Groups, Graphs, Algorithms: The Graph Isomorphism Problem*

László Babai
University of Chicago

February 18, 2018

Abstract

Graph Isomorphism (GI) is one of a small number of natural algorithmic problems with unsettled complexity status in the P / NP theory: not expected to be NP-complete, yet not known to be solvable in polynomial time.

Arguably, the GI problem boils down to filling the gap between *symmetry* and *regularity*, the former being defined in terms of automorphisms, the latter in terms of equations satisfied by numerical parameters.

Recent progress on the complexity of GI relies on a combination of the *asymptotic theory of permutation groups* and asymptotic properties of highly regular combinatorial structures called *coherent configurations*. Group theory provides the tools to infer either global symmetry or global irregularity from local information, eliminating the symmetry/regularity gap in the relevant scenario; the resulting global structure is the subject of combinatorial analysis. These structural studies are melded in a *divide-and-conquer* algorithmic framework pioneered in the GI context by Eugene M. Luks (1980).

1 Introduction

We shall consider finite structures only; so the terms "graph" and "group" will refer to finite graphs and groups, respectively.

*AMS Classification: Primary: 68Q25 Secondary: 20B05 05B30 05C68 05C60 05E30

1.1 Graphs, isomorphism, NP-intermediate status

A *graph* is a set (the set of *vertices*) endowed with an irreflexive, symmetric binary relation called *adjacency*. Isomorphisms are adjacency-preseving bijections between the sets of vertices. The Graph Isomorphism (GI) problem asks to determine whether two given graphs are isomorphic.

It is known that graphs are *universal* among explicit finite structures in the sense that the isomorphism problem for explicit structures can be reduced in polynomial time to GI (in the sense of *Karp-reductions*[1]) [**?**, **?**]. This makes GI a natural algorithmic problem. It is a *polynomial-time verifiable* problem: a candidate isomorphism is easily verified. This puts GI in the complexity class NP. Over time, increasingly strong conjectural evidence has been found that GI is not NP-*complete*, yet no polynomial-time algorithm is known to solve GI. This puts GI among the small number of natural NP-problems of potentially *intermediate complexity* (neither in P, nor NP-complete). (Another such problem is that of *factoring integers*, cf. Sec. **??**.) The interest in this status of GI was recognized at the dawn of the P / NP theory [**?**, **?**].

1.2 Brief history of the GI problem

Combinatorial heuristics such as individualization and refinement (I/R) (see Sec. **??**) have been used for the longest time to reduce the GI search space. It was shown that the "naive refinement" algorithm solves GI for almost all graphs in linear time [**?**, **?**]. Efficient algorithms were found for special classes such as planar graphs [**?**, **?**]. These algorithms exploited the combinatorial structure of the graphs concerned. However, combinatorial refinement methods alone cannot succeed in less than exponential time for the general GI problem, as shown in a seminal 1992 paper by Cai, Furer, and Immerman [**?**].

It has long been known that GI is equivalent to determining whether two vertices of a given graph belong to the same orbit of the automorphism group. Refinement procedures have been used to distinguish vertices, trying to refute *symmetry* by discovering *irregularity*. While this gives a first indication of the critical role of the gap between symmetry and regularity to GI, the CFI result shows the futility of trying to close this gap using combinatorial refinement heuristics alone. We use group theory to close a gap of this nature under particular circumstances (see Theorem **??** and

[1]For basic concepts of complexity theory we refer to [**?**].

the paragraph preceding it). The relevant new group theoretic result, the "Unaffected Stabilizers Lemma," is stated in Theorem **??**.

Elements of group theory were first introduced into the design of GI algorithms in 1979 [**?**]. The *tower of groups* method described in that paper produced the following results. A *vertex-colored* graph has a "color" assigned to each vertex; isomorphisms preserve the colors by definition. The *multiplicity* of a color is the number of vertices of that color. The *adjacency matrix* of a graph with n vertices is the $n \times n$ $(0,1)$-matrix whose (i,j)-entry is 1 if vertex i is adjacent to vertex j, and 0 otherwise. By the *eigenvalues of a graph* we mean the eigenvalues of its adjacency matrix.

Theorem 1.1. (a) [**?**, **?**] *Isomorphism of vertex-colored graphs of bounded color multiplicities can be tested in polynomial time.* (b) [**?**] *Isomorphism of graphs with bounded eigenvalue multiplicities can be tested in polynomial time.*

It turns out that the CFI pairs of graphs, i.e., the pairs of graphs shown in [**?**] to be hard to separate by combinatorial refinement, can be viewed as vertex-colored graphs with color multiplicity 4. This shows that elementary group theory (hardly more than the concept of cosets was used) was already capable of overcoming exponential barriers to combinatorial refinement methods. Modern extensions of the CFI result show that GI is hard for several more general refutation systems (see Sec. **??**), putting GI in a somewhat paradoxical position in complexity theory (cf. Sec. **??**).

In-depth use of group theory in the design of GI algorithms arrived with Luks's groundbreaking 1980 paper [**?**]. We state the main result of that paper. Adjacent vertices of a graph are called *neighbors*; the *degree* of a vertex is the number of its neighbors.

Theorem 1.2 (Luks, 1980)**.** *Isomorphism of graphs of bounded degree can be tested in polynomial time.*

Luks's group theoretic method, combined with a combinatorial refinement result by Zemlyachenko [**?**], have lead to the *moderately exponential* complexity bound of

$$\exp(O(\sqrt{n \log n})), \tag{1}$$

where n denotes the number of vertices (Luks, 1983, cf. [**?**, **?**]). In spite of intermittent progress on important special cases, notably for *strongly regular graphs* [**?**, **?**, **?**, **?**] and for *primitive coherent configurations* [**?**], Luks's bound (**??**) for the general case had not been improved until this author's recent annoucement [?] of a quasipolynomial-time algorithm. A *quasipolynomial* function is a function of the form $\exp(p(\log n))$ for some polynomial p. A quasipolynomial time bound is a bound of this form where n is the bit-length of the input; but if we take n to be the number of vertices of an input graph, the form of the bound will not be affected.

Theorem 1.3 (B 2015)**.** *Isomorphism of graphs can be tested in quasipolynomial time.*

In this paper we outline the main components of this result. For an introduction to the algorithmic theory of permutation groups we refer to the monograph [?].

Disclaimer. I should emphasize that the results discussed in this paper address the mathematical problem of the *asymptotic worst-case complexity* of GI and have little relevance to practical computation. A suite of remarkably efficient GI packages is available for practical GI testing; McKay and Piperno [?] give a detailed comparison of methods and performance. These algorithms employ ingenious shortcuts to backtrack search. While the worst-case performance of these heuristics seems to be exponential, this is increasingly difficult to demonstrate, cf. [?, ?, ?].

2 The string isomorphism problem

We now define a generalization of the GI problem, introduced by Luks [?].

Let Ω be a finite set; $\mathrm{Sym}(\Omega)$ denotes the symmetric group acting on Ω. Let Σ be finite *alphabet*. An Ω-string (or just "string") over Σ is a function $\mathfrak{x} : \Omega \to \Sigma$. There is a natural action $\mathfrak{x} \mapsto \mathfrak{x}^\sigma$ of $\mathrm{Sym}(\Omega)$ on the set Σ^Ω of strings ($\sigma \in \mathrm{Sym}(\Omega)$, $\mathfrak{x} \in \Sigma^\Omega$). We say that $\sigma \in \mathrm{Sym}(\Omega)$ is a *G-isomorphism* between the strings $\mathfrak{x}$ and $\mathfrak{y}$ if $\sigma \in G$ and $\mathfrak{x}^\sigma = \mathfrak{y}$. The strings $\mathfrak{x}$ and $\mathfrak{y}$ are *G-isomorphic*, denoted $\mathfrak{x} \cong_G \mathfrak{y}$, if such a σ exists. The *String Isomorphism (SI) problem* asks, given G, $\mathfrak{x}$, and $\mathfrak{y}$, does $\mathfrak{x} \cong_G \mathfrak{y}$ hold? We refer to G as the *ambient group*; it is given by a list of generators.

Luks pointed out [?] that GI reduces to SI by encoding each graph X by the characteristic function f_X of its adjacency relation, $f_X : \binom{\Omega}{2} \to \{0,1\}$, where $\binom{\Omega}{2}$ denotes the set of unordered pairs of elements of Ω. So f_X is an $\binom{\Omega}{2}$-string over the alphabet $\{0,1\}$. The pertinent ambient group is $\mathrm{Sym}(\Omega)^{(2)}$, the induced action of $\mathrm{Sym}(\Omega)$ on the set $\binom{\Omega}{2}$. It is easy to see that two graphs are isomorphic if and only if the corresponding $\binom{\Omega}{2}$-strings are $\mathrm{Sym}(\Omega)^{(2)}$-isomorphic. The actual result we shall discuss concerns the complexity of SI [?].

Theorem 2.1 (B 2015)**.** *String isomorphism can be tested in quasipolynomial time.*

Theorem **??** is then a corollary. The previous best bound for SI was $\exp(\widetilde{O}(n^{1/2}))$, where $n = |\Omega|$ is the length of the strings in question [**?**] (cf. [**?**]). (The tilde hides a polylogarithmic factor.)

Luks also observed that several other problems of computational group theory are polynomial-time equivalent to SI (under Karp-reductions), including the coset intersection, double coset membership, and 'centralizer in coset' problems. Given two subgroups G, H of the symmertic group S_n and two elements $\sigma, \pi \in S_n$, the *Coset Intersection* problem asks whether $G\sigma \cap H\pi \neq \emptyset$; the *double coset membership* problem asks whether $\sigma \in G\pi H$, and the *centralizer in coset* problem asks whether there exists an element in the coset $G\sigma$ that commutes with π. As a consequence, these problems, too, can be solved in quasipolynomial time.

The advantage of approaching GI through the SI problem is that SI permits recursion on the ambient group. This was Luks's core idea.

3 Divide-and-Conquer

In the theory of algorithms, the term "Divide-and-Conquer" refers to recursive procedures that reduce an instance of a computational problem to a moderate number of significantly smaller instances. If our input has size n, we shall consider instances of size $\leq 0.9n$ to be "significantly smaller." Let $q(n)$ be the number of such smaller instances to which our input is reduced; we refer to $q(n)$ as the *multiplicative cost* of the reduction. If $f(n)$ denotes the worst-case cost of processing an input of size n, this leads to the following recurrence (ignoring the additive cost of assembling all information from the smaller instances, which will typically not affect the cost estimate).

$$f(n) \leq q(n) f(0.9n) \tag{2}$$

Assuming that $q(n)$ is monotone, this gives the bound $f(n) \leq q(n)^{O(\log n)}$, so if $q(n)$ is quasipolynomially bounded then so is $f(n)$. Therefore our goal will be to significantly reduce the problem size at a quasipolynomial multiplicative cost.

4 Large primitive permutation groups

Not only did Luks point out that GI reduces to SI, but he also showed that (i) the SI problem for groups with restricted structure can be used to solve the GI problem for certain classes of graphs; and that (ii) SI can be solved efficiently under such structural constraints. The issue of relevance here is bounding the order of primitive permutation groups under structural constraints.

A *permutation group* acting on the set Ω (the *permutation domain*) is a subgroup $G \leq \mathrm{Sym}(\Omega)$. (The "$\leq$" sign stands for "subgroup.") The *degree* of G is $|\Omega|$. The set $x^G = \{x^\sigma \mid \sigma \in G\}$ is the *G-orbit* of x; the orbit has *length* $|x^G|$. We say that G is *transitive* if $x^G = \Omega$ for some (and therefore any) $x \in \Omega$. A transitive group $G \leq \mathrm{Sym}(\Omega)$ is *primitive* if $|\Omega| \geq 2$ and there is no nontrivial G-invariant equivalence relation on Ω.

In 1982, Pálfy [**?**] and Wolf [**?**] showed that primitive *solvable* groups of degree n have order $\leq n^c$ where $c \approx 3.243$. It turns out that the critical structural parameter of a group for polynomial bounds on the order of its primitive permutation representations is its "thickness."

Definition 4.1. The *thickness*[2] $\theta(G)$ of a group G is the largest t such that the alternating group A_t is involved in G as a quotient of a subgroup.

The following result characterizes those hereditary classes of groups (classes that are closed under subgroups and quotients) which have only small primitive permutation representations.

Theorem 4.2 (B, Cameron, Pálfy, 1982)**.** *If G is a primitive permutation group of degree n and thickness t then $|G| = n^{O(t)}$.*

This result first appeared in [**?**]; here it is stated with an improved exponent due to Pyber [**?**]. Refined versions were subsequently obtained by Liebeck, Shalev, Maróti; see [**?**, Sec. 3] for a survey of those developments. We note that while the initial motivation for Theorem **??** came from the GI problem, the result also found applications in other areas, such as the theory of profinite groups [**?**].

Luks [**?**] introduced a group theoretic divide-and-conquer technique to attack the SI problem. Luks's method, combined with the above bounds, yields the following.

Corollary 4.3. *The SI problem can be solved in polynomial time if the ambient group is solvable or more generally, if it has bounded thickness.*

Let G be the stabilizer of an edge in the automorphism group of a connected graph in which every vertex has degree $\leq k$. It is easy to see that every composition factor of G is a subgroup of the symmetrc group S_{k-1}.

[2] The term "thickness" was coined in [**?**].

In particular, $\theta(G) \le k-1$ and therefore the SI problem can be solved in polynomial time for such G as the ambient group. This fact is at the heart[3] of the proof of Theorem **??**.

While Theorem **??** is helpful for groups with small thickness, our interest is in the general case. Luks's technique for SI works in quasipolynomial time as long as the primitive groups involved in the ambient group have quasipolynomially bounded orders. In 1981, building on the then expected completion of the classification of the finite simple groups (CFSG), Cameron [**?**] gave a precise characterization of primitive groups of large order. The *socle* of a group is the product of its minimal normal subgroups. It is known that the socle of a primitive permutation group is a direct product of isomorphic simple groups. For a permutation group $T \le \mathrm{Sym}(\Delta)$, the *product action* of the direct power T^k on the Cartesian power Δ^k is the independent action of each copy of T on the corresponding coordinate. Wreath product in addition permutes the coordinates by some group "on the top." For a permutation group $G \le \mathrm{Sym}(\Omega)$ we denote by $G^{(t)}$ the induced action of G on the set $\binom{\Omega}{t}$ of unordered t-tuples of elements of Ω.

Definition 4.4. $G \le S_n$ is a *Cameron group* with parameters $s, t \ge 1$ and $k \ge \max(2t+1, 5)$ if we have $n = \binom{k}{t}^s$, the socle of G is isomorphic to A_k^s and acts as $(A_k^{(t)})^s$ in the product action, and $(A_k^{(t)})^s \le G \le S_k^{(t)} \wr S_s$ (wreath product, product action), moreover the induced action $G \to S_s$ on the direct factors of the socle is transitive.

Theorem 4.5 (Cameron 1981)**.** *For $n \ge 25$, if $G \le S_n$ has order $|G| \ge n^{1+\log_2 n}$ then G is a Cameron group.*

This sharp version of Cameron's theorem [**?**] is due to Maróti [**?**].

5 Luks's method and the bottleneck

In attacking the SI problem, Luks applies a combination of the following two types of recursive operations to the ambient group.

- Descend to a subgroup.
- Process orbits one by one.

[3] Theorem **??** was not available to Luks at the time; he used a further layer of recurrence so a weaker group-theoretic result was sufficient for his analysis [**?**].

Orbit-by-orbit processing leads to ultra-efficient (linear-time) recurrence. Descent to a subgroup $H \leq G$ incurs a heavy penalty, namely, a multiplicative cost of $|G : H|$, so this can only be used to replace the ambient group with a subgroup of small index, and to compensate for the multiplicative cost, such a step needs to lead to significantly reduced problem size. Small primitive groups acting on a minimal system of imprimitivity (system of maximal blocks of a G-invariant equivalence relation) provide such an opportunity; the orbits of the kernel of the action of such a primitive group have length $\leq n/2$, hence orbit-by-orbit processing reduces the problem to significantly smaller instances.

Using Theorem **??** we can identify the bottleneck for Luks's method.

Definition 5.1. We say that a group G has a *giant quotient of degree m* if G has an epimorphism onto S_m or A_m.

Proposition 5.2. *For any constant $C \geq 1$ one can use Luks recurrence for the SI problem to achieve one of the following at a multiplicative cost of $n^{O(\log n)}$.*

(a) Significantly reduce the problem size.

(b) Reduce the ambient group to a transitive group with a giant quotient of degree $\geq C \log n$.

Our work addresses case (b), the bottleneck situation. The goal is to either confirm or effectively break the symmetry represented by the giant quotient. This inserts another layer of recurrence into Luks's framework: significant reduction of m, the degree of the giant quotient.

More specifically, let $G \leq \mathrm{Sym}(\Omega)$ be our ambient group and $\mathfrak{x}, \mathfrak{y} : \Omega \to \Sigma$ be two strings of which we wish to determine the G-isomorphisms. Let, further, $\varphi : G \to H$ be an epimorphism where $\mathrm{Alt}(\Gamma) \leq H \leq \mathrm{Sym}(\Gamma)$ for some large set Γ, where $\mathrm{Alt}(\Gamma)$ denotes the alternating group (even permutations of Γ). Let $m = |\Gamma|$ and let $P(\mathfrak{x}) = \varphi(\mathrm{Aut}_G(\mathfrak{x})) \leq \mathrm{Sym}(\Gamma)$; define $P(\mathfrak{y})$ analogously. We say that a group $K \leq \mathrm{Sym}(\Psi)$ is a *giant* on Ψ if $\mathrm{Alt}(\Psi) \leq K \leq \mathrm{Sym}(\Psi)$.

Theorem 5.3 (Canonical obstruction to symmetry)**.** *Either $P(\mathfrak{x})$ acts as a giant on a P-orbit of length $\geq 0.9m$, or there exists a $P(\mathfrak{x})$-invariant canonical k-ary relational structure $\mathfrak{X}(\mathfrak{x})$ on Γ with $k = O(\log n)$ such that $\mathfrak{X}(\mathfrak{x})$ has symmetry defect > 0.1. Moreover, in each case, we can find, via efficient Luks recurrences, an effective representation of the stated objects.*

We explain the concepts involved in this statement.

By 'efficient Luks recurrence' we mean a sequence of Luks operations that significantly reduces the problem size at a multiplicative cost of $n^{O(\log n)}$.

In the first case, 'effective representation' means we can find a subgroup $M \leq \mathrm{Aut}_G(\mathfrak{x})$ such that $\varphi(M)$ has a large orbit on which it acts as a giant. Note that $\mathrm{Aut}_G(\mathfrak{x})$ is not known; in fact, determining $\mathrm{Aut}_G(\mathfrak{x})$ is equivalent to the SI problem.

We need to explain the second case. A k-ary relation on a set Γ is a subset of the Cartesian power Γ^k. A k-ary relational structure on Γ is a pair $\mathfrak{X} = (\Gamma, \mathcal{R})$ where $\mathcal{R} = (R_1, \dots, R_r)$ is a list of k-ary relations R_i on Γ. 'Effective representation' of $\mathfrak{X}$ simply means listing each R_i. We may assume the R_i are disjoint, so the total length of the lists is $\leq m^k$.

We say that the *symmetry defect* of $\mathfrak{X}$ is $\geq \alpha$ if every orbit of $\mathrm{Aut}(\mathfrak{X})$ on which $\mathrm{Aut}(\mathfrak{X})$ acts as a giant has size $\leq (1-\alpha)m$.

Canonicity of the $\mathfrak{x} \mapsto \mathfrak{X}(\mathfrak{x})$ assignment means this construction is a *functor* from the category of G-isomorphisms of strings in the set $\{\mathfrak{x}, \mathfrak{y}\}$ (two objects) to the category of isomorphisms of k-ary relational structures on Γ, so every G-isomorphism $\mathfrak{z}_1 \to \mathfrak{z}_2$ ($\mathfrak{z}_i \in \{\mathfrak{x}, \mathfrak{y}\}$) induces an isomorphism $\mathfrak{X}(\mathfrak{z}_1) \to \mathfrak{X}(\mathfrak{z}_2)$.

The two cases listed in Theorem **??** are mutually exclusive by the definition of symmetry defect. The result provides a *constructive obstruction* to certain type of very large symmetry (small symmetry defect); the stucture $\mathfrak{X}$ has sufficient *irregularity* to preclude such large symmetry. This is the sense in which, under our special circumstances, we have been able to close a *symmetry vs. regularity gap* (see Sec. **??**), a key step toward Theorem **??**.

6 Unaffected Stabilizers Lemma

In this section we state a group theoretic result, Theorem **??** (a), that is our main mathematical (non-algorithmic) tool for the proof of Theorem **??**.

For a group $G \leq \mathrm{Sym}(\Omega)$ and $x \in \Omega$, the *stabilizer* of x in G is the subgroup $G_x = \{\sigma \in G \mid x^\sigma = x\}$. For $\Delta \subseteq \Omega$, the *pointwise stabilizer* of Δ is the subgroup $G_{(\Delta)} = \bigcap_{x \in \Delta} G_x$.

For a group G and a set Γ we say that the action $\varphi : G \to \mathrm{Sym}(\Gamma)$ is a *giant representation* of G (or a *giant homomorphism*) if the image $\varphi(G)$ is a giant, i. e., $\varphi(G) \geq \mathrm{Alt}(\Omega)$. We now define our central new concept.

Definition 6.1 (Affected)**.** Let Ω and Γ be sets, $G \leq \mathrm{Sym}(\Omega)$, and let $\varphi : G \to \mathrm{Sym}(\Gamma)$ be a giant representation. We say that $x \in \Omega$ is *affected* by φ if the φ-image of the stabilizer G_x is not a giant, i. e., $\varphi(G_x) \not\geq \mathrm{Alt}(\Gamma)$.

We note that if $x \in \Omega$ is affected then every element of the orbit x^G is affected. So we can speak of *affected orbits.*

Theorem 6.2. *Let $G \leq \mathrm{Sym}(\Omega)$ be a permutation group of degree $n = |\Omega|$ and $\varphi : G \to S_k$ a giant representation, i. e., $\varphi(G) \geq A_k$. Let $U \subseteq \Omega$ denote the set of elements of Ω not affected by φ. Then the following hold.*

(a) (Unaffected Stabilizers Lemma) *Assume $k > \max\{8, 2 + \log_2 n\}$. Then φ restricted to $G_{(U)}$, the pointwise stabilizer of U, is still a giant representation, i. e., $\varphi(G_{(U)}) \geq A_k$. In particular, $U \neq \Omega$ (at least one element is affected).*

(b) (Affected Orbit Lemma) *Assume $k \geq 5$. If Δ is an affected G-orbit, i. e., $\Delta \cap U = \emptyset$, then* $\ker(\varphi)$ *is not transitive on Δ; in fact, each orbit of* $\ker(\varphi)$ *in Δ has length $\leq |\Delta|/k$.*

The affected/unaffected dichotomy underlies the core "local certificates" algorithm (Sec. **??**).

Part (b) is an easy exercise; its significance is that it permits efficient Luks reductions on affected orbits.

Part (a) is the central result mentioned. The proof of part (a) builds on the O'Nan–Scott–Aschbacher characterization of primitive permutation groups ([**?**, **?**], cf. [**?**, Thm. 4.1A]) and depends on the classification of Finite Simple Groups (CFSG)[4] through *Schreier's Hypothesis* (a consequence of CFSG) that asserts that the outer automorphism group of every finite simple group is solvable.

Note that part (a) is counter-intuitive: it asserts that if the stabilizer of each $x \in U$ maps onto A_k or S_k then even the intersection of these stabilizers maps onto A_k or S_k.

The condition $k > 2 + \log_2 n$ in part (a) is tight. In fact, there are infinitely many examples with $k = 2 + \log_2 n$ which have *no affected points*, as shown by the example of a semidirect product $\mathbb{Z}_2^{k-2} \rtimes A_k \leq AGL(k-2, 2)$ for even k, acting on $n = 2^{k-2}$ elements.

7 Local certificates

In this section we describe our core algorithmic result. The goal is to categorize ordered k-tuples of Γ, setting the stage for a combinatorial analysis

[4] A less tight version of the lemma, still sufficient for the quasipolynomial claim, was recently proved by Pyber [**?**] without the CFSG.

of the resulting k-ary relational structure. The method requires the construction of global automorphisms from local information; our key tool is the Unaffected Stabilizers Lemma.

We consider the Luks bottleneck situation. The input is a transitive group $G \leq \mathrm{Sym}(\Omega)$, a giant representation $\varphi : G \to \mathrm{Alt}(\Gamma)$, and two strings $\mathfrak{x}, \mathfrak{y} : \Omega \to \Sigma$. We write $n = |\Omega|$ and $m = |\Gamma|$. We fix a number $k > 2 + \log_2 n$ (but not much greater, e. g., $k = 3 + \lfloor \log_2 n \rfloor$) and assume $m \geq 10k$. Subsets $T \subset \Gamma$ of size $|T| = k$ will be referred to as "test sets."

If $L \leq G$ then L also acts on Γ via φ so for a test set T we can speak of the setwise stabilizer of T in L; we write L_T for this subgroup.

We say that T is L-invariant if $L_T = L$. We write $\psi_T : G_T \to \mathrm{Sym}(T)$ for the map that restricts the domain of φ to G_T and the codomain to $\mathrm{Sym}(T)$. The group G_T can be computed in polynomial time as $G_T = \varphi^{-1}(\mathrm{Sym}(\Gamma)_T)$. Our focus is the (unknown) group $P(T) := \psi_T(\mathrm{Aut}_{G_T}(\mathfrak{x}))$.

Definition 7.1 (Fullness)**.** Let T be a test set. We say that T is *full* with respect to the input string $\mathfrak{x}$ if $P(T) \geq \mathrm{Alt}(T)$, i. e., the G-automorphisms of $\mathfrak{x}$ induce a giant on T.

We consider the problem of deciding whether a given test set is full and compute useful certificates of either outcome. We show that this question can efficiently (in time $k!\, n^{O(1)}$) be reduced to the String Isomorphism problem on inputs of size $\leq n/k$.

Certificate of non-fullness. We certify non-fullness of the test set T by computing a permutation group $M(T) \leq \mathrm{Sym}(T)$ such that (i) $M(T) \not\geq \mathrm{Alt}(T)$ and (ii) $M(T) \geq P(T)$ ($M(T)$ is guaranteed to contain the projection of the G-automorphism group of $\mathfrak{x}$).

Such an "encasing group" $M(T)$ can be thought of as a constructive refutation of fullness.

Certificate of fullness. We certify fullness of the test set T by computing a permutation group $K(T) \leq \mathrm{Sym}(\Omega)$ such that (i) $K(T) \leq \mathrm{Aut}_{G_T}(\mathfrak{x})$ and (ii) $\psi_T(K(T)) \geq \mathrm{Alt}(T)$.

Note that $K(T) \leq P(T)$, so $K(T)$ represents a polynomial-time verifiable proof of fullness of T.

Our ability to find $K(T)$, the certificate of fullness, may be surprising because it means that from a local start (that may take only a small segment of $\mathfrak{x}$ into account), we have to build up global automorphisms (automorphisms of the full string $\mathfrak{x}$). Our ability to do so critically depends on the "Unaffected Stabilizers Lemma" (Theorem **??** (a)).

Theorem 7.2 (Local certificates)**.** *Let $T \subseteq \Gamma$ where $|T| = k$ is a test set. Assume* $\max\{8, 2 + \log_2 n\} < k \leq m/10$ *(where $m = |\Gamma|$). By making $\leq k!\,n^2$ calls to SI problems on domains of size $\leq n/k$ and performing $k!\,n^{O(1)}$ computation we can decide whether T is full and*

(a) if T is full, find a certificate $K(T) \leq \mathrm{Aut}_G(\mathfrak{x})$ of fullness

(b) if T is not full, find a certificate $M(T) \leq \mathrm{Sym}(T)$ of non-fullness.

To *aggregate* the local certificates, first we consider the group F generated by the fullness certificates. If the support of $\varphi(F) \leq \mathrm{Sym}(\Gamma)$ has at least $m/10$ elements then the structure of $\varphi(F)$ suffices for the proof of Theorem **??**. In the alternative, non-fullness certificates dominate. In this case a slight extension of Theorem **??** is needed, to encase not only the group $\psi_T(\mathrm{Aut}_{G_T}(\mathfrak{x})$ but also the images of the cosets $\mathrm{Iso}_{G_{T,T'}}(\mathfrak{z}_1, \mathfrak{z}_2)$ for all pairs T, T' of test sets and all choices of $\mathfrak{z}_1, \mathfrak{z}_2 \in \{\mathfrak{x}, \mathfrak{y}\}$. The result will be two classifications of the ordered k-tuples of Γ, one associated with $\mathfrak{x}$, the other with $\mathfrak{y}$, yielding the canonical assignment $\mathfrak{x} \mapsto \mathfrak{X}(\mathfrak{x})$ and $\mathfrak{y} \mapsto \mathfrak{X}(\mathfrak{y})$.

8 Individualization and refinement

We consider k-ary *partition structures* $\mathfrak{X} = (\Gamma, \mathcal{R})$ where $\mathcal{R} = (R_1, \ldots, R_r)$ is a partition of Γ^k. We think of such a structure as a *coloring* $c : \Gamma^k \to \{1, \ldots, r\}$ where $c(\vec{x}) = i$ if $\vec{x} \in R_i$ ($\vec{x} \in \Gamma^k$). We also write $\mathfrak{X} = (\Gamma, c)$ instead of $\mathfrak{X} = (\Gamma, \mathcal{R})$. A *refinement* of a coloring c is a coloring c' such that $(\forall \vec{x}, \vec{y} \in \Gamma^k)(c'(\vec{x}) = c'(\vec{y}) \implies c(\vec{x}) = c(\vec{y}))$.

An assignment $\mathfrak{X} \mapsto \mathfrak{X}'$ is *canonical* if it is defined by a functor between categories of isomorphisms of structures.

By a *binary configuration* we mean a binary partition structure $\mathfrak{X} = (\Gamma, c)$ such that

(i) $(\forall x, y, z \in \Gamma)(c(x, y) = c(z, z) \implies x = y)$ and

(ii) $(\forall x, y \in \Gamma)(c(x, y)$ determines $c(y, x))$.

The *Weisfeiler–Leman* canonical refinement process (WL) [**?**, **?**] takes a binary configuration and with every pair $(x, y) \in \Gamma^2$ associates the list $c'(x, y) = (c(x, y), d_{i,j}(x, y) \mid i, j = 1, \ldots, r)$ where $d_i(x, y) = |\{z \in \Gamma \mid c(x, z) = i, c(z, y) = j\}|$. This is clearly a canonical refinement.

Let $\mathfrak{X} = (\Gamma, c)$ be a k-ary partition structure. We assign colors to the elements by setting $c(x) = c(x, \ldots, x)$. *Individualizing* an element $x \in \Gamma$ means assigning it a special color, thereby introducing irregularity. This irregularity propagates via canonical refinement, reducing the isomorphism search space. Let $\mathfrak{X}_x$ denote $\mathfrak{X}$ with $x \in \Gamma$ individualized. Then $\mathfrak{X} \cong \mathfrak{Y} \iff (\exists y \in \Gamma)(\mathfrak{X}_x \cong \mathfrak{Y}_y)$. So progress comes at a multiplicative cost of $m = |\Gamma|$. The multiplicative cost of individualizing t points is n^t, so we need $t \leq$ polylog for a quasipolynomial complexity bound.

9 Coherent configurations

The stable configurations of the WL process (where no proper refinement is obtained) are called *coherent configurations*. This concept goes back to Schur [?] who abstracted its axiom from the *orbital configurations* of permutation groups. An *orbital* of $G \leq \mathrm{Sym}(\Omega)$ is an orbit of the induced action of G on $\Omega \times \Omega$. Let $\mathfrak{X}(G)$ denote the configuration on Ω with the orbitals as the relations. This configuration is clearly coherent, but there are many coherent configurations that do not arise this way. For $v \geq 2k+1$, the *Johnson scheme* $\mathfrak{J}(v, k)$ has $\binom{v}{k}$ vertices; it is defined as the orbital configuration of the group $S_v^{(k)}$ (induced action of S_v on unordered k-tuples).

A coherent configuration is *homogeneous* if every point has the same color. A homogeneous configuration is *primitive* if $|\Gamma| \geq 2$ and each off-diagonal color (relation) is a (strongly) connected (directed) graph. We note that the orbital configuration $\mathfrak{X}(G)$ of a permutation group G is homogeneous iff G is transitive and $\mathfrak{X}(G)$ is primitive iff G is primitive. The *rank* of a configuration is the number of colors, so for $|\Gamma| \geq 2$ the rank is at least 2. The only rank-2 configuration is the *clique*; its automorphism group is $\mathrm{Sym}(\Gamma)$. The Johnson scheme $\mathfrak{J}(v, k)$ has rank $k + 1$.

The WL process and its natural k-ary generalization play a key role in the combinatorial analysis of the k-ary relational structures handed down by the Local Certificates algorithm.

10 Combinatorial partitioning

Recall that we have a giant homomorphism $\varphi : G \to \mathrm{Sym}(\Gamma)$ for some 'ideal domain' Γ and we are given a canonical k-ary partition structure $\mathfrak{X}(\mathfrak{x}) = (\Gamma, c_{\mathfrak{x}})$ with symmetry defect ≥ 0.1 where $\mathfrak{x}$ is the input string. Here $k = O(\log n)$ where $n = |\Omega|$ is the size of our original domain. Recall that our recursive goal is to significantly reduce the size of the ideal domain at moderate multiplicative cost. Ideally we would like to achieve this by finding a *good canonical coloring* of Γ (no color has multiplicity greater than $0.9m$) or a *good equipartition*, i. e., a nontrivial canonical equipartition of the dominant ($> 0.9m$) vertex-color class.

This goal cannot be achieved because of the resilience of the Johnson schemes to canonical partitioning.

Proposition 10.1 (Resilience of Johnson schemes)**.** *The multiplicative cost of a good canonical coloring or a good canonical equipartition of the Johnson scheme* $\mathfrak{J}(v,t)$ *is* $\geq (4t)^{v/(4t)}$.

The proof shows that if we pay less than exponential multiplicative cost then our Johnson scheme is simply reduced to a slighly smaller Johnson scheme.

Note that $t = 2$ is an interesting case, largely responsible for the lack of progress over the $\exp(\widetilde{O}(\sqrt{n}))$ bound for a long time.

The good news is that in a sense, the Johnson schemes are the only obstacles.

So our modified goal will be to find either (a) a good canonical coloring, or (b) a good canonical equipartition, or (c) a canonically embedded Johnson scheme on a dominant vertex-color class. In item (c), canonical embedding means a functor from the isomorphisms of the input structures $\mathfrak{X}$ to the isomorphisms of the secondary structures whose vertex set is a dominant vertex-color class in Γ (under a canonical coloring).

We achieve this goal in two stages: first we go from k-ary to binary (Design Lemma) and then from binary to the desired goal (Split-or-Johnson).

Theorem 10.2 (Design lemma)**.** *Let* $\mathfrak{X} = (\Gamma, c)$ *be a* k*-ary partition structure with* $m = |\Gamma|$ *elements,* $2 \leq k \leq m/2$*, and symmetry defect* ≥ 0.1*. Then in time* $m^{O(k)}$ *we can find a sequence* S *of at most* $k-1$ *vertices such that after individualizing each element of* S *we can either find*

(a) a good canonical coloring of Γ*, or*

(b) a good canonical equipartition of Γ*, or*

(c) a good canonically embedded primitive coherent configuration of rank ≥ 3*.*

Here canonicity is relative to the arbitrary choice of the sequence S.

Outcomes (a) and (b) allow for efficient Luks reduction. Case (c) requires further processing.

Theorem 10.3 (Split-or-Johnson)**.** *Given a primitive coherent configuration* $\mathfrak{X} = (\Gamma, c)$ *of rank* ≥ 3*, at quasipolynomial multiplicative cost we can find either*

(a) a good canonical coloring of Γ*, or*

(b) a good canonical equipartition of Γ*, or*

(c) a good canonically embedded nontrivial Johnson scheme.

Here canonicity is relative to the arbitrary choices made that resulted in the multiplicative cost. The trivial Johnson schemes are the cliques $\mathfrak{J}(v,1)$.

Outcomes (a) and (b) again allow for efficient Luks reduction. Outcome (c) provides even greater efficiency. Assume the canonically embedded Johnson scheme is $\mathfrak{J}(m',t)$; so $m \geq \binom{m'}{t} \geq \binom{m'}{2}$ and therefore $m' < 1+\sqrt{2m}$. Now $\mathrm{Aut}(\mathfrak{J}(m',t)) \cong S_{m'}$, so we can replace Γ by a set Γ' of size $m' = O(\sqrt{m})$, a dramatic reduction of the size of the ideal domain.

Overall algorithm. We follow Luks's algorithm until we hit a bottleneck, at which time an "ideal domain" Γ arises and our recursive goal becomes to significantly reduce the size of the ideal domain. First we use our central group theoretic algorithm ("Local certificates"), based on the "Unaffected Stabilizers Lemma," to construct a canonical structure on Γ of logarithmic arity and with non-negligible symmetry defect. Then we use our combinatorial partitioning algorithms to achieve the desired reduction. Once Γ itself becomes very small (polylogarithmic), we can individualize all of its elements, yielding a significant reduction of n, the size of the input string.

11 Paradoxes of Graph Isomorphism

GI is perceived to be an "easy" computational problem. As discussed in the Introduction (see "Disclaimer"), it is efficiently solved in practice. It is also provably easy on average. Our result shows it has rather low worst-case time complexity. In comparison, the problem of factoring integers is perceived to be "hard" – the assumption that it is hard, not only in the worst case but even of average, is the basis of the RSA cryptosystem and many other cryptographic applications. Yet, by common measures used in structural complexity theory, GI seems harder than factoring. The decision version of the factorization problem is in NP $\cap$ coNP; this is not known to be the case for GI. Factoring is solvable in polynomial time in the quantum computation model; no quantum advantage has been found (in spite of significant effort) for GI. Most remarkable is the series of recent hardness results for GI in proof complexity, inspired by the CFI result. It turns out that in commonly studied hierarchies of semialgebraic and algebraic proof systems, isomorphism of certain pairs of graphs cannot be refuted on levels lower than cn for some constant $c > 0$ (where n is the number of vertices), corresponding to refutation proofs of exponential length in these systems [**?**, **?**, **?**]. (Cf. [**?**] for an overview of these and related systems.)

12 Open problems

Complexity theory. It is not known whether GI belongs to coNP. On the other hand, it is also not known whether P has logspace reductions to GI. This is equivalent to a logspace reduction of the *circuit value problem* (CVP) to GI. The CVP takes a Boolean circuit and an input to the circuit and asks to evaluate the circuit. Such a reduction would be viewed as strong evidence against the existence of an efficient parallel algorithm for GI.

While GI is universal over isomorphism problems for explicit structures, there are interesting classes of isomorphism problems for non-explicit structures that are also not expected to be NP-complete (based on strong evidence from the theory of interactive proofs), yet cannot currently be solved in less than exponential time. Perhaps the simplest among them is the *code equivalence problem* that asks, given two subspaces U and V of $\mathbb{F}^n$ for some finite field $\mathbb{F}$, is there a permutation $\sigma \in S_n$ such that $U^\sigma = V$? Here σ acts on $\mathbb{F}^n$ by permuting the coordinates.

Can GI be solved in quasipolynomial time and *polynomial space*? (Luks)

Can *canonical forms* of graphs be constructed in quasipolynomial time? (Cf. [**?**].)

Can isomophism of hypergraphs be decided in time, quasipolynomial in the number of vertices and *polynomial in the number of edges*?

Combinatorics. The author's decades-old project to find combinatorial relaxations of Cameron's Theorem **??** has seen major progress recently, made by PhD students. *Cameron schemes* are the orbital configurations of Cameron groups (Def. **??**). Let us say that a primitive coherent configuration is a *non-Cameron PCC* if it is not a Cameron scheme. The author has circulated various versions of the following conjectures for some time.

Conjecture 12.1. There exists a polynomial p such that the following hold. Let $\mathfrak{X}$ be a non-Cameron PCC with n vertices. Let $G = \mathrm{Aut}(\mathfrak{X})$. Then

(a) $\theta(\mathrm{Aut}(\mathfrak{X})) \le p(\log n)$ (where θ denotes the thickness, Def. **??**) (polylogarithmically bounded thickness)

(b) $|G| \le \exp(p(\log n))$ (quasipolynomially bounded order)

Part (a) obviously follows from part (b). Regarding (b), for non-Cameron PCCs, an upper bound $|G| \le \exp(\widetilde{O}(\sqrt{n}))$ was proved in [**?**] in 1981. After no progress for three and a half decades, in a recent *tour de force* of combinatorial reasoning, Sun and Wilmes reduced this upper bound to $\exp(\widetilde{O}(n^{1/3}))$, building a new combinatorial structure theory of primitive coherent configurations along the way. The weaker Conjecture (a) has been confirmed for rank-3 configurations (essentially, strongly regular graphs) in [**?**] (2014). Overcoming an array of technical obstacles through a powerful combination of structural and spectral theory, Bohdan Kivva [**?**] very recently confirmed (a) for rank-4 configurations. These are major steps, and raise the hope of further progress, although the technical challenges seem daunting.

References

[*] Abbreviations of frequently cited conferences:
STOC = Ann. ACM Symposium on Theory of Computing
FOCS = Ann. IEEE Symp. on Foundations of Computer Science

[AsS] Michael Aschbacher and Leonard L. Scott: Maximal subgroups of finite groups. *J. Algebra* **92** (1985), 44–80.

[AtM] Albert Atserias and Elitza Maneva: Sherali-Adams relaxations and indistinguishability in counting logics. *SIAM J. Comput.* **42(1)**, 2013, 112-137.

[AtM] Albert Atserias and Joanna Ochremiak: Proof Complexity Meets Algebra. `arXiv:1711.07320`, 2017, 69pp

[Ba79a] László Babai: Monte Carlo algorithms in graph isomorphism testing. Tech. Rep. 79–10, Dép. Math. et Stat., Univ. de Montréal, 1979, 42pp. `link to author's website`

[Ba81] László Babai: On the order of uniprimitive permutation groups. *Annals of Math.* **113(3)** (1981) 553–568.

[Ba83] László Babai: *Permutation Groups, Coherent Configurations, and Graph Isomorphism.* D. Sc. thesis, Hungar. Acad. Sci., 1983. (In Hungarian). Expanded SI proof on `author's website`

[Ba14] László Babai: On the Automorphism Groups of Strongly Regular Graphs I. In: *Proc. 5th Innovations in Theoretical Comp. Sci. Conf. (ITCS'14)*, ACM Press, 2014, pp. 359-368. `DOI`

[Ba15] László Babai: Graph Isomorphism in quasipolynomial time. `arXiv:1512.03547`, 2015

[BaCP] László Babai, Peter J. Cameron, Péter Pál Pálfy: On the orders of primitive groups with restricted nonabelian composition factors. *J. Algebra* **79** (1982), 161–168.

[BaCSTW] László Babai, Xi Chen, Xiaorui Sun, Shang-Hua Teng, John Wilmes: Faster Canonical Forms For Strongly Regular Graphs. *In: 54th IEEE FOCS*, 2013, pp. 157–166.

[BaES] László Babai, Paul Erdős, Stanley M. Selkow: Random graphs isomorphism, *SIAM J. Comput.* **9** (1980), 628-635.

[BaGM] László Babai, Dmitry Yu. Grigoryev, David M. Mount: Isomorphism of graphs with bounded eigenvalue multiplicity. In: *Proc. 14th ACM STOC*, 1982, pp. 310–324.

[BaKL] László Babai, William M. Kantor, Eugene M. Luks: Computational complexity and the classification of finite simple groups. *In: 24th IEEE FOCS*, pp. 162–171, 1983.

[BaKu] László Babai, Ludek Kučera: Canonical labelling of graphs in linear average time, in: *Proc. 20th IEEE FOCS* (1979), 39-46.

[BaL] László Babai, Eugene M. Luks: Canonical labeling of graphs. *In: 15th ACM STOC*, pp. 171–183, 1983.

[BaW] László Babai, John Wilmes: Quasipolynomial-time canonical form for Steiner designs. *In: 45th STOC*, 2013, pp. 261–270.

[BeG] Christoph Berkholz and Martin Grohe: Limitations of Algebraic Approaches to Graph Isomorphism Testing. `arXiv:1502.05912`, 2015, 27pp.

[BoPS] Alexandre V. Borovik, László Pyber, Aner Shalev: Maximal subgroups in finite and profinite groups. *Trans. Amer. Math. Soc.* **348(9)** (1996), 3745–3761.

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

[Ca] Peter J. Cameron: Finite permutation groups and finite simple groups. *Bull. London Math Soc.* **13** (1981) 1–22.

[CST] Xi Chen, Xiaorui Sun, Shang-Hua Teng: Multi-stage design for quasipolynomial-time isomorphism testing of Steiner 2-systems. *In: Proc. 45th ACM STOC*, 2013, pp. 271–280.

[DiM] John D. Dixon, Brian Mortimer: *Permutation Groups.* Springer Grad. Texts in Math. vol. 163, 1996

[FHL] Merrick Furst, John Hopcroft, Eugene Luks: Polynomial-time algorithms for permutation groups. In: *Proc. 21st IEEE FOCS*, 1980, pp. 36–41.

[HT] John Hopcroft, Robert E. Tarjan: Isomorphism of planar graphs. In: *Complexity of Computer Computations* R. M. Miller and J. W. Thatcher, eds.), Plenum Press, 1972, pp. 131–152.

[HW] John Hopcroft, J. K. Wong: Linear time algorithm for isomorphism of planar graphs. in: *Proc. 6th ACM STOC*, 1974, pp. 172–184.

[GaJ] Michael R. Garey and David S. Johnson: *Computers and Intractability: A Guide To NP-Completeness.* W. H. Freeman and Co., New York, 1979.

[HP] Zdeněk Hedrlín, Aleš Pultr: On full embeddings of categories of algebras. *Ill. J. Math.* **10** (1966) 392–406

[Ka] Richard M. Karp: Reducibility among combinatorial problems. in: *Complexity of Computer Computations*, Plenum Press, 1972, pp. 85–103. DOI

[Ki] Bohdan Kivva: On the automorphism groups of distance-regular graphs and rank-4 primitive coherent configurations. Manuscript, University of Chicago, 2017.

[LiS] Martin W. Liebeck, Aner Shalev: Bases of primitive permutation groups. *In: Groups, Combinatorics, and Geometry (Durham 2001)*, pp. 147-154. World Scientific 2003.

[Lu] Eugene M. Luks: Isomorphism of graphs of bounded valence can be tested in polynomial time. *J. Comput. Syst. Sci.* **25(1)** (1982) 42–65.

[Ma] Attila Maróti: On the orders of primitive groups. *J. Algebra* **258(2)** (2002) 631–640.

[McP] Brendan D. McKay and Adolfo Piperno: Practical Graph Isomorphism, II. `arXiv:1301.1493`, 2013

[Mil] Gary L. Miller: Graph isomorphism, general remarks. *J. Comp. Syst. Sci.* **18** (1979), 128–142.

[Miy] Takunari Miyazaki: Luks's reduction of Graph isomorphism to code equivalence. Comment on `The Math Forum`, Sep. 29, 1996.

[NeS] Daneil Neuen, Pascal Schweitzer: An exponential lower bound for Individualization–Refinement algorithms for Graph Isomorphism. `arXiv:1705.03283`, 2017

[OW+] Ryan O'Donnell, John Wright, Chenggang Wu, and Yuan Zhou: Hardness of robust graph isomorphism, Lasserre gaps, and asymmetry of random graphs. In: *Proc. 25th Ann. ACM-SIAM Symp. on Discrete Algorithms,* pp. 16591677, 2014.

[Pá] Péter P. Pálfy: A polynomial bound on the orders of primitive solvable groups. *J. Algebra* **77** (1982) 127–137

[Py] László Pyber. Unpublished, cca. 1990, cited by [**?**]

[Py] László Pyber: A CFSG-free analysis of Babai's quasipolynomial GI-algorithm. `arXiv:1605.08266`, 2016, 8pp.

[Sch] Issai Schur: Zur Theorie der einfach transitiven Permutationsgruppen. *Sitzungsb. Preuss. Akad. Wiss.*, 1933, 598-623

[Sco] Leonard L. Scott: Representations in characteristic *p. In: The Santa Cruz Conf. on Finite Groups*, 1980, AMS, pp. 319–322.

[Se] Ákos Seress: *Permutation Group Algorithms.* Cambridge Univ. Press, 2003

[Sp] Daniel A. Spielman: Faster isomorphism testing of strongly regular graphs. In: *Proc. 28th ACM STOC*, 1996, pp. 576–584.

[SuW] Xiaorui Sun and John Wilmes: Faster canonical forms for primitive coherent configurations. *In: 47th STOC*, 2015, pp. 693–702.

[We] Boris Weisfeiler (ed): *On Construction and Identification of Graphs.* Springer Lect. Notes in Math. Vol 558, 1976.

[WeL] Boris Weisfeiler, Andrei A. Leman: A reduction of a graph to a canonical form and an algebra arising during this reduction. *Nauchno-Technicheskaya Informatsiya* **9** (1968) 12–16.

[Wo] Thomas R. Wolf: Solvable and nilpotent subgroups of $GL(n, q^m)$. *Canad. J. Math.* **34** (1982), 1097–1111.

[ZKT] Viktor N. Zemlyachenko, Nikolai M. Korneenko, Regina I. Tyshkevich: Graph isomorphism problem. *Zapiski Nauchnykh Seminarov LOMI* **118** (1982), 83–158, 215.