{1, 2, 3\}$), there is no edge directly between them because there are inbetween elements: $\{2, 3\}$ and $\{1, 3\}$. However, there still exists a path from $\{3\}$ to $\{1, 2, 3\}$.

---

[1]The name of poset is an abbreviation for partially ordered set which was coined by Garrett Birkhoff in his influential book *Lattice Theory* [7].
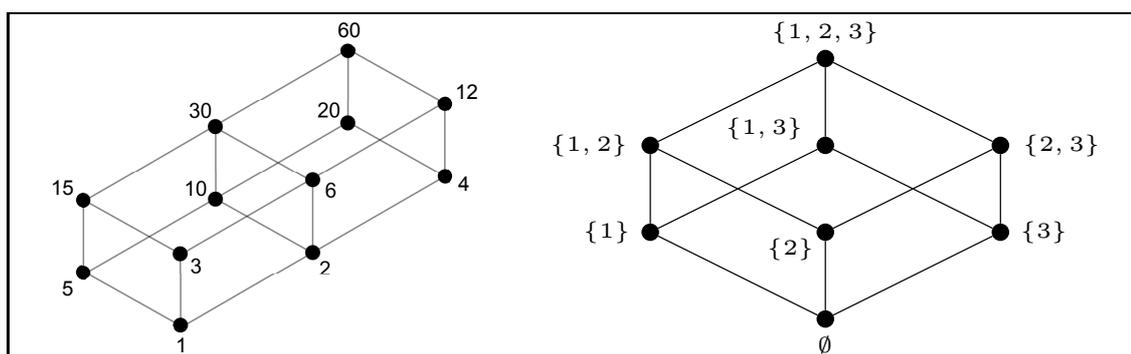
Figure 2.22: Hasse digrams.

Functions between orders

Definition 2.2.1. Let $(P, \leq_P)$ and $(Q, \leq_Q)$ be two orders, and f is a mapping from $(P, \leq_P)$ to $(Q, \leq_Q)$:

1. f is an *order preserving* or *monotone*, if $a \leq_P b$ implies $f(a) \leq_Q f(b)$ in Q [1].

2. f is an *order reflecting*, if $f(a) \leq_Q f(b)$ implies $a \leq_P b$ [2].

3. f is an *order embedding* E if it is both order preserving and order reflecting, i.e. $a \leq_P b$ if and only if $f(a) \leq_Q f(b)$ for every $a, b \in P$.

4. An *order isomorphism* is a surjective order embedding.

5. An *order automorphism* is an order isomorphism from an ordered set to itself.

Example 2.2.3. We again give some examples to illustrate these concepts:

1. The mapping that maps a natural number to its successor is clearly an order preserving with respect to the natural order; any mapping from a discrete order is an order preserving, for example, consider any mapping from a set ordered by the identity order '='.

2. Mapping each natural number to the corresponding real number gives an example for an order embedding.

3. There is an order isomorphism between two orders with the same Hasse diagrams.

---

[1] Like homomorphism in graphs

[2] In the language of graph theory, it is a full homomorphism.

4. The mapping that maps a natural number to its successor is an order automorphism with respect to the natural order.

Definition 2.2.2. Let $(P, \leq_P)$ and $(Q, \leq_Q)$ be two orders, if there is an order embedding E from $(P, \leq_P)$ to $(Q, \leq_Q)$, we say $(P, \leq_P)$ is a *suborder* of $(Q, \leq_Q)$.

Note that the suborder or order preserving relationship does not direct correspond to the relationship of their Hasse diagram. This is simply because the Hasse digraphs miss all the transitive edges.

The visualization of orders with Hasse diagrams has a straightforward generalization: instead of displaying lesser elements below greater ones, the direction of the order can also be depicted by giving directions to the edges of a graph. In this way, each order is seen to be equivalent to a directed acyclic graph (DAG), where the vertices are the elements of the order and there is a directed path from a to b if and only if $a \leq b$. For example, Figure 2.23 is a directed acyclic graph representation of the inclusion order in the second example.
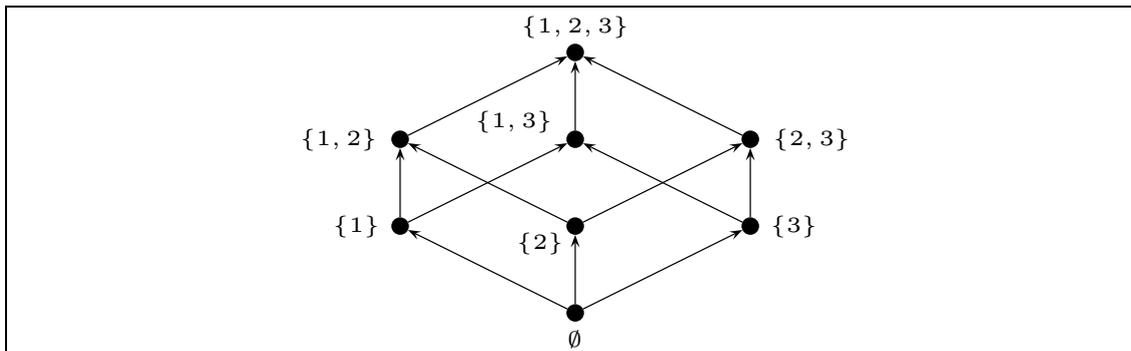
Figure 2.23: Directed acyclic graph.

Basic properties

CHAINS AND ANTICHAINS

Definition 2.2.3. An ordered set $(P, \leq_P)$ is a *chain*, or *totally ordered set*, if for all $x, y \in P$, either $x \leq_P y$ or $y \leq_P x$. An ordered set $(P, \leq_P)$ is an *antichain* if $x \leq_P y$ in $(P, \leq_P)$ implies $x =_P y$.

Clearly, any suborder of a chain (an antichain) is a chain (an antichain). The Hasse diagraph of a chain is a path, and that of an antichain is a set of isolated vertices.

Definition 2.2.4. A *increasing chain* is a sequence of elements $(x_1, x_2, \dots)$ in the set $(P, \leq_P)$ such that $x_1 \leq_P x_2 \leq_P x_3 \leq_P \dots$. Dually, a *decreasing chain* is a sequence of elements $(x_1, x_2, \dots)$ in set $(P, \leq_P)$ such that $\dots \leq_P x_3 \leq_P x_2 \leq_P x_1$.

As examples, the usual numerical order of positive integers is a (increasing) chain, and the inverse order $(\mathbb{N}, \geq)$ is a decreasing chain. The equality order of positive integers is an antichain.

LATTICE

Let $(P, \leq_P)$ be an ordered set and $(S, \leq_S)$ be a suborder of $(P, \leq_P)$. An element $x \in P$ is an *upper bound* of $(S, \leq_S)$ if $s \leq_P x$ for all $s \in S$. If x is the least element among all the upper bound, we call it *least upper bound* (or *join, supremum*). An element $x \in P$ is a *lower bound* of $(S, \leq_S)$ if $x \leq_P s$ for all $s \in S$. If x is the greatest element among all the lower bound, we call it *greatest lower bound* (or *meet, infimum*). Note that least upper bound and greatest lower bound are not necessary unique.

Definition 2.2.5. A *lattice* is an ordered set in which any two elements have a least upper bound and a greatest lower bound.

Example 2.2.4. There are some examples of lattices:

1. The real numbers ordered by the standard less-than-or-equal relation form a lattice, under the operations of 'min' and 'max'.

2. For any set A, the collection of all subsets of A, ordered by inclusion, is a lattice.

3. Although the set of all divisors of 60 ordered by divisibility is a lattice, the subset $\{1, 2, 3\}$ so ordered is not a lattice because the pair 2,3 lacks a join, $\{2, 3, 6\}$ neither because 2,3 lacks a meet.

As we already stated, graph-theoretically, an order is a *transitive, acyclic, directed, reflexive* graph. Note that an embedding is necessarily injective, since $f(a) = f(b)$ implies $a \leq_P b$ and $b \leq_P a$ (since an order is a reflexive graph).

Interestingly, except for graph theoretical interpretation, orders can straightforwardly be viewed from many other perspectives. For example: model-theoretically, an order is a set equipped with a *reflexive, antisymmetric, transitive* binary relation. An order embedding from P to Q is an isomorphism from P to an elementary substructure of Q. Category-theoretically, an order is a *small, skeletal* category such that each homset has at most one element. An order embedding P to Q is a full and faithful functor from P to Q which is injective on objects, or equivalently an isomorphism from P to a full subcategory of Q.

Density

An order $(P, \leq_P)$ is *dense*, if for any $x, y \in P$ with $x \leq_P y$, we can find another element $z \in P$ such that $x \leq_P z \leq_P y$. Conversely, if for a pair x, y with $x \leq_P y$ we cannot find another element z in between, we say the pair x, y is a *gap*, denote it by $[x, y]$.

For example, the real number ordered set $(\mathbb{R}, \leq)$ is dense, however, the integer number ordered set $(\mathbb{Z}, \leq)$ is not dense because there are gaps $[k, k+1]$ for any $k \in \mathbb{Z}$.

Local finiteness

For a given order $(P, \leq_P)$ we call $\{y \mid y \leq_P x\}$ the *down-set* of x in $(P, \leq_P)$. When there is no confusion about particular order, we denote the down-sets by $\downarrow x$. Similarly we call the set $\{y \mid x \leq_P y\}$ the *up-set* of x in $(P, \leq_P)$ and denote it by $\uparrow x$.

We say that a countable order is *past-finite* if every down-set is finite. Similarly a countable order is *future-finite* if every up-set is finite. An order $(P, \leq_P)$ is said to be *locally-finite* if for every $a, b \in P$, $a \leq_P b$, the interval $[a, b]_P = \{c \mid a \leq_P c \leq_P b\}$ is finite. Both past-finite and future-finite orders are special case of *locally-finite*, that is, if for every $a, b \in P$, $a \leq_P b$, the interval $[a, b]_P = \{c \mid a \leq_P c \leq_P b\}$ is finite.

Duality

We say that a pair (a, b) is a *simple duality pair* in an order $(P, \leq_P)$, if $a, b \in P$, and any element $p \in P$ is either in the up-set of a, or in the down-set of b.

Furthermore, let A and B be two sets of P, we say that (A, B) is a *generalized finite duality pair* in $(P, \leq_P)$, if for any element $p \in P$ there either exists an $a \in A$ such that $a \leq_P p$ or there exists a $b \in B$ such that $p \leq b$.

In Chapter 5, we will discuss more properties of orders, such as universality in Section 5.1, and present some more interesting results regarding orders.

Homomorphisms and Constrained Homomorphisms

This chapter concerns graph homomorphisms and several common variants of graph homomorphisms, and introduces a new variant of them that is key to our later work. The theory of graph homomorphisms, which underlies much of this thesis, developed from the well-known graph-theoretical problem of 'graph colouring'. Meanwhile, many graph-theoretical problems can be generalized to notions of *constrained homomorphisms*. This chapter is divided into four parts, in the first three of which we overview existing structural results. In Section 3.1 we first recall the basic properties of ordinary graph homomorphisms. In Section 3.2 we review several types of *globally constrained homomorphisms* such as monomorphism, embedding, surjective homomorphism, full homomorphism and compaction. In Section 3.3 we discuss three kinds of *locally constrained homomorphisms*: locally injective, locally bijective and locally surjective. In the final section we introduce a new notion that we call 'relation' which can be viewed as a generalization of surjective homomorphism. We develop the basic properties of 'relation', as preparation for the deeper investigation in Chapter 4.

3.1   Graph homomorphisms

Graph colouring is one of the most well-known problems in the field of graph theory. As an example, the four colour conjecture, which was postulated by Francis

Frucht graph	Graph K_3	An homomorphism from Frucht graph to K_3
		$f(1) = $ Red $f(2) = $ Blue $f(3) = $ Red $f(4) = $ Green $f(5) = $ Blue $f(6) = $ Green $f(7) = $ Blue $f(8) = $ Red $f(9) = $ Blue $f(10) = $ Blue $f(11) = $ Red $f(12) = $ Green

Table 3.1: Graph homomorphism.

Guthrie while trying to colour a map of the counties of England, stating that four colours were sufficient to colour the map so that no regions sharing a common border receive the same colour. In terms of graph theory: all planar graphs can be coloured by 4 colours. As a generalization of graph colouring, the study of graph homomorphisms dates from 1960s. It was pioneered by G. Sabidussi [74], and by Z. Hedrlín and A. Pultr [35], and rapidly developed by P. Hell and J. Nešetřil [40]. In this section we give a brief summary of the main concepts and results in this area. For more details please refer to the recent monograph [40].

Definition 3.1.1. Let $G = (V_G, E_G)$ and $H = (V_H, E_H)$ be two directed graphs, a *homomorphism* of G to H, is a mapping $f : V_G \to V_H$ such that $(f(u), f(v)) \in E_H$ whenever $(u, v) \in E_G$.

A homomorphism of G to H is also called an *H-colouring of G*. If there exists a homomorphism $f : G \to H$ we write, $G \to H$, and $G \nrightarrow H$ means there is no such homomorphisms.

Let f be a homomorphism of graph $G = (V_G, E_G)$ to graph $H = (V_H, E_H)$, and S a subset of V_G, M a subset of V_H, we define $f(S) := \{v \in V_H \mid \exists x \in S \text{ s.t. } f(x) = v\}$ as the *image of S*, $f^{-1}(M) := \{x \in V_G \mid \exists u \in M \text{ s.t. } f(x) = u\}$ as the *domain of M*. Moreover, the *image of homomorphism f*, denoted by $I(f)$, is defined as the image of V_G, and the *domain of f*, denoted by $D(f)$, is defined as the domain of V_H. Note that since f is a mapping, the domain of f is always V_G.

Example 3.1.1. In order to build an intuition about the existence of graph homomorphisms, we give some examples :

1. If graph G is a subgraph of graph H, then the subgraph structure induces a homomorphism from G to H (Figure 2.5a). However, there may have other homomorphisms of G to H which are not induced by subgraph structure.

2. There is a homomorphism of Frucht graph to complete graph K_3 as in Fig. 3.1. Because K_3 is a subgraph of Frucht graph, there is a homomorphism of K_3 to Frucht graph.

3. $P_2 \to K_3$, since P_2 is a subgraph of H_3. However $K_3 \not\to P_2$, that is because the size of K_3 is larger than P_2, then there are at least two vertices of K_3 which must be mapped to the same vertex of P_2, this would induce a loop. However P_2 is loop-free.

It is easy to verify that the composition of homomorphisms is also a homomorphism. Moreover, composition of homomorphisms is associative. Let $f : V_G \to V_H$ be a homomorphism of G to H. If the mapping f is injective, we call it an *injective homomorphism*. The homomorphism f naturally induces an edge mapping of E_G to E_H. If f and the induced edge mapping are both surjective, we call it is a *surjective homomorphism*.

Proposition 3.1.1 (Decomposition law [40]). *Let G and H be any directed graphs. Every homomorphism $f : G \to H$ can be written as $f = i \circ s$ where s is a surjective homomorphism and i is an injective homomorphism.*

We leave the proof to the section on surjective homomorphisms that begins on page 39.

Proposition 3.1.2. *A homomorphism $f : G \to K_k$ is precisely a k-colouring of the graph G.*

Proof. For $1 \le i \le k$, we colour the vertices of graph G in the set $f^{-1}(i)$ by colour i. Clearly for distinct i and j, $f^{-1}(i) \cap f^{-1}(j) = \emptyset$. And for any i, $f^{-1}(i)$ is an independent set, otherwise there is a loop on vertex i of K_k. Hence homomorphism f is a proper k-colouring of graph G. □

Proposition 3.1.2 explains the reason that a homomorphism from G to H is also called an H-colouring, as a graph homomorphism can be seen as a generalization of a graph colouring.

Homomorphisms preserve adjacency

The fact that homomorphisms preserve vertex adjacency has interesting implications. The following propositions are derived from these adjacency preservations.

Proposition 3.1.3. *A mapping $f : V_{P_k} \to V_G$ is a homomorphism of P_k to graph G if and only if the sequence $f(1), f(2), \ldots, f(k)$ is a walk in graph G.*

Consequently, we obtain the following useful corollary.

Corollary 3.1.4. *If $f : G \to H$ is a homomorphism, then for any two vertices u, v of graph G, the distance between $f(u)$ and $f(v)$ in graph H is at most the distance between u and v in G.*

Remark 3.1.1. Even though the existence of homomorphisms is monotone with respect to distance, it is not monotone with respect to radius or diameter. For example, $P_2 \to K_3$ and $P_2 \to P_5$, but the diameter as well as the radius of P_2 is between K_3 and P_5. In contrast, we will see an analogous structure which is monotone with respect to diameter as well in Corollary 4.1.10 on page 57.

Similar to paths, we have the following result for homomorphisms from cycle graphs.

Proposition 3.1.5. *A mapping $f : V_{C_k} \to V_G$ is a homomorphism of C_k to G if and only if the sequence $f(0), f(1), \ldots, f(k-1)$ is a closed walk in G.*

Corollary 3.1.6. $C_{2k+1} \to C_{2l+1}$ *if and only if $l \leq k$.*

Homomorphism equivalence

In the last section, we saw that the existence of homomorphisms is monotone with respect to chromatic number, so that if two directed graphs G and H have different chromatic numbers, then $G \to H$ and $H \to G$ cannot hold at the same time. Two directed graphs G and H such that each is homomorphic to the other are called *homomorphically equivalent*, and denoted by $G \sim H$. Homomorphic equivalence is indeed an equivalence relation. To see this, let G, H and K be directed graphs. Since every directed graph is homomorphically equivalent to itself (self-adjoint); $G \sim H$ implies $H \sim G$ (symmetric); If $G \sim H$ and $H \sim K$, then $G \sim K$ follows from the fact that homomorphisms are closed under composition (transitive). We immediately get that homomorphically equivalent directed graphs have the same chromatic number. In this section we focus on the question of how to characterize homomorphically equivalent directed graphs.

Definition 3.1.2. A *retraction* of a directed graph G to its subgraph H is a homomorphism $r : G \to H$ such that $r(x) = x$ for all $x \in V_H$.

If there is a retraction of G to H, we call H is a *retracted graph* of G. Recall that the a subgraph has a homomorphism to its original graph, so a graph and (any of) its retracted graphs are homomorphism equivalent. Figure 3.1 shows a series of retractions.
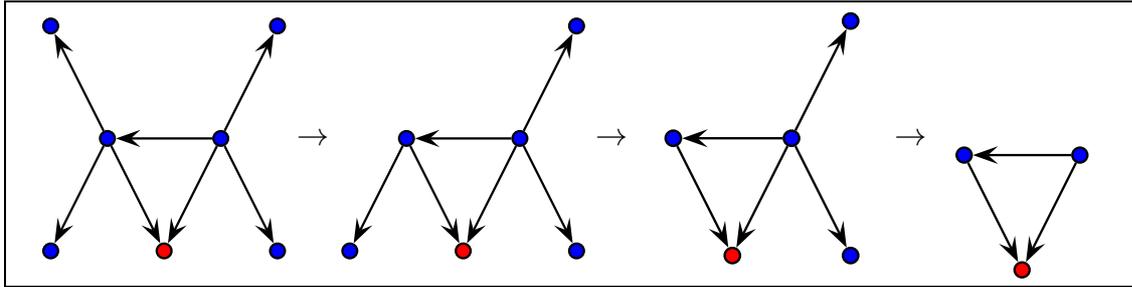
Figure 3.1: A series of retractions.

The most natural idea is to choose the smallest directed graph (in the sense of size) in an equivalence class as its representative. Firstly, given a graph G, we can try to find the smallest subgraph which G has a retraction to. This leads to an important concept which was independently discovered and was given different names such as *minimal graphs* and *unretractive graphs* [20, 38, 69].

Definition 3.1.3. A *core of (directed) graph* G is the smallest graph (in the sense of size) which G admits a retraction to. A (directed) graph is a *core* if it does not retract to a proper subgraph.

It is easy to see that every directed graph is homomorphically equivalent to a (up to isomorphism) unique core. Thus we can denote the core of a given graph G by G_{core}. For example, the core of first three directed graphs in Figure 3.1 is the fourth directed graph.

Two graphs are homomorphically equivalent if and only of their cores are isomorphic, therefore we can use core to be the representative for an equivalence class. However, the computational complexity of determining a core is NP-hard. As we know, the 3-colouring problem, i.e., testing whether a graph has a homomorphism to triangle K_3, is NP-complete. Given a graph G, consider G' to be the disjoint union of G and K_3. If we can determine the core of G', then it is easy to decide whether G is 3-colourable or not. If the core of G' is K_3, then G is 3-colourable, otherwise G is not 3-colourable.

H-colouring problem

In this section we introduce the main results about the existence of homomorphisms from an algorithmic perspective.

Let H be a fixed directed graph. The *homomorphism problem for H* asks whether or not an input directed graph G admits a homomorphism to H. Recall that a homomorphism of G to H is also called an H-colouring of G. Thus the homomorphism problem for H is also called the *H-colouring problem*, and denoted by H-COL.

The H-colouring problem can be analogously stated for any general relational system H, in this case the problem is also known as *constraint satisfaction problem* (or *CSP*) with a template H, and denoted as H-CSP [40]. Many studies have been carried out regarding the constraint satisfaction problem, and there is an introductory book by E. Tsang [83].

We first consider the H-colouring problem for graphs (instead of digraphs). Clearly all graphs which have a loop have core O, the graph with one vertex and a loop. The problem of O-colourablity is trivial.

Recall that the usual k-colouring problem for simple graphs is polynomial time solvable when $k \leq 2$ and is NP-complete for $k > 2$. Such a classification result implies that we have a *dichotomy* of possibilities — each k-colouring problem is polynomial time solvable or NP-complete. Analogously, H-clouring has also a dichotomy of possibilities.

Theorem 3.1.7. *Let H be a graph. If H is bipartite or contains a loop, then the H-colouring problem has a polynomial time algorithm. Otherwise the H-colouring problem is NP-complete.*

This theorem was first proved in [39] by Hell and Nešetřil. There is a new proof by Bulatov [11] which follows the same idea but uses an algebraic approach.

The H-colouring problem for directed graphs has received much attention, and yet no graph-theoretic classification has been obtained, or conjectured [40]. Even the following dichotomy conjecture is still open.

Conjecture 3.1.8. *Let H be a directed graph, the H-colouring problem is polynomial time solvable or NP-complete.*

There is a nice conjectured dichotomy classification for directed graphs with positive indegree and outdegree at each vertex. The conjecture is still open though many special cases of the conjecture have been verified [31].

Conjecture 3.1.9 (Bang-Jensen, Hell [5]). *Suppose H is a connected directed graph in which each vertex has positive indegree and outdegree. If the core of H is a directed cycle, then the H-colouring problem is polynomial. Otherwise it is NP-complete.*

3.2 Globally constrained homomorphisms

We consider several types of (globally) constrained homomorphisms.

1. A homomorphism $f : G \to H$ is a *monomorphism*[1], (also called *injective homomorphism* [40] or *M-morphism*) if it is injective.

2. A homomorphism $f : G \to H$ is a *full homomorphism*, or *F-morphism*, if $(f(u), f(v)) \in E_H$ implies $(u, v) \in E_G$.

3. A homomorphism $f : G \to H$ is an *embedding*, or *E-morphism* if it is both a monomorphism and a full homomorphism.

4. A homomorphism $f : G \to H$ is a *vertex surjective homomorphism*, or *VS-morphism*, if $f(V_G) = V_H$. A homomorphism $f : G \to H$ is a *surjective homomorphism*, or *S-morphism*, if it is vertex surjective and moreover for every $(u, v) \in E_H$ there is $(u', v') \in E_G$ such that $f(u') = u$ and $f(v') = v$.
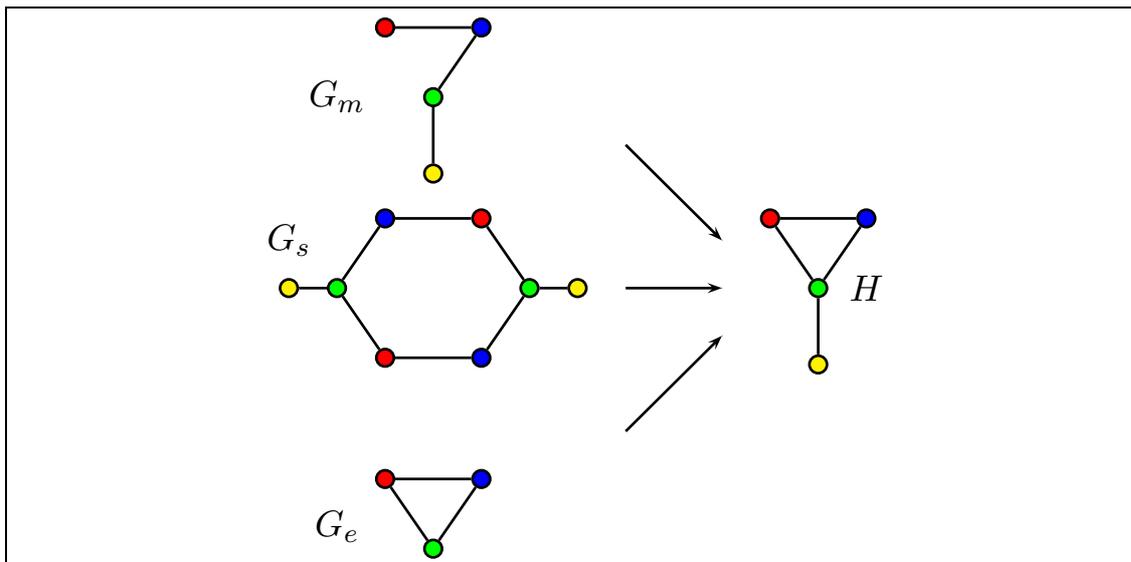
Figure 3.2: Monomorphism, surjective homomorphism, embedding.

Figure 3.2 shows a monomorphism (G_m to H), a surjective homomorphism (G_s to H) and an embedding (G_e to H).

Let G and H be directed graphs, there is a monomorphism from G to H if and only if G is a subgraph of H, there is an embedding from G to H if and only if G is an induced subgraph of H. Examples can be found in Section 2.1.1 on page 12.

---

[1]The name 'monomorphism' is from category theory.

Surjective homomorphisms

Example 3.2.1. Here are some examples of vertex or edge surjective homomorphisms.

1. The mapping from Frucht graph to K_3 shown in Figure 3.1 is a surjective homomorphism;

2. The homomorphism from P_3 (Figure 3.3)to K_3 by mapping $f(x_i) = v_i$ for $i = 1, 2, 3$ is vertex surjective but not edge surjective (because the preimage of edge (v_2, v_3) is empty);

3. The homomorphism from to P_3 to G shown in Figure 3.3 by mapping $f(x_1) = v_2$, $f(x_2) = v_3$ and $f(x_3) = v_2$ is edge surjective bot not vertex surjective (because the preimage of v_1 is empty).
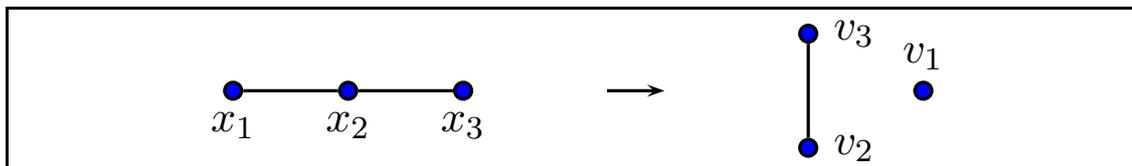
Figure 3.3: Example of an edge surjective but not vertex surjective homomorphism.

Note that a homomorphism that is vertex injective is also edge injective (but not conversely), and, as long as the resulting graph has no isolated vertices, a homomorphism that is edge surjective is also vertex surjective (but not conversely). In other words, injective homomorphisms are the same as vertex injective homomorphisms, while surjective homomorphisms are, in the absence of isolated vertices, the same as edge surjective homomorphisms [40].

QUOTIENT GRAPHS AND HOMOMORPHIC IMAGES

Recall that given an arbitrary partition P of V_G, with nonempty parts V_i, $i \in I$, we define the quotient graph of G induced by partition P to be the graph G/P with vertex set $\{V_i \mid i \in I\}$, in which there is a loop $(V_i, V_i) \in E_{G/P}$ if V_i contains some u, v with $(u, v) \in E_G$, and in which $(V_i, V_j) \in E_{G/P}$ if some $(u, v) \in E_G$ has $u \in V_i$, $v \in V_j$. Then the canonical mapping that assigns to each $u \in V_G$ the unique i such that $u \in V_i$ is a homomorphism $f : G \to G/P$, it is actually surjective.

For example, if we take the partition $P = \{V_1, V_2, V_3\}$ by different colours of Frucht graph in Figure 2.11, i.e. $V_1 = \{1, 3, 8, 11\}$ (red vertices), $V_2 = \{2, 5, 7, 9, 10\}$ (blue vertices), $V_3 = \{4, 6, 12\}$ (green vertices), then the quotient graph of Frucht graph induced by P is K_3.

Similarly to the quotient graph, there is another concept 'homomorphic image' which is also closely connected to surjective homomorphisms. Let G and H be directed graphs, and f a homomorphism of directed graph G to directed graph H. We define the *homomorphic image of G under f*, denoted by $f(G)$, as the directed graph with vertices $\{f(v) \mid v \in V_G\}$ and for which there is an edge $(f(v), f(w))$ if and only if $v, w \in E_G$.

Note that $f(G)$ is a subgraph of H, and that $f : G \to f(G)$ is a surjective homomorphism. Conversely, if $f : G \to H$ is a surjective homomorphism, then $H = f(G)$.

There is a one-to-one correspondence in between quotient graphs and homomorphic images.

Proposition 3.2.1. *Every quotient of G is a homomorphic image of G, and conversely, every homomorphic image of G is isomorphic to a quotient of G.*

Proof. Suppose G/P is a quotient graph of G with partition $P = \{V_1, V_2, \ldots, V_k\}$, then we assign the canonical mapping that assigns to each vertex $u \in V_G$ the unique $i \in \{1, 2, \ldots, k\}$ such that $u \in V_i$, $G/P = f(G)$.

For a homomorphic image $f(G)$, we assign an equivalence class by $u \sim v$ if $f(u) = f(v)$ for any $u, v \in V_G$. It is easily seen that for the partition P induced by $\sim$, G/P is isomorphic to $f(G)$. $\square$

Now we come to the proof of the decomposition law.

Proposition 3.2.2 (Hell, Nešetřil [40]). *Let G and H be any directed graphs. Every homomorphism $f : G \to H$ can be written as $f = i \circ s$ where s is a surjective homomorphism and i is an injective homomorphism.*

Proof. Recall that the image graph $f(G)$ is a graph with vertex set $f(V_G)$ and $u, v \in f(V_G)$ is an edge in $f(G)$ if and only if there exist $x, y \in V_G$ and $f(x) = u$, $f(y) = v$. $f(G)$ is a subgraph of H, thus there is a natural injective homomorphism i from $f(G)$ to H. On the other hand, f induces a quotient of G with respect to partition θ_f. Thus we can take s to be canonical surjective homomorphism of G onto $G/\theta_f = f(G)$, and i the inclusion homomorphism of $f(G)$ in H. $\square$

Full homomorphisms

Consider the surjective homomorphism from the Frucht graph to K_3 shown in Table 3.1. There is no edge for example between vertices 1 and 7 in the Frucht graph, but there is an edge between their images Red and Blue in K_3. This is because surjective homomorphisms do not necessarily preserve non-adjacency. In this part we discuss full homomorphisms which preserve both edges and non-edges.
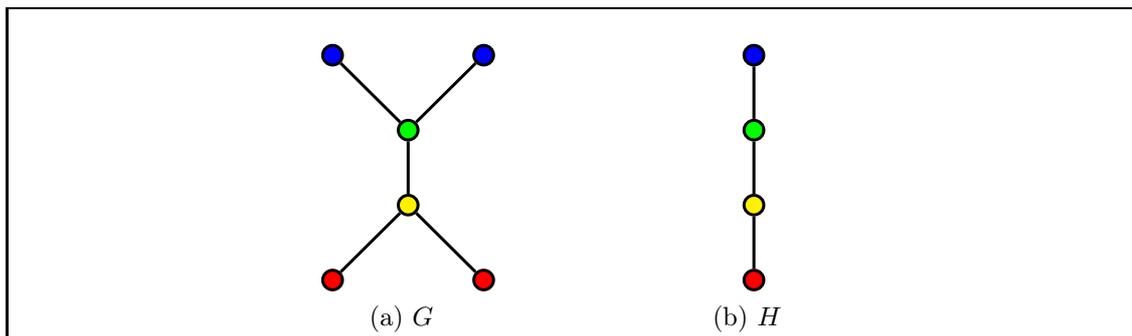
Figure 3.4: A full homomorphism from G to H.

Example 3.2.2. There is a full homomorphism from G to H as in Figure 3.4, by mapping the vertices in graph G to the vertex in the corresponding colour in graph H.

To our knowledge, the concept of full homomorphisms first appeared in 1966 in [33]. It has a close connection to other variants of graph homomorphisms, for example, a full homomorphism is a surjective homomorphism in the sense of absence of isolated vertices in H. A homomorphism is an embedding if it is both a monomorphism and a full homomorphism.

Observation 3.2.3 ([40]). *There are some easy facts regarding to full homomorphisms:*

1. *If G is an induced subgraph of H then the inclusion mapping $i : G \to H$ is a full injective homomorphism.*

2. *Full injective homomorphisms are embeddings.*

3. *There exists a full homomorphism $f : G \to H$ if and only if the vertices of G can be partitioned into independent sets S_x, $x \in V_H$, such that if $xy \notin E_H$ then no edge of G joins the set S_x to the set S_u, and if $xy \in E_H$ then all vertices of S_x are joined in G to all vertices of S_y.*

Recall that graph homomorphism is a generalization of colouring, while full homomorphism is closely connected to another colour-related concept, 'the achromatic number' [37].

POINT-DETERMINING GRAPHS

Full homomorphisms are also closely related to the notion of a point-determining graph, which was introduced by D. Sumner [78] and others in the 1970s.

Definition 3.2.1. *Point-determining graphs* (also known as *mating-type graphs, mating graphs, M-graph* or *thin graphs* [32]) are graphs in which no two vertices have the same open neighbourhood.

Example 3.2.3. In Fig. 3.4, the left graph is not a point-determining graph while the right graph is a point-determining graph. Moreover, there is a full homomorphism from the left graph to the right graph.

Definition 3.2.2. The *thinness relation S* of G is the equivalence relation on V_G defined by $(x, y) \in S$ if and only if $N_G(x) = N_G(y)$. We denote by $\mathcal{S}$ the corresponding partition of V_G, and write $R_S \subseteq V_G \times \mathcal{S}$ for the relation that associates each vertex with its S-equivalence class, i.e., $(x, \beta) \in R_S$ if and only if $x \in \beta$.

The *point-determining graph of G*, (also called *thin graph of G*), denoted by G_{pd}, is a graph with vertex $\mathcal{S}$, two equivalence classes σ and τ of S are adjacent in G_{pd} if and only if (x, y) is an edge of G with $x \in \sigma$ and $y \in \tau$.

Note that R_S is a full homomorphism of G to G_{pd} [19]. Also G_{pd} is indeed a point-determining graph [32]. We observe that G is obtained from G_{pd} by vertex duplication. Thinness and the quotients w.r.t. the thinness relation play a role in particular in the context of product graphs, see [46]. In this context it is well known that G can be reconstructed from G_{pd} and knowledge of the S-equivalence classes.

If there is a full homomorphism from graph G to graph H, then there is a point-determining graph K such that both G and H have a full homomorphism to K. The correspondence between full homomorphisms and point-determining graphs is explored in greater detail in [19]. In the following we summarize the results about point-determining graphs contained in [78].

Lemma 3.2.4 (Sumner [78]). *Let G be point-determining. If $N_G(a) = N_G(b) - c$ with $a \neq b$ and $N_G(d) = N_G(e) - a$ with $d \neq e$, then $d = c$.*

Proof. Suppose that $d \neq c$. Since $N_G(d) = N_G(e) - a$ we have e and a are adjacent, because $N_G(a) = N_G(b) - c$, then $e \in N_G(a) \subset N_G(b)$. Hence $b \in N_G(e) - \{a\} = N_G(d)$. Since $d \neq c$, $d \in N_G(b) - \{c\} = N(a)$. So $a \in N_G(d) = N_G(e) - \{a\}$, a contradiction. $\square$

Proposition 3.2.5 (Sumner [78]). *If G is nontrivial and point-determining, then there exists a vertex $x \in G$ such that $G - \{x\}$ is point-determining.*

Proof. Suppose no such point exists. Let $x \in G$ be the vertex with largest degree. Then since graph $G - \{x\}$ is not point-determining, there exist vertices $p, q \in G - \{x\}$ with $p \neq q$ and $N_G(p) = N_G(q) - \{x\}$. Similarly there exist vertices

$r, s \in G - \{p\}$ with $r \neq s$ such that $N_G(r) = N_G(q) - \{x\}$, by Lemma 3.2.4, $r = x$, so $N_G(x) = N_G(s) - \{p\}$ and thus $d(x) = d(s) - 1$, so that $d(s) > d(x)$, which is a contradiction. $\square$

3.3 Locally constrained homomorphisms

In this section we introduce three kinds of locally constrained homomorphisms: locally injective, surjective and bijective homomorphism. Locally constrained homomorphisms play a crucial rule in many application areas. A comprehensive survey [22] is available summarizing the most important results about locally constrained homomorphisms in structural, algorithmic and applicational aspects.

Definition 3.3.1. For undirected graphs,

1. A homomorphism $f : G \to H$ is a *locally injective homomorphism*, or *LI-morphism*, if for all $v \in V_G$, the restriction of the mapping f to the domain $N_G(v)$ and range $N_H(f(v))$ is injective.

2. A homomorphism $f : G \to H$ is a *locally surjective homomorphism*, or *LS-morphism*, if for all $v \in V_G$, the restriction of the mapping f to the domain $N_G(v)$ and range $N_H(f(v))$ is surjective.

3. A homomorphism $f : G \to H$ is a *locally bijective homomorphism*, or *LB-morphism*, if it is both a locally surjective and locally bijective homomorphism. That is if the restrictions to the neighbourhoods are the bijections.

There is an alternative and quite natural definition given in [25]:

1. A homomorphism $f : G \to H$ is a *locally injective homomorphism*, if for every edge (u, v) of H, the subgraph of G induced by $f^{-1}(u) \cup f^{-1}(v)$ is a matching.

2. A homomorphism $f : G \to H$ is a *locally surjective homomorphism*, if for every edge (u, v) of H, the subgraph of G induced by $f^{-1}(u) \cup f^{-1}(v)$ is a bipartite graph without isolated vertices.

3. A homomorphism $f : G \to H$ is a *locally bijective homomorphism*, if for every edge (u, v) of H, the subgraph of G induced by $f^{-1}(u) \cup f^{-1}(v)$ is a perfect matching.

In accordance with the notation used for ordinary graph homomorphisms we write $G \to^{LI} H$, $G \to^{LS} H$, $G \to^{LB} H$ for the existence of locally injective, surjective and bijective homomorphisms.

Compared to globally constrained homomorphisms, locally constrained homomorphisms have their own interesting properties and applications. Figure 3.5 shows examples of locally injective (G_i to H), locally bijective (G_b to H), locally surjective homomorphism (G_s to H). It is easily seen that a locally injective (locally bijective) homomorphism is not necessarily an injective (bijective) homomorphism, but a locally surjective homomorphism between two connected graphs is a surjective homomorphism.
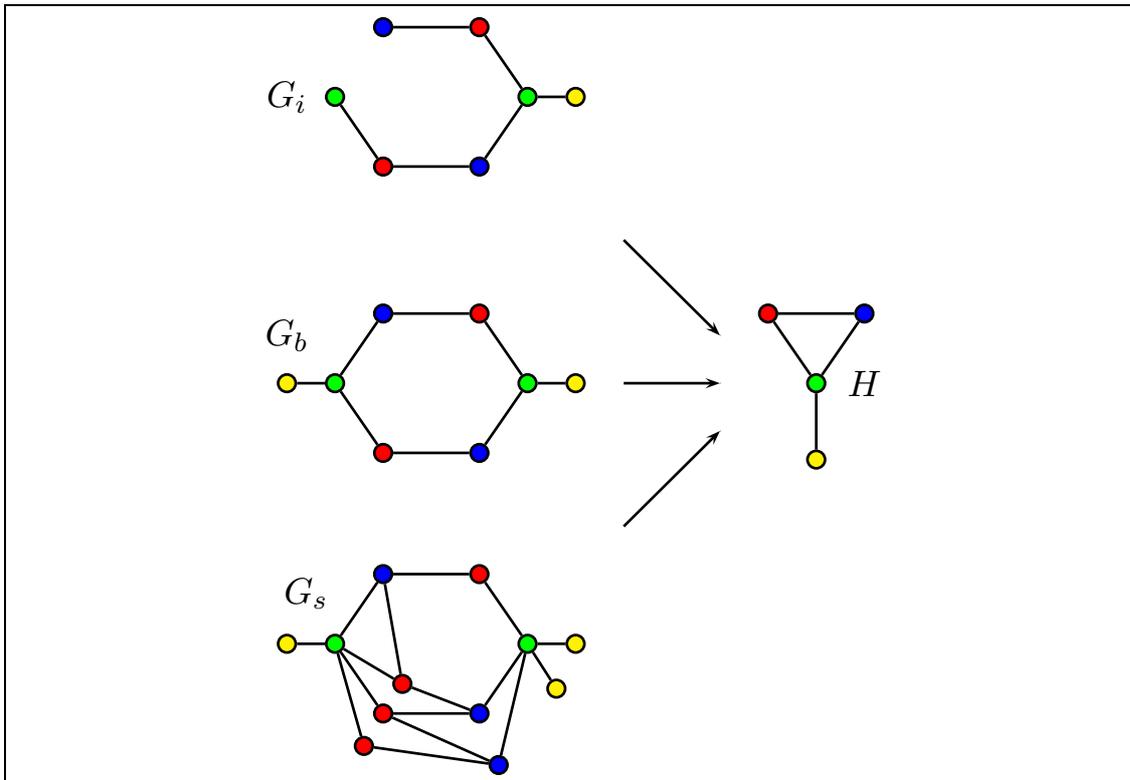
Figure 3.5: Locally injective, bijective, surjective homomorphisms [22].

Locally surjective homomorphisms were firstly introduced by Everett and Borgatti, who called them *role colourings* [18]. They originated in the theory of social behaviour. The source graph G represents relations between particular individuals of some group. The target graph H is called the role graph. The task is to assign roles to individuals so that each person given a particular role has, among its neighbours, every role prescribed by the role graph at least once, while no other roles may appear in the neighbourhood [22], where individuals of the same social role relate to other individuals in the same way [22]. In this thesis,

the other two kinds of locally constrained homomorphisms are of more interest to us, and so we accordingly introduce them in greater detail.

Locally injective homomorphisms

The notion of locally injective homomorphism, sometimes called *local monomorphism* [63] or *partial covering projection* [21], is a notion with an interesting history and wide application. It has a very close connection to the frequency assignment problem, which is a highly practical problem of interference-free frequency assignment for wireless networks [22]. Locally injective homomorphisms (also called local epimorphisms or partial covers) are also used in distance-constrained labelings of graphs [21] and as indicators of the existence of homomorphisms of derivative graphs (line graphs) [63].

Observe that if there is a locally injective homomorphism from graph G to graph H, then for any vertex $v \in V_G$, $|N(v)| \leq |N(f(v))|$. Consequently, a locally injective homomorphism from a graph to itself preserves degrees, i.e., if f is a locally injective homomorphism from a graph G to itself, then for any vertex $v \in V_G$, $d(v) = d(f(v))$. There is a classical result about locally injective homomorphisms from a graph to itself which was proved in 1971:

Theorem 3.3.1 (Nešetřil [63]). *Let G be a connected finite graph. Then every locally injective homomorphism of G into itself is an automorphism.*

Proof. We prove the statement by induction on $|G|$. It is easily seen that for $|G| = 2$ the statement of the theorem holds. Let $|G| > 2$, and suppose that the statement of the theorem holds for all connected graphs H with $|H| < |G|$. We assume there is a locally injective homomorphism f from G to itself which is not an automorphism, then $f(V_G) \subset V_G$ and the homomorphic image $f(G)$ is a connected graph. Take a vertex $v \in V_G \setminus f(V_G)$ and consider the graph $G - v$. Let $G^1, \ldots, G^n$ be the disjoint connected components of $G - v$ and, because G is a connected graph, for each G^i, there exists a vertex v^i which is connected to v. Suppose the image of v is in G^j where $1 \leq j \leq n$, then from the properties of locally injective homomorphism, for all i, $f(v^i) \in G^j$. Because of the connectivity, $f(G^i) \subset G^j$ for any i. It is easily seen that the restriction of f to G^j is also a locally injective homomorphism, from the induction hypothesis it is an automorphism. Consider any vertex w in the set of $N_G(v) \cap G^j$, then $f(N_{G^j}(w)) = N_{G^i}(w)$. But $v \in V_G(w)$ and $v \notin f(N_G(w))$, then f is not a locally injective homomorphism of G to itself, which is a contradiction. □

Locally bijective homomorphisms

A locally bijective homomorphism is also locally injective and locally surjective. Hence, any result valid for locally injective or locally surjective homomorphisms is also valid for locally bijective homomorphisms. Locally bijective homomorphisms (also known as *local isomorphisms* or *full covering projections* [21]) have important applications in many areas, for example distributed computing, recognizing graphs by networks of processors or constructing highly transitive regular graphs [25].

For locally bijective homomorphisms the preimage classes of all vertices have the same size and for locally surjective homomorphisms all the preimage classes have size at least one. If there is a locally bijective homomorphism from graph G to graph H, then either $|V_G| > |V_H|$ or else G and H are isomorphic.

Just as for graph isomorphism, a locally bijective homomorphism maintains vertex degrees and degrees of neighbours and degrees of neighbours of neighbours and so on. The existence of such a mapping from G to H therefore implies equality of the so-called *degree refinement matrices* of G and H.

Degree decomposition matrices

Consider a locally bijective homomorphism $f : G \rightarrow^{LB} H$. By definition, for every vertex x, f induces an isomorphism from the neighbourhood of x, $N_G(x)$, to the neighbourhood of $f(x)$, $N_H(f(x))$. In particular x and $f(x)$ have the same degree. (For every vertex $x' \in N_G(x)$ the neighbourhood of x' must be isomorphic to the neighbourhood of $f(x')$ that itself lies in the neighbourhood of $f(x)$.) It follows that locally bijective homomorphisms preserve not only the degrees of vertices, but also the degrees of neighbourhood vertices and so on. This property is well captured by the notion of degree refinement matrices [25], which we have already defined in Section 2.1.4.

Recall a partition of the vertex set of a graph G into disjoint classes is an *equitable partition* if the vertices in the same class have the same numbers of neighbours in all classes of the partition.

The relation 'being finer' defines a lattice on the set of all equitable partitions of a fixed graph. Every finite graph admits a unique minimal equitable partition [18].

Any equitable partition is characterized by the associated *degree decomposition matrix* whose rows and columns are indexed by the blocks of the partition, and the entry in the i-th row and j-th column describes how many neighbours a vertex from the i-th block has in the j-th block.

Every finite graph G admits a unique minimal equitable partition. In this case a canonical ordering can be imposed on the blocks, so the corresponding degree matrix, called the *degree refinement matrix*, drm(G), is also defined uniquely.

We put $G \sim^M H$ if and only if $\mathrm{drm}(G) = \mathrm{drm}(H)$. The relation $\sim^M$ is an equivalence relation on the class of finite connected graphs, ConnGraph.

It is easy to observe that given a pair of graphs (G, H) with $\mathrm{drm}(G) \neq \mathrm{drm}(H)$ there is no locally bijective homomorphism $G \to^{LB} H$. Consequently the relation $\to^{LB}$ is a sub-relation of $\sim^M$. The connections between $\sim^M$ and all three variants of locally constrained homomorphisms are captured by the following so-called Cantor-Bernstein type theorems.

Theorem 3.3.2 (Fiala, Kratochvíl [21]). *If two graphs G and H share the same degree refinement matrix, then every locally injective homomorphism from G to H is locally bijective.*

Theorem 3.3.3 (Kristiansen, Telle [51]). *If two graphs G and H share the same degree refinement matrix, then every locally surjective homomorphism from G to H is locally bijective.*

It is not necessarily the case that there is a locally bijective homomorphism between two graphs with the same degree refinement matrix, for example, C_3, C_4 and C_5 share the same degree refinement matrix $\{2\}$, but there are no locally bijective homomorphisms between them.

Because the degree refinement matrix is easy to compute, for many pair of graphs (G, H) we can easily determine that a locally bijective homomorphism from G to H does not exist.

UNIVERSAL COVERS

The notion of *universal cover* is well established in the topology of continuous spaces. One requirement of a universal cover is that it be simply connected, which translates in terms of graph theory to the requirement of being acyclic and connected, that is a tree. In the discrete case, formally, the *universal covering graph* (or *universal cover*) of a graph H is the only (possibly infinite) tree T_H that admits a locally bijective homomorphism to H.

If G is a tree, then G itself is the universal covering graph of G. The universal covering graph of other connected graph is a countably infinite (but locally finite) tree.

The universal covering graph T_G of a connected graph G can be constructed as follows[1]. Choose an arbitrary vertex r of G as a starting point. Each vertex of T_G is a non-backtracking walk that begins from r, that is, a sequence $w = (r, v_1, v_2, ..., v_n)$ of vertices of G such that v_i and v_{i+1} are adjacent in G and $v_{i-1} \neq$

---

[1]This construction is from *Covering graph. (2012, September 3). In Wikipedia, The Free Encyclopedia. Retrieved 09:51, February 20, 2014, from http://en.wikipedia.org/w/index.php?title=Covering_graph&oldid=510586814.*

v_{i+1} for all i. Then, two vertices of T_G are adjacent if one is a simple extension of another: the vertex $(r, v_1, v_2, ..., v_n)$ is adjacent to the vertex $(r, v_1, v_2, ..., v_{n-1})$. Up to isomorphism, the same tree T_G is constructed regardless of the choice of the starting point r. Thus the universal covering graph is unique up to isomorphism.

The covering map f maps the vertex (r) in T_G to the vertex r in G, and a vertex $(r, v_1, v_2, ..., v_n)$ in T to the vertex v_n in G.

Figure 3.6 gives an example of universal covering graph.
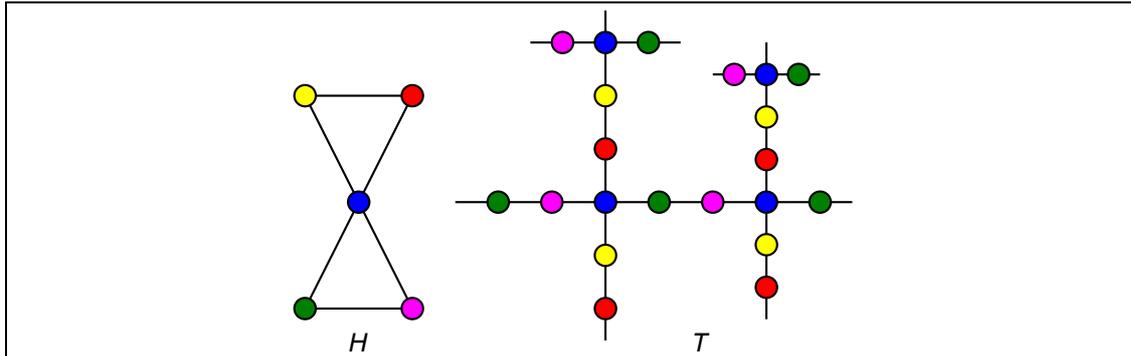
Figure 3.6: H and its universal covering graph.

Suppose f is a homomorphism from graph G to graph H. If f is locally injective, then the tree T_G is a subtree of T_H; if f is locally surjective, then T_H is a subtree of T_G; if f is locally bijective, then T_G and T_H are isomorphic.

The following theorem shows the connection between the degree refinement matrices and universal covers.

Theorem 3.3.4 (Leighton [55]). *Two graphs G and H share the same degree refinement matrix if and only if their universal covers are isomorphic.*

3.4 Relations

In the next chapter we will discuss the theory of relations in more detail.

Definition 3.4.1. Let $G = (V_G, E_G)$ be a graph, B a finite set, and $R \subseteq V \times B$ a binary relation, where for every element $b \in B$, we can find an element $v \in V_G$ such that $(v, b) \in R$. Then the graph $G * R$ has vertex set B and edge set

$$E_{G*R} = \{(a, b) \in B \times B \mid \text{ there is } (u, v) \in E_G \text{ such that } (u, a), (v, b) \in R\}.$$

Graphs with loops are not always a natural model, however, so that it may appear more appealing to consider the slightly modified definition.

Definition 3.4.2. Let $G = (V_G, E_G)$ be a simple graph, B a finite set, R a binary relation, where for every element $b \in B$, we can find an element $v \in V_G$ such that $(v, b) \in R$. Then the (simple) graph $G \star R$ has vertex set B and edge set

$$E_{G \star R} = \{(a, b) \in B \times B \mid a \neq b \text{ and there is } (u, v) \in E_G \text{ and } (u, a), (v, b) \in R\}.$$

We remark that these definitions remain meaningful for directed graphs, weighted graphs (where the weight of an edge is the sum of the weights of its pre-images) as well as relational structures. For simplicity, we restrict ourselves to undirected graphs (with loops). Most of the results can be directly generalized.

Graphs can be regarded as representations of symmetric binary relations. Using the same symbol for the graph and the relation it represents, we may re-interpret definition 3.4.1 as a conjugation of relations. R^+ is the *transpose* of R, i.e., $(u, x) \in R^+$ if and only if $(x, u) \in R$. The double composition $R^+ \circ G \circ R$ contains the pair (u, v) in $B \times B$ if and only if there are x and y such that $(u, x) \in R^+$, $(y, v) \in R$, and $(x, y) \in E_G$. Thus

$$G * R = R^+ \circ G \circ R.$$

Simple graphs, analogously, correspond to the irreflexive symmetric relations. For any relation R, let R^ι denote its irreflexive part, also known as the *reflexive reduction* of R. Since Definition 3.4.2 explicitly excludes the diagonals, it can be written in the form

$$G \star R = (R^+ \circ G \circ R)^\iota.$$

We have $G \star R = (G * R)^\iota$, and hence $E_{G \star R} \subseteq E_{G * R}$. The composition $G * R$ is of particular interest when G is also a simple graph, i.e., $G = G^\iota$.

If there is an R such that $G * R = H$ holds, we say there is a *relation* from G to H. If there is an R such that $G \star R = H$ holds, we say there is a *weak relation* from G to H.

The main part of this contribution will be concerned with the solutions of the equation $G * R = H$. The weak version, $G \star R = H$, will turn out to have much less convenient properties, and will be discussed only briefly in Section 4.6.

Throughout this thesis we use the following standard notations and terms.

For relation $R \subseteq X \times Y$ we define by $R(x) = \{p \in Y \mid (x, p) \in R\}$ the *image of x under R* and $R^{-1}(p) = \{x \in X \mid (x, p) \in R\}$ the *pre-image of p under R*.

The *domain* of R is defined by $\operatorname{dom} R = \{x \in X \mid \exists p \in Y \text{ s.t. } (x, p) \in R\}$, and the *image* of R is defined by $\operatorname{img} R = \{p \in Y \mid \exists x \in X \text{ s.t. } (x, p) \in R\}$. We say that the *domain of R is full* if for any $x \in X$ we have $R(x) \neq \emptyset$. Analogously, the *image is full* if for any $p \in Y$ we have $R^{-1}(p) \neq \emptyset$.

If $R \subseteq X \times Y$ is a binary relation, then R is *injective*, if for all x and z in X and y in Y it holds that if $(x, y) \in R$ and $(z, y) \in R$ then $x = z$. R is *functional*,

if for all x in X, and y and z in Y it holds that if $(x, y) \in R$ and $(x, z) \in R$ then $y = z$.

We denote by I_G the *identity map on G*, i.e., $\{(x, x) \mid x \in V_G\}$.

Matrix multiplication

The operation $*$ can also be formulated in terms of matrix multiplication. To see this, consider the following variant of the operation on weighted graphs.

Definition 3.4.3. If G is a weighted graph, we use $w(x, y)$ to denote the weight between x and y. Given a finite set B and a binary relation $R \subseteq V_G \times B$, $G \circledast R$ is defined as a weighted graph H with vertex set B, for any $u, v \in B$, $w(u, v) = \sum_{(x,u) \in R, (y,v) \in R} w(x, y)$.

Ignoring the weights, the operations $*$ and $\circledast$ are equivalent.

Using the language of matrices, $G \circledast R = H$ can be interpreted as matrix multiplication:

$$\mathbf{W}_{G \circledast R} = \mathbf{R}^+ \mathbf{W}_G \mathbf{R}$$

where $\mathbf{R}$ is the matrix representation of the relation R, i.e., $\mathbf{R}_{xu} = 1$ if and only if $(x, u) \in R$, otherwise $\mathbf{R}_{xu} = 0$, $\mathbf{R}^+$ denotes the transpose of $\mathbf{R}$, and $\mathbf{W}_G$ is the matrix of edge weights of G.

Homomorphisms and multihomomorphisms

The notion of relations between graphs is in many ways similar (but not equivalent) to the well-studied notion of graph homomorphism. The majority of our results focus on similarities and differences between these two concepts.

Recall that a *(strong) homomorphism* from a graph G to a graph H is a mapping $f : V_G \to V_H$ such that for every edge (x, y) of G, $(f(x), f(y))$ is an edge of H. Note that strong homomorphisms require loops in H whenever $(x, y) \in E_G$ and $f(x) = f(y)$. In contrast, we define f is a *weak homomorphism* if $(x, y) \in E_G$ implies that either $f(x) = f(y)$ or $(f(x), f(y)) \in E_H$. Every strong homomorphism from G to H induces also a weak homomorphism, but not conversely. Weak homomorphisms play an important role in graph product, see [46].

A map $f : V_G \to V_H$ is, of course, a special case of a relation. This is seen by setting $F = \{(x, f(x)) \mid x \in V_G\}$. Hence, there is a strong surjective homomorphism from G to H if and only if there is a functional relation F such that $G * F = H$. Another important connection to the graph homomorphisms is the following simple lemma.

Lemma 3.4.1. *If $G * R = H$, and the domain of R is full, then there is a homomorphism f from G to H contained in R.*

Proof. If $G * R = H$, then take any functional relation $f \subseteq R$, we have $G * f \subseteq H$, where f is a homomorphism from G to H. $\qquad\square$

Analogously, there is a *weak surjective homomorphism* from G to H if and only if there is a functional relation F such that $G \star F = H$, and there is a weak homomorphism from G to H if there is a functional relation $F \subseteq V_G \times V_H$ such that $G \star F$ is a subgraph of H. The existence of relations between graphs thus can be seen as a proper generalization of weak or strong graph homomorphisms, respectively.

Relations between graphs can be regarded also as a variant of multihomomorphisms. Multihomomorphisms are building blocks of Hom-complexes, introduced by Lovász, and are related to recent exciting developments in topological combinatorics [56], in particular to deep results involved in proof of the Lovász hypothesis [2].

A *multihomomorphism* $G \to H$ is a mapping $f : V_G \to 2^{V_H} \setminus \{\emptyset\}$ (i.e., associating a nonempty subset of vertices of H with every vertex of G) such that whenever $\{u_1, u_2\}$ is an edge of G, we have $(v_1, v_2) \in E_H$ for every $v_1 \in f(u_1)$ and every $v_2 \in f(u_2)$.

Functions from vertices to sets can be seen as a representation of relations. A relation with full domain thus can be regarded as *surjective multihomomorphism*, a multihomomorphism such that pre-image of every vertex in H is non-empty and for every edge (u, v) in H we can find an edge (x, y) in G satisfying $u \in f(x)$, $v \in f(y)$. Thus we call a relation with full domain also *R-morphism*.

Examples

Similarly to graph homomorphisms, the equation $G * R = H$ (or $G \star R = H$ respectively) may have multiple solutions for some pairs of graphs (G, H), while there may be no solution at all for other pairs.

As an example, consider K_2 (two vertices x, y connected by an edge) and C_3 (a cycle of three vertices u, v, w). Denote $R_1 = \{(u, x), (v, y)\}$, $R_2 = \{(v, x), (w, y)\}$, $R_3 = \{(w, x), (u, y)\}$, then it is easily seen that $C_3 * R_i = K_2$ for each $1 \leq i \leq 3$, i.e., the equation $C_3 * R = K_2$ has more than one solution.

On the other hand, there is no relation R such that $K_2 * R = C_3$. Otherwise, each vertex of C_3 is related to at most one vertex of K_2, since C_3 is loop free; hence there exists a vertex in K_2 which has no relations to at least two vertices in C_3, w.l.o.g., one can assume $(x, u), (x, v) \notin R$; then the definition of $*$ implies that there is no edge between u and v, which leads to a contradiction.

Because relations do not need to have full domain (unlike graph homomorphisms), there is always a relation from a graph G to its induced subgraph $G[W]$.

Relations with full domain are not restricted to surjective homomorphisms. As a simple example, consider paths P_2 with vertex set $\{x, y\}$ and P_3 with vertex set $\{u, v, w\}$, respectively, and set $R = \{(x, u), (x, w), (y, v)\}$. One can easily verify $P_2 * R = P_3$ by direct computation. Here, R is not functional since x has two images.

Note that, quite surprisingly, there is no relation R satisfying $P_3 * R = P_4$. In fact (P_2, P_3) is the only pair of paths such that there is a relation from one to another and vice versa. This will be further explained in Section 4.1.2.

R-Homomorphisms (Relations)

This chapter is organized as follows:

In Section 4.1 the basic properties of strong relations between graphs are compiled. It is shown that relations compose and every relation can be decomposed in a standard way into a surjective and an injective relation (Corollary 4.1.3). We discuss some structural properties of graphs preserved by relations.

Equivalence on the class of graphs induced by the existence of relations between graphs is the topic of section 4.2. We consider two forms: strong relational equivalence, where relations are required to be reversible, and weak relational equivalence. The equivalence classes of strong relational equivalence are characterized in Theorem 4.2.4. To describe the equivalence classes of weak relational equivalence we introduce the notion of an R-core of a graph and show that it is in many ways similar to the more familiar construction of the graph core (Theorem 4.2.11). We explore in particular the differences between core and R-core and provide an effective algorithm to compute the R-cores of given graphs.

Section 4.3 is concerned with the partial order induced on relations between two fixed graphs G and H by inclusion. Focusing on the special case $G = H$, we describe the minimal elements of this partial order. In Theorem 4.3.7 we give a, perhaps surprisingly simple, characterization of those graphs whose relations to themselves are automorphisms.

R-retractions are defined in Section 4.4 in analogy to retractions. This naturally gives rise to a notion of R-reduced graphs that we show coincides with the concept of graph cores. By reversing the direction of relations, however, we obtain the concept of a cocore of a graph, which does not have a non-trivial counterpart

in the world of ordinary graph homomorphisms, and we explore its properties. Finally, we give a full list of the inclusion relations between these notions and give examples to distinguish them.

The computational complexity of testing for the existence of a relation between two graphs is briefly discussed in Section 4.5. In Theorem 4.5.1 we describe the reduction of this problem to the surjective homomorphism problem.

Finally, in Section 4.6 we briefly summarize the most important similarities and differences between weak and strong relational compositions.

4.1 Basic properties

Composition

Recall that the composition of binary relations is associative, i.e., suppose $R \subseteq W \times X$, $S \subseteq X \times Y$, and $T \subseteq Y \times Z$. Then $R \circ (S \circ T) = (R \circ S) \circ T$. Furthermore, transposition of relations satisfies $(R \circ S)^+ = S^+ \circ R^+$. Interpreting a graph G as a relation on its vertex set, we easily derive the following identities.

Lemma 4.1.1 (Composition law). $(G * R) * S = G * (R \circ S)$.

Proof. $(G * R) * S = S^+ \circ (R^+ \circ G \circ R) \circ S = (S^+ \circ R^+) \circ G \circ (R \circ S)$
$$= (R \circ S)^+ \circ G \circ (R \circ S) = G * (R \circ S). \qquad \square$$

Now we show that every relation R can be decomposed, in a standard way, to a relation R_D duplicating vertices and a relation R_C contracting vertices.

Lemma 4.1.2. *Let $R \subseteq X \times Y$ be a relation. Then there exists a subset A of X, a set B, an injective relation with full domain $R_D \subseteq A \times B$ and a functional relation $R_C \subseteq B \times Y$, such that $R = I_A \circ R_D \circ R_C$, where I_A is the identity on X restricted to A.*

Proof. Put $A = \operatorname{dom} R$. Then the relation I_A removes vertices in $X \setminus \operatorname{dom} R$. Therefore, it remains to show, that any relation $R \subseteq X \times Y$ with full domain can be decomposed into an injective relation $R_D \subseteq X \times B$ and a functional relation $R_C \subseteq B \times Y$. To see this, set $B = R$ and declare $(x, \alpha) \in R_D$ if and only if $\alpha = (x, p) \in R$ for some $p \in Y$, and $(\beta, q) \in R_C$ if and only if $\beta = (y, q) \in R$ for some $y \in X$. By construction R_D is injective and R_C is functional. Furthermore, $(x_0, p_0) \in R_D \circ R_C$ if and only if there is an $\alpha \in R$ that is simultaneously of the form (x_0, p) and (x, p_0), i.e., $x = x_0$ and $p = p_0$. Hence $(x_0, p_0) \in R$. $\qquad \square$

Note that this decomposition is not unique. For instance, we could construct B from multiple copies of R. More precisely, let $B = R \times \{1, 2, \ldots, k\}$, then we would set $\big(x, (\alpha, i)\big) \in R_D$ $(1 \leq i \leq k)$ if and only if $\alpha = (x, p) \in R$ for some $p \in Y$, etc.

The set B as constructed in the proof of Lemma 4.1.2 has minimal size. To see this, it suffices to show that, given B there is a mapping from B onto R. Since R_D is injective and R_C is functional we may set

$$\alpha \in B \mapsto (R_D^{-1}(\alpha), R_C(\alpha)).$$

Since $R = I_A \circ R_D \circ R_C$ we conclude that the mapping is surjective, and hence $|B| \geq |R|$.

According to Lemma 4.1.1, the decomposition of R in Lemma 4.1.2 can be restated as follows:

Corollary 4.1.3. *Suppose $G * R = H$. Then there is a set B, an injective relation $R_D \subseteq \operatorname{dom} R \times B$ with full domain, and a surjective relation $R_C \subseteq B \times \operatorname{img} R$ such that $G[\operatorname{dom} R] * R_D * R_C = H$.*

In diagram form, this is expressed as
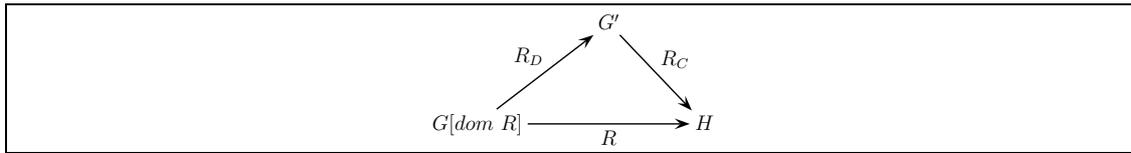
Figure 4.1: Commutative diagram.

We remark that from the fact that relations compose it follows that the existence of a relation implies a quasi-order on graphs that is related to the homomorphism order. This order is studied more deeply in [28].

Structural properties preserved by relations

In this subsection we investigate structural properties of H that can be derived from knowledge about certain properties of G and the fact that there is some relation R such that $G * R = H$.

Connected components

Proposition 4.1.4. *Let $G * R = H$ and denote by $H_1, \ldots, H_k$ the (non-empty) connected components of H. Then there are relations $R_i \subseteq V_G \times V_{H_i}$ such that $G * R_i = H_i$ for each $1 \leq i \leq k$ and $R = \bigcup_{i=1}^{k} R_i$. Furthermore, set $G_i = G[R^{-1}(V_{H_i})]$, then there are no edges between G_i and G_j for arbitrary $i \neq j$.*

Proof. Define the restriction of R to the connected components of H as $R_i = \{(x,y) \in R \mid y \in V_{H_i}\}$. Clearly, R is the disjoint union of R_i and $G * R_i \subseteq H_i$. The definition of $*$ implies $H = G * R = (\bigcup_i R_i)^+ \circ G \circ \left(\bigcup_j R_j\right) = \bigcup_i \bigcup_j R_i^+ \circ G \circ R_j$. Since for $i \neq j$, R_i and R_j relate vertices of G to different connected components of H, we have $R_i^+ \circ G \circ R_j = \emptyset$. It follows that $H = \bigcup_i \bigcup_j R_i^+ \circ G \circ R_j = \bigcup_i R_i^+ \circ G \circ R_i = \bigcup_i G * R_i$. Hence $G * R_i = H_i$.

Any edge between G_i and G_j would generate edges between H_i and H_j, which causes a contradiction to our assumptions. $\square$

Denote by $b_0(G)$ the number of connected components of G, then from Proposition 4.1.4 we arrive at:

Corollary 4.1.5. *Suppose both G and H do not have isolated vertices. If $G * R = H$ and R has full domain, then $b_0(G) \geq b_0(H)$.*

Proof. Our notation is the same as in Proposition 4.1.4. We claim for arbitrary connected component C of graph G, there exists a unique i, such that C is a connected component of G_i. Otherwise one can find two vertices $x, y \in C$, x and y adjacent, such that $x \in V_{G_i}$ and $y \in V_{G_j}$, since G has no isolated vertices, which contradicts $E(G_i, G_j) = \emptyset$. Thus it follows $b_0(G) \geq b_0(H)$. $\square$

From Corollary 4.1.5, we know that H is connected whenever G is connected. Conversely, the connectedness of G, however, cannot be deduced from the connectedness of H. For example, consider $G = P_1 \cup P_1$ with vertex set $\{x_1, x_2, x_3, x_4\}$ and edges $\{x_1, x_2\}$ and $\{x_3, x_4\}$, and $H = P_2$ with vertex set $\{v_1, v_2, v_3\}$. Set $R = \{(x_1, v_1), (x_2, v_2), (x_3, v_2), (x_4, v_3)\}$. One can easily verify that $G * R = H$. On the other hand, H is connected but G has 2 connected components. The point here is, of course, that R is not injective.

COLORINGS

Graph homomorphisms of simple graphs can be seen as generalizations of colourings. Thus, if R is a functional relation describing a vertex colouring, then $G * R \subseteq K_k$. Conversely, if $G * R \subseteq K_k$, where R has full domain, then from Lemma 3.4.1, there exists a homomorphism from G to K_k, which is a colouring of G.

Lemma 4.1.6. *If G is a simple graph and R has full domain, then $\chi(G) \leq \chi(G * R)$.*

Proof. Suppose $G * R = H$ and the domain of R is full, from Lemma 3.4.1 we know $G \to H$, so $\chi(G) \leq \chi(G * R)$. $\square$

DISTANCES

Observation 4.1.7. *If $P_k * R = G$, G is a simple graph and the domain of R is full, P_k with the vertex set $0, 1, \ldots, k$, then there is a walk $[v_0, v_1, \ldots, v_k]$ in G, where $(i, v_i) \in R$ for $0 \le i \le k$.*

Observation 4.1.8. *If $C_k * R = G$, G is a simple graph and the domain of R is full, then there is a closed walk $[v_0, v_1, \ldots, v_{k-1}]$ in G, where $(i, v_i) \in R$ for $0 \le i \le k - 1$.*

Recall that $d_G(x, y)$ denotes the *(canonical) distance* on graph G, i.e., $d_G(x, y)$ is the minimal length of a path in graph G that connects vertices x and y; if there is no path that connects vertices x and y, then the distance is infinite.

Lemma 4.1.9. *Suppose there exists a relation R with full domain s.t. $G * R = H$, $x, y \in V_G$, $u, v \in V_H$ and $(x, u) \in R, (y, v) \in R$. If $x \ne y$, then $d_H(u, v) \le d_G(x, y)$; If $x = y$ and x is not an isolated vertex, then $d_H(u, v) \le 2$.*

Proof. If $x = y$ and x is not isolated, pick a vertex z of graph G which is adjacent to vertex x, and find a vertex $w \in H$ satisfying $(z, w) \in R$. Then $(w, u) \in E_H$ and similarly $(w, v) \in E_H$. So $d_H(u, v) \le 2$.

If $x \ne y$, choose the shortest path $P = x, x_1, x_2, \ldots, x_k, y$ between x and y, and find corresponding vertices $u_1, u_2, \ldots, u_k \in H$ such that $(x_i, u_i) \in R$ for any $1 \le i \le k - 1$, it is easily seen that $(u, u_1) \in E_H$, $(u_i, u_{i+1}) \in E_H$ and $(u_k, v) \in E_H$, then $d(u, v) \le d(x, y)$. $\square$

Hence we immediately get the following corollary about radius.

Corollary 4.1.10. *Suppose $G * R = H$, G and H are connected graphs, and R has full domain, then $rad(H) \le \max\{rad(G), 2\}$.*

An analogous result holds for the diameters. In particular, if G is not a complete graph, then $diam(G) \ge diam(G * R)$. As a result, $k < l$ and $k \ne 1$ implies there is no relation from P_k to P_l. Also, as we state in the example section, (P_1, P_2) is the only pair of paths such that there is a relation from one to another and vice versa.

COMPLETE GRAPHS

Note that in this subsection we do not require that the domain of R is full.

Proposition 4.1.11. *Let H be a simple graph. Then there exists a relation R such that $K_k * R = H$ if and only if H is a complete m-partite graph, where $m \le k$.*

Proof. We denote the vertices of K_k by $1, \ldots, k$.

If H is a complete m-partite graph, let $R = \{(i, u) \mid i = 1, \ldots, m, u \in U_i\}\}$, then for each $1 \le i \le m$, all the vertices in U_i are independent in H, and u is adjacent to v whenever $u \in U_i$ and $v \in U_j$ for distinct i, j. Hence it is easily seen that $K_k * R = H$.

Conversely, if $K_k * R = H$, assume $\operatorname{dom} R = \{1, \ldots, m\}$. We claim that R is injective, otherwise H would have loops. Thus V_H is the disjoint union of $R(1), \ldots, R(m)$. For any two distinct vertices u, v in $R(i)$, u and v are independent in H, then H is a m-partite graph. For distinct i and j every vertex in $R(i)$ are adjacent with every vertex in $R(j)$ whenever $R(i) \ne \emptyset$, which means H is a complete m-partite graph. □

SUBGRAPHS

Relations between graphs intuitively imply relations between local subgraphs. In this section we make this concept more precise. Recall $N_G[x]$ was denoted to be the *closed neighbourhood* of x in G ($N_G[x] = N_G(x) \cup \{x\}$) and $\overline{N_G[x]} := V_G \setminus N_G[x]$ be the set of vertices that are not adjacent (or identical) to x in G. Furthermore, $\overline{G_x} := G - N_G[x]$, which is the induced subgraph of G obtained by removing the closed neighbourhood of a vertex x. Analogously, for a subset $S \subseteq V_G$, $\overline{S}$ is the induced subgraph obtained by removing all vertices in S and their neighbours.

Then we have the following result about relations between local subgraphs.

Proposition 4.1.12. *Suppose $G * R = H$ and $(x, u) \in R$. Suppose there is no isolated vertex in G. If there is no loop in x, then $\overline{G_x} * \widetilde{R} = \overline{G_u}$ where $\widetilde{R} \subseteq \overline{N_G[x]} \times \overline{N_H[u]}$ is the restriction of R.*

Proof. We claim that the pre-image of any vertex in $\overline{H}_u$ is not in $N_G(x)$. Otherwise, there is a vertex $v \in \overline{H}_u$ and a vertex $y \in N_G(x)$ such that $(y, v) \in R$; from $(x, u) \in R$ we deduce that v and u are adjacent, which contradicts to $v \in \overline{H}_u$. Next we claim that there is no vertex in $\overline{H}_u$ whose pre-image is x. Suppose $v \in \overline{H}_u$ and $R^{-1}(v) = \{x\}$. Since v is not isolated in $\overline{H}_u$, there exists a vertex v' which is adjacent with v in $\overline{H}_u$. Thus $R^{-1}(v') \cap N(x) \ne \emptyset$, a contradiction. Hence for any edge (v, v') in $\overline{H}_u$, we can find an edge (y, y') in $\overline{G_x}$ such that $(y, v), (y', v') \in R$. Thus $\overline{G_x} * \widetilde{R} = \overline{G_y}$. □

Example 4.1.1. Let us see an example. Let G and H be graphs as Figure 4.2. For a relation from G to H, we consider the pre-image of vertex u, because the domain of R is full, and the symmetry of P_4, at least either $(1, u) \in R$ or $(2, u) \in R$ holds. If $(1, u) \in R$, from the above theorem, we know that there is an induced relation from P_2 to P_4, but it is impossible. If $(2, u) \in R$ holds, also from above theorem, there is an induced relation from P_1 to P_4, which is impossible.
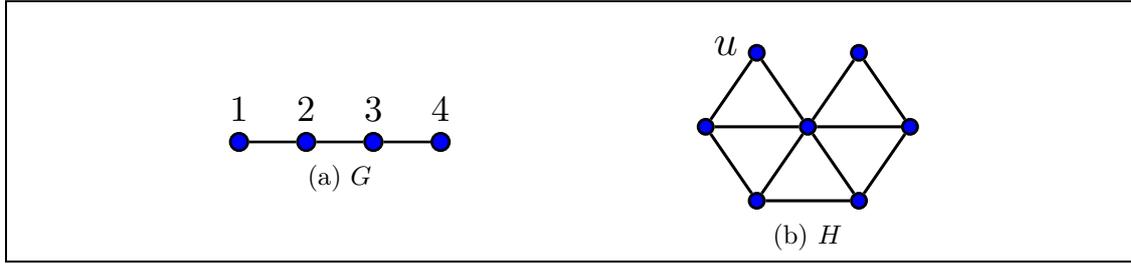
Figure 4.2: There is no relation from G to H.

Note that from Corollary 4.1.10, the radius of H is smaller than the radius of G, so there is no relation R such that $H * R = G$. Therefore there is neither a relation from G to H, nor one from H to G. Note that the two solutions can be different.

Proposition 4.1.13. *Suppose $G * R = H$ and S and D are subsets of V_G and V_H, respectively, such that*

1. *$G[S] * R|_{(S \times D)} = H[D]$;*

2. *$R|_{(S \times D)}$ has full domain on S;*

3. *there is no isolated vertex in $\overline{D}$.*

   Then $\overline{S} * \widetilde{R} = \overline{D}$, where $\widetilde{R} = R|_{(\overline{S} \times \overline{D})}$ is the corresponding restriction of R.

Proof. Obviously, $\overline{S} * \widetilde{R}$ is an induced subgraph of $\overline{D}$. We need to show the reverse inclusion. Given $u \in V_{\overline{D}}$ and $x \in R^{-1}(u)$, we first show that there are two possibilities:

1. x is a vertex of $\overline{S}$.

2. x is an isolated vertex of S.

Assume, to the contrary, x is either a non-isolated vertex of S or x is in the neighbourhood of some vertex of S. In both cases there is $y \in S$ connected by an edge to x. Consequently there is a vertex $v \in D$, such that $v \in R(y)$, and v is connected by an edge to u. It follows $u \notin V_{\overline{D}}$, a contradiction.

Now consider an arbitrary edge $(u, v) \in E_{\overline{D}}$. We have $(x, y) \in E_G$ such that $u \in R(x)$ and $v \in R(y)$. If any of x and y, say x, is an isolated vertex in S, then y is not in S. Thus y is in the neighbourhood of S, which contradicts the two possibilities. So both x and y are not isolated in S, from the above deduction x, y are in the case 1, i.e., they are vertices of $\overline{S}$. Consequently $\overline{S} * \widetilde{R}$ has precisely the same edges as $\overline{D}$. Because $\overline{D}$ has no isolated vertices and thus every vertex is an endpoint of some edge, we know that the vertex set of $\overline{S} * \widetilde{R}$ is same as the vertex set of $\overline{D}$. Hence $\overline{S} * \widetilde{R} = \overline{D}$. $\qquad\square$

This result is of particular practical use in the special case where S and D consist of a single vertex. When looking for a relation R such that $G * R = H$ one can remove a vertex including its neighbourhood from G as well as the prospective image including the neighbourhood from H and then solve the problem on the subgraphs.

4.2   Relational equivalence

Recall that in Section 3.1.2 we define graphs G and H are *homomorphically equivalent* (or *hom-equivalent*) if there exists homomorphisms $G \to H$ and $H \to G$. It is well known that every equivalence class of the homomorphism order contains a minimal representative that is unique up to isomorphism: the *graph core* [40].

We define similar equivalences implied by the existence of (special) relations between graphs. In this section, we require all relations to have full domain unless explicitly stated otherwise. With this condition we will show that these equivalences produce a rich structure closely related to but distinct from the structure of homomorphism equivalences.

This may come as a surprise: the equivalence implied by the existence of surjective homomorphisms is not interesting. Consider two graphs G and H and suppose there are surjective homomorphisms $f : G \to H$ and $g : H \to G$. Since every vertex in V_G has at most one image under f, we have $|V_G| \geq |V_H|$. Analogously $|V_H| \geq |V_G|$, and hence $|V_G| = |V_H|$. Thus f and g are both bijective, and G is isomorphic to H.

Reversible relations

Definition **4.2.1.** A relation R is *reversible with respect to a graph G* if $(G * R) * R^+ = G$.

Recall that in Proposition 4.1.2 we proved that R can be decomposed. Suppose $R = R_D \circ R_C$, where R_D and R_C are constructed as in the proof of Proposition 4.1.2. R_C is a functional relation and R_D is an injective relation with full domain. Then we can get the following result with respect to reversible relation.

Proposition **4.2.1.** *R is reversible with respect to G if and only if for every α and β satisfying $R_C(\alpha) = R_C(\beta)$ we have $N_{G*R_D}(\alpha) = N_{G*R_D}(\beta)$.*

Proof. We set $G_1 = G * R_D$, If $R_C(\alpha) = R_C(\beta)$ implies $N_{G_1}(\alpha) = N_{G_1}(\beta)$, then $H * R_C^+ = G_1$. Therefore,

$$G * R * R^+ = H * R^+ = H * (R_D * R_C)^+$$
$$= H * R_C^+ * R_D^+ = G_1 * R_D^+ = G.$$

The first equation is by assumption, the second is by Proposition 4.1.2, the third one is by the property of transpose, the fourth is from Lemma 4.1.1, the last equation is because R_D is injective. Thus by definition, R is reversible.

Conversely, since R is reversible, i.e., $H * R^+ = G$, setting $G_2 = H * R_C^+$ gives $G_2 * R_D^+ = G$. Hence $G_1 * R_C * R_C^+ = G_2$ and $G_2 * R_D^+ * R_D = G_1$. From $I_{G_1} \subseteq R_C * R_C^+$ we conclude $G_1 \subseteq G_2$, and similarly $I_{G_2} \subseteq R_D^+ * R_D$ yields $G_1 \supseteq G_2$. Hence $G_1 = G_2$. R_C^+ is injective, hence $\alpha, \beta \in V_{G_2} = V_{G_1}$ has the same open neighbourhood whenever the pre-image of α and β under R_C^+ coincide, i.e. $R_C(\alpha) = R_C(\beta)$. $\qquad\square$

R_D is an injective relation, hence one can easily get $N_{G*R_D}(\alpha) = R_D(N_G(x))$ provided that $(x, \alpha) \in R_D$. On the other hand, if we define R to be the image of R_D as in the proof of Proposition 4.1.2, then $R_C(\alpha) = R_C(\beta)$ implies there are two distinct vertices $x, y \in V_G$, s.t. $(x, u), (y, u) \in R$, where $u = R_C(\alpha) = R_C(\beta)$, and verse visa. Using Proposition 4.2.1 we thus obtain

Proposition 4.2.2. *A relation R is reversible with respect to G if and only if for every two vertices x and y such that $R(x) \cap R(y) \neq \emptyset$ we have $N_G(x) = N_G(y)$.*

Strong relational equivalence

Definition 4.2.2. Two graphs G and H are *strongly relationally equivalent*, denoted by $G \backsim_s H$, if there is a relation R such that $G * R = H$ and $H * R^+ = G$.

Lemma 4.2.3. *Relational equivalence is an equivalence relation on graphs.*

Proof. By definition $\backsim_s$ is symmetric. Because $G * I_G = G$, relation $\backsim_s$ is reflexive. Suppose $G \backsim_s G'$ and $G' \backsim_s G''$. Thus there are relations R, S, such that $G' = G * R$, $G'' = G' * S$, $G = G' * R^+$, and $G' = G'' * S^+$. By the composition law (Lemma 4.1.1) it follows that $G'' = G * (R \circ S)$ and $G = G'' * (S^+ \circ R^+) = G'' * (R \circ S)^+$, i.e, $G \backsim_s G''$. Hence $\backsim_s$ is transitive. $\qquad\square$

Recall that the *thinness relation S* of G is the equivalence relation on V_G defined by $(x, y) \in S$ if and only if $N_G(x) = N_G(y)$. We denote by $\mathcal{S}$ the corresponding partition of V_G, and write $R_S \subseteq V_G \times \mathcal{S}$ that associates each vertex with its S-equivalence class, i.e., $(x, \beta) \in R_S$ if and only if $x \in \beta$. In this context it is well known that G can be reconstructed from G_{pd} (the point-determining graph of G) and knowledge of the S-equivalence classes. In fact, we have

$$G_{pd} * R_S^+ = G. \tag{4.1}$$

Theorem 4.2.4. *G and H are in the same equivalence class w.r.t. $\backsim_s$ if and only if their point-determining graphs are isomorphic.*

Proof. Assume $G \backsim_s H$. From Equation(4.1) we know that $G \backsim_s G_{pd}$, $H \backsim_s H_{pd}$, so $G_{pd} \backsim_s H_{pd}$. Now we claim that G_{pd} and H_{pd} are isomorphic. Suppose $G_{pd} * R = H_{pd}$, then the pre-image of R is unique. Otherwise, there exist distinct vertices $x, y \in V_{G_{pd}}$ such that $R(x) = R(y)$, then $N_{G_{pd}}(x) = N_{G_{pd}}(y)$, contradicting thinness. Likewise, the pre-image of R^+ is unique, i.e., the image of R is unique. Hence R is one-to-one. □

(a) G (b) H (c) $G_{pd} = H_{pd}$
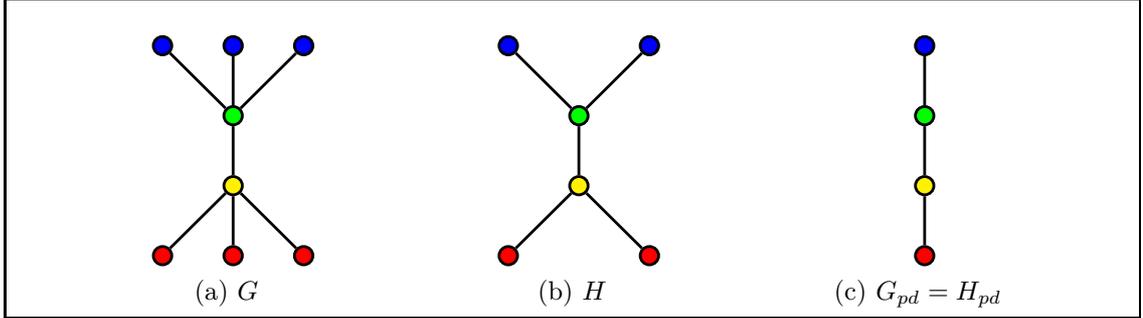
Figure 4.3: Non-isomorphic graphs G and H with isomorphic point-determining graphs.

The example in Fig. 4.3 shows that point-determining graphs can be isomorphic while G and H themselves are not isomorphic. Relational equivalence thus is coarser than graph isomorphism (surjective homomorphic equivalence) but stronger than homomorphic equivalence.

Weak relational equivalence

Definition **4.2.3.** Two graphs G and H are *(weak) relationally equivalent*, denoted by $G \backsim_w H$, if there are relations R and S such that $G * R = H$ and $H * S = G$.

Note that $\backsim_w$ is the same as $\sim^{PR}$ in Chapter 5.

Lemma **4.2.5.** *Weak relational equivalence is an equivalence relation on graphs.*

Proof. By definition $\backsim_w$ is symmetric. Because $G * I_G = G$, relation $\backsim_w$ is reflexive. Suppose $G \backsim_w G'$ and $G' \backsim_w G''$. Thus there are relations R, S, R', and S', such that $G' = G * R$, $G'' = G' * R'$, $G = G' * S$, and $G' = G'' * S'$. By the composition law (Lemma 4.1.1) it follows that $G'' = G * (R \circ R')$ and $G = G'' * (S' \circ S)$, i.e, $G \backsim_w G''$. Hence $\backsim_w$ is transitive. □
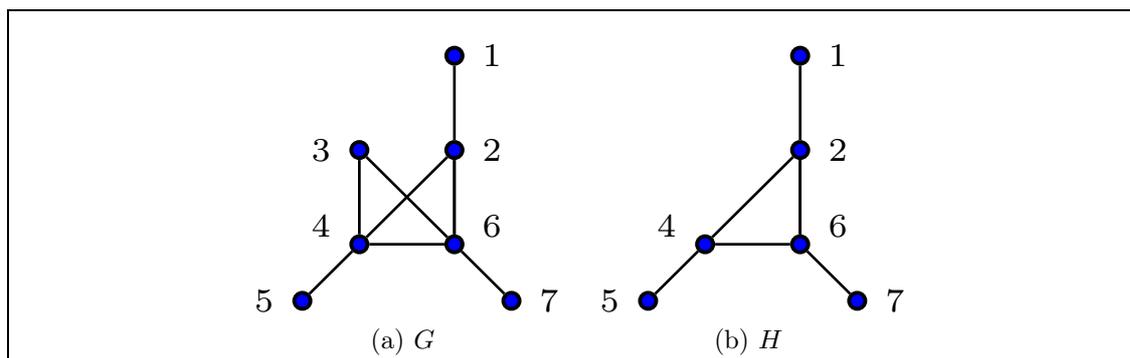
Figure 4.4: G and H are weakly relationally equivalent but have non-isomorphic graphs.

Strong relational equivalence implies weak relational equivalence. To see this, simply observe that the definition of the weak form is obtained from the strong one by setting $S = R^+$.

The converse is not true, as shown by the graphs G and H in Fig. 4.4: It is easy to see that their point-determining graphs are different and thus G and H are not strongly relationally equivalent. However, they are relationally equivalent. To get relation from G to H contract vertices 2 and 3 and keep other vertices on place, i.e.,

$$R = \{(1,1), (2,2), (3,2), (4,4), (5,5), (6,6), (7,7)\}.$$

To get relation from H to G, duplicate 5 and 7 and contract them together to 3,

$$S = \{(1,1), (2,2), (4,4), (5,5), (6,6), (7,7), (5,3), (7,3)\}.$$

Consequently, weak relational equivalence is coarser than strong relational equivalence.

R-cores

A graph is an *R-core*, if it is the minimal graph (in the number of vertices) in its equivalence class of $\smallfrown_w$.

This notion is analogous to the definition of graph cores. In this section we show properties of R-cores that are similar to the properties of graph cores. To this end we first need to develop a simple characterization of R-cores.

Again we start from a decomposition of relations. Consider a relation R such that $G * R = H$. We seek for pair of relations R_1 and R_2 such that $R = R_1 \circ R_2$. In contrast to Lemma 4.1.2, however, we now look for a decomposition so that the graph $G' = G * R_1$ is smaller (in the number of vertices) than G.

The existence of such a decomposition follows from a translation of the well-known Hall Marriage Theorem [76] to the language of relations. We say that
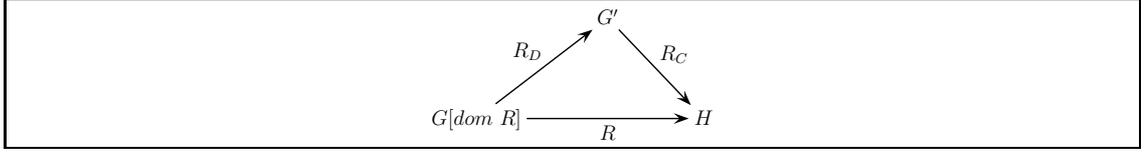
Figure 4.5: Commutative diagram.

the relation $R \subseteq A \times B$ satisfies the *Hall condition*, if for every $S \subseteq A$ we have $|S| \leq |R(S)|$.

We have already proved Hall's marriage theorem in 2.1.1. Here we restate it in terms of relations.

Theorem 4.2.6 (Hall's marriage theorem). *If $G * R = H$ and R satisfies the Hall condition, then R contains a monomorphism $f : G \to H$.*

Lemma 4.2.7. *If $G * R = H$ and relation R does not satisfy the Hall condition, then there are relations R_1 and R_2 such that $R = R_1 \circ R_2$, and the number of vertices of graph $G' = G * R_1$ is strictly smaller than the number of vertices of G.*

Proof. Without loss of generality assume that $V_G \cap V_H = \emptyset$. If R does not satisfy the Hall condition, then there exists a vertex set $S \subset V_G$ such that $|S| > |R(S)|$. Now we define relations R_1 and R_2 as follows:

$$R_1(x) = \begin{cases} R(x) \text{ for } x \in S, \\ x \text{ otherwise,} \end{cases} \qquad R_2(x) = \begin{cases} x \text{ for } x \in R(S), \\ R(x) \text{ otherwise.} \end{cases}$$

Obviously $R_1 \circ R_2 = R$ and $|V_{G'}| = |V_G| - (|S| - |R(S)|) < |V_G|$. $\qquad \square$

This immediately gives a necessary, but in general not sufficient, condition for a graph to be an R-core.

Corollary 4.2.8. *If G is an R-core, then every relation R such that $G * R = G$ satisfies the Hall condition and thus contains a monomorphism.*

Proof. Assume that there is a relation R that does not satisfy the Hall condition. Then there is a graph G', $|V_{G'}| < |V_G|$, and relations R_1 and R_2 such that $G * R_1 = G'$ and $G' * R_2 = G$. Consequently G' is a smaller representative of the equivalence class of $\backsim_w$, a contradiction with G being R-core. $\qquad \square$

To see that the condition of Corollary 4.2.8 is not sufficient, consider a graph G consisting of two independent vertices. It is not an R-core, because it can retract to an isolated vertex. However, the relation is required to be full, any relation from G to itself satisfies the Hall condition.

Next we show that R-cores are, up to isomorphism, unique representatives of the equivalence classes of $\backsim_w$.
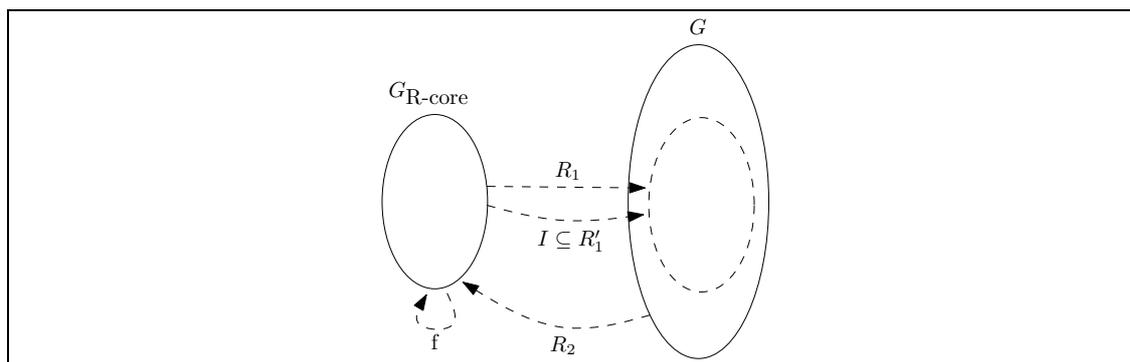
Figure 4.6: Construction of an embedding from $G_{\text{R-core}}$ to G.

Proposition 4.2.9. *If both G and H are R-cores in the same equivalence class of $\backsim_w$, then G and H are isomorphic.*

Proof. Because both G and H are R-cores, we know that $|V_G| = |V_H|$.

Consider relations R_1 and R_2 such that $G * R_1 = H$ and $H * R_2 = G$. Applying Lemma 4.2.7 we know that R_1 satisfies the Hall condition. Otherwise there would be a graph G' with $|V_{G'}| < |V_G|$ so that G' is relationally equivalent to both G and H which contradicts the fact that G and H are R-cores. Similarly, we can show that R_2 also satisfies the Hall condition.

From Theorem 4.2.6 we know that there is a monomorphism f from G to H, and monomorphism g from H to G. It follows that the number of edges of G is not larger than the number of edges of H and vice versa. Because G and H have the same number of edges and same number of vertices, G and H must be isomorphisms. $\square$

It thus makes sense to define a construction analogous to the core of a graph.

Definition 4.2.4. H is an *R-core of a graph G* if H is an R-core and $H \backsim_w G$.

All R-cores of graph G are isomorphic as an immediate consequence of Proposition 4.2.9. We denote the (up to isomorphism) unique R-core of graph G by $G_{\text{R-core}}$..

Lemma 4.2.10. *$G_{R\text{-}core}$ is isomorphic to a (not necessarily induced) subgraph of G.*

Proof. Take any relation R such that $G_{\text{R-core}} * R = G$. By the same argument as in Corollary 4.2.8, there is a monomorphism $f : G_{\text{R-core}} \to G$ contained in R. Consider the image of f on G. $\square$

Theorem 4.2.11. *$G_{R\text{-}core}$ is isomorphic to an induced subgraph of G.*

Proof. Fix R_1 and R_2 such that $G_{\text{R-core}} * R_1 = G$ and $G * R_2 = G_{\text{R-core}}$.

$R = R_1 \circ R_2$ is a relation such that $G_{\text{R-core}} * R = G_{\text{R-core}}$. By Corollary 4.2.8, R contains a monomorphism $f : G_{\text{R-core}} \to G_{\text{R-core}}$. Because such a monomorphism is a permutation, there exists an n such that f^n is identity.

Put $R_1' = R^{n-1} \circ R_1$. Because R^n contains the identity and $R^n = R_1' \circ R_2$, it follows that for every $x \in V_{G_{\text{R-core}}}$, there is a vertex $I(x) \in V_G$ such that $I(x) \in R_1'(x)$ and $x \in R_2(I(x))$. In other words, $(x, I(x)) \in R_1'(x)$ and $(I(x), x) \in R_2$. It is easily seen that I is a mapping from vertex set of $G_{\text{R-core}}$ to vertex set of G. We take the induced subgraph of G induced by the image of I, denoted by G', and claim that I is an isomorphism of $G_{\text{R-core}}$ to G'.

We first show that for two vertices $x \neq y$, we have $I(x) \neq I(y)$ and thus I is vertex injective (thus both vertex and edge injective). Assume, to the contrary, that there are two vertices $x \neq y$ such that $I(x) = I(y)$. Consider an arbitrary vertex z in the neighbourhood of x. It follows that $I(z)$ must be in the neighbourhood of $I(x)$ and consequently z is in the neighbourhood of y. Thus the neighbourhoods of x and y are the same. By Theorem 4.2.4, however, we know that the R-core is a point-determining graph (because weak relational equivalence is coarser than strong relational equivalence), a contradiction.

Then we claim that I is a homomorphism of $G_{\text{R-core}}$ to G', i.e., for every edge $(x, y) \in E_{G_{\text{R-core}}}$ we also have edge $(I(x), I(y)) \in E_G$. This is because $I \subset R_1'$, then $G_{\text{R-core}} = G_{\text{R-core}} * I \subset G_{\text{R-core}} * R_1' = G$.

By the construction, I is vertex surjective. Finally we prove that I is also a edge surjective homomorphism by claiming I^{-1} is a homomorphism of G' to $G_{\text{R-core}}$. We see $I^{-1} \subset R_2$ by observing $x \in R_2(I(x))$ implies $(I(x), x) \in R_2$ for all $x \in$ vertex set of $G_{\text{R-core}}$. Because I as well as I^{-1} is one-to-one, $G' = G' * I^{-1} \subset G' * R_2 = G_{\text{R-core}}$. $(I(x), I(y)) \in G'$ corresponds to an edge $(x, y) \in G_{\text{R-core}}$.

Hence we have proved I is a bijective (both edge and vertex) homomorphism, thus I is an isomorphism. $\square$

We close the section with an algorithm for computing the R-core of a graph. In contrast to graph cores, where the computation is known to be NP-complete, there is a simple polynomial algorithm for R-cores.

Observe that the R-core of a graph containing isolated vertices is isomorphic to the disjoint union of the R-core of the same graph with the isolated vertices removed and a single isolated vertex. The R-core of a graph without isolated vertices can be computed by Algorithm 1.

The algorithm removes all vertices $v \in G$ such that (1) the neighbourhood of v is union of the neighbourhood of some other vertices $v_1, v_2, \ldots, v_n$ and (2) there is vertex u such that $N_G(v) \subseteq N_G(u)$.

It is easy to see that the resulting graph H is relationally equivalent to G. Condition (1) ensures the existence of a relation R_1 such that $H * R_1 = G$, while

---

Algorithm 1 The R-core of a graph

Input:

Graph G with loops allowed and without isolated vertices, vertex set denoted by V, neighbourhoods $N_G(i)$, $i \in V$.

1: **for** $i \in V$ **do**
2:     $W(i) = \emptyset$
3:     found = FALSE
4:     **for** $j \in V \setminus \{i\}$ **do**
5:         **if** $N(j) \subseteq N(i)$ **then**
6:             $W(i) := W(i) \cup N(j)$
7:         **end if**
8:         **if** $N(i) \subseteq N(j)$ **then**
9:             found = TRUE
10:        **end if**
11:    **end for**
12:    **if** $W(i) = N(i)$ **and** found **then**
13:        delete i from V
14:        $N(i) = \emptyset$
15:    **end if**
16: **end for**
17: **return** The R-core $G[V]$ of G.

---

condition (2) ensures the existence of a relation R_2 such that $G * R_2 = H$.

We need to show that H is isomorphic to $G_{\text{R-core}}$. By Theorem 4.2.11 we can assume that $G_{\text{R-core}}$ is an induced subgraph of H that is constructed as an induced subgraph of G.

We also know that there are relations R_1 and R_2 such that $G_{\text{R-core}} * R_1 = H$ and $G * R_2 = G_{\text{R-core}}$. By the same argument as in the proof of Theorem 4.2.11 we can assume both R_1 and R_2 to contain an (restriction of) identity.

Now assume that there is a vertex $v \in V_H \setminus V_{G_{\text{R-core}}}$. We can put $u = R_2(v)$ and because R_2 contains an identity we have $N_G(v) \subseteq N_G(u)$. We can also put $\{v_1, v_2 \ldots v_n\}$ to be the set of all vertices such that $v \in R_1(v_i)$. It follows that the neighbourhood of v is the union of the neighbourhoods of $v_1, v_2, \ldots, v_n$ and by the construction of H we have $v \notin V_H$, a contradiction.

4.3   The partial order $\mathrm{Rel}(G, H)$

Basic properties

For fixed graphs G and H we consider the partial order $\mathrm{Rel}(G, H)$. The vertices of this partial order are all relations R such that $G * R = H$. We put $R_1 \leq R_2$ if and only if $R_1 \subseteq R_2$.

This definition is motivated by Hom-complexes, see [56]. In this section we show the basic properties of this partial order and concentrate on minimal elements in the special case of $\mathrm{Rel}(G, G)$.

Proposition 4.3.1. *Suppose $G * R' = H$, $G * R'' = H$ and $R' \subseteq R''$, then any relation R with $R' \subseteq R \subseteq R''$ also satisfies $G * R = H$.*

Proof. From $R' \subseteq R \subseteq R''$ we conclude $G * R' \subseteq G * R \subseteq G * R''$. Hence $G * R' = G * R''$ implies $G * R = H$. □

Hence it is possible to describe the partial order $\mathrm{Rel}(G, H)$ by listing minimal and maximal solutions R of $G * R = H$ w.r.t. set inclusion.

For example, if G is P_3 with vertices v_0, v_1, v_2, v_3 and H is P_1 with vertices x_0, x_1, it is easily seen that $R'' = \{(v_0, x_0), (v_2, x_0), (v_1, x_1), (v_3, x_1)\}$ is a maximal solution of $G * R = H$ and $R' = \{(v_0, x_0), (v_1, x_1)\}$ is a minimal solution, because $R' \subset R''$, then all the relations R with $R' \subseteq R \subseteq R''$ satisfy $G * R = H$. We note that minimal and maximal solutions need not be unique.

Solutions of $G * R = G$

For simplicity, we say that a relation R is an *automorphism* of G if it is of the form $R = \{(x, f(x)) \mid x \in V_G\}$ and $f : V_G \to V_G$ is an automorphism of G.

We will see that conditions related to thinness again play a major role in this context. Recall that G is point-determining if no two vertices have the same neighbourhood, in other words, $N_G(x) = N_G(y)$ implies $x = y$. Here we need an even stronger condition:

Definition 4.3.1. A graph G satisfies *condition N* if $N_G(x) \subseteq N_G(y)$ implies $x = y$.

In particular, graphs satisfying condition N are point-determining.

Proposition 4.3.2. *For a given graph G, the set $\mathrm{Rel}(G, G)$ of all relations satisfying $G * R = G$ forms a monoid.*

Proof. Firstly, $R, S \in \mathrm{Rel}(G, G)$ implies $G * R = G$ and $G * S = G$ and thus $G * (R \circ S) = G$, so that $R \circ S \in \mathrm{Rel}(G, G)$. Furthermore, for any $R, S, T \in \mathrm{Rel}(G, G)$ we have $(R \circ S) \circ T = R \circ (S \circ T)$. Finally, the identity relation I_G is a left and right identity for relational composition: $I_G \circ R = R \circ I_G = R$. □

A relation $R \subset V_G \times V_G$ can be interpreted as a directed graph $\vec{R}$ with vertex set V_G and a directed edge $u \to v$ if and only if $(u, v) \in R$. Note that $\vec{R}$ may have loops. We say that $v \in V_G$ is *recurrent* if and only if there exists a walk (of length at least 1) from v to itself. Let S_G be the set of all the recurrent vertices. Furthermore, we define an equivalence relation ξ on S_G by setting $(u, v) \in \xi$ if there are both a walk in $\vec{R}$ from u to v and from v to u. The equivalence classes w.r.t. ξ are denoted by $\vec{R}/\xi = \{D_1, D_2, \ldots, D_m\}$. We furthermore define a binary relation $\geq$ over $\vec{R}/\xi$ as follows: if there is a walk from a vertex u in D_i to a vertex v in D_j, then we say $u \geq v$. It is easily seen that $\geq$ is reflexive, antisymmetric, and transitive, hence $(\vec{R}/\xi, \geq)$ is a partially ordered set. W.l.o.g. we can assume $\{D_1, D_2, \ldots, D_s\}$ are the maximal elements w.r.t. $\geq$. Now let $G_r = G[D_1 \cup \cdots \cup D_s]$ be the subgraph of G induced by these maximal elements.

Lemma 4.3.3. *For arbitrary $x \in V_G$, there exist an $l \in \mathbb{N}$ and a recurrent vertex v such that $(v, x) \in R^l$, where R^l is l-times composition of relation R.*

Proof. Set $x_0 = x$ and choose $x_i \in R^{-1}(x_{i-1})$ for all $i \geq 1$. Since $|V_G| < \infty$, there are indices $j, k \in \mathbb{N}$, $j < k$, $x_j = x_k$. Then x_j is a recurrent vertex. The lemma follows by setting $l = j$ and $v = x_i$. □

Lemma 4.3.4. *For every $v \in V_{G_r}$, $R^{-1}(v) \subseteq V_{G_r}$.*

Proof. We prove this by two steps:
First, we claim that for any $x \in R^{-1}(v)$, x is recurrent. Suppose $x \in R^{-1}(v)$ is not recurrent. Lemma 4.3.3 implies that there is an $l \in \mathbb{N}$ and a recurrent vertex w such that $(w, x) \in R^l$. Hence the definitions of E and $\geq$ imply $[w] \geq [v]$, where $[v]$ denotes the equivalence class (w.r.t. E) containing the vertex v. Since $[v]$ is maximal w.r.t. $\geq$, we have $[v] = [w]$. Consequently, there exists an index $k \in \mathbb{N}$ such that $(v, w) \in R^k$. On the other hand, we have $(x, x) = (x, v) \circ (v, w) \circ (w, x) \in R^{k+l+1}$. Thus, x is recurrent, a contradiction. Therefore, every vertex $x \in R^{-1}(v)$ is recurrent.
Second, we prove x is a maximal element in the partial order. Hence $[x] \geq [v]$ together with the maximality of $[v]$ gives $[x] = [v]$, and thus $x \in V_{G_r}$. □

Lemma 4.3.5. *For every $x \in V_G$, there is $l \in \mathbb{N}$ such that, for arbitrary $i \geq l$, there exists $u \in V_{G_r}$ satisfying $(u, x) \in R^i$.*

Proof. From Lemma 4.3.3 and Lemma 4.3.4 we conclude that it is sufficient to show that for an arbitrary recurrent vertex v there is a $k \in \mathbb{N}$ and $w \in V_{G_r}$ such that $(w, v) \in R^k$. The lemma now follows easily from the finiteness of V_G. $\square$

Now we introduce a lemma which connects relation R and the neighbourhood of vertex in a graph. The lemma is simple but very useful.

Lemma 4.3.6. *Suppose $G * R = H$. For any $u, v \in V_H$, if $R^{-1}(u) \subseteq R^{-1}(v)$, then $N_H(u) \subseteq N_H(v)$.*

Proof. Suppose there exist $u, v \in V_H$, $R_{-1}(u) \subseteq R_{-1}(v)$ but $N_H(u) \nsubseteq N_H(v)$, then there exist a vertex w of H, $w \in N_H(u)$ but $w \notin N_H(v)$. Because (u, w) is an edge in H and $G * R = H$, we can find $x, y \in V_G$ such that $x \in R_{-1}(w), y \in R_{-1}(u)$ and (x, y) is an edge in G. $R_{-1}(u) \subseteq R_{-1}(v)$ implies $y \in R_{-1}(v)$. We already have $(x, w) \in R$ and (x, y) is an edge in G, by the definition of $*$ we have (v, w) is an edge of H, which contradicts to $w \notin N_H(v)$. $\square$

From these lemmata we can deduce:

Theorem 4.3.7. *All solutions of $G * R = G$ are automorphisms if and only if G has property N.*

Proof. Suppose there are distinct vertices $x, y \in V_G$ such that $N_G(x) \subseteq N_G(y)$. Then $R = I_G \cup (x, y)$, which is not functional, satisfies $G * R = G$. Thus $G * R = G$ is also solved by relations that are not automorphisms of G. This proves the 'only if' part.

Conversely, suppose G has property N. **Claim:** There is a $k \in \mathbb{N}$ such that $R^k \cap (V_{G_r} \times V_{G_r}) = I_{G_r}$.

For each $v_i \in V_{G_r}$ there is a walk of length $s_i \geq 1$ from v_i to itself. Hence $(v_i, v_i) \in R^{s_i}$. Let s be the least common multiple of all the s_i. Then $(v_i, v_i) \in R^s$ for all $v_i \in V_{G_r}$. Define $Q := R^s \cap (V_{G_r} \times V_{G_r})$. Thus $I_{G_r} \subseteq Q$ and moreover $Q^j \subseteq Q^{j+1}$ for all $j \in \mathbb{N}$. Since V_{G_r} is finite, there is an $n \in \mathbb{N}$ such that $Q^{n+1} = Q^n$, and hence $Q^{2n} = Q^n$. Let us write $R^{-i}(v) := \{u \in V_G \mid (u, v) \in R^i\}$. For $v \in V_{G_r}$ we have $R^{-i}(v) \subset V_{G_r}$ (from Lemma 4.3.4) and hence $Q^{-n}(v) = R^{-sn}(v)$ for all $v \in V_{G_r}$. If $Q^n \neq I_{G_r}$, then there are two distinct vertices $u, v \in V_{G_r}$, such that $(u, v) \in Q^n$. From Lemma 4.3.6 $N_G(u) \nsubseteq N_G(v)$ and $G = G * R^{sn}$ allow us to conclude that $R^{-sn}(u) \nsubseteq R^{-sn}(v)$ and $R^{-sn}(v) \nsubseteq R^{-sn}(u)$. Hence, there is a vertex w, such that $(w, u) \in Q^n$ and $(w, v) \notin Q^n$. From $(u, v) \in Q^n$ and $(w, u) \in Q^n$ we conclude $(w, v) \in Q^n \circ Q^n = Q^{2n}$, contradicting to $Q^{2n} = Q^n$. Therefore $Q^n = I_{G_r}$. Setting $k = sn$ now implies the claim.

Finally, we show $V_{G_r} = V_G$. For any $v \in V_G \setminus V_{G_r}$, Lemma 4.3.5 implies the existence of $w \in V_{G_r}$ and $m \in \mathbb{N}$ such that $(w, v) \in R^{mk}$. However, we have claimed $R^{-k}(w) = \{w\}$, hence $R^{-mk}(w) = \{w\}$. From Lemma 4.3.6, this, however,

implies $N_G(w) \subseteq N_G(v)$ and thus contradicts property N. Therefore, $V_G = V_{G_r}$ and moreover $R^k = I_G$. This R is an automorphism. $\qquad\square$

4.4 R-Retractions

Recall that an important special case of ordinary graph homomorphisms are homomorphisms to subgraphs, and in particular so-called retractions. Let H be a subgraph of G, a *retraction* of G to H is a homomorphism $r : V_G \to V_H$ such that $r(x) = x$ for all $x \in V_H$.

We introduced the graph cores in Section 4.2 as minimal representatives of the homomorphism equivalence classes. The classical and equivalent definition is the following: A *(graph) core* is a graph that does not retract to a proper subgraph. Every graph G has a unique core H (up to isomorphism), hence one can speak of H as *the core of G*, see [40].

Here, we introduce a similar concept based on relations between graphs. Again to obtain a structure related to graph homomorphisms, in this section we require all relations to have full domain unless explicitly stated otherwise.

Definition 4.4.1. Let H be a subgraph of G. An *R-retraction* of G to H is a relation R such that $G * R = H$ and $(x, x) \in R$ for all $x \in V_H$. If there is an R-retraction of G to H we say that H is a *retract* of G.

Lemma 4.4.1. *If H is an R-retract of G and K is an R-retract of H, then K is an R-retract of G.*

Proof. Suppose T is an R-retraction of H to K and S is an R-retraction of G to H. Then $(G * S) * T = G * (S \circ T) = K$. Furthermore $(x, x) \in T$ for all $x \in V_K \subseteq V_H$, and $(u, u) \in S$ for all $u \in V_H$, hence $(x, x) \in S \circ T$ for all $x \in V_k$. Therefore $S \circ T$ is an R-retraction from G to K. $\qquad\square$

Hence, the following definition is meaningful.

Definition 4.4.2. A graph is *R-reduced* if there is no R-retraction to a proper subgraph.

Thus, we can also speak about 'the R-reduced graph of a graph G' as the smallest subgraph on which it can be retracted. We will see below that the R-reduced graph of a graph is always unique up to isomorphism.

We remark that R-reduced graphs differ from R-cores introduced in Section 4.2, thus we chose an alternative name used also in homomorphism setting (cores are also called reduced graphs).

Lemma 4.4.2. *Let G be a graph with loops and o a vertex of G with a loop on it. Then the R-reduced graph of G is the subgraph induced by $\{o\}$.*

Proof. Let O be the graph induced by $\{o\}$, and $R = \{(x, o) \mid x \in V_G\}$, then it is easily seen R is an R-retraction of G to O. Moreover, since O has only one vertex, there is no R-retraction to its subgraphs. So O is an R-reduced graph of G.

Conversely, let H be an R-reduced graph of G and denote by R the R-retraction from G to H. Then a loop of G must generate a loop of H via R, denote it by O. Similarly to above, we see O is an R-retract of H, hence it is also an R-retract of G (by Lemma 4.4.1). Therefore the definition of R-reduced graph implies $H = O$. $\square$

In the remainder of this section, therefore, we will only consider graphs without loops.

**Lemma 4.4.3.** *If G is R-reduced, then G has property N.*

Proof. Suppose there are two distinct vertices $x, y \in V_G$ with $N_G(x) \subseteq N_G(y)$ and consider the induced graph $G/x := G[V_G \setminus \{x\}]$ obtained from G by deleting the vertex x and all edges incident with x. The relation $R = \{(z, z) \mid z \in V_G \setminus \{x\}\} \cup \{(x, y)\}$ satisfies $G * R = G/x$: the first part is the identity on G/x and already generates all necessary edges in G/x. The second part transforms edges of the form $(x, z) \in E_G$ to edges (y, z). Since R has full domain and contains the identity relation restricted to G/x, it is an R-retraction of graph G, and hence G is not R-reduced. $\square$

Proposition 4.4.4. *A graph G is R-reduced if and only if it has no relation to a proper subgraph.*

Proof. The 'if' part is trivial. Now we suppose that H is a proper induced subgraph of graph G with the minimal number of vertices such that there is a relation R satisfying $G * R = H$. Then H does not have a relation to a proper subgraph of itself. We claim that H has property N; otherwise, one can find a vertex $u \in V_H$ and construct a retraction from H to H/u as in Lemma 4.4.3, which causes a contradiction. Denote $\widetilde{R} = R \cap (V_H \times V_H)$, then $K = H * \widetilde{R}$ is a subgraph of H. From our assumptions on H we obtain $K = H$. By virtue of Theorem 4.3.7, $\widetilde{R}$ is induced by an automorphism of H. Hence $R \circ \widetilde{R}^+$ is again a relation of G to H that contains the identity on H, i.e., it is an R-retraction. $\square$

Since graph cores are induced subgraphs and retractions are surjective they also imply relations. Proposition 4.4.4 is also a consequence of this fact. We refer to [40] for a formal proof.

We call R an *minimal R-retraction* if there is no R-retraction R' such that $R \supset R' \supset I_H$.
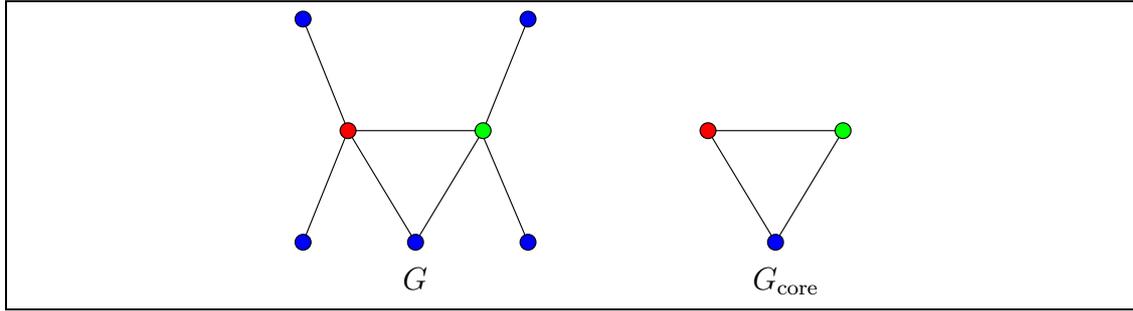
Figure 4.7: A graph G and its core.

Lemma 4.4.5. *Let H be an R-retract of G. Then any minimal R-retraction of G to H is functional.*

Proof. Suppose R is a minimal R-retraction of G to H. If R is not functional, then there exist distinct $x, y \in V_H$ such that $(u, x), (u, y) \in R$. Hence we could always pick a vertex from $\{x, y\}$ which is different of u, w.l.o.g. suppose it is x. Then $R \setminus (u, x)$ is an R-retraction, which contradicts minimality. To see this, set $R' = R \setminus (u, x)$, then $R \supset R' \supset I_H$ and moreover $H = G * I_H \subseteq G * R' \subseteq G * R = H$, and thus $G * R' = H$. □

Proposition 4.4.6. *A graph is R-reduced if and only if it is a graph core.*

Proof. If H is R-reduced from G there is an R-retraction from G to H which can be chosen minimal and hence by Lemma 4.4.5 is functional and hence is a homomorphism retraction. Conversely, a homomorphism retraction is also an R-retraction. Hence the R-reduced graphs coincide with the graph cores. □

Proposition 4.4.7. *Suppose H is the core of graph G. If $H * R = K$ then there is a relation R' such that $G * R' = K$. If $K * S = G$, then there is a relation S' such that $K * S' = H$.*

Proof. Since H is the core of graph G, there is a relation R_1 such that $G * R_1 = H$. If $H * R = K$ we have $G * R_1 * R = K$ and $R' = R_1 \circ R$ satisfies $G * R' = K$. If $K * S = G$ we have $K * S * R_1 = H$ and $S' = S \circ R_1$ satisfies $K \circ S' = G$. □

Cocores

In the classical setting of maps between graphs, one can only consider retractions from a graph to its subgraphs, since graph homomorphisms of an induced subgraph to the original graph are just the identity map. In the setting of relations between graphs, however, it appears natural to consider relations with identity restriction between a graph and an induced subgraph. This gives rise

to the notions of R-coretraction and R-cocore in analogy with R-retractions and R-reduced graphs.

Definition **4.4.3.** Let H be a subgraph of graph G. An *R-coretraction* of H to G is a relation R such that $H * R = G$ and $(x, x) \in R$ for all $x \in V_H$. We say that H is an *R-coretract* of G.

Lemma **4.4.8.** *If H is an R-coretract of a graph G and K is an R-coretract of H, then K is an R-coretract of G.*

Proof. Suppose T is an R-coretraction of K to H and S is an R-coretraction of H to G. Then $(K * T) * S = K * (T \circ S) = G$. Furthermore $(x, x) \in T$ for all $x \in V_K \subseteq V_H$, and $(v, v) \in S$ for all $v \in V_H$, hence $(x, x) \in T \circ S$ for all $x \in V_K$. Therefore $T \circ S$ is an R-coretraction from K to G. □

Hence, the following definition is meaningful.

Definition **4.4.4.** An R-coretract H of a graph G is an *R-cocore of G* if H does not have a proper subgraph that is an R-coretract of H (and hence of G).


(a) G
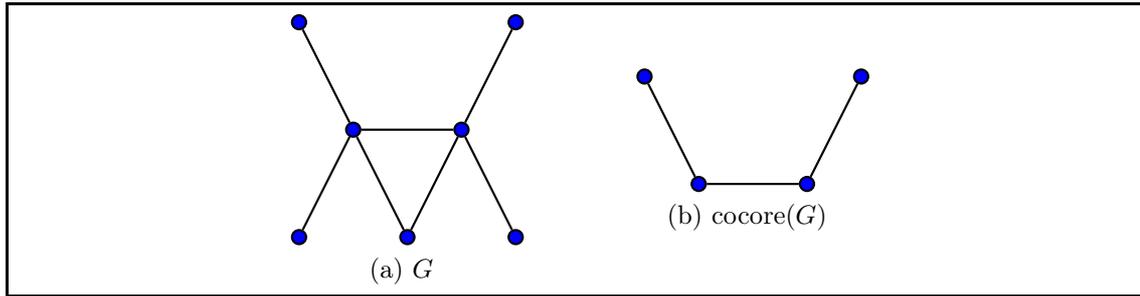
(b) cocore(G)

Figure 4.8: A graph and its cocore.

Clearly, the reference to G is irrelevant: A graph G is an *R-cocore* if there is no proper subgraph of G that is an R-coretract of G. Similarly, we call R a *minimal R-coretraction* of H to G if there exists no R-coretraction R', such that $R' \subset R$.

Lemma **4.4.9.** *Let H be an R-coretract of graph G, and let R be a minimal R-coretraction of H to G. Then the restriction of R to H equals to I_H.*

Proof. Suppose $R \cap (V_H \times V_H) \neq I_H$ and define $R_1 = R \setminus \{(x, y) \in R : x, y \in V_H, x \neq y\}$. Then $H * R_1 \subseteq H * R = G$. We claim that $H * R_1 = H * R$ and thus R_1 is an R-coretraction of H to R, contradicting the minimality of R.

To prove this claim, it is sufficient to show that any edge $e \in E_G$ is contained in $H * R_1$. If e is not incident with any vertex in V_H or $e \in E_H$, the conclusion is trivial. So we only need to consider $e = (z, u)$ with $z \in E_H$ and $u \in V_G \setminus V_H$.

Since $G = H * R$, one can find $x_1, x_2 \in V_H$ such that $(x_1, z), (x_2, u) \in R$ and $(x_1, x_2) \in E_H$. Because $H \subseteq H * (I_H \cup (x_1, z)) \subset H * (R \cap (V_H \times V_H)) = H$, we get $N_H(x_1) \subseteq N_H(z)$. It follows that $(z, x_2) \in E_H$ and hence $e = (z, u) \in G * R_1$. □

Like R-reduced graphs, R-cocores also satisfy a stringent condition on their neighbourhood structure.

Definition 4.4.5. A graph G satisfies property N* if, for every vertex $x \in V_G$, there is no subset $U_x \subseteq V_G \setminus \{x\}$ such that

$$N_G(x) = \bigcup_{y \in U_x} N_G(y)$$

In other words, no neighbourhood can be represented as the union of neighbourhoods of other vertices of graph G.

Proposition 4.4.10. *G is an R-cocore if and only if G has property N*.*

Proof. Consider a vertex set U_x as in Definition 4.4.5 and suppose that there is a vertex $x \in V_G$ such that $N_G(x) = \bigcup_{y \in U_x} N_G(y)$. Then the relation $R := I \setminus (x, x) \cup \{(y, x) : y \in U_x\}$ is an R-coretraction from G/x to G. Thus G is not an R-cocore.

Conversely, suppose that G is not an R-cocore, let H be a coretract of G, and denote by R a minimal R-coretraction of H to G. Then, by Lemma 4.4.9, $R \cap (V_H \times V_H) = I_H$. Consider a vertex $v \in V_G \setminus V_H$ and set $R^{-1}(v) = \{x_1, \ldots, x_i\}$. Then $N(v) = \bigcup_i N(x_i)$, contradicting property N*. □

Lemma 4.4.11. *An R-cocore of graph G is isomorphic to an R-cocore of G_{pd}.*

Proof. Suppose graph H is an R-cocore of graph G, then there exists a coretraction R such that $H * R = G$. By the definition of G_{pd}, we know there exists an R-retraction R' such that $G * R' = G_{pd}$. Then $H * (R \circ R') = G$. It is easily seen that $R \circ R'$ is a coretraction. Also H is the smallest subgraph of G_{pd} which has a coretraction to G_{pd}, otherwise, suppose there are smaller graph K such that K has a coretraction to G_{pd}, then K also has a coretraction to G, which contradicts to the assumption that H is an R-cocore. Therefore, H is an R-cocore of G_{pd}.

Conversely, if H is an R-cocore of graph G_{pd}, then it is also an R-cocore of G. □

We call a subset B of V_G a *basis* of graph G, if for any $x \in V_G$, there exists a subset U_x of B such that $N_G(x) = \bigcup_{y \in U_x} N_G(y)$. Of course a basis of a graph always exists, because V_G is a basis of G. We call a basis of graph G with minimal cardinality a *minimal basis*. Because we consider only finite graphs, a minimal basis always exists. Now we prove that minimal basis of a point-determining graph G is unique.

Lemma 4.4.12. *Minimal basis of a point-determining graph G is unique.*

Proof. Assume there are two distinct minimal basis A, B. Neither contains the other by their minimality. Because A, B are distinct, there exists an element $a \in A$ and $a \notin B$.

Since B is a basis of graph G, by the definition, there is a set $U_a \subset B$ such that $N_G(a) = \bigcup_{b \in U_a} N_G(b)$, thus $|N_G(a)| \geq |N_G(b)|$ for all $b \in U_a$. Since G is a point-determining graph, any two vertices of G do not have the same neighbourhood, thus $|N_G(a)| > |N_G(b)|$ for all $b \in U_a$. Because A is also a basis of graph G, then for any $b \in U_a$, $N_G(b)$ can be represented as the union of the neighbourhood of some elements of $A \setminus \{a\}$, a is not contained since $|N_G(a)| > |N_G(b)|$. Thus $N_G(a) = \bigcup_{b \in U_a} N_G(b)$ can be represented as the union of the neighbourhood of some elements of $A \setminus \{a\}$, which contradicts the minimality of A. □

Lemma 4.4.13. *R-cocore of a point-determining graph is unique.*

Proof. From Lemma 4.4.12, we know that the basis of a point-determining graph is unique. We claim that the vertex set of any R-cocore of graph G is the minimal basis of G. First, if H is an R-cocore, then V_H is a basis of graph G. Because by the definition of R-cocore, there is a coretraction R such that $H * R = G$. For any vertex $x \in V_G$, we assign $U_x = R^{-1} \subset V_H$, it is easily seen that $N_G(x) = \bigcup_{y \in U_x} N_G(y)$.

Then we prove V_H is the minimal basis. Suppose there exists a subset K of V_G is a basis, and the cardinality of K is smaller than V_G. By the definition of basis, for any $x \in V_G$ we fix the set U_x, then there is an R-coretraction from H to G, induced by $R(x) = \{y \in V_G \mid x \in U_y\}$. That is because for any vertex $x \in K$, we have $U_x = \{x\}$. It contradicts to the assumption that H is a cocore.

Therefore the R-cocore of a point-determining graph is unique. □

Because G_{pd} is a point-determining graph, it follows from Lemma 4.4.13 that G_{pd} has unique R-cocore for any graph G. And from lemma 4.4.11 we immediately get the following proposition.

Proposition 4.4.14. *R-cocore of G is unique up to isomorphism.*

These results allow us to construct an algorithm that computes the cocore of given graph G in polynomial time. First observe that the cocore of a graph G that contains isolated vertices is the disjoint union of the cocore of the graph G' obtained from G by removing isolated vertices and the graph consisting of a single isolated vertex. It is thus sufficient to compute cocores for graphs without isolated vertices in Algorithm 2.

Proposition 4.4.15. *Suppose H is a cocore of G. If $K * R = H$, then there is a relation R' such that $K * R' = G$. If $G * S = K$, then there is a relation S' such that $H * S' = K$.*

Algorithm 2 The cocore of a graph

Input:

  Graph G with loops and without isolated vertices specified by its vertex set V
  and the neighbourhoods $N_G(i)$, $i \in V$.

1: **for** $i \in V$ **do**
2:    $W(i) = \emptyset$
3:    **for** $j \in V \setminus \{i\}$ **do**
4:      **if** $N(j) \subseteq N(i)$ **then**
5:        $W(i) := W(i) \cup N(j)$
6:      **end if**
7:    **end for**
8:    **if** $W(i) = N(i)$ **then**
9:      delete i from V
10:     $N(i) = \emptyset$
11:   **end if**
12: **end for**
13: **return** $G[V]$, the cocore of G.

Proof. Since H is a cocore of G, there exists an R-coretraction R_1 such that
$H * R_1 = G$. If $K * R = H$, then letting $R' = R \circ R_1$ implies $K * R' = G$. If
$G * S = K$, we have $H * R_1 * S = K$. Let $S' = R_1 \circ S$, then $H * S' = K$. $\square$

We can find a graph which is not an R-core, and meanwhile it does not satisfy
the Hall condition. To see that, consider the graph G in the example of Fig. 4.4.
Obviously $R = \{(1, 1), (2, 2), (3, 2), (4, 4), (6, 6), (5, 3), (7, 3), (5, 5), (7, 7)\}$ satisfies
$G * R = G$. For the subset $S = \{2, 3\}$ of $V(G)$, $R(S) = \{2\}$, so $|R(S)| < |S|$, it
does not satisfy the Hall condition. But not every graph which are not R-cores do
not satisfy the Hall condition. That means the condition of Corollary 4.2.8 is not
sufficient. To see this, we consider a graph consisting of two independent isolated
vertices.

Note that relations from an R-core or R-cocore to itself also satisfy the Hall
condition.

Proposition 4.4.16. *If G is an R-core, and R is a relation satisfying $G * R = G$,
then any monomorphism $f \subset R$ satisfies $G * f = G$.*

Proof. From corollary 4.2.8 we know that R which is a relation from an R-core
to itself satisfies the Hall condition thus it contains a monomorphism. Suppose
$G * f \neq G$, then $G * f \subset G$. By iteration we have $G * f^m \subset G$ for any m. On the
other hand, f is a permutation, then there exists a n such that f^n is identity, thus
$G * f^n = G$. A contradiction. $\square$

Without the constraint of R with full domain, we have:

Proposition 4.4.17. *All solutions of $G * R = G$ satisfy the Hall condition if and only if G is an R-cocore.*

Proof. If G is an R-cocore, then all the solutions of $G * R = G$ are with full domain. Note that G is an R-cocore thus it is also an R-core, from Corollary 4.2.8 we know that all solutions R satisfy the Hall condition.

Conversely, if all solutions satisfy the Hall condition, then all the solutions have full domains, Hence G is an R-cocore, otherwise we can find an induced subgraph H of G and a relation R satisfying $H * R = G$. This relation R is also the solution of $G * R = G$ in general (without constraint of full domain) case. □

Actually, it is very easy to get the following result which characterize the cases when all the solutions satisfy the Hall condition:

Proposition 4.4.18. *If $G * R = G$, and all solutions satisfy the Hall condition, then either of the two cases happens:*

1. *G is an R-core;*

2. *There is a solution contains a relation with non-full domain.*

Inclusion relation

Figure 4.9 shows the inclusion relation of cores, R-cores, R-cocores, graphs with property N and point-determining graphs: the set of cores is a proper subset of the set of graphs with property N, which is a proper subset of R-cocores. The latter is a proper subset of R-cocores, the set of R-cores is a proper subset of the set of point-determining graphs.

Example 4.4.1. We also give a list of examples to illustrate the proper inclusion relations, some of the examples have been already shown or explained before:

1. K_3 is a core, then of course it has property N and is an R-core, R-cocore and point-determining graph.

2. C_6 has property N but is not a core. Its core is P_2.

3. P_4 is an R-cocore but not with property N.

4. The graph H in Figure 3.2 is an R-core but not an R-cocore, its R-cocore is P_4.

5. The graph in Figure 4.4a is a point-determining graph but not an R-core.
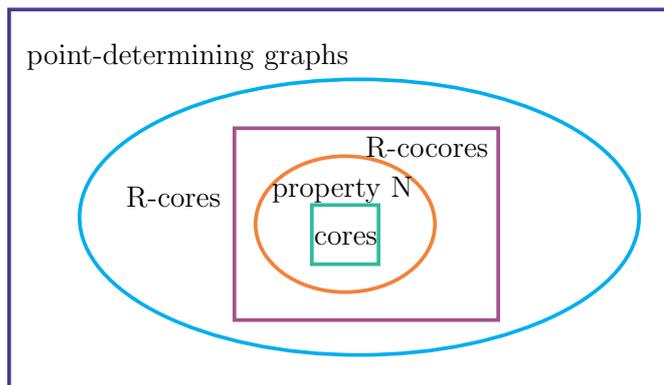
Figure 4.9: Inclusion digram

4.5   Computational complexity

In this section we briefly consider the complexity of computational problems related to graph homomorphisms. For the readers who are not familiar with computational complexity, please refer to Appendix A. The *homomorphism problem* $\textsc{Hom}(H)$ takes as input some finite G and asks whether there is a homomorphism from G to H. The computational complexity of the homomorphism problem is fully characterized. It is known that $\textsc{Hom}(H)$ is NP-complete if and only if H has no loops and contains odd cycles. All the other cases are polynomial, see [40].

The analogous problem for relations between graphs can be phrased as follows: The *(full) relation problem* $\textsc{Ful-Rel}(H)$ takes as input some finite G and asks whether there is a relation with full domain from G and asks whether there is a relation from G to H. We show that this problem can be easily converted to a related problem on surjective homomorphisms. The *surjective homomorphism problem* $\textsc{Sur-Hom}(H)$ takes as input some finite G and asks whether there is a surjective homomorphism from G to H.

Let $\leq_{\mathrm{P}}^{\mathrm{Tur}}$ indicate polynomial time Turing reduction.

Theorem 4.5.1. *For finite H our relation problem sits in the following relationship.*

$$\textsc{Hom}(H) \leq_{P}^{Tur} \textsc{Ful-Rel}(H) \leq_{P}^{Tur} \textsc{Sur-Hom}(H). \qquad (4.2)$$

Proof. First we show that $\textsc{Hom}(H)$ is polynomially reducible to $\textsc{Ful-Rel}(H)$. If there is a homomorphism from G to H, then there is also a surjective homomorphism from $G + H$ to H. On the other hand, suppose G has no homomorphism to H, from Lemma 3.4.1 we conclude that $G + H$ has no relation to H since G has no relation to H.

Now we show that the relation problem $\textsc{Ful-Rel}(H)$ is polynomially reducible to $\textsc{Sur-Hom}(H)$. From Corollary 4.1.3 we know $G * R = H$ if and only if there is a graph $G' = G * R_D$ which has a full homomorphism to G and has a surjective homomorphism to H.

We construct G'', by duplicating all the vertices of G precisely $|V_H|$ times. We claim that G' exists if and only if G'' has a full homomorphism to G. We can put $G' = G''$ because the surjective homomorphism can easily undo the redundant duplications. This gives the polynomial reduction from $\textsc{Ful-Rel}(H)$ to $\textsc{Sur-Hom}(H)$. $\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\quad\square$

To our knowledge, $\textsc{Sur-Hom}(H)$ is not fully classified. A recent survey of the closely related complexity problem concerning the existence of vertex surjective homomorphisms [8] provides some arguments why the characterization of complexity is likely to be hard. We observe that the existence of a homomorphism from G to H is equivalent to the existence of a surjective homomorphism from $G + H$ to H. Thus $\textsc{Sur-Hom}(H)$ is clearly hard for all graphs for which $\textsc{Hom}(H)$ is hard, i.e., for all loop-less graphs with odd cycles.

Testing the existence of a homomorphism from a fixed G to H is polynomial (there is only a polynomial number $|V_H|^{|V_G|}$ of possible functions from G to H). Similarly the existence of a relation from a fixed G to H is also polynomial. In fact, an effective algorithm exists. For fixed G there are finitely many point-determining graphs T which G has relation to. The algorithm thus first constructs the point-determining graph of H and then, using a decision tree recognizes all isomorphic copies of all point-determining graphs G has relation to.

4.6   Weak relational composition

In this section we will briefly discuss the 'loop-free' version, i.e., equations of the form $G \star R = H$.

Most importantly, there is no simple composition law analogous to Lemma 4.1.1. The expression

$$(G \star R) \star S = (S^+ \circ (R^+ \circ G \circ R)^\iota \circ S)^\iota$$

does not reduce to relational composition in general. For example, let $G = K_3$ with vertex set $V = \{x, y, z\}$ and consider the relations $R = \{(x, 1), (z, 1), (y, 2)\} \subseteq \{x, y, z\} \times \{1, 2\}$ and $S = \{(1, x')(1, z')(2, y')\} \subseteq \{1, 2\} \times \{x', y', z'\}$. One can easily verify

$$(G \star R) \star S = P_2 \neq G \star (R \circ S) = K_3$$

The most important consequence of the lack of a composition law is that R-retractions cannot be meaningfully defined for the weak composition. Similarly, the results related to R-equivalence heavily rely on the composition law.

Nevertheless, many of the above results, in particular basic properties derived in Section 4.1, remain valid for the weak composition operation. As the proofs are in many cases analogous, we focus here mostly on those results where strong and weak composition differ, or where we need different proofs. In particular, Lemma 4.1.2 also holds for the weak composition. Thus, we still have a result similar to Corollary 4.1.3, but the proof is slightly different.

Corollary 4.6.1. *Suppose $G \star R = H$. Then there is a set C, an injective relation $R_D \subseteq \operatorname{dom} R \times C$, and a surjective relation $R_C \subseteq C \times \operatorname{img} R$ such that $G[\operatorname{dom} R] \star R_D \star R_C = H[\operatorname{img} R]$.*

Proof. From Proposition 4.1.2 we know $R = I' \circ R_D \circ R_C \circ I''$. And we know $G[\operatorname{dom} R] \star R_D = G[\operatorname{dom} R] * R_D$. From the properties of $\star$, we have

$$
\begin{aligned}
G[\operatorname{dom} R] \star R &= (R^+ \circ G[\operatorname{dom} R] \circ R)^l \\
&= ((R_D \circ R_C)^+ \circ G[\operatorname{dom} R] \circ R_D \circ R_C)^l \\
&= (R_C^+ \circ R_D^+ \circ G[\operatorname{dom} R] \circ R_D \circ R_C)^l \\
&= (R_C^+ \circ (R_D^+ \circ G[\operatorname{dom} R] \circ R_D) \circ R_C)^l \\
&= (R_C^+ \circ G[\operatorname{dom} R] * R_D \circ R_C)^l \\
&= (R_C^+ \circ G[\operatorname{dom} R] \star R_D \circ R_C)^l \\
&= G[\operatorname{dom} R] \star R_D \star R_C \\
&= H[\operatorname{img} R]. \qquad \square
\end{aligned}
$$

Assume $G \star R = H$ and let $H_1, \ldots, H_k$ be the connected components of H. From the definition of $\star$ and $*$, if we denote $\widetilde{H} = G * R$, then $\widetilde{H}$ could be decomposed into the union of connected components $\widetilde{H}_i (1 \leq i \leq k)$, such that $(\widetilde{H}_i)^\iota = H_i$. Hence the conclusion of the proposition 4.1.4 also holds true for weak relations.

Lemma 4.1.6 does not hold for weak relations. For example, there is a weak relation of K_5 to K_3, but $\chi(K_5) = 5 > \chi(K_3) = 3$.

Lemma 4.1.7 and Lemma 4.1.8 do not hold for weak relations. For example, if G is a graph consisting of a single isolated vertex isolated, then $P_3 \star R = G$ and $C_3 \star R = G$, but there are no walk in G.

With respect to complete graphs, weak relational composition also behaves differently from strong composition. If $K_k \star R = H$ then $R(i)$ can contain more than one vertex in V_H. Comparing to Proposition 4.1.11, we also obtain a different result:

Theorem 4.6.2. *There is a relation R such that $K_k \star R = H$ if and only if every connected component of $\overline{H}$ is a complete graph, and the number of connected components of $\overline{H}$ containing at least 2 vertices is at most k.*

Proof. If every connected component of $\overline{H}$ is a complete graph, we denote the vertex sets of the connected components containing at least 2 vertices by $H_1, \ldots, H_m$, $m \leq k$ and the vertices of K_k by $1, \ldots, k$. Let $R = \{(i, u) \mid i = 1, \ldots, k, u \in V_{H_i}\} \cup \{(j, v) : 1 \leq j \leq k, v \in V_H \setminus \bigcup_{i=1}^m V_{H_i}\}$. One easily checks that $K_k \star R = H$.

Conversely, let R be a relation satisfying $K_k \star R = H$. Consider the set $U_i = \{u \in V_H \mid R^{-1}(u) = \{i\}\}$. Then u and v are not adjacent for arbitrary $u, v \in U_i$, while u is adjacent to w for every $w \in V_H \setminus U_i$. Hence $\overline{H}(U_i)$ is a connected component of $\overline{H}$, which is also a complete graph. Given $w \in V_H \setminus \bigcup_{i=1}^m U_i$, $R^{-1}(w)$ must have at least 2 vertices in K_k, hence w is adjacent to every vertex in H except itself; in other words, w is an isolated vertex in $\overline{H}$. Therefore the number of connected components of $\overline{H}$ containing at least 2 vertices is no more than k. $\square$

The results in subsection 4.2.1 also remain true for weak relations.

Chapter 5

Constrained Homomorphisms Orders

While homomorphism orders have been a fruitful research topic for several decades, relatively little effort has been put into the study of orders induced by constrained homomorphisms. The aim of this chapter is to establish some of the properties of these orders and their similarities and differences. We survey existing results and provide several new results about cores, universality, gaps and dualities for constrained homomorphisms.

Due to the nature of our topic, many of the results turn out to be simple observations. With a systematic analysis of related orders we can often establish the properties of more complex structures by using the properties of easier structures. We give several non-trivial results.

We spend the first two sections developing new methods that we shall use throughout the chapter. In Section 5.1 we introduce notions of past-finite-universal and future-finite-universal orders, and provide a practical way to prove past-finite universality or future-finite universality of orders and then from these to build universality orders. In Section 5.2 we give a standard approach for the characterization of density, gaps and duality of past-finite or future-finite orders.

In Section 5.3 we recall some results about the homomorphism order. We also give a new and significantly easier proof of the universality of graph homomorphisms (simplifying the one given by the Hubička and Nešetřil [43]). For the first time we show that the homomorphism order is universal even on the class of oriented cycles. We use this result to resolve the universality of locally constrained homomorphisms. In later sections we summarize the properties of constrained homomorphism orders. In Section 5.5 we characterize gaps in the full

homomorphism order and give a new and simple proof of the existence of left
duals (simplifying the results independently obtained by Feder and Hell [19] and
Ball, Nešetřil, Pultr [3, 4]). In Section 5.7 we prove the universality of locally
injective homomorphisms on connected graphs and give a partial characterization
of gaps. We also prove the universality of all three kinds of locally constrained
homomorphisms on graphs.

In Section 5.8 we prove that for every $d \geq 3$ the homomorphism order on the
class of line graphs of graphs with maximum degree d is universal.

In Section 5.9, using notions derived from Chapter 4, we characterize cores
for relation orders. We also describe their gaps and dualities by extending the
corresponding results for surjective homomorphism orders.

NOTATION

The most generic way to think of structures induced by constrained homomor-
phisms is in terms of category theory and relational structures [71]. Because we
are however interested in specific examples of orders induced by specific types of
constrained homomorphisms, we believe that it makes the thesis more readable to
use the standard language of graph theory and graph homomorphisms with just
the following extension.

We use the abbreviations we assigned to each constrained homomorphism (such
as F-morphism for the full homomorphisms) to denote the variant of the order of
interest. We write $G \to^* H$ and $G \leq^* H$ to denote the existence of $*-morphism$
from G to H. For example, $G \to^F H$ denote the existence of a full homomorphism.

We use the analogous notation for other concepts introduced above. For
example, just as the core is the smallest representative of its equivalence class
of $\sim$, the $*$-core is the smallest representative of its equivalence class of $\sim^*$ (that
is, the equivalence induced on DiGraphs by $\to^*$).

5.1   Universal orders

It is a well-known fact that every finite order $(P, \leq_P)$ can be represented by the
inclusion order on finite sets. More precisely, for every finite order $(P, \leq_P)$ there
exists an embedding E assigning to every element $x \in P$ a finite set $E(x)$ such
that

$$x \leq_P y \text{ if and only if } E(x) \subseteq E(y) \text{ for every } x, y \in P.$$

The easiest way to obtain this embedding is to put $E(x) = \downarrow x$.

The choice of $(P, \leq_P)$ is arbitrary and thus we always assume that P is a subset
of a fixed countable set, for example $\mathbb{N}$ (the set of natural numbers). Consequently

the order formed by all finite sets of natural numbers ordered by the inclusion contains every finite order as a suborder.

Countable orders containing every finite order as a suborder are *finite-universal*. We denote by $P_{\mathrm{fin}}(A)$ the set of all finite subsets of A. We can now state the following folklore fact.

Proposition 5.1.1 (The existence of future-finite orders). *For any countably infinite set A, $(P_{\mathrm{fin}}(A), \subseteq)$ is a finite-universal order.*

Finite-universality orders have a rich structure. It immediately follows that they are of infinite dimension and contain finite chains, antichains and decreasing chains of arbitrary length. While finite-universal orders can be seen as 'rich' specimens, they are rather easy to construct. There are only countably many finite orders (up to isomorphism). Thus one can consider an order formed from the disjoint union of all finite orders. However this is no longer true when countable universality is considered. A countable order is *universal* if it contains every countable order as a suborder. The existence of universal orders seems to be a counter-intuitive fact. Countable orders have a rich structure and there are uncountably many of them, yet all of them can be 'packed' into a single countable structure.

The easiest way to show the existence of a universal order is by the Fraïssé Theorem [30] which leads to an implicit construction of such order. We will give a constructive proof of its existence. Having an explicit representation makes the embedding into (constrained) homomorphism orders easy.

We build a universal order in two layers.

1. First we turn our attention to a 'lesser' notion of universality. Recall that in Chapter 2 we defined that a countable order is *past-finite* if every down-set is finite. Similarly a countable order is *future-finite* if every up-set is finite. Again, we say a countable order is *past-finite-universal*[1] (or *future-finite-universal*) if it contains every past-finite (or future-finite) order as a suborder. We now strengthen Proposition 5.1.1.

Lemma 5.1.2. *For any countably infinite set A, $(P_{\mathrm{fin}}(A), \subseteq)$ is a past-finite-universal order.*

Proof. Fix a past-finite ordered set $(P, \leq_P)$. Without loss of generality we assume that $P \subseteq A$. Consider the same mapping as in the proof of Proposition 5.1.1, that

---

[1] 'Past-finite' or 'future-finite' is a standard name used in the context of study of causal sets (that is an alternative name for locally finite partial orders). We use these names due to the apparent lack of a standard name of these properties in the order theoretical context. Note that past finiteness is different from well-quasi-ordering.

is to assign to every $x \in P$ a set $E(x) = \downarrow x$. It is easy to verify that E is an embedding from $E : (P, \leq_P) \to (P_{\mathrm{fin}}(A), \subseteq)$. $\qquad \square$

Because every past-finite order turns to be a future-finite order when the direction of inequalities is reversed, we immediately obtain:

Corollary 5.1.3. *Let A be any countably infinite set. Then $(P_{\mathrm{fin}}(A), \supseteq)$ is a future-finite-universal order.*

2. For a given order $(P, \leq_P)$ we construct its *subset order*, $(P_{\mathrm{fin}}(P), \leq_P^{\mathrm{dom}})$, by putting

$$A \leq_P^{\mathrm{dom}} B \iff \text{ for every } a \in A \text{ there exists } b \in B \text{ such that } a \leq_P b$$

(in this case we say *A is dominated by B* in $(P, \leq_P)$).

It is easy to see that for every order $(P, \leq_P)$ the relation $\leq_P^{\mathrm{dom}}$ (on finite subsets of P) is transitive and reflexive and thus forming a quasi-order. We again choose a unique representative for every equivalence class to get an order. $A \in P_{\mathrm{fin}}(P)$ is a representative if for every $a, b \in A$, $a \neq b$ the elements a and b are incomparable in $(P, \leq_P)$ (that is, A is an antichain in $(P, \leq_P)$).

As a consequence of Lemma 5.1.2 we easily obtain:

Corollary 5.1.4. *Let $(P, \leq_P)$ be a countably infinite discrete order (that is the relation $\leq_P$ is empty). Then $(P_{\mathrm{fin}}(P), \leq_P^{\mathrm{dom}})$ is a past-finite-universal order.*

If $(P, \leq_P)$ is a countably infinite discrete order, then for any $A, B \subseteq P$, $A \leq_P^{\mathrm{dom}} B$ if and only if for any $a \in A$, there exists a $b \in B$ such that $a = b$, this is true if and only if $A \subseteq B$. For this reason, the order $(P_{\mathrm{fin}}(A), \subseteq)$ is in fact a special case of subset orders.

More surprisingly, the subset orders can be layered to create a universal order.

Theorem 5.1.5. *For every future-finite-universal order $(F, \leq_F)$, the order $(P_{\mathrm{fin}}(F), \leq_F^{\mathrm{dom}})$ is universal.*

Proof. Take an arbitary order $(P, \leq_P)$. Without loss of generality we can assume that $P \subseteq \mathbb{N}$. This way we enforce the linear order $\leq$[1] on the elements of P. Note that the linear order $\leq$ is completely unrelated to the order $\leq_P$. We can think of $\leq$ as a specification of the time of creating the elements of P.

We define two new orders on the elements of P: $\leq_f$ (*forwarding order*) and $\leq_b$ (*backwarding order*):

1. We put $x \leq_f y$ if and only if $x \leq_P y$ and $x \leq y$.

---

[1]Note that here $\leq$ does not mean the existence of homomorphisms.

2. We put $x \leq_b y$ if and only if $x \leq_P y$ and $x \geq y$.

Thus we decompose the order $(P, \leq_P)$ to $(P, \leq_f)$ and $(P, \leq_b)$. For every element $x \in P$ both of the sets $\{y \mid y \leq_f x\}$ and $\{y \mid x \leq_b y\}$ are finite. It follows that $(P, \leq_f)$ is past-finite and $(P, \leq_b)$ is future-finite.

Since $(F, \leq_F)$ is future-finite-universal, there is an embedding $E : (P, \leq_b) \to (F, \leq_F)$. For every $x \in P$, we put

$$U(x) = \{E(y) \mid y \leq_f x\}.$$

We show that $U : (P, \leq_P) \to (P_{\text{fin}}(F), \leq_F^{\text{dom}})$ is an embedding.

First we show that $U(x) \leq_F^{\text{dom}} U(y)$ implies $x \leq_P y$. From the definition of $\leq_F^{\text{dom}}$ we know that there is an element $w \in P$, $E(w) \in U(y)$, such that $E(x) \leq_F E(w)$. By the definition of U, $E(w) \in U(y)$ if and only if $w \leq_f y$. By the definition of E, $E(x) \leq_F E(w)$ if and only if $x \leq_b w$. It follows that $x \leq_b w \leq_f y$ and thus also $x \leq_P w \leq_P y$ and consequently $x \leq_P y$.

To show that $x \leq_P y$ implies $U(x) \leq_F^{\text{dom}} U(y)$ we consider two cases.

1. When $x \leq y$. Then $U(x) \subseteq U(y)$ and thus $U(x) \leq_F^{\text{dom}} U(y)$.

2. When $x > y$. Then we take any $w \in P$, $E(w) \in U(x)$. From the construction of $U(x)$ we know that $w \leq_f x \leq_P y$. If $w \leq y$, then $E(w) \in U(y)$. In the other case we have $w \leq_b y$ and thus $E(w) \leq_F E(y)$. It follows that $U(x)$ is dominated by $U(y)$. □

It is easily seen that the embeddings constructed in proofs of Theorem 5.1.5 have the property that the embedding of vertex x depends only on vertices $\{y \mid y < x\}$. Such embeddings are known as *on-line embeddings* because they can be constructed inductively without a priori knowledge of the whole order. See for example [42]. All embeddings shown in this chapter are in fact on-line.

Example 5.1.1. Consider the order $(P, \leq_P)$ as specified by Figure 5.1. Denote by $(P, \leq_f)$ the order consisting of forwarding inequalities in $(P, \leq_P)$ and by $(P, \leq_b)$ the order of backwarding inequalities. By Corollary 5.1.3 we can obtain the following on-line embedding f from $(P, \leq_b)$ to $(P_{\text{fin}}(\mathbb{P}), \supseteq)$, for any element $x \in P$, $f(x) = \{y \mid x \leq_b y\}$:

$$f(3) = \{3\}, f(5) = \{5\}, f(7) = \{3, 5, 7\}, f(11) = \{5, 11\}.$$

By Theorem 5.1.5 we get the following on-line embedding U from $(P, \leq_P)$ to $(P_{\text{fin}}(P_{\text{fin}}(\mathbb{P})), \leq_{\supseteq}^{\text{dom}})$, for any element $x \in P$, $U(x) = \{f(y) \mid y \leq_f f(x)\}$:

$$U(3) = \{\{3\}\}, U(5) = \{\{5\}, \{3\}\},$$

$$U(7) = \{\{3, 5, 7\}\}, U(11) = \{\{3, 5, 7\}, \{5, 11\}\}.$$
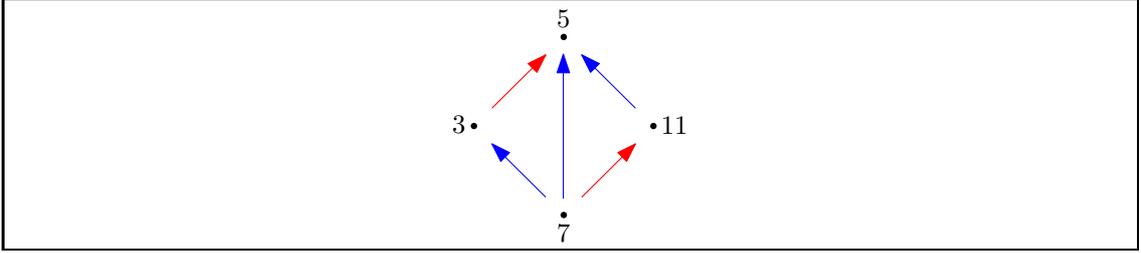
Figure 5.1: The order $(P, \leq_P)$. Blue lines represent backwarding edges

We remark that the universality of the order $(P_{\text{fin}}(F), \leq_F^{\text{dom}})$ and the proof of Theorem 5.1.5 are shown in several works on universal orders. The precise formulations differ, but the key ideas remain. To the best of our knowledge Hedrlín first introduced equivalent structure as an example of the universal order in [34]. In [64] the structure was re-discovered to show the universality of the homomorphism order of set-systems (for the first time, its universality was shown by on-line embedding). In [45] it is used to show the universality of the homomorphism order of oriented paths (Theorem 5.3.4). Finally in [42] a more streamlined variant of the same structure appears as a tool to show the universality of multiple naturally defined orders.

Here we use a more layered approach with the notions of future-finite-universal and past-finite-universal sets. The existence of past-finite-universal structures has been shown in a less constructive way in [15]. We apply these notions to several universality results on the constrained homomorphism order.

5.2   Past-finite and future-finite orders

Many constrained homomorphism orders we consider have finite down-sets or up-sets. We thus generally characterize the orders into three categories: universal, past-finite universal or future-finite universal.

In this section we make some basic observations about past-finite and future-finite orders. Both types of orders are special cases of locally-finite order [10, 15]. An order $(P, \leq_P)$ is said to be *locally-finite* if for every $a, b \in P$, $a \leq_P b$, the interval $[a, b]_P = \{c \mid a \leq_P c \leq_P b\}$ is finite. This immediately gives the following result.

Proposition 5.2.1 (Density of locally-finite orders). *If $(P, \leq_P)$ is a past-finite, future-finite or locally-finite order, then $(P, \leq_P)$ is not dense.*

For every $u \in P$ that is not maximal in $(P, \leq_P)$, there is an element $v \in P$ such that (u, v) is a gap in $(P, \leq_P)$. For every $u' \in P$ that is not minimal in $(P, \leq_P)$, there is a $v' \in P$ such that (v', u') is a gap in $(P, \leq_P)$.

The concept of duality was studied for some past-finite and future-finite partial orders, see [4, 19, 36, 89]. We simplify our presentation by the following simple observation that holds for all future-finite and past-finite orders.

The *generalized finite duality pair* $(\mathcal{F}, \mathcal{D})$ in $(P, \leq_P)$ can be seen as an equation:

$$\bigcup_{f \in \mathcal{F}} \uparrow f = P \setminus \bigcup_{d \in \mathcal{D}} \downarrow d.$$

This is depicted in Figure 5.2.
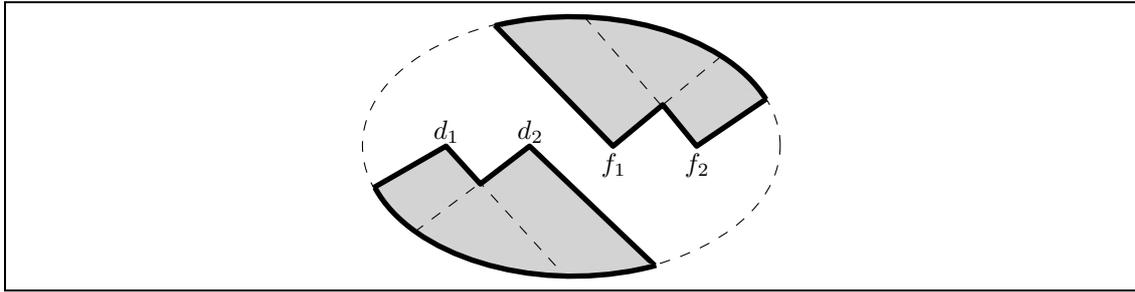
Figure 5.2: Generalized duality pair $(\{f_1, f_2\}, \{d_1, d_2\})$.

If we assume that $(P, \leq_P)$ is future-finite, it follows that left hand side of the equation is always finite, while the right hand side may be infinite. Obviously $(\mathcal{F}, \mathcal{D})$ can be a duality pair only if both sides of the equation describe a finite set and thus only for relatively degenerated choices of set $\mathcal{D}$ one can find a set $\mathcal{F}$ such as $(\mathcal{F}, \mathcal{D})$ is a generalized finite duality pair.

In the case of future-finite orders we will thus ask for the characterization of sets $\mathcal{F}$ with $\mathcal{D}$ forming a generalized finite duality pair $(\mathcal{F}, \mathcal{D})$. We show that the existence of finite duality pairs is closely related to the existence of gaps.

Proposition 5.2.2 (Dualities in a future-finite order). *Let $(P, \leq_P)$ be a future-finite order and $\mathcal{F}$ a finite subset of P. There is a set $\mathcal{D}$ such that $(\mathcal{F}, \mathcal{D})$ is a generalized finite duality pair in $(P, \leq_P)$ if and only if*

1. *for every element a in $\bigcup_{f \in \mathcal{F}} \uparrow f$ there are only finitely many elements b in P such that (b, a) is a gap in $(P, \leq_P)$,*

2. *$(P, \leq_P)$ has only finitely many maximal elements.*

The situation is schematically depicted in Figure 5.3.

Proof. Put

$$A = \bigcup_{f \in \mathcal{F}} \uparrow f,$$
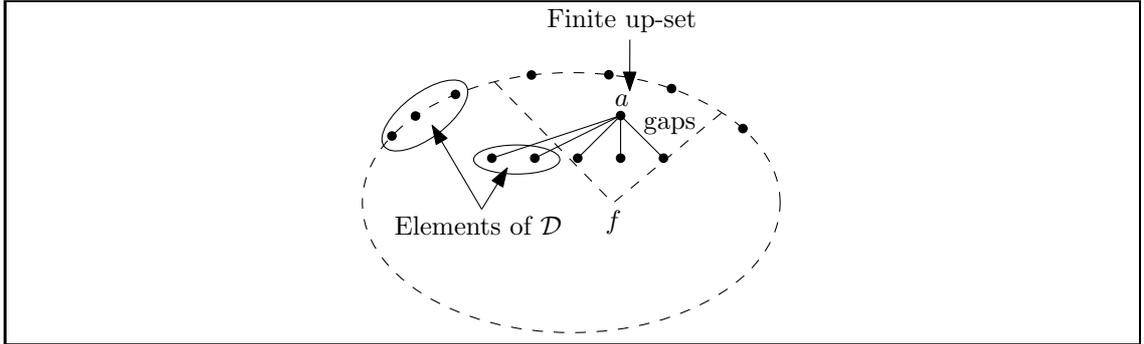
Figure 5.3: Dualities in a future-finite order.

$$M = \{m \mid m \text{ is a maximal element of } (P, \leq_P) \text{ and } m \notin A\}, \text{ and}$$

$$\mathcal{D} = \{b \mid b \in P \backslash A \text{ and there exists } a \in A \text{ such that } (b, a) \text{ is a gap in } (P, \leq_P)\} \cup M.$$

It follows from Proposition 5.2.1 and the future-finiteness of $(P, \leq_P)$ that $\mathcal{D}$ is the smallest set such that $(\mathcal{F}, \mathcal{D})$ is a generalized finite duality pair.

The conditions 1 and 2 in the statement of this proposition are precisely the ones assuring that $\mathcal{D}$ is finite, thus $(\mathcal{F}, \mathcal{D})$ is a generalized finite duality pair. $\quad\square$

We say that there are *all right generalized finite dualities* in $(P, \leq_P)$ if for every finite set $\mathcal{F} \subseteq P$ there is a finite $\mathcal{D} \subseteq P$ such that $(\mathcal{F}, \mathcal{D})$ is a generalized finite duality pair.

Corollary 5.2.3. *If $(P, \leq_P)$ is a future-finite order such that for every $a \in P$ there is only finitely many $b \in P$ such that (b, a) is a gap in $(P, \leq_P)$, then there are all right generalized finite dualities in $(P, \leq_P)$.*

In an order $(P, \leq_P)$, every element $a \in P$ with only one $b \in P$ such that (a, b) is a gap in $(P, \leq_P)$ is called *left realization* or *left realization of b*. Symmetrically *right realization* is the element $a \in P$ with only one $b \in P$ such that (b, a) is a gap.

The following observation (that was a special case about full homomorphisms used by Feder and Hell in [19]) follows from the construction of $\mathcal{D}$ shown above.

Proposition **5.2.4.** *Let $(P, \leq_P)$ be a future finite order and $\mathcal{F} = \{f\}$ a set consisting of single element of P. For inclusion minimal $\mathcal{D}$ such that $(\mathcal{F}, \mathcal{D})$ is a generalized finite duality pair, the only elements $r \in \mathcal{D}$ such that (r, f) is a gap in $(P, \leq_P)$ are left realizations of f.*

From the symmetry, in the context of past-finite orders the question about the existence of duality pairs $(\mathcal{F}, \mathcal{D})$ for a given $\mathcal{D}$ is more interesting than the converse question. As we will show, some of the orders we consider have duality for every choice of $\mathcal{F}$, while others do not.

We also remark that one of the main motivations for graph dualities — the connection to computational complexity— does not hold in the case of past-finite and future-finite orders; the set of objects described by dualities is finite and thus it is always easy to write algorithms recognizing them for a fixed generalized finite duality pair. Yet those types of question are interesting, see, for example [3, 4] where the existence of such dualities is related to the concept of Ramsey lists, see also [19, 89].

5.3 Homomorphism orders

For two graphs G and H, we put a relation $G \to H$ if there is a graph homomorphism of G to H. The relation $\to$ is reflexive (identity is a homomorphism) and transitive (composition of two homomorphisms is still a homomorphism). Thus the existence of homomorphisms induces a quasi-order on the class of all finite directed graphs. We denote this quasi-order by $(\mathsf{DiGraphs}, \leq)$. When speaking of orders, we will thus use $G \leq H$ in the same sense as $G \to H$. This allows us to write shortly $G < H$, $G > H$ and $G \geq H$ with the obvious meanings. In this section we first outline the main properties of quasi-orders induced by the existence of homomorphisms on Graphs and $\mathsf{DiGraphs}$; for a detailed discussion, see [40, Chap. 3]. Also we revisit universality on restricted graph classes and give a new proof of universality of the homomorphism order on the class of oriented cycles.

There are two standard ways to transform a quasi-order to an order—by identifying equivalent objects, or by choosing a particular representative for each equivalence class. We do the latter; cores fit the purpose perfectly. Recall that in Chapter 3 we defined that a (finite) directed graph is a *graph core* when it is the smallest graph (in the number of vertices) in its equivalence class of $\sim$ (also see [40]). It can be shown that every equivalence class of $\sim$ contains, up to isomorphism, a unique core. The *core of graph G* is an up to isomorphism unique graph H such that $H \sim G$ and H is a core. Further the core of graph G is always an induced subgraph of G, see [40]. Consequently it makes sense to speak of cores in both $\mathsf{DiGraphs}$, Graphs and in any other class of graphs which is hereditary (i.e. closed under taking induced subgraphs). We shall note that the graph core is usually defined in terms of retractions. We use an alternative, but equivalent, definition for the sake of carrying the term more easily over into the context of constrained homomorphism orders. The relation $\to$ induces an order on the class of all isomorphism types of cores. We take the liberty of denoting this order in the same way as the quasi order, that is by $(\mathsf{DiGraphs}, \leq)$, and implicitly assume that the class $\mathsf{DiGraphs}$ is restricted to cores. We call this order the *homomorphism order of directed graphs*. We will make similar assumptions on all other constrained

homomorphism orders discussed thoroughout the chapter.

Both homomorphism orders $(\mathsf{DiGraphs}, \leq)$ and $(\mathsf{Graphs}, \leq)$ have been extensively studied.

Universality

It is a relatively easy exercise to find an infinite antichain, a descending chain and an increasing chain in $(\mathsf{Graphs}, \leq)$. This suggests that the homomorphisms induce a rich structure on the class of graphs. It is however a non-trivial result that every countable order can be found as a suborder of $(\mathsf{DiGraphs}, \leq)$. The initial result has been proved in the even stronger setting of category theory, see [71]. Subsequently it has been shown that many restricted classes of graphs and similar structures (such as homomorphisms of set systems [64], oriented trees [43], oriented paths [45], orders and lattices [53]) admit this universality property.

The following result was the culmination of a series of papers in the context of categories.

Theorem 5.3.1 ([71]). *Any countable order is isomorphic to a suborder of* $(\mathsf{DiGraphs}, \leq)$.

This is a non-trivial result and it can be strengthened to say that $(\mathsf{Graphs}, \leq)$ is universal.

Density and duality

Recall in Chapter 2 we introduced the definition of density, gaps and duality of an order. In this section we consider these properties for two orders: $(\mathsf{Graphs}, \leq)$ and $(\mathsf{DiGraphs}, \leq)$.

DENSITY AND GAPS

As a special case, the order of rational numbers, $(\mathbb{Q}, \leq)$, can be embedded into the universal order $(\mathsf{Graphs}, \leq)$. It easily follows that there are graphs $G < H$ such that the set of graphs $\{H' | G < H' < H\}$ contains infinitely many mutually non-isomorphic graph cores. Consider, for example, the images of 0 and 1 in $(\mathsf{Graphs}, \leq)$. On the other hand, there are no graphs strictly in between K_1 (that is the graph formed by a single isolated vertex) and K_2 (that is, two vertices connected by an edge). This is written as the following theorem:

Theorem 5.3.2 (Density [67]). *Given two simple graphs G_1, G_2 with $G_1 < G_2$, $G_1 \leq O$ and $G_1 \leq K_1$, there is a graph G satisfying $G_1 < G < G_2$.*

There are three proofs of this result: a probabilistic proof [67], a constructive proof via products due to M. Perles and J. Nešetřil [68], and the third one relies on a result about duality which we will introduce later in this section.

All gaps in both (Graphs, ≤) and (DiGraphs, ≤) have been characterized [64, 87]. It follows that (K_1, K_2) is the only gap in (Graphs, ≤).

By contrast, the order (DiGraphs, ≤) has many gaps. In this case, the existence of H' can be shown only for graphs H containing an oriented cycle. We also mention that density in other classes of graphs has also been studied, for example, C. Tardif [80] proved the class of vertex-transitive graphs is dense.

(Generalized) duality

The structure of gaps in (DiGraphs, ≤) is related to another concept of graph dualities. We say that a pair of directed graphs (F, D) is a *simple duality pair* in class $\mathcal{K}$ if

$$G \nrightarrow D \text{ if and only if } F \to G \text{ for all } G \in \mathcal{K}.$$

An example of a such duality pair is given in Figure 5.4.
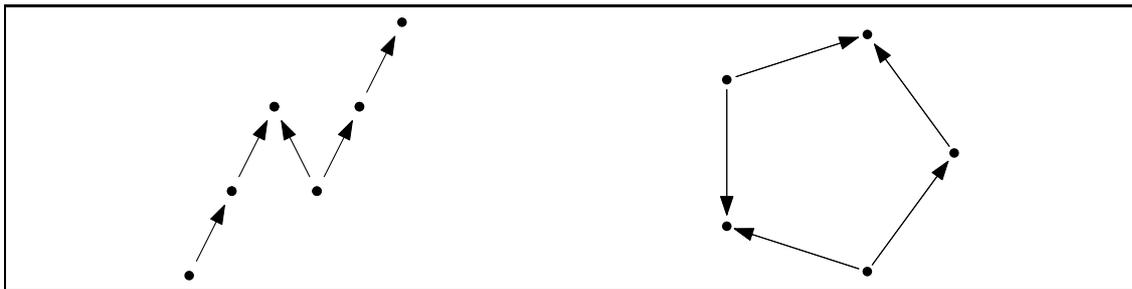
Figure 5.4: Duality pair.

For homomorphism dualities the class $\mathcal{K}$ is usually implicitly Graphs or DiGraphs. In both cases dualities are in one-to-one correspondence with gaps [64]. It is easy to see that (K_2, K_1) is a simple duality pair and in fact it is the only duality pair in (Graphs, ≤). In (DiGraphs, ≤) there exists a duality pair (F, D) if and only if F is an oriented tree (that is, when core of F contains no oriented cycle).

Graph duality is an important concept of graph theory. It provides better understanding of other structural properties, such as maximal antichains [61], computational complexity properties of homomorphism testing, etc. Several variants of graph duality have been studied.

Considering classes $\mathcal{K} \subset$ DiGraphs leads to the notion of restricted dualities [58]. For two finite sets of directed graphs $\mathcal{F}$ and $\mathcal{D}$ we say that $(\mathcal{F}, \mathcal{D})$ is a *generalized finite duality pair* in $\mathcal{K}$ if for any directed graph $G \in \mathcal{K}$ there exists $F \in \mathcal{F}$ such

that $F \to G$ if and only if $G \to D$ for no $D \in \mathcal{D}$ [29]. Recently the infinite-finite dualities have been considered and partially characterized. In a *generalized infinite-finite duality pair* the set $\mathcal{F}$ can be infinite. See [17] for details.

Maximal antichains

From the definition, a simple duality pair is a maximal antichain of size two. Since every other digraph admits either a homomorphism to H or a homomorphism from F. It is not known whether there are other maximal antichains of size $t \geq 2$ in $(\mathsf{DiGraphs}, \leq)$. The following propostion characterises all maximal antichains of size one in $(\mathsf{DiGraphs}, \leq)$.

Proposition 5.3.3 (Antichain [40]). *The order* $(\mathsf{DiGraphs}, \leq)$ *has exactly three maximal antichains of size one—*$\{\overrightarrow{P_1}, \overrightarrow{P_2}, \overrightarrow{P_3}\}$.

However, in contrast to $(\mathsf{DiGraphs}, \leq)$, $(\mathsf{Graphs}, \leq)$ has no maximal antichains of any finite size $t > 1$.

Restricted homomorphism universality

As a warm-up, we first recall the most common restricted graph classes which are already defined in Chapter 2. We denote by Path the class of all finite oriented paths, by $\mathsf{DiCycle}$ the class of directed graphs formed all $\overrightarrow{C}_k$, $k \geq 3$, and $\mathsf{DiCycles}$ the class of directed graphs formed by disjoint union of finitely many graphs in $\mathsf{DiCycle}$. Similarly, denote by Cycle the class of graphs formed by all C_k, $k \geq 3$, denote by Cycles the class of graphs formed by disjoint union of finitely many graphs in Cycle. Finally denote by $\mathbb{P}$ the class of all odd prime numbers.

It is a classical result that the order $(\mathsf{Graphs}, \leq)$ is a universal order. In fact graph homomorphisms are universal also in the categorical sense [71]. More recently it has been shown that the universality of the homomorphism order is a relatively persistent property and the homomorphism order remains universal even for very restricted classes of (directed) graphs. In [45] the order $(\mathsf{Path}, \leq)$ is shown to be universal. This result can be used to show the universality of the homomorphism order in many naturally defined classes of (directed) graphs, such as 3-colourable graphs, planar graphs, series-parallel graphs, etc. Dichotomy results on the classes of graphs specified by chromatic and achromatic number are provided in [66].

The arguments in the proofs of those results build on techniques used in the proof of the following theorem (first stated in [45], an alternative and more streamlined proof appeared in [42]).

Theorem 5.3.4 (Hubička, Nešetřil [42, 45]). *The order* $(\mathsf{Path}, \leq)$ *is universal.*

These techniques however can not be easily applied in the context of locally constrained homomorphisms. In particular, locally constrained homomorphisms do not allow any sort of 'flipping' operation necessary for non-trivial path homomorphisms.

We prove the universality of $(\mathsf{DiCycles}, \leq)$ with a new argument replacing 'flipping' by 'rolling'. As a pleasant surprise, this result is significantly easier to prove (compared to the ones appearing in [52, 53, 54, 65, 75]) and has similar applications as Theorem 5.3.4

Lemma 5.3.5. *The order* $(\mathsf{DiCycle}, \leq)$ *is future-finite-universal.*

Proof. By Corollary 5.1.3 the order $(P_{\mathrm{fin}}(\mathbb{P}), \supseteq)$ is future-finite-universal.

For the finite set $A \subset \mathbb{P}$ we put

$$p(A) = \prod_{p \in A} p.$$

It immediately follows that

$$E(A) = \overrightarrow{C}_{p(A)}$$

is an embedding from $(P_{\mathrm{fin}}(\mathbb{P}), \supseteq)$ to $(\mathsf{DiCycle}, \leq)$; there is a homomorphism $\overrightarrow{C}_k \to \overrightarrow{C}_l$ if and only if l divides k. $\qquad\square$

Theorem 5.3.6. *The order* $(\mathsf{DiCycles}, \leq)$ *is universal.*

Proof. By Lemma 5.3.5, the order $(\mathsf{DiCycle}, \leq)$ is isomorphic to $(P_{\mathrm{fin}}(\mathbb{P}), \supseteq)$ that is future-finite-universal. It is easy to see that the order $(\mathsf{DiCycles}, \leq)$ is actually isomorphic to the subset order $(P_{\mathrm{fin}}(P_{\mathrm{fin}}(\mathbb{P})), \leq_{\supseteq}^{\mathrm{dom}})$[1]. The latter is known to be universal from Theorem 5.1.5. $\qquad\square$

Example 5.3.1. Consider the embedding of order given in Example 5.1.1. We can further construct an embedding E from $(P_{\mathrm{fin}}(P_{\mathrm{fin}}(\mathbb{P})), \leq_{\supseteq}^{\mathrm{dom}})$ to $(\mathsf{DiCycles}, \leq)$ as follows:

$$E(3) = \overrightarrow{C}_3, E(5) = \overrightarrow{C}_5 \oplus \overrightarrow{C}_3, E(7) = \overrightarrow{C}_{105}, E(11) = \overrightarrow{C}_{105} \oplus \overrightarrow{C}_{55}.$$

Here $\oplus$ denote the disjoint union of two graphs.

---

[1] By $\leq_{\supseteq}^{\mathrm{dom}}$ we mean $\leq_{P}^{\mathrm{dom}}$ where $\leq_P$ is $\supseteq$.

5.4 Embedding and monomorphism orders

Now we are ready to start exploring the properties of individual orders that are of interest to us. As a warm-up we start with the very restricted orders induced by monomorphisms and embeddings.

First we show that these orders are past-finite for a simple reason: For two directed graphs G and H, there is an embedding (or monomorphism) from G to H if and only if the number of vertices of G is not greater than the number of vertices of H. We immediately get the following proposition.

Proposition 5.4.1 (Past-finiteness). *The orders* (DiGraphs, $\leq^E$) *and* (DiGraphs, $\leq^M$) *are past-finite.*

We can show past-finite-universality of (DiGraphs, $\leq^E$) and (DiGraphs, $\leq^M$) by Lemma 5.1.2. There are infinitely many mutually incomparable connected graphs (isomorphism types). For example, every element of $S \in P_{\mathrm{fin}}(\mathbb{N})$ can be represented by the disjoint union of cycles C_i, $i \in S$.

Proposition 5.4.2 (Universality). *The orders* (Cycles, $\leq^E$), (DiCycles, $\leq^E$), (Cycles, $\leq^M$) *and* (DiCycles, $\leq^M$) *are past-finite-universal.*

Finally we show that dualities in the context of monomorphisms and embeddings are very simple.

Proposition 5.4.3 (Dualities). *For any finite set of directed graphs $\mathcal{D}$, there is a finite set of directed graphs $\mathcal{F}_E$ such that $(\mathcal{F}_E, \mathcal{D})$ is a generalized finite E-duality pair and a finite set of directed graphs $\mathcal{F}_M$ such that $(\mathcal{F}_M, \mathcal{D})$ is a generalized finite M-duality pair.*

Proof. We apply Corollary 5.2.3 (flipped for past-finite orders). Observe that for an E-gap or an M-gap (G, H), graphs G and H differ by at most one vertex or edge. Thus for every G there is only finitely many H such that (G, H) is an E-gap or an M-gap. Both embedding and monomorphism orders have only one minimal element (the empty graph). □

We however show a direct proof which gives a better understanding of the structure of $\mathcal{D}$ and we will extend it later.

Alternative proof. We prove the statement only for embeddings. Monomorphisms follow in a completely analogous way.

Fix a set $\mathcal{D}$ and denote by n the maximal number of vertices of graph in $\mathcal{D}$.

We construct a set $\mathcal{F}'_E$ as a set of all graphs F such that $|V_F| \leq n + 1$ and $F \to^E D$ for no $D \in \mathcal{D}$. Construct $\mathcal{F}_E$ as the set of all mutually non-isomorphic minimal elements of $(\mathcal{F}'_E, \leq^E)$.

It is easy to see that $(\mathcal{F}_E, \mathcal{D})$ is a generalized finite E-duality pair. (Obviously $\mathcal{F}_E$ is finite.) Consider a graph G such that there is no $D \in \mathcal{D}$ satisfying $G \to^E D$. There are two cases.

1. If $|V_G| \le n + 1$, then $G \in \mathcal{F}'_E$ and thus there is $F \in \mathcal{F}_M$, $F \to^E G$.

2. If $|V_G| > n + 1$, we consider any subgraph of G on $n + 1$ vertices. $\qquad \square$

Now consider $\mathcal{D} = \{D\}$ where D has n vertices and let $(\mathcal{F}, \mathcal{D})$ be an E-duality pair as constructed above. By Proposition 5.2.4 only graph $F \in \mathcal{F}$ with $n + 1$ vertices can be the right E-realization of D. We show that except for the degenerate case there is at most one such graph F.

A graph G is *vertex transitive* if for any distinct vertices $v, w \in V_H$, there exist a graph automorphism f such that $f(v) = w$.

Proposition 5.4.4 (Realizations). *Graph G is a right E-realization if and only if it is vertex transitive.*

Proof. It is easy to observe that vertex transitive graphs are right E-realizations. We show the opposite implication.

Consider a graph G that is a right E-realization. This means that removing any vertex from g leads to the same graph. It follows that in-degrees and out-degrees of vertices are the same: if not, removing one vertex can not lead to the same graph as removing different vertices with different degrees. Fix vertices v and v' and denote by H and H' graphs created from G by removing v and v' respectively. By definition H and H' are isomorphic. Denote by d the in-degree of vertices in G and (the out-degree must be the same). It follows that both H and H' have d vertices with in-degree $d - 1$ and the removed vertex is connected to all of them. The isomorphism of H to H' must map vertices of in-degree $d - 1$ to vertices of in-degree $d - 1$. The same holds for out-degrees. Therefore the isomorphism can be extended to an automorphism of G mapping v to v'. The choice of v and v' was arbitrary and thus it follows that G is vertex transitive.

Consider graph H. Because it has a (right) E-realization, we know that there is d such that H has d vertices of in-degree $d - 1$ and the rest of the vertices are of in-degree d and the same holds for out-degrees. This also specifies the vertex extending H to its realization. The only degenerate case is when $d = 0$ and thus there is no vertex of degree $d - 1$. In the case of single vertex graph the realizations are all graphs on 2 vertices. $\qquad \square$

5.5   Full homomorphism orders

As a first non-trivial case we consider full homomorphisms (mappings that are both edge and non-edge preserving). Knowledge about the full homomorphism

order is somewhat developed: dualities in the full homomorphism order have been studied by Ball, Nešetřil and Pultr in [3, 4]. The problem of the existence of a full homomorphism from G to H is also known as *full H-colouring* problem and in this language it has been independently studied by Feder and Hell in [19] and, more recently, by Hell and Hernández-Cruz [36]. In both cases the motivation was the correspondence of full homomorphisms to a certain matrix partition problem, see [19].

We closely relate the full homomorphism order to the embedding order. First we show that cores in full homomorphisms correspond to point-determining graphs which were studied in the 1970s by Sumner [78] (c.f. Feder and Hell [19]) and consequently we show that the cores are ordered by embeddings.

In directed graph G, the *out-neighbourhood* of a vertex $v \in G_V$, denoted by $N_G^{\rightarrow}(v)$, is the set of all vertices v' of G such that there is an edge from v to v' in G. Similarly *in-neighbourhood* of v, denoted by $N_G^{\leftarrow}(v)$, is the set of all vertices v' of G such that there is an edge from v' to v in G. We say that two vertices v and v' have the same neighbourhoods if both the in- and out-neighbourhoods match. Recall that *point-determining graphs* (also known as *mating-type graphs*, *mating graphs*, *M-graphs* or *thin graphs*) are graphs in which no two vertices have the same neighbourhoods. If we start with any graph G, and identify vertices with the same neighbourhoods, we obtain a point-determining graph we denote by G_{pd}.

It is easy to observe G_{pd} is always an induced subgraph of G and moreover that for every directed graph G, $G_{\mathrm{pd}} \leq^F G \leq^F G_{\mathrm{pd}}$ and thus $G \sim^F G_{\mathrm{pd}}$. This motivates the following proposition:

Proposition 5.5.1 (Cores). *Graph G is an F-core if and only if it is point-determining.*

Proof. Recall that G is an F-core if it is minimal (in number of vertices) in its equivalence class of $\sim^F$. If G is a F-core, G_{pd} can not be smaller than G and thus $G = G_{\mathrm{pd}}$.

It remains to show that every point-determining graph is an F-core. Consider two point-determining graphs $G \sim^F H$ that are not isomorphic. There are full homomorphisms $f : G \rightarrow^F H$ and $g : H \rightarrow^F G$. Because injective full homomorphisms are embeddings, it follows that either f or g is not injective. Without loss of generality, assume that f is not injective. Consider $u, v \in V_G$, $u \neq v$, such that $f(u) = f(v)$. Because full homomorphisms preserve both edges and non-edges it is easy to see that $N_G^{\leftarrow}(u) = N_G^{\leftarrow}(v)$ and $N_G^{\rightarrow}(u) = N_G^{\rightarrow}(v)$, a contradiction with the assumption that G is point-determining. $\square$

Now we apply the observations of Section 5.4 to get the following proposition:

Proposition 5.5.2. *For F-cores G and H we have $G \leq^F H$ if and only if $G \leq^E H$.*

Proof. Every embedding is also a full homomorphism. In the opposite direction consider a full homomorphism $f : G_{\mathrm{pd}} \to H_{\mathrm{pd}}$. By the same argument as in the proof of Proposition 5.5.1 we obtain that f is injective. $\square$

Corollary 5.5.3 (Past-finiteness). *For every $G \leq^F H$ we also have $|V_{G_{\mathrm{pd}}}| \leq |V_{H_{\mathrm{pd}}}|$. Thus the order $(\mathsf{DiGraphs}, \leq^F)$ is past-finite.*

Corollary 5.5.4 (Universality). *The order $(\mathsf{DiCycles}, \leq^F)$ is past-finite-universal.*

Proof. Observe that oriented cycles are point-determining graphs and apply Proposition 5.4.2. $\square$

Sumner [78] has defined a *nucleus* of a point-determining graph G as a set of vertices $G^0 \subseteq G$ such that for every $v \in G^0$ the graph created from G by removing v is also point-determining. All subgraphs that can be induced on a nucleus of connected point determining graphs have been characterized. It was shown that every (non-trivial) unoriented connected point determining graph has nucleus of size at least 2. This is a non-trivial result with an alternative proof appearing in [19].

Because all these notions were considered on undirected graphs only, we now prove some of these results for directed graphs. We recently became aware that the same work was also done independently in [36]:

Proposition 5.5.5. *Every F-core G with at least 2 vertices contains a F-core H with $|V_G| - 1$ vertices as an induced subgraph.*

While the Proposition is proved as Theorem 1 of [36], we give a simple self-contained proof.

Proof. If there is a vertex v of G such that the graph H created from G by removing v is point-determining, by Proposition 5.5.3 there is no F-core strictly between G and H and we are done.

Assume to the contrary that for every vertex $v \in V_F$ there are two vertices $\{v_1, v_2\}$ such that neighbourhoods of v_1 and v_2 become the same after removing v. Denote by G' the graph induced on vertices of G by those pairs. Thus G' is a graph with $|V_G|$ edges.

Take any path $p_1, p_2, \ldots, p_n$ in G'. Denote by $v_1, \ldots, v_{n-1}$ the vertices such that neighbourhoods of p_i and p_{i+1} differ only by v_i. By construction of G' all vertices v_i are unique. It follows that p_1 and p_n differs by all vertices $v_1, \ldots, v_{n-1}$. Consequently G' is a tree and thus it has at most $|V_G| - 1$ edges. A contradiction. $\square$

Very similar ideas give the characterization of gaps in the full homomorphism order.

Theorem 5.5.6 (Gaps). *If G and H are F-cores and (G, H) is an F-gap, then G is created from H by removing one vertex. Consequently the gaps in full homomorphism orders coincide with the gaps in embedding order.*

Proof. Assume to the contrary that there are F-cores G and H such that (G, H) is an F-gap, but the size of G differs from H by more than one vertex.

By Proposition 5.5.2 we know that G is a subgraph of H. We can assume that G is induced on H by V_G. Denote by A the set of vertices $V_H \setminus V_G$. By our assumption no vertex of A can be removed from H so that the resulting graph stays point-determining.

For every vertex $v \in A$ denote by $\{v_1, v_2\}$ a pair of vertices such that their neighbourhoods differ only by v. Denote by H' the graph induced on V_H by those pairs.

Because G is point-determining, we know that every edge of H' contains at least one vertex of A. Denote by $N_{H'}[A]$ the closed neighbourhood of A in H'.

We say that vertex v *determines* pair $\{v_1, v_2\}$ if the neighbourhoods of v_1 and v_2 differ only by v. On the induced subgraph of H' on vertices of $N_{H'}[A]$ we can apply the same analysis as in Proposition 5.5.5. It follows that there is at least one vertex v of $N_{H'}[A]$ which does not determine any pair of vertices of $N_{H'}[A]$. We know that $v \notin A$ and thus there is an edge (v, v') in H', $v' \in A$.

Denote by H'' the graph created from H by removing v'. First observe that G is an induced subgraph of H''; it is induced on vertices $(V_G \setminus \{v\}) \cup \{v'\}$. (neighbourhoods of v and v' differ only by a vertex in A.)

Finally observe that H'' is point-determining. By the choice of v no pair of neighbourhoods vertices u, u' such that $u \in A$ became the same. Also no pair of neighbourhoods in $(V_G \setminus \{v\}) \cup \{v'\}$ is equivalent because G is point-determining. This is a contradiction with the assumption that (G, H) is an F-gap. $\qquad\square$

Now we are ready to give a very simple proof of the existence of generalized dualities.

Theorem 5.5.7. *For every finite set of directed graphs $\mathcal{D}$ there is a finite set of graphs $\mathcal{F}$ such that $(\mathcal{F}, \mathcal{D})$ is a generalized finite F-duality pair.*

Proof. We again apply Corollary 5.2.3 (flipped for past-finite orders). By Theorem 5.5.6, for every F-gap (G, H), graphs G and H differ by at most one vertex. Thus for every G there are only finitely many H such that (G, H) is an F-gap. The empty graph is the unique minimum of the full homomorphism order. $\qquad\square$

Essentially the same duality result (for undirected graphs) was proved by Ball, Nešetřil and Pultr [4] (stated in terms of generalized finite dualities) and by Feder and Hell [19] (stated as a bound on the obstruction to full H- colourability).

The second was extended to directed graphs by Hell and Hermánez-Cruz [36] independently of our proof ([4] speaks of relational structures so their approach covers the case of directed graphs too). A simple argument for the bound also appears in [89]. It is interesting to observe that all proofs are different. Both [4] and [89] use a Ramsey type argument. Our argument is based on embeddings of point-determining graphs and is analogous to [19], but we use a simpler argument in Proposition 5.5.5 than the one needed to characterize the nucleus of a point determining graph. In [19, 36] the results are however further strengthened. For $\mathcal{D}$ consisting of single connected graph, one can show that $\mathcal{F}$ contains at most two graphs of size $n + 1$ (in addition to smaller graphs).

Thus one might say that proving dualities for full homomorphisms is a popular task. Before becoming aware about prior research, the authors also enjoyed work on an independent proof. To give a further contribututtion we present it here. We believe it has its own merit — it does not need to use Ramsey type arguments, analyse gaps or characterize the nucleus. It does however lead to a weaker description of the set $\mathcal{F}$ within the duality pair.

Alternative proof. We adapt the alternative proof of Proposition 5.4.3.

Without loss of generality, we assume that $\mathcal{D}$ is a set of F-cores. Fix a set $\mathcal{D}$, denote by n the maximum size of a graph in $\mathcal{D}$ and construct a set $\mathcal{F}'$ as union of the set of all F-cores F such that $|V_F| \leq n + 1 + k \leq n + 1 + \binom{n+1}{2}$ and there is $F \to^F D$ for no $D \in \mathcal{D}$. Finally construct $\mathcal{F}$ as the set of all mutually non-isomorphic minimal elements of $(\mathcal{F}', \leq^F)$.

Obviously $\mathcal{F}$ is finite because it contains graphs of bounded size. To see that $(\mathcal{F}, \mathcal{D})$ is a generalized finite F-duality pair consider the less trivial case where for a given G there is no $D \in \mathcal{D}$ such that $G \to D$.

If $|V_G| \leq n$ we have $G \in \mathcal{F}'$ and thus there is $F \in \mathcal{F}$, $F \leq^F G$.

Consider $|V_G| > n$. Denote by A an arbitrary subset of V_G such that $|A| = n+1$. Denote by $G[A]$ the subgraph of G induced by A. We construct a point-determining subgraph of V_G containing A in the following way. Put

$$A_0 = A.$$

Enumerate by $p_1, p_2, \ldots, p_k$ all pairs (u, v) of vertices of $G[A]$ such that $u \neq v$ and $N_{G[A]}(u) = N_{G[A]}(v)$.

Now for $i = 1, \ldots, k$ put $p_i = (u, v)$ and

1. $A_i = A_{i-1}$ if $N_{G[A_{i-1}]}(u) \neq N_{G[A_{i-1}]}(v)$;

2. $A_i = A_{i-1} \cup \{v'\}$ otherwise. v' is a vertex of G which is in the neighbourhood of u and not in the neighbourhood of v (or vice versa). Such vertex exists because G is point-determining.

It is easy to see that $|A_k| \leq n+1+k \leq n+1+\binom{n+1}{2}$ and moreover $G[A_k]$ is point-determining and thus its isomorphic copy is contained in $\mathcal{F}'$ leading to existence of $F \in \mathcal{F}$ such that $F \leq^F G[A_k] \leq^F G$. $\square$

By the results above it may seem that the structure of the full homomorphism order is basically the same as the structure of the embedding order. We close the section by pointing out one surprising difference.

The main result of [19] and [36] shows that in homomorphism order for a given graph H there are at most two right F-realizations of H. There are examples of graphs for all three cases: no realizations, one realization or two realizations. As we have shown in Proposition 5.2.4, a similar situation does not hold in the embedding order, where (up to a degenerate case) there is always one realization.

5.6   Surjective homomorphism orders

Vertex and edge surjective homomorphisms are very natural kinds of special homomorphisms. While they are generally less studied and understood than homomorphisms, there is a lively area of research with respect to the computational complexity problem of surjective colouring, see a recent survey [8].

From our point of view, surjective homomorphisms are similarly easy as embeddings and monomorphisms. We consider three cases—vertex surjective homomorphisms, edge surjective homomorphisms, and surjective homomorphisms (that are both edge and vertex surjective).

Proposition 5.6.1 (Cores). *Every finite graph is an S-core and VS-core. ES-core of graph G is created from G by removing all isolated vertices.*

Proof. Consider graph G and its S-core or VS-core H. By definition, there are surjective homomorphisms $f : G \to^{VS} H$ and $g : H \to^{VS} G$. By surjectivity of f we have $|V_G| \geq |V_H|$. From the surjectivity of g we have $|V_G| \leq |V_H|$ and thus $|V_G| = |V_H|$. It follows that both f and g are monomorphisms. Consequently $|E_G| \geq |E_H| \geq |E_G|$ and thus f and g are also isomorphisms.

On graphs without isolated vertices edge surjective homomorphisms are surjective homomorphisms. From the fact that removing isolated vertices does not affect any edges we know that if G' is created from G by removing (some) isolated vertices, then $G \sim^S G'$. $\square$

It follows that $(\mathsf{DiGraphs}, \leq^S)$ is actually a suborder of $(\mathsf{DiGraphs}, \leq^S)$ that is induced on $(\mathsf{DiGraphs}, \leq^{ES})$ by the class of all ES-cores (that is graphs without isolated vertices). All the basic properties of locally surjective homomorphism order are almost immediate:

Proposition 5.6.2 (Future-finiteness). *The orders* $(\mathsf{DiGraphs}, \leq^S)$, $(\mathsf{DiGraphs}, \leq^{ES})$ *and* $(\mathsf{DiGraphs}, \leq^{VS})$ *are future-finite.*

Proof. To see that $(\mathsf{DiGraphs}, \leq^{VS})$ is future-finite, fix graph G and consider its up-set. By vertex surjectivity it is easily seen that the size of every graph H in the up-set of G is no greater than the number of vertices of graph G.

$(\mathsf{DiGraphs}, \leq^{ES})$ is a suborder of $(\mathsf{DiGraphs}, \leq^S)$ that is a non-induced suborder of $(\mathsf{DiGraphs}, \leq^{VS})$. □

Proposition 5.6.3 (Universality). *The orders* $(\mathsf{DiCycle}, \leq^S)$, $(\mathsf{DiCycle}, \leq^{ES})$ *and* $(\mathsf{DiCycle}, \leq^{VS})$ *are future-finite-universal.*

Proof. Observe that any homomorphism between two oriented cycles is also an surjective homomorphism. The universality follows from Lemma 5.3.5. □

Proposition 5.6.4 (Gaps). (G, H) *is a VS-gap if and only if* $|G| = |H|$ *and* G *is created from* H *by removing an edge, or* $|G| = |H| + 1$, $G \to^{VS} H$ *and there is no way to add an edge to* G *preserving this property.*
 (G, H) *is a S-gap if and only if* $|G| = |H| + 1$, $|E_G| = |E_H|$, $G \to^S H$.
 ES-gaps correspond to S-gaps for pairs of graphs with no isolated vertices.

Proof. It is easy to see that in all cases $|G|$ is at most $|H| + 1$. Otherwise a graph inbetween can be constructed by partly concatenating vertices of G as given by the surjective homomorphism $G \to^S H$. The extra condition given prevents the existence of a graph inbetween G and H in this case.

For a vertex surjective mapping, there are gaps (G, H) with $|G| = |H|$: the mapping must be a monomorphism. If the graphs G and H differ by precisely one edge, they represent a gap in the monomorphism order. In the edge surjective case however the mapping must be an embedding and there are no gaps in the embedding order such that $|G| = |H|$. □

Proposition 5.6.5 (Dualities). *For every finite set of directed graphs* $\mathcal{F}$ *there are finite sets of directed graphs* $\mathcal{D}_S$, $\mathcal{D}_{VS}$ *and* $\mathcal{D}_{ES}$, *such that* $(\mathcal{F}, \mathcal{D}_S)$ *is a generalized finite S-duality pair,* $(\mathcal{F}, \mathcal{D}_{VS})$ *is a generalized finite VS-duality pair, and* $(\mathcal{F}, \mathcal{D}_{ES})$ *is a generalized finite ES-duality pair.*

Proof. Again the existence of dualities follows from Corollary 5.2.3 and from the characterization of gaps (Proposition 5.6.4). Observe that a single vertex with a loop on it and the empty graph are the only maximal elements of the vertex surjective homomorphism order. In edge surjective homomorphisms there are three maximal elements; single vertex, vertex with a loop and the empty graph. All gaps are the pairs of graphs that differ by 1 in number of vertices. □

Note that loops on vertices have to be allowed to make finite dualities possible: for graphs without loops the set $\mathcal{D}$ would need to contain cliques of arbitrary sizes, because all cliques are maximal elements in surjective homomorphism order on (directed) graphs without loops. Thus there are no dualities in the case of loopless graphs.

5.7   Locally constrained homomorphism orders

In this section we consider three forms of locally constrained homomorphisms: locally bijective, locally injective and locally surjective mappings. All three are defined by a particular behavior on the neighbourhoods of vertices and thus share many of common properties. There are also a number of other non-trivial relations between the three kinds of homomorphisms. See [22] for a recent survey of this area. The orders of locally constrained homomorphisms have been studied in [25]. We review and extend these results.

To our knowledge, locally constrained homomorphisms have not been considered for directed graphs. Moreover for locally constrained homomorphisms the orders induced on connected graphs differ significantly from the orders induced on Graphs. For example, it is easy to observe that a locally surjective homomorphism between two connected graphs is always surjective. This is not true for non-connected graphs. In this section we will restrict our attention to properties of locally constrained homomorphism orders on connected undirected graphs. We however still consider loops. We denote by ConnGraph the class of all finite undirected connected graphs. The properties of these orders on Graphs can be derived easily from our results by considering mappings between individual connected components. Directed graphs would require more generalizations, because some of the underlying concepts we use are formulated for undirected graphs only.

The following results show the main correspondence between all three types of locally constrained homomorphisms.

Theorem 5.7.1 (Fiala, Maxová [23]). *If a graph G admits a locally injective homomorphism f to a finite and connected graph H as well as a locally surjective homomorphism g to H, then all locally constrained homomorphisms between G and H are locally bijective.*

Or in the language of homomorphism orders we get:

Theorem 5.7.2 (Fiala, Paulusma, Telle [25]). *The order $(\mathsf{ConnGraph}, \leq^{LB})$ is the intersection of orders $(\mathsf{ConnGraph}, \leq^{LS})$ and $(\mathsf{ConnGraph}, \leq^{LI})$.*

Orders of degree refinement matrices

Recall that in Section 3.3 we defined degree refinement matrices, which are closely connected to locally constrained homomorphisms.

We put $G \sim^M H$ if and only if $\mathrm{drm}(G) = \mathrm{drm}(H)$. The relation $\sim^M$ is an equivalence relation on the class of finite connected graphs, ConnGraph.

From Theorem 3.3.2 and Theorem 3.3.3 it follows that in a single equivalence class of $\sim^M$ all three locally constrained homomorphisms coincide. We thus first look for the common properties of these orders within a single equivalence class. Observe that 2-regular undirected graphs are undirected cycles and thus Cycle is the class of graphs with degree refinement matrix (2). Consequently by Proposition 5.3.6 there is a rich, future-finite-universal order within a single equivalence classes of $\sim^M$.

We define *acyclic graphs* are trees with optional loops attached its vertices. We can summarize the properties of the orders within a single equivalence class of $\sim^M$ as follows:

Theorem 5.7.3. *Let $\mathcal{K}$ be an equivalence class of $\sim^M$. Then*

1. *If $\mathcal{K}$ consists only of acyclic graphs, then $\mathcal{K}$ is trivial (i.e. consisting of only one acyclic graph up to isomorphism).*

2. *If $\mathcal{K}$ contains graphs with a cycle, then the orders $(\mathcal{K}, \leq^{LB})$, $(\mathcal{K}, \leq^{LS})$ and $(\mathcal{K}, \leq^{LI})$ are all equivalent. They have the following properties:*

   (a) *all three orders are future-finite,*

   (b) *all three orders are future-finite-universal,*

   (c) *for every pair of graphs $G, H \in \mathcal{K}$ there exists a graph $C \in \mathcal{K}$ such that $C \leq^{LB} G$ and $C \leq^{LB} H$, and*

   (d) *there are no minimal elements.*

Note that the minimal elements within equivalence classes of $\sim^M$ may be established when infinite graphs are considered. For every equivalence class $\mathcal{K}$ containing a graph with a cycle, there exists a graph $C_{\mathcal{K}}$ (known as the universal cover) such that for every $G \in \mathcal{K}$, $C_{\mathcal{K}} \leq^{LB} G$, see [55].

Proof. The first part follows from the fact that acyclic graphs are uniquely described by their degree refinement matrices. See [24] for a formal proof.

The equivalence of orders $(\mathcal{K}, \leq^{LB})$, $(\mathcal{K}, \leq^{LS})$ and $(\mathcal{K}, \leq^{LI})$ follows from Theorems 3.3.2 and Theorem 3.3.3.

2(a) follows from Proposition 5.6.2.

2(b) follows from the fact that the same construction as used in the proof of Proposition 5.4.2 can be applied to any equivalence class of $\sim^M$ that contains at least one graph G with a cycle. Construct for every k a graph G_k created from G by multiplying the length of the cycle k times and duplicating its neighbourhood accordingly so $\mathrm{drm}(G_k) = \mathrm{drm}(G)$.

A non-trivial method of constructing graphs C for 2(c) is shown in [55], and alternate proofs in terms of Bass-Serre theory appear in [6] and [62].

2(d) follows from the fact that for every graph G containing a cycle C, one can construct a graph $G' \leq^{LB} G$ that contains a cycle on $2|C|$ vertices by sufficiently expanding G. □

Locally surjective and locally bijective orders

For connected graphs G and H, it is easy to observe that both locally bijective and locally surjective homomorphisms are also surjective homomorphisms (but not vice versa; a surjective homomorphism may not be locally surjective). By Proposition 5.6.1 we have:

Proposition 5.7.4 (Cores). *Every finite connected graph is LS-core and LB-core.*

By Proposition 5.6.2 we have:

Proposition 5.7.5 (Future-finiteness). *The orders* $(\mathsf{ConnGraph}, \leq^{LB})$ *and* $(\mathsf{ConnGraph}, \leq^{LS})$ *are future-finite.*

We denote by $\mathsf{UnorPath}$ the class of all unoriented paths. By an analogous argument as in Lemma 5.3.5 give:

Proposition 5.7.6. *The order* $(\mathsf{Cycle}, \leq^{LB})$ $(= (\mathsf{Cycle}, \leq^{LS}) = (\mathsf{Cycle}, \leq^{LI}))$ *is future-finite-universal. The order* $(\mathsf{UnorPath}, \leq^{LS})$ *is future-finite-universal.*

Proof. It is easy to observe that for all three types of locally constrained homomorphisms there is a homomorphism from $C_k \to C_l$ if and only if l divides k. The same holds for unoriented paths with k and l edges ordered by the locally surjective homomorphisms. The rest of the statement follows the same way as Lemma 5.3.5. □

Here we make an exception in our focus on connected graphs and state explicitly the following universality result. It can be proved analogously as Theorem 5.3.6 by applying Proposition 5.7.6 and Theorem 5.1.5.

Corollary 5.7.7 (Universality). *The order* $(\mathsf{Cycles}, \leq^{LB})$ $(= (\mathsf{Cycles}, \leq^{LS}) = (\mathsf{Cycles}, \leq^{LI}))$ *is universal.*

We know of no reasonable characterization of gaps in the locally bijective order. We however can show that:

Lemma 5.7.8 (Gaps). *If G is graph with a cycle, then there are infinitely many graphs H, $\mathrm{drm}(G) = \mathrm{drm}(H)$ such that (H, G) is a LB-gap, and this pair (H, G) is also a LI-gap and a LS-gap.*

Proof. We first consider locally bijective homomorphisms.

As an example, consider C_3 (cycle on 3 vertices). Then for every prime p the pair (C_{3p}, C_p) is a gap. The general case is similar. In a locally bijective homomorphism $f : H \to^{LB} G$, the pre-images of all vertices of G have the same size (and in locally surjective homomorphisms they are all non-empty). It remains thus to construct for every p a graph on $p|H|$ vertices with a locally bijective homomorphism to H. This can be done by considering any cycle in G and expanding its length p times and sufficiently duplicating all vertices connected to it.

Now observe that those LB-gaps are also LS-gaps. Assume to the contrary we have a LB-gap (H, G) which is not a LS-gap. It means that there is graph G' such that $H \to^{LS} G'$ and $G' \to^{LS} G$, and at least one of those locally surjective homomorphisms is not locally bijective. It follows that even their composition is also not locally bijective, a contradiction with Theorem 3.3.3.

LI-gaps follow in a completely analogous way. $\qquad\square$

Lemma 5.7.9 (Gaps of locally surjective homomorphism order). *If G is an acyclic graph, then there are infinitely many graphs H such that (H, G) is a LS-gap.*

Proof. Let us first consider the simple case of path P_{n+1} with n edges. It is not difficult to see that every path P_{pn+1} where p is a prime number forms a gap: the only graphs in between P_n and P_{pn+1} and locally surjective homomorphism of two paths must map the initial vertex to the initial or terminal vertex and the terminal vertex to the initial or terminal vertex. Also it can not flip in the middle of the path.

The same argument can be used for acyclic graphs in general. Ignoring loops, those are trees and thus they have at least two leaf vertices of degree 1. Denote two leaves of the tree as initial and terminal vertex and connect p copies of the acyclic graph into a sequence to form a gap. $\qquad\square$

Proposition 5.7.10 (Dualities). *There are no generalized finite duality pairs in $(\mathsf{ConnGraph}, \leq^{LB})$ and in $(\mathsf{ConnGraph}, \leq^{LS})$.*

Proof. We apply Proposition 5.2.2.

In the locally bijective case the analysis is easy. If $\mathrm{drm}(G) \neq \mathrm{drm}(H)$, then there is no locally bijective homomorphism from G to H, and because there are infinitely many degree refinement matrices and thus there are infinitely many maximal elements in $(\mathsf{ConnGraph}, \leq^{LB})$.

In the case of locally surjective homomorphism order, the maximal element is a vertex with loop. Because there are infinitely many gaps below any graph with a cycle (Lemma 5.7.8) as well as below any acyclic graph (Lemma 5.7.9), there are no duality pairs in both orders.   □

Locally injective homomorphism orders

The surjectivity argument can not be applied to locally injective homomorphisms. We will show that in fact the locally injective homomorphism order has some properties that are in sharp contrast with the locally surjective and locally bijective homomorphism orders.

To characterize cores of the locally injective homomorphism order we apply the classical result of Theorem 3.3.1 (also a special case of Theorem 3.3.2): If G is a finite connected graph, then every locally injective homomorphism $f : G \to^{LI} G$ is an automorphism of G.

Corollary 5.7.11 (Cores). *Every finite connected graph is a LI-core.*

Proof. The composition of the locally injective homomorphisms $f : G \to^{LI} H$ and $f' : H \to^{LI} G$ is a locally injective homomorphism $f \circ f' : G \to^{LI} G$. By Theorem 3.3.1, $f \circ f'$ is surjective and thus f' is surjective and it follows that $|V_H| \geq |V_G|$. By the same argument on $f' \circ f$, we have f is surjective and thus $|V_G| \geq |V_H|$. Consequently $|V_G| = |V_H|$ and both f and f' are isomorphisms.   □

For the first (and last) time we are able to show a variant of Welzl's density theorem [87]. This shows that the locally injective homomorphism order on undirected connected graphs is not locally finite.

Theorem 5.7.12 (Density). *Let G and H be connected graphs such that $\mathrm{drm}(G) \neq \mathrm{drm}(H)$, $G \to^{LI} H$ and H has no vertices of degree 1. Then:*

(a) *There exists a connected graph F, such that $G <^{LI} F <^{LI} H$, $\mathrm{drm}(F) \neq \mathrm{drm}(G)$ and $\mathrm{drm}(F) \neq \mathrm{drm}(H)$.*

(b) *When G has no vertices of degree 1 and H has at least one cycle with a vertex of degree greater than 2, then F can be constructed to have no vertices of degree 1 and contain a cycle with a vertex of degree greater than 2.*
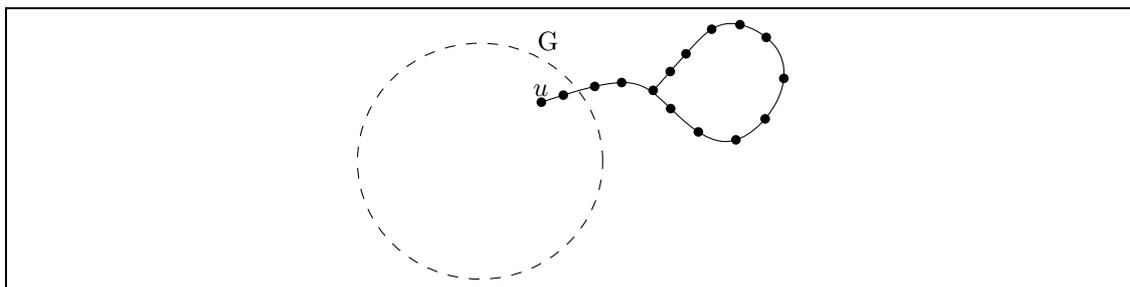
Figure 5.5: Graph G with vertex u extended by a (5,11)-lasso.

Proof. Fix a locally injective homomorphism $f : G \to H$. Because $\mathrm{drm}(G) \neq \mathrm{drm}(H)$ we know that f is not locally bijective. Denote by u a vertex such that $f(u)$ is strictly injective (that means injective but not bijective).

First construct F_1 from a graph G by extending a single vertex u' and a single edge (u, u'). It is easy to observe that F_1 satisfies the conditions given on F by (a). Obviously there are locally injective homomorphisms $G \to^{LI} F_1$ (G is a subgraph of F_1) and $F_1 \to^{LI} H$ (by extending the homomorphism f that is strictly injective at vertex u). $\mathrm{drm}(F_1)$ differs from $\mathrm{drm}(H)$ because H has no vertex of degree 1. Suppose $\mathrm{drm}(G) = \mathrm{drm}(H)$, from Theorem 3.3.2 there is a locally bijective homomorphism from G to F_1. Because every locally bijective homomorphism is surjective. we have $|G| \geq |F_1|$ that is a contradiction with $|G| < |F_1|$. Thus $\mathrm{drm}(G) \neq \mathrm{drm}(H)$.

Now we extend our construction to meet the requirements of (b). Denote by v the vertex of H that lies on a cycle and has degree greater than 2. Because H is connected, there is a path of length l from $f(u)$ to v. Finally denote by c the length of the cycle containing v.

We call the graph created by connecting a path of length a to a cycle of length b an (a, b)-*lasso*.

Now construct $F_{k,k'}$ as disjoint union of graph G and $(l + ck, ck')$-lasso with vertex u and the single vertex of degree 1 in the lasso identified.See Figure 5.5.

It is easy to observe that for every $k, k' \geq 1$ there is a locally injective homomorphism $G \to F_{k,k'}$ (G is an induced subgraph of $F_{k,k'}$). Similarly there is a locally injective homomorphism $F_{k,k'} \to H$ that can be constructed by extending f. The length of the cycle and the path has been chosen in a way so the first l vertices can be mapped into the path from $f(u)$ to v in H and the remaining vertices are mapped on the cycle containing v in H.

For a given vertex of degree 2 denote by a distance pair (a, b) the distances to nearest vertices of degree different than two, such that $a \leq b$. Two vertices of degree 2 with different distance pairs can not be in the same block of degree refinement matrix. Consequently $\mathrm{drm}(F_{k,k'}) \neq \mathrm{drm}(F_{m,m'})$ whenever $k \neq m$ or $k' \neq m'$. It follows that there exists a choice of k and k' such that $\mathrm{drm}(F_{k,k'}) \neq \mathrm{drm}(G)$ and

$\mathrm{drm}(F_{k,k'}) \neq \mathrm{drm}(H)$. We put $F = F_{k,k'}$ for such choice of k and k'. $\square$

The constraints given on graph H may seem artificial. However they lead to a full characterization of graphs H such that there is G, $\mathrm{drm}(G) \neq \mathrm{drm}(H)$ and (G, H) is a gap.

It remains to explore the cases where H has neither a vertex of degree 1, nor a cycle with vertex of degree greater than 2. We know that H is not a tree and thus it contains a cycle. Because H is connected it follows that all such graphs H are cycles. Graphs with a locally injective homomorphism to a cycle are either cycles or paths. When G is a path of length l, the graphs F can be chosen as a path of length $l+1$. It is easy to observe that $G \to^{LI} F \to^{LI} G$ and the degree matrices of G, H and F are mutually disjoint. When G is a cycle, we have $\mathrm{drm}(G) = \mathrm{drm}(H)$ and thus we can not hope for density result in general. As an example, consider G to be a cycle of length 4 and H to be a cycle of length 8.

Combining all results together we get:

Theorem 5.7.13 (Gaps). *Let H be a connected graph.*

(a) *There exists a connected graph G such that $\mathrm{drm}(G) = \mathrm{drm}(H)$ and (G, H) is a gap in $(\mathsf{ConnGraph}, \leq^{LI})$ if and only if H contains cycles.*

(b) *There exists a connected graph G, such that $\mathrm{drm}(G) \neq \mathrm{drm}(H)$ and (G, H) is a gap if and only if H has at least one vertex of degree 1.*

Proof. When H is acyclic, then its equivalence class is trivial. Consequently one implication of (a) is true.

We prove the other implication. As a simple consequence of Theorem 3.3.2, for any choice of G' and H with $\mathrm{drm}(G') = \mathrm{drm}(H)$, graph G such that $G' \to^{LI} G \to^{LI} H$ must have $\mathrm{drm}(G) = \mathrm{drm}(G') = \mathrm{drm}(H)$. We thus seek the gaps in the order $(\mathsf{ConnGraph}, \leq^{LI})$ restricted to a single equivalence class of $\sim^M$. By Theorem 5.7.3, this order is future-finite. Consequently there are only finitely many choices of G and every maximal choice of G (in $(\mathsf{ConnGraph}, \leq^{LI})$) produces a gap.

One implication of (b) is a consequence of Theorem 5.7.12 and the discussion concerning cycles given below.

In the other direction when H has a vertex v of degree 1, consider graph $G = H \setminus v$ and we show it has the desired properties. Assume, to the contrary that there is graph F such that $G <^{LI} F <^{LI} H$.

First assume that H is acyclic. In this case F is also acyclic and the locally injective homomorphisms are embeddings. Consequently F is a proper subgraph of H and G is a proper subgraph of F. A contradiction.

Assume that H contains a cycle. Denote by G' the largest induced subgraph of G containing no vertices of degree 1. Denote by g the locally injective

homomorphism from G to F and by f the locally injective homomorphism from F to H. Then the locally injective homomorphisms $g \circ f$ partialized to G' may only use vertices of G' and thus, by Theorem 3.3.1, it is an automorphism. It follows that any component of $G \setminus G'$ must be mapped to acyclic component of $H \setminus G'$. Such mappings are embeddings, too. Consequently $g \circ f$ is an embedding and thus g is also an embedding. Again, it follows that F is isomorphic either to G or H, which causes a contradiction. $\square$

Further developing these ideas, it is possible to show the universality of locally constrained order on connected graphs.

Theorem 5.7.14 (Universality). (ConnGraph, $\leq^{LI}$) *is a universal order.*

Fix an order $(P, \leq_P)$. For brevity we assume that $P = \{p_1, p_2, \ldots\}$ consists of prime numbers greater than or equal to 5. We construct an on-line embedding from $(P, \leq_P)$ to (ConnGraph, $\leq^{LI}$) by representing the elements by cycles similarly as in Theorem 5.3.6. Main complication lies in the fact that we need to connect cycles representing an element of $(P, \leq_P)$ to connected graph.

We build the representations from gadgets containing the cycles of desired lengths. *Sunlet on vertices $v_0, v_1, \ldots, v_n$ and $v'_1, v'_2, \ldots, v'_n$*[1] is a graph on vertices $\{v_0, v_1, \ldots, v_n, v'_1, v'_2, \ldots, v'_n\}$ containing a cycle on vertices $v_0, v_1, \ldots, v_n$ and edges (v_0, v'_0), (v_1, v'_1), $\ldots$, (v_n, v'_n). Vertices $v_0, v_1, \ldots, v_n$ are called *internal vertices* while vertices $v'_1, v'_2, \ldots, v'_n$ are *external*. The length of sunlet corresponds to the length of cycle on its internal vertices. For every $n \in P$ put:

$$l(n) = \{p \mid p \in P, p \leq n \text{ and } p \geq_P n\},$$

$$p(n) = \prod_{p \in l(n)} p.$$

Denote by $H(n)$ the graph constructed as a disjoint union of two sunlets:

1. a left sunlet on vertices $l_{n,0}, l_{n,1}, l_{n,2}, \ldots, l_{n,2^n p(n)-1}$ and $l'_{n,0}, l'_{n,1}, l'_{n,2}, \ldots, l'_{n,2^n p(n)-1}$,

2. a right sunlet on vertices $r_{n,0}, r_{n,1}, r_{n,2}, \ldots, r_{n,2^n 3-1}$ and $r'_{n,0}, r'_{n,1}, r'_{n,2}, \ldots, r'_{n,2^n 3-1}$,

with an additional edge $(l'_{n,p(n)}, r'_{n,0})$.

Graphs $E(p)$, $p \in P$ for the sample order $(P, \leq_P)$ shown in Figure 5.6 are schematically depicted in Figure 5.7.

---

[1]It is basically the same as n-sunlet S_n in Definiton 5.8.2, just the vertices are marked.
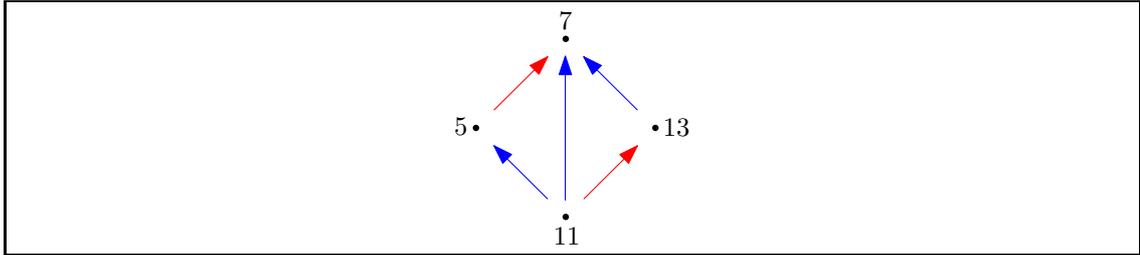
Figure 5.6: The order $(P, \leq_P)$.
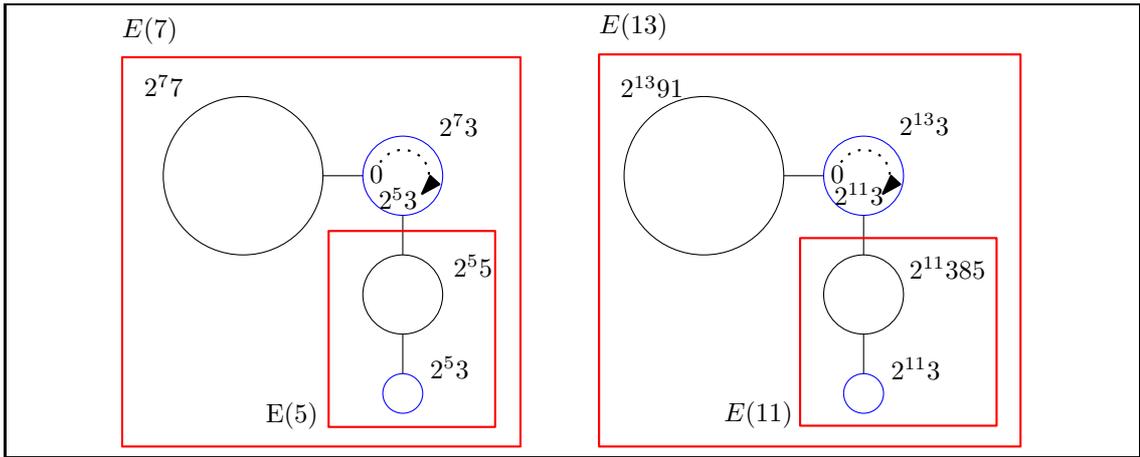
Figure 5.7: Representation of order $(P, \leq_P)$. Black circles denote left sunlets, blue circles right sunlets.

Lemma 5.7.15. *For $n > n' \in P$ there exists a locally injective homomorphism $H(n) \to^{LI} H(n')$ if and only if $n \leq_P n'$. If such locally injective homomorphism exists, then it maps vertex $l_{n,0}$ of $H(n)$ to vertex $l_{n,p(n)}$ of $H(n')$.*

Proof. Both $H(n)$ and $H(n')$ consist of left and right sunlets connected by an edge. The only vertices of degree 3 are internal vertices of sunlets. Consequently every locally injective homomorphism $f : H(n) \to^{LI} H(n')$ must map sunlets of $H(n)$ to sunlets of $H(n')$.

The length of the left sunlet is always divisible by at least one prime number greater or equal to 5 and is not divisible by 3. The length of the right sunlet is always multiple of powers of 2 and 3. Consequently every locally injective homomorphism $f : H(n) \to^{LI} H(n')$ must map the left sunlet of $H(n)$ to the left sunlet of $H(n')$ and the right sunlet of $H(n)$ to the right sunlet of $H(n')$. Such homomorphism exists if and only if $p(n)$ divides $p(n')$ and that is true, by definition of $p(n)$, if and only of $n \leq n'$ and $n \leq_P n$. $\square$

Proof (of Theorem 5.7.14). Denote by $E(n)$ the graph created by the disjoint

union of $H(n)$ and graphs $E(k)$, $k < n, p_k \leq_P p_n$, with additional edges between $l_{k,0}$ and $r_{n,2^{k-1}3}$.

We claim that E is an embedding from $(P, \leq_P)$ to $(\mathsf{ConnGraph}, \leq^{LI})$.

We show by induction on n' that for every $n \leq n'$

1. there is a locally injective homomorphism $f_{n,n'} : E(n) \to E(n')$ if and only if $p_n \leq_P p_{n'}$ and moreover $f_{n,n'}$ maps $l_{n,0}$ to $l_{n',p(n)}$;

2. there is a locally injective homomorphism $f_{n',n} : E(n') \to E(n)$ if and only if $p_{n'} \leq_P p_{n'}$.

We consider the following individual cases:

1. p_n is incomparable with p'_n in $(P, \leq_P)$. The left cycle of $H(n)$ have length that is multiple of p_n, but the left cycle of any $H(n'')$, $n \leq_P n'$ does not. Consequently there is no homomorphism $H(n) \to^{LI} H(n')$. The opposite direction follows in complete analogy.

2. $n \leq n', n \leq_P n'$. $E(n')$ contains copy of $E(n)$.

3. $n \leq n', n \geq_P n'$. We build homomorphism $f_{n',n}$ as follows:

    (a) $f_{n',n}(l_{n',i}) = l_{n,(p(n)+i) \mod 2p(n)}$.

    (b) $f_{n',n}(r_{n',i}) = r_{n,i \mod 2^n3}$.

    (c) For every $n'' < n$, $n'' \leq_P n'$, the vertices of copy of $E(n'')$ in $E(n')$ are mapped to copy of $E(n'')$ in $E(n)$.

    (d) For every $n'' > n$, $n'' \leq_P n'$, the vertices of copy of $E(n'')$ in $E(n')$ are mapped to $E(n)$ by $f_{n'',n}$. ☐

Proposition 5.7.16 (Dualities). *For a finite set of undirected connected graphs $\mathcal{D}$ there is a finite set of undirected connected graphs $\mathcal{F}$ such that $(\mathcal{F}, \mathcal{D})$ is a generalized finite LI-duality pair if and only if $\mathcal{D}$ consists of acyclic graphs.*

Proof. Every duality pair must be also a duality pair when restricted to a given equivalence class of degree refinement matrices. By the same argument as in Proposition 5.7.10 this is not possible for equivalence classes consisting of infinitely many elements.

The only equivalence classes consisting of finitely many elements are those corresponding to acyclic graphs.

Let $\mathcal{F}$ be a given set of acyclic connected undirected graphs. Denote by n the maximal number of vertices of graph in $\mathcal{F}$. $\mathcal{D}$ can be formed as the set of all acyclic graphs T on at most $n + 1$ elements such that there is no locally bijective homomorphism from $F \in \mathcal{F}$ to T.

Because the locally injective homomorphism order restricted to acyclic graphs is an embedding order, it remains to show that for every graph G containing a cycle there is graph $D \in \mathcal{D}$ such that there is an locally injective homomorphism from $D \to^{LI} G$. This follows from fact that $\mathcal{F}$ contains a path on $n+1$ vertices. $\qquad\square$

Remarks on the orders of DRMs

Fiala, Paulusma and Telle [25] have considered the orders implied by locally constrained homomorphisms on degree refinement matrices. Denote the class of all degree refinement matrices of graphs in ConnGraph by DRM. We define three relations $\to^{LB}$, $\to^{LS}$ and $\to^{LI}$ as follows: for two matrices $M, N \in$ DRM we have $M \to^{*} N$ if there exist graphs $G \in$ Graphs with $\mathrm{drm}(G) = M$ and $H \in$ Graphs with $\mathrm{drm}(G) = H$ such that $G \to^{*} H$ holds for the appropriate local constraint.

As stated earlier $(\mathsf{DRM}, \to^{LB})$ is a trivial order where no distinct elements are comparable. We further show the following two results that shed more light on the interplay among the individual variants of locally constrained homomorphisms:

Theorem 5.7.17 (Fiala, Paulusma, Telle [25]). *For every constraint $* = LB$, LI, LS the relation $(\mathsf{DRM}, \leq^{*})$ is an order. It arises as a factor order $(\mathsf{Graphs}, \leq^{*})$ when we unify graphs that have the same degree refinement matrices.*

Corollary 5.7.18 (Fiala, Paulusma, Telle [25]). *The intersection of orders $(\mathsf{DRM}, \leq^{LS})$ and $(\mathsf{DRM}, \leq^{LI})$ is a trivial order $(\mathsf{DRM}, \leq^{LB})$.*

This gives an unexpected insight: while all three locally constrained homomorphisms agree within the equivalence classes of degree refinement matrices, they fully disagree inbetween; the existence of a locally surjective homomorphism rules out the existence of locally injective homomorphism.

5.8   The intervals of the line graph order

Line graph homomorphisms are another variant of homomorphisms, which defines the mapping on edges. In this section, we consider the line graph order. Recently D. E. Roberson proposed a systematic study of homomorphism order on the class of line graphs [73]. Here, a *line graph* of an undirected graph G, denoted by $L(G)$, is graph $H = (V_H, E_H)$ such that $V_H = E_G$ and two vertices of $L(G)$ are adjacent if and only if their corresponding edges share a common endpoint in G. Because edges of G play the role of vertices of $L(G)$, we will refer vertices of line graphs as *nodes*.

The classical Vizing theorem gives an insight into the structure of the homomorphism order of line graphs in terms of *chromatic index* $\chi'(G)$ that is

the chromatic number of $L(G)$, i.e. the minimum number of colours needed to colour edges of a graph G such that edges with a common vertex receive different colours:

Theorem 5.8.1 (Vizing theorem [86]). *For any graph G of maximum degree d it holds that $\chi'(G) \leq d + 1$.*

Since the line graph of a graph with a vertex of degree d contains a d-clique, the Vizing theorem splits graphs into two classes. *Vizing class 1* contains the graphs whose chromatic index is the same as the maximal degree of a vertex, while *Vizing class 2* contains the remaining graphs.

The approach taken by Roberson [73] divides the class of line graphs into intervals. By $[K_n, K_{n+1})_{\mathcal{L}}$ we denote the class of all line graphs $L(G)$ such that $K_n \leq L(G) < K_{n+1}$. The line graphs in each interval have a particularly simple characterization:

Corollary 5.8.2. *The intervals $[K_d, K_{d+1})_{\mathcal{L}}$ consist of line graphs of graphs whose maximum degree is d.*

Proof. The existence of a $(d+1)$-edge colouring is equivalent with $L(G) \leq K_{d+1}$. Note that for the Vizing class 1 we have $L(G) \leq K_d \leq K_{d+1}$.

As G contains a vertex of degree d, we have $K_d \leq L(G)$, indeed $K_d \subseteq L(G)$. On the other hand, a clique on $d + 1 \geq 4$ vertices can be formed only from $d + 1$ edges sharing a common vertex, hence $K_{d+1} \not\leq L(G)$. The same argument used for $K_d \leq L(G)$ implies that G contains a vertex of degree d. □

Line graphs can be considered as almost perfect graphs (a *perfect graph* is a graph in which the chromatic number of every induced subgraph equals to the size of the largest clique of that subgraph). The homomorphism order of the class of perfect graphs is a trivial chain, since the core of every perfect graph is a clique. The almost-perfectness of the class of line graphs suggests that the homomorphism order of this class may be more constrained in its structure than the homomorphism order of graphs in general, and indeed many of the results about properties of the homomorphism order can not be easily restricted to the line graphs.

Roberson, in [73], showed that the homomorphism order of line graphs contains many gaps. This is the first important difference from the structure of the homomorphism order of graphs which was shown (up to one exception) to be dense by Welzl [87]. Roberson also asked whether every interval $[K_d, K_{d+1})_L$, $d \geq 3$ contains infinitely many incomparable elements. The answer is trivially negative for graphs with maximal degree 1 and 2. We give an affirmative answer to this problem. Indeed, we show the following (the proof is at the end of this section):

Theorem 5.8.3. *The homomorphism order of line graphs is universal on every interval* $[K_d, K_{d+1})_{\mathcal{L}}$ *for* $d \geq 3$.

This further develops the results on the universality of the homomorphism order of special classes of graphs (see e.g. [43, 45, 65, 66]), and on universal partially ordered structures in general (see e.g. [34, 41, 42, 47, 52, 53, 54]).

As a special case, universality of the interval $[K_3, K_4)_L$ follows from the construction given by Šámal [75]. This is not an obvious observation — one has to carefully check that for the graphs constructed in [75] the existence of a circulation coincides with the existence of a homomorphism of line graphs. Our proof use a new approach based on a new divisibility argument which we have introduced for a similar occasion in Section 5.1. This argument leads to a simpler construction without the need of complex gadgets (Blanuša snarks) used by Šámal [75].

Let $(P, \leq_P)$ be the order where P consists of all finite sets of integers and for $A, B \in P$ we put $A \leq_P B$ if and only if for every $a \in A$ there is $b \in B$ such that b divides a.

Theorem 5.8.4. *The order* $(P, \leq_P)$ *is a universal order.*

By the *divisibility order*, denoted by $(\mathbb{Z}, \leq_d)$, we mean an order where elements are natural numbers and n is smaller than m if n is divisible by m.

Lemma 5.8.5. *The divisibility order* $(\mathbb{Z}, \leq_d)$ *is future-finite-universal.*

Proof. Denote by $\mathbb{P}$ the set of all odd prime numbers[1]. Apply Lemma 5.1.2 for $A = \mathbb{P}$. Observe that $A \in P_{\text{fin}}(\mathbb{P})$ is a subset of $B \in P_{\text{fin}}(\mathbb{P})$ if and only if $\prod_{p \in A} p$ divides $\prod_{p \in B} p$. □

Then by Theorem 5.1.5 we prove Theorem 5.8.4.

Dragon graphs

We use a simple gadget called d-dragon which is also used in several constructions developed by Roberson [73]. In our constructions, the parameter d specifies the maximal degree of a vertex:

Definition 5.8.1. For $d \geq 3$, the d-*dragon*[2], denoted by D_d, is the graph created from K_{d+1} by replacing one of its edges by a path on 3 vertices.

The 3-dragon is depicted in Figure 5.8.
We proceed by a simple lemma about edge-colourings of dragons.

---

[1]Here we could use the set of all prime numbers, choose the odd ones is for consistency of the thesis

[2]The name is derived from the visual similarity of this graph to a kite, which in Czech is 'dragon'.
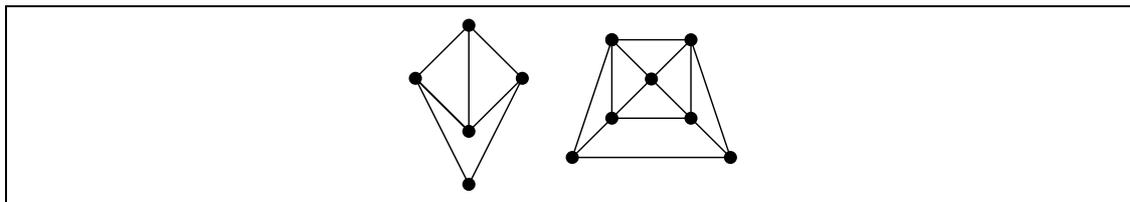
Figure 5.8: The 3-dragon D_3 and its line graph $L(D_3)$.

Lemma 5.8.6. *For all $d \geq 3$ it holds that D_d is Vizing class 2 graph, i.e. its chromatic index is $d + 1$.*

Proof. By Vizing theorem, $L(D_d)$ is $(d + 1)$-colourable. We prove that $L(D_d)$ is not d-colourable. The number of edges of D_d is $\frac{d(d+1)}{2} + 1 = \frac{d^2+d+2}{2}$, the number of vertices is $d+2$. We use the fact that every k-edge-colouring yields a decomposition of the graph into k disjoint matchings. We consider two cases:

1. If d is odd, then the maximum size of a matching in D_d is $\frac{d+1}{2}$, so the partition contains at least $d + 1$ matchings. Thus the chromatic number of $L(D)$ is $d + 1$.

2. If d is even, then the maximum size of a matching in D_d is $\frac{d+2}{2}$. However note that there is a vertex with degree 2, so there are at most 2 maximal matchings. The others matchings have the size at most $\frac{d}{2}$. It follows that d matching can cover at most $2(\frac{d+2}{2}) + (d - 2)\frac{d}{2} = \frac{4+d^2}{2}$ edges. For $d \geq 4$ we have $\frac{4+d^2}{2} \leq \frac{d^2+d+2}{2}$ and thus the partition contains at least $d+1$ matchings, i.e. colour classes. □

In our construction we will use the fact that d-dragons are cores as we show in the following lemma.

Lemma 5.8.7. *The graph $L(D_d)$, $d \geq 3$, is a core.*

Proof. For $d = 3$ observe that $L(D_3)$, depicted in Figure 5.8, is not 3-colourable, while each of its induced subgraphs is. Hence the statement holds for $d = 3$.

For $d \geq 4$ denote the vertices of D_d by $1, 2, \ldots, d, d+1, d+2$, where vertices $1, 2, \ldots, d+1$ have degree $d-1$ and the vertex $d+2$ has degree 2. The vertices of degree d correspond to d-cliques in $L(D_d)$. Each pair of those d-cliques share at most one node that corresponds to the edge connecting the original pair of vertices. Note that the shared node is unique for each such pair. Observe also that there are no other d-cliques in $L(D_d)$. This follows from the fact that the only way to create a 4-clique in a line graph is by a vertex of degree at least 4. See Figure 5.9.

Consider a homomorphism $f : L(D_d) \to L(D_d)$. Every homomorphism must map a d-clique to a d-clique, and thus it defines a vertex mapping $f' : \{1, 2, \ldots, d+1\} \to \{1, 2, \ldots, d+1\}$ in D_d.
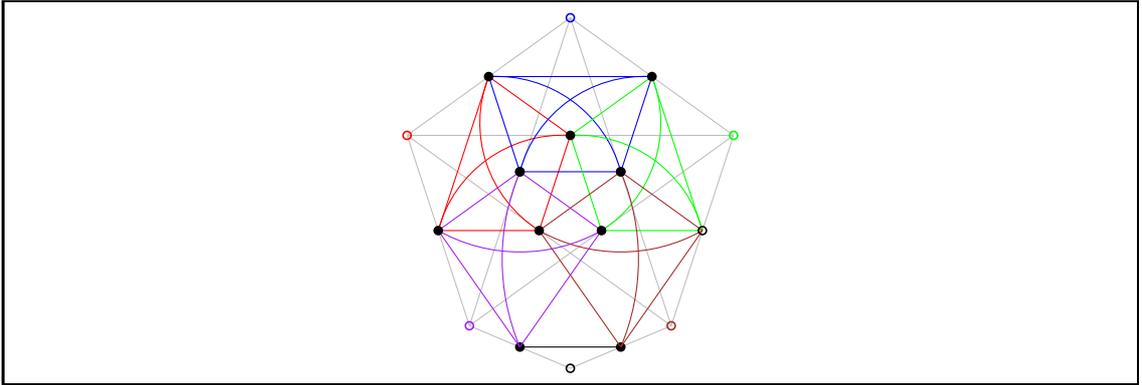
Figure 5.9: Line graph of 4-dragon with cliques corresponding to the neighbourhoods of two vertices distinguished.
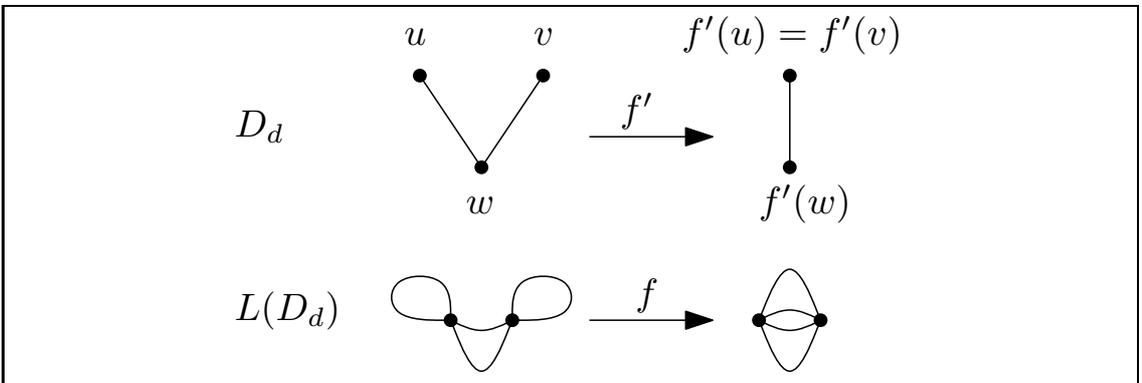
Figure 5.10: Mapping two d-cliques to the same target.

Assume that there are distinct $u, v \in \{1, \ldots, d+1\}$ such that $f'(u) = f'(v)$, see Figure 5.10. Take any $w \in \{2, \ldots, d\} \setminus \{u, v\} \neq \emptyset$. Because the node shared by the cliques corresponding to u and w is unique, it is different from the node shared by the cliques corresponding to v and w. Consequently, the cliques corresponding to $f'(u) = f'(v)$ and $f'(w)$ share at least two nodes. Since distinct d-cliques of $L(D_d)$ may share at most one node, it follows that $f'(w) = f'(u)$. Hence f' is either a bijection or a constant function. On the other hand, f' can not be a constant function by Lemma 5.8.6, as otherwise such mapping would yield an edge colouring of D_d by d colours.

Since f' is a bijection on vertices $\{1, 2, \ldots, d+1\}$ of D_d, the mapping f' must be a bijection on the edges between these vertices. The only way to get a homomorphism f of the whole D_d is to extend the mapping bijectively also on edges $\{1, d+2\}$ and $\{d+1, d+2\}$. By this argument we have proved that f is an isomorphism. $\square$

Indicator construction

We briefly describe the indicator technique, called often the 'arrow construction' in [40]. Informally, this construction means replacing every edge of a given graph G by a copy of graph I (an indicator) with two distinguished vertices identified with the endpoints of the edge. Figure 5.13 (left) shows result of indicator construction on graph in Figure 5.12 with indicator shown in Figure 5.11 (left). We give a precise definition of this standard notion:
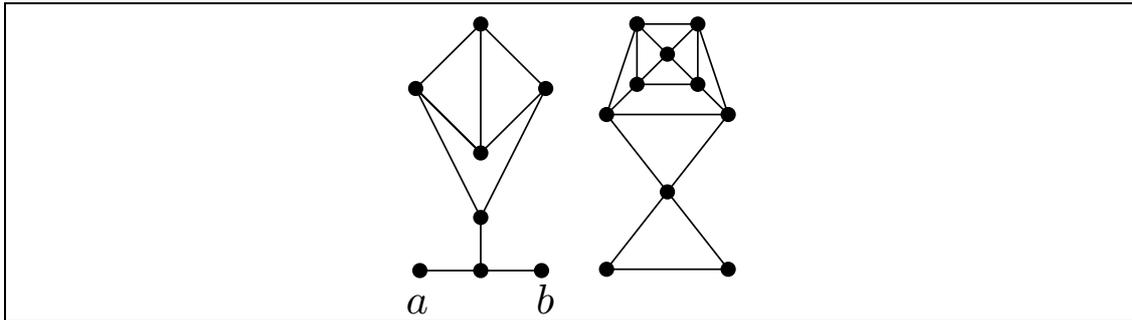
Figure 5.11: Indicator $I_3(a, b)$ and its line graph.

An *indicator* is any graph $I = (V_I, E_I)$ with two distinguished vertices a,b.

Given a graph $G = (V_G, E_G)$, we denote by $G*I(a, b)$[1] the graph $H = (V_H, E_H)$, where each edge is replaced by an extra copy of $I(a, b)$, where the vertices a and b are identified with the original vertices.

Formally, to obtain V_H we first take the Cartesian product $E_G \times V_I(a, b)$ and factorize this set by the equivalence relation $\sim$ consisting of the following pairs:

$$((x, y), a) \sim ((x, y'), a),$$

$$((x, y), b) \sim ((x', y), b),$$

$$((x, y), b) \sim ((y, z), a).$$

In other words, the vertices of H are equivalence classes of the equivalence $\sim$. For a pair $(e, x) \in E \times V_I$, the symbol $[e, x]$ denotes its equivalence class.

Vertices $[e, x]$ and $[e', x']$ are adjacent in H if $e = e'$ and $\{x, x'\} \in E_I$, we add no other edges.

---

[1] Here the *-operation is different from the one in relations.

Final construction

It is a standard technique to use the indicator construction to represent the class of graphs which is known to be universal (such as oriented paths) within another class of graphs (such as planar graphs) by using an appropriate rigid indicator (see [45]). It is then possible to show that the structure induced by the homomorphism order is preserved by the embedding via the indicator construction.

While our construction also uses indicators, the application is not so direct. It is generally impossible to have an indicator that would turn a graph into a line graph. We use the indicator to make graphs more rigid with respect of homomorphisms of their line graphs and model the divisibility order directly.

Our basic building blocks are the following:

Definition 5.8.2. The *n-sunlet graph*, denoted by S_n, is the graph on $2n$ vertices obtained by attaching n pendant edges to a cycle C_n, see Figure 5.12.
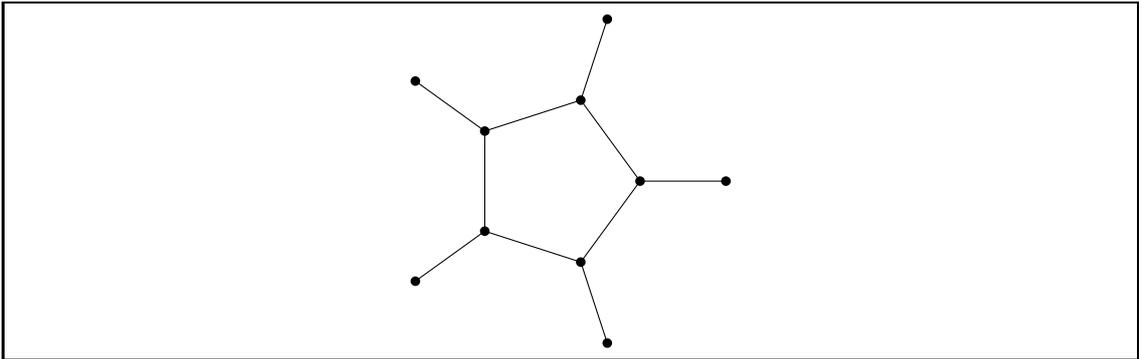
Figure 5.12: The 5-sunlet S_5.

Definition 5.8.3. For $d \geq 3$ the *indicator* $I_d(a,b)$ is the graph created from the disjoint union of the dragon D_d and a path on vertices a, c, b, where the vertex c is connected by an edge to the vertex of degree 2 in D_d, see Figure 5.11.

The desired class of graphs to show universality of the interval $[K_d, K_{d+1})_\mathcal{L}$, $d \geq 3$ consists of graphs $S_n * I_d(a,b)$ for $n \geq 3$. We abbreviate $S_n * I_d(a,b)$ by the symbol $G_{n,d}$. An example, the graph $G_{5,3}$, is shown in Figure 5.13. By squares are indicated vertices of degree three of the original sunlet graph. The three incident edges are in the line graph drawn as the triplets joined by the dashed triangles.

By Corollary 5.8.2 the graph $L(G_{n,m})$ is in the interval $[K_m, K_{m+1})_\mathcal{L}$ for every $n \geq 3$ and $m \geq 3$.

It remains to show the following property of the graphs $G_{n,m}$.
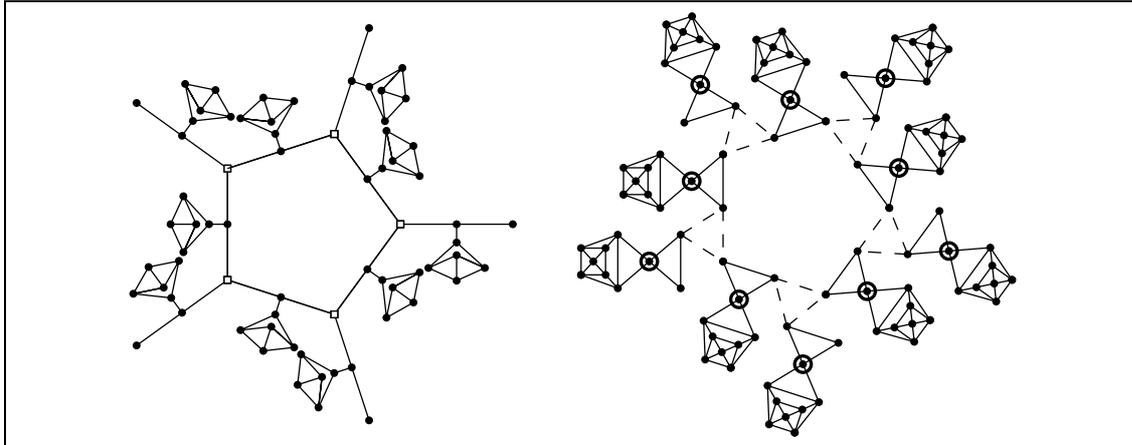
Figure 5.13: The graph $G_{5,3} = S_5 * I_3(a, b)$ and its line graph.

Proposition 5.8.8. *For every $d \geq 3$, $n \geq 3$, $n' \geq 3$ there is a homomorphism from $L(G_{n,d})$ to $L(G_{n',d})$ if and only if n is divisible n'.*

One direction of the proposition is trivial. If n is divisible by n' then the homomorphism is given by a homomorphism from S_n to $S_{n'}$ that cyclically wraps the bigger cycle around the smaller cycle. We call this homomorphism *cyclic*.

The other implication is a consequence of the following two lemmas.

The nodes of $L(G_{n,d})$ corresponding to the edges connecting the dragons with the vertices c are called *special*. In Figure 5.13 they are highlighted by circles.

Lemma 5.8.9. *For $d \geq 3$, $n \geq 3$, $n' \geq 3$ every homomorphism $f : L(G_{n,d}) \to L(G_{n',d})$ must map special vertices to special vertices.*

Proof. In $L(G_{n,d})$ the only 2-connected components of chromatic number 4 are the line graphs of dragons. It follows that the image of every line graph of a dragon must be in a line graph of a dragon. By Lemma 5.8.7, the line graphs of dragons are cores, thus for any special node u holds that its two neighbours v and w in the associated dragon D_d in $L(G_{n,d})$ must be mapped into some dragon D_d from $L(G_{n',d})$, but bijectively onto the two neighbours of the attached special node. (See Figure 5.11.)

Since u, v and w form a triangle, the only way to complete a triangle containing $f(v)$, $f(w)$ is to map u to the adjacent special node, as such triangle cannot be completed inside the dragon. □

A triangle in $L(G_{n,d})$ is called a *connecting triangle* if it originates from an original node of degree three in S_m. In Figure 5.13 the connecting triangles are denoted by dashed lines.

Lemma 5.8.10. *For every $d \geq 3$, $n \geq 3$ and $n' \geq 3$, every homomorphism $f : L(G_{n,d}) \to L(G_{n',d})$ must map connecting triangles to connecting triangles.*

Proof. Consider an arbitrary connecting triangle of $G_{n,d}$. It has the property that each of its node is adjacent to a special node. By Lemma 5.8.9, special nodes are preserved by the homomorphism f. The only triangles with this property in $G_{n',d}$ are precisely the connecting triangles. □

Proof of Proposition 5.8.8. By Lemma 5.8.10, the connecting triangles map to the connecting triangles. The other triangle in the cyclic configuration are those containing a special node. By Lemma 5.8.9 these triangles can not map to connecting triangles. Consequently, two connecting triangles joined by an edge can not map into one connecting triangle and thus the homomorphism f must be cyclic. □

Proof of Theorem 5.8.3. We apply Theorem 5.8.4 and show an embedding of $(P, \leq_P)$ to the homomorphism order of the interval $[K_d, K_{d+1})_{\mathcal{L}}$. For the chosen $d \geq 3$ we assign to every $A \in P$ a line graph $L(d, A)$ consisting of the disjoint union of graphs $L(G_{a,d}), a \in A$. Since any homomorphism must map connected components to connected components, we know by Proposition 5.8.8 that $L(d, A)$ allows a homomorphism to $L(d, B)$ if and only if $A \leq_P B$. □

Concluding remarks

Our results confirm that the homomorphism order of line graphs is rich. It is interesting that our embedding differs considerably from the one used in the proof of the universality of oriented paths by Hubička and Nešetřil [43].

Our construction is based on the retrospect of a homomorphism $f : L(G) \to L(H)$ to a binary relation $f' \subseteq V_G \times V_H$. We put $f'(v) = v'$ if all edges adjacent to v are mapped by f to the edges adjacent v' to H. This mapping is not always well defined. In particular:

(a) The image of an edge adjacent to a vertex of degree 1 of G is contained in the set of edges adjacent to two different vertices u, v connected by an edge in H. In this sense f' is not a function.

(b) Edges adjacent to a vertex v of degree 3 in G correspond to a triangle in $L(G)$. Because the line graph of a triangle is also a triangle, the image of these edges may thus map to a line graph of a triangle. In this case $f'(v)$ is not defined.

The basic idea behind the proof of Lemma 4.4.3 is the fact that f' (if it is a function) is close to a graph homomorphism $G \to H$ with two main differences:

(c) It may happen that $f'(u) = f'(v)$ for two adjacent vertices of G.

(d) For vertices of degree at least 3 the mapping f' is locally injective with the exception of (c).

Recall that a homomorphism $h : G \to H$ is *locally injective* if the restriction of the mapping h to the domain consisting of the vertex neighbourhood of v and range consisting of the vertex neighbourhood of v is injective. In one direction, every locally constrained homomorphism $h : G \to H$ yields a homomorphism $h' : L(G) \to L(H)$. Our observations above show that this direction can be reversed in special cases.

It is thus not a surprise that our universality proof is based on the ideas developed for the proof of universality of locally injective homomorphisms. We get closer to graph homomorphisms by means of the indicator construction. In the proof of Proposition 5.8.8 we consider a mapping $f'' : V_G \to V_H$ that retrospects a homomorphism $L(G * I_d(a, b)) \to L(H * I_d(a, b))$ in a similar way as f'. This mapping is a locally injective homomorphism with the exception of (b) and vertices of degree 2, where the local injecitivity is not enforced. For this reason we use sunlets instead of cycles and our embedding of the divisibility order is based on the fact that there is a locally injective homomorphism from between sunlet graphs S_n and S_m if and only if m divides n.

With this insight it is not difficult to see that our construction can be altered to form 3-regular graphs. This can be done by adding a cycle connecting pendant vertices to every sunlet. For degrees $d > 3$, the edges connecting both inner and outer can be turned into a multi-edges of a given degree, but also the indicator needs to be modified to become d-regular except for the two vertices of degree 1. By replacing the edge connecting dragon with vertex c by a clique of the corresponding degree and by adding a separate copy of a dragon to all but one vertex (the one connected to the base).

Several properties of locally injective homomorphism order are given by degree refinement matrices of the graphs considered [25]. As a special case, for d-regular graphs G the degree refinement matrix is trivial consisting of only one value d.

Roberson [73] shows that the homomorphism order line graphs is dense above every line graph of K_n, $n \geq 2$. Our construction gives many extra pairs with infinitely many different graphs strictly in between. Further such pairs can be obtained by an application of Theorem 5.7.12 in Section 5.7.

Assume that G and H have no vertex of degree 2 and the degree of vertices are either bounded by $d - 1$, or bounded by d and moreover that G and H are in Vizing class 1. Then the graph F given by Theorem 5.7.12 can be easily extended to F', where an extra edge is added to every vertex of degree 2. It is also easy to see that it cannot have vertices of degree greater than d. From the proof of Theorem 5.7.12 it follows that F is in Vizing class 1.

Consequently, $L(F' * I_d(a, b))$ is strictly in between $L(G * I_d(a, b))$ and $L(H * I_d(a, b))$. Afterwards, a graph strictly in between $G * I_d(a, b)$ and $F' * I_d(a, b)$ can be again constructed by applying Theorem 5.7.12 on G and F. This construction provides new examples of dense pairs in the homomorphism order of line graphs.

There are more results about the locally constrained homomorphism order that seem to suggest a strategy to attack problems about the homomorphism order of line graphs. It appears likely that the characterization of gaps in the locally injective homomorphism order will give new gaps in the homomorphism order of line graphs. The proof of universality of the locally constrained order of connected graphs (Theorem 5.7.14) in Section 5.7 can be translated to yet another proof of the universality of the homomorphism order of line graphs, this time however the graphs used are connected but not d-regular — since locally constrained homomorphism order is not universal on d-regular connected graphs. This suggests the question of whether the homomorphism order of line graphs is universal on the class of finite connected d-regular graphs.

Finally, Leighton's construction of a common covering for graphs [55] may give an insight into a way of constructing a suitable product for line graphs.

5.9   The orders of Relations

Relations between graphs have been introduced in Chapter 4 and show an interesting correspondence to graph homomorphisms. In this section we consider properties of the order induced by the existence of relations.

Recall that for directed graphs G and H we say that a binary relation is a *relation from G to H*, or *PR-morphism*, if there is a relation $R \subseteq V_G \times V_H$ such that (1) for every $(u, v) \in E_G, (u, u') \in R, (v, v') \in R$ we also have $(u', v') \in E_H$, (2) for every $(u', v') \in E_H$ we have some $(u, v) \in E_G, (u, u') \in R, (v, v') \in R$ and (3) for every $v' \in V_H$ we have some $v \in V_G$ such that $(u, v) \in H$. We say that there is a *relation from G to H with full domain*, or *R-morphism* if there is relation R from G to H and for every $v \in V_G$ there is some v' such that $(v, v') \in R$.

From the decomposition lemma, the order induced by relations is an interesting mix of the surjective homomorphism order and the transposed full homomorphism order: $\leq^{PR}$ is the transitive closure of $\geq^F \cup \leq^S$.

Moreover, without the full domain condition, we are for the first time allowed to forget parts of graphs G while looking for a relation $G \to^{PR} H$[1]. For this reason we first show properties of order induced by relations with full domain ($\leq^R$).

We know from Section 4.2.4 that R-cores are the unique minimal elements of the equivalence classes of $\sim^R$. Now we need to investigate the cores of $(\mathsf{DiGraphs}, \leq^R)$.

---

[1]$\to^{PR}$ is the same as $\backsim_w$ in Chapter 4.

PR-cores

$\leq^{PR}$ is the transitive closure of $\leq^R$ and $\geq^E$. Consequently $\sim^{PR}$ is coarser than $\sim^R$ and we immediately obtain:

Observation 5.9.1. *Every PR-core is an R-core.*

Lemma 5.9.2. *Let graphs G and H such that $G \sim^{PR} H$. If H is a PR-core, then H is isomorphic to induced subgraph of G.*

Proof. Fix $R_1 : H \to G$ and $R_2 : G \to H$. Denote by G_1 the graphs induced on domain of R_2 by G. It is easy to see that $R = R_2 \circ R_1$ is a relation for $G_1 \to^{PR} G$ and thus also $G_1 \sim^{PR} G$.

R_2 is also a relation from G_1 to H with full domain. Put $R'_1 = R_1 \cup V_H \times V_{G_1}$ and H_1 the graph induced by H on domain of R'_1. It is easy to see that $H_1 \sim^{PR} G_1 \sim^{PR} G$ and thus $H_1 = H$ from minimality of H.

Consequently H is the R-core of G_1 and thus by Theorem 4.2.11 H is isomorphic to induced subgraph of G_1 that is induced subgraph of G. ☐

From Lemma 5.9.2 and the observation that all relations involved can contain an identity we obtain:

Theorem 5.9.3. *Graph G is a PR-core if and only if it is an R-cocore.*

Properties of $(\mathsf{DiGraphs}, \leq^{PR})$ and $(\mathsf{DiGraphs}, \leq^R)$

Proposition 5.9.4 (Future-finiteness). *The orders $(\mathsf{DiGraphs}, \leq^{PR})$ and $(\mathsf{DiGraphs}, \leq^R)$ are future-finite.*

Proof. We have shown that both R-cores and PR-cores are point-determining graphs. For given graph G it remains to show that there are only finitely many point-determining graphs G' such that there is relation $R : G \to G'$.

Fix graph G. Consider point-determining G' and relation $R : G \to G'$. For two distinct vertices $u, v \in V'_G$, consider their pre-images $R^{-1}(u)$ and $R^{-1}(v)$. Because G' is point-determining, we know that $R^{-1}(u) \neq R^{-1}(v)$. Pre-images are subsets of V_G and there are only $2^{|G|}$) such subsets. Consequently, the size of G' is bounded by $2^{|G|}$. ☐

The relations with non-full domains are quite diffrent from the common notion of graph homomorphism, so it is not completely trivial to find an easy antichain. We given a simple example. Let $\overline{G}$ denote the complement of graph G. In our directed graph settings with loops, the complement of a graph without loops is graph with loops everywhere.

Lemma 5.9.5. *For any $n, m \geq 4$, $\overline{C_n}$ has a relation to $\overline{C_m}$ if and only if $m = n$.*

Proof. One direction is trivial; there is always relation from $\overline{C_n}$ to $\overline{C_n}$.

Assume that $m \neq n$ and the existence of relation $R : \overline{C_n} \to \overline{C_m}$. Every vertex v in a complement of cycle is not connected precisely to two additional vertices (u, u'). Every vertex v has unique pair (u, u'). By duplicating a vertex one obtain vertices that are not connected to the same pair of vertices. By the unification of vertices (that was not created by duplication of the same vertex) one gets vertex connected to everything. It follows that these operations are "useless" in an attempt to change $\overline{C_n}$ to $\overline{C_m}$. Finally observe that removing a vertex will break the cycle and turn it into a path. □

Theorem 5.9.6 (Universality). *The orders $(\mathsf{DiCycle}, \leq^R)$ and $(\mathsf{Graphs}, \leq^{PR})$ are future-finite-universal.*

Proof. We know that $\leq^S$ is a sub-relation of $\leq^R$ which is in turn a sub-relation of $\leq$. On the class of oriented cycles we have $\leq^S = \leq$. By Proposition 5.6.2 we know that $(\mathsf{DiCycle}, \leq^R)$ is future-finite-universal.

For every $S \in P_{\text{fin}}(\mathbb{N})$ we denote by $E(S)$ the graph created as the complement of the disjoint union of $E(S) = C_{i+4}, i \in S$. By the same argument as in the proof of Lemma 5.9.5 we get that $E(S) \leq^{PR} E(S')$ if and only if $S' \subseteq S$. by Corollary 5.1.3 we know that the class of complements of disjoint unions of cycles is future-finite-universal and thus $(\mathsf{Graphs}, \leq^{PR})$ is future-finite universal. □

Proposition 5.9.7 (Gaps). *Fix a directed graph H. There are only finitely many directed graphs G such that (G, H) is an R-gap or PR-gap.*

Proof. Assume that (G, H) is an R-gap, then G and H are R-cores, and there exists a relation R from G to H. We give an upper bound on the number of vertices of G based on the number of vertices of H.

We say that vertices u and v (of G) are equivalent if $R(u) = R(v)$. This equivalence has at most $2^{|H|}$ classes. Assume that there are two different vertices, u, and v in the same equivalence class. Denote by $G_{u,v}$ the graph created by unification of those two vertices. Since (G, H) is an R-gap, $G_{u,v}$ must be R-equivalent to H. And from the fact that there is a relation from H to $G_{u,v}$, we know that there are at most $2^{|H|}$ vertices in $G_{u,v}$ which have distinct neighbourhoods. Each neighbourhood of a vertex in $G_{u,v}$ can correspond to at most three different neighbourhoods in G (if a vertex w is connected in $G_{u,v}$ to the vertex created by unification of u and v, in G the vertex w may be connected to one of u and v or to both of them). It turns out that if G has at least $2^{|H|} + 1$ vertices (so at least one equivalence class is non-trivial) than there are at most $6^{|H|}$ different neighbourhoods. Because H is point determining we know it has no more than $6^{|H|}$ vertices.

Bounds in the case of partial relations follow in a completely analogous fashion. □

Proposition 5.9.8 (Dualities). *For every finite set of directed graphs $\mathcal{F}$ there is finite set of directed graphs $\mathcal{D}_R$, such that $(\mathcal{F}, \mathcal{D}_R)$ is a generalized finite R-duality pair, and a finite set of directed graphs $\mathcal{D}_{PR}$, such that $(\mathcal{F}, \mathcal{D}_{PR})$ is a generalized finite PR-duality pair.*

Proof. By Proposition 5.9.4 the orders are future-finite, by Proposition 5.9.8 there are only finitely many gaps below every graph. The maximal elements in both $(\mathsf{DiGraphs}, \leq^R)$ and $(\mathsf{DiGraphs}, \leq^{PR})$ are a single vertex and a vertex with loop. With these observations, we apply Corollary 5.2.3. □

5.10   Concluding Remarks

Our task of analysing variants of graph homomorphisms and proving their order-theoretic properties is open-ended. We have considered eleven variants of graph homomorphisms and answered questions related to cores, density, universality and dualities. We have found numerous references among previous research showing that many of these problems have been considered in different contexts and terminologies. Naturally our approach can be extended to new kinds of mappings (for example quantum homomorphisms studied in [73]), to wider or narrower classes of structures (such as graphs without loops, undirected graphs, general relational structures [4], partial orders [52] or minor-closed classes [65]) and to new properties, such as the existence of lattice operations.

We believe that the framework we have developed shows interesting connections between individual areas and sheds more light on some earlier results. For example, rather simple observations about the structure of full homomorphisms better explain earlier results on dualities. Our techniques seems to apply relatively smoothly to new special mappings and suggest a general line of attack to follow when analysing these properties. A systematic approach also led to a greatly simplified proof of the universality of the homomorphism order with applications to locally constrained homomorphisms and line graphs. We were able to show that the question of universality of oriented paths, solved in [43], is a lot more complex than universality of oriented cycles.

We have discussed many types of partial order — future-finite, past-finite, universal orders, with or without dualities. Among these the one closest to the homomorphism order is the order of locally injective homomorphisms on unoriented connected graphs. Only in this case have we been able to show universality and the existence of dualities, and partially characterize the gaps (the techniques have been used in the proofs are completely different to the ones used in

the analogous results of the homomorphisms orders). The only property it seems to lack is an interesting notion for cores. In fact we have not been able to identify any mapping with a notion of core that would be NP-complete to compute just like the graph core is. Our most complex example is given in Section 5.9). The full characterization of gaps in the locally constrained homomorphism orders seems to be an interesting open problem.

In some cases we have given only a partial characterizations of gaps, in particular in the case of relations. It seems that an approach similar to the one for full homomorphisms may lead to a better description. Also, we did not fully resolve questions about realizations that seem to be difficult already in the case of the full homomorphism order.

Appendix A: Computational complexity

The aim of this appendix is to give a brief exposition of the necessary background from computational complexity that we assume in the main body of the thesis. This introduction will be limited to the bare essentials, with no intention of comprehensiveness. For a thorough treatment of computational complexity theory one may refer to the monographs of Papadimitriou [70] and Arora and Barak [1].

Computational complexity is a branch of theoretical computer science. In contrast to the design and analysis of algorithms, computational complexity theory focuses on the hardness of computational problems. That is, it tries to answer the question why some problems cannot be computed *efficiently*. In a *computational problem*, an input is given and one is asked to return an output satisfying some property. For example, in the *factorization problem*, the input is a positive integer a, and the output is required to be all the prime factors of a. Another example is the *reachability problem*: given a graph G and two vertices $v_1, v_n \in V_G$, is there a path from v_1 to v_n? The input is the pair of vertices (v_1, v_n), and the output is yes or no.

We consider two kinds of computational problems. If the output of a computational problem is either yes or no, we call it *decision problem*. The reachability problem is a decision problem. If a computational problem has a single output (of a total function) for every input, but the output is more complex than that of a decision problem, that is, it isn't just yes or no, we call it a *function problem*. The factorization problem is a function problem.

A particular solution to a computational problem is called an *algorithm*. An algorithm is a detailed step-by-step procedure for solving a problem. For example, reachability can be solved by the so-called *search algorithm* [70]. This algorithm

works as follows: Throughout the algorithm we maintain a set of nodes, denoted by S. Initially, $S = \{v_1\}$. Each node can be either marked or unmarked. Node v_i is marked means that v_i has been in S at some point in the past or it is presently in S. Initially only v_1 is marked. At each iteration of the algorithm, we choose a node $v_i \in S$ and remove it from S. We then process one by one all edges (v_i, v_j) out of v_i. If node v_j is unmarked, then we mark it, and add it to S. This process continues until S becomes empty. At this point, we answer yes if node v_n is marked, and no if it is not marked. It is clear that this algorithm solves reachability problem. Moreover, it works efficiently. Here we can only give a rough explanation. We claim that we only spend about n^2 operations processing edges out of the chosen nodes, because there can be at most n^2 edges in a graph [1].

In order to introduce formally what is the time complexity of an algorithm, we need a model of computation. In 1936 Alan Turing invented the Turing machine, which is a mathematical model of a real machine. Since Turing machines are easy to analyse mathematically, and are believed to be as powerful as any other model of computation, besides, it is the most commonly used model in complexity theory and it can express an arbitrary algorithm.

Definition A.0.1 ([1]). A *(deterministic) Turing machine* is a quadruple $M = (K, \Sigma, \delta, s)$. Here K is a finite set of *states*; $s \in K$ is the *initial state*. Σ is a finite set of *symbols* (we say Σ is the *alphabet* of M). We assume that K and Σ are disjoint sets. Σ always contains the special symbols: the *blank* and the *first symbol*. Finally, δ is a transition function, which maps $K \times \Sigma$ to $(K \times \{h, \text{'yes'}, \text{'no'}\}) \times \Sigma \times \{\leftarrow, \rightarrow, - \}$. We assume that h (the *halting state*), 'yes' (the *accepting state*), 'no' (the *rejecting state*), and the *cursor directions* $\leftarrow$ for 'left', $\rightarrow$ for 'right', and $-$ for 'stay', are not in $K \cup \Sigma$.

When the Turing machine starts to work, the initial state is s. The string is initialized to a first symbol, followed by a input string x. The cursor is pointing to the first symbol. And then the machine takes a step according to δ, changing its states, printing a symbol, and moving the cursor, then continue take another step, and another. Only when the machine reaches one of the three halting states: h, 'yes', 'no', the machine does not continue. If this happens, we say that the machine has *halted*. At this time, if the states 'yes' has been reached, we say the machine *accepts* its input; if 'no' has been reached, we say the machine *rejects* its input. If h has been reached, the output is the string of M at the time of halting.

For decision problems, the machine always reaches either 'yes' or 'no'. We define the collection of all the inputs of Turing machine M which reach the states 'yes' as the *language* decided by M. For the function problem, the machine always

---

[1]It is usual to also consider space complexity. However in this thesis we only discuss time complexity.

reaches h, we say this kind of Turing machine computes functions. We use the terms of languages and decision problems interchangeably, as in [1].

TIME COMPLEXITY

Definition A.0.2. Let M be a (deterministic) Turing machine that halts on all inputs. The *running time* or *time complexity* of M is the function $f : \mathbb{N} \to \mathbb{N}$, where $f(n)$ is the maximum number of steps that M halts on any input of length n.

In the anaysis of complexity, usually we only need to know the order of $f(n)$, thus the complexity of an Turing machine is often expressed using *big $\mathcal{O}$ notation*: $f(n)$ is $\mathcal{O}(g(n))$ if there exists a n_0 and $c > 0$ such that $n \leq n_0$ implies $f(n) \leq cg(n)$. The following proposition explains why this estimation works.

Proposition A.0.1 ([70]). *Suppose that a (deterministic) Turing machine M decides a language L within time $f(n)$, where f is a proper function. Then there is a precise Turing machine M', which decides the same language in time $\mathcal{O}(f(n))$.*

COMPLEXITY CLASSES

Definition A.0.3. Let $t : \mathbb{N} \to \mathbb{R}^+$ be a function. Define the *time complexity class*, $\text{TIME}(t(n))$, to be the collection of all languages that are decidable by an $\mathcal{O}(t(n))$ time Turing machine.

Definition A.0.4. P is the class of languages that are decidable on a polynomial time deterministic Turing machine. In other words,

$$\text{P} = \bigcup_k \text{TIME}(n^k).$$

In a deterministic Turing machine, the set of rules prescribes at most one action to be performed for any given situation. A non-deterministic Turing machine, by contrast, may have a set of rules that prescribes more than one action for a given situation. A *nondeterministic Turing machine* is a quadruple $N = (K, \Sigma, \Delta, s)$, where K, Σ, s are as normal Turing machine, and the transition function Δ is a relation $\Delta \subset (K \times \Sigma) \times [(K \times \{h, \text{'yes'}, \text{'no'}\}) \times \Sigma \times \{\leftarrow, \rightarrow, - \}]$.

The set of languages decided by nondeterministic Turing machines within time f is denoted $\text{NTIME}(f(n))$. NP is the class of languages that are decidable on a polynomial time nondeterministic Turing machine, In other words,

$$\text{NP} = \bigcup_k \text{NTIME}(n^k).$$

We would like to point out that NP does not mean non-polynomial.

It is easily seen that P $\subseteq$ NP, since any language that is decidable by a deterministic Turing machine can also be decided by a nondeterministic Turing machine.

Polynomial time Turing reductions

Let Σ be an alphabet. We define $\Sigma^* := \{x_1, x_2, \ldots, x_k \mid k \geq 0 \text{ and each } x_i \in \Sigma\}$.

Definition A.0.5. A function $f : \Sigma^* \to \Sigma^*$ is a *polynomial time computation function* if some polynomial time Turing machine M exists that halts with just $f(w)$ on its tape, when started on any input w.

Definition A.0.6. Language $L \subseteq \Sigma^*$ is *polynomial time reducible*, to language $L' \subseteq \Sigma^*$, written $L \leq_{\text{P}}^{\text{Tur}} L'$, if a polynomial time computation function $f : \Sigma^* \to \Sigma^*$ exists, where for every x, $x \in L$ if and only if $f(x) \in L'$.

NP-complete problems

A central aim of the study of computational complexity is to sort out which problems can be solved in polynomial time and which cannot. In this context, we introduce the class of NP-complete problems, which are those the least likely to be in P.

Definition A.0.7. A problem H is NP-*hard* (nondeterministic polynomial time hard) if and only if every NP problem is polynomial time Turing reducible to H.
  NP-*complete problem* is a NP problem that is NP-hard.

When we need to prove that a problem is NP-complete, we need only to prove that (1) it is in NP, and (2) show that it is polynomial time reducible to a problem already known to be NP-complete. The hard part of that was finding the first example of an NP-complete problem: that was done by Steve Cook in Cook's Theorem [12].

References

[1] ARORA, S. & BARAK, B. (2009). *Computational complexity: a modern approach*. Cambridge University Press. 129, 130, 131

[2] BABSON, E. & KOZLOV, D. (2006). Complexes of graph homomorphisms. *Israel Journal of Mathematics*, **152**, 285–312. 3, 50

[3] BALL, R.N., NEŠETŘIL, J. & PULTR, A. (2007). Finite dualities, in particular in full homomorphisms. *KAM-DIMATIA series*, **835**. 7, 84, 91, 98

[4] BALL, R.N., NEŠETŘIL, J. & PULTR, A. (2010). Dualities in full homomorphisms. *European Journal of Combinatorics*, **31**, 106–119. 7, 84, 89, 91, 98, 100, 101, 127

[5] BANG-JENSEN, J. & HELL, P. (1990). The effect of two cycles on the complexity of colourings by directed graphs. *Discrete applied mathematics*, **26**, 1–23. 36

[6] BASS, H. & KULKARNI, R. (1990). Uniform tree lattices. *Journal of the American Mathematical Society*, **3**, 843–902. 106

[7] BIRKHOFF, G. (1995). *Lattice theory*, vol. 25. Amer Mathematical Society. 25

[8] BODIRSKY, M., KÁRA, J. & MARTIN, B. (2012). The complexity of surjective homomorphism problems—a survey. *Discrete Applied Mathematics*, **160**, 1680–1690. 80, 102

[9] BRIGHTWELL, G.R. & TROTTER, W.T. (1997). The order dimension of planar maps. *SIAM journal on Discrete Mathematics*, **10**, 515–528. 5

[10] BRITZ, T. & CAMERON, P. (2001). Partially ordered sets. 88

[11] BULATOV, A.A. (2005). H-coloring dichotomy revisited. *Theoretical Computer Science*, **349**, 31–39. 36

[12] COOK, S.A. (1971). The complexity of theorem-proving procedures. In *Proceedings of the third annual ACM symposium on Theory of computing*, STOC '71, 151–158, ACM, New York, NY, USA. 132

[13] DAVEY, B.A. & PRIESTLEY, H.A. (2002). *Introduction to Lattices and Order*. Cambridge University Press; 2 edition, second edition edn. 24

[14] DIESTEL, R. (2000). *Graph Theory*. Springer, Berlin, third edition edn. 14, 15

[15] DROSTE, M. (2005). Universal homogeneous causal sets. *Journal of Mathematical Physics*, **46(12)**. 88

[16] DUFFUS, D., KIERSTEAD, H.A. & TROTTER, W.T. (1991). Fibres and ordered set coloring. *Journal of Combinatorial Theory, Series A*, **58**, 158–164. 5

[17] ERDŐS, P.L., PÁLVÖLGYI, D., TARDIF, C. & TARDOS, G. (2012). On infinite-finite tree-duality pairs of relational structures. *CoRR*, **abs/1207.4402**. 94

[18] EVERETT, M.G. & BORGATTI, S. (1991). Role colouring a graph. *Mathematical Social Sciences*, **21**, 183–188. 4, 43, 45

[19] FEDER, T. & HELL, P. (2008). On realizations of point determining graphs, and obstructions to full homomorphisms. *Discrete Mathematics*, **308**, 1639–1652. 7, 41, 84, 89, 90, 91, 98, 99, 100, 101, 102

[20] FELLNER, W. (1982). On minimal graphs. *Theoretical Computer Science*, **17**, 103–110. 35

[21] FIALA, J. & KRATOCHVÍL, J. (2001). Complexity of partial covers of graphs. In *ISAAC*, 537–549. 44, 45, 46

[22] FIALA, J. & KRATOCHVÍL, J. (2008). Locally constrained graph homomorphisms — structure, complexity, and applications. *Computer Science Review*, **2**, 97–111. x, 4, 42, 43, 44, 104

[23] FIALA, J. & MAXOVÁ, J. (2006). Cantor-bernstein type theorem for locally constrained graph homomorphisms. *European Journal of Combinatorics*, **27**, 1111–1116. 104

[24] FIALA, J., PAULUSMA, D. & TELLE, J.A. (2005). Algorithms for comparability of matrices in partial orders imposed by graph homomorphisms. In D. Kratsch, ed., *Graph-Theoretic Concepts in Computer Science*, vol. 3787 of *Lecture Notes in Computer Science*, 115–126, Springer Berlin Heidelberg. 105

[25] FIALA, J., PAULUSMA, D. & TELLE, J.A. (2005). Matrix and graph orders derived from locally constrained graph homomorphisms. In *MFCS*, 340–351. 22, 42, 45, 104, 114, 123

[26] FIALA, J., PAULUSMA, D. & TELLE, J.A. (2008). Locally constrained graph homomorphisms and equitable partitions. *European Journal of Combinatorics*, **29**, 850–880. 22

[27] FIALA, J., HUBIČKA, J. & LONG, Y. (2014). Universality of intervals of line graph order. *arXiv preprint arXiv:1402.3736*. 7

[28] FIALA, J., HUBIČKA, J. & LONG, Y. (In preparation). Constrained homomorphism orders. 7, 55

[29] FONIOK, J., NEŠETŘIL, J. & TARDIF, C. (2008). Generalised dualities and maximal finite antichains in the homomorphism order of relational structures. *European Journal of Combinatorics*, **29**, 881–899. 94

[30] FRAÏSSÉ, R. (1986). *Théorie des relations*. North Holland. 85

[31] HAHN, G. & McGILLIVRAY, G. (2002). Graph homomorphisms: Computational aspects and infinite graphs. 36

[32] HAMMACK, R., IMRICH, W. & KLAVŽAR, S. (2011). *Handbook of Product Graphs*. Discrete Mathematics and Its Applications, CRC Press. 41

[33] HEDETNIEMI, S.T. (1966). *Homomorphisms of graphs and automate.*. 4, 40

[34] HEDRLÍN, Z. (1969). On universal partly ordered sets and classes. *Journal of Algebra*, **11(4)**, 503–509. 88, 116

[35] HEDRLÍN, Z. & PULTR, A. (1964). Relations (graphs) with given finitely generated semigroups. *Monatshefte für Mathematik*, **68**, 213–217. 2, 32

[36] HELL, P. & HERNÁNDEZ-CRUZ, C. (2013). Point determining digraphs, {0, 1}-matrix partitions, and dualities in full homomorphisms. *arXiv preprint arXiv:1308.0377*. 89, 98, 99, 101, 102

[37] HELL, P. & MILLER, D.J. (1976). Graph with given achromatic number. *Discrete Mathematics*, **16**, 195–207. 40

[38] HELL, P. & NEŠETŘIL, J. (1992). The core of a graph. *Discrete Mathematics*, **109**, 117–126. 35

[39] HELL, P. & NEŠETŘIL, J. (1990). On the complexity of h-coloring. *Journal of Combinatorial Theory, Series B*, **48**, 92–110. 36

[40] HELL, P. & NEŠETŘIL, J. (2004). *Graphs and homomorphisms*. Oxford University Press, Oxford, UK. 2, 4, 9, 32, 33, 36, 37, 38, 39, 40, 60, 71, 72, 79, 91, 94, 119

[41] HUBIČKA, J. & NEŠETŘIL, J. (2005). Finite presentation of homogeneous graphs, posets and ramsey classes. *Israel Journal of Mathematics*, **149**, 21–44. 116

[42] HUBIČKA, J. & NEŠETŘIL, J. (2011). *Model Theoretic Methods in Finite Combinatorics*, vol. 558, chap. Some examples of universal and generic partial orders, 293. American Mathematical Society. 87, 88, 94, 116

[43] HUBIČKA, J. & NEŠETŘIL, J. (2005). Universal partial order represented by means of trees and other simple graphs. *European Journal of Combinatorics*, **26**, 765–778. 6, 83, 92, 116, 122, 127

[44] HUBIČKA, J., JOST, J., LONG, Y., STADLER, P.F. & YANG, L. (2012). Relations between graphs. *Ars Mathematica Contemporanea*, **6**. 6

[45] HUBIČKA, J. & NEŠETŘIL, J. (2004). Finite paths are universal. *Order*, **21**, 181–200. 88, 92, 94, 116, 120

[46] IMRICH, W. & KLAVŽAR, S. (2000). *Product Graphs: Structure and Recognition*. Wiley, New York. 41, 49

[47] JOHNSTON, J.B. (1956). Universal infinite partially ordered sets. *Proceedings of the American Mathematical Society*, **7**, 507–514. 116

[48] KEDEM, G. & WATANABE, H. (1984). Graph-optimization techniques for ic layout and compaction. *Computer-Aided Design of Integrated Circuits and Systems, IEEE Transactions on*, **3**, 12–20. 4

[49] KELLY, D. & TROTTER JR, W.T. (1982). Dimension theory for ordered sets. In *Ordered Sets*, 171–211, Springer. 5

[50] KIERSTEAD, H.A. (1981). An effective version of Dilworth's theorem. *Transactions of the American Mathematical Society*, **268**, 63–77. 5

[51] KRISTIANSEN, P. & TELLE, J.A. (2000). Generalized H-coloring of graphs. In *ISAAC*, 456–466. 46

[52] KWUIDA, L. & LEHTONEN, E. (2011). On the homomorphism order of labeled posets. *Order*, **28**, 251–265. 95, 116, 127

[53] LEHTONEN, E. (2008). Labeled posets are universal. *European Journal of Combinatorics*, **29**, 493–506. 92, 95, 116

[54] LEHTONEN, E. & NEŠETŘIL, J. (2010). Minors of boolean functions with respect to clique functions and hypergraph homomorphisms. *European Journal of Combinatorics*, **31**, 1981–1995. 95, 116

[55] LEIGHTON, F.T. (1982). Finite common coverings of graphs. *Journal of Combinatorial Theory, Series B*, **33**, 231–238. 47, 105, 106, 124

[56] MATOUŠEK, J. (2003). *Using the Borsuk-Ulam theorem: lectures on topological methods in combinatorics and geometry*. Universitext (1979), Springer. 3, 50, 68

[57] NENTHEIN, S. & LERTWICHITSILP, P. (2012). Surjective multihomomorphisms between cyclic groups. *Thai Journal of Mathematics*, **4**, 35–42. 3

[58] NEŠETŘIL, J. & DE MENDEZ, P.O. (2008). Grad and classes with bounded expansion iii. restricted graph homomorphism dualities. *European Journal of Combinatorics*, **29**, 1012–1024. 93

[59] NEŠETŘIL, J. & PULTR, A. (1978). On classes of relations and graphs determined by subobjects and factorobjects. *Discrete Mathematics*, **22**, 287–300. 5

[60] NEŠETŘIL, J. & TARDIF, C. (2000). Duality theorems for finite structures (characterising gaps and good characterisations). *Journal of Combinatorial Theory, Series B*, **80**, 80–97. 5

[61] NEŠETŘIL, J. & TARDIF, C. (2003). On maximal finite antichains in the homomorphism order of directed graphs. *Discussiones Mathematicae Graph Theory*, **23**, 325–332. 93

[62] NEUMANN, W.D. (2011). On Leighton's graph covering theorem. *Groups, Geometry, and Dynamics*, **4**, 863–872. 106

[63] NEŠETŘIL, J. (1971). Homomorphisms of derivative graphs. *Discrete Mathematics*, **1**, 257–238. 4, 44

[64] NEŠETŘIL, J. (2000). On universality of set systems. *KAM-DIMATIA Series, Charles University*, **491**. 5, 88, 92, 93

[65] NEŠETŘIL, J. & NIGUSSIE, Y. (2006). Minimal universal and dense minor closed classes. *European Journal of Combinatorics*, **27**, 1159–1171. 95, 116, 127

[66] NEŠETŘIL, J. & NIGUSSIE, Y. (2007). Density of universal classes of series-parallel graphs. *Journal of Graph Theory*, **54**, 13–23. 94, 116

[67] NEŠETŘIL, J. (1999). Aspects of structural combinatorics (graph homomorphisms and their use). *Taiwanese Journal of Mathematics*, **3**, pp–381. 92, 93

[68] NEŠETŘIL, J. (1999). The homomorphism structure of classes of graphs. *Combinatorics, Probability and Computing*, **8**, 177–184. 93

[69] NOWAKOWSKI, R. & RIVAL, I. (1979). Fixed-edge theorem for graphs with loops. *Journal of Graph Theory*, **3**, 339–350. 35

[70] PAPADIMITRIOU, C.H. (2003). *Computational complexity*. John Wiley and Sons Ltd. 129, 131

[71] PULTR, A. & TRNKOVÁ, V. (1980). *Combinatorial, algebraic, and topological representations of groups, semigroups, and categories*. North-Holland mathematical library, North-Holland Pub. Co. 5, 84, 92, 94

[72] REIDEMEISTER, K. (1932). Einführung in die kombinatorische topologie. braunschweig: Friedr. vieweg. sohn a. *G. XII*, **209**. 4

[73] ROBERSON, D.E. (2013). *Variations on a Theme: Graph Homomorphisms*. Ph.D. thesis, University of Waterloo. 114, 115, 116, 123, 127

[74] SABIDUSSI, A. (1961). Graph derivatives. *Mathematische Zeitschrift*, **76**, 385–401. 2, 32

[75] ŠÁMAL, R. (2013). Cycle-continuous mappings — order structure. In J. Nešetřil & M. Pellegrini, eds., *The Seventh European Conference on Combinatorics, Graph Theory and Applications*, vol. 16 of *CRM Series*, 513–520, Scuola Normale Superiore. 95, 116

[76] SCHRIJVER, A. (2003). *Combinatorial optimization: polyhedra and efficiency*, vol. 24. Springer. 63

[77] STRANG, G. (2006). *Linear Algebra and Its Applications*. Thomson Brooks/Cole Cengage learning. 11

[78] SUMNER, D.P. (1973). Point determination in graphs. *Discrete Mathematics*, **5**, 179–187. 40, 41, 98, 99

[79] SYLVESTER, J.J. (1878). Chemistry and algebra. *Nature*, **17**, 284. 1

[80] TARDIF, C. (1999). Fractional multiples of graphs and the density of vertex-transitive graphs. *Journal of Algebraic Combinatorics*, **10**, 61–68. 93

[81] TRIPHOP, N., HARNCHOOWONG, A. & KEMPRASIT, Y. (2008). Multihomomorphisms between cyclic groups. *Set-valued Math. and Appl*, **1**, 9–18. 3

[82] TROTTER, W.T. (1996). Graphs and partially ordered sets: recent results and new directions. *Congressus Numerantium*, 253–278. 5

[83] TSANG, E. (1993). *Foundations of constraint satisfaction*, vol. 289. Academic press London. 36

[84] VIKAS, N. (2002). Computational complexity of compaction to reflexive cycles. *SIAM Journal on Computing*, **32**, 253–280. 4

[85] VIKAS, N. (2004). Compaction, retraction, and constraint satisfaction. *SIAM Journal on Computing*, **33**, 761–782. 4

[86] VIZING, V.G. (1964). On an estimate of the chromatic class of a p-graph. *Diskret. Analiz*, **3**, 25–30. 115

[87] WELZL, E. (1982). Color-families are dense. *Theoretical Computer Science*, **17**, 29–41. 5, 93, 108, 115

[88] WHYBURN, G.T. (1965). Continuity of multifunctions. *Proceedings of the National Academy of Sciences of the United States of America*, **54**, 1494. 3

[89] XIE, W. (2006). *Obstructions to Trigraph Homomorphisms*. Master's thesis, Simon Fraser University. 89, 91, 101

Index

Veröffentlichungen und Vorträge

Begutachtete Artikel

- *Constrained Homomorphism Orders*, joint with J. Fiala, J. Hubička, extended abstract, Proceedings of Bordeaux Graph Workshop 2012.

- *Relations between Graphs*, joint with J. Hubička, J. Jost, P. Stadler, L. Yang, Ars Mathematica Contemporanea, (2012), Volume 6, no. 2.

- *The Correspondence between Zero-divisor Graphs with 6 Vertices and Their Semigroups*, joint with Y. Huang, Journal of Algebra and its Applications.6 (2007), no. 2, 287–290. 13-04.

Eingereicht Artikel

- *Universality of intervals of line graph order*, joint with J. Fiala, J. Hubička, submitted to European Journal of Combinatorics. ArXiv:1402.3736

Vorträge

- *Relations between Graphs*, Bordeaux Graph Workshop, (Bordeaux, FR.), November 2012

- *Some interesting notions from Relations between Graphs*, Shanghai Conference on Algebraic Combinatorics, (Shanghai, CN.), August 2012

- *Relations between Graphs*, Workshop on Graph Theory and Combinatorics/Symposium for Young Combinatorists, National Sun Yat-sen University, (Kaohsiung, TW.), August 2012

- *The Full Homomorphism and Surjective Homomorphism Orders*, (Leipzig, DE.), International Max Planck Research School Workshop, June 2012

- *Graph relations and graph homomorphisms*, Workshop on graph homomorphisms and related topics, (Nova Louka, CZ.), December 2011

- *Relations between Graphs*, Workshop on Algebraic Combinatorics, Shanghai Jiao Tong University, (Shanghai, CN.), September 2011

- *Relations between Complex Networks*, Complex System Summer School, Sante Fe Insitute, (Santa Fe, US.), June 2011

- *Graph relations, a concept from biology*, Autumn seminar Bioinformatics 2010, (Vysoka Lipa, CZ.), October 2010

Selbstständigkeitserklärung

Hiermit erkläre ich, die vorliegende Dissertation selbständig und ohne unzulässige fremde Hilfe angefertigt zu haben. Ich habe keine anderen als die angeführten Quellen und Hilfsmittel benutzt und sämtliche Textstellen, die wörtlich oder sinngemäß aus veröffentlichten oder unveröffentlichten Schriften entnommen wurden, und alle Angaben, die auf mündlichen Auskünften beruhen, als solche kenntlich gemacht. Ebenfalls sind alle von anderen Personen bereitgestellten Materialien oder erbrachten Dienstleistungen als solche gekennzeichnet.

Leipzig, den 15. Oktober, 2014

. . . .
(Yangjing Long)

Short CV

Name:	Yangjing Long
Date of birth:	08.07.1985
09/2004 - 02/2008	Bachelor in Mathematics and Applied Mathematics Shanghai Jiao Tong University (China)
05/2008-10/2009	Graduate student at the Max Planck Institute for Mathematics in the Sciences (Germany)
since 11/2009	PhD student at the Max Planck Institute for Mathematics in the Sciences (Germany)

Daten zum Autor

Name:	Yangjing Long
Geburtsdatum:	08.07.1985
09/2004 - 02/2008	Mathematikstudium Universität Shanghai Jiao Tong
05/2008-10/2009	Studentin am Max-Planck-Institut für Matematik in den Naturwissenschaften Leipzig
seit 11/2009	Doktorandin am Max-Planck-Institut für Matematik in den Naturwissenschaften Leipzig