comput. complex. **26** (2017), 627 – 685

1016-3328/17/030627-59
published online December 19, 2016
DOI 10.1007/s00037-016-0147-6

computational complexity

GRAPH ISOMORPHISM, COLOR REFINEMENT, AND COMPACTNESS

V. Arvind, Johannes Köbler, Gaurav Rattan, and Oleg Verbitsky

Abstract. *Color refinement* is a classical technique used to show that two given graphs G and H are non-isomorphic; it is very efficient, although it does not succeed on all graphs. We call a graph G *amenable* to color refinement if the color refinement procedure succeeds in distinguishing G from any non-isomorphic graph H. Babai et al. (SIAM J Comput 9(3):628–635, 1980) have shown that random graphs are amenable with high probability. We determine the exact range of applicability of color refinement by showing that amenable graphs are recognizable in time $O((n+m)\log n)$, where n and m denote the number of vertices and the number of edges in the input graph.

We use our characterization of amenable graphs to analyze the approach to Graph Isomorphism based on the notion of *compact graphs*. A graph is called compact if the polytope of its fractional automorphisms is integral. Tinhofer (Discrete Appl Math 30(2–3):253–264, 1991) noted that isomorphism testing for compact graphs can be done quite efficiently by linear programming. However, the problem of characterizing compact graphs and recognizing them in polynomial time remains an open question. Our results in this direction are summarized below:

◦ We show that all amenable graphs are compact. In other words, the applicability range for Tinhofer's linear programming approach to isomorphism testing is at least as large as for the combinatorial approach based on color refinement.

◦ Exploring the relationship between color refinement and compactness further, we study related combinatorial and algebraic graph properties introduced by Tinhofer and Godsil. We show that the corresponding classes of graphs form a hierarchy, and we prove that recognizing

each of these graph classes is P-hard. In particular, this gives a first complexity lower bound for recognizing compact graphs.

Keywords. Graph Isomorphism, color refinement, linear programming relaxation, polytope of fractional automorphisms.

Subject classification. 68Q25 Analysis of algorithms and problem complexity.

1. Introduction

The well-known *color refinement* (also known as *naive vertex classification*) procedure for Graph Isomorphism works as follows: it begins with a uniform coloring of the vertices of two graphs G and H and refines the vertex coloring step by step. In a refinement step, if two vertices have identical colors but differently colored neighborhoods (with the multiplicities of colors counted), then these vertices get new different colors. The procedure terminates when no further refinement of the vertex color classes is possible. Upon termination, if the multisets of vertex colors in G and H are different, we can correctly conclude that they are not isomorphic. However, color refinement sometimes fails to distinguish non-isomorphic graphs. The simplest example is given by any two non-isomorphic regular graphs of the same degree with the same number of vertices. Nevertheless, color refinement turns out to be a useful tool not only in isomorphism testing but also in a number of other areas; see Grohe *et al.* (2014), Kersting *et al.* (2014), Shervashidze *et al.* (2011) and references there.

For which pairs of graphs G and H does the color refinement procedure succeed in solving Graph Isomorphism? This question has motivated the study of color refinement from different perspectives.

Immerman & Lander (1990), in their highly influential paper, established a close connection between color refinement and 2-variable first-order logic with counting quantifiers. They showed that color refinement distinguishes G and H if and only if these graphs are distinguishable by a sentence in this logic.

A well-known approach to tackling intractable optimization problems is to consider an appropriate linear programming relaxation. Consider a natural linear algebra formulation of Graph Isomorphism. Let G and H be two graphs on n vertices with adjacency matrices A and B, respectively. Then, G and H are isomorphic if and only if there is an $n \times n$ permutation matrix X such that $AX = XB$. A linear programming relaxation of this system of equations is to allow X to be a doubly stochastic matrix. If such an X exists, it is called a *fractional isomorphism* from G to H, and these graphs are said to be *fractionally isomorphic*. Building on Tinhofer (1986), it is shown by Ramana *et al.* (1994) (see also Godsil 1997) that two graphs are indistinguishable by color refinement if and only if they are fractionally isomorphic.

We say that color refinement *succeeds* on a graph G if it distinguishes G from any non-isomorphic H. A graph on which color refinement succeeds is called *amenable*. There are interesting classes of amenable graphs:

1. An obvious class of amenable graphs is formed by *unigraphs*. Unigraphs are graphs that are determined up to isomorphism by their degree sequences; see, e.g., Borri *et al.* (2011) and Tyshkevich (2000).

2. Trees are known to be amenable (Edmonds; see, e.g., Busacker & Saaty 1965; Valiente 2002).

3. It is easy to see that all graphs for which the color refinement procedure terminates with singleton color classes (i.e., with the discrete partition of the vertex set) are amenable. We call such graphs *discrete*. Babai *et al.* (1980) have shown that a random graph $G_{n,1/2}$ is discrete with high probability (moreover, the discrete partition of $G_{n,1/2}$ is reached within at most two refinement steps). Thus, almost all graphs are amenable, which makes Graph Isomorphism efficiently solvable in the average case (see also Babai & Kučera 1979).

What is the class of graphs on which color refinement succeeds? The logical and linear programming-based characterizations of color refinement do not provide an efficiently testable criterion answering this question.

We aim at determining the exact range of applicability of color refinement. We find an efficient characterization of the entire class of amenable graphs, which allows for a quasilinear-time test whether or not color refinement succeeds on a given graph. This result is shown in Section 5, after we unravel the structure of amenable graphs in Sections 3 and 4. We note that a weak *a priori* upper bound for the complexity of recognizing amenable graphs is $\mathsf{coNP}^{\mathsf{GI}[1]}$, where the superscript means the one-query access to an oracle solving the Graph Isomorphism problem. To the best of our knowledge, no better upper bound was known before.

By the result of Immerman & Lander (1990) mentioned above, our result implies that the class of graphs definable by first-order sentences with 2 variables and counting quantifiers is recognizable in polynomial time.

Note that, if G is an amenable graph and H is any other graph, then G and H are isomorphic if and only if they are fractionally isomorphic. The concept of a fractional isomorphism was used by Tinhofer (1991) as a basis for yet another linear programming approach to isomorphism testing. Tinhofer calls a graph G *compact* if the polytope of all its fractional automorphisms is integral; more precisely, if A is the adjacency matrix of G, then the polytope in $\mathbb{R}^{n^2}$ consisting of the doubly stochastic matrices X such that $AX = XA$ has only integral extreme points (i.e., all coordinates of these points are integers).

If a compact graph G is isomorphic to another graph H, then the polytope of fractional isomorphisms from G to H is also integral. If G is not isomorphic to H, then this polytope has *no* integral extreme point (and in fact no integral point at all). Thus, isomorphism testing for a compact graph G and an arbitrary graph H can be done in polynomial time by using linear programming to compute an extreme point of the polytope and testing if it is integral. Before testing isomorphism in this way, we need to know that G is compact. Unfortunately, no efficient characterization of compact graphs is currently known.

As our second main result, in Section 6 we show that all amenable graphs are compact. This implies that the applicability range for Tinhofer's (1991) approach to Graph Isomorphism is at least as large as for the combinatorial approach based on color refinement. More precisely, whenever the restriction of Graph Isomorphism to input graphs G and H such that G belongs to a class C is solvable by the latter approach, it is also solvable by the former approach. Consider, for example, the class C of unigraphs. The restricted Graph Isomorphism problem can obviously be solved by the color refinement algorithm in this case. As a particular consequence of our general result, it can also be solved by computing an extreme point of the polytope of fractional isomorphisms for the input graphs. In general, Tinhofer's approach is even more powerful than color refinement because it is known that the class of compact graphs contains many regular graphs that are not amenable as, for example, the cycles of length more than 5 (Tinhofer 1986).

In Section 7, we look at the relationship between the concepts of compactness and color refinement also from the other side. Let us call a graph G *refinable* if the color partition produced by color refinement coincides with the orbit partition of the automorphism group of G. It is interesting to note that the color refinement procedure gives an efficient algorithm to check if a given refinable graph has a non-trivial automorphism. It follows from the results by Tinhofer (1991) that all compact graphs are refinable. The inclusion $\mathsf{Amenable} \subset \mathsf{Compact}$, therefore, implies that all amenable graphs are refinable as well. The last result is independently obtained by Kiefer *et al.* (2015) by a different argument. In the particular case of trees, this fact was observed long ago by several authors; see a survey by Tinhofer & Klin (1999).

Taking a finer look at the inclusion $\mathsf{Compact} \subset \mathsf{Refinable}$, we discuss two other properties of compact graphs established by Tinhofer (1991) and Godsil (1997), respectively. Tinhofer (1991) shows that if a graph G is compact and H is an arbitrary graph, then a simple individualization-refinement algorithm correctly decides whether or not G and H are isomorphic, for every choice of vertices to be individualized. Let us denote the class of all graphs G for which this is true by $\mathsf{Tinhofer}$; we postpone precise definitions till Section 7. Godsil (1997) proves that, if G is compact, then every equitable partition of G is generated by a subgroup of the automorphism group of G. Denote the class of all graphs with this property by Godsil. We note that, along with the other graph classes under consideration, the classes Tinhofer and Godsil form a hierarchy under inclusion:

$$\mathsf{Discrete} \subset \mathsf{Amenable} \subset \mathsf{Compact} \subset \mathsf{Godsil} \subset \mathsf{Tinhofer} \subset \mathsf{Refinable}. \tag{1.1}$$

We show the following results on these graph classes:

- The hierarchy (1.1) is strict.
- Testing membership in each of these graph classes is P-hard.

Tinhofer (1991) had asked whether the classes Compact and Tinhofer coincide. The strictness of the hierarchy (1.1) answers this question. More precisely, by (1.1), the inequality Compact $\neq$ Tinhofer follows from either of the two inequalities Compact $\neq$ Godsil and Godsil $\neq$ Tinhofer. The class Godsil is separated from Compact by the Petersen graph, whose non-compactness is established by Evdokimov *et al.* (1999). The membership of the Petersen graph in Godsil follows from the computer algebra calculations performed by Ziv-Av (2013) or from a non-computer-assisted proof that we present in Appendix A. The class Tinhofer is separated from Godsil by the Johnson graphs $J(n, 2)$ for $n \geq 7$, that are not in Godsil according to Chan & Godsil (1997) and whose membership in Tinhofer is proved in Section 7.

In order to prove the P-hardness result, we give a suitable uniform AC^0 many-one reduction from the P-complete monotone Boolean circuit value problem (MCVP). More precisely, for a given MCVP instance (C, x), our reduction outputs a graph $G_{C,x}$ such that if $C(x) = 1$ then $G_{C,x}$ is discrete and if $C(x) = 0$ then $G_{C,x}$ is not refinable. In particular, the graph classes Discrete and Amenable are P-complete. We note that Grohe (1999) established, for each $k \geq 2$, the P-completeness of the equivalence problem for first-order k-variable logic with counting quantifiers; according to Immerman & Lander (1990), this implies the P-completeness of indistinguishability of two input graphs by color refinement. We adapt the gadget construction of Grohe (1999), that goes back to Cai *et al.* (1992), to show our P-hardness results.

Related work. A characterization of amenable graphs similar to that in the present paper has been suggested independently by Kiefer *et al.* (2015). Moreover, they obtain a generalization of this result to arbitrary relational structures (which includes, in particular, directed graphs, while our treatment is focused on vertex-colored undirected graphs).

Particular families of compact graphs were identified by Brualdi (1988), Godsil (1997), Schreck & Tinhofer (1988), and Wang & Li (2005); see also Chapter 9.10 in the monograph of Brualdi (2006). The concept of compactness is generalized to *weak compactness* by Evdokimov *et al.* (1999, 2000).

The linear programming approach of Tinhofer (1986) and Ramana *et al.* (1994) to isomorphism testing is extended by Atserias & Maneva (2013), Grohe & Otto (2015), and Malkin (2014), who showed that this extension corresponds to the k-dimensional Weisfeiler-Leman algorithm (which is just color refinement if $k = 1$; see Babai 1995 or Cai *et al.* 1992 for the definitions).

Notation. The vertex set of a graph G is denoted by $V(G)$. The vertices adjacent to a vertex $u \in V(G)$ form its neighborhood $N(u)$. A set of vertices $X \subseteq V(G)$ induces a subgraph of G, that is denoted by $G[X]$. For two disjoint sets X and Y, $G[X, Y]$ is the bipartite graph with vertex classes X and Y formed by all edges of G connecting a vertex in X with a vertex in Y. The vertex-disjoint union of graphs G and H will be denoted by $G + H$. Furthermore, we write mG for the disjoint union of m copies of G. Recall that the *complement* of a graph G has the same vertex set and exactly those edges that are absent in G. The *bipartite complement* of a bipartite graph G with vertex classes X and Y is the bipartite graph G' with the same vertex classes such that $\{x, y\}$ with $x \in X$ and $y \in Y$ is an edge in G' if and only if it is not an edge in G. We use the standard notation K_n for the complete graph on n vertices, $K_{s,t}$ for the complete bipartite graph whose vertex classes have s and t vertices, and C_n for the cycle on n vertices.

2. Basic definitions and facts

For technical convenience, we will consider graphs to be vertex-colored throughout the paper. A *vertex-colored graph* is an undirected simple graph G endowed with a vertex coloring $c : V(G) \to \{1, \dots, k\}$. Automorphisms of a vertex-colored graph and isomorphisms between vertex-colored graphs are required to preserve vertex colors. We get usual graphs when c is constant.

Given a graph G, the *color refinement* algorithm (to be abbreviated as *CR*) iteratively computes a sequence of colorings C^i of $V(G)$. The initial coloring C^0 is the vertex coloring of G, i.e., $C^0(u) = c(u)$. Then,

$$C^{i+1}(u) = \left(C^i(u), \{\!\{\, C^i(a) : a \in N(u) \}\!\}\right), \tag{2.1}$$

where $\{\!\{ \dots \}\!\}$ denotes a multiset.

The partition $\mathcal{P}^{i+1}$ of $V(G)$ into the color classes of C^{i+1} is a refinement of the partition $\mathcal{P}^i$ corresponding to C^i. It follows that, eventually, $\mathcal{P}^{s+1} = \mathcal{P}^s$ for some s; hence, $\mathcal{P}^i = \mathcal{P}^s$ for all $i \geq s$. The partition $\mathcal{P}^s$ is called the *stable partition* of G and denoted by $\mathcal{P}_G$.

Given a partition $\mathcal{P}$ of the vertex set of a graph G, we call its elements *cells*. We call $\mathcal{P}$ *equitable* if:

(i) Each cell $X \in \mathcal{P}$ is monochromatic, i.e., all vertices $u, v \in X$ have the same color $c(u) = c(v)$.

(ii) For each cell $X \in \mathcal{P}$ the graph $G[X]$ induced by X is *regular*, that is, all vertices in $G[X]$ have equal degrees.

(iii) For all pairs of cells $X, Y \in \mathcal{P}$ the bipartite graph $G[X, Y]$ induced by X and Y is *biregular*, that is, all vertices in X have equally many neighbors in Y and vice versa.

It is easy to see that the stable partition of G is equitable; our analysis in the next section will make use of this fact.[1]

A straightforward inductive argument shows that the colorings C^i are preserved under isomorphisms.

---

[1] The stable partition $\mathcal{P}_G$ is the *coarsest* equitable partition of G in the sense that all other equitable partitions of G are subpartitions of $\mathcal{P}_G$; see Cardon & Crochemore (1982, Lemma 1).

LEMMA 2.2. *If ϕ is an isomorphism from G to H, then $C^i(u) = C^i(\phi(u))$ for each vertex u of G.*

Lemma 2.2 readily implies that if graphs G and H are isomorphic, then

$$\{\!\{ C^i(u) : u \in V(G) \}\!\} = \{\!\{ C^i(v) : v \in V(H) \}\!\} \tag{2.3}$$

for all $i \geq 0$.

When used for isomorphism testing, the CR algorithm accepts two graphs G and H as isomorphic exactly when the above condition is met on input $G + H$. Note that this condition is actually finitary: If Equality (2.3) is false for some i, it must be false for some $i < 2n$, where n denotes the number of vertices in each of the graphs. This follows from the observation that the partition $\mathcal{P}^{2n-1}$ induced by the coloring C^{2n-1} must be the stable partition of the disjoint union of G and H. In fact, Equality (2.3) holds true for all i if it is true for $i = n$; see, e.g., Krebs & Verbitsky (2015). Thus, it is enough that CR verifies Equality (2.3) for $i = n$.

Note that computing the vertex colors literally according to (2.1) would lead to an exponential growth of the lengths of color names. This can be avoided by renaming the colors after each refinement step. Then, CR never needs more than n color names (appearance of more than n colors is an indication that the graphs are non-isomorphic).

DEFINITION 2.4. *We call a graph G* amenable *if for every graph H, procedure CR works correctly on the input pair G and H. That is, Equality (2.3) is false for $i = n$ whenever $H \not\cong G$.*

3. Local structure of amenable graphs

Consider the stable partition $\mathcal{P}_G$ of an amenable graph G. The following lemma gives a list of all possible regular and biregular graphs that can occur, respectively, as $G[X]$ and $G[X, Y]$ for cells X, Y of $\mathcal{P}_G$. Throughout the paper, we call a graph *empty* if it has no edge (i.e., its edge set is empty while the vertex set can be arbitrary).

LEMMA 3.1. *The stable partition $\mathcal{P}_G$ of an amenable graph G fulfills the following properties:*

(A) *For each cell $X \in \mathcal{P}_G$, $G[X]$ is an empty graph, a complete graph, a matching graph mK_2, the complement of a matching graph, or the 5-cycle;*

(B) *For all pairs of cells $X, Y \in \mathcal{P}_G$, $G[X, Y]$ is an empty graph, a complete bipartite graph, a disjoint union of stars $sK_{1,t}$ where X and Y are the set of s central vertices and the set of st leaves, or the bipartite complement of the last graph.*

The proof of Lemma 3.1 is based on the following facts.

LEMMA 3.2 (Johnson 1975). *A regular graph of degree d with n vertices is a unigraph if and only if $d \in \{0, 1, n-2, n-1\}$ or $d = 2$ and $n = 5$.*[2]

LEMMA 3.3 (Koren 1976). *A biregular graph where every of the m vertices in one part has degree c and every of the n vertices in the other part has degree d is determined up to isomorphism by the parameters c and d if and only if $c \in \{0, 1, n-1, n\}$ or $d \in \{0, 1, m-1, m\}$.*

If G contains a subgraph $G[X]$ or $G[X, Y]$ that is induced by some $X, Y \in \mathcal{P}_G$ but not listed in Lemma 3.1, then Lemmas 3.2 and 3.3 imply that this subgraph can be replaced by a non-isomorphic regular or biregular graph with the same parameters. Hence, in order to prove Lemma 3.1 it suffices to show that the resulting graph H is indistinguishable from G by color refinement. The graphs G and H in the following lemma have the same vertex set. Given a vertex u, we distinguish its neighborhoods $N_G(u)$ and $N_H(u)$ and its colors $C^i_G(u)$ and $C^i_H(u)$ in the two graphs.

LEMMA 3.4. *Let X and Y be cells of the stable partition of a graph G.*

[2] The last case, in which the graph is the 5-cycle, is missing from the statement of this result in Johnson (1975, Theorem 2.12). The proof in Johnson (1975) tacitly considers only graphs with at least 6 vertices.

(i) If H is obtained from G by replacing the edges of the subgraph $G[X]$ with the edges of an arbitrary regular graph of the same degree on the same vertex set X, then $C^i_G(u) = C^i_H(u)$ for each $u \in V(G)$ and each i.

(ii) If H is obtained from G by replacing the edges of the subgraph $G[X, Y]$ with the edges of an arbitrary biregular graph with the same vertex partition such that the vertex degrees remain unchanged, then $C^i_G(u) = C^i_H(u)$ for each $u \in V(G)$ and each i.

Proof of Lemma 3.4. We proceed by induction on i. In the base case of $i = 0$, the claim is trivially true. Assume that $C^i_G(a) = C^i_H(a)$ for all $a \in V(G)$. We consider an arbitrary vertex u and prove that

$$C^{i+1}_G(u) = C^{i+1}_H(u). \tag{3.5}$$

From now on we treat Parts (i) and (ii) separately.

(i) Suppose first that $u \notin X$. Since the transformation of G into H does not affect the edges incident to u, we have $N_G(u) = N_H(u)$. Looking at the definition (2.1), we immediately derive (3.5) from the induction assumption.

If $u \in X$, we only have the equality $N_G(u) \setminus X = N_H(u) \setminus X$, implying that

$$\{\!\!\{\, C^i_G(a) : a \in N_G(u) \setminus X \}\!\!\} = \{\!\!\{\, C^i_H(a) : a \in N_H(u) \setminus X \}\!\!\} . \tag{3.6}$$

The equality $N_G(u) \cap X = N_H(u) \cap X$ is not necessarily true. However, u has equally many neighbors from X in G and in H. Furthermore, for vertices a and a' in X we have $C^i_G(a) = C^i_G(a')$ because X is a cell of $\mathcal{P}_G$, and $C^i_H(a) = C^i_G(a) = C^i_G(a') = C^i_H(a')$ by the induction assumption. That is, all vertices in X have the same C^i-color both in G and in H. It follows that

$$\{\!\!\{\, C^i_G(a) : a \in N_G(u) \cap X \}\!\!\} \;=\; \{\!\!\{\, C^i_H(a) : a \in N_H(u) \cap X \}\!\!\} . \tag{3.7}$$

Combining (3.6) and (3.7), we conclude that (3.5) always holds.

(ii) If $u \notin X \cup Y$, we have $N_G(u) = N_H(u)$ and Equality (3.5) readily follows from the induction assumption.

Suppose that $u \in Y$. In this case, we still have (3.6) and, exactly as in Part (i), we also derive (3.7). Equality (3.5) follows. The case of $u \in X$ is symmetric. □

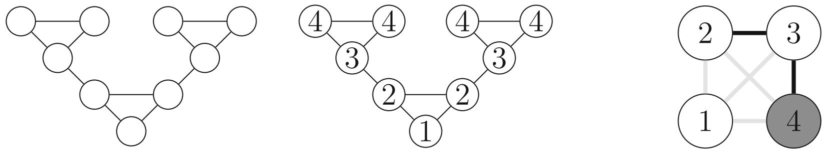
PROOF OF LEMMA 3.1. **(A)** If $G[X]$ is a graph not from the list, by Lemma 3.2, it is not a unigraph. Hence, we can modify G locally on X by replacing $G[X]$ with a non-isomorphic regular graph with the same parameters. Part (i) of Lemma 3.4 implies that the resulting graph H satisfies Equality (2.3) for each i, implying that CR does not distinguish between G and H. The graphs G and H are non-isomorphic because, by Part (i) of Lemma 3.4 and by Lemma 2.2, an isomorphism from G to H would induce an isomorphism from $G[X]$ to $H[X]$. This shows that G is not amenable.

(B) We derive this condition, similarly to Condition **A**, from Lemma 3.3 and Part (ii) of Lemma 3.4. □

4. Global structure of amenable graphs

Recall that $\mathcal{P}_G$ is the stable partition of the vertex set of a graph G and that we call elements of a partition *cells*. We define the auxiliary *cell graph* $C(G)$ of G to be the complete graph on the vertex set $\mathcal{P}_G$ with the following labeling of vertices and edges. A vertex X of $C(G)$ is designated *homogeneous* if the graph $G[X]$ is complete or empty and *heterogeneous* otherwise. An edge $\{X, Y\}$ of $C(G)$ is designated *isotropic* if the bipartite graph $G[X, Y]$ is either complete or empty and *anisotropic* otherwise; for an example see Figure 4.1. A path $X_1X_2 \dots X_l$ in $C(G)$ where every edge $\{X_i, X_{i+1}\}$ is anisotropic will be referred to as an *anisotropic path*. If also $\{X_l, X_1\}$ is an anisotropic edge, we speak of an *anisotropic cycle*. In the case that $|X_1| = |X_2| = \dots = |X_l|$, such a path (or cycle) is called *uniform*.

For graphs fulfilling Conditions **A** and **B** of Lemma 3.1, we refine the labeling of the vertices and edges of $C(G)$ as follows. A heterogeneous cell $X \in \mathcal{P}_G$ is called *matching*, *co-matching*, or *pentagonal* depending on the type of $G[X]$. Note that a matching or co-matching cell X always consists of at least 4 vertices. Further, an anisotropic edge $\{X, Y\}$ is called *constellation* if $G[X, Y]$ is a



A graph G. The stable coloring of G. The cell graph $C(G)$.

Figure 4.1: An example of the cell graph. The homogeneous vertices of $C(G)$ are *white*, the heterogeneous vertex is *black*. The isotropic edges of $C(G)$ are gray, the anisotropic edges are *black*

disjoint union of stars, and *co-constellation* otherwise (in the latter case, the bipartite complement of $G[X, Y]$ is a disjoint union of stars). Likewise, homogeneous cells X (and isotropic edges $\{X, Y\}$) are called *empty* if the graph $G[X]$ (respectively, $G[X, Y]$) is empty, and *complete* otherwise.

Note that if an edge $\{X, Y\}$ of a uniform path or cycle is a constellation, then $G[X, Y]$ is a matching graph.

LEMMA 4.1. *The cell graph $C(G)$ of an amenable graph G has the following properties:*

(C) *$C(G)$ contains no uniform anisotropic path connecting two heterogeneous cells;*

(D) *$C(G)$ contains no uniform anisotropic cycle;*

(E) *$C(G)$ contains neither an anisotropic path $XY_1 \ldots Y_l Z$ such that $|X| < |Y_1| = \cdots = |Y_l| > |Z|$ nor an anisotropic cycle $XY_1 \ldots Y_l X$ such that $|X| < |Y_1| = \cdots = |Y_l|$;*

(F) *$C(G)$ contains no anisotropic path $XY_1 \ldots Y_l$ such that $|X| < |Y_1| = \cdots = |Y_l|$ and the cell Y_l is heterogeneous.*

PROOF. **(C)** Suppose that P is a uniform anisotropic path in $C(G)$ connecting two heterogeneous cells X and Y. Let $k = |X| = |Y|$. Complementing $G[A, B]$ for each co-constellation edge $\{A, B\}$ of P, in G we obtain k vertex-disjoint paths connecting X and Y. These paths determine a one-to-one correspondence between X and Y. Given $v \in X$, denote its mate in Y by v^*. Call P *conducting* if this correspondence is an isomorphism between $G[X]$ and $G[Y]$, that is, two vertices u and v in X are adjacent exactly when their mates u^* and v^* are adjacent. We also call P *conducting* if one of X and Y is a matching and the other is a co-matching, and additionally the correspondence is an isomorphism between $G[X]$ and the complement of $G[Y]$.

We construct a non-isomorphic graph H such that CR does not distinguish between G and H. Since X and Y are heterogeneous, we can replace the edges of the subgraph $G[X]$ with the edges of an isomorphic but different graph on the same vertex set X such that P is a conducting path in the resulting graph H if and only if P is a non-conducting path in G. Now, Part (i) of Lemma 3.4 implies that CR computes the same coloring for G and H and does not distinguish between them. On the other hand, Lemma 2.2 implies that any isomorphism ϕ between G and H must map each cell to itself. Since $\phi(v^*) = \phi(v)^*$, ϕ must also preserve the conducting property along the path P. It follows that G and H are not isomorphic. Hence, G is not amenable.

(D) Suppose that $C(G)$ contains a uniform anisotropic cycle Q of length m. All cells in Q have the same cardinality, say k. Complementing $G[A, B]$ for each co-constellation edge $\{A, B\}$ of Q, in G we obtain the vertex-disjoint union of cycles whose lengths are multiples of m. As two extreme cases, we can have k cycles of length m each or we can have a single cycle of length km. Denote the isomorphism type of this union of cycles by $\tau(Q)$. Note that this type is isomorphism invariant: For an isomorphism ϕ from G to another graph H, $\tau(\phi'(Q)) = \tau(Q)$ for the induced isomorphism ϕ' from $C(G)$ to $C(H)$.

Let X and Y be two consecutive cells in Q. We can replace the subgraph $G[X, Y]$ with an isomorphic but different bipartite graph so that in the resulting graph H, $\tau(Q)$ becomes either kC_m or C_{km}, whatever we wish. In particular, we can replace the subgraph $G[X, Y]$ in such a way that $\tau(Q)$ is changed.

Similarly as for Condition **C**, we use Part (ii) of Lemma 3.4 to argue that CR does not distinguish between G and H. Furthermore, $G \not\cong H$ because the types $\tau(Q)$ in G and H are different. Therefore, G is not amenable.

(E) Suppose that $C(G)$ contains an anisotropic path $P = XY_1 \ldots Y_l Z$ such that $|X| < |Y_1| = \cdots = |Y_l| > |Z|$ (for the case of a cycle, where $Z = X$, the argument is virtually the same). Let $G[X, Y_1] = sK_{1,t}$ and $G[Z, Y_l] = aK_{1,b}$, where $s, a, t, b \geq 2$ (if a subgraph is a co-constellation, we consider its complement). Thus, $|X| = s$, $|Z| = a$, and $|Y_1| = |Y_l| = st = ab$.

Like in the proof of Condition **C**, the uniform anisotropic path $Y_1 \ldots Y_l$ determines a one-to-one correspondence between the cells Y_1 and Y_l that can be used to make the identification $Y_1 = Y_l = \{1, 2, \ldots, st\} = Y$. For each $x \in X$, let Y_x denote the set of vertices in Y adjacent to x. The set Y_z is defined similarly for each $z \in Z$. Note that for all $x \neq x'$ in X and $z \neq z'$ in Z,

$$|Y_x| = t, \quad |Y_z| = b, \quad Y_x \cap Y_{x'} = \emptyset, \quad \text{and} \quad Y_z \cap Y_{z'} = \emptyset.$$

We regard $\mathcal{Y}_G = \{\, Y_x \,\}_{x \in X} \cup \{\, Y_z \,\}_{z \in Z}$ as a hypergraph on the vertex set Y. Note that $\mathcal{Y}_G$ has multiple hyperedges if $Y_x = Y_z$ for some x and z. Without loss of generality, we can assume that the hyperedges Y_z, $z \in Z$, form consecutive intervals in Y. We call the anisotropic path P *flat*, if there exists no pair $(x, z) \in X \times Z$ such that one of the two hyperedges Y_x and Y_z is contained in the other.

We construct a non-isomorphic graph H such that CR does not distinguish between G and H. If P is flat in G, we replace the edges of the subgraph $G[X, Y_1]$ by the edges of an isomorphic but different biregular graph such that P becomes non-flat in the resulting graph H. More precisely, we replace the edges in such a way that all hyperedges of $\mathcal{Y}_H$ form consecutive intervals in Y by letting $\mathcal{Y}_H = \{\, Y_i \,\}_{i \in [s]} \cup \{\, Y_z \,\}_{z \in Z}$, where $Y_i = \{(i-1)t + 1, \ldots, it\}$. Likewise, if P is non-flat in G, we replace the edges of $G[X, Y_1]$ such that P becomes flat in H by letting $Y_i = \{i, i+s, \ldots, i+(t-1)s\}$.

Now, Part (i) of Lemma 3.4 implies that CR computes the same coloring for G and H and does not distinguish between them. On the other hand, Lemma 2.2 implies that every isomorphism ϕ between G and H maps each cell to itself. As ϕ must also preserve the flatness property of the path P, it follows that G and H are not isomorphic. Hence, G is not amenable.

(F) Suppose that $C(G)$ contains an anisotropic path $XY_1 \ldots Y_l$ where $|X| < |Y_1| = \cdots = |Y_l|$ and Y_l is heterogeneous. Let

$G[X, Y_1] = sK_{1,t}$ (in the case of a co-constellation, we consider the complement). Since $s, t \geq 2$ and $|Y_1| = st$, the cell Y_l cannot be pentagonal. Considering the complement if needed, we can assume without loss of generality that Y_l is matching. Like in the proof of Condition **E**, the uniform anisotropic path $Y_1 \dots Y_l$ determines a one-to-one correspondence between the cells Y_1 and Y_l that can be used to make the identification $Y_1 = Y_l = \{1, 2, \dots, st\} = Y$. Consider the hypergraph $\mathcal{Y}_G = \{\, Y_x \,\}_{x \in X} \cup \mathcal{E}$, where $Y_x = N(x) \cap Y_1$ and $\mathcal{E}$ consists of the pairs of adjacent vertices in $G[Y_l]$. Now, exactly as in the proof of Condition **E**, we can change the isomorphism type of $\mathcal{Y}_G$ by replacing the edges of the subgraph $G[X, Y_1]$ by the edges of an isomorphic biregular graph. This yields a non-isomorphic graph H that is indistinguishable from G by CR. □

It turns out that Conditions **A**–**F** are not only necessary for amenability (as shown in Lemmas 3.1 and 4.1) but also sufficient. As a preparation, we first prove the following Lemma 4.2 that reveals a tree-like structure of amenable graphs. By an *anisotropic component* of the cell graph $C(G)$, we mean a maximal connected subgraph of $C(G)$ whose edges are all anisotropic. Note that if a vertex of $C(G)$ has no incident anisotropic edges, it forms a single-vertex anisotropic component.

Lemma 4.2. *Suppose that a graph G satisfies Conditions* **A**–**F**. *Then for each anisotropic component A of $C(G)$, the following is true.*

- **(G)** *A is a tree with the following monotonicity property. Let R be a cell in A of minimum cardinality and let A_R be the rooted directed tree obtained from A by rooting A at R. Then $|X| \leq |Y|$ for each directed edge (X, Y) of A_R.*
- **(H)** *A contains at most one heterogeneous vertex. If R is such a vertex, it has minimum cardinality among the cells of A.*

Proof. **(G)** A cannot contain a uniform cycle by Condition **D** and any other cycle by Condition **E**. The monotonicity property follows from Condition **E**.

(H) Assume that A contains more than one heterogeneous cell. Consider two such cells S and T. Let $S = Z_1, Z_2, \dots, Z_l = T$ be the path from S to T in A. The monotonicity property stated in Condition **G** implies that there is an index $j \in \{1, \ldots, l\}$ such that $|Z_1| \geq \cdots \geq |Z_j| \leq \cdots \leq |Z_l|$. Since the path cannot be uniform by Condition **C**, at least one of the inequalities is strict. However, this contradicts Condition **F**.

Suppose that R is a heterogeneous cell in A. Consider now a path $R = Z_1, Z_2, \ldots, Z_l = S$ in A where S is a cell with the smallest cardinality. By the monotonicity property and Condition **F**, this path must be uniform, proving that $|R| = |S|$. □

In combination with Conditions **A** and **B**, Conditions **G** and **H** on anisotropic components give a very stringent characterization of amenability.

THEOREM 4.3. *For a graph G, the following conditions are equivalent:*

(i) G is amenable.

(ii) G satisfies Conditions **A**–**F**.

(iii) G satisfies Conditions **A, B, G** *and* ***H***.

PROOF. It only remains to show that each graph G fulfilling the Conditions **A, B, G** and **H** is amenable. Let H be a graph indistinguishable from G by CR. Then, we have to show that G and H are isomorphic.

Consider the coloring C^s corresponding to the stable partition $\mathcal{P}^s$ of the disjoint union $G+H$. Since G and H satisfy Equality (2.3) for $i = s$, there is a bijection $f : \mathcal{P}_G \to \mathcal{P}_H$ matching each cell X of the stable partition of G to the cell $f(X) \in \mathcal{P}_H$ such that the vertices in X and $f(X)$ have the same C^s-color. Moreover, Equality (2.3) implies that $|X| = |f(X)|$. We claim that for all cells X and Y of G,

(a) $G[X] \cong H[f(X)]$ and

(b) $G[X, Y] \cong H[f(X), f(Y)]$,

implying that f is an isomorphism from $C(G)$ to $C(H)$.

Indeed, since X and $f(X)$ are cells of the stable partitions $\mathcal{P}_G$ and $\mathcal{P}_H$, both $G[X]$ and $H[f(X)]$ are regular. Since $X \cup f(X)$

is a cell of the stable partition $\mathcal{P}^s$ of $G + H$, the graphs $G[X]$ and $H[f(X)]$ have the same degree. By Condition **A**, $G[X]$ is a unigraph, implying Property (a). Property (b) follows from Condition **B** by a similar argument.

We now construct an isomorphism ϕ from G to H. According to Lemma 2.2, we should have $\phi(X) = f(X)$ for each cell X. Therefore, we have to define the map $\phi : X \to f(X)$ on each X.

By Condition **H**, an anisotropic component A of the cell graph $C(G)$ contains at most one heterogeneous cell. Denote it by R_A if it exists. Otherwise, fix R_A to be an arbitrary cell of the minimum cardinality in A.

For each A, define ϕ on $R = R_A$ to be an arbitrary isomorphism from $G[R]$ to $H[f(R)]$, which exists according to (a). After this, propagate ϕ to other cells in A as follows. By Condition **G**, A is a tree. Let A_R be the directed rooted tree obtained from A by rooting it at R. Suppose that ϕ is already defined on X and (X, Y) is an edge in A. By the monotonicity property in Condition **G** and our choice of R, we can assume that $|X| \le |Y|$. Then, ϕ can be extended to Y so that this is an isomorphism from $G[X, Y]$ to $H[f(X), f(Y)]$. This is possible by (b) due to the fact that all vertices in Y have degree 1 in $G[X, Y]$ or its bipartite complement (and the same holds for all vertices in $f(Y)$ in the graph $H[f(X), f(Y)]$).

It remains to argue that the map ϕ obtained in this way is indeed an isomorphism from G to H. It suffices to show that ϕ is an isomorphism between $G[X]$ and $H[f(X)]$ for each cell X of G and between $G[X, Y]$ and $H[f(X), f(Y)]$ for each pair of cells X and Y.

If X is homogeneous, $f(X)$ is homogeneous of the same type, complete or empty, according to (a). In this case, each ϕ is an isomorphism from $G[X]$ to $H[f(X)]$. If X is heterogeneous, the assumption of the lemma says that it belongs to a unique anisotropic component A (and $X = R_A$). Then, ϕ is an isomorphism from $G[X]$ to $H[f(X)]$ by construction.

If $\{X, Y\}$ is an isotropic edge of $C(G)$, then (b) implies that $\{f(X), f(Y)\}$ is an isotropic edge of $C(H)$ of the same type, complete or empty. In this case, ϕ is an isomorphism from $G[X, Y]$

to $H[f(X), f(Y)]$, no matter how it is defined. If $\{X, Y\}$ is anisotropic, it belongs to some anisotropic component A, and ϕ is an isomorphism from $G[X, Y]$ to $H[f(X), f(Y)]$ by construction. $\square$

5. Examples and applications

Theorem 4.3 is a convenient tool for verifying amenability. For example, amenability of discrete graphs is a well-known fact. Recall that those are graphs whose stable partitions consist of singletons. Since the cell graph has no anisotropic edge in this case, each anisotropic component of a discrete graph consists of a single cell. Hence, Conditions **A** and **B** as well as Conditions **G** and **H** on anisotropic components are fulfilled by trivial reasons.

Checking these four conditions, we can also reprove the amenability of trees. Moreover, our argument extends to the class of forests. The amenability of forests follows also from Ramana *et al.* (1994, Theorem 2.5); here, we give an alternative proof to illustrate applicability of our criterion. Note that the extension to forests is not a straightforward fact because the class of amenable graphs is not closed under disjoint unions (for example, $C_3 + C_4$ is indistinguishable by CR from C_7 and, hence, is not amenable).

Corollary 5.1. *All forests are amenable.*

Proof. A regular acyclic graph is either an empty or a matching graph. This implies Condition **A**. Condition **B** follows from the observation that biregular acyclic graphs are either empty or disjoint unions of stars.

Let $C^*(G)$ be the version of the cell graph $C(G)$ where all empty edges are removed. If $C^*(G)$ contains a cycle, G must contain a cycle as well. Therefore, if G is acyclic, then $C^*(G)$ is acyclic too, and any anisotropic component of $C(G)$ must be a tree. To prove the monotonicity property in Condition **G**, it suffices to show that $C(G)$ cannot contain an anisotropic path $XY_1 \dots Y_l Z$ with $|X| < |Y_1| = \cdots = |Y_l| > |Z|$. But this easily follows since in this case each vertex of the induced subgraph $G[X \cup Y_1 \cup \cdots \cup Y_l \cup Z]$ has degree at least 2 in G, contradicting the acyclicity of G.

To prove Condition **H**, suppose that $C(G)$ contains an anisotropic path $X_0, X_1, \ldots, X_l$ connecting two heterogeneous cells X_0 and X_l. Then, each vertex of the induced subgraph $G[X_0 \cup X_1 \cup \cdots \cup X_{l-1} \cup X_l]$ has degree at least 2 in G, a contradiction. The same contradiction arises if such a path connects a heterogeneous cell X_0 with an arbitrary cell X_l, where $|X_l| < |X_{l-1}|$. Hence, X_0 must have minimum cardinality among all cells belonging to the same anisotropic component. □

The closure properties of amenable graphs under disjoint unions admit a combinatorial characterization in terms of covering graphs. The relationship of the last concept to CR was first noticed by Angluin (1980), who used it in the area of distributed computations. A surjective homomorphism from a graph K onto a graph G is a *covering map* if its restriction to the neighborhood of each vertex in K is bijective. We suppose that G is a finite graph, while K can be infinite. If there is a covering map from K to G (in other terms, K *covers* G), then K is called a *covering graph* of G. Restricting these notions to connected graphs, we say that a graph U is a *universal cover* of a graph G if U covers every covering graph of G. A universal cover $U = U_G$ of G is unique up to isomorphism. Alternatively, U_G can be defined as a tree that covers G. If G is itself a tree, then $U_G \cong G$; otherwise, the tree U_G is infinite. Two graphs G and H have a common covering graph if and only if $U_G \cong U_H$.

A straightforward inductive argument shows that a covering map α preserves the coloring produced by CR, that is, $C^i(u) = C^i(\alpha(u))$ for all i, where C^i is defined by (2.1). This generalizes Lemma 2.2. It follows that, if two graphs G and H have a common covering graph, then

$$\left\{ C^i(u) : u \in V(G) \right\} = \left\{ C^i(v) : v \in V(H) \right\} \text{ for all } i. \tag{5.2}$$

On the other hand, Equality (5.2) is false for all $i \geq n$ if G and H have no common cover and each of the graphs has at most n vertices. This justifies the use of CR by Angluin (1980) for deciding if two given graphs have a common cover. Moreover, if G and H do not have a common cover and have at most n vertices each, then it holds (see Krebs & Verbitsky 2015) that

$$\{ C^{2n}(u) : u \in V(G) \} \cap \{ C^{2n}(v) : v \in V(H) \} = \emptyset. \tag{5.3}$$

Using Theorem 4.3 and the connection between CR and universal covers, we are able to prove the following result.

THEOREM 5.4. *(i) Suppose that G and H are connected amenable graphs. Then $G + H$ is amenable if and only if G is a tree or $U_G \ncong U_H$.*

(ii) Suppose, in addition, that G and H have an equal number of vertices. Then $G + H$ is amenable if and only if G is a tree or $G \ncong H$.

PROOF. (i) We split the proof into two cases depending on whether or not $U_G \cong U_H$, that is, whether or not G and H have a common covering graph.

Suppose first that $U_G \ncong U_H$. By Equality (5.3), we have $\mathcal{P}_{G+H} = \mathcal{P}_G \cup \mathcal{P}_H$, that is, the stable partition of $G+H$ consists of the cells of the stable partitions of G and H. Since G and H are amenable, their cells fulfill Condition **A**. Therefore, Condition **A** holds true also for $G+H$. Since the cell graph $C(G+H)$ is obtained from $C(G)$ and $C(H)$ by joining each cell of G with each cell of H by an anisotropic (empty) edge, also Condition **B** is fulfilled for $G+H$. Moreover, every anisotropic component of $C(G+H)$ is an anisotropic component either of $C(G)$ or of $C(H)$. It follows that $G+H$ satisfies also Conditions **G** and **H** and, therefore, $G+H$ is amenable by Theorem 4.3.

Consider now the case that $U_G \cong U_H$. If G is a tree, then $H \cong G$, and $G+H$ is amenable by Corollary 5.1. Suppose that G is not a tree. We have to show that $G+H$ is not amenable.

Equality (5.2) implies that there is a one-to-one correspondence between the cells in $\mathcal{P}_G$ and $\mathcal{P}_H$ such that the vertices in the corresponding cells always have the same colors in the course of the CR procedure. It follows that $\mathcal{P}_{G+H} = \{ X \cup X' : X \in \mathcal{P}_G \}$, where $X' \in \mathcal{P}_H$ denotes the counterpart of a cell $X \in \mathcal{P}_G$.

Using the existence of a cycle in G, we show that $G+H$ is not amenable by constructing a graph F such that F is connected (hence, non-isomorphic to $G+H$) and indistinguishable from $G+H$ by CR. Our construction of F is based on two pairs of vertices $u, v \in V(G)$ and $u', v' \in V(H)$ satisfying the following conditions:

(a) u and v are adjacent in G, and u' and v' are adjacent in H.

(b) The edge $\{u, v\}$ belongs to a cycle C in G.

(c) Let X and Y be the cells of $\mathcal{P}_G$ containing the vertices u and v, respectively, and let X' and Y' be their counterparts in $\mathcal{P}_H$. Then $u' \in X'$ and $v' \in Y'$.

We obtain F from $G+H$ by switching the edges $\{u, v\}$ and $\{u', v'\}$ to $\{u, v'\}$ and $\{u', v\}$.

In order to show that F is connected, consider two arbitrary vertices x and y in this graph. If both of them are in $V(G)$, they were connected by a path in G. If this path went through the missing edge $\{u, v\}$, it can be rerouted along the cycle C in Condition (b). If both x and y are in $V(H)$, then a path from x to y in H via the missing edge $\{u', v'\}$ can be rerouted also along the cycle C via the new edges $\{u', v\}$ and $\{v', u\}$. Finally, if $x \in V(G)$ and $y \in V(H)$, then these vertices can be connected by a path in F because the two new edges of F are between $V(G)$ and $V(H)$.

If $X = Y$, then u, v, u', and v' are in the same cell $X \cup X'$ of $\mathcal{P}_{G+H}$, and F is indistinguishable from $G + H$ by Part (i) of Lemma 3.4 (applied to the regular subgraph of $G + H$ induced by the cell $X \cup X'$). If $X \neq Y$, then u and u' are in the cell $X \cup X'$, and v and v' are in the cell $Y \cup Y'$ of $\mathcal{P}_{G+H}$. In this case, F is indistinguishable from $G + H$ by Part (ii) of Lemma 3.4 (applied to the biregular bipartite subgraph of $G + H$ induced by the cells $X \cup X'$ and $Y \cup Y'$).

To complete the proof, we have to secure u, v, u', and v' with the properties (a)–(c) above. Choose an edge $\{u, v\}$ of an arbitrary cycle in G. Let K be a common covering graph of G and H, α be a covering map from K to G, and β be a covering map from K to H. Choose u'' to be an arbitrary vertex of K such that $\alpha(u'') = u$. Let v'' be the vertex determined by the conditions that $v'' \in N(u'')$ and $\alpha(v'') = v$. Finally, set $u' = \beta(u'')$ and $v' = \beta(v'')$. These vertices are adjacent because they are images of adjacent vertices u'' and

v'' under a homomorphism. Since covering maps α and β preserve the colorings produced by CR, we have $C^i(u) = C^i(u'') = C^i(u')$ and $C^i(v) = C^i(v'') = C^i(v')$ for all i. This implies Condition (c).

(ii) We split the proof into two cases depending on whether or not $G \cong H$. Suppose first that $G \ncong H$. Since G and H are amenable, they are distinguishable by CR. We use the following fact (Leighton 1982): Two graphs G and H with the same number of vertices are distinguishable by CR if and only if they have no common covering graph, i.e., $U_G \ncong U_H$. Now, the amenability of $G + H$ readily follows from Part (i).

If $G \cong H$, then $U_G \cong U_H$. It follows directly from Part (i) that $G + H$ is amenable if G is a tree and not amenable otherwise. □

Theorem 5.4 easily extends to disjoint unions of connected amenable graphs $G_1, \dots, G_k$. If there are two non-tree graphs G_i and G_j sharing a common cover, Part (i) readily implies that $G_1 + \dots + G_k$ is not amenable. If there is no such pair G_i, G_j, then the argument for Part (i) shows that $G_1 + \dots + G_k$ is amenable (here we also need the fact that, by Corollary 5.1, the forest part of $G_1 + \dots + G_k$ is amenable).

Our characterization of amenable graphs via Conditions **A**, **B**, **G** and **H** in Theorem 4.3 leads to an efficient test for amenability of a given graph, that has the same time complexity as CR. It is known (Cardon & Crochemore 1982; see also Berkholz *et al.* 2013) that the stable partition of a given graph G can be computed in time $O((n + m) \log n)$. It is supposed that G is presented by its adjacency list.

Corollary 5.5. *The class of amenable graphs is recognizable in time $O((n+m) \log n)$, where n and m denote the number of vertices and edges of the input graph.*

Proof. Using known algorithms, we first compute the stable partition $\mathcal{P}_G = \{X_1, \dots, X_k\}$ of the input graph G. Let $C^*(G)$ be the version of the cell graph $C(G)$ where all empty edges are removed. We can compute the adjacency list of each vertex X_i of $C^*(G)$ by traversing the adjacency list of an arbitrary vertex $u \in X_i$ and listing all cells X_j that contain a vertex v adjacent to

u. Simultaneously, we compute for each pair (i, j) such that $i = j$ or $\{X_i, X_j\}$ is an edge of $C^*(G)$ the number d_{ij} of neighbors in X_j of a vertex in X_i. Knowing the numbers $|X_i|$, $|X_j|$, and d_{ij} allows us to determine whether all the subgraphs $G[X_i]$ and $G[X_i, X_j]$ fulfill Conditions **A** and **B** of Lemma 3.1.

To check Conditions **G** and **H**, we use breadth-first search in the graph $C^*(G)$ to find all anisotropic components A of $C(G)$ and, simultaneously, to check that each component A is a tree containing at most one heterogeneous cell. If we restart the search from an arbitrary cell in A having minimum cardinality, we can also check for each forward edge of the resulting search tree whether the monotonicity property of Condition **G** is fulfilled. $\square$

We conclude this section by considering logical aspects of our result. A *counting quantifier* $\exists^m$ opens a formula saying that there are at least m elements satisfying some property. Immerman & Lander (1990) discovered an intimate connection between color refinement and 2-variable first-order logic with counting quantifiers. This connection implies that amenability of a graph G is equivalent to its definability in this logic, that is, to the existence of a first-order sentence using only two variables and counting quantifiers that is true on G and false on every graph non-isomorphic to G. Corollary 5.5, recast in formal logic, yields the following fact.

Corollary 5.6. *The class of graphs definable in 2-variable first-order logic with counting quantifiers can be recognized in polynomial time.*

6. Amenable graphs are compact

An $n \times n$ real matrix X is *doubly stochastic* if its elements are nonnegative and all its rows and columns sum up to 1. Doubly stochastic matrices are closed under products and convex combinations. The set of all $n \times n$ doubly stochastic matrices forms the *Birkhoff polytope* $B_n \subset \mathbb{R}^{n^2}$. *Permutation matrices* are exactly 0-1 doubly stochastic matrices. By Birkhoff's Theorem (see, e.g. Brualdi 2006), the $n!$ permutation matrices form precisely the set of all extreme points of B_n. Equivalently, every doubly stochastic matrix is a convex combination of permutation matrices.

Let G and H be graphs with vertex set $\{1,\ldots,n\}$. An isomorphism π from G to H can be represented by the permutation matrix $P_\pi = (p_{ij})$ such that $p_{ij} = 1$ if and only if $\pi(i) = j$. Denote the set of matrices P_π for all isomorphisms π by $\mathrm{Iso}(G,H)$, and let $\mathrm{Aut}(G) = \mathrm{Iso}(G,G)$.

Let A and B be the adjacency matrices of graphs G and H, respectively. If the graphs are uncolored, a permutation matrix X is in $\mathrm{Iso}(G,H)$ if and only if $AX = XB$. For vertex-colored graphs, X must additionally satisfy the condition $X[u,v] = 0$ for all pairs of differently colored u and v, i.e., this matrix must be block diagonal with respect to the color classes. We say that (vertex-colored) graphs G and H are *fractionally isomorphic* if $AX = XB$ for a doubly stochastic matrix X, where $X[u,v] = 0$ if u and v are of different colors. The matrix X is called a *fractional isomorphism.*

Denote the set of all fractional isomorphisms from G to H by $S(G,H)$ and note that it forms a polytope in $\mathbb{R}^{n^2}$. The set of isomorphisms $\mathrm{Iso}(G,H)$ is contained in $\mathrm{Ext}(S(G,H))$, where $\mathrm{Ext}(Z)$ denotes the set of all extreme points of a set Z. Indeed, $\mathrm{Iso}(G,H)$ is the set of integral extreme points of $S(G,H)$.

The set $S(G) = S(G,G)$ is the polytope of *fractional automorphisms* of G. A graph G is called *compact* (Tinhofer 1986) if $S(G)$ has no other extreme points than $\mathrm{Aut}(G)$, i.e., $\mathrm{Ext}(S(G)) = \mathrm{Aut}(G)$. Compactness of G can equivalently be defined by either of the following two conditions.

- ◦ The polytope $S(G)$ is integral;
- ◦ Every fractional automorphism of G is a convex combination of automorphisms of G, i.e., $S(G) = \langle \mathrm{Aut}(G) \rangle$, where $\langle Z \rangle$ denotes the convex hull of a set Z.

Example 6.1. Complete graphs are compact as a consequence of Birkhoff's theorem. The compactness of trees and cycles is established by Tinhofer (1986). Matching graphs mK_2 are also compact. This is a particular instance of a much more general result by Tinhofer (1991): If a connected graph G is compact, then mG is compact for any m. Tinhofer (1991) also observes that compact graphs are closed under complement.

For a negative example, note that the graph $C_3 + C_4$ is not compact. This follows from a general result in Tinhofer (1991): All regular compact graphs must be vertex-transitive (and C_3+C_4 is not). $\Diamond$

Tinhofer (1991) noted that, if G is compact, then for every graph H, either all or none of the extreme points of the polytope $S(G, H)$ are integral. As mentioned in the introduction, this yields a linear programming-based polynomial-time algorithm to test if a compact graph G is isomorphic to another given graph H. The following result shows that Tinhofer's approach works for all amenable graphs.

THEOREM 6.2. *All amenable graphs are compact.*

Theorem 6.2 unifies and extends several earlier results providing examples of compact graphs. In particular, it gives another proof of the fact that almost all graphs are compact, which also follows from a result of Godsil (1997, Corollary 1.6). Indeed, while Babai *et al.* (1980) proved that almost all graphs are discrete, we already mentioned in Section 1 that all discrete graphs are amenable.

Furthermore, Theorem 6.2 subsumes Tinhofer's result that trees are compact.[3] By Corollary 5.1, we can extend this result to forests. This extension is not straightforward as compact graphs are not closed under disjoint union; see Example 6.1. Tinhofer (1989) proves compactness for the class of *strong tree-cographs*, which includes forests only with pairwise non-isomorphic connected components. To the best of our knowledge, compactness of unigraphs, which also follows from Theorem 6.2, has not been observed earlier. Summarizing, we note the following result.

COROLLARY 6.3. *Discrete graphs, forests, and unigraphs are compact.*

In the rest of this section, we prove Theorem 6.2. We will use a known fact on the structure of fractional automorphisms. For a partition $V_1, \dots, V_m$ of $\{1, \dots, n\}$ let $X_1, \dots, X_m$ be matrices,

[3] The proof of Theorem 6.2 uses only compactness of complete graphs, matching graphs, and the 5-cycle.

where the rows and columns of X_i are indexed by elements of V_i. Then, we denote the block diagonal matrix with blocks $X_1, \ldots, X_m$ by $X_1 \oplus \cdots \oplus X_m$. The following result can be considered an analog of Lemma 2.2 for fractional isomorphisms.

LEMMA 6.4 (Ramana *et al.* 1994). *Let G be a (vertex-colored) graph on vertex set $\{1, \ldots, n\}$ and assume that the elements $V_1, \ldots, V_m$ of the stable partition $\mathcal{P}_G$ of G are intervals of consecutive integers. Then, any fractional automorphism X of G has the form $X = X_1 \oplus \cdots \oplus X_m$.*

Note that the assumption of the lemma can be ensured for every graph by appropriately renaming its vertices. An immediate consequence of Lemma 6.4 is that a graph G is compact if and only if it is compact with respect to its stable coloring.

PROOF OF THEOREM 6.2. Given an amenable graph G and a fractional automorphism X of G, we have to express X as a convex combination of permutation matrices in $\mathrm{Aut}(G)$. Our proof strategy consists in exploiting the structure of amenable graphs as described by Theorem 4.3. Given an anisotropic component A of the cell graph $C(G)$, we define the *anisotropic component G_A of G* as the subgraph of G induced by the union of all cells belonging to A. Our overall idea is to prove the claim separately for each anisotropic component G_A, applying an inductive argument on the number of cells in A. A key role will be played by the fact that, according to Theorem 4.3, A is a tree with at most one heterogeneous cell.

We can assume that G is colored by the stable coloring because, by Lemma 6.4, the colored version has the same polytope of fractional automorphisms. We first consider the case when G consists of a single anisotropic component A. By Theorem 4.3, the corresponding cell graph $C(G)$ has at most one heterogeneous vertex, and A forms a spanning tree of $C(G)$. Without loss of generality, we can number the cells $V_1, \ldots, V_m$ of G so that V_1 is the unique heterogeneous cell if it exists; otherwise, V_1 is chosen among the cells of minimum cardinality. Moreover, we can suppose that, for each $i \leq m$, the cells $V_1, \ldots, V_i$ induce a connected subgraph in the tree A.

We will prove by induction on $i = 1, \ldots, m$ that the graphs $G_i = G[V_1 \cup \cdots \cup V_i]$ are compact. In the base case of $i = 1$, the graph $G_1 = G[V_1]$ is one of the graphs listed in Condition **A** of Theorem 4.3. All of them are known to be compact; see Example 6.1. As induction hypothesis, assume that the graph G_{i-1} is compact. For the induction step, we have to show that also G_i is compact.

Denote $D = V_i$. Since G has no more than one heterogeneous cell, $G[D]$ is complete or empty. It will be instructive to think of D as a "leaf" cell having a unique anisotropic link to the remaining part G_{i-1} of G_i. Let $C \in \{V_1, \ldots, V_{i-1}\}$ be the unique cell such that $\{C, D\}$ is an anisotropic edge of $C(G_i)$. To be specific, suppose that $G[C, D] \cong sK_{1,t}$. If $G[C, D]$ is the bipartite complement of $sK_{1,t}$, we can consider the complement of G_i, using the fact that the polytope of fractional automorphisms is the same for a graph and its complement. By the monotonicity property stated in Condition **C** of Theorem 4.3, $|C| = s$ and $|D| = st$. Let $C = \{c_1, c_2, \ldots, c_s\}$ and, for each j, $N(c_j) \subseteq D$ be the neighborhood of c_j in $G[C, D]$. Thus, $D = \bigcup_{j=1}^{s} N(c_j)$.

Let X be a fractional automorphism of G_i. It is convenient to break it up into three blocks $X = X' \oplus Y \oplus Z$, where Y and Z correspond to C and D, respectively, and X' is the rest. By induction hypothesis, we have the convex combination

$$X' \oplus Y = \sum_{P' \oplus P \in \mathrm{Aut}(G_{i-1})} \alpha_{P',P}\, P' \oplus P, \tag{6.5}$$

where $P' \oplus P$ are permutation matrices corresponding to automorphisms π of the graph G_{i-1}, such that the permutation matrix block P denotes the action of π on the color class C and P' the action on the remaining color classes of G_{i-1}.

We need to show that X is a convex combination of automorphisms of G_i. Let A denote the adjacency matrix of G_i, and $A_{S,T}$ denote the submatrix of A row-indexed by $S \subset V(G_i)$ and column-indexed by $T \subset V(G_i)$. Since X is a fractional automorphism of G_i, we have $XA = AX$. Recall that Y and Z are blocks of X corresponding to color classes C and D. Looking at the corner fragments of the matrices XA and AX, we get

$$\begin{pmatrix} Y & 0 \\ 0 & Z \end{pmatrix} \begin{pmatrix} A_{C,C} & A_{C,D} \\ A_{D,C} & A_{D,D} \end{pmatrix} = \begin{pmatrix} A_{C,C} & A_{C,D} \\ A_{D,C} & A_{D,D} \end{pmatrix} \begin{pmatrix} Y & 0 \\ 0 & Z \end{pmatrix},$$

which implies

$$YA_{C,D} = A_{C,D}\,Z, \tag{6.6}$$
$$A_{D,C}\,Y = Z\,A_{D,C}. \tag{6.7}$$

Consider Z as an $st \times st$ matrix whose rows and columns are indexed by the elements of sets $N(c_1), N(c_2), \ldots, N(c_r)$ in that order. We can thus think of Z as an $s \times s$ block matrix of $t \times t$ matrix blocks $Z^{(k,\ell)}, 1 \le k, \ell \le s$. The next claim is a consequence of Eqs. (6.6) and (6.7).

Claim 6.8. *Each block $Z^{(k,\ell)}$ in Z is of the form*

$$Z^{(k,\ell)} = y_{k,\ell} W^{(k,\ell)}, \tag{6.9}$$

where $y_{k,\ell}$ is the (k,ℓ)th entry of Y, and $W^{(k,\ell)}$ is a doubly stochastic matrix.

Proof. We first note from Eq. (6.6) that the (k,j)th entry of the $s \times st$ matrix $YA_{C,D} = A_{C,D}Z$ can be computed in two different ways. In the left-hand side matrix, it is $y_{k,\ell}$ for each $j \in N(c_\ell)$. On the other hand, the right-hand side matrix implies that the same (k,j)th entry is also the sum of the jth column of the $N(c_k) \times N(c_\ell)$ block $Z^{(k,\ell)}$ of the matrix Z.

We conclude, for $1 \le k, \ell \le s$, that each column in $Z^{(k,\ell)}$ adds up to $y_{k,\ell}$. By a similar argument, applied to Eq. (6.7) this time, it follows, for each $1 \le k, \ell \le s$, that for all blocks $Z^{(k,\ell)}$ of Z, each *row* of the block adds up to $y_{k,\ell}$.

We conclude that, if $y_{k,\ell} \neq 0$, then the matrix $W^{(k,\ell)} = \frac{1}{y_{k,\ell}} Z^{(k,\ell)}$ is doubly stochastic. If $y_{k,\ell} = 0$, then (6.9) is true for all choices of $W^{(k,\ell)}$. □

For every $P = (p_{k\ell})$ appearing in an automorphism $P' \oplus P$ of G_{i-1} (see Eq. (6.5)), we define the $st \times st$ doubly stochastic matrix W_P by its $t \times t$ blocks indexed by $1 \le k, \ell \le s$ as follows:

$$W_P^{(k,\ell)} = \begin{cases} W^{(k,\ell)} & \text{if } p_{k\ell} = 1, \\ 0 & \text{if } p_{k\ell} = 0. \end{cases} \tag{6.10}$$

Equations (6.5) and (6.9) imply that

$$(6.11) \qquad X = X' \oplus Y \oplus Z = \sum_{P' \oplus P \in \mathrm{Aut}(G_{i-1})} \alpha_{P',P} \, P' \oplus P \oplus W_P.$$

In order to see this, on the left-hand side consider the (k,ℓ)th block $Z^{(k,\ell)}$ of Z. On the right-hand side, note that the corresponding block in each $P' \oplus P \oplus W_P$ is the matrix $W^{(k,\ell)}$. Clearly, the overall coefficient for this block equals the sum of $\alpha_{P',P}$ over all P' and P such that $p_{k,\ell} = 1$, which is precisely $y_{k,\ell}$ by Eq. (6.5).

Since each $W^{(k,\ell)}$ is a doubly stochastic matrix, by Birkhoff's theorem we can write it as a convex combination of $t \times t$ permutation matrices $Q_{j,k,\ell}$, whose rows are indexed by elements of $N(c_k)$ and columns by elements of $N(c_\ell)$:

$$W^{(k,\ell)} = \sum_{j=1}^{t!} \beta_{j,k,\ell} \, Q_{j,k,\ell}.$$

Substituting the above expression in Eq. (6.10), that defines the doubly stochastic matrix W_P, we express W_P as a convex combination of permutation matrices $W_P = \sum_Q \delta_{Q,P} \, Q$ where Q runs over all $st \times st$ permutation matrices indexed by the vertices in color class D. Notice here that $\delta_{Q,P}$ is nonzero only for those permutation matrices Q that have structure similar to that described in Eq. (6.10): The block $Q^{(k,\ell)}$ is a null matrix if $p_{k\ell} = 0$ and it is some $t \times t$ permutation matrix if $p_{k\ell} = 1$. For each such Q, the $(s + st) \times (s + st)$ permutation matrix $P \oplus Q$ is an automorphism of the subgraph $G_i[C, D] = sK_{1,t}$ (because Q maps $N(c_i)$ to $N(c_j)$ whenever P maps c_i to c_j). Since $P \in \mathrm{Aut}(G_i[C])$ and D is a homogeneous set in G_i, we conclude that, moreover, $P \oplus Q$ is an automorphism of the subgraph $G_i[C \cup D]$.

Now, if we plug the above expression for each W_P in Eq. (6.11), we will finally obtain the desired convex combination

$$X = \sum_{P',P,Q} \gamma_{P',P,Q} \, P' \oplus P \oplus Q.$$

It remains to argue that every $P' \oplus P \oplus Q$ occurring in this sum is an automorphism of G_i. Recall that a pair P', P can appear here only if $P' \oplus P \in \mathrm{Aut}(G_{i-1})$. Moreover, if such a pair is extended to a matrix $P' \oplus P \oplus Q$, then $P \oplus Q \in \mathrm{Aut}(G_i[C \cup D])$. Since $G_i[B, D]$ is isotropic for every color class $B \neq D$ of G_i, we conclude that $P' \oplus P \oplus Q \in \mathrm{Aut}(G_i)$. This completes the induction step and finishes the case when G has one anisotropic component.

Next, we consider the case when $C(G)$ has several anisotropic components $T_1, \ldots, T_k$, $k \geq 2$. Let $G_1, \ldots, G_k$, where $G_i = G[\bigcup_{U \in V(T_i)} U]$, be the corresponding anisotropic components of G. By the proof of the previous case, we already know that G_i is compact for each i.

Claim 6.12. *The automorphism group* $\mathrm{Aut}(G)$ *of* G *is the product of the automorphism groups* $\mathrm{Aut}(G_i)$, $1 \leq i \leq k$.

Proof. Recall that an automorphism of G must map each color class of G, which is a cell of the underlying amenable graph, onto itself. Thus, an automorphism π of G is of the form $(\pi_1, \ldots, \pi_k)$, where π_i is an automorphism of the subgraph G_i. Now, for two subgraphs G_i and G_j, we examine the edges between $V(G_i)$ and $V(G_j)$. For color classes $U \subseteq V(G_i)$ and $U' \subseteq V(G_j)$, the edge $\{U, U'\}$ is isotropic because it is not contained in an anisotropic component of $C(G)$. Therefore, the bipartite graph $G[U, U']$ is either complete or empty. It follows that for automorphisms π_i of G_i, $1 \leq i \leq k$, the permutation $\pi = (\pi_1, \ldots, \pi_k)$ is an automorphism of the graph G. □

As follows from Lemma 6.4, a fractional automorphism X of G is of the form $X = X_1 \oplus \cdots \oplus X_k$, where X_i is a fractional automorphism of G_i for each i. As each G_i is compact, we can write each X_i as a convex combination

$$X_i = \sum_{\pi \in \mathrm{Aut}(G_i)} \alpha_{i,\pi} P_\pi.$$

This implies

$$\begin{aligned} (6.13) \quad & I \oplus \cdots \oplus I \oplus X_i \oplus I \oplus \cdots \oplus I \\ & \qquad = \sum_{\pi \in \mathrm{Aut}(G_i)} \alpha_{i,\pi} \, I \oplus \cdots \oplus I \oplus P_\pi \oplus I \oplus \cdots \oplus I, \end{aligned}$$

where block diagonal matrices in the above expression have X_i and P_π, respectively, in the ith block (indexed by elements of $V(G_i)$) and identity matrices as the remaining blocks.

We now decompose the fractional automorphism X as a matrix product of fractional automorphisms of G

$$
\begin{aligned}
(6.14)\quad X &= X_1 \oplus \cdots \oplus X_k \\
&= (X_1 \oplus I \oplus \cdots \oplus I) \cdot \cdots \cdot (I \oplus \cdots \oplus I \oplus X_k).
\end{aligned}
$$

Substituting for $I \oplus \cdots \oplus I \oplus X_i \oplus I \oplus \cdots \oplus I$ from Eq. (6.13) in the above expression and writing the product of sums as a sum of products, we see that X is a convex combination of permutation matrices of the form $P_{\pi_1} \oplus \cdots \oplus P_{\pi_k}$ where $\pi_i \in \mathrm{Aut}(G_i)$ for each i. By Claim 6.12, all the terms $P_{\pi_1} \oplus \cdots \oplus P_{\pi_k}$ correspond to automorphisms of G. Hence, G is compact, completing the proof of Theorem 6.2. $\square$

7. A color refinement-based hierarchy of graphs

Let $u \in V(G)$ and $v \in V(H)$ be vertices of two graphs G and H. By *individualization* of u and v, we mean assigning the same *new color* to u and v, which makes them distinguished from the remaining vertices of G and H. Tinhofer (1991, Theorem 4) has shown that if G is compact, then the following polynomial-time algorithm correctly decides whether G and H are isomorphic.

TINHOFER'S ALGORITHM

1. Run CR on $G+H$ until the coloring of $V(G) \cup V(H)$ stabilizes.

2. If the multisets of colors in G and H are different, then output "non-isomorphic" and stop. Otherwise, each color class contains equally many vertices from G and from H. In this case,

   (a) if all color classes of $G+H$ are doubletons (that is, each contains one vertex from G and one from H), output "isomorphic" and stop;

(b) else pick a color class with at least two vertices in both G and H, select an arbitrary $u \in V(G)$ and $v \in V(H)$ in this color class and individualize them. Goto Step 1.

If G and H are two arbitrary non-isomorphic graphs, then Tinhofer's algorithm will always output "non-isomorphic." Indeed, the output "isomorphic" is possible only in Step 2(a) of some iteration of the algorithm. At this point, the coloring of $G + H$ determines a one-to-one correspondence between $V(G)$ and $V(H)$. This correspondence is an isomorphism between G and H for else the coloring of $G + H$ computed in Step 1 would not be stable.

However, Tinhofer's algorithm can fail for isomorphic input graphs, in general. We call G a *Tinhofer graph* if the algorithm works correctly on G and every H for all choices of vertex pairs to be individualized (as specified in Step 2(b)). Thus, the result of Tinhofer (1991) can be stated as the inclusion $\mathsf{Compact} \subseteq \mathsf{Tinhofer}$.

We now notice that, if G is a Tinhofer graph, then Tinhofer's algorithm can be used to even find a *canonical labeling* of G, i.e., a relabeling of the vertices converting isomorphic graphs to the same graph. To see this, consider an equivalent *canonical coloring* problem: Given a graph G, one has to find its vertex-colored version G^* with all vertices having pairwise distinct colors such that $G^* \cong H^*$ for every two isomorphic input graphs G and H (an isomorphism between G^* and H^* is unique as it is completely determined by the vertex coloring). Immerman & Lander (1990) (independently of Tinhofer 1991) consider the following non-backtracking refinement-individualization procedure, that can be regarded as a version of Tinhofer's algorithm for a single-input graph G.

Canonical coloring algorithm

1. Run CR on G until the coloring of $V(G)$ stabilizes.

2. (a) If all color classes are singletons, return this coloring of G as the output G^*.

   (b) If there are non-singleton color classes, choose the one with the lexicographically least color and pick an arbitrary vertex u in this class. Individualize u by coloring it in the lexicographically least available (unused) color. Goto Step 1.

We already mentioned in Section 2 that the CR algorithm, invoked in Step 1, has to rename the colors after each refinement round in order to keep the color names short. We suppose that the renaming scheme depends only on the multiset of colors produced by the refinement round (and does not otherwise depend on the input graph G). Note that this implies the following feature of the CR subroutine. For a vertex-colored graph A, let A' denote the graph obtained from A by applying CR. Then, for all pairs of graphs A and B, an isomorphism between A and B is also an isomorphism between A' and B'. For more efficient implementations of CR, this issue is addressed by Berkholz *et al.* (2013).

LEMMA 7.1. *Let G be a Tinhofer graph. Then, for all choices of vertices to be individualized, the output G^* of the above algorithm is a canonical coloring of G.*

PROOF. Given an arbitrary graph H isomorphic to a Tinhofer graph G, run the algorithm on G and on H. For the outputs G^* and H^*, we have to prove that $H^* \cong G^*$. Let $G_0 = G$ and let G_i be the colored version of G after the i-th iteration of the algorithm. The graphs H_i for $i \geq 0$ are defined similarly. Let u_i and v_i be the vertices individualized in the i-th iteration of Step 2(b) in G and H, respectively. Using the induction on i, we will prove the following.

CLAIM 7.2. *If on input G or on input H, the algorithm executes Step 2(b) in the i-th iteration, then*

(i) the algorithm executes Step 2(b) in the i-th iteration on input G and on input H;

(ii) u_i and v_i have the same color in $(G_{i-1})'$ and $(H_{i-1})'$ (recall that A' denotes the color-refined version of a graph A);

(iii) u_i and v_i have the same color in G_i and H_i;

(iv) $G_i \cong H_i$.

Part (i) implies that the algorithm performs the same number t of iterations on both G and H, i.e., $G^* = G_t$ and $H^* = H_t$. Moreover, by Part (iv) we have $G_{t-1} \cong H_{t-1}$, implying that $H^* \cong G^*$, as $G_t = (G_{t-1})'$ and $H_t = (H_{t-1})'$.

It remains to prove the claim. It is obviously true for $i = 0$. Using the fact that $G_0 \cong H_0$ and the induction assumption that the claim it true for the indices $0, \ldots, i-1$, we show that it is also true for the index i.

Part (i) readily follows from the isomorphism $G_{i-1} \cong H_{i-1}$ because the refined versions of these graphs are also isomorphic, i.e., $(G_{i-1})' \cong (H_{i-1})'$. The last relation readily implies parts (ii) and (iii). Part (ii) follows because u_i and v_i are selected in suitable color classes with the lexicographically least color, which is the same in isomorphic graphs. Part (iii) follows because both u_i and v_i are assigned the lexicographically least color absent in $(G_{i-1})'$ (or, equivalently, in $(H_{i-1})'$).

Let us prove part (iv). For every $j < i$, we already know that $G_j \cong H_j$ and, hence, $(G_j)' \cong (H_j)'$. Moreover, u_{j+1} and v_{j+1} have the same color in $(G_j)'$ and $(H_j)'$. This implies that the pairs $(u_1, v_1), \ldots, (u_i, v_i)$ are a legitimate choice for individualization in the first i iterations of Tinhofer's algorithm run on input (G, H). It follows that G_i and H_i must be isomorphic because otherwise Tinhofer's algorithm would output "non-isomorphic" on the input (G, H), contradicting the condition that G is a Tinhofer graph. $\square$

Combining Lemma 7.1 with Tinhofer (1991, Theorem 4) and our Theorem 6.2, we obtain the following result.

Corollary 7.3. *Tinhofer and, in particular, amenable and compact graphs admit canonical labeling in polynomial time.*

Immerman & Lander (1990) show that the canonical coloring algorithm works correctly on all graphs that are characterizable in two-variable counting logic in the sense of Immerman & Lander (1990, Definition 1.4.2). This class of graphs actually coincides with the class of amenable graphs, as it readily follows from the logical characterization of CR by Immerman & Lander (1990) and the fact that the class of amenable graphs is closed under finer colorings (see Kiefer *et al.* 2015, Corollary 7). This gives another way of proving Corollary 7.3 in the particular case of amenable graphs.

Recall that a partition $\mathcal{P}$ of the vertex set of a graph G is equitable if all elements of $\mathcal{P}$ are monochromatic and, for sets $X, Y \in \mathcal{P}$, every vertex $x \in X$ has the same number of neighbors in Y. Let A be a subgroup of the automorphism group $Aut(G)$ of a graph G. Then, the partition of $V(G)$ into the A-orbits is called an *orbit partition* of G. An orbit partition of G is equitable, but the converse is not true, in general. However, Godsil (1997, Corollary 1.3) has shown that the converse holds for compact graphs. We define *Godsil graphs* as the graphs for which the two notions of an equitable and an orbit partition coincide, that is, every equitable partition is the orbit partition of some subgroup A of $Aut(G)$. Thus, the result of Godsil (1997) can be stated as the inclusion $\mathsf{Compact} \subseteq \mathsf{Godsil}$. Now, the inclusion $\mathsf{Compact} \subseteq \mathsf{Tinhofer}$ can easily be strengthened as follows.

Lemma 7.4. *Godsil graphs are Tinhofer graphs.*

Proof. Assume that G is a Godsil graph. It suffices to show that Tinhofer's algorithm is correct whenever G and H are isomorphic. Let ϕ be an isomorphism from G to H. We will prove that, after the i-th refinement step made by the algorithm, there exists an isomorphism ϕ_i from G to H that preserves colors of the vertices. If this is true for each i, the algorithm terminates only when the discrete partition is reached on each of the input graphs and, therefore, decides isomorphism.

We prove the claim by induction on i. At the beginning, $\phi_1 = \phi$. Assume that an isomorphism ϕ_i exists and the partition is still not discrete. Suppose that now the algorithm individualizes $u \in V(G)$ and $v \in V(H)$. If $v = \phi_i(u)$, then $\phi_{i+1} = \phi_i$. Otherwise, consider the vertices u and $\phi_i^{-1}(v)$, which are in the same monochromatic class of G. Note that the partition of G produced in each refinement step is equitable. Since G is a Godsil graph, there is an automorphism α preserving the partition such that $\alpha(u) = \phi_i^{-1}(v)$. We can, therefore, take $\phi_{i+1} = \phi_i \circ \alpha$. $\square$

The orbit partition of G with respect to $Aut(G)$ is always a refinement of the stable partition $\mathcal{P}_G$ of G. We call G *refinable* if $\mathcal{P}_G$ is exactly the orbit partition of $Aut(G)$. A Godsil graph is refinable by definition. It is easy to show that Tinhofer graphs are also refinable.

LEMMA 7.5. *Tinhofer graphs are refinable.*

PROOF. Suppose that a graph G is not refinable. We show that G is not a Tinhofer graph. By assumption, G has vertices u and v that are in different orbits but belong to the same element of the stable partition $\mathcal{P}_G$. Let us run Tinhofer's algorithm on two copies G' and G'' of G and, after the refinement step, individualize u in G' and v in G''. At this stage, G' and G'' are non-isomorphic colored graphs, because u and v have the same unique color and there is no isomorphism between G' and G'' that maps u to v. Thus, if we continue running Tinhofer's algorithm on this pair of graphs, when it terminates it will incorrectly output that G' and G'' are not isomorphic. It follows that G is not a Tinhofer graph. $\square$

Summarizing Theorem 6.2, Lemmas 7.4 and 7.5, and Godsil (1997, Corollary 1.3), we obtain the following hierarchy result.

THEOREM 7.6. *The classes of graphs under consideration form the inclusion chain*

$$\mathsf{Discrete} \subset \mathsf{Amenable} \subset \mathsf{Compact} \subset \mathsf{Godsil} \subset \mathsf{Tinhofer} \subset \mathsf{Refinable}. \tag{7.7}$$

Moreover, all of the inclusions are strict.

It is worth noting that the hierarchy (7.7) collapses to Discrete if we restrict ourselves to only asymmetric graphs, i.e., graphs with trivial automorphism group. Indeed, all discrete graphs are asymmetric, and asymmetric refinable graphs are obviously discrete. The fact that asymmetric compact graphs are discrete was observed by Godsil (1997, Corollary 1.6).

PROOF OF THEOREM 7.6. The following separating examples prove that all inclusions are strict.

Separation of Discrete *and* Amenable*:* For $n \geq 2$, the complete graph K_n is amenable but not discrete.

Separation of Amenable *and* Compact*:* For $n \geq 6$, the cycles C_n are not amenable, while they are known to be compact graphs (Tinhofer 1986, Theorem 2). For another family of separating examples, consider the disjoint union $2G$ of two copies of an arbitrary connected amenable graph G that is not a tree. By Part (ii) of Theorem 5.4, $2G$ is not amenable. On the other hand, G is compact by Theorem 6.2, and $2G$ is compact as well by the closure property of compact graphs established by Tinhofer (1991).

Separation of Compact *and* Godsil*:* These classes are separated by the Petersen graph. Evdokimov *et al.* (1999, Corollary 5.4) prove that the Petersen graph is not compact. It remains to show that the Petersen graph belongs to the class Godsil. This problem is solvable by modern computer algebra tools; see Ziv-Av (2013) where equitable and orbit partitions are counted for various strongly regular graphs, including the Petersen graph. We give a non-computer-assisted proof in Appendix A.

Separation of Godsil *and* Tinhofer*:* These classes are separated by the Johnson graphs $J(n, 2)$ for $n \geq 7$. The *Johnson graph* $J(n, k)$ has the k-element subsets of $[n] = \{1, \dots, n\}$ as vertices; two vertices are adjacent if their intersection consists of $k - 1$ elements. Note that $J(n, 1) = K_n$. Furthermore, the graph $J(n, 2)$ is the line graph of K_n: It has all 2-element subsets of $[n]$ as vertices, and two of them are adjacent if their intersection is non-empty. It is noticed by Chan & Godsil (1997) that $J(n, 2)$ is not a Godsil graph for $n \geq 7$. For establishing the separation, in Theorem 7.9 below we prove that $J(n, 2)$ is indeed a Tinhofer graph.

Separation of Tinhofer *and* Refinable*:* Consider the gadget $\mathrm{CFI}(P_i, P_j, P_k)$ depicted in Figure 7.1, with two input pairs P_i and P_j and one output pair P_k. This gadget (Cai *et al.* 1992) has the property that each automorphism must flip an even number of the three pairs P_i, P_j, and P_k. We can combine two instances of $\mathrm{CFI}(P_i, P_j, P_k)$ sharing input pairs $P_i = P_1$ and $P_j = P_2$ but having different output pairs $P_k = P_3$ and $P_k = P_4$, respectively. We assume that the four pairs P_1, P_2, P_3, and P_4 and the intermediate sets of four connecting vertices are all different color classes.

This defines a refinable graph G, also depicted in Figure 7.1, with four color classes P_1, P_2, P_3, and P_4 of size 2, and two color

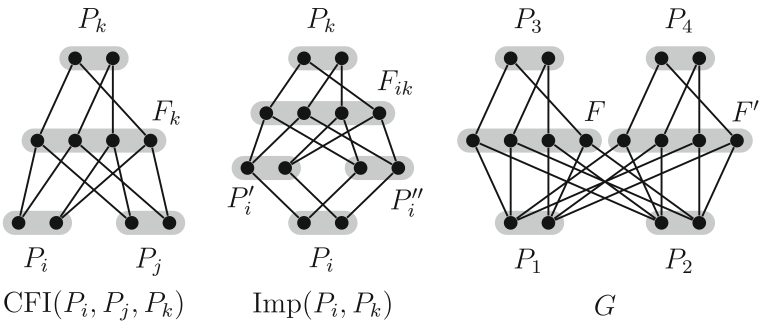


Figure 7.1: The CFI(P_i, P_j, P_k)- and Imp(P_i, P_k)-gadgets and a graph G separating Refinable from Tinhofer

classes F and F' of size 4, that form the orbit partition of G. The graph G has the property that all its automorphisms must flip either both pairs P_3 and P_4 or none of them. Let us run the Tinhofer procedure on two identical copies G' and G'' of G. In the first round, we individualize the same vertex in color class P_3 in both G' and G''. The subsequent refinement will only split color class F. In the second round, let us individualize color class P_4, but now choosing different vertices of P_4 in G' and G''. The resulting colored graphs are non-isomorphic because otherwise there were an isomorphism between G' and G'' flipping exactly one of the color classes P_3 and P_4. The procedure will, therefore, decide non-isomorphism regardless of the subsequent choices of vertices for individualization. It follows that G is not a Tinhofer graph.

Note that the vertex colors of G can be removed if we connect the four vertices in F by six edges and the two vertices in P_1 by one edge. □

We now give a characterization of the Tinhofer graphs that will be used in our proof that Johnson graphs $J(n, 2)$ are not in this class. Given a graph G and a vertex $v \in V(G)$, by $Aut_v(G)$ we denote the stabilizer subgroup of the automorphism group $Aut(G)$ that fixes the vertex v. Let $F \subset V(G)$. Then, we define $Aut_F(G) = \bigcap_{v \in F} Aut_v(G)$. Furthermore, let G_F be the colored version of G where each vertex in F is individualized and let $\mathcal{P}_F(G)$ denote the stable partition of G_F. Note that the orbit partition of $Aut_F(G)$ is a subpartition of $\mathcal{P}_F(G)$.

LEMMA 7.8. *G is a Tinhofer graph if and only if, for every $F \subset V(G)$, the partition of $V(G)$ into the orbits of $Aut_F(G)$ coincides with the partition $\mathcal{P}_F(G)$.*

PROOF. Suppose that the two partitions do not coincide for some F, that is, there are vertices u and v that are in different orbits of $Aut_F(G)$ but belong to the same element of $\mathcal{P}_F(G)$. Run Tinhofer's algorithm on two copies G' and G'' of G, individualizing, one by one, all vertices in F (some of them can get individual colors automatically after refinement steps). After individualizing each vertex in F, G' and G'' become identical colored graphs with color partition $\mathcal{P}_F(G)$. Like in the proof of Lemma 7.5, now we can fool the algorithm by next individualizing u in G' and v in G''. This shows that G is not a Tinhofer graph.

For the other direction, suppose that the partitions are equal for every F. Run Tinhofer's algorithm on G and an isomorphic graph H. Denote the colored versions of G and H after the i-th iteration of the algorithm by G_i and H_i, respectively. If the individualization step was executed in this iteration, that is, it was not the last one, let u_i and v_i denote the vertices that were individualized in G and H, respectively. Also, let $G_0 = G$ and $H_0 = H$. Using the induction on i, we will prove that $G_i \cong H_i$, which will imply that the algorithm decides isomorphism and, therefore, G is a Tinhofer graph. The claim is true for $i = 0$. Assume that $G_{i-1} \cong H_{i-1}$ and let ϕ be an isomorphism from G_{i-1} to H_{i-1}. Note that ϕ is also an isomorphism from $(G_{i-1})'$ to $(H_{i-1})'$, the modified versions of G_{i-1} to H_{i-1} after the subsequent refinement step, respectively. In particular, $\phi(u_j) = v_j$ for all $j \leq i - 1$.

Consider $F = \{u_1, \dots, u_{i-1}\}$ and note that the partition $\mathcal{P}_F(G)$ coincides with the partition of $V(G)$ into color classes of $(G_{i-1})'$. Note also that u_i and $\phi^{-1}(v_i)$ belong to the same element of $\mathcal{P}_F(G)$. By assumption, there is an automorphism $\alpha \in Aut_F(G)$ such that $\alpha(u_i) = \phi^{-1}(v_i)$. It follows that $\phi \circ \alpha$ is an isomorphism from $(G_{i-1})'$ to $(H_{i-1})'$ taking u_i to v_i and, therefore, an isomorphism from G_i to H_i. $\square$

Theorem 7.9. *$J(n, 2)$ is a Tinhofer graph for all n.*

Proof. By Lemma 7.8, it suffices to consider an arbitrary set F of vertices in $G = J(n, 2)$ and to prove that the partition into the orbits of $Aut_F(G)$ and the partition $\mathcal{P}_F(G)$ coincide. One way to do this is to show that each orbit O of $Aut_F(G)$ is definable in terms of F in two-variable first-order logic. Here, "in terms of F" means the existence of a defining formula $\Phi_O(x)$ using constant symbols (names) for each vertex in F. Furthermore, $\Phi_O(x)$ contains occurrences of only two variables, x and y. At least one occurrence of x is free. $\Phi_O(x)$ uses two binary relation symbols $\sim$ and $=$ for adjacency and equality of vertices. This formula is true on G for $x = v$ exactly when $v \in O$.

Once $\Phi_O(x)$ is found for each O, the equality of the partitions follows by a similar argument as in Immerman & Lander (1990, Theorem 1.8.1) or directly from the definitions of orbits, as those will imply that every two orbits are separated by color refinement starting from the individualization of F. The number of refinement steps sufficient to separate O from all other orbits can be only one greater than the quantifier depth of $\Phi_O(x)$.

In order to implement this scenario for $G = J(n, 2)$, it will be convenient to assume that $V(G) = \binom{[n]}{2}$ (note, however, that the formulas $\Phi_O(x)$ do not involve variables over $[n]$). Given $\alpha \in S_n$, by $\ell(\alpha)$ we denote the corresponding permutation of $\binom{[n]}{2}$. Obviously, every $\ell(\alpha)$ is an automorphism of G, and the automorphism group A contains nothing else by the Whitney theorem (1932).

Before formalizing the definitions $\Phi_O(x)$, we will describe the group $Aut_F(G)$ and the orbits of $Aut_F(G)$. We will first do it without using formal logic. Subsequently, we will express the descriptions in two-variable first-order logic.

We now proceed to the detailed proof. Note that $J(2, 2) = K_1$, $J(3, 2) = K_3$, and $J(4, 2)$ is the octahedral graph, whose complement is $K(4, 2) = 3K_2$. Thus, these three graphs are amenable and, hence, Tinhofer. We can, therefore, assume that $n \geq 5$.

Call a fixed vertex $p \in F$ *isolated* if F contains no vertex adjacent to p. Let $F = F_1 \cup F_2$ be the partition of F into non-isolated and isolated vertices. Furthermore, we define the partition

$$[n] = W_1 \cup W_2 \cup W_3$$

as follows: W_1 is the union of all non-isolated pairs p (i.e., all p in F_1), and W_2 is the union of all isolated pairs p (i.e., all p in F_2). Thus, W_3 consists of the points of $[n]$ that are not included in any fixed pair.

Note now that $\ell(\alpha) \in Aut_F(G)$ if and only if α either fixes or transposes the two points in each fixed pair. It follows that $\ell(\alpha) \in Aut_F(G)$ exactly when

- $\alpha(w) = w$ for every $w \in W_1$ and
- $\alpha(p) = p$ for every $p \in F_2$.

Given a vertex $u = \{a, b\}$ of G, let $O(u)$ denote its orbit with respect to $Aut_F(G)$. There are six kinds of orbits. Below we describe all of them along with providing suitable formal definitions $\Phi_{O(u)}(x)$.

Case 1: $\{a, b\} \subseteq W_1$. Then $O(u) = \{u\}$. *Formal definition:* $x = u$.

Case 2: $\{a, b\} \subseteq W_2$. Here, we have two subcases. If $u \in F_2$, then $O(u) = \{u\}$ again. Otherwise, F_2 contains two pairs $p_1 = \{a, a'\}$ and $p_2 = \{b, b'\}$. In this subcase,

$$O(u) = \{\{a, b\}, \{a', b\}, \{a, b'\}, \{a', b'\}\},$$

which is exactly the common neighborhood of p_1 and p_2. *Formal definition:* $x \sim p_1 \wedge x \sim p_2$.

Case 3: $\{a, b\} \subseteq W_3$. Now $O(u) = \binom{W_3}{2}$, which are exactly the non-fixed vertices with no neighbor in F. *Formal definition:* $\bigwedge_{p \in F}(x \neq p \wedge x \not\sim p)$.

Case 4: $a \in W_1$, $b \in W_2$. Let $p = \{b, b'\}$ be the pair in F_2 containing b. Then,

$$O(u) = \{\{a, b\}, \{a, b'\}\}.$$

To give a formal definition of $O(u)$, we consider two subcases.

(i) a belongs to two adjacent vertices $q_1 = \{a, a_1\}$ and $q_2 = \{a, a_2\}$ in F_1.

Formal definition: $x \sim p \wedge x \sim q_1 \wedge x \sim q_2$. Indeed, the condition $x \sim p$ forces x to contain either b or b'. This excludes the possibility that $x = \{a_1, a_2\}$ and, therefore, x is forced to contain a by the adjacency to q_1 and q_2.

(ii) a belongs to a single vertex $q_1 = \{a, a'\}$ in F_1. By definition, F_1 contains also a vertex $q_2 = \{a', a''\}$. *Formal definition:* $x \sim p \wedge x \sim q_1 \wedge x \not\sim q_2$.

Case 5: $a \in W_1$, $b \in W_3$. Then

$$O(u) = \{\, \{a, b'\} : b' \in W_3 \}\,.$$

Similarly to the preceding case, we distinguish two subcases.

(i) a belongs to two adjacent vertices $q_1 = \{a, a_1\}$ and $q_2 = \{a, a_2\}$ in F_1.

Formal definition: First of all, we say that $x \sim q_1 \wedge x \sim q_2$. It remains to exclude the possibility that $x \subseteq W_1 \cup W_2$ (in particular, this will exclude $x = \{a_1, a_2\}$ and force x to contain a). We do this by adding the following expression

$$\text{(7.10)} \quad \bigwedge_{p \in F} x \neq p \wedge \bigwedge_{p,q \in F, p \not\sim q} \neg(x \sim p \wedge x \sim q)$$
$$\wedge \bigwedge_{p,q \in F_1, p \sim q} (x \sim p \wedge x \sim q \to \exists y\, (y \sim x \wedge y \sim p \wedge y \sim q)).$$

The first conjunctive term prevents x to be one of the pairs in F. The second term excludes the case that x is covered by two disjoint pairs p and q in F. The third term excludes the case that x is covered by two intersecting pairs p and q in F or, equivalently, the case where x, p, and q form a triangle. It would be not enough just to forbid x, p, and q from forming a clique because this could also exclude a permissible case where x, p, and q form a star (which is captured by the subformula beginning with $\exists y$). Note that we need the assumption $n \geq 5$ in this place.

(ii) a belongs to a single vertex $q_1 = \{a, a'\}$ in F_1, and $q_2 = \{a', a''\}$ is another vertex in F_1. *Formal definition:* $x \sim q_1 \wedge x \not\sim q_2 \wedge x \not\subseteq W_1 \cup W_2$, the last being expressed by the formula (7.10).

Case 6: $a \in W_2$, $b \in W_3$. In this case, F_2 contains a pair $p = \{a, a'\}$ and

$$O(u) = \{\, \{a, b'\} : b' \in W_3\} \cup \{\, \{a', b'\} : b' \in W_3\}\,.$$

Formal definition: $x \sim p \wedge x \not\subseteq W_1 \cup W_2$, the latter being expressed by (7.10).

The proof is complete. □

Finally, we show that testing membership in each of the graph classes in the hierarchy (7.7) is P-hard.

Theorem 7.11. *The recognition problem of each of the classes in the hierarchy (7.7) is* P*-hard under uniform* AC^0 *many-one reductions.*

Proof. We show a uniform AC^0 many-one reduction from the *monotone circuit value problem (MCVP)*, whose P-completeness is established by Goldschlager (1977). An instance of MCVP consists of a monotone Boolean circuit C with AND and OR gates and constant input gates, and one has to decide if C evaluates to 1. Given such a circuit C, we construct a graph G as follows:

- For each gate g_k of C, G contains a pair $P_k = \{a_k, b_k\}$ of vertices.
- If g_k is a constant input gate with value 1, then a_k and b_k get different colors (i.e., they form singleton color classes); otherwise, a_k and b_k both get the same color (i.e., they form a color class of size 2).
- For each AND gate g_k with input gates g_i and g_j, G additionally contains a color class F_k of size 4 that together with the two input pairs P_i and P_j as well as the output pair P_k forms a CFI(P_i, P_j, P_k)-gadget; see Figure 7.1.

- For each OR gate g_k with input gates g_i and g_j, G additionally contains two color classes F_{ik} and F_{jk} of size four, and four color classes P_i', P_i'', P_j', P_j'' of size 2. The color classes P_i', P_i'', P_k and F_{ik} form a $\mathrm{CFI}(P_i', P_i'', P_k)$-gadget and each of the pairs P_i' and P_i'' is linked to P_i by two parallel edges. Henceforth, we denote this gadget by $\mathrm{Imp}(P_i, P_k)$; see Figure 7.1. Likewise, the color classes P_j', P_j'' and F_{jk} are used to form an $\mathrm{Imp}(P_j, P_k)$-gadget.

A straightforward induction on the height of the AND and OR gates in C shows that CR on input G refines a color class P_k if and only if the corresponding gate g_k outputs value 1. This follows from the following observations.

- If g_k is an AND gate with input gates g_i and g_j, then the vertices in P_k get different C^{r+2} colors if and only if the vertices in P_i as well as the vertices in P_j have different C^r colors.

- If g_k is an OR gate with input gates g_i and g_j, then the vertices in P_k get different C^{r+3} colors if and only if either the vertices in P_i or the vertices in P_j have different C^r colors.

Now, let G' be the graph that results from G by connecting the vertex pair P_l corresponding to the output gate g_l by two parallel edges with each pair P_k corresponding to a constant 0 input gate g_k. Then, C evaluates to 1 if and only if G' is discrete (i.e., CR on input G' individualizes all vertices of G').

Moreover, if we connect the output pair P_l via an additional $\mathrm{Imp}(P_l, P_{l+1})$-gadget to a new vertex pair P_{l+1}, then the resulting graph G'' is not even refinable if C evaluates to 0. The reason is that no automorphism of G'' flips the pair P_{l+1}, but CR only refines the color class P_{l+1} if C evaluates to 1.

Hence, the mapping $C \mapsto G''$ simultaneously reduces MCVP to each of the graph classes in the hierarchy (7.7). □

We observe that the graph G'' used in the proof of the hardness results can be easily replaced by an uncolored graph. In fact, the vertex colors can be substituted by suitable graph gadgets in such a way that the automorphism group as well as the stable partition remain essentially unchanged (up to the addition of several singleton cells). Hence, the hardness results are also valid for the restricted versions of the classes in the hierarchy (7.7) where we consider only uncolored graphs.

8. Concluding comments and questions

Our Theorem 4.3 (see also Corollary 5.5) gives an efficiently checkable criterion for a graph G that ensures the correctness of the CR algorithm on (G, H) for all graphs H. Theorem 6.2 implies that Tinhofer's approach to isomorphism testing based on the compactness concept has strictly larger potential of applicability. The most important question, that still remains open, is whether the applicability range of this approach admits an efficient characterization (like in the case of CR). In other terms, what is the complexity of recognizing compact graphs? We proved in Theorem 7.11 that this problem is P-hard. The only known complexity upper bound, noted by Tinhofer (1991), is coNP, because testing if every vertex of a given polytope is integral is in coNP.

Another intriguing problem is the complexity of recognizing refinable graphs. Using an AND function for Graph Isomorphism (see, e.g., Köbler *et al.* 1993), it is easy to show that this recognition problem is polynomial-time many-one reducible to Graph Isomorphism. On the other hand, recognition of Refinable is at least as hard as recognition of vertex-transitive graphs (which, in turn, is at least as hard as testing isomorphism of two vertex-transitive graphs).

Applicability of Tinhofer's isomorphism algorithm described in the beginning of Section 7 seems to be a subtle issue. Recall that it works correctly on an input pair (G, H) whenever G is compact (Tinhofer 1991). By our Theorem 7.6, the applicability range of this algorithm is even larger. It would be interesting to analyze the correctness of the algorithm on various classes of vertex-transitive graphs, for example, on classes of Cayley graphs. Note that compactness of Cayley graphs is studied by Schreck & Tinhofer (1988) and Wang & Li (2005).

The CR procedure can be regarded as the one-dimensional version of the more general k-dimensional Weisfeiler-Leman (k-WL) algorithm (Babai 1995; Cai *et al.* 1992; Weisfeiler & Leman 1968). It would, therefore, be natural to try to obtain k-dimensional analogs of our results. More specifically, let us fix the parameter $k \geq 2$ and consider the class of graphs distinguishable from every non-isomorphic graph by the k-WL algorithm. Equivalently, this is the class of graphs definable in $(k+1)$-variable first-order logic with counting quantifiers. Are graphs in this class recognizable in polynomial time? Note that an affirmative answer for $k = 2$ would imply a possibility to efficiently decide if a set of parameters determines a strongly regular graph uniquely up to isomorphism (as 2-WL fails to distinguish non-isomorphic strongly regular graphs with the same parameters). Note also that, as it is shown by Kiefer *et al.* (2015), the 2-dimensional analog of the inclusion $\mathsf{Amenable} \subseteq \mathsf{Refinable}$ is false.

Acknowledgements

The first three authors were supported by the Alexander von Humboldt Foundation in its research group linkage program. The second author was supported by DFG grant KO 1053/7-2. The fourth author was supported by DFG grant VE 652/1-2.

The results of this paper were presented in the 20th International Symposium on Fundamentals of Computation Theory (Arvind *et al.* 2015a) and the 40th International Symposium on Mathematical Foundations of Computer Science (Arvind *et al.* 2015b).

We are thankful to Mikhail Klin for providing us with useful references and to Frank Fuhlbrück and Sebastian Kuhnert for numerous helpful discussions. We also thank the anonymous referees for useful comments on the paper.

A. The Petersen graph is in Godsil

It is well known that the Petersen graph, denoted by P, is isomorphic to the Kneser graph $K(5, 2)$. The *Kneser graph* $K(n, k)$ has the k-element subsets of $[n] = \{1, \ldots, n\}$ as vertices, and two vertices are adjacent if they are disjoint. An important fact about $K(5, 2)$ is that its automorphism group is isomorphic to the symmetric group S_5 acting on the set $\{1, \ldots, 5\}$. In fact, automorphisms of $K(5, 2)$ can be realized by extending the action of some permutation $\pi \in S_5$ to the vertex set of $K(5, 2)$ (Whitney 1932).

First, we state some useful facts about the Petersen graph.

PROPOSITION A.1. *The Petersen graph has the following properties:*

(i) There are no cycles of length 3, 4, and 7.

(ii) There are no independent sets of size greater than 4.

(iii) Adjacent vertices have no common neighbors, and each pair of non-adjacent vertices has a unique common neighbor.

We will need some definitions regarding partitions of the vertex set of a graph $G = (V, E)$. Given a partition $\Sigma = \{S_1, \ldots, S_k\}$ of V, we refer to the sets $S_1, \ldots, S_k$ as the *cells* of Σ. If the size of a cell is k, we call it a *k-cell*. Two cells S and S' are said to be *compatible* if the induced bipartite graph $P[S, S']$ is biregular (it can be empty). Otherwise, we say they are *incompatible*. Recall that a cell S of an equitable partition induces a regular graph $G[S]$. Moreover, in that case, all pairs of cells S, S' are compatible and the number of edges in the biregular graph $G[S, S']$ is a common multiple of $|S|$ and $|S'|$.

Now, we are ready to prove the following theorem.

THEOREM A.2. *The Petersen graph P is a Godsil graph.*

PROOF. To prove the theorem, we will enumerate all equitable partitions of P. For each such partition Σ, we describe a subgroup of $Aut(P)$ such that its orbit partition coincides with Σ. We represent the vertices of P by the two-element subsets of the set $\Omega = \{a, b, c, d, e\}$, where two vertices are adjacent if they are disjoint. This representation allows us to describe a subgroup of $Aut(P)$ as a subgroup of the permutation group S_Ω on Ω.

The two trivial partitions of $V(P)$ into one set and into ten singleton sets are clearly orbit partitions, since the Petersen graph is vertex-transitive. For our case analysis, we classify the remaining non-trivial equitable partitions of P by the minimum size δ of the cells in the partition. Clearly, $\delta \leq 5$. In the following claims, we show for each $k \in \{1, 2, 3, 4, 5\}$ that equitable partitions of P with $\delta = k$ are all orbit partitions of P.

Claim A.3. *P does not have equitable partitions with $\delta = 3$.*

Proof. Suppose that there is an equitable partition Σ with $\delta = 3$ and let S be a 3-cell in it. Then, Σ either has the form $\Sigma = \{S, T\}$, where $|T| = 7$, or the form $\Sigma = \{S, U, V\}$ where $|U| = 3$ and $|V| = 4$. The first case is ruled out since $P[T]$ can never be regular (P has neither independent sets of size 7 nor cycles of size 7). Suppose the second case is possible. Then, $P[S]$ and $P[U]$ must be empty (since P has no triangles). Furthermore, the bipartite graphs $P[S, V]$ and $P[U, V]$ must be both biregular. The graph $P[S, V]$ (likewise, $P[U, V]$) is empty or it has 12 edges. It is not possible that $P[S, V]$ has 12 edges because then $P[V]$ has only 3 edges and cannot be regular. If both $P[S, V]$ and $P[U, V]$ are empty then V is disconnected from the rest of the graph, which is a contradiction. □

Claim A.4. *All equitable partitions of P with $\delta = 4$ are orbit partitions.*

Proof. We first show that every equitable partition Σ with $\delta = 4$ has one 4-cell S and one 6-cell T, where $P[S]$ is empty and $P[T]$ is a 3-matching (a matching with 3 edges). Clearly, Σ must be of the form $\{S, T\}$, where $|S| = 4$ and $|T| = 6$. Moreover, $P[S]$ must be empty (0-regular) or 2-matching (1-regular) since it cannot be a 4-cycle (2-regular). In fact, the case of 2-matching can also be ruled out by counting the number of edges as follows. For S and T to be compatible, there must be 12 edges in the graph $P[S, T]$. Then, there is exactly one edge left in the induced graph $P[T]$ which is impossible. Therefore, $P[S]$ must be empty. This also implies that the graph $P[S, T]$ has $4 \times 3 = 12$ edges. Hence, $P[T]$ must be a 3-matching.

Now observe that an independent set S of size 4 in P must be of the kind $S = \{ab, ac, ad, ae\}$ (up to automorphisms), implying that $T = \{bc, bd, be, cd, ce, de\}$. The partition $\{S, T\}$ can be easily verified to be equitable and that it is the orbit partition of the subgroup $S_{\{b,c,d,e\}}$. $\square$

Claim A.5. *All equitable partitions of P with $\delta = 5$ are orbit partitions.*

Proof. In this case, Σ must have the form $\Sigma = \{S, T\}$ where $|S| = |T| = 5$. Moreover, since P does not have independent sets of size 5, $P[S]$ and $P[T]$ must be 5-cycles. Clearly, such partitions exist, and such a partition has a matching between sets S and T.

It remains to show that $\Sigma = \{S, T\}$ is indeed an orbit partition of some subgroup of $Aut(P)$. Denote the 5-cycle in S by 1-2-3-4-5. Let $1'$ be the matching partner of 1 in T and so on. Now, $1'$ and $2'$ cannot be adjacent, else there is a 4-cycle in P. The unique common neighbor of $1'$ and $2'$ must be $4'$; otherwise, it is easy to verify that we will have a 4-cycle in P. The partners $3'$ and $5'$ can also be uniquely determined in T. The permutation $\pi = (12345)(1'2'3'4'5')$ can be verified to be an automorphism of P and the orbits of the subgroup generated by π are precisely $\{S, T\}$. $\square$

Claim A.6. *All equitable partitions of P with $\delta = 2$ are orbit partitions.*

Proof. Let Σ be an equitable partition of P with $\delta = 2$ and let $S = \{u, v\}$ be a 2-cell in it. We first show that uv must be an edge. This holds because a pair of non-adjacent vertices has a unique common neighbor x. The cell containing x can only be a singleton set, which contradicts $\delta = 2$.

Next, we show that the neighborhood $N(S) = \bigcup_{x \in S} N(x) \setminus S$ of S is also a cell of Σ (see Figure A.1). Since uv is an edge, there are no common neighbors of u and v. Therefore, $|N(S)| = 4$. Moreover, $N(S)$ is an independent set since edges among vertices in $N(S)$ can be used to construct a cycle of length 3 or 4 passing through the edge uv. This is not possible by Proposition A.1. Now, let $R = V(P) \setminus (S \cup N(S))$ be the set of the four remaining vertices as shown in Figure A.1. Observe that no cell can contain vertices

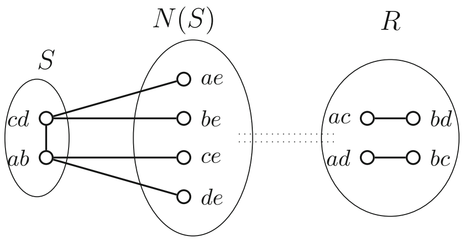


Figure A.1: The case $\delta = 2$

from both $N(S)$ and R, since then it would be incompatible with S. Since $N(S)$ is an independent set, there cannot be a 2-cell inside $N(S)$. Clearly, there cannot be 1-cells and hence 3-cells inside $N(S)$. Therefore, $N(S)$ must indeed be a cell.

By accounting for edges of S and $N(S)$, it is easy to verify that R has exactly two edges, and hence, $P[R]$ must be a 2-matching. Since $\delta = 2$, R does not contain 1-cells and hence does not contain 3-cells. This leaves us with only two cases.

Case 1: R is a cell. We characterize all such partitions by naming a typical case. W.l.o.g, let $S = \{ab, cd\}$ since S is an edge. Then, $N(S)$ must be $\{ae, be, ce, de\}$ and R must be $\{ac, ad, bc, bd\}$. The partition $\{ab, cd\}, \{ae, be, ce, de\}, \{ac, ad, bc, bd\}$ can be easily verified to be equitable. Moreover, it is easy to check that it is the orbit partition of the subgroup of all permutations in S_Ω which preserve the Ω-partition $\{ab\}, \{cd\}, \{e\}$. This is also the subgroup generated by the automorphisms $(ab), (cd), (ac)(bd)$.

Case 2: Σ partitions R in two sets A and B where $|A| = |B| = 2$. Since each 2-cell has to be an edge (see above), the sets A and B must be $\{ac, bd\}$ and $\{bc, ad\}$. The partition $\{ab, cd\}, \{ae, be, ce, de\}, \{ac, bd\}, \{ad, bc\}$ can be easily verified to be equitable. It is easy to check that this is the orbit partition of the subgroup of all permutations in S_Ω which preserve the Ω-partition $\{ab\}, \{cd\}, \{e\}$ and additionally, stabilize the sets $\{ac, bd\}$ and $\{ad, bc\}$. This is also the subgroup generated by the automorphisms $(ac)(bd)$, $(ad)(bc)$, and $(ab)(cd)$.

The proof of Claim A.6 is complete. □

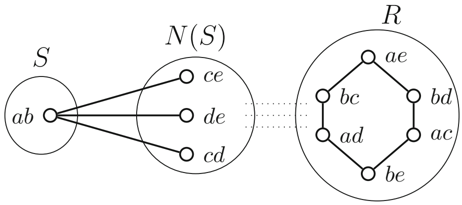


Figure A.2: The case $\delta = 1$

Claim A.7. *All equitable partitions of P with $\delta = 1$ are orbit partitions.*

Proof. Let S be a singleton set in such an equitable partition. Similar to a previous argument, a cell cannot have vertices from both $N(S)$ and $V \backslash N(S)$. Therefore, every equitable partition refines the partition $S, N(S), R$ (see Figure A.2). Observe that $N(S)$ must be an independent set (otherwise, there is a 3-cycle). Moreover, if we assume that $S = \{ab\}$, $N(S)$ must be $\{ce, de, cd\}$, and therefore, $R = \{ae, be, ac, bc, ad, bd\}$ forms a 6-cycle, as shown in the figure. We proceed by further classifying equitable partitions on the basis of the partition induced by them inside $N(S)$. Since $|N(S)| = 3$, we have three possible cases. Either $N(S)$ is a cell, it contains three 1-cells or it contains one singleton and one 2-cell.

Case 1: $N(S)$ is a cell. We further classify the equitable partitions in this case on the basis of the partition induced on the set R. First, we examine the possible cells X in R which are compatible with $N(S)$. X cannot be of size 1 or 2; otherwise, $P[N(S), X]$ has at most two edges. Also, X cannot be of size 4 or 5 since this would imply a cell of size 1 or 2 in R. Therefore, either R is a cell or there are two 3-cells in R.

(a) R is a cell. The partition $\{ab\}$, $\{de, cd, ce\}$, $\{ac, ad, ae, bc, bd, be\}$ can be verified to be an equitable partition. Moreover, it is easy to check that it is the orbit partition of the subgroup $S_{\{c,d,e\}} \times S_{\{a,b\}}$.

(b) The partition induced on R is of the form $\{A, B\}$, where $|A| = |B| = 3$. Because of regularity, the only possible 3-cells in R are the independent sets $\{ad, ac, ae\}$ and $\{bc, bd, be\}$. The partition

$\{ab\}$, $\{de, cd, ce\}$, $\{ad, ac, ae\}$, $\{bc, bd, be\}$ is clearly equitable. Moreover, it is easy to check that this partition is the orbit partition of the subgroup $S_{\{c,d,e\}}$.

Case 2: $N(S)$ contains three 1-cells. Again, we classify the equitable partitions on the basis of the partition induced on the set R. We can check that a cell of size more than two in R will have at least one edge to some singleton in $N(S)$ and will be incompatible with that singleton. Therefore, cells in R must have size at most 2. Moreover, a 2-cell must be of the form $\{ax, bx\}$ for some $x \in \{d, c, e\}$ since all other 2-cells can be seen to be incompatible with some singleton cell in $N(S)$. Finally, it can be seen that every possible 1-cell is incompatible with these three 2-cells. Hence, R must consist of three cells of size 2, namely $\{ad, bd\}$, $\{ac, bc\}$, $\{ae, be\}$. The partition $\{ab\}$, $\{cd\}$, $\{ce\}$, $\{de\}$, $\{ad, bd\}$, $\{ac, bc\}$, $\{ae, be\}$ can be easily seen to be equitable. Moreover, it is easy to check that it is the orbit partition of the subgroup $S_{\{a,b\}}$.

Case 3: $N(S)$ contains a 2-cell $U = \{ce, de\}$ and a 1-cell $V = \{cd\}$. Again, we need to classify the equitable partitions on the basis of the partition induced on the set R. First, we examine the possible cells X in R which are compatible with U and V. Clearly, X cannot be a 5-cell since $P[X]$ cannot be regular. It cannot be a 3-cell as well since the two candidate 3-cells are the independent sets $\{ad, ac, ae\}$ and $\{bc, bd, be\}$. Neither of them can be compatible with the singleton set V. Also, R cannot be a cell since it is incompatible with the singleton set V. Moreover, the only possible 4-cell is the neighborhood of the set U, i.e., $\{ac, bd, ad, bc\}$. All other 4-cells are incompatible with U. Overall, we have no cells of size 3, 5, or 6 in R. Therefore, we have only the following four remaining subcases.

(a) R consists of one 4-cell and two 1-cells. This case is not possible since a 1-cell cannot be compatible with a 4-cell.

(b) R consists of one 4-cell and one 2-cell. The cells are $\{ac, bd, ad, bc\}$ and $\{ae, be\}$. The partition $\{ab\}$, $\{cd\}$, $\{ce, de\}$, $\{ae, be\}$, $\{ac, bd, ad, bc\}$ can be verified to be an equitable partition. Moreover, it is easy to check that it is the orbit partition of the subgroup $S_{\{a,b\}} \times S_{\{c,d\}}$

(c) R consists of three 2-cells. First, ae and be must be in the same 2-cell; otherwise, the cell containing one of them would be incompatible with V. For the remaining vertices ac, ad, bc, bd, we can pair them up in three ways: (i) ac, ad and bc, bd, (ii) ac, bc and ad, bd, or (iii) ac, bd and ad, bc The first case is not possible since $\{ae, be\}$ and $\{ac, ad\}$ are not compatible. The second case is not possible because $\{ac, bc\}$ and $U = \{ce, de\}$ are not compatible. The third case gives an equitable partition $\{ab\}$, $\{cd\}$, $\{ce, de\}$, $\{ae, be\}$, $\{ac, bd\}$, $\{ad, bc\}$. Moreover, it is easy to check that it is the orbit partition of the subgroup generated by $(ab)(cd)$.

(d) R consists of a bunch of 1-cells and 2-cells. Clearly, the vertices ac, ad, bc, bd cannot form a singleton cell, since such a 1-cell will not be compatible with U. Therefore, $\{ae\}$ and $\{be\}$ are the only possible singleton cells. Neither of them can pair up with one of ac, ad, bc, bd since that cell would be incompatible with V. Therefore, they are forced to be singleton cells. It remains to partition ac, ad, bc, bd into two 2-cells. The vertex ac cannot be paired up with bd or bc since it will be incompatible with be. Therefore, the only possible case is to have 2-cells $\{ac, ad\}$ and $\{bc, bd\}$. The partition $\{ab\}$, $\{cd\}$, $\{ce, de\}$, $\{ae\}$, $\{be\}$, $\{ac, ad\}$, $\{bc, bd\}$ can be verified to be equitable. Moreover, it is easy to check that it is the orbit partition of the subgroup $S_{\{c,d\}}$. (This case is identical to Case 2).

The proof of Claim A.7 is complete. □

Theorem A.2 is proved. □

References

DANA ANGLUIN (1980). Local and global properties in networks of processors. In *Proceedings of the 12th Annual ACM Symposium on Theory of Computing*, 82–93. ACM.

V. ARVIND, JOHANNES KÖBLER, GAURAV RATTAN & OLEG VERBITSKY (2015a). On the power of color refinement. In *Proceedings of the*

20th International Symposium on Fundamentals of Computation Theory (FCT), volume 9210 of *Lecture Notes in Computer Science*, 339–350. Springer.

V. Arvind, Johannes Köbler, Gaurav Rattan & Oleg Verbitsky (2015b). On Tinhofer's linear programming approach to isomorphism testing. In *Proceedings of the 40th International Symposium on Mathematical Foundations of Computer Science (MFCS)*, volume 9235 of *Lecture Notes in Computer Science*, 26–37. Springer.

Albert Atserias & Elitza N. Maneva (2013). Sherali-Adams relaxations and indistinguishability in counting logics. *SIAM Journal on Computing* **42**(1), 112–137.

László Babai (1995). Automorphism groups, isomorphism, reconstruction. In *Handbook of Combinatorics*, chapter 27, 1447–1540. Elsevier.

László Babai, Paul Erdős & Stanley M. Selkow (1980). Random graph isomorphism. *SIAM Journal on Computing* **9**(3), 628–635.

László Babai & Ludek Kučera (1979). Canonical labelling of graphs in linear average time. In *Proceedings of the 20th Annual Symposium on Foundations of Computer Science*, 39–46.

Christoph Berkholz, Paul Bonsma & Martin Grohe (2013). Tight lower and upper bounds for the complexity of canonical colour refinement. In *Proceedings of 21st Annual European Symposium on Algorithms (ESA)*, volume 8125 of *Lecture Notes in Computer Science*, 145–156. Springer.

Alessandro Borri, Tiziana Calamoneri & Rossella Petreschi (2011). Recognition of unigraphs through superposition of graphs. *Journal of Graph Algorithms and Applications* **15**(3), 323–343.

Richard A. Brualdi (1988). Some applications of doubly stochastic matrices. *Linear Algebra and its Applications* **107**, 77–100.

Richard A. Brualdi (2006). *Combinatorial matrix classes.* Cambridge University Press.

Robert G. Busacker & Thomas L. Saaty (1965). *Finite graphs and networks: an introduction with applications.* International Series in Pure and Applied Mathematics. McGraw-Hill Book Company, New York etc.

Jin-Yi Cai, Martin Fürer & Neil Immerman (1992). An optimal lower bound on the number of variables for graph identification. *Combinatorica* **12**(4), 389–410.

A. Cardon & Maxime Crochemore (1982). Partitioning a graph in $O(|A| \log_2 |V|)$. *Theoretical Computer Science* **19**, 85–98.

Ada Chan & Chris D. Godsil (1997). Symmetry and eigenvectors. In *Graph symmetry*, volume 497, 75–106. Kluwer Acad. Publ., Dordrecht.

Sergei Evdokimov, Marek Karpinski & Ilia N. Ponomarenko (1999). Compact cellular algebras and permutation groups. *Discrete Mathematics* **197–198**, 247–267.

Sergei Evdokimov, Ilia N. Ponomarenko & Gottfried Tinhofer (2000). Forestal algebras and algebraic forests (on a new class of weakly compact graphs). *Discrete Mathematics* **225**(1–3), 149–172.

Chris D. Godsil (1997). Compact graphs and equitable partitions. *Linear Algebra and its Applications* **255**(1–3), 259 – 266.

Leslie M. Goldschlager (1977). The monotone and planar circuit value problems are log space complete for P. *SIGACT News* **9**, 25–29.

Martin Grohe (1999). Equivalence in finite-variable logics is complete for polynomial time. *Combinatorica* **19**(4), 507–532.

Martin Grohe, Kristian Kersting, Martin Mladenov & Erkal Selman (2014). Dimension reduction via colour refinement. In *Proceeding of 22th Annual European Symposium on Algorithms (ESA)*, volume 8737 of *Lecture Notes in Computer Science*, 505–516. Springer.

Martin Grohe & Martin Otto (2015). Pebble games and linear equations. *The Journal of Symbolic Logic* **80**(3), 797–844.

Neil Immerman & Eric Lander (1990). Describing graphs: A first-order approach to graph canonization. In *Complexity Theory Retrospective*, 59–81. Springer.

Robert H. Johnson (1975). Simple separable graphs. *Pacific Journal of Mathematics* **56**, 143–158.

Kristian Kersting, Martin Mladenov, Roman Garnett & Martin Grohe (2014). Power iterated color refinement. In *Proceedings of the Twenty-Eighth AAAI Conference on Artificial Intelligence*, 1904–1910. AAAI Press.

Sandra Kiefer, Pascal Schweitzer & Erkal Selman (2015). Graphs identified by logics with counting. In *Proceedings of the 40th International Symposium on Mathematical Foundations of Computer Science (MFCS)*, volume 9234 of *Lecture Notes in Computer Science*, 319–330. Springer. A full version is available as an e-print http://arxiv.org/abs/1503.08792.

Johannes Köbler, Uwe Schöning & Jacobo Torán (1993). *The graph isomorphism problem: its structural complexity.* Boston, MA: Birkhäuser.

Michael Koren (1976). Pairs of sequences with a unique realization by bipartite graphs. *Journal of Combinatorial Theory, Series B* **21**(3), 224–234.

Andreas Krebs & Oleg Verbitsky (2015). Universal covers, color refinement, and two-variable logic with counting quantifiers: Lower bounds for the depth. In *Proceedings of the 30-th ACM/IEEE Annual Symposium on Logic in Computer Science (LICS)*, 689–700. IEEE Computer Society.

Frank Thomson Leighton (1982). Finite common coverings of graphs. *Journal of Combinatorial Theory, Series B* **33**(3), 231–238.

Peter N. Malkin (2014). Sherali-Adams relaxations of graph isomorphism polytopes. *Discrete Optimization* **12**, 73–97.

Motakuri V. Ramana, Edward R. Scheinerman & Daniel Ullman (1994). Fractional isomorphism of graphs. *Discrete Mathematics* **132**(1-3), 247–265.

Helmut Schreck & Gottfried Tinhofer (1988). A note on certain subpolytopes of the assignment polytope associated with circulant graphs. *Linear Algebra and its Applications* **111**, 125–134.

Nino Shervashidze, Pascal Schweitzer, Erik Jan van Leeuwen, Kurt Mehlhorn & Karsten M. Borgwardt (2011).

Weisfeiler-Lehman graph kernels. *Journal of Machine Learning Research* **12**, 2539–2561.

Gottfried Tinhofer (1986). Graph isomorphism and theorems of Birkhoff type. *Computing* **36**, 285–300.

Gottfried Tinhofer (1989). Strong tree-cographs are Birkhoff graphs. *Discrete Applied Mathematics* **22**(3), 275–288.

Gottfried Tinhofer (1991). A note on compact graphs. *Discrete Applied Mathematics* **30**(2–3), 253–264.

Gottfried Tinhofer & Mikhail Klin (1999). Algebraic combinatorics in mathematical chemistry. Methods and algorithms III. Graph invariants and stabilization methods. Technical Report TUM-M9902, Technische Universität München.

Regina Tyshkevich (2000). Decomposition of graphical sequences and unigraphs. *Discrete Mathematics* **220**(1–3), 201–238.

Gabriel Valiente (2002). *Algorithms on trees and graphs.* Springer.

Ping Wang & Jiong Sheng Li (2005). On compact graphs. *Acta Mathematica Sinica* **21**(5), 1087–1092.

Boris Yu. Weisfeiler & Andrei A. Leman (1968). A reduction of a graph to a canonical form and an algebra arising during this reduction. *Nauchno-Technicheskaya Informatsia, Seriya 22* **9**, 12–16. In Russian.

Hassler Whitney (1932). Congruent graphs and connectivity of graphs. *American Journal of Mathematics* **54**, 150–168.

Matan Ziv-Av (2013). Results of computer algebra calculations for triangle free strongly regular graphs. *E-print* http://www.math.bgu.ac.il/~zivav/math/eqpart.pdf.

Manuscript received 2 October 2015

V. Arvind
The Institute of Mathematical Sciences
Chennai 600 113, India
arvind@imsc.res.in

Johannes Köbler
Institut für Informatik
Humboldt Universität zu Berlin
Unter den Linden 6
Berlin 10099, Germany
koebler@informatik.hu-berlin.de

Gaurav Rattan
The Institute of Mathematical Sciences
Chennai 600 113, India
grattan@imsc.res.in

Oleg Verbitsky
Institut für Informatik
Humboldt Universität zu Berlin
Unter den Linden 6
Berlin 10099, Germany
verbitsk@informatik.hu-berlin.de

Oleg Verbitsky: On leave from the Institute for Applied Problems of Mechanics and Mathematics, Lviv, Ukraine.