Graph Isomorphism in Quasipolynomial Time

László Babai
University of Chicago

Version 2.5

November 2, 2018

Abstract

We show that the Graph Isomorphism (GI) problem and its generalizations, the String Isomorphism (SI) and Coset Intersection (CI) problems, can be solved in quasipolynomial $(\exp\left((\log n)^{O(1)}\right))$ time. The best previous bound for GI was $\exp(O(\sqrt{n \log n}))$, where n is the number of vertices (Luks, 1983); for SI and CI, the bound was similar, $\exp(\widetilde{O}(\sqrt{n}))$, where n is the size of the permutation domain (Babai, 1983).

The SI problem takes as input two strings, $\mathfrak{x}$ and $\mathfrak{y}$, of length n, and a permutation group G of degree n (the "ambient group") and asks if some element of G transforms $\mathfrak{x}$ into $\mathfrak{y}$. Our algorithm builds on Luks's SI framework and attacks its bottleneck, characterized by an epimorphism φ of the ambient group onto the alternating group acting on a set Γ (the "ideal domain") of size $k > c \log n$.

Our goal is to break the homogeneity of the ideal domain. The crucial first step is to find a canonical t-ary relational structure on the ideal domain, with not too much symmetry, for some $t = O(\log n)$. We say that an element x in the domain of the ambient group is *affected* by φ if φ maps the stabilizer of x to a proper subgroup of A_k. The affected/unaffected dichotomy provides a device to construct *global symmetry* from *local information* through the core group-theoretic "local certificates" routine. This algorithm in turn produces the required t-ary structure and thereby sets the stage for symmetry breaking via combinatorial methods of canonical partitioning. The latter lead to the emergence of the Johnson graphs as the sole obstructions to effective canonical partitioning.

For a list of updates compared to the first two arXiv versions, see the Acknowledgments (Sec. 17.1).

WARNING. While the present version fills significant gaps of the previous versions and improves the presentation of some components of the paper, the revision is incomplete; at the current stage, it includes notational, conceptual, and organizational inconsistencies. A fuller explanation of this disclaimer appears in the Acknowledgments (Sec. 17.1) at the end of the paper.

* Research supported in part by NSF Grants CCF-7443327 (2014-current), CCF-1017781 (2010-2014), and CCF-0830370 (2008–2010). Any opinions, findings, and conclusions or recommendations expressed in this paper are those of the author and do not necessarily reflect the views of the National Science Foundation (NSF).

Contents

1 Introduction

1.1 Results and philosophy

1.1.1 Results: the String Isomorphism problem

Let G be a group of permutations of the set $[n] = \{1, \ldots, n\}$ (the "ambient group") and let $\mathfrak{x}, \mathfrak{y}$ be strings of length n over a finite alphabet. The *String Isomorphism (SI) problem* asks, given G, $\mathfrak{x}$, and $\mathfrak{y}$, does there exist an element of G that transforms $\mathfrak{x}$ into $\mathfrak{y}$. So we are looking for "anagrams under a group action." (See the precise definition in Def. 11.1.2. Permutation groups are given by a list of generators.) A function $f(n)$ is *quasipolynomially bounded* if there exist constants c, C such that $f(n) \leq \exp(C(\log n)^c)$ for all sufficiently large n. "Quasipolynomial time" refers to quasipolynomially bounded time.

We prove the following result.

Theorem 1.1.1. *The String Isomorphism problem can be solved in quasipolynomial time.*

The Graph Isomorphism (GI) Problem asks to decide whether two given graphs are isomorphic. The Coset Intersection (CI) problem asks, given two subcosets of the symmetric group, do they have a nonempty intersection.

The SI and CI problems were introduced by Luks [Lu82] (cf. [Lu93]) who also pointed out that these problems are polynomial-time equivalent (under Karp reductions) and GI easily reduces to either. For instance, GI for graphs with n vertices is identical, under obvious encoding, with SI for binary strings of length $\binom{n}{2}$ with respect to the induced action of the symmetric group of degree n on the set of $\binom{n}{2}$ unordered pairs.

Corollary 1.1.2. *The Graph Isomorphism problem and the Coset Intersection problem can be solved in quasipolynomial time.*

The previous best bound for each of these three problems was $\exp(\widetilde{O}(n^{1/2}))$ (the tilde hides polylogarithmic factors[1]), where for GI, n is the number of vertices, for the two other problems, n is the size of the permutation domain. For GI, this bound was obtained in 1983 by combining Luks's group-theoretic algorithm [Lu82] with a combinatorial partitioning lemma by Zemlyachenko (see [ZKT, BaL, BaKL]). For SI and CI, additional group-theoretic observations were used ([Ba83], cf. [BaKL]). No improvement over either of these results was found in the intervening decades.

As an immediate consequence we obtain a slightly stronger result: only the length of the largest orbit of G matters.

[1] Accounting for those logs, the best bound for GI for more than three decades was $\exp(O(\sqrt{n \log n}))$, established by Luks in 1983, cf. [BaKL].

Corollary 1.1.3. *The SI problem can be solved in time, polynomial in n (the length of the strings) and quasipolynomial in $n_0(G)$, the length of the largest orbit of G.*

The first class of graphs for which an efficient isomorphism test was designed using group theory was the class of vertex-colored graphs (isomorphisms preserve color by definition) with bouded color multiplicity [Ba79a] (1979).

Corollary 1.1.4. *The GI problem for vertex-colored graphs can be solved in time, polynomial in n (the number of vertices) and quasipolynomial in the largest color multiplicity.*

1.1.2 The set of G-isomorphisms of strings

We say that the permutation $\sigma \in G$ is a *G-isomorphism* from the string $\mathfrak{x}$ to the string $\mathfrak{y}$ is σ transforms $\mathfrak{x}$ into $\mathfrak{y}$ (notation: $\mathfrak{x}^\sigma = \mathfrak{y}$). The algorithm will not only solve the decision problem "Does there exist a G-isomorphims from $\mathfrak{x}$ to $\mathfrak{y}$" but also compute the set

$$\mathrm{Iso}_G(\mathfrak{x}, \mathfrak{y}) = \{\sigma \in G \mid \mathfrak{x}^\sigma = \mathfrak{y}\} \tag{1}$$

of G-isomorphisms from $\mathfrak{x}$ to $\mathfrak{y}$. This set is either empty or a right coset of the G-automorphism group $\mathrm{Aut}_G(\mathfrak{x}) := \mathrm{Iso}_G(\mathfrak{x}, \mathfrak{x})$. Such a coset is concisely represented by a list of generators of $\mathrm{Aut}_G(\mathfrak{x})$ and a coset representative. It is Luks's seminal discovery [Lu82] that the sets $\mathrm{Iso}_G(\mathfrak{x}, \mathfrak{y})$ are amenable to efficient "Divide-and-Conquer" (recursive) computation, where the recursion is on the group G, under assumptions on the structure of G such as the boundedness of the composition factors. We eliminate the restrictions on G at the cost of relaxing the notion of "efficient" from polynomial time to quasipolynomial time.

1.1.3 Divide-and-Conquer algorithms, quasipolynomial complexity analysis, multiplicative cost

Our algorithm uses the "Divide-and-Conquer" strategy on multiple levels. The idea is to reduce an instance of size n to a *moderate number* of *significantly smaller* instances. In our context, "significantly smaller" means less than, say, 90%. (Any factor bounded away from 1 would do.) Let $q(n)$ denote the maximum number of smaller instances in the reduction where the maximum is taken over all instances of size $\le n$. This is the branching factor in the algorithm; we shall refer to it as the *multiplicative cost.* We then obtain the following recurrence on the complexity of the algorithm:

$$f(n) \le q(n) f(9n/10) \tag{2}$$

where $f(n)$ denotes the maximum cost of solving an instance of size $\le n$. The functions f and q are positive and monotone non-decreasing by definition. For not necessarily integral x we let $f(x) = f(\lceil x \rceil)$. We then infer from Eq. (2) that

$$f(n) \le q(n)^{O(\log n)}. \tag{3}$$

In particular, if $q(n)$ is quasipolynomially bounded then so is $f(n)$. So our goal will be to achieve a significant reduction in the problem size, at a quasipolynomial multiplicative cost.

There is also an additive cost to the reduction (constructing the smaller instances and then assembling their solutions into a solution to the whole problem) but this will typically be absorbed by the multiplicative cost.

In fact, our algorithm uses double recursion. Our ambient group G acts on a domain of size n. In addition to this "original domain," we shall also build an auxiliary set we call the "ideal domain," of size $m \leq n$, along with an action of G on the ideal domain as the symmetric or alternating group. Our focus is on reducing m. Significant progress will be deemed to have occurred if we significantly reduce m, say $m \leftarrow 9m/10$, while not increasing n. When m drops below a threshold $\ell(n)$ that is polylogarithmic in n, we perform brute force enumeration of all permutations of the ideal domain, at a multiplicative cost of $\ell(n)!$. This eliminates the current ideal domain and results in a significant reduction of n. Subsequently a new ideal domain, of size $m \leq$ the new value of n, may arise, and the game starts over. If $f(n,m)$ denotes the maximum cost of solving an instance with original domain size $\leq n$ and ideal domain size $\leq m$ and $q_1(n)$ is the maximum multiplicative cost of significantly reducing m for all instances with $m \leq n$ then we obtain the recurrence

$$f(n,m) \leq q_1(n) f(n, 9m/10) \quad (m \geq \ell(n)) \tag{4}$$

with boundary condition

$$f(n,\ell(n)) \leq (\ell(n)!) f(9n/10) \tag{5}$$

where $f(n) := f(n,n)$. The overall cost estimate becomes

$$f(n) \leq q_1(n)^{O(\log^2 n)} (\ell(n)!)^{O(\log n)}. \tag{6}$$

This bound is quasipolynomial as long as $\ell(n)$ is polylogarithmic and $q_1(n)$ is quasipolynomial.

The actual rate of growth of $\ell(n)$ will be $O(\log n)$. In Section 4.1.2 we shall show how to reduce the $\ell(n)!$ term to $\exp(O(\ell(n)))$, eliminating an annoying $\log\log$ factor from the exponent.

1.1.4 The Luks barrier

We follow Luks's general SI framework [Lu82], developed for his celebrated polynomial-time algorithm to test isomorphism of graphs of bounded valence.

Luks's method applies recursion on the ambient group G. An analysis via a result of Cameron [Cam81] shows that the multiplicative cost of the recursive steps is $n^{O(\log n)}$ unless, for some $m \geq 1 + \log n$, the ambient group G has an epimorphism onto the symmetric group $\mathfrak{S}_m$ or the alternationg group $\mathfrak{A}_m$, in which case the multiplicative cost becomes exponential in m.

We interpret such an epimorphism as a high degree of symmetry of the ambient group. If $\mathrm{Aut}_G(\mathfrak{x})$ shares this symmetry (projects onto a large portion of $\mathfrak{S}_m$) then we can determine $\mathrm{Iso}_G(\mathfrak{x}, \mathfrak{y})$ by efficient Luks recurrence. If this is not the case, we need to break the symmetry, effectively reducing the ambient group and thus enabling recursion on G. Our contribution is breaking this symmetry.

1.1.5 The strategy

We outline our strategy to address the Luks barrier. For a set Γ we denote the symmetric group on Γ by $\mathfrak{S}(\Gamma)$, the alternating group acting on Γ by $\mathfrak{A}(\Gamma$, and call these two groups collectively the *giants* on Γ.

Being in the Luks-barrier case, we have an epimorphism $\varphi : G \to H$ where H is a giant on some set Γ of size m where $c \log n < m \leq n$. We refer to Γ as the "ideal domain."

Let $A = (\mathrm{Aut}_G(\mathfrak{x}))^\varphi$ be the projection of the G-automorphism group of our input string $\mathfrak{x}$ to $\mathfrak{S}(\Gamma)$. We can test, using efficient Luks reduction, whether A has small index in $\mathfrak{S}(\Gamma)$, and if so, find $\mathrm{Iso}_G(\mathfrak{x}, \mathfrak{y})$. In the alternative case, our job is to reduce the ambient group. Since A is not known, we try to *encase* A, i.e., find a group M such that $A \leq M \leq \mathfrak{S}(\Gamma)$ and M has large index in $\mathfrak{S}(\Gamma)$. Such a group would permit us to reduce the ambient group to $\varphi^{-1}(M)$, significant progress.

As a first step toward this goal, we construct a canonical t-ary relational structure $\mathfrak{X}$ on Γ that does not have too much symmetry (has non-negligible *symmetry defeect*, see Def. 2.3.13). Canonicity means the $\mathfrak{x} \mapsto \mathfrak{X}$ assignment is preserved under G-isomorphisms of strings (it is a functor from the category of G-isomorphisms of strings to the category of isomorphisms of t-ary relational structures).

This construction is the heart of the algorithm. It is accomplished by the "local certificates algorithm" that canonically partitions the set Γ^t, yielding the requisite t-ary structure.

We elaborate on this briefly in Sec. 1.1.6.

Once $\mathfrak{X}$ has been found, we use combinatorial partitioning techniques based on the generalized Weisfeiler–Leman refinement method to significantly reduce Γ, and ultimately to break up the set of positions, thus significantly reducing n to permit recurrence.

1.1.6 Philosophy: local to global

We try to extract information about the unknown group $A = (\mathrm{Aut}_G(\mathfrak{x}))^\varphi$, the projection of the G-automorphism group of our input string $\mathfrak{x}$ to $\mathfrak{S}(\Gamma)$.

Our strategy is an interplay between local and global symmetry, formalized through a technique we call *"local certificates."* We shall certify both the presence and the absence of ample local symmetry.

Locality in our context refers to two things. First, we try to understand the action of A on logarithmic-size subdomains of the ideal domain Γ we call "test sets." For a test set $T \subset \Gamma$, we consider the reduced ambient group G_T, the setwise stabilizer of T in G, and the corresponding G_T-automorphism group of the string $\mathfrak{x}$.

We say that the test set T is *full* if $\mathrm{Aut}_{G_T}(\mathfrak{x})$ projects onto a giant on T. We certify fullness by finding a subgroup $B \leq \mathrm{Aut}_{G_T}(\mathfrak{x})$ that projects onto a giant on T. We certify non-fullness by finding an encasing subgroup $M \leq \mathfrak{S}(T)$ such that M is guaranteed to contain the projection of $\mathrm{Aut}_{G_T}(\mathfrak{x})$ into $\mathfrak{S}(T)$ and the index of M in $\mathfrak{S}(T)$ is large.

Being unable to determine $\mathrm{Aut}_{G_T}(\mathfrak{x})$, we look at groups of "local automorphisms:" permutations that respect a substring of the input string $\mathfrak{x}$. We carefully select certain subsets W of the set of positions. We call such a subset a *windows* and then look at the group $H(W)$ of permutations in G_T that respect the substring $\mathfrak{x}^W$ of the input string "visible through the window" (the restriction of $\mathfrak{x}$ to W). (The windows we consider are are invariant under $\mathrm{Aut}_{G_T}(\mathfrak{x})$.) These windows represent the second, deeper sense of locality involved in the *local certificates* algorithm.

The central new concept of this paper is the "affected/unaffected dichotomy" (Sec. 1.2.4). Given a homomorphism of a permutation group $G \leq \mathfrak{S}(\Omega)$ onto a giant on a set Γ, we say that an element $x \in \Omega$ is *affected* by φ if G_x, the stabilizer of x in G, is not mapped by φ onto a giant on Γ.

Using this concept we build an increasing sequence of windows and a corresponding decreasing sequence of local automorphism groups as follows.

Our initial window is empty: the input string is entirely ignored, so our current local automorphism group is $H(\emptyset) = G_T$. At any stage, if $H(W)$ projects onto a giant on T then our next W is the set of positions affected by $H(W)$. The loop terminates when either (i) the projection of M is not a giant on T or (ii) the window stops growing.

All windows built in the process have the property that we are able to determine the local automorphism group $H(W)$ using efficient Luks recurrence. This follows from the fact that in each round we only add point affected by the projection of the current group $H(W)$ to $\mathfrak{S}(\Gamma)$. Here we use the "Affected orbits lemma" (part (b) of Theorem 1.2.1). Moreover, $H(W) \geq \mathrm{Aut}_{G_T}(\mathfrak{x})$.

If for some W the projection M of the group $H(W)$ to $\mathfrak{S}(T)$ is not a giant on T (case (i) of termination of the loop) then the group M encases the projection of $\mathrm{Aut}_{G_T}(\mathfrak{x})$, so M is our non-fullness certificate.

If this is not the case for any W then at some point the window W stops growing while $H(W)$ still projects onto a giant on T (case (ii) of termination of the loop). Our task is to find a subgroup B of $H(W)$ that consists of global automorphisms (G_T-automorphisms of the entire string $\mathfrak{x}$) such that B still projects onto a giant on T.

Our ability to do so critically depends on the "Unaffected Stabilizers Lemma" (part (a) of Theorem 1.2.1)), demonstrating the significance of the affected/unaffected dichotomy. The lemma will imply that we can choose B to be the pointwise stabilizer of the complement of W in $H(W)$.

This transition from local to global symmetry is the key novelty of the paper.

1.1.7 Aggregating the local certificates

The next phase is that we aggregate these $\binom{m}{t}$ local certificates (where $t = |T|$ is the size of the test sets; we shall choose t to be $O(\log n)$) into global information. In fact, not only do we study test sets T but compare pairs T, T' of test sets, and we also compare test sets for the input $\mathfrak{x}$ and for the input $\mathfrak{y}$, so our data for the aggregation procedures take about $4\binom{m}{t}^2$ items of local information as input.

Aggregating the positive certificates is rather simple; these are subgroups of the automorphism group, so we study the group F they generate, and the structure of its projection F^Γ into $\mathfrak{S}(\Gamma)$. If this group is all of $\mathfrak{S}(\Gamma)$ then $\mathfrak{x}$ and $\mathfrak{y}$ are G-isomorphic if and only if they are N-isomorphic where $N = \ker(\varphi)$. The situation is not much different when F^Γ acts as a giant on a large portion of Γ (Section 12).

Otherwise, if F^Γ has large support in Γ but is not a giant on a large orbit of this support, then we can take advantage of the structure of F^Γ (orbits, orbitals (orbits on pairs) of the stabilizer of a small number of points) to obtain the desired split of Γ or a canonically embedded nontrivial regular graph on a large portion of Γ (Section 13.2).

The aggregate of the negative certificates will be a canonical t-ary relational structure on Γ and the subject of our combinatorial reduction techniques (Design Lemma, Sec. 8, and Split-or-Johnson algorithm, Sec. 9) which, in combination, will achieve the desired reduction of Γ.

1.2 The ingredients

The algorithm is based on Luks's classical framework. It has four principal new ingredients: a group-theoretic result, a group-theoretic "local-to-global" algorithm, and two combinatorial partitioning algorithms. The group-theoretic algorithm implements the idea of "local certificates" and provides the structure to which the combinatorial partitioning algorithms will be applied to complete the "divide" phase of the Divide-and-Conquer algorithm. The "conquer" phase remains the same as for Luks.

1.2.1 Notation, terminology. Strings, giants, Johnson groups

For groups G, H we write $H \le G$ to indicate that H is a subgroup of G.

For a set Γ we write $\mathfrak{S}(\Gamma)$ to denote the symmetric group acting on Γ, and $\mathfrak{A}(\Gamma)$ for the alternating group. We refer to these two subgroups of $\mathfrak{S}(\Gamma)$ as the *giants*. If $|\Gamma| = m$ then we also generically write $\mathfrak{S}_m$ and $\mathfrak{A}_m$ for the giants acting on m elements. We say that a homomorphism $\varphi : G \to \mathfrak{S}(\Gamma)$ is a *giant representation* or a *giant action on* Γ if the image G^φ is a giant, i. e., $G^\varphi \ge \mathfrak{A}(\Gamma)$.

We write $\mathfrak{S}^{(t)}(\Gamma)$ for the induced action of $\mathfrak{S}(\Gamma)$ on the set $\binom{\Gamma}{t}$ of t-subsets of Γ. We define $\mathfrak{A}^{(t)}(\Gamma)$ analogously. We call the groups[2] $\mathfrak{S}^{(t)}(\Gamma)$ and $\mathfrak{A}^{(t)}(\Gamma)$ *Johnson groups* and also denote them by $\mathfrak{S}_m^{(t)}$ and $\mathfrak{A}_m^{(t)}$ if $|\Gamma| = m$. Here we assume $1 \le t \le m/2$.

By a *string* over the set Ω of *positions* we mean a function $\mathfrak{x} : \Omega \to \Sigma$ where Σ is a finite alphabet.

Permutations $\sigma \in \mathfrak{S}(\Omega)$ induce an action $\mathfrak{x} \mapsto \mathfrak{x}^\sigma$ on the set of strings $\Omega \to \Sigma$ as follows.

$$\mathfrak{x}^\sigma(i) = \mathfrak{x}(i^{\sigma^{-1}}) . \tag{7}$$

Given a permutation group $G \le \mathfrak{S}\Omega$ we say that $\sigma \in \mathfrak{S}(\Omega)$ is a *G-isomorphism* of the strings $\mathfrak{x}, \mathfrak{y} : \Omega \to \Sigma$ if $\sigma \in G$ and $x^\sigma = \mathfrak{y}$. If a G-isomorphism $\sigma : \mathfrak{x} \mapsto \mathfrak{y}$ exists, we say that $\mathfrak{x}$ and $\mathfrak{y}$ are *G-isomorphic*, and denote this circumstance by $\mathfrak{x} \cong_G \mathfrak{y}$. The set $\mathrm{Iso}_G(\mathfrak{x}, \mathfrak{y})$ of G-isomorphisms of the strings $\mathfrak{x}, \mathfrak{y}$ is defined by Equation (1). The group $\mathrm{Aut}_G(\mathfrak{x}) =: \mathrm{Iso}_G(\mathfrak{x}, \mathfrak{x}) \le \mathfrak{S}(\Omega)$ is the group of *G-automorphisms* of $\mathfrak{x}$. If $\mathrm{Iso}_G(\mathfrak{x}, \mathfrak{y})$ is not empty then

$$\mathrm{Iso}_G(\mathfrak{x}, \mathfrak{y}) = \mathrm{Aut}_G(\mathfrak{x}) \cdot \sigma \quad \text{for any} \quad \sigma \in \mathrm{Iso}(\mathfrak{x}, \mathfrak{y}). \tag{8}$$

[2] We call these groups *Johnson groups* because $\mathfrak{S}_m^{(t)}$ is the automorphism group of the *Johnson graph* (Def. 1.2.3) and both for $\mathfrak{S}_m^{(t)}$ and for $\mathfrak{A}_m^{(t)}$, the *orbital configuration* (Obs. 3.1.12) is the Johnson scheme (Def. 3.2.13). The term "Johnson group" is not standard terminology but the terms "Johnson graph" and "Johnson scheme" are.

The input to the *String Isomorphism problem* is a permutation group $G \leq \mathfrak{S}(\Omega)$ acting on the "original domain" Ω (the "set of positions") and two strings $\mathfrak{x}, \mathfrak{y} : \Omega \to \Sigma$. The String Isomorphism *decision problem* asks whether $\mathfrak{x} \cong_G \mathfrak{y}$. The String Isomorphism *computation problem* asks to compute the set $\mathrm{Iso}_G(\mathfrak{x}, \mathfrak{y})$. If the output $\mathrm{Iso}_G(\mathfrak{x}, \mathfrak{y})$ is not empty then it must be represented by a list of generators of $\mathrm{Aut}_G(\mathfrak{x})$ and a particular G-isomorphism $\sigma \in \mathrm{Iso}_G(\mathfrak{x}, \mathfrak{y})$.

1.2.2 Luks barrier revisited

Luks's SI algorithm proceeds by processing the permutation group $G \leq \mathfrak{S}(\Omega)$ orbit by orbit, reducing to the transitive case with extreme efficiency. If G is transitive, we find a minimal system Φ of imprimitivity (Φ is a G-invariant partition of the permutation domain Ω into maximal blocks). The G-action on the blocks defines a primitive permutation group $\mathfrak{G} \leq \mathfrak{S}(\Phi)$. The naive approach is then to enumerate all elements of $\mathfrak{G}$, each time reducing to the kernel of the $G \to \mathfrak{G}$ epimorphism. So the algorithm is efficient unless we encounter a large primitive group. In fact, one more step of "Luks descent" can be made if $\mathfrak{G}$ contains a transitive, imprimitive normal subgroup of small index.

In 1981, Cameron classified all large primitive groups [Cam81]. It turns out that among those primitive groups $G \leq \mathfrak{S}_n$ that have order at least $n^{1+\log_2 n}$, only the Johnson groups do not possess a transitive but imprimitive proper normal subgroup of index $\leq n$ (Theorem 11.2.1). So the barrier to quasipolynomially efficient application of Luks's method occurs when $\mathfrak{G}$ is a Johnson group, $\mathfrak{S}_m^{(t)}$ or $\mathfrak{A}_m^{(t)}$, for some value m deemed too large to permit full enumeration of $\mathfrak{G}$. (Under brute force enumeration, the number m will go into the exponent of the complexity.) We shall set this threshold at $c \log n$ for some constant c.

Given $\mathfrak{G}$ one can decide in polynomial time whether $\mathfrak{G}$ is a Johnson group and if so, find an isomorphism of $\mathfrak{G}$ to a giant on some set Γ, the "ideal domain" that is constructed along the way; we write $m = |\Gamma|$. Combined with the $G \to \mathfrak{G}$ epimorphism this gives us a giant representation $\varphi : G \to \mathfrak{S}(\Gamma)$.

It is easy to decide recursively whether $\mathrm{Aut}_G(\mathfrak{x})$ maps onto a small-index subgroup of $\mathfrak{S}(\Gamma)$, and if the answer is positive, we can also find the G-isomorphisms of $\mathfrak{x}$ and $\mathfrak{y}$ via efficient Luks-recursion.

So our goal is to significantly reduce $\mathfrak{G}$ unless $\mathrm{Aut}_G(\mathfrak{x})$ maps onto a large portion of $\mathfrak{S}(\Gamma)$.

This reduction was outlined in Sec. 1.1.6.

1.2.3 Group theory required

The algorithm and its analysis heavily depend on the Classification of Finite Simple Groups (CFSG) through Cameron's classification of large primitive permutation groups. Another ingredient where we rely on CFSG occurs in the proof of Lemma 10.2.5 that depends on "Schreier's Hypothesis" (that the outer automorphisms group of a finite simple group is solvable), a consequence of CFSG.

No deep knowledge of group theory is required for reading this paper. The cited consequences of the CFSG are simple to state, and aside from these, we only use elementary group theory.

We should also note that we are able to dispense with Cameron's result using our combinatorial partitioning technique, significantly reducing the dependence of our analysis on the CFSG. Moreover, we can completely eliminate the dependence on the CFSG and still obtain a quasipolynomial bound, albeit with an increased exponent of the exponent, by using a weaker version of Lemma 10.2.5 proved by Pyber without the CFSG [Py17]. We comment on the former in Section 16.1 and on the latter in Remark 10.2.7.

1.2.4 The group-theoretic "local-to-global" tool

In this section we describe our main group theoric tool.

Let $G \leq \mathfrak{S}(\Omega)$ be a permutation group. Recall that we say that a homomorphism $\varphi : G \to \mathfrak{S}_m$ is a *giant representation* of G if G^φ (the image of G under φ) contains $\mathfrak{A}_m$. We say that an element $x \in \Omega$ is *affected* by φ if $G_x^\varphi \not\geq \mathfrak{A}_m$, where G_x denotes the stabilizer of x in G. Note that if x is affected then every element of the orbit x^G is affected. So we can speak of affected orbits.

Theorem 1.2.1. *Let $G \leq \mathfrak{S}(\Omega)$ be a permutation group and let n_0 denote the length of the largest orbit of G. Let $\varphi : G \to \mathfrak{S}_m$ be a giant representation. Let $U \subseteq \Omega$ denote the set of elements of Ω not affected by φ. Then the following hold.*

(a) (Unaffected Stabilizers Lemma) *Assume $m > \max\{8, 2 + \log_2 n_0\}$. Then φ maps $G_{(U)}$, the pointwise stabilizer of U, onto $\mathfrak{A}_m$ or $\mathfrak{S}_m$ (so $\varphi : G_{(U)} \to \mathfrak{S}_m$ is still a giant representation). In particular, $U \neq \Omega$ (at least one element is affected).*

(b) (Affected Orbit Lemma) *Assume $m \geq 5$. 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 $\leq |\Delta|/m$.*

This result is proved in Section 10. Part (b) is a simple observation (Corollary 10.3.7). Part (a) is the main content of the result; it appears as Theorem 10.3.5 in Section 10.

Remark 1.2.2. We note that part (a) becomes false if we relax the condition $m > 2+\log_2 n_0$ to $m \geq 2 + \log_2 n_0$. In Remark 10.2.6 we exhibit infinitely many transitive groups with giant actions with $m = 2+\log_2 n$ where none of the elements is affected (and the kernel is transitive).

The affected/unaffected dichotomy is our central *local-to-global* tool[3], underlying our divide-and-conquer strategy.

The two results stated are employed in Procedure LocalCertificates, the heart of the entire algorithm, in Section 13. It is Theorem 1.2.1 that allows us to build up local symmetry to global automorphism unless an explicit obstruction is found.

1.2.5 Combinatorial partitioning; emergence of a canonically embedded large Johnson graph

The partitioning algorithms take as input a set Ω related in some way to a structure X. The goal is either to establish high symmetry of X or to find a canonical structure on Ω that represents an explicit obstruction to such high symmetry.

[3]The discovery of this tool was the turning point of this project, cf. footnote 10 in Sec. 13.1.

Significant partitioning is expected at modest "multiplicative cost" (explained below). Favorable outcomes of the partitioning algorithms are (a) a canonical coloring of Ω where each color-class has size $\leq 0.9n$ ($n = |\Omega|$), or (b) a canonical equipartition of a canonical subset of Ω of size $\geq 0.9n$.

Definition 1.2.3 (Johnson graph)**.** Let $t \geq 2$ and $v \geq 2t+1$. The Johnson graph $J(v,t)$ is an unidrected graph with $n = \binom{v}{t}$ vertices labeled by the t-subsets $T \subseteq [v]$. The t-subsets T_1, T_2 are adjacent if $|T_1 \setminus T_2| = 1$.

Johnson graphs do not admit a coloring/partition as described, even at quasipolynomial multiplicative cost, if t is subpolynomial in v (i. e., $t = v^{o(1)}$). (Johnson graphs with $t = 2$ have been the most notorious obstacles to breaking the $\exp(\widetilde{O}(\sqrt{n}))$ bound on GI.) One of the main results of the paper is that in a well–defined sense, Johnson graphs are the *only* obstructions to effective partitioning: either partitioning succeeds as desired or a canonically embedded Johnson graph on a subset of size $\geq 0.9n$ is found. Here is a corollary to the result.

Theorem 1.2.4. *Let $X = (V,E)$ be a nontrivial regular graph (neither complete, nor empty) with n vertices. At a quasipolynomial multiplicative cost we can find one of the following structures. We call the structure found Y.*

(a) A coloring of V with no color-class larger than $0.9n$;

(b) A coloring of V with a color-class C of size $\geq 0.9n$ and a nontrivial equipartition of C (the blocks of the partition are of equal size ≥ 2 and there are at least two blocks);

(c) A coloring of V with a color-class C of size $\geq 0.9n$ and a Johnson graph $J(v,t)$ $(t \geq 2)$ with vertex-set C,

such that the index of the subgroup $\mathrm{Aut}(X) \cap \mathrm{Aut}(Y)$ in $\mathrm{Aut}(X)$ is quasipolynomially bounded.

The index in question (and its natural extension to isomorphisms) represents the multiplicative cost incurred. The full statement can be found in Theorem 9.2.1.

The same is true if X is a k-ary relational structure that does not admit the action of a symmetric group of degree $\geq 0.9n$ on its vertex set (has "symmetry defect" $\geq 0.1n$, see Def. 2.3.13) assuming k is polylogarithmically bounded. The reduction from k-ary relations ($k \geq 3$) to regular graphs (and to highly regular binary relational structures called "uniprimitive coherent configurations" or UPCCs) is the content of the Design Lemma (Theorem 8.1.2).

Note that the Johnson graph will not be a subgraph of X; but it will be "canonically embedded" relative to an arbitrary choice from a quasipolynomial number of possibilities, with the consequence of not reducing the number of automorphisms/isomorphisms by more than a quasipolynomial factor.

The number 0.9 is arbitrary; the result would remain valid for any constant $0.5 < \alpha < 1$ in place of 0.9.

We note that the *existence* of such a structure Y can be deduced from the Classification of Finite Simple Groups. We not only assert the existence but also find such a structure in quasipolynomial time, and the analysis is almost entirely combinatorial, with a modest use of elementary group theory.

The structure Y is "canonical relative to an arbitrary choice" from a quasipolynomial number of possibilities. These arise by individualizing a polylogarithmic number of "ideal points" of Y. An "ideal point" of X is a point of a structure X' canonically constructed from X, much like "ideal points" of an affine plane are the "points at infinity." Individualizing a point at infinity means individualizing a parallel class of lines in the affine plane.

Canonicity means being preserved under isomorphisms in a category of interest. This category is always very small, it often has just two objects (the two graphs or strings of which we wish to decide isomorphism); sometimes it has a quasipolynomial number of objects (when checking local symmetry, we need to compare every pair of polylogarithmic size subsets of the domain). In any case, this notion of canonicity does not require canonical forms for the class of all graphs or strings, a problem we do not address in this paper. We say that we incur a "multiplicative cost" τ if a choice is made from τ possibilities. This indeed makes the algorithm branch τ ways, giving rise to a factor of τ in the recurrence.

Canonicity and "relative canonicity at a multiplicative cost" are formalized in the language of functors in Section 6.

2 Preliminaries

2.1 Fraktur

We list the Roman equivalents of the letters in Fraktur we use:
$\mathfrak{x}$ – x, $\mathfrak{y}$ – y, $\mathfrak{z}$ – z,
$\mathfrak{A}$ – A, $\mathfrak{B}$ – B, $\mathfrak{G}$ – G, $\mathfrak{H}$ – H, $\mathfrak{J}$ – J, $\mathfrak{L}$ – L, $\mathfrak{P}$ – P, $\mathfrak{S}$ – S, $\mathfrak{X}$ – X, $\mathfrak{Y}$ – Y, $\mathfrak{Z}$ – Z

2.2 Permutation groups

All groups in this paper are finite. Our principal reference for permutation groups is the monograph by Dixon and Mortimer [DiM]. Wielandt's classic [Wi3] is a sweet introduction. Cameron's article [Cam81] is very informative. For the basics of permutation group algorithms we refer the reader to Seress's monograph [Se]. Even though we summarize Luks's method in our language in Sec. 11.1, Luks's seminal paper [Lu82] is a prerequisite for this one.

2.2.1 Notation, terminology

For a set Ω we write $\mathfrak{S}(\Omega)$ for the symmetric group consisting of all permutations of Ω and $\mathfrak{A}(\Omega)$ for the alternating group on Ω (set of even permutations of Ω). We write $\mathfrak{S}_n$ for $\mathfrak{S}([n])$ and $\mathfrak{A}_n$ for $\mathfrak{A}([n])$ where $[n] = \{1, \ldots, n\}$. We also use the symbols $\mathfrak{S}_n$ and $\mathfrak{A}_n$ when the permutation domain is not specified (only its size). For a function f we usually write x^f for $f(x)$. In particular, for $\sigma \in \mathfrak{S}(\Omega)$ and $x \in \Omega$ we denote the image of x under σ by x^σ. For $x \in \Omega$, $\sigma \in \mathfrak{S}(\Omega)$, $\Delta \subseteq \Omega$, and $H \subseteq \mathfrak{S}(\Omega)$ we write

$$x^H = \{x^\sigma \mid \sigma \in H\} \text{ and } \Delta^\sigma = \{y^\sigma \mid y \in \Delta\} \text{ and } \Delta^H = \{\Delta^\sigma \mid \sigma \in H\}. \tag{9}$$

For groups G, H we write $H \leq G$ to indicate that H is a subgroup of G. The expression $|G : H|$ denotes the *index* of H in G. Subgroups $G \leq \mathfrak{S}(\Omega)$ are the *permutation groups* on the domain Ω. The size of the permutation domain, $|\Omega|$, is called the *degree* of G while $|G|$ is the *order* of G. We refer to $\mathfrak{S}(\Omega)$ and $\mathfrak{A}(\Omega)$, the two largest permutation groups on Ω, as the *giants* on Ω.

By a *representation* of a group G we shall always mean a *permutation representation*, i. e., a homomorphism $\varphi : G \to \mathfrak{S}(\Omega)$. We also say in this case that G *acts on* Ω (via φ). We say that Ω is the *domain* of the representation and $|\Omega|$ is the *degree* of the representation. If φ is evident from the context, we write x^π for x^{π^φ}. For $x \in \Omega$, $\sigma \in G$, $\Delta \subseteq \Omega$, and $H \subseteq G$, we define x^H and Δ^σ and Δ^H by Eq. (9).

We denote the image of G under φ by G^φ, so $G^\varphi \cong G/\ker(\varphi)$. If $G^\varphi \geq \mathfrak{A}(\Omega)$ we say φ is a *giant representation* and G acts on Ω "as a giant."

A subset $\Delta \subseteq \Omega$ is *G-invariant* if $\Delta^G = \Delta$.

Notation 2.2.1. If $\Delta \subseteq \Omega$ is G-invariant then G^Δ denotes the image of the representation $G \to \mathfrak{S}(\Delta)$ defined by restriction to Δ. So $G^\Delta \leq \mathfrak{S}(\Delta)$.

The *stabilizer* of $x \in \Omega$ is the subgroup $G_x = \{\sigma \in G \mid x^\sigma = x\}$. The *orbit* of $x \in \Omega$ is the set $x^G = \{x^\sigma \mid \sigma \in G\}$. The orbits partition Ω. A simple bijection shows that

$$|x^G| = |G : G_x|. \tag{10}$$

For $T \subseteq \Omega$ and $G \leq \mathfrak{S}(\Omega)$ we write G_T for the *setwise stabilizer* of T and $G_{(T)}$ for the *pointwise stabilizer* of T, i. e.,

$$G_T = \{\alpha \in G \mid T^\alpha = T\} \tag{11}$$

and

$$G_{(T)} = \{\alpha \in G \mid (\forall x \in T)(x^\alpha = x)\}. \tag{12}$$

So $G_{(T)}$ is the kernel of the $G_T \to \mathfrak{S}(T)$ homomorphism obtained by restriction to T; in particular, $G_{(T)} \triangleleft G_T$.

For $t \geq 0$ we write $\binom{\Omega}{t}$ to denote the set of t-subsets of Ω. So if $|\Omega| = k$ then $\left|\binom{\Omega}{t}\right| = \binom{k}{t}$. A permutation group $G \leq \mathfrak{S}(\Omega)$ naturally acts on $\binom{\Omega}{t}$; we refer to this as the *induced action on t-sets* and denote the resulting subgroup of $\mathfrak{S}\binom{\Omega}{t}$ by $G^{(t)}$. This in particular defines the notation $\mathfrak{S}_k^{(t)}$ and $\mathfrak{A}_k^{(t)}$; these are subgroups of $\mathfrak{S}_{\binom{k}{t}}$. We refer to $\mathfrak{S}_k^{(t)}$ and $\mathfrak{A}_k^{(t)}$ as *Johnson groups* since they act on the "Johnson schemes" (see below)[4].

The group G is *transitive* if it has only one orbit, i. e., $x^G = \Omega$ for some (and therefore any) $x \in \Omega$. The G-invariant sets are the unions of orbits.

A *G-invariant partition* of Ω is a partition $\{B_1, \ldots, B_m\}$ where the B_i are nonempty, pairwise disjoint subsets of which the union is Ω such that G permutes these subsets, i. e., $(\forall \sigma \in G)(\forall i)(\exists j)(B_i^\sigma = B_j)$. The B_i are the *blocks* of this partition.

[4] "Johnson schemes" is a standard term; we introduce the term "Johnson groups" for convenience.

A nonempty subset $B \subseteq \Omega$ is a *block of imprimitivity* for G if $(\forall g \in G)(B^g = B$ or $B^g \cap B = \emptyset)$. A subset $B \subseteq \Omega$ is a block of imprimitivity if and only if it is a block in an invariant partition.

A *system of imprimitivity* for G is a G-invariant partition $\mathcal{B} = \{B_1, \ldots, B_m\}$ of a G-invariant subset $\Delta \subseteq \Omega$ such that G acts transitively on $\mathcal{B}$. (So $\Delta = \bigcup_i B_i$; we assume here that $(\forall i)(B_i \neq \emptyset)$). The B_i are then blocks of imprimitivity, and every system of imprimitivity arises as the set of G-images of a block of imprimitivity. The group G acts on $\mathcal{B}$ by permuting the blocks; this defines a representation $G \to \mathfrak{S}_m$.

A *maximal system of imprimitivity* for G is a system of imprimitivity of blocks of size ≥ 2 that cannot be refined, i. e., where the blocks are minimal (do not properly contain any block of imprimitivity of size ≥ 2).

$G \leq \mathfrak{S}(\Omega)$ is *primitive* if $|G| \geq 2$ and G has no blocks of imprimitivity other than Ω and the singletons (sets of one element). In particular, a primitive group is transitive. Examples of primitive groups include the cyclic group of prime order p acting naturally on a set of p elements, and the Johnson groups $\mathfrak{S}_k^{(t)}$ and $\mathfrak{A}_k^{(t)}$ for $t \geq 1$ and $k \geq 2t+1$.

Definition 2.2.2. The *support* of a permutation $\sigma \in \mathfrak{S}(\Omega)$ is the set of elements that σ moves: $\operatorname{supp}(\sigma) = \{x \in \Omega \mid x^\sigma \neq x\}$. The *degree* of σ is the size of its support. The *minimal degree* of a permutation group G is $\min_{\sigma \in G, \sigma \neq 1} |\operatorname{supp}(\sigma)|$.

2.2.2 Degree of transitivity

A group $G \leq \mathfrak{S}(\Omega)$ is *doubly transitive* if its induced action on the set of $n(n-1)$ ordered pairs is transitive (where $n = |\Omega|$).

More generally, $G \leq \mathfrak{S}(\Omega)$ is *t-transitive* if its induced action on the set of $n(n-1)\cdots(n-t+1)$ ordered t-tuples of distinct elements is transitive (where $n = |\Omega|$).

Definition 2.2.3. We say that $\operatorname{degtrans}(G)$, the *degree of transitivity* of G, is t if G is t-transitive but not $(t+1)$-transitive.

The giants have high degree of transitivity: $\operatorname{degtrans}(\mathfrak{S}_n) = n$ and $\operatorname{degtrans}(\mathfrak{A}_n) = n-2$. For all other permutation groups, the degree of transitivity is ≤ 5.

Theorem 2.2.4 (Degree of transitivity)**.** *Let $G \leq \mathfrak{S}_n$ be t-transitive. Assume G is not a giant. Then*

(a) (Curtis, Kantor, Seitz [CuKS]) $\quad t \leq 5$

(b) ([CuKS]) $\quad$ *If $n \geq 25$ then $t \leq 3$.*

These results depend on the Classification of Finite Simple Groups (CFSG). For our purposes, the following elementary result will suffice.

Theorem 2.2.5 (Wielandt)**.** *Let $G \leq \mathfrak{S}_n$ be t-transitive. Assume G is not a giant. Then*

a $\quad t < 3 \ln k$.

b $\quad t \leq 7$ *assuming Schreier's Hypothesis.*

Result (a) appears in Wielandt's dissertation (1934) [Wi1] and is cited in the Remarks after Thm. 9.7 in [Wi3]. This result does not depend on CFSG. In fact, an even more elementary $O(\log^2 n/\log\log n)$ bound by Bochert [Bo92] would suffice (see [BaS] for a 2-page proof). (Jordan cites and slightly improves Bochert's bound in [Jor2] (improving only a lower-order term).) Result (b) appears in Wielandt [Wi2] (see [DiM, Thm. 7.3A]).

2.2.3 Polynomial-time algorithms in permutation groups

A few well-known facts, cf. [Se]

TO BE WRITTEN *******

Proposition 2.2.6. (a) [Kernel of action] *Let Ω and Γ be sets, $G \le \mathfrak{S}(\Omega)$, and let $\varphi : G \to \mathfrak{S}(\Gamma)$ be a G-action on Γ. Then one can, in polynomial time, determine* $\ker(\varphi)$*, the kernel of this action.*

(b) [Lifting] *Let, in addition, $\tau \in \mathfrak{S}(\Gamma)$. Then one can, in polynomial time, determine the set $\varphi^{-1}(\tau)$ (which is either empty or a coset of* $\ker(\varphi)$ *in G).*

2.3 Relational structures, twins, symmetricity and symmetry defect

2.3.1 Relational structures, isomorphism

Definition 2.3.1. A *k-ary relation* on the set Ω is a subset $R \subseteq \Omega^k$. We say that the *arity* of R is k. A *relational structure* $\mathfrak{X} = (\Omega; \mathcal{R})$ consists of Ω, the set of *vertices*, and $\mathcal{R} = (R_1, \dots, R_r)$, a list of relations on Ω. We write $\Omega = V(\mathfrak{X})$. We say that $\mathfrak{X}$ is a *k-ary relational structure* if each R_i is k-ary.

Notation 2.3.2. Let $\vec{x} = (x_1, \dots, x_k) \in \Omega^k$ and let $f : \Omega \to \Omega'$ be a function. Then we write $\vec{x}^f = (x_1^f, \dots, x_k^f) \in \Omega'^k$.

Notation 2.3.3. Let $\sigma : \Omega \to \Omega'$ be a bijection and $R \subseteq \Omega^k$ a k-ary relation. Then we define $R^\sigma \subseteq \Omega'^k$ by setting $R^\sigma = \{\vec{x}^\sigma \mid \vec{x} \in R\}$. For a relational structure $\mathfrak{X} = (\Omega; \mathcal{R})$, where $\mathcal{R} = (R_1, \dots, R_r)$, and a bijection $\sigma : \Omega \to \Omega'$, we set $\mathfrak{X}^\sigma = (\Omega'; \mathcal{R}^\sigma)$, where $\mathcal{R}^\sigma = (R_1^\sigma, \dots, R_r^\sigma)$.

Definition 2.3.4 (Isomorphisms)**.** Let $\mathfrak{X} = (\Omega; \mathcal{R})$ and $\mathfrak{X}' = (\Omega'; \mathcal{R}')$ be relational structures where $\mathcal{R} = (R_1, \dots, R_r)$ and $\mathcal{R}' = (R'_1, \dots, R'_r)$. A bijection $\sigma : \Omega \to \Omega'$ is an $\mathfrak{X} \to \mathfrak{X}'$ *isomorphism* if $\mathfrak{X}' = \mathfrak{X}^\sigma$. We say that $\mathfrak{X} \cong \mathfrak{X}'$ ($\mathfrak{X}$ and $\mathfrak{X}'$ are *isomorphic*) if such σ exists.

Observation 2.3.5. Using the notation of Def. 2.3.4, if $\mathfrak{X} \cong \mathfrak{X}'$ then $r = r'$ and for each i, the relations R_i and R'_i have the same arity.

Proposition 2.3.6 (Testing candidate isomorphism)**.** *Given two explicit relational structures $\mathfrak{X} = (\Omega, \mathcal{R})$ and $\mathfrak{X}' = (\Omega', \mathcal{R}')$ and a bijection $\sigma : \Omega \to \Omega'$, one can decide in linear time whether $\sigma \in \mathrm{Iso}(\mathfrak{X}, \mathfrak{X}')$. In particular, given an explicit relational structure $\mathfrak{X} = (\Omega, \mathcal{R})$ and a permutation $\sigma \in \mathfrak{S}(\Omega)$, one can decide in linear time whether $\sigma \in \mathrm{Aut}(\mathfrak{X})$.*

"Explicitness" means each relation is given as the list of its elements.

Proof. Performing this test in polynomial time would be straightforward and we shall never need more than that. To get it down to linear time, we first construct $\mathfrak{X}^\sigma$, then lexicographically sort each relation in $\mathcal{R}^\sigma$ as well as in $\mathcal{R}'$ with respect to an arbitrary ordering of Ω' (this can be done in linear time using *radix sort*), and finally comparing each relation in $\mathcal{R}^\sigma$ with the corresponding relation in $\mathcal{R}'$ (in linear time since now each relation is sorted). $\square$

Definition 2.3.7 (Induced substructure)**.** Let $\Delta \subseteq \Omega$ and let $\mathfrak{X} = (\Omega; R_1, \ldots, R_r)$ be a k-ary relational structure. Let $R_i^\Delta = R_i \cap \Delta^k$. We define the *induced substructure* $\mathfrak{X}[\Delta]$ of $\mathfrak{X}$ on Δ as $\mathfrak{X}[\Delta] = (\Delta; R_1^\Delta, \ldots, R_r^\Delta)$.

2.3.2 Twins, symmetricity, symmetry defect

Convention 2.3.8. Let $\Psi \subseteq \Omega$. We view $\mathfrak{S}(\Psi)$ as a subgroup of $\mathfrak{S}(\Omega)$ under the natural embedding, extending each element of $\mathfrak{S}(\Psi)$ to act trivially on $\Omega \setminus \Psi$.

Definition 2.3.9 (Twins)**.** Let $G \leq \mathfrak{S}(\Omega)$ and $x, y \in \Omega$. We say[5] that the points $x \neq y$ are *twins* with respect to G if the transposition $\tau = (x, y)$ belongs to G.

Observation 2.3.10. The "twin or equal" relation is an equivalence relations on Ω.

Definition 2.3.11. We call the equivalence classes of the twin-or-equal relation the *twin equivalence classes* of G. We shall say that a set $\Psi \subseteq \Omega$ is a *set of twins* if Ψ is a subset of a twin equivalence class, i. e., if all pairs in Ψ are twins.

Observation 2.3.12. A subset $\Psi \subseteq \Omega$ is a set of twins if and only if $\mathfrak{S}(\Psi) \leq G$. The twin equivalence classes are the maximal sets of twins.

Definition 2.3.13 (Symmetricity and symmetry defect of permutation groups)**.** The (absolute) *symmetricity* $s(G)$ of $G \leq \mathfrak{S}(\Omega)$ is the size of its largest twin equivalence class. The *relative symmetricity* of G is $s(G)/|\Omega|$. The *symmetry defect* of G is the complementary quantity $|\Omega| - s(G)$. The *relative symmetry defect* of G is $\operatorname{defect}(G) = 1 - s(G)/|\Omega|$. Note that $0 \leq \operatorname{defect}(G) < 1$.

We shall often omit the terms "absolute" and "relative"; if we say that "the symmetry defect is β," it should be clear from the context whether β is an integer (absolute) or $0 \leq \beta < 1$ (relative). (If $\beta = 0$ then two interpretations have the same meaning, namely, $G = \mathfrak{S}(\Omega)$.)

Examples 2.3.14. The symmetry defect of $\mathfrak{S}_n$ is 0, and the symmetry defect of $\mathfrak{A}_n$ is $n-1$. If $\Omega = \Omega_1 \dot\cup \Omega_2$ and $G = \mathfrak{S}(\Omega_1) \times \mathfrak{S}(\Omega_2)$ then the symmetry defect of G is $\min(|\Omega_1|, |\Omega_2|)$.

Observation 2.3.15. The symmetry defect is monotone decreasing:

$$\text{if } H \leq G \text{ then } \operatorname{defect}(H) \geq \operatorname{defect}(G).$$

[5]Versions 1 and 2 of this paper posted on arXiv speak about "strong" and "weak" twins. The present definition corresponds to "strong twins"; we do not need the notion of "weak twins."

We now define the symmetry defect of structures, a key parameter that will play a central role as the loop invariant in our algorithms. By "structure" in the next statement we mean any member of a concrete category; the only thing that matters is that a structure $\mathfrak{X}$ has an underlying set $\Omega = \square(\mathfrak{X})$ and the automorphisms of $\mathfrak{X}$ are permutations of Ω. ("$\square$" is a *forgetful functor* from a category to the category of sets and mappings.) Examples we shall use are relational structures and hypergraphs.

Definition 2.3.16 (Symmetricity and symmetry defect of structures)**.** Let $\mathfrak{X}$ be a structure with underlying set Ω. We say that $x, y \in \Omega$ are twins with respect to $\mathfrak{X}$ if they are twins with respect to $\mathrm{Aut}(\mathfrak{X})$. We define the (absolute and relative) *symmetricity* and *symmetry defect* of $\mathfrak{X}$ as the corresponding quantity for $\mathrm{Aut}(\mathfrak{X})$. We use the notation $\mathrm{defect}(\mathfrak{X})$ to denote $\mathrm{defect}(\mathrm{Aut}(\mathfrak{X}))$.

Observation 2.3.17 (Computing symmetricity and symmetry defect)**.** We observed that candidate isomorphisms of explicit relational structures can be tested in polynomial time (Prop. 2.3.6). In particular, candidate automorohisms can be tested in polynomial time.

Consequently, one can find the twin equivalence classes in polynomial time (test each transposition). Therefore, the symmetricity and the symmetry defect of explicit relational structures can also be computed in polynomial time. A similar observation holds for explicit hypergraphs (see Obs. 2.5.8).

2.4 Binary relations

2.4.1 Digraphs

We give a brief self-contained introduction to directed graphs for two reasons: (1) terminology and notation in the relevant textbooks are not uniform, and (2) we offer the reader immediate access to those basic facts that we shall need.

Notation 2.4.1 (In-neighbors, out-neighbors)**.** Let $R \subseteq \Omega \times \Omega$ be a binary relation on the set Ω. The *inverse* relation R^- is defined as $R^- = \{(y, x) \mid (x, y) \in R\}$. If $(x, y) \in R$, we say that y is an *out-neighbor* of x and x is an *in-neighbor* of y. For $x \in \Omega$ we write $R(x) = \{y \in \Omega \mid (x, y) \in R\}$ for the set of out-neighbors of x. Note that $R^-(x)$ is the set of in-neighbors of x.

Definition 2.4.2. The *diagonal* of the set Ω is the set $\mathrm{diag}(\Omega) = \{(x, x) \mid x \in \Omega\}$. The relation $R \subseteq \Omega \times \Omega$ is *irreflexive* if $R \cap \mathrm{diag}(\Omega) = \emptyset$.

Definition 2.4.3. A *digraph* (directed graph) is a pair $X = (\Omega, E)$ where $E \subseteq \Omega \times \Omega$ is a binary relation on Ω. We say that Ω is the set of *vertices* and E is the set of *edges* of X. Vertex u is the *tail* and vertex v the *head* of the edge (u, v). The edge *emanates* from its tail and is *absorbed* by its head. We call E the *adjacency relation.* If $(u, v) \in E$, we say that u is *adjacent to* v. If we reverse every edge, we obtain the digraph $X^- = (\Omega, E^-)$.

We call the substructures of a digraph "subgraphs," avoiding the cumbersome term "subdigraph." Thus for $\Delta \subseteq \Omega$, the *induced subgraph* $X[\Delta]$ of the digraph $X = (\Omega, E)$ is defined as $X[\Delta] = (\Delta, E \cap (\Delta \times \Delta)$.

We say that X is *empty* if it has no edges ($E = \emptyset$). The *inverse* of X is the digraph $X^- = (\Omega, E^-)$. The *out-degree* of vertex $u \in \Omega$ is $\deg^+(u) = |E(v)|$, the number of out-neighbors of u. Analogously, the in-degree $\deg^-(u) = |E^-(u)|$ is the number of in-neighbors of u. An *isolated vertex* is a vertex u with $\deg^+(u) = \deg^-(u) = 0$. An edge of the form (u, u) (diagonal element of $\Omega \times \Omega$) is referred to as a *self-loop* attached to vertex u. The digraph is *oriented* if E is antisymmetric, i. e., $E \cap E^- = \emptyset$.

We extend the notions of in- and out-neighbors (Notation 2.4.1) to subsets of the vertex set.

Definition 2.4.4 (Neighborhood)**.** Let $X = (\Omega, E)$ be a digraph and let $\Delta \subseteq \Omega$. The *out-neighborhood* of Δ is the set

$$E(\Delta) = \bigcup_{x \in \Delta} E(x) = \{y \in \Omega \mid (\exists x \in \Delta)((x, y) \in E).$$

The in-neighborhood is $E^-(\Delta)$.

Observation 2.4.5 (Directed "handshake theorem")**.**

$$\sum_{u \in \Omega} \deg^+(u) = \sum_{u \in \Omega} \deg^-(u) = |E|. \qquad \square$$

Definition 2.4.6 (Biregular digraph)**.** X is *biregular* if all vertices have the same in-degree and all vertices have the same out-degree (so these two numbers are necessarily equal).

Definition 2.4.7 (Complement)**.** We say that the digraph X is *irreflexive* if the relation E is irreflexive. The *irreflexive complement* of an irreflexive digraph $X = (\Omega, E)$ is $\overline{X} = (\Omega, \overline{E})$ where $\overline{E} = (\Omega \times \Omega) \setminus (\mathrm{diag}(\Omega) \cup E)$.

Definition 2.4.8 (Complete digraph)**.** Let $X = (\Omega, E)$. If $E = \Omega \times \Omega$, we call X the *complete reflexive digraph* on Ω; and if $E = (\Omega \times \Omega) \setminus \mathrm{diag}(\Omega)$, we call X the *complete irreflexive digraph* on *clique* on Ω.

Definition 2.4.9 (Trivial digraphs)**.** We say that $X = (\Omega, E)$ is *trivial* if $\mathrm{Aut}(X) = \mathfrak{S}(\Omega)$, i. e., if X is empty, the diagonal, or the refexive or irreflexive complete digraph.

Definition 2.4.10 (Independent set)**.** A subset of $A \subseteq \Omega$ is an *independent* set in the digraph $X = (\Omega, E)$ if A contains no edges, i. e., $E \cap (A \times A) = \emptyset$. Note that an independent set cannot contain a self-adjacent vertex (a vertex with a self-loop).

The following observation will be used directly in Case 3a2 in Section 9.7 and indirectly through Cor. 2.4.12 below.

Proposition 2.4.11 (Independent sets in biregular digraphs)**.** *Let $X = (\Omega, E)$ be a non-empty biregular digraph. Then X has no independent set of size greater than $n/2$ where $n = |\Omega|$.*

Proof. Let $d > 0$ be the out-degree (and therefore the in-degree) of each vertex. Let $A \subseteq \Omega$ be an independent set. Then $\Omega \setminus A$ has to absorb all edges emanating from A, so $d(n - |A|) \geq d|A|$. Now $d > 0$ since X is non-empty, hence $|A| \leq n/2$ follows. $\square$

The following corollary will be used in item 2b2 of the algorithm described in Section 13.2.

Corollary 2.4.12 (Symmetry defect of biregular digraphs)**.** *The symmetry defect of any nontrivial irreflexive biregular digraph is* $\geq 1/2$.

Proof. Let $X = (\Omega, E)$ be the digraph in question. Let $A \subseteq \Omega$ be a set of twins. So $\mathrm{Aut}(X) \geq \mathfrak{S}(A)$. Then A is either an independent set in X, or independent in the irreflexive complement of X. In each case, Prop. 2.4.11 guarantees that $|A| \leq n/2$. $\square$

Definition 2.4.13 (Graph)**.** By a *graph* $X = (\Omega, E)$ we mean an irreflexive digraph where the relation E is symmetric ($E = E^-$). The *degree* of a vertex is its common in- and out-degree. X is *regular* of degree k if each vertex has degree k.

Definition 2.4.14 (Strongly regular graphs)**.** A graph $X = (\Omega, E)$ is *strongly regular* (SR) with parameters $(n, k, \lambda . \mu)$ if it has n vertices, it is regular of degree k, every pair of adjacent vertices has λ common neighbors and every pair of distinct, non-adjacent vertices has μ common neighbors.

Examples 2.4.15. The pentagon is SR with parameters $(5, 2, 0, 1)$. Petersen's graph is SR with parameters $(10, 3, 0, 1)$. An n-clique is SR with parameters $(n, n-1, n-2, *)$, where $*$ can be any number.

Definition 2.4.16 (Symmetrization)**.** Let $X = (\Omega, E)$ be a digraph. The *symmetrization* of X is the digraph $\widetilde{X} = (\Omega, E \cup E^-)$.

Note that the symmetrization of an irreflexive digraph is a graph.

Definition 2.4.17 (Walks, strong components)**.** A *walk* of length $t \geq 0$ in the digraph $X = (\Omega, E)$ is a sequence $(u_0, \dots, u_t)$ of vertices such that $(\forall i \geq 1)((u_{i-1}, u_i) \in E)$. We say that this walk starts at u_0 and ends at u_t. We say that vertex v is *accessible* from vertex u if there exists a walk that starts at u and ends at v. We say that u and v are mutually accessible if each is accessible from the other. Mutual accessibility is an equivalence relation on Ω; its equivalence classes are called the *strong components* of X. We say that X is *strongly connected* if every vertex is accessible from every vertex, i. e., there is just one strong component.

Notation 2.4.18. Let $X = (\Omega, E)$ be a digraph and $A, B \subseteq \Omega$. Set $E(A, B) = E \cap (A \times B)$.

The following characterization of strong connectedness is well-known.

Definition 2.4.19 (Cut)**.** A *cut* (A, B) of a digraph $X = (\Omega, E)$ is an ordered pair of nonempty sets $A, B \subseteq \Omega$ where $B = \Omega \setminus A$.

Proposition 2.4.20 (Cut characterization)**.** *The digraph* $X = (\Omega, E)$ *is strongly connected if and only if for every cut* (A, B) *we have* $E(A, B) \neq \emptyset$. $\square$

Definition 2.4.21 (Weak components)**.** A *weak walk* of length t in the digraph $X = (\Omega, E)$ is a walk of length t in its symmetrization $\widetilde{X}$. We say that vertex v is *weakly accessible* from vertex u if v is accessible from u in $\widetilde{X}$. The *weak components* of X are the (strong) components of $\widetilde{X}$. We say that $\widetilde{X}$ is *weakly connected* if $\widetilde{X}$ is (strongly) connected.

Definition 2.4.22 (Eulerian digraph)**.** A digraph $X = (\Omega, E)$ is *eulerian* if $(\forall u \in \Omega)(\deg^+(u) = \deg^-(u))$.

Note that a biregular digraph is necessarily eulerian; the converse is not true.

The following well-known fact will be important to the structure theory of classical coherent configurations.

Proposition 2.4.23 (Weak is strong)**.** *The weak components of an eulerian digraph are its strong components.*

For completeness we sketch a proof.

Lemma 2.4.24 (Cuts in Eulerian digraphs)**.** *Let $X = (\Omega, E)$ be an eulerian digraph. Then for every cut (A, B) we have $|E(A, B)| = |E(B, A)|$.*

Proof. $|E(A, B)| - |E(B, A)| = \sum_{u \in A} (\deg^+(u) - \deg^-(u)) = 0$. □

Proof of Prop. 2.4.23. We need to show that a weakly connected eulerian digraph is strongly connected. Let $C \subseteq \Omega$ be a weak component. Suppose for a contradiction that the induced subgraph $X[C]$ is not strongly connected. Then by Prop. 2.4.20 there exists a cut (A, B) (where $A \dot\cup B = C$) such that $E(A, B) = \emptyset$. Consequently, by Lemma 2.4.24, $E(B, A) = \emptyset$ and therefore $\widetilde{X}[C]$ is disconnected, a contradiction. □

2.4.2 Bipartite graphs, semiregularity, equitable partition

We use the term *"bipartite graph"* to denote a triple of the form $X = (A, B; E)$ where $E \subseteq A \times B$. So, in our terminology, a bipartite graph is a digraph with the vertex set split into two parts, A and B, with all edges pointing from A to B. By the "degree" of vertices in A we mean their out-degree, and for vertices in B their in-degree. We say that X is *semiregular* if every vertex in A has the same degree and every vertex in B has the same degree. The *trivial* bipartite graphs are the *empty* ($E = \emptyset$) and *complete* ($E = A \times B$) bipartite graphs.

The *bipartite complement* of $X = (A, B; E)$ is $X^c = (A, B; (A \times B) \setminus E)$.

The *density* of X is defined as $d(X) = |E|/(|A| \cdot |B|$; this quantity is between 0 and 1. We have $d(X) + d(X^c) = 1$.

Definition 2.4.25 (Induced bipartite subgraph)**.** Let $X = (\Omega, E)$ be a digraph and A, B disjoint subsets of Ω. We define the *induced bipartite subgraph* $X[A, B]$ as $X[A, B] = (A, B; E(A, B))$.

Here $E(A, B) = E \cap (A \times B)$ (Notation 2.4.18).

Definition 2.4.26 (Equitable partition, coloring)**.** Let $X = (\Omega, E)$ be a digraph and $\Omega = \Omega_1 \dot\cup \ldots \dot\cup \Omega_k$ be a partition of the vertex set. We say that this partition is *equitable* if

(a) each induced subgraph $X[\Omega_i]$ is biregular;

(b) each induced bipartite subgraph $X[\Omega_i, \Omega_j]$ $(i \neq j)$ is semiregular.

We say that the coloring $d : \Omega \to \mathscr{C}$ is equitable if the partition $\ker(d) = \{d^{-1}(i) \mid i \in \mathscr{C}\}$ is equitable.

Remark 2.4.27. We shall generalize this concept to configurations in Def. 3.1.9. Equitable partitions are the stable partitions under the naive refinement process (see Sec. 4.1). They play a central role in the analysis of coherent configurations (see Sec. 3.4.3).

Observation 2.4.28 (Twins in bipartite graphs)**.** Let $X = (\Omega_1, \Omega_2; E)$ be a bipartite graph and $x \neq y$ two non-isolated vertices. Then x and y are twins if and only if

(a) x, y belong to the same part Ω_i, and

(b) they have the same neighborhood in the other part: $\widetilde{X}(x) = \widetilde{X}(y)$ (where $\widetilde{X}$ denotes the symmetrization of X (see Def. 2.4.16)).

Definition 2.4.29 (Symmetricity and symmetry defect in bipartite graphs)**.** Let $X = (\Omega_1, \Omega_2; E)$ be a bipartite graph. Let T_i be a largest set of twins in Ω_i. We say that $|T_i|$ is the *absolute symmetricity* and $|\Omega_i \setminus T_i|$ is the *absolute symmetry defect* of Ω_i in X. Accordingly, $|T_i|/|\Omega_i|$ is the *relative symmetricity* and $1 - |T_i|/|\Omega_i|$ is the *relative symmetry defect* of Ω_i in X.

We now prove the bipartite analogue of Cor. 2.4.12.

Proposition 2.4.30 (Symmetry defect of semiregular bipartite graphs)**.** *The symmetry defect of each part of a nontrivial semiregular bipartite graph is* $\geq 1/2$.

Proof. Let $X = (\Omega_1, \Omega_2; E)$ be the bipartite graph in question. By taking the bipartite complement if necessary, we may assume that the density of X is $\leq 1/2$.

Let T be a set of twins in Ω_i. Let $x \in T$. Since X is nontrival, x has a neighbor $y \in \Omega_{3-i}$. But then $X(y) \supseteq T$ and therefore $|T| \leq \deg(y) \leq |\Omega_i|/2$. The reason of the last inequality is the density assumption. □

In Prop. 2.4.11 we showed that the largest independent set in a biregular digraph cannot have relative size greater than 1/2. We shall also need the bipartite analogue of this observation.

Proposition 2.4.31 (Independent sets in semiregular bipartite graphs)**.** *Let* $X = (\Omega_1, \Omega_2; E)$ *be a non-empty semiregular bipartite graph. Let* $A_i \subseteq \Omega_i$ $(i = 1, 2)$*. If* $A_1 \cup A_2$ *is an independent set in* X *then the relative sizes of the* A_i *add up to* ≤ 1*, i. e.,*

$$\frac{|A_1|}{|\Omega_1|} + \frac{|A_2|}{|\Omega_2|} \leq 1. \tag{13}$$

Proof. Let d_i denote the degree of the vertices of Ω_i in X. Since all edges emanating from A_1 are absorbed by $\Omega_2 \setminus A_2$, we have

$$d_1|A_1| \leq d_2(|\Omega_2| - |A_2|). \tag{14}$$

We have $d_1|\Omega_1| = d_2|\Omega_2| = |E|$. Let us divide both sides of Eq. (14) by $|E|$ by dividing the left-hand side by $d_1|\Omega_1|$ and the right-hand side by $d_2|\Omega_2|$. We obtain inequality (13). $\square$

The following corollary will be used in the analysis of Case 2 of the "block-design case" of the Split-or-Johnson routine (Sec. 9.7).

Corollary 2.4.32 (Trivial subgraphs in semiregular bipartite graphs)**.** *Let $X = (\Omega_1, \Omega_2; E)$ be a nontrivial semiregular bipartite graph. Let $A_i \subseteq \Omega_i$ $(i = 1, 2)$. If the induced bipartite subgraph $X[A_1, A_2]$ is trivial (empty or complete) then inequality* (13) *holds.*

Proof. Apply Prop. 2.4.31 to X and to its bipartite complement. $\square$

MORE TO BE WRITTEN ******

2.5 Hypergraphs

2.5.1 Basic terminology

A *hypergraph* $\mathcal{H} = (V, \mathcal{E})$ consists of a vertex set V and a multiset $\mathcal{E} = \{E_1, \ldots, E_m\}$ of hyperedges, where $E_i \subseteq V$. If there no multiple edges ($E_i = E_j \implies i = j$) we say that $\mathcal{H}$ is a *simple hypergraph.* In this case $\mathcal{E}$ can be viewed as a subset of the power-set of V.

We say that $\mathcal{H}$ is d*-uniform* if $|E_i| = d$ for all $i \leq m$.

We say that $\mathcal{H}$ is an *empty hypergraph* if $\mathcal{E} = \emptyset$. The *complete d-uniform hypergraph* is the simple hypergraph with edge set $\mathcal{E} = \binom{V}{d}$. The *trivial* d-uniform hypergraphs are the empty and the complete ones.

The *degree* of a vertex $x \in V$ is the number of indices i such that $x \in E_i$. The hypergraph $\mathcal{H}$ is r*-regular* if every vertex has degree r.

The *incidence graph* of the hypergraph $\mathcal{H}$ is the bipartite graph $X(\mathcal{H}) = ([m], V; I)$ where I is the incidence relation: $(i, x) \in [m] \times V$ belongs to I if $x \in E_i$. Two vertices $i, j \in [m]$ are twins in $X(\mathcal{H})$ exactly if $E_i = E_j$, so there are no twins in $[m]$ with respect to $X(\mathcal{H})$ if and only if $\mathcal{H}$ is simple.

Definition 2.5.1 (Induced subhypergraph)**.** For a subset $W \subseteq V$ we define the *induced subhypergraph* $\mathcal{H}[W]$ as follows: the vertex set of $\mathcal{H}[W]$ is W and $E_i \in \mathcal{E}$ is an edge of $\mathcal{H}[W]$ if and only if $E_i \subseteq W$.

In particular, every induced subhypergraph of a d-uniform hypergraph is d-uniform.

Definition 2.5.2 (Trace of hypergraph)**.** Let $\mathcal{H} = (V, \mathcal{E})$ be a hypergraph. The *trace* $\mathcal{E}_S$ on the set $S \subseteq V$ is the multiset $\{E \cap S \mid E \in \mathcal{E}\}$, and the trace of $\mathcal{H}$ is $\mathcal{H}_S = (S, \mathcal{E}_S)$.

We can treat a simple d-uniform hypergraph as a d-ary relational structure (V, R) with a symmetric relation $R \subseteq \Omega^d$, i.e., $(\forall \pi \in \mathfrak{S}_d)(R^\pi = R)$, with the additional condition that if $(x_1, \ldots, x_d) \in R$ then all the x_i are distinct.

Moreover, if $\mathcal{H}$ is not simple, we can still treat $\mathcal{H}$ as a d-ary relational structure $(V; R_1, \ldots, R_t)$ where R_i corresponds to the d-subsets that occur with multiplicity i in $\mathcal{E}$.

So some results on d-ary relational structures apply to d-uniform hypergraphs. We shall in particular apply the Design Lemma (Theorem 8.1.2) to uniform hypergraphs.

2.5.2 Isomorphisms, twins, symmetricity and symmetry defect

Definition 2.5.3 (Isomorphism)**.** We use the following definition of *isomorphism* of hypergraphs $\mathcal{H}_1 = (V_1; \mathcal{E}_1)$ and $\mathcal{H}_2 = (V_2; \mathcal{E}_2)$. An isomorphism $\mathcal{H}_1 \to \mathcal{H}_2$ is a bijection $f : V_1 \to V_2$ such that for every subset $A \subseteq V_1$ the multiplicity of A in $\mathcal{E}_1$ is equal to the multiplicity of A^f in $\mathcal{E}_2$. In particular, $\mathrm{Aut}(\mathcal{H}) \leq \mathfrak{S}(V)$.

Observation 2.5.4. *Note that* $\mathrm{Aut}(\mathcal{H})$ *is the restriction of* $\mathrm{Aut}(X(\mathcal{H}))$ *to the set* V*. This restriction is an isomorphism if and only if* $\mathcal{H}$ *is simple.*

Observation 2.5.5. *With this definition we note that a simple d-uniform hypergraph $\mathcal{H} = (V, \mathcal{E})$ is trivial if and only if its automorphism group is* $\mathfrak{S}(V)$*.*

Definition 2.5.6 (Twins, symmetricity and symmetry defect)**.** The definition of *twins* in a hypergraph is implied by the definition of automorphisms (see Def. 2.3.16): $x, y \in V$ $(x \neq y)$ are twins in $\mathcal{H}$ if the transposition $\tau = (x, y)$ is an automorphism of $\mathcal{H}$. The symmetricity and the symmetry defect of $\mathcal{H}$ are defined as the corresponding parameters of its automorphism group (Def. 2.3.13).

Remark 2.5.7. Note that twins in $\mathcal{H}$ are not necessarily twins in the incidence graph $X(\mathcal{H})$. For instance, if $\mathcal{H}$ is the complete d-uniform hypergraph where $1 \leq d < |V|$ then all vertices are twins in $\mathcal{H}$ but there are no twins in $X(\mathcal{H})$.

Observation 2.5.8. Candidate isomorphisms (and therefore automorphisms) of explicit[6] hypergraphs can be tested in polynomial time (cf. Prop. 2.3.6). In particular, one can find the twin equivalence classes of an explicit hypergraph in polynomial time. Consequently, the symmetry defect of a hypergraph can also be computed in polynomial time. (A similar observation was made regarding explicit relational structures, Obs. 2.3.17).

2.5.3 Skeletons

The "Skeleton defect lemma" (Lemma 2.5.12) below will play an important role in the analysis of the Split-or-Johnson routine (see Section 9.7, Case 3b).

Definition 2.5.9. The t-skeleton of the hypergraph $\mathcal{H} = (V, \mathcal{E})$ is the t-uniform simple hypergraph $\mathcal{H}^{(t)} = (V, \mathcal{E}^{(t)})$ where $F \in \binom{V}{t}$ belongs to $\mathcal{E}^{(t)}$ exactly if there exists $E \in \mathcal{E}$ such that $F \subseteq E$.

[6] See the comment after Prop. 2.3.6 for the concept of "explicitness."

Proposition 2.5.10. *Let $\mathcal{H}$ be a nontrivial d-uniform simple hypergraph with n vertices and m edges, where $d \le n/2$. Then there exists $t \le \min\{d, 1 + \log_2 m\}$ such that the t-skeleton $\mathcal{H}^{(t)}$ is nontrivial.*

Proof. Choose $t = d$ if $d \le 1 + \log_2 m$. Otherwise let $t = 1 + \lfloor \log_2 m \rfloor$. Let $x_1, \dots, x_t$ be independently uniformly selected vertices of $\mathcal{H}$. The probability that all of them belong to an edge $E \in \mathcal{E}$ is $(|E|/n)^t \le 1/2^t$. The probability that there exists an edge to which all the x_i belong is less than $m/2^t$ which is less than 1 if $t > \log_2 m$. So $\mathcal{H}^{(t)}$ is not complete. It is also not empty since $t \le d$. $\square$

Proposition 2.5.11. *Let $\mathcal{H} = (V, \mathcal{E})$ be a nonempty, regular, d-uniform hypergraph. Let $S \subseteq V$. Let $\alpha = |S|/|V|$. Then there is an edge $E_i \in \mathcal{E}$ such that $|E_i \cap S| \ge \alpha d$.*

Proof. Let $|V| = n$ and $|\mathcal{E}| = m$. Each vertex belongs to md/n edges, so for each vertex x, the probability that $x \in E$ for a randomly selected edge is d/n. Therefore the expected number of vertices in $|S \cap E_i|$ for a random $i \in [m]$ is $|S|d/n = \alpha d$. $\square$

Lemma 2.5.12 (Skeleton defect lemma)**.** *Let $\mathcal{H} = (V, \mathcal{E})$ be a nontrivial, regular, d-uniform simple hypergraph with n vertices and m edges where $d \le n/2$. Let $(7/4)\log_2 m \le t \le (3/4)d$. Then the symmetry defect of the t-skeleton $\mathcal{H}^{(t)}$ is greater than $1/4$.*

Proof. Let $S \subseteq V$ be a set of twins in $\mathcal{H}$. Assume for a contradiction that $|S| \ge 3n/4$. Then, by Prop. 2.5.11, there is an edge $E \in \mathcal{E}$ such that $|S \cap E| \ge (3/4)d \ge t$. Let $T \subseteq S \cap E$, $|T| = t$. So $T \in \mathcal{E}^{(t)}$. Since S is a symmetrical set, it follows that $\binom{S}{t} \subseteq \mathcal{E}^{(t)}$. Since every edge of $\mathcal{H}$ contains at most $\binom{d}{t}$ of these t-sets, it follows that

$$m \ge \frac{\binom{|S|}{t}}{\binom{d}{t}} > \left(\frac{3n/4}{d}\right)^t \ge \left(\frac{3}{2}\right)^t > m, \tag{15}$$

a contradiction. $\square$

CHAPTER 1: COMBINATORICS

In this chapter we build our combinatorial tools and present the combinatorial partitioning algorithms.

3 Classical coherent configurations

Coherent configurations were first introduced by Schur [Sch] in the context of his study of permutation groups with a regular subgroup.

ADD HISTORY ********

3.1 The definition

3.1.1 Configurations

Definition 3.1.1 (Partition structure)**.** By a (binary) *partition structure* we mean a binary relational structure $\mathfrak{X} = (\Omega; R_1, \dots, R_r)$ where the sets $R_i \subseteq \Omega \times \Omega$ are not empty and they partition $\Omega \times \Omega$:

$$\Omega \times \Omega = R_1 \dot{\cup} R_2 \dot{\cup} \dots \dot{\cup} R_r. \tag{16}$$

This is equivalent to coloring the edges of the complete reflexive digraph, i. e., a function $c : \Omega \times \Omega \to [r]$; here $c(x, y) = i$ exactly if $(x, y) \in R_i$. We refer to $c(x, y)$ as the *color* of the "edge" (x, y). (We refer to all pairs of vertices as "edges," namely, the edges of the complete reflexive digraph.) We call the digraph $X_i = (\Omega(i), R_i)$ the *color-i constituent* digraph of $\mathfrak{X}$, where $\Omega(i)$ denotes the set of non-isolated vertices of the digraph (Ω, R_i). The *extended color-i constitutent* is (Ω, R_i) (we do not ignore the isolated vertices). We shall refer to R_i as a constituent relation of $\mathfrak{X}$. Unless there is a possibility of confusion, we shall refer to both X_i and R_i as constituents of $\mathfrak{X}$.

For $x \in \Omega$ we write $\deg_i^+(x)$ to denote the out-degree of x in extended constituent (Ω, R_i). We define $\deg^-(x)$ analogously. We call r the *rank* of $\mathfrak{X}$; if $|\Omega| \geq 2$ then $r \geq 2$. We shall often use the alternative notation $\mathfrak{X} = (\Omega, c)$ to denote this partition structure, implying that $R_i = c^{-1}(i)$.

We shall intechangeably use the notation $\mathfrak{X} = (\Omega, c)$ and $\mathfrak{X} = (\Omega; R_1, \dots, R_r)$ for partition structures, each notation implying the other, i. e., $R_i = c^{-1}(i)$ is implied if c is given, and $c(x, y) = i$ is implied if $(x, y) \in R_i$ where the R_i are given.

Definition 3.1.2 (Configuration)**.** We say that the partition structure $\mathfrak{X} = (\Omega, c) = (\Omega; R_1, \dots, R_r)$ is a *(binary) configuration* if

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

(ii) $(\forall u, v, x, y \in \Omega)(c(u, v) = c(x, y) \implies c(v, u) = c(y, x))$.

Terminology 3.1.3. Axiom (i) says that $(\forall i)$(either $R_i \subseteq \operatorname{diag}(\Omega)$ or $R_i \cap \operatorname{diag}(\Omega) = \emptyset$). In the former case we say that i is a *diagonal color* and X_i is a *diagonal constituent*; in the latter case we speak of an off-diagonal color and constituent.

Axiom (ii) says that $c(x, y)$ determines $c(y, x)$, i. e., $(\forall i)(\exists j)(R_j = R_i^-)$. We write $j = i^-$ if $R_j = R_i^-$. If i is an off-diagonal color and $i^- = i$ (i. e., $R_i^- = R_i$) then we say that the color i and the constituent X_i are *undirected*; otherwise (in this case $R_i \cap R_i^- = \emptyset$), we say that i and X_i are *oriented*.

Definition 3.1.4 (Vertex colors)**.** Let $\mathfrak{X} = (\Omega, c)$ be a configuration. We view the diagonal colors as a coloring of the vertices, setting $c(x) := c(x, x)$. We write $\Omega_i = \{x \in \Omega \mid c(x) = i\}$. If there are s vertex-colors, we may assume they form the set $[s]$, i. e., we have the partition $\Omega = \bigcup_{i=1}^s \Omega_i$ into the vertex-color classes Ω_i.

Definition 3.1.5 (Stable set)**.** A subset $\Delta \subseteq \Omega$ is *stable* if it is the union of some vertex-color classes.

Definition 3.1.6 (Homogeneous configuration)**.** The configuration $\mathfrak{X} = (\Omega, c)$ is *homogeneous* if all vertices have the same color, i. e., $R_1 = \mathrm{diag}(\Omega)$.

Example 3.1.7. A *graph*, and more generally, an irreflexive digraph $X = (\Omega, E)$ can be viewed as a (homogeneous) configuration $\mathfrak{X}(X) = (\Omega; \mathrm{diag}(\Omega), E, \overline{E})$ where $(V, \overline{E})$ is the complement of X.

The configuration $\mathfrak{X}(X)$ has rank 3 unless X is trivial (empty or complete) (empty relations are omitted), in which case it has rank 2.

Definition 3.1.8 (Clique configuration)**.** The *clique configuration* on Ω ($|\Omega| \geq 2$) is the unique rank-2 configuration on Ω (corresponding to the clique graph).

We defined equitable partitions and colorings for digraphs (Def. 2.4.26). We extend the definition to configurations.

Definition 3.1.9 (Equitable partition, coloring)**.** Let $\mathfrak{X} = (\Omega; R_1, \ldots, R_r)$ be a configuration and $\Pi = \{\Delta_1, \ldots, \Delta_k\}$ a partition of Ω, i. e., $\Omega = \Delta_1 \dot\cup \ldots \dot\cup \Delta_k$. We say that Π is an *equitable partition* of $\mathfrak{X}$ if Π is equitable for each of the extended constitutents (Ω, R_i). We say that the coloring $d : \Omega \to \mathscr{C}$ is equitable if the partition $\ker(d)$ is equitable.

Observation 3.1.10. *An equitable coloring d of the configuration $\mathfrak{X} = (\Omega, c)$ is a refinement of the coloring of $\mathfrak{X}$, i. e., $(\forall x, y \in \Omega)(d(x) = d(y) \implies c(x) = c(y))$.*

Proof. Let $c(x) = i$ and $d(x) = d(y) = h$. Let $D = d^{-1}(h)$; so $x, y \in D$. Consider the induced subgraph $Y = R_i[D]$. We have $\deg^+_Y(x) = 1$ so by equitability $\deg^+_Y(y) = 1$, meaning that $c(y) = i$. □

3.1.2 Coherent configurations, intersection numbers

Definition 3.1.11. A *(classical) coherent configuration* of rank r is a binary configuration $\mathfrak{X} = (\Omega, c)$ of rank r satisfying the following additional axiom.

(iii) There exists a family of r^3 nonnegative integers p_{ij}^k ($1 \leq i, j, k \leq r$) such that

$$(\forall i, j, k \leq r)(\forall x, y \in \Omega)(c(x, y) = k \implies |\{z \mid c(x, z) = i \text{ and } c(z, y) = j\}| = p_{ij}^k)\,. \tag{17}$$

The p_{ij}^k are called the *intersection numbers* of $\mathfrak{X}$.

Coherent configurations are the configurations fixed by the classical Weisfeiler-Leman canonical refinement process [WeL, We], see Sec. 4.2.

3.1.3 Orbitals, Schurian coherent configurations

Let $G \leq \mathfrak{S}(\Omega)$ be a permutation group. An *orbital* of G is an orbit of the G-action on $\Omega \times \Omega$.

Observation 3.1.12 (Orbital configuration)**.** Let $R_1, \ldots, R_r$ denote the orbitals of G. Then $\mathfrak{X}(G) := (\Omega; R_1, \ldots, R_r)$ is a coherent configuration. □

We say that a coherent configuration is *Schurian* if it is the orbital configuration of some permutation group.

Observation 3.1.13. $$G \leq \mathrm{Aut}(\mathfrak{X}(G))$$ □

Remark 3.1.14. Not every coherent configuration is Schurian; in fact, there are large families of strongly regular graphs with no non-identity automorphisms (line graphs of Steiner triple systems, point graphs of Latin squares [Ba80, Cam80]).

Observation 3.1.15. The orbits of $G \leq \mathfrak{S}(\Omega)$ are the vertex-color classes of $\mathfrak{X}(G)$. In particular, $\mathfrak{X}(G)$ is homogeneous if and only if G is transitive.

3.2 Important classes of homogeneous coherent configurations

3.2.1 Primitive and uniprimitive coherent configurations

Recall that $\mathfrak{X}$ is *homogeneous* if all vertices have the same color.

Definition 3.2.1 (UPCC)**.** $\mathfrak{X}$ is *primitive* if it is homogeneous and all constituents other than the diagonal are connected. $\mathfrak{X}$ is *uniprimitive* if it is primitive and has rank ≥ 3, i. e., it is not the clique configuration.

Notation 3.2.2. We abbreviate "uniprimitive coherent configuration" as UPCC.

Observation 3.2.3. Let $G \leq \mathfrak{S}(\Omega)$ be a permutation group. The orbital configuration $\mathfrak{X}(G)$ is primitive if and only if the group G is primitive. $\mathfrak{X}(G)$ is a clique configuration if and only if G is doubly transitive. Therefore $\mathfrak{X}(G)$ is a UPCC if and only if G is uniprimitive (primitive but not doubly transitive). □

A graph theoretic study of UPCCs was initiated by the author [Ba81] in 1980, with an immediate contribution by Viktor Zemlyachenko [ZKT] (see [Ba81, updated version]). This line of work reached great depth in a recent major paper by Sun and Wilmes [SuW] (2015).

UPCCs play an important role in the study of GI as the obstacles to natural combinatorial partitioning. One of the main technical contributions of this paper is that we overcome this obstacle at a logarithmic multiplicative cost (Section 9).

3.2.2 Association schemes, metric schemes, Johnson schemes

We say that the coherent configuration $\mathfrak{X}$ is an *association scheme* if $c(x, y) = c(y, x)$ for every $x, y \in \Omega$. It follows that association schemes are homogeneous.

Let $X = (\Omega, E)$ be connected undirected graph. The *distance-configuration* generated by X is the configuration $\mathcal{M}(X) = (\Omega, \mathrm{dist}_X)$ where $\mathrm{dist}_X(.,.)$ is the distance metric on X, i. e., $\mathrm{dist}_X(x, y)$ is the length of a shortest path between x and y in X. This configuration is necessarily homogeneous.

Observation 3.2.4. For graphs X, Y we have $\mathrm{Iso}(\mathcal{M}(X), \mathcal{M}(Y)) = \mathrm{Iso}(X, Y)$). In particular, $\mathrm{Aut}(\mathcal{M}(X)) = \mathrm{Aut}(X)$. □

Definition 3.2.5 (Distance-regular graph, metric scheme)**.** The connected undirected graph X is said to be *distance-regular* if $\mathcal{M}(X)$ is an association scheme; in this case we call $\mathcal{M}(X)$ the *metric scheme* generated by X.

Observation 3.2.6. The connected strongly regular graphs are precisely the distance-regular graphs of diameter ≤ 2. □

Definition 3.2.7. A connected undirected graph X is *distance-transitive* if for every pair $\{(x, y), (x', y')\}$ of pairs of vertices, if $\operatorname{dist}(x, y) = \operatorname{dist}(x', y')$ then

$$(\exists \sigma \in \operatorname{Aut}(X))(x^\sigma = x' \text{ and } y^\sigma = y').$$

Observation 3.2.8. A distance-transitive graph is necessarily distance-regular. □

The converse is false, as shown by large families of strongly regular graphs with no nontrivial automorphisms (see Remark 3.1.14).

A particularly important class of metric schemes arises from Johnson graphs (Def. 1.2.3). We slightly rephrase the definition.

Definition 3.2.9 (Johnson graph)**.** Let $t \geq 2$ and let Γ be set of size $|\Gamma| = k \geq 2t + 1$. The *Johnson graph* $J(\Gamma, t) = (\Omega; c)$ is an undirected graph with $\binom{k}{t}$ vertices corresponding to the t-subsets of a k-set Γ,

$$\Omega = \left\{ v_T \,\middle|\, T \in \binom{\Gamma}{t} \right\}.$$

For $S, T \in \binom{\Gamma}{t}$, the vertices v_S and v_T are adjacent if $|S \setminus T| = 1$. We write $J(k, t)$ for $J(\Gamma, t)$ if $\Gamma = [k]$ or we do not want to specify Γ.

Observation 3.2.10. Consider the Johnson graph $J(\Gamma, t) = (\Omega, E)$. For $S, T \in \binom{\Gamma}{t}$, the distance between the vertices v_S and v_T is $|S \setminus T|$. □

An important functor (see Section 6) maps the category of k-sets Γ to the category of Johnson graphs $J(\Gamma, t)$. This functor is *surjective* (on $\operatorname{Iso}(\mathfrak{X}, \mathfrak{Y})$ for any pair $(\mathfrak{X}, \mathfrak{Y})$ of objects). The principal content of this nontrivial statement is the following.

Proposition 3.2.11. *Let us identify the vertex set of the Johnson graph $J(\Gamma, t)$ with $\binom{\Gamma}{t}$. Then* $\operatorname{Aut}(J(\Gamma, t)) = \mathfrak{S}^{(t)}(\Gamma)$.

This result motivates our term "Johnson groups" (see Sec. 1.2.1). The inclusion $\operatorname{Aut}(J(\Gamma, t)) \geq \mathfrak{S}^{(t)}(\Gamma)$ is straightforward. The opposite inclusion can be derived for instance from the Erdős–Ko–Rado theorem. □

The following is immediate from the trivial direction of Prop. 3.2.11.

Observation 3.2.12. The Johnson graphs are distance-transitive and therefore distance-regular.

Definition 3.2.13 (Johnson scheme)**.** A *Johnson scheme* is the metric scheme generated by a Johnson graph. In other words, let $t \geq 2$ and let Γ be set of size $|\Gamma| = k \geq 2t+1$. The *Johnson scheme* $\mathfrak{J}(\Gamma, t) = (\Omega; c)$ is an association scheme with $\binom{k}{t}$ vertices corresponding to the t-subsets of an k-set Γ,

$$\Omega = \left\{ v_T \;\middle|\; T \in \binom{\Gamma}{t} \right\}.$$

For $S, T \in \binom{\Gamma}{t}$, the color of the edge (v_S, v_T) is $c(v_S, v_T) = |S \setminus T|$. We write $\mathfrak{J}(k, t)$ for $\mathfrak{J}(\Gamma, t)$ if $\Gamma = [k]$ or we do not want to specify Γ.

Observation 3.2.14. The Johnson schemes are UPCCs. $\square$

Remark 3.2.15 (Degenerate Johnson schemes)**.** We may view the complete graph K_k as a degenerate Johnson graph $J(k, 1)$; and the clique configuration the corresponding degenerate Johnson scheme $\mathfrak{J}(k, 1)$. We excluded them from among the Johnson graphs/schemes for the sake of Obs. 3.2.14.

3.3 Basic combinatorial properties of coherent configurations

Convention 3.3.1. For the rest of Section 3, all results will tacitly refer to a (classical) **coherent configuration** $\mathfrak{X} = (\Omega, c) = (\Omega; R_1, \ldots, R_r)$ (where $[r]$ is the set of colors and $c : \Omega \times \Omega \to [r]$ is the coloring of pairs) except where explicitly stated otherwise.

Observation 3.3.2 (Stable is coherent)**.** If $\Delta \subseteq \Omega$ is a stable subset (see Def. 3.1.5) then the induced substructure $\mathfrak{X}[\Delta]$ is a coherent configuration. $\square$

Observation 3.3.3. For a graph X, the configuration $\mathfrak{X}(X)$ defined in Example 3.1.7 is coherent if and only if X is *strongly regular*, including the case when X is a clique. $\square$

In the rest of this section we refer to a coherent configuration $\mathfrak{X} = (\Omega, c) = (\Omega; R_1, \ldots, R_r)$.

Observation 3.3.4 (Vertex-color awareness)**.** The color of an edge determines the colors of its tail and head.

Proof. Assume $c(x, y) = c(x', y') = i$. We need to show that $c(x) = c(x')$ and $c(y) = c(y')$. Let $c(x) = \ell$, so $p_i^{\ell, i} \geq 1$. It follows that $(\exists z')(c(x', z') = \ell$ and $c(z', y') = i)$. But ℓ is a diagonal color, so $z' = x'$ and therefore $c(x') = c(x', x') = \ell$. So the color of the tail of the edge (x, y) is determined by $c(x, y)$. Now apply this fact to the color i^- to see that the color of the head is also determined. $\square$

Observation 3.3.5 (Degree awareness)**.** The color of a vertex determines its out- and in-degree in any given color.

Proof. Let $c(x) = \ell$. Then the out-degree of x in color i is p_ℓ^{i, i^-} and the in-degree of x in color i is $p_\ell^{i^-, i}$. $\square$

Notation 3.3.6. Let $R_j \subseteq \Omega_\ell \times \Omega_m$. We write $\deg_j^+$ to denote the value $\deg_j^+(x)$ for any (and therefore all) $x \in \Omega_\ell$ and $\deg_j^-$ for $\deg_j^-(y)$ for any (and therefore all) $y \in \Omega_m$.

Combining the two preceding observations, we infer a classification of the constituents.

Corollary 3.3.7 (Constituents: homogeneous or bipartite)**.** *For $k \leq r$, the constituent digraph $X_k = (\Omega(k), R_k)$ is either*

(i) (homogeneous case) a biregular digraph with vertex set $\Omega(k) = \Omega_i$ for some vertex color i (in particular, all vertices of X_k have the same color), or

(ii) (bipartite case) a semiregular bipartite graph of the form $(\Omega_i, \Omega_j; R_k)$ for some distinct vertex colors i and j (so $R_k \subseteq \Omega_i \times \Omega_j$; in particular, the vertices have two colors).

Proof. The first case arises when the tail and the head of an edge of color k have the same color, i; this is in particular the case when k is a diagonal color. The second when the color of the tail is i and the color of the head is $j \neq i$. Biregularity and semiregularity follow from Obs. 3.3.5. ☐

We can rephrase this corollary in terms of equitable colorings.

Corollary 3.3.8 (Vertex-coloring is equitable)**.** *Let $\mathfrak{X} = (\Omega, c)$ be a coherent configuration. Then the vertex coloring $c : \Omega \to [r]$ is equitable for $\mathfrak{X}$.* ☐

A stronger connection of coherent configurations with equitability follows in Sec. 3.4.3

In our discussion of walks, it will be convenient to use the language of *strings* in the classical sense (strings over $[n]$ for some non-negative integer n in the sense we used the term "string" in Sec. 11.1), including the "Kleene star" notation.

Notation 3.3.9 (Strings, Kleene star)**.** Let Σ be a finite alphabet. Σ^* denotes the set of strings (words, finite sequences) over Σ. The symbol Λ denotes the empty string. For strings $x, y \in \Sigma^*$, we write xy for the concatenation of x and y. If $s \in \Sigma$ the we also write s to denote the string of length one consisting of s. In particular, if $x \in \Sigma^*$ and $s \in \Sigma$ then xs denotes the string x with the letter s appended. For subsets $L, M \subseteq \Sigma^*$, we write $LM = \{xy \mid x \in L, y \in M$. For $k \geq 2$ we define L^k inductively as $L^k = L^{k-1}L$. We write $L^1 = L$ and $L^0 = \{\Lambda\}$. Finally, L^* is defined as $L^* = \bigcup_{k=0}^{\infty} L^k$.

Definition 3.3.10 (Walks)**.** Let $I = i_1 \dots i_t$ be a finite string of colors. A walk of length t of color composition I from vertex x to vertex y is a sequence $(u_0, \dots, u_t)$ of vertices such that $(\forall j \geq 1)(c(u_{i-1}, u_i) = i_j)$, where $u_0 = x$ and $u_t = y$.

Proposition 3.3.11 (Counting walks)**.** *For every string $I = i_1 \dots i_t$ of colors and color k there exists a number $p(I, k)$ such that for any $x, y \in \Omega$ satisfying $c(x, y) = k$, the number of walks of length t of color composition I from x to y is $p(I, k)$.*

Proof. For $t = 0$ we have $p(\Lambda, k) = 1$ if k is a diagonal color and 0 otherwise.
For $t = 1$ we have $p(i_1, k) = 1$ if $i_1 = k$ and 0 otherwise.
For $t = 2$ we have $p(i_1 i_2, k) = p_k^{i_1, i_2}$.
For the inductive step we observe that for $t \geq 2$ we have

$$p(i_1 \dots i_t, k) = \sum_{\ell \leq r} p(i_1 \dots i_{t-2}\ell, k) p_\ell^{i_{t-1}, i_t} \tag{18}$$

☐

Definition 3.3.12 (Accessibility along strings of colors)**.** Given a (not necessarily finite) set $\mathcal{S} \subseteq [r]^*$ of finite strings of colors (strings over the alphabet $[r]$), we say that vertex y is accessible from vertex x along $\mathcal{S}$ if $(\exists w \in \mathcal{S})(y$ is accessible from x along a walk of color composition $w)$. We write $\mathcal{S}(x)$ to denote the set of vertices accessible from x along $\mathcal{S}$.

Corollary 3.3.13 (Accessibility set)**.** *Given a (not necessarily finite) set $\mathcal{S} \subseteq [r]^*$ of finite strings of colors there exists a set $J \subseteq [r]$ of colors such that $(\forall x, y \in \Omega)(y \in \mathcal{S}(x) \iff c(x,y) \in J)$. We write $J = J(\mathcal{S})$ and call this the* accessibility set *of $\mathcal{S}$.*

Proof. Define J by letting $j \in J \iff (\exists w \in \mathcal{S})(p(wj) \neq 0)$ where the function p is defined in Prop. 3.3.11. □

We are ready to derive a corollary that will be used multiple times.

Theorem 3.3.14 (Accessibility)**.** *Let $\mathcal{S} \subseteq [r]^*$ be a (not necessarily finite) set of finite strings of colors. Let $x, y \in \Omega$ have the same color: $c(x) = c(y)$. Then $|\mathcal{S}(x)| = |\mathcal{S}(y)|$.*

Proof. Let $i = c(x) = c(y)$. Let $J = J(\mathcal{S})$ be the accessibility set of $\mathcal{S}$. Then $|\mathcal{S}(x)| = \sum_{j \in J} \deg_j^+(x)$. Let $J' \subseteq J$ consist of those $j \in J$ for which the tail of an edge of color j has color i (see Obs. 3.3.4). So $|\mathcal{S}(x)| = \sum_{j \in J'} \deg_j^+ = |\mathcal{S}(y)|$ by the Degree-awareness lemma (Obs. 3.3.5). □

3.4 Toward the analysis of combinatorial partitioning

In this section we further study the structure of coherent configurations with the aim to develop combinatorial tools for the analysis of the Split-or-Johnson procedure (Sec. 9). The "Large clique lemma" (Lemma 3.4.25) will also be the central tool in the analysis of the Design Lemma (Sec. 8).

3.4.1 Connected components of constituents

Proposition 3.4.1 (Weak is strong)**.** *The weak components of a homogeneous constitutent digraph are its strong components.*

Proof. Homogeneous constituents are eulerian by part (i) of Cor. 3.3.7. Therefore their weakly connected components are strongly connected by Prop. 2.4.23. □

Remark 3.4.2. The proof shows that the same holds if we consider a weak component of a union of homogeneous constituents.

Definition 3.4.3 (Equipartition)**.** An *equipartition* of a set Ω is a partition of Ω into blocks of equal size.

Proposition 3.4.4 (Homogeneous connected components)**.** *If X_k is a homogeneous constituent digraph in color class Ω_i then the components of X_k equipartition Ω_i.*

Proof. Let $x \in \Omega_i$ and let $Y(x)$ denote the set of vertices of the component of the constituent X_k that contains x. Then $Y(x) \subseteq \Omega_i$ and $Y(x)$ consists of those vertices that are accessible from x along a sequence of edges of color k. Using the notation of Cor. 3.3.12, this means that $Y(x) = (k^*)(x)$ where $k^* = \{\Lambda, k, kk, kkk, \dots\}$ (set of strings of the color k). It follows by the Accessibility Theorem (Thm. 3.3.14) that $|Y(x)| = |(k^*)(x)|$ only depends on k and $c(x)$. $\square$

Proposition 3.4.5 (Bipartite connected components)**.** *If $X_k = (\Omega(k), R_k)$ is a bipartite constituent digraph with $R_k \subseteq \Omega_i \times \Omega_j$ then the weak components of X_k equipartition each of the two color classes, Ω_i and Ω_j. In particular, all weak components of X_k have the same number of vertices.*

Proof. Let $x \in \Omega_i$ and let $Y(x)$ denote the set of vertices of the weak component of the constituent X_k that contains x. Then $Y(x) \cap \Omega_i$ consists of those vertices accessible from x along a string of colors in the set $(kk^-)^* = \{\Lambda, kk^-, kk^-kk^-, \dots\}$. Using the notation of Cor. 3.3.12, this means that $Y(x) = ((kk^-)^*)(x)$. It follows by the Accessibility Theorem (Thm. 3.3.14) that $|Y(x) \cap \Omega_i| = |((kk^-)^*)(x)|$ only depends on k and $c(x)$.

The case $x \in \Omega_j$ follows by applying the foregoing to k^- in the place of k (swapping the roles of i and j). $\square$

While most of the material so far in Section 3 (except perhaps the Accessibility Theorem) is probably folklore (although we could not find a convenient reference), the Contraction Theorem (Thm. 3.4.9 below) does not seem to have been stated. It will be used in the justification of a subroutine in the Split-or-Johnson routine, see Lemma 9.6.2. We start with three preliminary lemmas.

Lemma 3.4.6 (Neighborhood of connected component of constituent)**.** *Let Ω_1 and Ω_2 be two distinct vertex-color classes; let $i = c(z)$ for $z \in \Omega_i$ $(i = 1, 2)$. Let $B_1, \dots, B_m$ be the connected components of the homogeneous constituent digraph $X_3 = (\Omega_1, R_3)$ in Ω_1 and let $X_4 = (\Omega_1, \Omega_2; R_4)$ be a bipartite constituent between Ω_1 and Ω_2 (so $R_4 \subseteq \Omega_1 \times \Omega_2$). For $j = 1, \dots, m$ let M_j denote the set of R_4-out-neighbors of B_j, i. e., the set of vertices $v \in \Omega_2$ such that there exists $w \in B_j$ such that $c(w, v) = 4$. Then $|M_1| = \cdots = |M_m|$.*

Proof. Let $x \in B_j$. Then $y \in M_j \iff y$ is accessible from x along 3^*4. Using the notation of Cor. 3.3.12, this means that $M_j = (3^*4)(x)$. It follows by the Accessibility Theorem (Thm. 3.3.14) that $|M_j| = |(3^*4)(x)|$ only depends on the colors $3, 4$, and $c(x) = 1$, which are given; so it is the same number of all $x \in \Omega_1$, it does not depend on j. $\square$

Lemma 3.4.7. *Using the notation of Lemma 3.4.6, let $x \in \Omega_1$ and $y \in \Omega_2$ be joined by an edge of color $c(x, y) = 4$. Assume $x \in B_i$. Let $M(x, y) = \{z \in B_i \mid c(z, y) = 4\}$. Then $|M(x, y)|$ does not depend on the choice of x and y (and in particular on i).*

Proof. Let J be the set of those colors j for which the head of an edge of color j is accessible from its tail along edges of color 3. Now $z \in M(x, y) \iff c(x, z) \in J$ and $c(z, y) = 4$. Therefore $|M(x, y)| = \sum_{j \in J} p_4^{j,3}$. $\square$

Lemma 3.4.8. *Using the notation of Lemma 3.4.6, for $y \in \Omega_2$ let $E(y)$ denote the set of those i for which there exists $x \in B_i$ satisfying $c(x,y) = 4$. Then $|E(y)|$ does not depend on y.*

Proof. Let $q = |M(x,y)| > 0$ be the quantity shown not to depend on x, y in Lemma 3.4.7 (as long as $c(x,y) = 4$). Recall that $c(y) = 2$. Now, $p^2_{4^-,4} = \deg_4^-(y) = q|E(y)|$. $\square$

The next result states that if we contract the connected components of a homogeneous color-class, then bipartite color-classes remain semiregular.

Theorem 3.4.9 (Contraction)**.** *Using the notation of Lemma 3.4.6, let Y be the bipartite graph $Y = ([m], \Omega_2; E)$ where $(i, y) \in E$ if $(\exists x \in B_i)(c(x,y) = 4)$. Then Y is semiregular.*

Proof. Regularity on the $[m]$ side is the content of Lemma 3.4.6. Regularity on the Ω_2 side is the content of Lemma 3.4.8. $\square$

3.4.2 Twin awareness

Theorem 3.4.11 below provides a critical tool for the analysis of the Split-or-Johnson routine (Sec. 9.4).

Lemma 3.4.10 (Twin awareness 1)**.** *Let Ω_i, Ω_j be two distinct vertex-color classes in the coherent configuration $\mathfrak{X} = (\Omega, c)$. Consider the bipartite constituent $X_k = (\Omega(k), R_k)$ where $R_k \subseteq \Omega_i \times \Omega_j$. Then for all pairs $x, y \in \Omega_i$, $x \neq y$, the color $c(x,y)$ determines whether x, y are twins in X_k.*

Proof. Let $c(x,y) = \ell$. Now x, y are twins in X_k if and only if $R_k(x) = R_k(y)$. The latter is equivalent to saying that $p^\ell_{k,k^-} = p^i_{k,k^-}$. This equality depends only on the colors involved. $\square$

Theorem 3.4.11 (No twins in primitive color)**.** *Let Ω_i, Ω_j be two distinct vertex-color classes in the coherent configuration $\mathfrak{X} = (\Omega, c)$. Consider the bipartite constituent $X_k = (\Omega(k), R_k)$ where $R_k \subseteq \Omega_i \times \Omega_j$. Assume X_k is non-trivial (non-empty and not complete). Assume further that the induced subconfiguration $\mathfrak{X}[\Omega_i]$ is primitive. Then there are no X_k-twins in Ω_i.*

Proof. Assume for a contradiction that $x, y \in \Omega_i$ ($x \neq y$) are twins for X_k. Let $c(x,y) = \ell$. By Lemma 3.4.10 it follows that every pair $(x', y') \in R_\ell$ are twins. Now the twin-or-equal relation is an equivalence relation, so the transitive closure of R_ℓ is a subset of the twin-or-equal relation. But $\mathfrak{X}[\Omega_i]$ is primitive, so R_ℓ is (strongly) connected and therefore its transitive closure is $\Omega_i \times \Omega_i$.

We have shown that Ω_i is a single twin equivalence class. This means the set $R_\ell(x)$ is the same for all $x \in \Omega_i$; let us call this set $W \subseteq \Omega_j$. Now $W \neq \emptyset$ and $W \neq \Omega_j$ since in either of these cases, X_k would be trivial. Let $u \in W$ and $v \in \Omega_j \setminus W$. Then $\deg_k^-(u) = |\Omega_i| \neq 0$ and $\deg_k^-(v) = 0$, contradicting the semiregulary of X_k. $\square$

We mention for completeness that Lemma 3.4.10 holds for homogeneous constitutents as well, permitting a significant generalization of the result (Theorem 3.4.14), although we shall not use this fact, so the reader may skip the remainder of Sec. 3.4.2.

Lemma 3.4.12 (Twin awareness 2)**.** *Let Ω_i be a vertex-color class in the coherent configuration $\mathfrak{X} = (\Omega, c)$. Consider the homogeneous constituent digraph $X_k = (\Omega_i, R_k)$ where $R_k \subseteq \Omega_i \times \Omega_i$. Then for all pairs $x, y \in \Omega_i$, $x \neq y$, the color $c(x, y)$ determines whether x, y are twins in X_k.*

Proof. Let $c(x, y) = \ell$.
Case 1. $\quad \ell \notin \{k, k^-\}$.
In this case, x, y are twins in X_k if and only if $R_k(x) = R_k(y)$ which is again equivalent to saying that $p^\ell_{k,k^-} = p^i_{k,k^-}$.
Case 2. $\quad \ell \in \{k, k^-\}$.
In this case, if $k \neq k^-$ then x and y are not twins. Assume now $k = k^- = \ell$. In this case, x, y are twins in X_k if and only if $R_k(x) \setminus \{y\} = R_k(y) \setminus \{x\}$ which is equivalent to saying that $p^k_{k,k} = p^i_{k,k} - 1$. □

Lemma 3.4.13. *Let $k \in [r]$ be a color. Then for all pairs $x, y \in \Omega$, $x \neq y$, the color $c(x, y)$ determines whether x, y are twins in the extended constituent (Ω, R_k).*

Proof. If $c(x) \neq c(y)$ then x, y are not twins; and this relation only depends on $c(x, y)$ by vertex-color awareness (Obs. 3.3.4). Assume now that $c(x) = c(y) = i$. If $\Omega_i \not\subseteq \Omega(k)$ then x, y are twins in (Ω, R_k) because they are isolated. Finally assume $\Omega_i \subseteq \Omega(k)$. In this case, x, y are twins in the extended constituent (Ω, R_k) if and only if they are twins in the constituent X_k. So if X_k is homogeneous, the result follows from Lemma 3.4.12; and if X_k is bipartite, the result follows by applying Lemma 3.4.10 to k or to k^-. □

Theorem 3.4.14 (Twin awareness 3)**.** *Let $K \subseteq [r]$ be a set of colors. Consider the configuration $\mathfrak{Y} = \left(\Omega, \bigcup_{k \in K} R_k\right)$. Then for all pairs $x, y \in \Omega$, $x \neq y$, the color $c(x, y)$ determines whether x, y are twins in $\mathfrak{Y}$.*

Proof. x, y are twins in $\mathfrak{Y}$ if and only if they are twins in the extended constituent (Ω, R_k) for each $k \in K$. So the result follows from Lemma 3.4.13. □

3.4.3 Local constituents

Theorem 3.4.19 below is the key ingredient of the analysis of the UPCC case (Case (iii) of step 9 of *Procedure Bipartite Split-or-Johnson* (Sec. 9.4). It fixes the error found by Harald Helfgott on January 1, 2017, and also eliminates a previously separate case (Johnson scheme). The analysis is described in Sec. 9.8.

Definition 3.4.15 (Local coloring)**.** Let $\mathfrak{X} = (\Omega, c)$ be a configuration and $x \in \Omega$ a vertex. The x-local coloring c_x of Ω is defined as $c_x(y) = c(x, y)$ for $y \in \Omega$.

Observation 3.4.16. *If $\mathfrak{X}$ is a coherent configuration then the local coloring c_x is a refinement of the coloring c of Ω, i. e., if $(\forall y, z \in \Omega)(c_x(y) = c_x(z) \implies c(y) = c(z))$.*

Proof. Immediate from vertex-color awareness (Obs. 3.3.4). □

Proposition 3.4.17 (Equitability of local colorings)**.** *Each local coloring of a coherent configuration is equitable.*

Proof. Let $x \in \Omega$ and consider the x-local coloring c_x. The color classes are the sets $R_i(x)$ for $i \in [r]$. Consider the extended constituent $\widehat{X}_k = (\Omega, R_k)$. We need to prove that c_x is an equitable coloring for $\widehat{X}_k$. Let $\ell, m \in [r]$ be (not necessarily distinct) colors. We need to show that for $u \in R_\ell(x)$, the quantities $|R_k(u) \cap R_m(x)|$ and $|R_{k^-}(u) \cap R_m(x)|$ do not depend on the particular choice of u, only on the colors k, ℓ, m. Indeed, $|R_k(u) \cap R_m(x)| = p^\ell_{m,k^-}$ and $|R_{k^-}(u) \cap R_m(x)| = p^\ell_{m,k}$. □

Definition 3.4.18 (Local constituents)**.** Let $\mathfrak{X} = (\Omega, c)$ be a configuration, $x \in \Omega$ a vertex, and $k, \ell, m \in [r]$ colors; $X_k = (\Omega(k), R_k)$ is the color-k constituent of $\mathfrak{X}$. We define the x-(k, ℓ, m)-local constituent Y of $\mathfrak{X}$. If $\ell = m$ then we define Y as the induced subgraph $X_k[R_\ell(x)]$. If $\ell \neq m$ then we define Y as the induced bipartite subgraph $X_k[R_\ell(x), R_m(x)]$.

Theorem 3.4.19 (Nontriviality of local constituent)**.** *Let $\mathfrak{X} = (\Omega, c)$ be a coherent configuration. Let i, j be diagonal colors, $\ell \neq m$ off-diagonal colors, and $x \in \Omega_i$ such that $R_\ell(x) \subseteq \Omega_i$ and $R_m(x) \subseteq \Omega_j$. Assume $|\Omega_j|/2 < |R_m(x)| < |\Omega_j|$. Assume further that the induced subconfiguration $\mathfrak{X}[\Omega_i]$ is primitive. Then the x-(m, ℓ, m)-local constituent $Y = X_m[R_\ell(x), R_m(x)]$ is nontrivial.*

Proof. Y is semiregular by Prop. 3.4.17. Let $u \in R_\ell(x)$; so $u \in \Omega_i$. Therefore $\deg^+_m = |R_m(u)| = |R_m(x)| > |\Omega_j|/2$, hence $R_m(x) \cap R_m(u) \neq \emptyset$. Let $w \in R_m(x) \cap R_m(u)$. That means $(u, w) \in R_m$ hence Y is not empty.

What we need to prove is that Y is not complete. Suppose for a contradiction that it is. So for $u \in R_\ell(x)$ we have $R_m(u) \supseteq R_m(x)$. But $\deg^+_m = |R_m(u)| = |R_m(x)|$ and therefore $R_m(u) = R_m(x)$. This means u and x are twins, contradicting the "no twins in primitive color" theorem (Theorem 3.4.11), given our assumption that $\mathfrak{X}[\Omega_i]$ is primitive. □

3.4.4 Bipartite configurations, sections, links, bihomogeneous coherent configurations

Definition 3.4.20 (Bipartite configuration)**.** By a (binary) *bipartite configuration* $\mathfrak{X} = (\Omega_1, \Omega_2; R_1, \dots, R_t)$ we mean a binary relational structure on the disjoint union $\Omega = \Omega_1 \dot\cup \Omega_2$ such that the relations R_i are non-empty and they partition $\Omega_1 \times \Omega_2$:

$$\Omega_1 \times \Omega_2 = R_1 \dot\cup R_2 \dot\cup \dots \dot\cup R_t. \tag{19}$$

This is equivalent to coloring the edges of the complete bipartite graph $(\Omega_1, \Omega_2; \Omega_1 \times \Omega_2)$, i. e., a function $c : \Omega_1 \times \Omega_2 \to [t]$; here $c(x, y) = i$ exactly if $(x, y) \in R_i$. We call the bipartite graph $X_i = (\Omega_1, \Omega_2; R_i)$ the *color-i constituent* of $\mathfrak{X}$. We say that the bipartite configuration $\mathfrak{X}$ is *trivial* if $t = 1$ (there is just one color).

If we reverse every edge, we obtain the bipartite configuration $\mathfrak{X}^- = (\Omega_2, \Omega_1; R_1^-, \dots, R_t^-)$.

Definition 3.4.21 (Induced bipartite subconfiguration)**.** Let $\mathfrak{X} = (\Omega; R_1, \dots, R_r)$ be a configuration. Let A and B be disjoint subsets of Ω and let $K = \{k \in [r] \mid R_k \cap (A \times B) \neq \emptyset\}$. We define the *induced bipartite subconfiguration* $\mathfrak{X}[A, B]$ as the bipartite configuration

$$\mathfrak{X}[A, B] = (A, B; R_k \cap (A \times B) \mid k \in K). \tag{20}$$

We say that R_k (or the color k) is *involved* in $\mathfrak{X}[A, B]$ if $k \in K$.

Let $\mathfrak{X} = (\Omega; c)$ be a coherent configuration with vertex-color classes $\Omega_1, \ldots, \Omega_s$. We call the induced homogeneous coherent configuration $\mathfrak{X}[\Omega_i]$ the *homogeneous section* of $\mathfrak{X}$ in color i.

Let $i \neq j$ be two vertex colors. We call the induced bipartite subconfiguration $\mathfrak{X}[\Omega_i, \Omega_j]$ the *link* between the two homogeneous sections $\mathfrak{X}[\Omega_i]$ and $\mathfrak{X}[\Omega_j]$, or the link between vertex colors i and j.

Observation 3.4.22. *Let $\mathfrak{X}$ be a coherent configuration. The color k is involved in the link between vertex colors i and j if and only if $R_k \subseteq \Omega_i \times \Omega_j$. The link is trivial exactly if $(\exists k \in [r])(R_k = \Omega_i \times \Omega_j)$.* □

Notation 3.4.23. Let $\mathfrak{X}$ be a coherent configuration. If it causes no confusion, we write $\mathfrak{X}_i = \mathfrak{X}[\Omega_i]$ for the homogeneous section of color i and $\mathfrak{X}_{ij} = \mathfrak{X}[\Omega_i, \Omega_j]$ for the link between colors i and j.

Definition 3.4.24 (Bihomogeneous coherent configuration)**.** We say that the coherent configuration is *bihomogeneous* if it has two vertex-color classes.

Much of the Split-or-Johnson routine will depend on the study of bihomogeneous coherent configurations.

3.4.5 Large clique lemma

The following lemma will be at the heart of the proof of the Design Lemma. The lemma asserts that if in a coherent configuration, the largest vertex-color class C is unique and it induces a clique then C is a twin equivalence class. The proof is based on Fisher's inequality for block designs.

Lemma 3.4.25 (Large clique lemma)**.** *Let $\mathfrak{X} = (\Omega, c)$ be a coherent configuration. Let $\Omega_1, \ldots, \Omega_s$ be the vertex-color classes. Assume $|\Omega_1| > |\Omega_i|$ for all $i \geq 2$ and Ω_1 induces a clique configuration in $\mathfrak{X}$. Then Ω_1 is a twin equivalence class in $\mathfrak{X}$. In particular, all links between color 1 and the other vertex colors are trivial. Moreover, the symmetry defect of $\mathfrak{X}$ is $1 - |\Omega_1|/|\Omega|$.*

We recall Fisher's inequality for BIBDs (balanced incomplete block designs).

Definition 3.4.26. A possibly degenerate BIBD with parameters (v, b, r, k, λ) is an r-regular k-uniform hypergraph with v vertices and b not necessarily distinct edges such that each pair of vertices belongs to exactly λ "blocks" (edges) where $k, r \geq 1$ and $k < v$; the latter is the "incompleteness" condition.

A BIBD is a possibly degerenate BIBD satisfying $\lambda \geq 1$.

Note that $v \geq 1$ (in fact, $v \geq 2$ because $v > k \geq 1$) and therefore $b \geq r \geq 1$.

For a degenerate BIBD we have $\lambda = 0$, so each block is a singleton, i.e., $k = 1$ and therefore $b = rv \geq v$.

Theorem 3.4.27 (Fisher's inequality)**.** *For a possibly degenerate BIBD with parameters (v, b, r, k, λ) we have $b \geq v$.*

Remark 3.4.28. Fisher's inequality is usually stated for BIBDs; however, as mentioned above, the degenerate case also satisfies the conclusion.

Proof of Lemma 3.4.25. Let $C = \Omega_1$. First we prove that all links between C and the other colors is trivial. Given that $\mathfrak{X}[C]$ is a clique, this immediately implies that C is a twin equivalence class in $\mathfrak{X}$.

We need to prove that for all $x \in \Omega \setminus C$ and all $y, z \in C$ we have $c(x,y) = c(x,z)$. For a contradiction assume this is false and let ℓ be a color and $x \in \Omega \setminus C$ such that $c(x,y) = \ell$ for some but not all $y \in C$. In other words, $1 \le \deg_\ell^+ < |C|$.

Let $j = c(x)$, so x belongs to the vertex-color class $B := \Omega_j$. Note that $B \cap C = \emptyset$ and $|B| < |C|$.

Recall that $R_\ell(u)$ denotes the set of out-neighbors of vertex u in color ℓ. For $u \in B$ we have $R_\ell(u) \subseteq C$. Moreover, $|R_\ell(u)| = \deg_\ell^+$ does not depend on the choice of $u \in B$. So the hypergraph

$$\mathcal{H} = (C, \{R_\ell(u) \mid u \in B\}) \tag{21}$$

is k-uniform with $k = \deg_\ell^+$. Moreover, $1 \le k \le |C| - 1$ by the choice of ℓ.

For $v, w \in C$, $v \neq w$, let $m = c(v,w)$. Note that m does not depend on the choice of v and w since C induces a clique in $\mathfrak{X}$.

For $v \in C$, let $B_v = R_{\ell^-}(v)$. Note that $B_v \subseteq B$ and $|B_v| = \deg_{\ell^-}^+$ does not depend on the choice of $v \in C$, hence the hypergraph $\mathcal{H}$ is r-regular with $r = \deg_{\ell^-}^+$. Moveover, for $v, w \in C$, $v \neq w$ we have $|B_v \cap B_w| = p_{\ell^-,\ell}^m$. This quantity does not depend on the choice of the pair (v,w), hence $\mathcal{H}$ is a possibly degenerate BIBD with $\lambda = p_{\ell^-,\ell}^m$.

The number of vertices of this design is $|C|$ and the number of (not necessarily distinct) blocks is $|B|$. Hence, by Fisher's inequality, $|C| \le |B|$, a contradiction, proving all but the last statement in Lemma 3.4.25.

Regarding the last statement, about the symmetry defect, we note that C is the largest twin equivalence class in $\mathfrak{X}$ since by definition, twins have the same color. $\square$

4 Individualization and canonical refinement – ADD DETAILS

Let $\mathfrak{X}$ be a structure such as a graph, digraph, k-ary relational structure, hypergraph, with colored elements (vertices, edges, k-tuples, hyperedges). The colors form an ordered list. A *refinement* of the coloring c is a new coloring c' of the same elements such that if $c'(x) = c'(y)$ for elements x, y then $c(x) = c(y)$; this results in the refined structure $\mathfrak{X}'$. We say that the refinement is *canonical* with respect to a set $\{\mathfrak{X}_i \mid i \in I\}$ of objects of the same type if it is executed simultaneously on each $\mathfrak{X}_i$ and for all $i, j \in I$ we have

$$\mathrm{Iso}(\mathfrak{X}_i', \mathfrak{X}_j') = \mathrm{Iso}(\mathfrak{X}_i, \mathfrak{X}_j). \tag{22}$$

(This is consistent with the functorial notion of canonicity explained in Sec. 6.) Naive vertex refinement (refine vertex colors by number of neighbors of each color) has been the basic isomorphism rejection heuristic for ages. More sophisticated canonical refinement methods are explained in the next section.

Another classical heuristic is *individualization:* the assignment of a unique color to an element. Let $\mathfrak{X}_x$ denote $\mathfrak{X}$ with the element x individualized. If the number of elements of the given type is m then individualization incurs a multiplicative cost of m: when testing isomorphism of structures $\mathfrak{X}$ and $\mathfrak{Y}$, if we individualize $x \in \mathfrak{X}$, we need compare $\mathfrak{X}_x$ with all $\mathfrak{Y}_y$ for $y \in \mathfrak{Y}$: for any $x \in \mathfrak{X}$ we have

$$\mathrm{Iso}(\mathfrak{X}, \mathfrak{Y}) = \bigcup_{y \in \mathfrak{Y}} \mathrm{Iso}(\mathfrak{X}_x, \mathfrak{Y}_y). \tag{23}$$

(Compare this with the more general categorical concept in Sec. 6.)

The individualization/refinement method (I/R) (individualization followed by refinement) is a powerful heuristic and has also been used to proven advantage (see e.g., [Ba79a, Ba81, BaL, BaCo, BaW1, ChST, BaCh+]), even though strong limitations of its isomorphism rejection capacity have also been proven [CaiFI]. I/R combines well with the group theory method and the combination is not subject to the CFI limitations ([Ba79a, BaL, BaCo, BaCh+]). The power of this combination is explored in this paper.

4.1 Naive vertex-refinement

Let $X = (V, E, c)$ be a vertex-colored digraph without loops, i.e., $E \subseteq (V \times V) \setminus \mathrm{diag}(V)$ and $c : V \to \mathscr{C}$ is a vertex coloring. Recall that the set $\mathscr{C}$ of colors is an ordered set. (The exclusion of loops is not a severe restriction; we can replace loops by encoding them into the colors, doubling the number of colors.)

For vertex $x \in V$ and each color $i \in \mathscr{C}$, let $d_i^+(x)$ be the number of out-neighbors of x of color i and $d_i^-(x)$ be the number of in-neighbors of x of color i. Let $c'(x) = (c(x), d_i^+(x), d_i^-(x) \mid i \in \mathscr{C})$, where the terms in the string $c'(x)$ are arranged in the order defined by the ordering of $\mathscr{C}$.

We regard the set $\mathscr{C}' = \{c'(x) \mid x \in V\}$ of strings as a new set of colors, ordered lexicographically. Clearly, the coloring c' is a refinement of c. Let $X' = (V, E, c')$.

The $c \mapsto c'$ refinement constitutes one round of the *naive vertex-refinement process.*

Let $\Pi(c)$ denote the equivalence relation on V defined by c, i.e., $(x, y) \in \Pi(c)$ if $c(x) = c(y)$. Let $N(c)$ denote the number of colors used by c.

We say that X is *stable* with respect to naive vertex-refinement if $\Pi(c') = \Pi(c)$ (no proper refinement is obtained). This is equivalent to saying that $N(c') = N(c)$.

The following is immediate.

Observation 4.1.1. The vertex-colored digraph $X = (V, E, c)$ is stable with respect to naive vertex-refinement if and only if the partition $\Pi(c)$ is equitable (see Def. 2.4.26). $\square$

We now describe the *naive vertex-refinement process.* Let $X_0 = X$ and $X_{j+1} = X_j'$. We write $X_j = (V, E, c_j)$, so $c_{j+1} = c_j'$. We stop when X_k is stable, i.e., when $N(c_{k+1}) = N(c_k)$. We call the colored digraph $X^* = X_k$ the *stable refinement* of X. This is the output of the naive refinement process.

The next observation asserts canonicity of the procedure.

Observation 4.1.2. If X, Y are vertex-colored digraphs then

$$\mathrm{Iso}(X, Y) = \mathrm{Iso}(X', Y') = \mathrm{Iso}(X^*, Y^*) \tag{24}$$

This method can be extended to configurations.

4.1.1 Complexity of naive refinement; tagged structures

The naive refinement process clearly terminates in $k \leq n-1$ rounds where $n = |V|$ is the number of vertices.

Let Σ be a finite alphabet that includes $[n]$ and the three separator symbols: comma, opening and closing paretheses. If each color in X is described by a string of length k over Σ then colors in X' are described by strings of length $k + O(n)$. Therefore the description of each color in X_j has length $k + O(jn)$ and therefore in X^*, the length is $k + O(n^2)$. The cost of round j is $O(m)$ comparison of colors, where $m = |E|$, so the total cost of round j is $O(jnm)$. The overall total cost is $O(nk + n^2 m)$.

This large cost is due to the growth of the strings describing the colors. This can be avoided by reducing, in each round, the set of colors to an initial segment of the integers (specifically to $[N(c_j)]$ in round j) and remembering this substitution of colors. More formally, add a *tag* (a string that will represent the color substitution history) to each object we consider. Next we describe this formally.

Our objects are *tagged colored digraphs* (without loops), i. e., quadruples $X = (V, E, c, t)$ where $E \subseteq (V \times V) \setminus \mathrm{diag}(V)$, $c : V \to \mathscr{C}$ is a coloring, and t is the "tag" (a string). For two tagged colored digraphs X and $Y = (W, F, d, u)$ we set

$$\mathrm{Iso}(X, Y) = \begin{cases} \mathrm{Iso}((V, E, c), (W, F, d)) & \text{if } u = t \\ \emptyset & \text{if } u \neq t \end{cases} \tag{25}$$

We say that the coloring c is in *standard form* if $\mathscr{C} = [N(c)]$ where $N(c)$ is the size of the range of c. By *standardizing* the coloring $c : V \to \mathscr{C}$ we mean replacing c by the coloring $\mathrm{st}(c) : V \to [N(c)]$ where $\mathrm{st}(c)(x) = i$ is $c(x)$ is the i-th element of the ordered set $\mathscr{C}$. By *standardizing the colored tagged digraph* we mean replacing $X = (V, E, c, t)$ by $\mathrm{st}(X) = (V, E, \mathrm{st}(c), t{*}\mathscr{C})$, so the new tag is the concatenation of the old tag with the ordered list of the old colors. The asterisk (a special symbol) serves to separate the concatenated items.

We now adapt the *naive refinement step* to tagged digraphs. Let $X = (V, E, c, t)$ be a tagged digraph. Let us define $X' = (V, E, d, u)$ as follows: let $\mathrm{st}(X) = (V, E, \mathrm{st}(c), u)$ and let $X' = (V, E, (\mathrm{st}(c))', u)$ where the coloring $\mathrm{st}(c)'$ is defined, as above, by the the application of one round of naive refinement to the untagged colored digraph $(V, E, \mathrm{st}(c))$. So the difference compared to the naive refinement step described above is that first we standardize the coloring and update the tag by appending the list of original colors; then perform one round of naive refinement as above, without changing the tag.

Now let $X_0 = X$ and $X_{j+1} = X'_j$. We write $X_j = (V, E, c_j, t_j)$. We stop when X_k is stable, i. e., when $N(c_{k+1}) = N(c_k)$. We write $X^* = \mathrm{st}(X_k)$ and call it the *stable refinement* of X. We write t^* for the final tag.

Each tag t_j is an isomorphism invariant, justifying the second line on the right-hand side of Eq. (25) (isomorphism rejection if the tags are not equal).

The role of the tag is that from each X_j we can reconstruct X, so Eq. 24 continues to hold.

Now the complexity analysis changes as follows. The length of tag t_1 is the length t_0 (typically 0) plus the size of the list of initial colors. For $j \geq 1$ the increment $|t_{j+1}| - |t_j|$ is $O(n^2)$, so the length of the final tag is $|t^*| = |t_1| + O(n^3)$. The cost of each refinement round is $O(m)$, total $O(nm) + |t_1|$. We note that t_1 is part of the input, so the cost item $|t_1|$ is linear in terms of the length of the input.

The $O(nm)$ term can be further reduced to $O(m \log n)$ as follows (Hopcroft–Tarjan [HoT]). Let $p_j(i)$ denote the level-$(j-1)$ *parent* of level-j color i, i. e., if $c_j(x) = i$ then $c_{j-1}(x) = p_j(i)$. Now in computing c_{j+1}, omit those terms $d_i^{\pm}(x)$ corresponding to colors i such that $|c_j^{-1}(i)| > (1/2)|c_{j-1}^{-1}(p(i))|$ (the color class did not shrink to half or less in the previous round). This way, in recomputing the colors, every vertex is visited at most $\log_2 n$ times.

****

TO BE WRITTEN

4.1.2 Splitting a semiregular bipartite graph – minor savings

In this section we use the following notation. Let $X = (\Omega_1, \Omega_2; E)$ be a bipartite graph, so $E \subseteq \Omega_1 \times \Omega_2$. Let $n_i = |\Omega_i|$.

This section provides a termination tool for the Split-or-Johnson process, to be used when n_2 is very small (logarithmic), see Sec. 9.4. This will only save an annoying $\log\log$ factor in the exponent, so the reader not interested is such fine estimation of the complexity may skip this section.

Observation 4.1.3. Let $X = (\Omega_1, \Omega_2; E)$ be a bipartite graph. If there are no twins in Ω_1 then individualizing each element of Ω_2 completely splits Ω_1. $\square$

Note that the multiplicative cost incurred is individualizing each element of of Ω_2 is $n_2! \approx \exp(n_2 \log n_2)$. If our goal is only to get a good partition, rather than a complete split, of Ω_1, the next observation allows us to save a factor of $\log n_2$ in the exponent.

Let $X = (\Omega_1, \Omega_2; E)$ be a bipartite graph and $f : \Omega_2 \to \mathscr{C}$ be a k-coloring of Ω_2. In this section we shall view such a coloring f as a $(k+1)$-coloring of the vertices of X by assigning Ω_1 a separate color.

Proposition 4.1.4. *Let $X = (\Omega_1, \Omega_2; E)$ be a nontrivial semiregular bipartite graph. Then there is a 3-coloring $f : \Omega_2 \to [3]$ such that naive refinement of the colored bipartite graph $(\Omega_1, \Omega_2; E, f)$ yields a $(1/2)$-coloring of Ω_1: each color-subclass of Ω_1 has size $\leq |\Omega_1|/2$.*

Note that the multiplicative cost incurred by selecting a 3-coloring of Ω_2 is 3^{n_2}.

Lemma 4.1.5. *Let $X = (\Omega_1, \Omega_2; E)$ be a nontrivial semiregular bipartite graph. Then there is a 2-coloring $g : \Omega_2 \to [2]$ such that the naive refinement of the colored bipartite graph $(\Omega_1, \Omega_2; E, g)$ yields a $(2/3)$-coloring of Ω_1: each color-class in Ω_1 will h ave size $\le 2|\Omega_1|/3$.*

Proof of Lemma 4.1.5. By complementing if necessary, we may assume that the density of X is $|E|/(n_1 n_2) \le 1/2$. Let $\deg_i$ denote the degree of the vertices in Ω_i; so $\deg_i \le n_{3-i}/2$.

Let us fix an ordering of $\Omega_2 = \{v_1, \ldots, v_{n_2}\}$. The *prefix* P_k in this ordering is the subset $P_k = \{v_1, \ldots, v_k\}$.

Claim 4.1.6. *There is a prefix P_k of which the neighborhood $X(P_k)$ has size $n_1/3 < |X(P_k)| \le 2n_1/3$.*

Proof. Let k be the smallest number such that $|X(P_k)| > n_1/3$. We claim that $|X(P_k)| \le 2n_1/3$. Indeed, $|X(P_k)| \le |X(P_{k-1})| + \deg_2 < n_2/3 + \deg_2$, so we are done if $\deg_2 \le n_2/3$. If $\deg_2 > n_2/3$ then we necessarily have $k = 1$ and again we are done. □

Let $g(x) = 1$ for $x \in P_k$ and $g(x) = 2$ for $x \in \Omega_2 \setminus P_k$.

Left h denote the naive refinement of $(\Omega_1, \Omega_2; E, g)$. The coloring h satisfies the prescriptions of Lemma 4.1.5. □

Proof of Prop. 4.1.4. Let g be as in Lemma 4.1.5 and let h denote the coloring of the vertices of X after naive refinement of $(\Omega_1, \Omega_2; E, g)$.

If each h-color-class in Ω_1 has size $\le n_1/2$ then we can set $f := g$, there is no need for a third color.

Assume now that there is a dominant h-color class $\Gamma \subseteq \Omega_1$, so $|\Gamma| > n_1/2$. Since for any $u \in P_k$ and $v \notin P_k$ we have $g(u) \ne g(v)$, it follows that either $\Gamma \subseteq X(P_k)$ or either $\Gamma \subseteq \Omega_1 \setminus X(P_k)$; in either case, $|\Gamma| \le 2n_1/3$.

Let $(x, y) \in E$ such that $x \in \Gamma$. Let Δ denote the h-color-class of y (so $\Delta \subseteq \Omega_2$). Let $Y = X[\Gamma, \Delta]$ be the bipartite induced subgraph of X on (Γ, Δ).

We claim that Y is nontrivial. It is not empty since (x, y) is an edge of Y. It is not complete since $\deg(y) \le n_1/2 < |\Gamma|$.

Y is semiregular since h is an equitable coloring. Let us now apply Lemma 4.1.5 to Y. This yields a 2-coloring $g' : \Delta \to \mathscr{C}$ that after naive refinement gives a $(2/3)$-coloring h' of Γ, so each h'-color class in Γ will have size $\le 2|\Gamma|/3 \le 4n_1/9$.

Let the g-color of Δ be $i \in \{1, 2\}$. Let the 2-coloring g' use the colors $\{i, 3\}$. We now combine g and g' to a 3-coloring $f : \Omega_2 \to [3]$ as follows. For $u \in \Omega_2$ let

$$f(u) = \begin{cases} g(u) & \text{if } u \notin \Delta \\ g'(u) & \text{if } u \in \Delta \end{cases}$$

Let h^* denote the refinement of the coloring of $(\Omega_1, \Omega_2; E, f)$. Notice that h^* is a refinement of h because f is a refinement of g. It follows that each h^*-color-class is either a subset of Γ or a subset of $\Omega \setminus \Gamma$. The latter have size $< n_1/2$ because $|\Gamma| > n_1/2$. Moreover, on Γ, the coloring h^* is a refinement of h', so the h^*-color-classes inside Γ will have size $\le 4n_1/9 < n_1/2$. □

4.2 Weisfeiler-Leman canonical refinement

4.3 Classical WL refinement

The classical Weisfeiler–Leman[7] (WL) refinement [WeL, We] takes as input a binary configuration and refines it to a coherent configuration (see Sec. 3.2.1) as follows. The process proceeds in rounds. Let $\mathfrak{X}$ be the input to a round of refinement. For $(x, y) \in \Omega \times \Omega$, we encode in the new color $c'(x, y)$ the following information: the old color $c(x, y)$, and for all $j, k \leq r$, the number $|\{z \in \Omega \mid c(x, z) = j \text{ and } c(z, y) = k\}|$. These data form a list, naturally ordered. To each list we assign a new color; these colors are sorted lexicographically. This gives a refined coloring that defines a new configuration $\mathfrak{X}'$. We stop when we reach a stable configuration ($\mathfrak{X} = \mathfrak{X}'$, i. e., no refinement occurs, i. e., no R_i is split).

Observation 4.3.1. The stable configurations under WL refinement are precisely the coherent configurations.

The process is clearly *canonical* in the following sense. Let $\mathfrak{X}$ and $\mathfrak{Y}$ be configurations. We simultaneously execute each round of refinement (merging the lists of refined colors). Let $\mathfrak{X}^*$ and $\mathfrak{Y}^*$ be the coherent configurations obtained. Then

$$\mathrm{Iso}(\mathfrak{X}, \mathfrak{Y}) = \mathrm{Iso}(\mathfrak{X}^*, \mathfrak{Y}^*). \tag{26}$$

In particular, if one of the colors of $\mathfrak{X}^*$ does not occur in $\mathfrak{Y}^*$ then $\mathfrak{X}$ and $\mathfrak{Y}$ are not isomorphic, so WL gives an isomorphism rejection tool.

****

5 Higher coherent configurations

5.1 k-ary partition structures, k-ary configurations

5.1.1 Notation: strings

As before, Ω will denote a fixed set of n elements. We shall refer to Ω as the "underlying set" or the set of "vertices."

We write Ω^k to denote the set of strings of length k over Ω; so $|\Omega^k| = n^k$. We write strings as $\vec{x} = x_1 \dots x_k \in \Omega^k$. On rare occasions we also write $\vec{x} = (x_1, \dots, x_k)$ to denote the same string if the omission of the commas might cause confusion.

For $\vec{x} \in \Omega^k$ and $\vec{y} \in \Omega^\ell$ we write $\vec{x}\vec{y} \in \Omega^{k+\ell}$ to denote the concatenation of the strings $\vec{x}$ and $\vec{y}$. We denote the empty string by Λ, so $\Omega^0 = \{\Lambda\}$. We denote the *length* of the string $\vec{x}$ by $\ell(\vec{x}\,)$, so if $\vec{x} \in \Omega^k$ then $\ell(\vec{x}\,) = k$. The *support* of the string $\vec{x} = x_1 \dots x_k \in \Omega^k$ is the set $\mathrm{supp}(\vec{x}\,) = \{x_1, \dots, x_k\} \subseteq \Omega$; so $|\mathrm{supp}(\vec{x}\,)| \leq \ell(\vec{x}\,)$.

We denote the set of strings of length not greater than k by $\Omega^{\leq k} = \bigcup_{0 \leq \ell \leq k} \Omega^\ell$.

[7] Weisfeiler's book [We] transliterates Leman's name from the original Russian as "Lehman." However, Andreĭ Leman (1940–2012) himself omitted the "h." (Sources: private communications by Mikhail Klin, Aug. 2006, and by Ilya Ponomarenko, Jan. 2016. Both Klin and Ponomarenko forwarded to me email messages they had received in the late 1990s from Leman. The "From" line of each message reads "`From: Andrew Leman <andyleman@etc.>`," and Leman also verbally expressed this preference.)

Notation 5.1.1 (Tree of strings)**.** For $k \geq 1$ and $\vec{x} = x_1 \dots x_k \in \Omega^k$ let $p(\vec{x}\,) = x_1 \dots x_{k-1} \in \Omega^{k-1}$ be the "parent" of $\vec{x}$. The parent links define a rooted tree structure on $\Omega^{\leq k}$, rooted at Λ.

Definition 5.1.2 (Prefix)**.** For $\ell \leq k$, the *prefix* of length ℓ of the string $x_1 \dots x_k \in \Omega^k$ is the substring $x_1 \dots x_\ell \in \Omega^\ell$.

Observe that $\vec{y} \in \Omega^\ell$ is a prefix of $\vec{x} \in \Omega^k$ if and only if $\vec{y}$ is a prefix of $\vec{x}$.

Notation 5.1.3 (Strings of distinct elements)**.** We write

$$\Omega^{\langle k \rangle} = \{x_1 \dots x_k \in \Omega^k \mid \text{ all the } x_i \text{ are distinct } \} \tag{27}$$

to denote the subset of Ω^k consisting of the $n(n-1)\dots(n-k+1)$ strings of length k of distinct elements of Ω. We let $\Omega^{\langle \leq k \rangle} = \bigcup_{\ell \leq k} \Omega^{\langle \ell \rangle}$.

Generally we write operators in the exponent, so if $f : \Omega \to \Omega'$ is a function ("domain transformation") then we denote the f-image of $x \in \Omega$ by x^f. This is consistent with the convention we use to evaluate composition of operators left to right, so $x^{fg} = (x^f)^g$.

Notation 5.1.4 (Induced maps)**.** For a map $f : \Omega \to \Omega'$ and a string $\vec{x} = x_1 \dots x_k \in \Omega^k$ we write $\vec{x}^f = x_1^f \dots x_k^f$.

Notation 5.1.5 (Symmetric monoid)**.** For a set Δ, we write $\mathfrak{M}(\Delta)$ to denote the *symmetric monoid* over Δ; so $\mathfrak{M}(\Delta)$ consists of all $\Delta \to \Delta$ maps.

We denote the set $\{1, \dots, k\}$ by $[k]$ and set $\mathfrak{M}_k := \mathfrak{M}[k]$. The symmetric monoid $\mathfrak{M}_k$ induces an action on Ω^k by acting on the subscripts ("index transformation"): for a map $\tau \in \mathfrak{M}_k$ and a string $\vec{x} = x_1 \dots x_k \in \Omega^k$ we write $\tau(\vec{x}\,) = x_{1^\tau} \dots x_{k^\tau}$. We make this exception to writing operators in the exponent to avoid writing different kinds of operators in the exponent. This has the usual unfortunate side-effect: $(\tau\mu)(\vec{x}\,) = \mu(\tau(\vec{x}\,))$. There is also a beneficial notational effect: the domain transformation and the index transformation operators commute; this will now appear as an associativity rule: for $f : \Omega \to \Omega'$ and $\tau \in \mathfrak{M}_k$ we have $\tau(\vec{x}^f) = (\tau(\vec{x}))^f$. (Another exception to exponential notation of operators will occur in Remark 5.1.9 as a corollary to this exception.)

Futher we note that

$$\operatorname{supp}(\tau(\vec{x})) \subseteq \operatorname{supp}(\vec{x}). \tag{28}$$

5.1.2 k-ary relational structures

Let $\mathscr{C}$ (the set of "colors") be a finite linearly ordered set. A *k-ary relation* on Ω is a subset $R \subseteq \Omega^k$. A *k-ary relational structure* or *k-ary structure* over the index set $\mathscr{C}$ is a pair $\mathfrak{X} = (\Omega, \mathcal{R})$ where $\mathcal{R} = (R_i : i \in \mathscr{C})$ where each R_i is a k-ary relation on Ω. An isomorphism between $\mathfrak{X}$ and $\mathfrak{X}' = (\Omega', \mathcal{R}')$ where $\mathcal{R}' = (R_i' : i \in \mathscr{C})$ is a bijection $f : \Omega \to \Omega'$ such that for all $i \in \mathscr{C}$ and all $\vec{x} \in \Omega^k$ we have $\vec{x} \in R_i$ if and only if $\vec{x}^f \in R_i'$. (Isomorphism can occur only when the two structures are indexed over the same set of colors.) The set of $\mathfrak{X} \to \mathfrak{X}'$ isomorphisms is denoted $\operatorname{Iso}(\mathfrak{X}, \mathfrak{X}')$. The automorphism group of $\mathfrak{X}$ is $\operatorname{Aut}(\mathfrak{X}) = \operatorname{Iso}(\mathfrak{X}, \mathfrak{X})$.

5.1.3 k-ary partition structures, coloring

We say that $\mathfrak{X}$ is a k-ary *partition structure* if the R_i partition Ω^k and none of the R_i is empty. There is a functor F_1, computable in time $|\mathscr{C}| \cdot n^{O(k)}$, that converts k-ary structures into k-ary partition structures without changing their underlying sets, such that $\mathrm{Iso}(\mathfrak{X}, \mathfrak{Y}) = \mathrm{Iso}(F_1(\mathfrak{X}), F_1(\mathfrak{Y}))$ for all pairs $(\mathfrak{X}, \mathfrak{Y})$ of k-ary relational structures. We define $F_1(\mathfrak{X})$ as follows. Assign to each $\vec{x} \in \Omega^k$ the color $c(\vec{x}\,) = \{i \in \mathscr{C} \mid \vec{x} \in R_i\}$. Let the new set $\mathscr{C}'$ of colors be the range of the function c, and define $F_1(\mathfrak{X}) = (\Omega, \mathcal{R}')$ where $\mathcal{R}' = \{R'_j \mid j \in \mathscr{C}'\}$ where $R'_j = \{\vec{x} \in \Omega^k \mid c(\vec{x}\,) = j\}$. So if $\mathfrak{X}$ is indexed over $m = |\mathscr{C}|$ colors then $F_1(\mathfrak{X})$ will be indexed over $|\mathscr{C}'| \le \max\{n^k, 2^m\}$ colors. The linear ordering of $\mathscr{C}$ induces the lexicographic ordering of $\mathscr{C}'$.

Henceforth we assume that $\mathfrak{X}$ is a k-ary partition structure. For $\vec{x} \in \Omega^k$ we write $c(\vec{x}\,) = i$ if $\vec{x} \in R_i$; in this case we refer to i as "the color" of $\vec{x}$. We shall alternatively denote $\mathfrak{X} = (\Omega, \mathcal{R})$ by $\mathfrak{X} = (\Omega, c)$ since $\mathcal{R}$ can be uniquely reconstructed from c.

Let $\Phi \subseteq \Omega$. The *induced substructure* (Φ, c_Φ) is defined by letting c_Φ be the restriction of the coloring c to Φ^k.

5.1.4 Skeleton, extended coloring

For $1 \le \ell \le k$ we define an embedding $\mathrm{pad} : \Omega^\ell \to \Omega^k$ that will allow us to extend the coloring of Ω^k to $\Omega^{\le k}$.

Definition 5.1.6 (Padding)**.** Fix a set Ω and an integer $k \ge 1$. For $1 \le \ell \le k$ and $\vec{x} = x_1 \dots x_\ell \in \Omega^\ell$ let $\mathrm{pad}(\vec{x}) = x_1 \dots x_k \in \Omega^k$ where for $j > \ell$ we set $x_j := x_\ell$.

Definition 5.1.7 (Skeleton)**.** Let $\mathfrak{X} = (\Omega, c)$ be a k-ary partition structure. For $1 \le \ell \le k$ we define the coloring $c^{(\ell)} : \Omega^\ell \to \mathscr{C}$ by setting, for $\vec{x} \in \Omega^\ell$,

$$c^{(\ell)}(\vec{x}\,) := c(\mathrm{pad}(\vec{x}\,)). \tag{29}$$

We define the *ℓ-skeleton* of $\mathfrak{X}$ as $\mathfrak{X}^{(\ell)} = (\Omega, c^{(\ell)})$.

Note that $\mathfrak{X}^{(k)} = \mathfrak{X}$.

Clearly, $\mathfrak{X}^{(\ell)}$ is an ℓ-ary partition structure and the assignment $\mathfrak{X} \mapsto \mathfrak{X}^{(\ell)}$ defines a (forgetful) functor; $\mathrm{Iso}(\mathfrak{X}^{(\ell)}, \mathfrak{Y}^{(\ell)}) \supseteq \mathrm{Iso}(\mathfrak{X}, \mathfrak{Y})$.

For the empty string Λ we reserve a special color $c^{(0)}(\Lambda)$ since the convention above does not assign Λ a color.

With some abuse of notation, for $\vec{x} \in \Omega^{\le k}$ we shall write $c(\vec{x})$ to denote $c^{(\ell(\vec{x}))}(\vec{x})$ whenever this does not lead to ambiguity. This way we have extended the coloring c from Ω^k to $\Omega^{\le k}$.

5.1.5 k-ary configurations

A string $\vec{x} = x_1 \dots x_k \in \Omega^k$ defines an equivalence relation $\rho(\vec{x}\,)$ on $[k]$ as follows: $i \sim j$ if $x_i = x_j$.

Definition 5.1.8 (Configuration)**.** We say that the k-ary partition structure $\mathfrak{X} = (\Omega, c)$ is a *k-ary configuration* if the following two axioms hold. For all $\vec{x}, \vec{y} \in \Omega^k$ and $\tau \in \mathfrak{M}_k$, if $c(\vec{x}\,) = c(\vec{y}\,)$ then

(i) $\rho(\vec{x}\,) = \rho(\vec{y}\,)$;

(ii) $c(\tau(\vec{x}\,)) = c(\tau(\vec{y}\,))$.

In other words, the color of $\vec{x}$ determines the partition associated with $\vec{x}$ as well as the color of $\tau(\vec{x}\,)$ for any given τ.

Remark 5.1.9. The latter means that there is a monoid homomorphism $\eta : \mathfrak{M}_k \to \mathfrak{M}(\mathscr{C})$ such that for all $\vec{x} \in \Omega^k$ and $\tau \in \mathfrak{M}_k$ we have $c(\tau(\vec{x}\,)) = \eta(\tau)(c(\vec{x}\,))$.

We highlight the consequence that the color of any string "knows" the color of each vertex in the string. We state this formally.

Observation 5.1.10 (Vertex-color awareness)**.** *Let $\mathfrak{X} = (\Omega, c)$ be a configuration. Let $\vec{x} = x_1 \dots x_k$ and $\vec{y} = y_1 \dots y_k$ be strings in Ω^k. If $c(\vec{x}) = c(\vec{y})$ then for each $i \in [k]$ we have $c(x_i) = c(y_i)$.*

Proof. Let $\tau_i \in \mathfrak{M}_k$ be defined as the constant map to i, i.e., $\tau_i(j) = i$ for all $j \in [k]$. So, using Axiom (ii) from Def. 5.1.8 we obtain $c(x_i) = c^{(1)}(x_i) = c(\tau_i(\vec{x})) = c(\tau_i(\vec{y})) = c^{(1)}(y_i) = c(y_i)$. □

There is a functor F_2, computable in time $n^{O(k)}$, that converts k-ary partition structures into k-ary configurations without changing their underlying sets, such that $\mathrm{Iso}(\mathfrak{X}, \mathfrak{Y}) = \mathrm{Iso}(F_2(\mathfrak{X}), F_2(\mathfrak{Y}))$ for all pairs $\mathfrak{X}, \mathfrak{Y}$ of k-ary partition structures. F_2 assigns to each partition structure its unique coarsest refinement that is a configuration (with an appropriate assignment of colors).

****** ADD DETAILS HERE ******

Proposition 5.1.11 (Skeleton 1)**.** *For $1 \le \ell \le k$, the ℓ-skeleton $\mathfrak{X}^{(\ell)}$ of a k-ary configuration $\mathfrak{X}$ is an ℓ-ary configuration.*

For the proof, we need to define the padding of a transformation $\tau \in \mathfrak{M}_\ell$.

Definition 5.1.12. Let $\tau \in \mathfrak{M}_\ell$. Let $\tau' = \mathrm{pad}(\tau) \in \mathfrak{M}_k$ be defined by setting, for $j \in [k]$,

$$j^{\tau'} = \begin{cases} j^\tau & \text{if } j \le \ell \\ \ell^\tau & \text{if } j \ge \ell \end{cases} \tag{30}$$

Remark 5.1.13. The mapping $\tau \mapsto \mathrm{pad}(\tau)$ is a semigroup embedding (injective semigroup homomorphism) $\mathfrak{M}_\ell \to \mathfrak{M}[k]$ (but not a monoid embedding: the idenity does not map to the identity).

Observation 5.1.14. *Let $\vec{x} = x_1 \dots x_\ell \in \Omega^\ell$ and $\tau \in \mathfrak{M}_\ell$. Then*

$$(\mathrm{pad}(\tau))(\mathrm{pad}(\vec{x})) = \mathrm{pad}(\tau(\vec{x})). \tag{31}$$

Proof. Let $j \in [k]$. Let us compare the j-th letter of the strings on each side. First assume $j \le \ell$. Then on the right-hand side we have x_{j^τ}. On the left-hand side we have $x_{j^{\mathrm{pad}(\tau)}}$. As $j^\tau = j^{\mathrm{pad}(\tau)}$, we are done. Assume now $j \ge \ell$. Then on the right-hand side we have x_{ℓ^τ}. On the left-hand side we have $x_{\ell^{\mathrm{pad}(\tau)}}$. We conclude as before. □

Proof of Prop. 5.1.11. Let $\vec{x}, \vec{y} \in \Omega^\ell$ and assume $c^{(\ell)}(\vec{x}) = c^{(\ell)}(\vec{y})$. In other words this means $c(\mathrm{pad}(\vec{x})) = c(\mathrm{pad}(\vec{y}))$. Therefore $\rho(\mathrm{pad}(\vec{x})) = \rho(\mathrm{pad}(\vec{y}))$. Restricting these equivalence relations to $[\ell]$ we obtain $\rho(\vec{x}) = \rho(\vec{y})$, verifying (i) in Def. 5.1.8. Let now $\tau \in \mathfrak{M}_\ell$. To verify item (ii) we need to show that $c^{(\ell)}(\tau(\vec{x})) = c^{(\ell)}(\tau(\vec{y}))$. We known that $c(\mathrm{pad}(\vec{x})) = c(\mathrm{pad}(\vec{y}))$. Therefore, applying item (ii) to $\mathfrak{X}$ we obtain that $c((\mathrm{pad}(\tau))(\mathrm{pad}(\vec{x}))) = c((\mathrm{pad}(\tau))(\mathrm{pad}(\vec{y})))$. By Obs. 5.1.14 we infer that $c(\mathrm{pad}(\tau(\vec{x}))) = c(\mathrm{pad}(\tau(\vec{y})))$. According to Def. 5.4.1, the left-hand side is equal to $c^{(\ell)}(\tau(\vec{x}))$ while the right-hand side is equal to $c^{(\ell)}(\tau(\vec{y}))$, so $c^{(\ell)}(\tau(\vec{x})) = c^{(\ell)}\tau(\vec{y})$, as desired. □

Proposition 5.1.15 (Induced subconfiguration)**.** *For $\Phi \subseteq \Omega$, the induced substructure $\mathfrak{X}[\Phi]$ of a k-ary configuration is a k-ary configuration.*

Proof. We only need to observe that if $\mathrm{supp}(\vec{x}) \subseteq \Phi$ then $\mathrm{supp}(\vec{x}) \subseteq \Phi$ by Eq. (28). □

5.2 k-ary coherent configurations

Notation 5.2.1 (Substitution)**.** For $\vec{x} \in \Omega^k$, $z \in \Omega$, and $j \in [k]$ we write $\vec{x}^{\,j}(z)$ to denote the string $\vec{y} = y_1 \dots y_k$ where

$$y_i = \begin{cases} z & \text{if } i = j \\ x_i & \text{if } i \neq j \end{cases} \tag{32}$$

Definition 5.2.2 (k-ary coherent configuration)**.** Let $\mathfrak{X}$ be a k-ary configuration. We say that $\mathfrak{X}$ is a *k-ary coherent configuration* if additionally it satisfies the following axiom.

(iii) There is a collection of $|\mathscr{C}|^{k+1}$ parameters, $\left(\gamma(\vec{i}j) \,\middle|\, \vec{i} = i_1 \dots i_k \in \mathscr{C}^k, j \in \mathscr{C}\right)$ such that for all $\vec{i} \in \mathscr{C}^k$, $j \in \mathscr{C}$, and $\vec{x} \in \Omega^k$ such that $c(\vec{x}) = j$,

$$\left|\left\{z \in \Omega \,\middle|\, (\forall t \in [k])(c(\vec{x}^{\,t}(z)) = i_t)\right\}\right| = \gamma(\vec{i}j). \tag{33}$$

The $\gamma(\vec{i}j)$ are called the *intersection numbers* of $\mathfrak{X}$.

There is a functor F_3, computable in time $n^{O(k)}$, that converts k-ary configurations into k-ary coherent configurations without changing their underlying sets, such that $\mathrm{Iso}(\mathfrak{X}, \mathfrak{Y}) = \mathrm{Iso}(F_3(\mathfrak{X}), F_3(\mathfrak{Y}))$ for all pairs $\mathfrak{X}, \mathfrak{Y}$ of k-ary configurations. F_3 assigns to each configuration its unique coarsest refinement that is a coherent configuration (with an appropriate assignment of colors).

****** ADD DETAILS HERE ******

Proposition 5.2.3 (Skeleton 2)**.** *For $\ell \leq k$, the ℓ-skeleton $\mathfrak{X}^{(\ell)}$ of a k-ary coherent configuration $\mathfrak{X}$ is an ℓ-ary coherent configuration.*

Proof. By Prop. 5.1.11, we only need to verify item (iii) in Def 5.2.2. We claim that $\mathfrak{X}^{(\ell)}$ has intersection numbers $\gamma^{(\ell)}(\vec{i}j)$ each of which is a sum of a subset of the intersection numbers of $\mathfrak{X}$.

Let $\vec{x} \in \Omega^{(\ell)}$ with $c^{(\ell)}(\vec{x}) = j$. Let $\vec{i} = i_1 \dots i_\ell \in \mathscr{C}^\ell$. We need to show that the number $\gamma^{(\ell)}(\vec{i}j)$ of those $z \in \Omega$ for which $(\forall t \in [\ell])(c^{(\ell)}(\vec{x}^{\,t}(z)) = i_t)$ does not depend on the specific choice of $\vec{x}$ except for the assumption that $c(\vec{x}) = j$.

The assumption translates to $c(\mathrm{pad}(\vec{x})) = j$; the conditions on z translate to

$$(\forall t \in [\ell])(c(\mathrm{pad}(\vec{x}^{\,t}(z))) = i_t). \tag{34}$$

For $s = \ell, \ell+1, \dots, k$ define the following transformation $\tau_s \in \mathfrak{M}_k$. For $q \in [k]$ set

$$q^{\tau_s} = \begin{cases} q & \text{if } q \leq \ell - 1 \\ s & \text{if } q \geq \ell \end{cases} \tag{35}$$

Let us define the color-transformation $f_s : \mathscr{C} \to \mathscr{C}$ as $f_s = \eta(\tau_s)$ where η is defined in Remark 5.1.9. So if $c(\vec{x}) = h$ then $c(\tau_s(\vec{x})) = f_s(h)$.

Let us now analyze Eq. (34). For $t \leq \ell - 1$ we have

$$\mathrm{pad}(\vec{x}^{\,t}(z)) = (\mathrm{pad}(\vec{x}))^t(z) \tag{36}$$

and therefore the condition $c(\mathrm{pad}(\vec{x}^{\,t}(z))) = i_t$ is equivalent to $c((\mathrm{pad}(\vec{x}))^t(z)) = i_t$. This is not true for $t = \ell$, however. Instead we have $c(\mathrm{pad}(\vec{x}^{\,\ell}(z))) = i_\ell$ if and only if for some $s \geq \ell$ we have $f_s(c((\mathrm{pad}(\vec{x}))^s(z))) = i_\ell$. We have thus proved the following equation.

Claim 5.2.4.

$$\gamma^{(\ell)}(\vec{i}j) = \sum \gamma(\vec{i}'j) \tag{37}$$

where the summation is over those strings $\vec{i}' = i'_1 \dots i'_k \in \mathscr{C}^k$ *satisfying* $i'_s = i_s$ *for* $s \leq \ell - 1$ *and* $f_s(i'_s) = i_\ell$ *for* $s \geq \ell$.

This completes the proof of Prop. 5.2.3. □

Proposition 5.2.5 (Induced coherent subconfiguration)**.** *Let* $\mathfrak{X} = (\Omega, c)$ *be a* k*-ary coherent configuration. Let* $\Phi \subseteq \Omega$ *be a union of color classes of* Ω *(in the 1-skeleton of* $\mathfrak{X}$*). Then the induced substructure* $\mathfrak{X}[\Phi]$ *of is a* k*-ary coherent configuration.*

Proof. From Obs. 5.1.10 it is immediate that each intersection number $\gamma'(\vec{i}j)$ of $\mathfrak{X}[\Phi]$ is either equal to the corresponding intersection number $\gamma(\vec{i}j)$ for $\mathfrak{X}$ or it is zero. □

Classical coherent configurations. We refer to the case $k = 2$ (2-ary coherent configurations, the case studied by Weisfeiler and Leman) as "classical coherent configurations." So for $k \geq 2$, the 2-skeleton $\mathfrak{X}^{(2)}$ of a k-ary coherent configuration $\mathfrak{X}$ is a classical coherent configuration.

5.2.1 Restriction – ADD DETAILS

Let $\mathfrak{X} = (\Omega, c)$ be a k-ary partition structure. Let $\ell < k$ and $\vec{x} \in \Omega^\ell$. We assign a coloring $c_{\vec{x}}$ to $\Omega^{k-\ell}$ as follows.

$$\text{For } \vec{y} \in \Omega^{k-\ell} \text{ we set } c_{\vec{x}}(\vec{y}\,) = c(\vec{x}\vec{y}\,). \tag{38}$$

We denote the resulting $(k-\ell)$-ary partition structure $(\Omega, c_{\vec{x}})$ by $\mathfrak{X}_{\vec{x}}$. We call $\mathfrak{X}_{\vec{x}}$ the *restriction of $\mathfrak{X}$ by $\vec{x}$.*

Proposition 5.2.6. *Let $\mathfrak{X} = (\Omega, c)$ be a k-ary configuration.*

(a) $c_{\vec{x}}$ is a refinement of the $c^{(k-\ell)}$ (the coloring of the $(k-\ell)$-skeleton $\mathfrak{X}^{(k-\ell)}$)

(b) The assignment $\mathfrak{X} \mapsto \mathfrak{X}_{\vec{x}}$ is canonical relative to $\vec{x}$.

(c) If $\mathfrak{X}$ is a k-ary coherent configuration then $\mathfrak{X}_{\vec{x}}$ is a $(k-\ell)$-ary coherent configuration.

*** PROVE! ***

Item (a) of Prop. 5.2.6 follows from the following.

Claim 5.2.7. *If $\ell < k$ and $\vec{x}, \vec{y} \in \Omega^\ell$ and $z, w \in \Omega$ and $c(\vec{x}z) = c(\vec{y}w)$ then $c(\vec{x}\,) = c(\vec{y}\,)$.*

5.3 k-ary Weisfeiler–Leman canonical refinement

The composition of the functors F_1, F_2, and F_3 is the *k-dimensional Weifeiler–Leman* (k-WL) canonical refinement procedure. It transforms k-ary relational structures into k-ary coherent configurations. It reduces the isomorphism problem for k-ary relational structures to the isomorphism problem for k-ary coherent configurations in time $|\mathscr{C}| \cdot n^{O(k)}$ without changing the underlying sets, where $\mathscr{C}$ denotes the set of colors for the input. Typically, $|\mathscr{C}| \leq n^k$; this is automatically the case for partition structures. So it is reasonable to say that k-WL refinement takes time $n^{O(k)}$.

****

5.4 k-ary configurations — OLD — REMOVE THIS SUBSECTION

Definition 5.4.1 (t-skeleton)**.** For $R \subseteq \Omega^k$ and $t \leq k$ let $R^{(t)} = \{(x_1, \ldots, x_t) \mid (x_1, \ldots, x_t, x_t, \ldots, x_t) \in R\}$. We define the *$t$-skeleton* $\mathfrak{X}^{(t)} = (\Omega; \mathcal{R}^{(t)})$ of the k-ary relational structure $\mathfrak{X} = (\Omega; \mathcal{R}) = (\Omega; R_1, \ldots, R_r)$ by setting $\mathcal{R}^{(t)} = (R_1^{(t)}, \ldots, R_r^{(t)})$.

The group $\mathfrak{S}_k$ acts naturally on Ω^k by permuting the coordinates.

Notation 5.4.2 (Substitution)**.** For $\vec{x} = (x_1, \ldots, x_k) \in \Omega^k$ and $y \in \Omega$ let $\vec{x}_i^{\,y} = (x'_1, \ldots, x'_k)$ where $x'_j = x_j$ for all $j \neq i$ and $x'_i = y$.

We shall especially be interested in the case when the R_i partition Ω^k. This is equivalent to coloring Ω^k; if $\vec{x} = (x_1, \ldots, x_k) \in R_i$ then we call i the *color* of the k-tuple $\vec{x}$ and write $c(\vec{x}) = i$.

Definition 5.4.3 (Configuration)**.** We say that the k-ary relational structure $\mathfrak{X}$ is a *k-ary configuration* if the following hold:

(i) the R_i partition Ω^k and all the R_i are nonempty;

(ii) if $c(x_1,\dots,x_k) = c(x'_1,\dots,x'_k)$ then $(\forall i,j \le k)(x_i = x_j \iff x'_i = x'_j)$;

(iii) $(\forall \pi \in \mathfrak{S}_k)(\forall i \le r)(\exists j \le r)(R_i^\pi = R_j)$.

Here R^π denotes the relation $R^\pi = \{(x_{1^\pi},\dots,x_{k^\pi}) \mid (x_1,\dots,x_k) \in R\}$.

We call r the *rank* of the configuration. We note that the t-skeleton of a configuration of rank r is a configuration of rank $\le r$ (we keep only one copy of identical relations).

Vertex colors are the colors of the diagonal elements: $c(x) = c(x,\dots,x)$. We say that the configuration $\mathfrak{X}$ is *homogeneous* if all vertices have the same color. We note that the s-skeleton of a k-ary homogeneous configuration is homogeneous.

Definition 5.4.4 (k-ary coherent configurations)**.** We call a k-ary configuration $\mathfrak{X} = (\Omega; R_1,\dots,R_r)$ *coherent* if, in addition to items (i)–(iii), the following holds:

(iv) There exists a family of r^{k+1} nonnegative integer *intersection numbers* $p(i_0,\dots,i_k)$ $(1 \le i_0,\dots,i_k \le r)$ such that for all $\vec{x} \in R_{i_0}$ we have

$$|\{y \in \Omega \mid (\forall j \le k)(c(\vec{x}_j^{\,y}) = i_j)\}| = p(i_0,\dots,i_k). \tag{39}$$

These are the stable configurations under the *k-ary Weisfeiler-Leman canonical refinement process* (Sec. 4.2).

Observation 5.4.5. For all $t \le k$, the t-skeleton of a k-ary coherent configuration is an t-ary coherent configuration.

5.4.1 k-ary WL refinement

The k-ary version of this process, to which we refer as "k-ary WL refinement," was introduced by Mathon and this author[8] [Ba79b] in 1979 and independently by Immerman and Lander [ImL] in the context of counting logic, cf. [CaiFI]. The refinement step is defined as follows. Let $\mathfrak{X} = (\Omega; R_1,\dots,R_r)$ be a k-ary configuration (Sec. 2.3). For $\vec{x} = (x_1,\dots,x_k) \in \Omega^k$ we encode in the new color $c'(\vec{x})$ the following information: the old color $c(\vec{x})$, and for all $i_1,\dots,i_k \le r$, the number $|\{y \in \Omega \mid (\forall j \le r)(c(\vec{x}_j^{\,y}) = i_j)\}|$. As before, these data form a list, naturally ordered. To each list we assign a new color; these colors are sorted lexicographically. This gives a refined coloring that defines a new configuration $\mathfrak{X}'$. We stop when we reach a stable configuration ($\mathfrak{X} = \mathfrak{X}'$). Observation 4.3.1 remains valid, as is the canonicity of the stable configuration stated in Eq. (26).

[8] The term used in [Ba79b] and adopted by [CaiFI] and also used in Versions 1 and 2 of this paper on arXiv was "k-dimensional WL refinement." Although I initiated it, I am not entirely happy with that term; an edge (binary relation) is a one-dimensional object. A possible justification of the term is that k-ary WL counts $(k+1)$-tuples, which can be viewed as k-dimensional simplices. The term "k-ary WL" offers a convenient way out of this dilemma.

As far as I know, this paper is the first to derive analyzable gain from employing the k-ary WL method for unbounded values of k (or any value $k > 4$). (In fact, I am only aware of one paper that goes beyond $k = 2$ [BaCh+].) We use k-ary WL in the proof of the Design Lemma (Thm. 8.1.2). In our applications of the Design Lemma, the value of k is polylogarithmic (see Secs. **??**, 13.2).

5.4.2 Complexity of WL refinement.

The stable refinement (k-ary coherent configuration) can trivially be computed in time $O(k^2 n^{2k+1})$ and nontrivially in time $O(k^2 n^{k+1} \log n)$ [ImL, Sec. 4.9].

6 Functors, canonical constructions

It is critical that all our constructions be *canonical*. We shall employ a considerable variety of constructions, so to define canonicity for all of them at once, we find the language of categories convenient. (No "category theory" will be required, only the concept of categories and functors.)

The only type of category we consider will be *Brandt groupoids*, i. e., categories in which every morphism is invertible. Our categories will be *concrete*, i. e., the objects X have an *underlying set* $\square(X)$ and the morphisms are mappings between the objects (bijections in our case). (Strictly speaking, $\square$ is a functor from the given category to Sets.) We assume $\square$ is *faithful*, i. e., if objects X and Y have the same underlying set $\square(X) = \square(Y)$ and the identity map on this set is a morphism between X and Y then $X = Y$. We refer to the elements of $\square(X)$ as the *points* or the *vertices* or the *elements* of X. When using the term "category," we shall tacitly assume it is a concrete, faithful Brandt groupoid. In fact, we can limit ourselves to categories where all objects have the same underlying set, so all morphisms are permutations.

We write $\mathrm{Iso}(X, Y)$ for the set of $X \to Y$ morphisms and $\mathrm{Aut}(X) = \mathrm{Iso}(X, X)$. For a category we write $X \in \mathcal{C}$ if X is an object in $\mathcal{C}$.

We shall consider categories of various types of relational structures, including uniform hypergraphs, bipartite graphs with a declared partition into first and second parts, partitions (i. e., equivalence relations), any of these structures with colored vertices and/or edges, and special subcategories of these such as uniprimitive coherent configurations. Three categories to be referred to have self-explanatory names: Sets, $\mathsf{ColoredSets}$, $\mathsf{PartitionedSets}$. A group $G \le \mathfrak{S}(\Omega)$ defines the category of G-isomorphisms of strings on the domain Ω; the natural notation for this category, the central object of study in this paper, would seem to be "G-$\mathsf{Strings}$."

Given two categories $\mathcal{C}$ and $\mathcal{D}$, a mapping $F_o : \mathcal{C} \to \mathcal{D}$ is *canonical* if it is the mapping of objects from a functor $F : \mathcal{C} \to \mathcal{D}$. For an object $X \in \mathcal{C}$ we shall usually only describe the construction of the object $F(X)$; the assignment of a morphism $F(f) : F(X) \to F(Y)$ to a morphism $f : X \to Y$ will usually be evident. In such a case we refer to F_o as a canonical assignment (or, most often, a canonical construction). Canonical color refinement procedures are examples of canonical constructions.

A *canonical embedding* of objects from category $\mathcal{D}$ into objects from category $\mathcal{C}$ is a functor $F : \mathcal{C} \to \mathcal{D}$ such that for every object $X \in \mathcal{C}$ we have $\square(F(X)) \subseteq \square(X)$ and for each morphism $f : X \to Y$ the mapping $F(f) : F(X) \to F(Y)$ is the restriction of f to $\square(F(X))$.

Thus, a *canonical subset* of objects in $\mathcal{C}$ is a canonical embedding of objects from the category Sets into the objects of $\mathcal{C}$. Note that the vertex set of a canonically embedded object is a canonical subset. If F is a canonical embedding then the restriction of $\mathrm{Aut}(X)$ to $\square(F(X))$ is a subgroup of $\mathrm{Aut}(F(X))$. In particular, a canonical subset of $\square(X)$ is invariant under $\mathrm{Aut}(X)$.

We say that F is a canonical embedding of objects from $\mathcal{D}$ *onto* objects from $\mathcal{C}$ if $\square(F(X)) = \square(X)$ for all $X \in \mathcal{C}$.

A *canonical vertex-coloring* of objects in $\mathcal{C}$ is a canonical embedding of objects from $\mathsf{ColoredSets}$ *onto* the objects of $\mathcal{C}$ (all vertices receive a color). Similarly, a *canonical partition* of objects in $\mathcal{C}$ is a canonical embedding of objects from $\mathsf{PartitionedSets}$ onto the objects of $\mathcal{C}$ (all vertices belong to some block of the partition).

Finally, we would like to formalize the notion of *canonicity relative to an arbitrary choice*, such as individualization. In this case we consider a canonical set of objects; the objects individually are not canonical. Here is a possible definition.

Definition 6.0.1 (Category of tuples)**.** Let $\mathcal{D}$ be a category. Let $\mathcal{E}$ be a class of non-empty sets of objects from $\mathcal{D}$ with the following properties:

(i) if $X, X' \in \mathfrak{X} \in \mathcal{E}$ then $\square(X) = \square(X')$

(ii) if $X, X' \in \mathfrak{X} \in \mathcal{E}$ and $Y \in \mathfrak{Y} \in \mathcal{E}$ and $f \in \mathrm{Iso}(X, Y)$ then there exists $Y' \in \mathfrak{Y}$ such that $f \in \mathrm{Iso}(X', Y')$.

Under these conditions we turn $\mathcal{E}$ into a category as follows:

(a) for $\mathfrak{X} \in \mathcal{E}$ we set $\square(\mathfrak{X}) = \square(X)$ for any $X \in \mathfrak{X}$

(b) for $\mathfrak{X}, \mathfrak{Y} \in \mathcal{E}$, we set $\mathrm{Iso}(\mathfrak{X}, \mathfrak{Y}) = \bigcup\{\mathrm{Iso}(X, Y) \mid X \in \mathfrak{X}, Y \in \mathfrak{Y}\}$.

Proposition 6.0.2. *$\mathcal{E}$ is a category.*

Proof. We need to show that the morphisms in $\mathcal{E}$ are closed under composition. Let $f \in \mathrm{Iso}(\mathfrak{X}, \mathfrak{Y})$ and $g \in \mathrm{Iso}(\mathfrak{Y}, \mathfrak{Z})$. We need to show that $fg \in \mathrm{Iso}(\mathfrak{X}, \mathfrak{Z})$. By definition, there exist objects $X \in \mathfrak{X}$ and $Y \in \mathfrak{Y}$ such that $f \in \mathrm{Iso}(X, Y)$. Now $g \in \mathrm{Iso}(Y', Z')$ for some objects $Y' \in \mathfrak{Y}$ and $Z' \in \mathfrak{Z}$. By assumption (ii) there exists $Z \in \mathfrak{Z}$ such that $g \in \mathrm{Iso}(Y, Z)$. Therefore $fg \in \mathrm{Iso}(X, Z) \subseteq \mathrm{Iso}(\mathfrak{X}, \mathfrak{Z})$. $\square$

Definition 6.0.3 (Reduction at multiplicative cost)**.** By a *reduction of the isomorphism problem* for objects $X, Y \in \mathcal{C}$ to objects in $\mathcal{D}$ *"at multiplicative cost s"* we mean a functor $F : \mathcal{C} \to \mathcal{E}$ for some category $\mathcal{E}$ of tuples of $\mathcal{D}$ such that $|F(Y)| = s$.

Proposition 6.0.4. *If F is a reduction of $\mathrm{Iso}(X, Y)$ to $\mathcal{D}$ as above then for any $X' \in F(X)$ we have*

$$\mathrm{Iso}(X, Y) = \bigcup \left\{ F^{-1}(\mathrm{Iso}(X', Y')) \mid Y' \in F(Y) \right\}. \tag{40}$$

Moreover, the terms in this union are disjoint, and all the nonempty terms have the same cardinality.

Note that X' is fixed in this union and is chosen arbitrarily from $F(X)$.

Proof. Clear. □

So if F and F^{-1} are efficiently computable per item then the cost of computing $\mathrm{Iso}(X, Y)$ is essentially the cost of s instances of computing $\mathrm{Iso}(X', Y')$ in $\mathcal{D}$, where X' is up to us to choose from $F(X)$.

Definition 6.0.5. Let F be a reduction of the isomorphism problem in $\mathcal{C}$ to $\mathcal{D}$ at a multiplicative cost. Consider the category $\mathcal{D}^F$ whose objects are the pairs (X, X') where $X \in \mathcal{C}$ and $X' \in F(X)$. We set $\square(X, X') = \square(X)$ and $\mathrm{Iso}((X, X'), (Y, Y')) = F^{-1}\,\mathrm{Iso}(X', Y')$.

Proposition 6.0.6. *$\mathcal{C}^F$ is a category.*

Definition 6.0.7. Let $H : \mathcal{C}^F \to \mathcal{H}$ be a functor and let $(X, X') \in \mathcal{C}^F$. We say that $F(X, X')$ is *canonically assigned to X relative to X'.*

An example of this procedure is individualization. Let $\mathcal{C}$ have two objects, each of them a hypergraph. Suppose we individualize an ordered set of t vertices of the hypergraph X; we do the same with Y. We consider the category $\mathcal{D}$ of all individualized versions of X and Y. The category $\mathcal{E}$ will have two objects, the set of individualized versions of X and the set of individualized versions of Y. Suppose after some choice $\vec{u} = (u_1, \dots, u_t)$ of the ordered set of individualized vertices we find a canonically (in $\mathcal{C}^F$) embedded large UPCC U in X. We then say that U is canonical *relative to $\vec{u}$*. For those $\vec{u}$ for which the procedure does not work, we embed the empty UPCC. The multiplicative cost will be $s = n(n-1)\dots(n-t+1) \le n^t$ where n is the number of vertices of X.

But this type of argument will also occur when it cannot be phrased in terms of individualizing vertices of an object; for instance, we shall canonically construct other objects and individualize vertices of those with similar effect.

7 Breaking symmetry: colored partitions

7.1 Colored α-partitions

Definition 7.1.1. A *colored partition* of a set Ω is a coloring of the elements of Ω along with a partition of each color class. We say that this is a *colored equipartition* if all blocks within the same color class have equal size. Given a colored partition Π, let $C_1, \dots, C_r$ be the color classes and $\{B_{ij} \mid 1 \le j \le k_i\}$ be the blocks of C_i. We say that Π is *admissible* if for each color class C_i of size $|C_i| \ge 2$, all the blocks of C_i have size $|B_{ij}| \ge 2$. ($B_{ij} = C_i$ is permitted.) Let $\rho(\Pi) = \max_{i,j} |B_{ij}|$. For $0 < \alpha \le 1$, a *colored α-partition* is an admissible colored partition Π such that $\rho(\Pi) \le \alpha n$ where $n = |\Omega|$.

The category $\mathsf{ColoredPartitions}$ has as its objects sets with a colored partition. The morphisms are the bijection that preserve color and preserve the given equivalence relation (partition) in each color class.

Definition 7.1.2. A *canonical colored partition* of objects of a category $\mathcal{C}$ is a canonical embedding of objects from the category $\mathsf{ColoredPartitions}$ onto the objects of $\mathcal{C}$.

In other words this means assigning a colored partition of the vertex set of each object in $\mathcal{C}$ such that isomorphisms in $\mathcal{C}$ preserve colors and preserve the equivalence relation on each color class.

Proposition 7.1.3. *Given a colored partition, one can canonically refine it to a colored equipartition. Here refinement means refining the colors; the blocks will not change, so if the partition was admissibe, it remains admissible.*

Proof. Encode the size of each block in the color of its elements. □

Finding canonical colored 4/5-partitions will be one of our key indicators of progress.

Observation 7.1.4. Let $\alpha \geq 1/2$. A colored equipartition is an α-partition if either each color class has size $\leq \alpha n$, or the unique color-class of size $> n/2$ (the "dominant color class") is nontrivially partitioned (at leat least two blocks, the blocks have size ≥ 2).

7.2 Effect of coloring on t-tuples

Let Γ be a set and $\Phi = \binom{\Gamma}{t}$ the set of t-subsets of Γ. Let $|\Gamma| = m$; so $|\Phi| = \binom{m}{t}$. We shall need to examine the effect of a coloring of Γ on Φ. This will be used repeatedly in Section 14.

Lemma 7.2.1. *Let Γ be the disjoint union of color classes $\Delta_1, \ldots, \Delta_k$. This induces a canonical coloring of $\Phi = \binom{\Gamma}{t}$ as follows: the color of $T \in \binom{\Gamma}{t}$ is the vector $(\,|T \cap \Delta_i| \mid 1 \leq i \leq k)$. Then*

(a) the size of each color class in Φ is $\leq (2/3)|\Phi|$ with the possible exception of one of the k sets $\binom{\Delta_i}{t}$.

(b) $|\binom{\Delta_i}{t}|/|\Phi| \leq (|\Delta_i|/m)^t$.

Proof. Item (b) is trivial. We prove item (a) by induction on k. The statement is vacuously true for $k = 1$. The case $k = 2$ is the content of Prop. 7.2.3 below with $m_i = |\Delta_i|$ and $t_i = |T \cap \Delta_i|$. Let $k \geq 3$ and let $\Gamma' = \Delta_{k-1} \cup \Delta_k$. Apply the inductive hypothesis to the coloring $(\Delta_1, \ldots, \Delta_{k-2}, \Gamma')$ of Γ. We are done except that we need to consider the color classes included in $\binom{\Gamma'}{t}$. But applying the case $k = 2$ we see that all of those color classes have size $\leq (2/3)\binom{|\Gamma'|}{t} < (2/3)|\Phi|$ with the possible exception of the two sets $\binom{\Delta_i}{t}$ for $i = k-1, k$. □

Corollary 7.2.2. *We use the notation of Lemma 7.2.1. Let $\alpha < 1$ and $t \geq 2$. Then any α-coloring of Γ (every color class has size $\leq \alpha|\Gamma|$) induces a $\max(2/3, \alpha)$-coloring of $\binom{\Gamma}{t}$.*

Proof. Combine the two conclusions in Lemma 7.2.1. □

7.2.1 A binomial inequality

Proposition 7.2.3. *Let m_1, m_2, t_1, t_2 be integers; let $m = m_1 + m_2$ and $t = t_1 + t_2$. Assume $t \le m/2$ and $t_i \ge 1$ for $i = 1, 2$. Then*

$$\binom{m_1}{t_1}\binom{m_2}{t_2} \le \frac{2}{3}\binom{m}{t}. \tag{41}$$

We first make the following observation.

Claim 7.2.4. *Let $1 \le k \le n-1$. Then*

$$\binom{n}{k}^2 \le 4\binom{n}{k-1}\binom{n}{k+1}. \tag{42}$$

Proof. Expanding and simplifying, the Claim reduces to the statement

$$\frac{k+1}{k} \le 4 \cdot \frac{n-k}{n-k+1}. \tag{43}$$

This is true because $(k+1)/k \le 2$ and $(n-k)/(n-k+1) \ge 1/2$. □

Proof of Prop. 7.2.3. By Claim 7.2.4, if $1 \le t_i \le m_i - 1$ then we have

$$\binom{m_i}{t_i}^2 \le 4\binom{m_i}{t_i-1}\binom{m_i}{t_i+1}. \tag{44}$$

Let $a_s = \binom{m_1}{s}\binom{m_2}{t-s}$. Then, if $1 \le s \le m_i - 1$ and $1 \le t - s \le m_2 - 1$, multiplying Eq. (44) for $i = 1, 2$ and substituting $t_1 = s$ and $t_2 = t - s$, we obtain

$$a_s^2 \le 16 a_{s-1} a_{s+1} \le 4(a_{s-1} + a_{s+1})^2 \tag{45}$$

and therefore $a_s \le 2(a_{s-1} + a_{s+1})$. Observe that $\sum_{s=0}^{t} a_s = \binom{m}{t}$. It follows that under the conditions $1 \le s \le m_i - 1$ and $1 \le t - s \le m_2 - 1$ we have $(3/2)a_s \le a_{s-1} + a_s + a_{s+1} \le \binom{m}{t}$, hence $a_s \le (2/3)\binom{m}{t}$, as desired.

It remains to consider the cases when $t_i = m_i$ for $i = 1$ or 2. Let us say $i = 1$, so $t_1 = m_1$. So we have

$$\binom{m_1}{t_1}\binom{m_2}{t_2} = \binom{m_2}{t_2} \le \binom{m-1}{t_2} \le \binom{m-1}{t-1} = \frac{t}{m}\binom{m}{t} \le \frac{1}{2}\binom{m}{t}. \tag{46}$$

□

This inequality will be used many times in the analysis of our algorithms; we shall refer to it each time we find a canonical coloring of our set Γ.

8 Breaking symmetry: the Design Lemma

In this section we describe the first of two combinatorial symmetry-breaking tools, a canonical reduction of k-ary relational structures to binary relational structures.

8.1 The Design Lemma: reducing k-ary relations to binary

Given a relational structure $\mathfrak{X} = (\Omega, \mathcal{R})$ of moderate arity ($k = O(\log n)$ in our application) and non-negligible symmetry defect (bounded away from zero in our case), we wish to efficiently find a subgroup $G \le \mathfrak{S}(\Omega)$ such that $\mathrm{Aut}(\mathfrak{X}) \le G$ and G is substantially smaller than $\mathfrak{S}(\Omega)$. (In our applications, we wish the index $|\mathfrak{S}(\Omega) : G|$ to be exponentially large, $2^{\Omega(n)}$.) We are not able to achieve this, but we do achieve it after individualizing a small (polylogarithmic) number of vertices. We divide the task into two parts: first we reduce the general case of k-ary relational structures to UPCCs (uniprimitive coherent configurations – recall that these are binary relational structures ($k = 2$)) (the "Design Lemma," Section 8.1), and, second, we solve the problem for UPCCs (Section 9).

We now state the first of these two main combinatorial results of the paper.

Definition 8.1.1. Given a threshold parameter $1/2 \le \alpha < 1$ and a coloring of a set Ω, we call a color, and the corresponding color class $C \subseteq \Omega$, *dominant* if $|C| > \alpha n$ where $n = |\Omega|$. (Note: since $\alpha \ge 1/2$, there is at most one dominant color under any coloring.)

Theorem 8.1.2 (Design lemma)**.** *Let $1/2 \le \alpha < 1$ be a threshold parameter. Let $\mathfrak{X} = (\Omega, \mathcal{R})$ be a k-ary relational structure with $n = |\Omega|$ vertices, $2 \le k \le n/2$, and symmetry defect $\ge 1 - \alpha$. Then in time $n^{O(k)}$ we can find a value $\ell \le k - 1$ and a string $\vec{x} = x_1 \dots x_\ell \in \Omega^{\langle \ell \rangle}$ of ℓ distinct vertices such that by individualizing each x_j we obtain either*

(a) a canonical coloring of Ω with no dominant color, or

(b) a canonical coloring of Ω and a nontrivial canonical equipartition of the dominant color class, or

(c) a canonical coloring of Ω and a canonically embedded uniprimitive coherent configuration (UPCC) whose vertex set is the dominant color class.

Canonicity in the above statements is relative to $\vec{x}$.

Observation 8.1.3. Let $\mathrm{DL}(\alpha)$ be the statement of the Design Lemma for a particular $\alpha \ge 1/2$. If $1 > \alpha' \ge \alpha \ge 1/2$ then $\mathrm{DL}(\alpha')$ follows from $\mathrm{DL}(\alpha)$.

Proof. Assume $\mathrm{DL}(\alpha)$ holds. Let $U \subseteq \Omega$ be a largest symmetric subset of Ω, i. e., a largest subset such that $\mathfrak{S}(U) \le \mathrm{Aut}(\mathfrak{X})$. Let $\beta = |U|/n$. Assume $\beta \le \alpha'$ so the assumption of $\mathrm{DL}(\alpha')$ holds.

Case 1. $\beta \le \alpha$.

In this case we can apply $\mathrm{DL}(\alpha)$. If $\mathrm{DL}(\alpha)$ returns case (a) or (b), we are done (case (a) or (b) holds for $\mathrm{DL}(\alpha')$). If $\mathrm{DL}(\alpha)$ returns case (c) (a certain set C with $|C| > \alpha n$), then we are done (case (c) of $\mathrm{DL}(\alpha')$) if $|C| > \alpha' n$. If $\alpha n < |C| \le \alpha' n$ then the coloring $(C, \Omega \setminus C)$ puts us in Case (a) of $\mathrm{DL}(\alpha')$ (since $\alpha \ge 1/2$).

Case 2. $\alpha < \beta \le \alpha'$.

In this case the coloring $(U, \Omega \setminus U)$ puts us in case (a) for $\mathrm{DL}(\alpha')$. $\square$

It follows that it would suffice to prove the Design Lemma for $\alpha = 1/2$.

Remark 8.1.4. Let $\mathfrak{X}^*$ denote the UPCC obtained in case (c). If we can compute $\mathrm{Aut}(\mathfrak{X}^*)$ then we achieve a major reduction in $\mathrm{Aut}(\mathfrak{X})$ because $|\mathrm{Aut}(\mathfrak{X}^*)| \le \exp(\widetilde{O}(\sqrt{n}))$ [Ba81].

There are two ways to compute $\mathrm{Aut}(\mathfrak{X}^*)$: either directly or recursively.

Direct computation of $\mathrm{Aut}(\mathfrak{X}^*)$ can be done in $\exp(\widetilde{O}(n^{1/3}))$ (Sun–Wilmes [SuW]). Using this result would yield an overall $\exp(\widetilde{O}(n^{1/3}))$ GI test, sufficient to break the decades-old $\exp(\widetilde{O}(\sqrt{n}))$ barrier.

Rather than relying on the Sun–Wilmes theorem which would incur an $\exp(\widetilde{O}(n^{1/3}))$ multiplicative cost, we shall further reduce the UPCC case to Johnson schemes (Split-or-Johnson routine, Sec 9) at a quasipolynomial multiplicative cost. The automorphism group of the Johnson scheme $\mathfrak{J}(k,t)$ is known, it is the corresponding Johnson group $\mathfrak{S}_k^{(t)}$ (cf. ****). This leads to our overall quasipolynomial bound.

The formulation of the Design lemma, given above, is the most helpful for the applications that will follow. Below we simplify the statement by combining cases (b) and (c); this will also eliminate several lines from the pseudocode.

Theorem 8.1.5 (Design lemma, rephrased)**.** *Let $1/2 \le \alpha < 1$ be a threshold parameter. Let $\mathfrak{X} = (\Omega, \mathcal{R})$ be a k-ary relational structure with $n = |\Omega|$ vertices, $2 \le k \le n/2$, and symmetry defect $\ge 1-\alpha$. Then in time $n^{O(k)}$ we can find a value $\ell \le k-1$ and a string $\vec{x} = x_1 \dots x_\ell \in \Omega^{\langle \ell \rangle}$ of ℓ distinct vertices such that by individualizing each x_j we obtain either*

(i) a canonical coloring of Ω with no dominant color, or

(ii) a canonical (classical) coherent configuration $\mathfrak{X}^*$ on vertex set Ω such that the subconfiguration induced on the dominant vertex-color class of $\mathfrak{X}^*$ is not a clique.

Canonicity in the above statements is relative to $\vec{x}$.

Proof of equivalence. First we prove that Theorem 8.1.5 follows from Theorem 8.1.2. Outcome (a) is identical with outcome (i). In case (b), let c' denote the given canonical coloring of Ω, C the dominant c'-class, and E the equivalence relation on C corresponding to the given equipartition. Define the coloring d of $C \times C$ as follows: For $x \in C$, let $d(x,x) = c'(x)$; for $(x,y) \in E$, let $d(x,y) = d_0$, and for $x \ne y$, $(x,y) \notin E$, let $d(x,y) = d_1$, where d_0, d_1 are two special colors. The structure $\mathfrak{Y} = (C,d)$ is clearly a non-clique coherent configuration on C.

In case (c), let C be the dominant color class and $\mathfrak{Y} = (C,d)$ be the given UPCC.

Let us now define the coherent configuration $\mathfrak{X}' = (\Omega, c'')$ required by (ii). For $x,y \in \Omega$, if $x = y$ or at least one of x,y does not belong to C then let $c''(x,y) = (c'(x), c'(y))$. If $x \ne y$ and $x,y \in C$ then let $c(x,y) = d(x,y)$. It is easy to see that $\mathfrak{X}^*$ is coherent; C is its dominant color class; and $\mathfrak{X}^*[C] = \mathfrak{Y}$ is not a clique.

Next we prove that Theorem 8.1.2 follows from Theorem 8.1.5. Again, outcome (i) is identical with outcome (a). In case (ii), let $\mathfrak{X}^* = (\Omega, c'')$ be the coherent configuration obtained. If $\mathfrak{X}^*$ has no dominant vertex-color class, we are in case (a). Otherwise, let C denote the dominant vertex-color class of $\mathfrak{X}^*$. If $\mathfrak{X}^*[C]$ is a UPCC, we are in case (c). Otherwise, $\mathfrak{X}^*[C]$ is imprimitive; the connected components of its first disconnected non-diagonal constituent form the required canonical equipartition (Prop. 3.4.4). $\square$

8.2 The algorithm

Procedure "Design Lemma"

Input: a k-ary structure $\mathfrak{X} = (\Omega, \mathcal{R})$ with symmetry defect $\geq 1 - \alpha$ where $1/2 \leq \alpha < 1$.

Output: (i) or (ii) of Theorem 8.1.5.

01 apply k-WL to $\mathfrak{X}$ (: henceforth we assume $\mathfrak{X}$ is k-ary coherent :)
02 **for** $\ell = 0$ **to** $k - 1$
03   **for** $\vec{x} \in \Omega^{\langle \ell \rangle}$
04   **if** each vertex-color class in the restriction $\mathfrak{X}_{\vec{x}}$ has size $\leq \alpha n$
05   **then return** the vertex-coloring, **exit** (: goal (i) achieved :)
06   **else** (: we have a unique vertex-color class $C(\vec{x}\,)$ of size $> \alpha n$:)
07   **if** $\ell \leq k - 2$ and the 2-skeleton $(\mathfrak{X}_{\vec{x}})^{(2)}$ does not induce
        a clique configuration on $C(\vec{x}\,)$ **then**
08   **return** $\mathfrak{X}^* := (\mathfrak{X}_{\vec{x}})^{(2)}$, **exit** (: goal (ii) achieved :)

Theorem 8.2.1. *Under the conditions of Theorem 8.1.5,* Procedure "Design Lemma" *terminates, achieving goal (i) or (ii) of Theorem 8.1.5.*

Note that if the 2-skeleton $(\mathfrak{X}_{\vec{x}})^{(2)}$ considered on line 07 induces a clique configuration on $C(\vec{x}\,)$ then the procedure discards the current $\vec{x}$ and moves on to the next $\vec{x}$.

What the Theorem asserts is that this will not always be the case; for some $\vec{x}$, either $\ell(\vec{x}\,) \leq k - 1$ and we succeed on line 05, or $\ell(\vec{x}\,) \leq k - 2$ and we succeed on line 08.
****

Remark 8.2.2. DO WE NEED THIS?

Observe that this result immediately proves Theorem 8.2.1 for $k = 2$. However, we shall not exclude the case $k = 2$ from the general proof below.

****

8.3 k-ary coherent configurations with a dominant vertex-color

The next lemma, in combination with the "Large Clique lemma" (Lemma 3.4.25), will provide the contradiction required for our proof of correctness of the algorithm.

Lemma 8.3.1 (Non-clique lemma)**.** *Let $1/2 \leq \alpha \leq 1$ be a threshold parameter. Let $\mathfrak{X} = (\Omega, \mathcal{R})$ be a k-ary coherent configuration with $n = |\Omega|$ vertices, $2 \leq k \leq n/2$, and symmetry defect $\geq 1 - \alpha$. Assume $\mathfrak{X}$ has a dominant vertex-color class C, so $|C| > \alpha n$. The there exists a string $\vec{x} \in \Omega^{\langle \leq k-1 \rangle}$ such that either*

(a) $C \setminus \operatorname{supp}(\vec{x}\,)$ is not homogeneous under the coloring $c_{\vec{x}}$ (not all elements get the same color), or

(b) $\ell(\vec{x}\,) \leq k - 2$ and the classical coherent configuration induced by the 2-skeleton $(\mathfrak{X}_{\vec{x}})^{(2)}$ on the set $C \setminus \operatorname{supp}(\vec{x}\,)$ is not a clique.

Remark 8.3.2. Note that C is a union of vertex-color classes in $\mathfrak{X}_{\vec{x}}$. Therefore, if $\ell(\vec{x}\,) \le k-2$, then the subconfiguration of $(\mathfrak{X}_{\vec{x}})^{(2)}$ induced on $C \setminus \mathrm{supp}(\vec{x}\,)$ is indeed coherent (Prop. 5.2.5).

Proof of Lemma 8.3.1. C is too large to be a twin equivalence class (because of the lower bound on the symmetry defect). So there exist $u, v \in C$, $u \neq v$, such that the transposition $\tau = (u, v)$ does not belong to $\mathrm{Aut}(\mathfrak{X})$. Therefore there exists $\vec{z} \in \Omega^k$ such that $c(\vec{z}^{\,\tau}) \neq c(\vec{z}\,)$. Extending the coloring to $\Omega^{\le k}$ (Sec. 5.1.4), let $\vec{y}$ be a shortest string such that $c(\vec{y}^{\,\tau}) \neq c(\vec{y}\,)$. Let $\vec{y} = y_1 \dots y_q$. It follows from the axioms of configurations that the y_i are all distinct.

Now $\mathrm{supp}(\vec{y}\,) \cap \{u, v\} \neq \emptyset$ (since otherwise we would have $\vec{y}^{\,\tau} = \vec{y}$), so without loss of generality we may assume $u \in \mathrm{supp}(\vec{y}\,)$. (Vertex v may or may not belong to $\mathrm{supp}(\vec{y}\,)$.) Again by the axioms of configurations we may assume that

(u) $u = y_q$ if $v \notin \mathrm{supp}(\vec{y}\,)$, and

(uv) $u = y_{q-1}$ and $v = y_q$ if $v \in \mathrm{supp}(\vec{y}\,)$.

Let $\ell = \max\{i \mid y_i \notin \{u, v\}\}$, so $\ell = q-1$ if $v \notin \mathrm{supp}(\vec{y}\,)$ and $\ell = q-2$ if $v \in \mathrm{supp}(\vec{y}\,)$. In particular, $y_{\ell+1} = u$ in each case.

Let $\vec{x} = y_1 \dots y_\ell$. So $\mathrm{supp}(\vec{x}\,) \cap \{u, v\} = \emptyset$. In Case (u) we have $\vec{y} = \vec{x}u$ and in Case (uv) we have $\vec{y} = \vec{x}uv$. In particular, in Case (uv) we have $\ell(\vec{x}\,) = \ell \le k-2$.

PICTURE!

Claim 8.3.3. *(a) In Case (u) we have $c_{\vec{x}}(u) \neq c_{\vec{x}}(v)$.*

(b) In Case (uv) we have $c_{\vec{x}}(uv) \neq c_{\vec{x}}(vu)$.

Proof. (a) By definition, $c_{\vec{x}}(u) = c(\vec{x}u) = c(\vec{y}\,) \neq c(\vec{y}^{\,\tau}) = c(\vec{x}v) = c_{\vec{x}}(v)$.
(b) By definition, $c_{\vec{x}}(uv) = c(\vec{x}uv) = c(\vec{y}\,) \neq c(\vec{y}^{\,\tau}) = c(\vec{x}vu) = c_{\vec{x}}(vu)$. ☐

Since $u, v \in C \setminus \mathrm{supp}(\vec{x}\,)$, we see that in Case (u), $C \setminus \mathrm{supp}(\vec{x}\,)$ is not homogeneous under $c_{\vec{x}}$ (u and v have different colors) and in Case (uv), the configuration induced by $\mathfrak{X}_{\vec{x}}^{(2)}$ on $C \setminus \mathrm{supp}(\vec{x}\,)$ is not a clique (uv and vu have different colors). This completes the proof of Lemma 8.3.1. ☐

8.4 Completing the proof of the Design Lemma

Proof of Theorem 8.2.1. After line 01 we assume that $\mathfrak{X}$ is a k-ary coherent configuration.

Assume for a contradiction that the algorithm fails. It follows that for all $\vec{x} \in \Omega^{\langle \le k-1 \rangle}$

(A) we have a (unique) dominant vertex-color class $C(\vec{x}\,)$ in $\mathfrak{X}_{\vec{x}}$ (so $|C(\vec{x}\,)| > \alpha n$) (otherwise we succeed on line 05) and

(B) if $\ell(\vec{x}) \le k-2$ then the 2-skeleton $\mathfrak{X}_{\vec{x}}^{(2)}$ induces a clique configuration on $C(\vec{x})$ (otherwise we succeed on line 08).

Let $C = C(\Lambda)$ (where Λ is the empty string). It is clear that if the string $\vec{y}$ is a prefix of the string $\vec{x}$ then $C(\vec{x}\,) \subseteq C(\vec{y}\,)$ (since the vertex-coloring by $c_{\vec{x}}$ is a refinement of the vertex-coloring of $c_{\vec{y}}$ and no vertex-color class of $c_{\vec{y}}$ other than the largest has room to accommodate $C(\vec{x}\,)$). In particular, $C(\vec{x}\,) \subseteq C$ for all $\vec{x}$. In fact, since $C(\vec{x}\,) \cap \operatorname{supp}(\vec{x}\,) = \emptyset$, we have, for all $\vec{x}$,

$$C(\vec{x}\,) \subseteq C \setminus \operatorname{supp}(\vec{x}\,). \tag{47}$$

We shall use the "Large clique lemma" (Lemma 3.4.25) through the following statement.

Claim 8.4.1. *Let $\vec{x} \in \Omega^{\langle \leq k-1 \rangle}$ be a non-empty string with parent $\vec{y}$, so $\vec{x} = \vec{y}z$ for some $z \in \Omega$. Then*

$$C(\vec{x}\,) = C(\vec{y}\,) \setminus \{z\}. \tag{48}$$

Proof. We have $\ell(\vec{y}\,) = \ell(\vec{x}) - 1 \leq k - 2$. Therefore, by item (B) above, $C(\vec{y}\,)$ induces a clique in the 2-skeleton $\mathfrak{X}_{\vec{y}}^{(2)}$.

We have $|C(\vec{y}\,)| > \alpha n \geq n/2$, so by the "Large clique lemma" (Lemma 3.4.25), $C(\vec{y}\,)$ is a twin equivalence class in $\mathfrak{X}_{\vec{y}}^{(2)}$. In particular, for all $u \in C(\vec{y}\,) \setminus \{z\}$, the color $c_{\vec{y}}(zu)$ is the same (independent of u); call this color i_z. So for all $u \in C(\vec{y}\,) \setminus \{z\}$, we have $c_{\vec{x}}(u) = c_{\vec{y}}(zu) = i_z$, independent if u. It follows that i_z is the dominant vertex-color in $\mathfrak{X}_{\vec{x}}$ and Eq. (48) holds. □

The following corollary to Claim 8.4.1 should be compared with Eq. (47).

Claim 8.4.2. *For all $\vec{x} \in \Omega^{\langle \leq k-1 \rangle}$ we have*

$$C(\vec{x}\,) = C \setminus \operatorname{supp}(\vec{x}\,). \tag{49}$$

Proof. By induction on $\ell(\vec{x})$. True by definition for $\ell(\vec{x}) = 0$. Assume now $\ell(\vec{x}) \geq 1$. The inductive step is provided by Claim 8.4.1. □

But Claim 8.4.2 creates a contradiction between the "Non-clique lemma" (Lemma 8.3.1) and our items (A) and (B) above. Indeed, let $\vec{x} \in \Omega^{\langle \leq k-1 \rangle}$ be a string of which the existence is guaranteed by Lemma 8.3.1. Now option (a) of Lemma 8.3.1 says $C \setminus \operatorname{supp}(\vec{x})$ is not homogeneous under $c_{\vec{x}}$, but this is contradicted by item (A), given Claim 8.4.2. But then option (b) of Lemma 8.3.1 contradicts item (B), given Claim 8.4.2. So no such $\vec{x}$ can exist, a contradiction, completing the proof of Theorem 8.2.1 and thereby the proof of the Design Lemma. □

9 Breaking symmetry: Split-or-Johnson

In this section we provide our second main combinatorial symmetry-breaking tool. The output of the Design Lemma was either a canonical colored α-partition for, say, $\alpha = 3/4$, or a canonically embedded large UPCC. In this section our algorithm takes a UPCC as input and attempts to find a canonical colored α-partition for, say, $\alpha = 3/4$.

This is not always possible. Johnson schemes are barriers to good partitions; the Johnson scheme $\mathfrak{J}(m, t)$ requires a multiplicative cost of $\exp(\Omega(m/t))$ for a canonical α-partition with any constant $\alpha < 1$ to arise. This follows from Prop. 9.1.1 below.

Since $n = \binom{m}{t}$, this cost is prohibitive: for bounded t it results in an exponential, $\exp(\Omega(n^{1/t}))$, algorithm.

We shall demonstrate that in a well–defined sense, *Johnson schemes are the only barriers.* Our algorithm takes a UPCC and returns, at a quasipolynomial multiplicative cost, a canonical colored 3/4-partition or a canonically embedded Johnson scheme that takes up at least a 3/4 fraction of the vertex set.

This cost is equivalent to the cost of individualizing a polylogarithmic number of vertices, although this is not how it happens. Canonical auxiliary structres are constructed, and vertices of those are individualized – these could be called "ideal vertices" from the point of view of the input UPCC.

The bulk of the work is the same task – find a good partition or return a large Johnson scheme – where the input is an uneven bipartite graph with large symmetry defect. We want to partition the large part, or find an embedded Johnson scheme in it; so that part stays essentially constant, while we iteratively reduce the small part.

9.1 Resilience of Johnson schemes

Johnson schemes are highly resilient against partitioning. Here is a formal statement of this observation.

Proposition 9.1.1. *Let $0 < \epsilon \le 1/3$. The multiplicative cost of a (relative) canonical $(1-\epsilon)$-partition of the Johnson scheme $\mathfrak{J}(m,t)$ is $\ge (t/\epsilon)^{\epsilon m/t}$.*

Proof. This is an immediate consequence of Corollary 9.1.3 below. ☐

The following lemma says that if we try to break up a Johnson scheme at moderate multiplicative cost, we fail badly; a large Johnson subscheme remains intact.

Lemma 9.1.2 (Intact Johnson subscheme)**.** *Let $G \le \mathrm{Aut}(\mathfrak{J}(m,t))$. Assume $m!/|G| < \binom{m}{r+1}$ for some $r < m/2 - 1$. Then $G \ge \mathfrak{A}_{m-r}^{(t)}$ where $\mathfrak{A}_{m-r}^{(t)}$ acts on a $\mathfrak{J}(m-r,t)$ subscheme of $\mathfrak{J}(m,t)$ corresponding to a subset of size $m-r$ of $[m]$.*

Proof. Let us view G as a subgroup of $\mathfrak{S}_m$, so $G^{(t)} \le \mathfrak{S}_m^{(t)}$ is the subgroup of $\mathrm{Aut}(\mathfrak{J}(m,t))$ in question. Then, by the Jordan–Liebeck Theorem (Thm. 10.4.2) we have that $G \ge (\mathfrak{A}_m)_{(T)}$ for some $T \subset [m]$, $|T| \le r$. Let $\Gamma = [m] \setminus T$. This means that $G^{(t)} \ge \mathfrak{A}^{(t)}(\Gamma)$ where $|\Gamma| \ge m - r$. ☐

Corollary 9.1.3. *Let $G \le \mathrm{Aut}(\mathfrak{J}(m,t))$ and $0 < \epsilon \le 1/3$. If $m!/|G| < (t/\epsilon)^{\epsilon m/t}$ then G acts as a primitive group on a subset of relative size $\ge (1-\epsilon)$.*

Proof. The condition implies that $|\mathfrak{S}_m : G| < 1.9^m$. Let r be the smallest value such that $|\mathfrak{S}_m : G| < \binom{m}{r+1}$. By Lemma 9.1.2, we have a Johnson group $\mathfrak{A}_{m-r}^{(t)} \le G^{(t)}$ act on a subset of size $\binom{m-r}{t}$. This group is primitive on this subset. Now $\binom{m-r}{t}/\binom{m}{t} \ge (1-r/m)^t > 1-(rt/m)$. So we are done if $rt/m > \epsilon$. Let us assume $rt/m \le \epsilon$. Then

$$|\mathfrak{S}_m : G| \ge \binom{m}{r} \ge (m/r)^r = \left((m/r)^{r/m}\right)^m \ge (t/\epsilon)^{\epsilon m/t}, \tag{50}$$

contrary assumption. □

Remark 9.1.4. This result means that for fixed t (e. g., $t = 2$, the most severe bottleneck case for decades), the multiplicative cost of obtaining a constant-factor reduction in the domain size $n = \binom{m}{t}$ is exponential in m; and $m > n^{1/t}$.

9.2 Split-or-Johnson: the Extended Design Lemma

In each result in this section, canonicity involves a combination of the following categories (cf. Section 6): binary relational structures (Theorem 9.2.1), vertex-colored bipartite graphs (Theorem 9.2.2), k-ary relational structures (Theorem 9.2.3), and the category of colored partitions in each result.

Recall the definitions of *canonical colored partition* and an *α-partition* (Defs. 7.1.2 and 7.1.1).

Recall that "UPCC" means *uniprimitive coherent configuration* (Def. 3.2.1).

We can now state the two main results of Section 9.

Theorem 9.2.1 (UPCC Split-or-Johnson)**.** *Let $\mathfrak{X} = (V; R_1, \dots, R_r)$ be a UPCC with n vertices and let $2/3 \le \beta < 1$ be a threshold parameter. Then at quasipolynomial multiplicative cost we can find either*

(a) a canonical colored β-partition of V, or

(b) a canonically embedded nontrivial Johnson scheme on a subset of V of size $\ge \beta n$.

(The time bounds do not depend on β.)

Theorem 9.2.2 (Bipartite Split-or-Johnson)**.** *Let $X = (V_1, V_2; E, f)$ be a vertex-colored bipartite graph with $|V_1| \ge 2$ and let $2/3 \le \alpha < 1$ be a threshold parameter. Assume $|V_2| < \alpha|V_1|$. Assume moreover that the symmetry defect of X on V_1 is at least $1 - \alpha$. Then at quasipolynomial multiplicative cost we can find either*

(a) a canonical colored α-partition of V_1, or

(b) a canonically embedded nontrivial Johnson scheme on a subset of V_1 of size $\ge \alpha|V_1|$.

(The time bounds do not depend on α.)

These results will be proved recursively by mutual reduction to each other.

Combining the Design Lemma and Theorem 9.2.1 we obtain our overall combinatorial partitioning tool, the main result of the combination of Sections 8 and 9.

Theorem 9.2.3 (Extended Design Lemma)**.** *Let $3/4 \le \alpha < 1$ be a threshold parameter. Let $\mathfrak{X} = (\Omega, \mathcal{R})$ be a k-ary relational structure with n vertices, $2 \le k \le n/4$, and relative strong symmetry defect $> 1 - \alpha$. Then at a multiplicative cost of $q(n)n^{O(k)}$, where $q(n)$ is a quasipolynomial function, we can find either*

(a) a canonical colored α-partition of the vertex set, or

(b) a canonically embedded nontrivial Johnson scheme on a subset $W \subseteq \Omega$ of size $|W| \ge \alpha n$.

(The time bounds do not depend on α.)

9.3 Minor subroutines

First we describe a reduction of Theorem 9.2.1 to Theorem 9.2.2. The procedure will also serve as a subroutine to the algorithm for Theorem 9.2.2.

Lemma 9.3.1 (UPCC-to-bipartite)**.** *Let $\mathfrak{X} = (V;\mathcal{R})$ be a UPCC with n vertices and let $2/3 \le \beta \le 1$ be a threshold parameter. Then at a multiplicative cost of $\le n$ and polynomial additive cost one can either*

(i) achieve objective (a) *of Theorem 9.2.1, or*

(ii) reduce the given instance of Theorem 9.2.1 to Theorem 9.2.2 by computig a threshold parameter $\alpha \ge 2/3$ and a (relative) canonically embedded semiregular bipartite graph $X = (V_1, V_2; E)$ with $V_1 \cup V_2 \subseteq V$, and $|V_1| \ge \beta n$ such that a solution to each part of Theorem 9.2.2 for X is also a solution to the corresponding part of Theorem 9.2.1 for $\mathfrak{X}$.

Proof. Let $\mathfrak{X} = (V; R_1, \dots, R_r)$ where $R_1 = \text{diag}(V)$ is the diagonal. Let d_i be the out-degree of the vertices in R_i; so $d_1 = 1$. Pick a vertex $x \in V$. Let $C_i = \{y \in V \mid (x, y) \in R_i\}$; so $|C_i| = d_i$. Individualize x; this splits V into the (relative) canonical subsets C_i. (See the definition of relative canonicity in Sec. 6.) If $d_i \le \beta n$ for all i, we are done (objective (a) has been achieved).

Assume now that (say) $d_2 > \beta n$; so (V, R_2) is an undirected graph (since $d_2 \ge n/2$) and its complement has diameter 2 (see the proof of [Ba81, Prop. 4.10]). Let $(x, z) \in R_2$ and let $y \in V$ be such that $(x, y) \in R_i$ and $(z, y) \in R_j$ where $i, j \ge 3$. Consider the bipartite graph $X = (C_2, C_i; E)$ where $E = (C_2 \times C_i) \cap R_j$.

X is a semiregular (Prop. 3.4.5) bipartite graph with $|C_2| > \beta n \ge 2n/3$ and therefore $|C_i| < n/3 < |C_2|/2$. We have $E \ne \emptyset$ since $(z, y) \in E$. The degree of $y \in C_i$ in X is $d_j < n/3 < 2n/3 \le d_2$ and therefore E is not complete, i. e., $E \ne C_2 \times C_i$. It follows that in each part, the relative symmetry defect of X is $\ge 1/2$ (Prop. 2.4.30).

Let now $\alpha = \beta n / d_2$. So $\alpha > \beta \ge 2/3$.

If the relative symmetry defect of X in C_2 is between $1/2$ and and α then we have a canonical colored β-partition of V_1 (the nontrivial twin equivalence classes of X, one block for the vertices in C_2 without twins, and one block $V_1 \setminus C_2$).

Else, apply Theorem 9.2.2 to X to obtain either obtain a canonical colored α-partition of C_2 (and thereby a canonical colored β-partition of V_1 as above) or the embedded nontrivial Johnson scheme of the required size. □

Our next routine takes a colored bipartite graph $X = (V_1, V_2; E)$ and helps make V_2 homogeneous. Recall that we say that $x, y \in V_1$ are *twins* if the transposition $\tau = (x, y)$ is an automorphism of X, i. e., if x and y have the same neighborhood.

Procedure Reduce-Part2-by-Color

Input: A colored bipartite graph $X = (V_1, V_2; E, f)$ where and $|V_2| < \alpha |V_1|$ such that there are no twins in V_1;
a partition $V_2 = C_1 \cup C_2$ where each C_j is a union of color classes.

Output: $j \in \{1,2\}$ such that in the induced colored bipartite subgraph $X_j = X[V_1, C_j]$ the symmetry defect of V_1 is $\geq (n_1 - 1)/2$ where $n_1 = |V_1|$

The procedure computes the symmetry defect of V_1 in each X_j.

Lemma 9.3.2. *In at least one of X_1 and X_2, the symmetry defect of V_1 is at least $(n_1 - 1)/2$.*

Proof. Assume for a contradiction that for $j = 1, 2$ there exists a twin equivalence class $D_j \subseteq V_1$ of size $|D_j| \geq 1 + (n_1/2)$ in X_j. It follows that $|D_1 \cap D_2| \geq 2$. On the other hand, all vertices in D_j are twins with respect to C_j; therefore all vertices in $D_1 \cap D_2$ are twins in X. Since X has no twins, we infer $|D_1 \cap D_2| \leq 1$, a contradiction. □

****

9.4 Bipartite Split-or-Johnson

In the rest of Sec. 9 we prove Theorem 9.2.2.

Proof. We use the notation of Theorem 9.2.2. Let $n_i = |V_i|$. We view X as a vertex-colored graph where the vertex-colors discriminate between V_1 and V_2. We may assume at all times that $|E| \leq |V_1||V_2|/2$ (otherwise take the bipartite complement). E is not empty because of the positive symmetry defect assumption.

Procedure Bipartite Split-or-Johnson

Input: a threshold parameter $3/4 \leq \alpha < 1$
a vertex-colored bipartite graph $X = (V_1, V_2; E, f)$ such that
$|V_2| < \alpha|V_1|$ and the symmetry defect of X on V_1 is at least $1 - \alpha$

Output: Output: item (a) or (b) of Theorem 9.2.2.

Below we use the abbreviation "CC" for "coherent configuration."

1. If $n_1 \leq C_0$ for some absolute constant C_0, individualize $(1-\alpha)n_1$ vertices of V_1, exit (objective (a) achieved) (: $n_1 > C_0$:)

2. If $n_2 \leq c \log n_1$ for some specific constant $c > 0$ then individualize all vertices of V_2, apply naive vertex refinement, return colored partition of V_1, exit
Claim. This is a colored α-partition.

   Proof. All vertices of the same color in V_1 are twins. □

   Instead of individualizing all vertices of V_2 at a multiplicative cost of $n_2! \leq (c \log n_1)! \approx n_1^{c' \log\log n_1}$, we can apply the method of Sec. 4.1.2 to achieve goal (a) at a multiplicative cost of $n_1^{O(1)}$.

   In the next five steps we shall achieve a reduction to bihomogeneous CCs, cf. Def. 3.4.24.

3. Apply WL refinement to X. Let $\mathfrak{X} = (V; R_1, \dots, R_r)$ denote the resulting CC. Let $\mathfrak{X}_i = \mathfrak{X}[V_i]$ be the subconfiguration induced by V_i. Let $\mathfrak{X}_{12} = \mathfrak{X}[V_1, V_2]$ be the bipartite subconfiguration induced by the pair (V_1, V_2).
   (: $\mathfrak{X}_1$ and $\mathfrak{X}_2$ are coherent; $\mathfrak{X}_{12}$ is a refinement of E and therefore $\mathfrak{X}_{12}$ is nontrivial :)
   (: $\mathrm{Aut}(X) = \mathrm{Aut}(\mathfrak{X}_{12})$ because $\mathrm{Aut}(X) \leq \mathrm{Aut}(\mathfrak{X}_{12})$ by the canonicity of $\mathfrak{X}_{12}$; and $\mathrm{Aut}(\mathfrak{X}_{12}) \leq \mathrm{Aut}(X)$ because $\mathfrak{X}_{12}$ is a refinement of E. :)

4. If all vertex-color classes in V_1 have size $\leq \alpha n_1$ then return the colored partition of V_1, exit (objective (a) achieved) (: dominant vertex-color class has size $> \alpha n_1$:)

5. Let $W_1 \subseteq V_1$ be the dominant color class: $|W_1| > \alpha n_1$. Update $\alpha \leftarrow \alpha n_1/|W_1|$, $V_1 \leftarrow W_1$ (this automatically updates n_1), $X \leftarrow X[W_1, V_2]$ (induced subgraph), $\mathfrak{X} \leftarrow \mathfrak{X}[W_1 \cup V_2]$; this automatically updates $\mathfrak{X}_1$, $\mathfrak{X}_2$, $\mathfrak{X}_{12}$ (: $\mathfrak{X}_1$ is homogeneous :)

6. If $\mathfrak{X}_1$ is imprimitive, return the connected components of the first disconnected off-diagonal color, exit (objective (a) achieved) (: $\mathfrak{X}_1$ is primitive :)

   Claim. $\mathfrak{X}_1$ is uniprimitive.

   Proof. Given that $\mathfrak{X}_1$ is primitive, we only need to rule out that $\mathfrak{X}_1$ is the clique configuration. V_1 is larger than any other color-class in $\mathfrak{X}$. Therefore, if $\mathfrak{X}_1$ were the clique configuration, it would follow by the Large clique lemma (Lemma 3.4.25) that V_1 is a twin equivalence class in $\mathfrak{X}$, a contradiction with the positive symmetry defect assumption. □

7. If $\mathfrak{X}_2$ is not homogeneous, let $(D_1, \dots, D_k)$ be the partition of V_2 into vertex-color classes. Pick the smallest j such that the induced bipartite substructure $\mathfrak{X}[V_1, D_j]$ is nontrivial. (Such a j exists because otherwise V_1 would be twin equivalence class in $\mathfrak{X}_{12}$.)
   Set $\mathfrak{X} \leftarrow \mathfrak{X}[V_1 \cup D_j]$. This automatically updates $V_2 \leftarrow D_j$ (: $\mathfrak{X}$ bihomogeneous :)

8. (: $\mathfrak{X}_1$ is a UPCC, $\mathfrak{X}_2$ is homogeneous, and $\mathfrak{X}_{12}$ is nonotrivial :)
   Let i be the smallest index such that R_i is involved in $\mathfrak{X}_{12}$. Replace $X \leftarrow (V_1, V_2; R_i)$
   (: X is semiregular, nontrivial; therefore its symmetry defect is $\geq 1/2$ in each part by Prop. 2.4.30 :)
   Update $\mathfrak{X} \leftarrow$ WL refinement of X. The only effect of this is that it may merge some of the edge-color classes in $\mathfrak{X}$; but $\mathfrak{X}_{12}$ remains nontrival (because R_i remains one of its constituents). $\mathfrak{X}_1$ remains a UPCC by Claim above.

   (: $\mathfrak{X}_1$ is a UPCC, $\mathfrak{X}_2$ is homogeneous, and $\mathfrak{X}_{12}$ is nontrivial; X is a constituent of $\mathfrak{X}_{12}$, so X is semiregular, nontrivial :)

   Claim. There are no X-twins in V_1.

   Proof. This follows from the "no twins in primitive color" theorem (Theorem 3.4.11), given that $\mathfrak{X}_1$ is primitive. □

9. We need to consider the following cases:

   (i) $\mathfrak{X}_2$ is imprimitive: Section 9.6

   (ii) $\mathfrak{X}_2$ is the clique configuration; we refer to this as the "block design case," Section 9.7

   (iii) $\mathfrak{X}_2$ is uniprimitive: Section 9.8

□

9.5 Measures of progress

Throughout the process, $n_1 = |V_1|$ will not increase. We say that a parameter m is *significantly reduced* if $m_{\text{new}} \leq 0.9m_{\text{old}}$.

We deem to have made major progress if any of the following occurs:

- goal (a) or (b) is achieved
- n_2 is significantly reduced
- $\mathfrak{X}_2$ moves from clique to UPCC while n_2 does not increase

Goal (a) is automatically achieved if Step 2 is executed.

9.6 Imprimitive case

Case: $\mathfrak{X}_2$ is a homogeneous, imprimitive coherent configuration; $\mathfrak{X}_1$ is a UPCC; the link $\mathfrak{X}_{12}$ is nontrivial. In particular, there are no R_i-twins in V_1 with respect to any of the constituents R_i of $\mathfrak{X}_{12}$.

Lemma 9.6.1. *Under the assumptions stated in "Case" above, we can either return a canonical colored* $1/2$*-partition of* V_1 *at a multiplicative cost of* $< n_2$*, or return, at only additive polynomial cost (no multiplicative cost) a canonical bipartite graph* $Y = (V_1, W_2; F)$ *such that* $|W_2| \leq |V_2|/2$ *such that the symmetry defect of* V_1 *in* Y *is* $\geq 1/2$*.*

Proof. Let $B_1, \ldots, B_m$ be the connected components of a disconnected non-diagonal color in $\mathfrak{X}_2$, say R_2. The idea is either to replace V_2 by one of the blocks (reducing n_2 to $n_2/m \leq n_2/2$) or to contract each block (reducing n_2 to $m \leq n_2/2$), significant progress in each case. We shall see that one of these is always possible without reducing the symmetry defect on V_1 below $1/2$.

Let $J = \{c(x, y) \mid x \in V_1, y \in V_2\}$. (Here we are talking about colors in $\mathfrak{X}_{12}$.) Let d_j be the in-degree of $y \in V_2$ in color $j \in J$. (d_j does not depend on y because of the homogeneity of $\mathfrak{X}_2$.) Note that $|J| \geq 2$ because the coloring of $V_1 \times V_2$ is a refinement of E; so $d_j < n_1$ for all $j \in J$.

Procedure ImprimitiveCase

1. If $(\forall j \in J)(d_j \leq n_1/2)$ then individualize some $x \in V_2$. This splits V_1 into color classes of size d_j. Return this partition of V_1, exit.
   (: canonical colored 1/2-partition of V_1 found :)

2. else (: for some $j \in J$ we have $d_j > n_1/2$:)
   For $i = 1, \dots, m$ let $Z_i = X(V_1, B_i; R_j)$.

   (i) if $(\exists i)$(the symmetry defect of V_1 in Z_i is $\geq 1/2$) then $Y \leftarrow Z_i$
   (: This involves choosing i at a multiplicative cost of m. The gain is a reduction $n_2 \leftarrow n_2/m$:)

   (ii) (: the symmetry defect of V_1 in each Z_i is less than 1/2 :)
   Let $h \in J$, $h \neq j$. Let $Y = (V_1, [m]; \overline{R}_h)$ where $(x, i) \in \overline{R}_h$ if $(\exists y \in B_i)((x, y) \in R_h)$
   (: contracting each block, $n_2 \leftarrow m$:)

   return Y

Lemma 9.6.2. *In subcase (ii) of item 2 (contracting the blocks), V_1 has symmetry defect $\geq 1/2$ in the contracted bipartite graph Y.*

Proof. Y is semiregular by Cor. 3.4.9. Moreover $\overline{R}_h$ is not empty because R_h is not empty.

Claim. Y is not complete.

Proof. For each $i \leq m$ there is a (unique) Z_i-twin equivalence class $C_i \subseteq V_1$ such that $|C_i| > n_1/2$.

Subclaim. $C_i \times B_i \subseteq R_j$.

Proof. The vertices of C_i are twins in Z_i. In other words, for each $x \in B_i$ the set $C_i \times \{x\}$ is monochromatic (has a single color), i. e., $C_i \times \{x\} \subseteq R_\ell$ for some $\ell \in J$. It follows that $d_\ell > n_1/2$. Therefore $\ell = j$, proving the Subclaim. □

Now Y is not complete because it has no edge from i to C_i. □

Since Y is semiregular, nonempty and not complete, we infer by Prop. 2.4.30 that Y has symmetry defect $\geq 1/2$, as claimed. □

This also completes the proof of Lemma 9.6.1. □

9.7 Block design case

Assumptions: no twins in V_1, $\mathfrak{X}_2$ is the clique configuration (rank-2).

Let $\mathcal{H} = (V_2, \mathcal{E})$ be the hypergraph of neighborhoods of vertices in V_1. This hypergraph has no multiple edges because there are no twins in V_1.

Case 1. V_2 is a set of twins in $\mathcal{H}$.

This means $\mathrm{Aut}(\mathcal{H})$ acts on V_2 as $\mathfrak{S}(V_2)$. Since $\mathcal{H}$ has no multiple edges and is uniform of rank d_1, it follows that $\mathcal{H}$ is the complete d_1-uniform hypergraph.

Let us label $v \in V_1$ by the set $X(v)$. This establishes a bijection between V_1 and the set $\binom{V_2}{d_1}$. Since isomorphisms preserve the number of common neighbors, this correspondence gives a canonical embedding of the Johnson scheme $\mathfrak{J}(n_2, d_1)$ on V_1, achieving goal (b) of Theorem 9.2.2. Note that the vertices of this Johnson scheme (elements of V_1) come labeled by the d_1-subsets of V_2. With this we not only exit this routine but exit the main algorithm.

Case 2. There is an $\mathcal{H}$-twin equivalence class $C \subseteq V_2$ of size $|C| \geq n_2/2$. So $\mathfrak{S}(C) \leq \mathrm{Aut}(\mathcal{H})$. Note that the vertices of C are not necessarily twins in X.

Apply procedure Reduce-Part2-by-Color to the coloring $(C, V_2 \setminus C)$. If $V_2 \setminus C$ is selected, we have made significant progress (reduced $|V_2|$ by half).

If C is selected, let $X' = X[V_1, C]$ and $\mathcal{H}' = (C, \mathcal{E}')$ be the corresponding neighborhood hypergraph, so $\mathcal{E}' = \{E \cap C \mid E \in \mathcal{E}\}$. Multiple edges are possible in $\mathcal{H}'$ ($\mathcal{E}'$ is a multiset). We note that $\mathrm{Aut}(\mathcal{H}')$ contains the restriction of the automorphisms of $\mathcal{H}$ to $\mathcal{H}'$, so C continues to be a set of twins in $\mathcal{H}'$.

Color each vertex of V_1 by the size of the corresponding hyperedge in $\mathcal{H}'$: for $v \in V_1$ let $c'(v) = |X(v) \cap C|$. If all c'-color classes have size less than αn_1, return this coloring, exit.

Otherwise, let $A \subseteq V_1$ be the dominant c'-color class, so $|A| \geq \alpha n_1 > n_1/2$. Let $X^* = X[A, C]$ and let $\mathcal{H}^* = (C, \mathcal{E}^*)$ be the corresponding neighborhood hypergraph. We observe that C continues to be a set of twins in $\mathcal{H}^*$ (because $\mathrm{Aut}(\mathcal{H}^*)$ contains the restriction of the automorphisms of $\mathcal{H}'$ to $\mathcal{H}^*$). It follows that $\mathcal{H}^*$ is regular. $\mathcal{H}^*$ is also uniform since A is a c'-color class. So X^* is semiregular. It cannot be trivial because the relative sizes of A (in V_1) and C (in V_2) add up to more than 1, contradicting Cor. 2.4.32.

Let us replace X by X^* and reduce α accordingly.

Since $\mathrm{Aut}(\mathcal{H}^*)$ acts on C as the symmetric group, all edges of $\mathcal{H}^*$ have the same multiplicity. In other words, A is equipartitioned by the twin equivalence classes in X^*. If this is a nontrivial partition, return this partition, exit.

Since X^* is nontrivial, A cannot be a set of twins in X^*, so the only remaining option is that there are no twins in A, i. e., $\mathcal{E}^*$ has no multiple edges. Since C is a set of twins in $\mathcal{H}^*$, we are back to Case 1. Proceed as in Case 1, exit.

Case 3. The relative symmetry defect of $\mathcal{H}$ is $\geq 1/2$.

Case 3a. $d_1 \leq (7/3)\log_2 n_1$.

Apply the Design lemma to $\mathcal{H}$, viewed as a d_1-ary relational structure. (Multiplicative cost $n_2^{d_1} < n_2^{(7/3)\log_2 n_1}$).

Case 3a1. The Design lemma returns a canonical colored 3/4-partition of V_2.

Apply Cor. **??**. Significant progress is made in time, polynomial in n_1, namely, n_2 is reduced to $\leq 3n_2/4$. If a multiplicative cost of m is incurred, the progress is more significant: n_2 is reduced to $\leq n_2/m$ $(2 \leq m \leq n_2/2)$.

Case 3a2. The Design lemma returns a UPCC $\mathfrak{Y}$ canonically embedded on a subset $W \subseteq V_2$ with $|W| \geq (3/4)n_2$.

Apply Reduce-Part2-by-Color to the partition $(W, V_2 \setminus W)$. If the procedure selects $V_2 \setminus W$, significant progress (n_2 reduced to $\leq n_2/4$). If it selects W, go to Sec. 9.8.

Let $U \subseteq V_2$ be the part selected, and $(\mathfrak{X}_2)_{\text{new}}$ the homogeneous coherent configuration obtained on U. If $(\mathfrak{X}_2)_{\text{new}}$ is a UPCC, exit, significant progress.

If $(\mathfrak{X}_2)_{\text{new}}$ is not a UPCC, i. e., it has rank 2, then U was a clique in $\mathfrak{Y}$ and therefore $|U| \le |W|/2 \le n_2/2$ by Prop. 2.4.11, a significant reduction of $|V_2|$.

Case 3b. $d_1 > 7/3 \log_2 n_1$.

Let $t = \lceil (7/4) \log_2 n_1 \rceil$. So $t \le (3/4)d_1$. By Lemma 2.5.12, the symmetry defect of the t-skeleton $\mathcal{H}^{(t)}$ of $\mathcal{H}$ is greater than $1/4$. Let us apply the Design lemma to $\mathcal{H}^{(t)}$.

If the result is a $3/4$-partition of V_2, ********* an application of Reduce-Part2-by-Color and if necessary, and application of the impromitive case

Else (the result is a UPCC on a subset of V_2 of size $\ge (3/4)n_2$ (significant progress, exit).

9.8 UPCC case

Case: Both $\mathfrak{X}_1$ and $\mathfrak{X}_2$ are UCPPs; the link $\mathfrak{X}_{12}$ is nontrivial.

The analysis of this case is based on Sec. 3.4.3 ("Local constituents"); we recommend that the reader review that section before reading on.

Let $\{R_j \mid j \in J\}$ be the set of constituent relations involved in the link $\mathfrak{X}_{12}$; so $|J| \ge 2$ because $\mathfrak{X}_{12}$ is nontrivial.

We say that for some $j \in J$, the color j is *α-dominant* in $\mathfrak{X}_{12}$ if $|R_j| > \alpha n_1 n_2$. We say "dominant" for (1/2)-dominant.

Procedure Reduce-by-UPCC

1. Individualize any $x \in V_2$ (: multiplicative cost n_2 :)
2. If there is no α-dominant color in $\mathfrak{X}_{12}$, return the x-local coloring c_x of V_1 (set $c_x(y) = c(x, y)$ for all $y \in V_1$), exit. (Each c_x-class in V_1 has relative size $\le \alpha$; goal (a) achieved.)
3. Else, let $m \in J$ be the dominant color in $\mathfrak{X}_{12}$. Let ℓ be the first non-dominant off-diagonal color in $\mathfrak{X}_2$, so $R_\ell(x) \subseteq V_2$ and $|R_\ell(x)| < n_2/2$. Let Y be the local constituent $Y = X_m[R_m^-(x), R_\ell(x)]$ (the edges in color m connecting the m-in-neighbors of x to the ℓ-out-neighbors of x).

   Replace $X \leftarrow Y$. This implies the following updates: $V_1 \leftarrow R_m^-(x)$, the corresponding update of α, and $V_2 \leftarrow R_\ell(x)$.

   Return X (End Procedure Reduce-by-UPCC)

Claim. Procedure Reduce-by-UPCC makes significant progress. Specifically, it either achieves goal (a) or reduces n_2 by half while maintaining symmetry defect $\ge 1/2$ in each part. The multiplicative cost is n_2.

Proof. We need to justify the "else" case.

Clearly Y is canonical relative to the choice of x.

Y is semiregular by Observation 3.4.17. It is nontrivial by Theorem 3.4.19, applied to Y^-. We need to verify the assumptions of Theorem 3.4.19. First we note that both V_1 and V_2 are vertex-color classes in $\mathfrak{X}$. Second, we need $n_1/2 < |R_m^-(x)| < n_1$. The first of these inequalities follows because m is α-dominant and therefore dominant. The second inequality is equivalent to the nontriviality of the link $\mathfrak{X}_{12}$. Finally, we need that $\mathfrak{X}_2$ is a UPCC. (Primitivity guarantees that there are no X_m-twins in V_2; and we need "uni" (not a clique) to guarantee that a non-dominant off-diagonal color ℓ exists in $\mathfrak{X}_2$.)

Summarizing, Y is a nontrivial semiregular bipartite graph; therefore its symmetry defect is $\geq 1/2$ (Prop. 2.4.30).

Finally, the update did not increase the value of n_1 and reduced n_2 to less than $n_2/2$. $\square$

CHAPTER 2: Group theory

10 Alternating quotients of a permutation group

To understand the structure of the groups where Luks reduction stops, we need some group theory.

In Luks's barrier situation we have a *giant representation* $\varphi : G \to \mathfrak{S}(\Gamma)$, meaning the image of G is a giant, i. e., $G^\varphi \geq \mathfrak{A}(\Gamma)$. We shall assume that $k = |\Gamma| > \max\{8, 2 + \log_2 n_0\}$ where n_0 is the length (size) of the largest orbit of G. We say that $x \in \Omega$ is *affected* by φ if $G_x^\varphi \not\geq \mathfrak{A}(\Gamma)$. The key result is that the pointwise stabilizer of all unaffected points is still mapped onto $\mathfrak{S}(\Gamma)$ or $\mathfrak{A}(\Gamma)$ by φ (Unaffected Stabilizers Lemma, Thm. 10.3.5). This result will be responsible for the key algorithm of the paper (Procedure LocalCertificates in Sec. 13).

Finally we show that if a permutation group $G \leq \mathfrak{S}_n$ has an alternating quotient of degree $k \geq \max\{9, 2\log_2 n\}$ this can only happen in the trivial way, namely, that for some $t \geq 1$, G has a system of imprimitivity with $\binom{k}{t}$ blocks on which G acts as the Johnson group $\mathfrak{A}_k^{(t)}$. Moreover, in every orbit there is a canonical choice of the blocks corresponding to φ which is unique; we refer to these as the *standard blocks.* These are some of the items in our "Main Structure Theorem" (Theorem 10.5.1).

10.1 Simple quotient of subdirect product

First we state a lemma that is surely well known but to which I could not find a convenient reference.

Lemma 10.1.1 (Simple quotient of subdirect product)**.** *Let $G \leq K_1 \times \cdots \times K_m$ be a subdirect product; let M_i be the kernel of the $G \to K_i$ epimorphism. Assume there is an epimorphism $\varphi : G \to S$ where S is a nonabelian simple group. Then $(\exists i)(M_i \leq \ker\varphi)$. In particular, one of the K_i admits an epimorphism onto S.*

Simplified proof by P. P. Pálfy. Let $N = \ker\varphi$. Assume for a contradiction that $N \not\geq M_i$ for all i. Then $M_iN = G$ (because N is a maximal normal subgroup). It follows that $[G, \dots, G] = [M_1N, \dots, M_mN] \leq N[M_1, \dots, M_m] \leq N\left(\bigcap_{i=1}^m M_i\right) = N$, so $[G/N, \dots, G/N] = 1$, a contradiction because $G/N \cong S$ is nonabelian simple. □

In our applications, S will be $\mathfrak{A}_k$ and the K_i the restrictions of the permutation group G to its orbits.

Remark 10.1.2. A group H is *perfect* if $H' = H$ (where $H' = [H, H]$ denotes the commutator subgroup of H). Pálfy points out that the following result appears as Lemma 2.8 in [Me].

Lemma 10.1.3 (Meierfrankenfeld)**.** *Let G be a finite group and $N \triangleleft G$ such that G/N is perfect. Then there exists a unique subnormal subgroup R of G which is minimal with respect to $G = RN$.*

Lemma 10.1.1 follows from this result. Indeed, observe that if $M_iN = G$ for all i then $R \leq M_i$ for all i and therefore $R \leq \bigcap_i M_i = 1$, contradicting the equation $RN = G$.

We believe, though, that Lemma 10.1.1 must have been folklore decades before this 1995 reference.

10.2 Large alternating quotient of a primitive group

The result of this section is Lemma 10.2.5.

First we need to state a corollary to the basic structure theorem of primitive groups, the O'Nan–Scott–Aschbacher Theorem ([Sco, AsS], cf. [DiM, Thm. 4.1A]). The proof of this theorem is elementary. In fact, according to Peter Neumann (cited by Peter Cameron [Cam11]) much of it already appears in Jordan's 1870 monograph [Jor].

Definition 10.2.1. The *socle* $\mathrm{Soc}(G)$ of a group G is the product of its minimal normal subgroups.

Fact 10.2.2. (i) The socle of any group is a direct product of simple groups.

(ii) The socle of a primitive permutation group is a direct product of isomorphic simple groups.

We need the following result, pointed out by Luks [Lu82].

Proposition 10.2.3. *Let $G \leq \mathfrak{S}_n$ be a primitive group with socle $\mathrm{Soc}(G) \cong R^s$ where R is a nonabelian simple group. Then $n \geq 5^s$.*

This result is an immediate consequence of the "summary" of the O'Nan–Scott–Aschbacher Theorem [Sco, AsS, LiePS] given by Dixon and Mortimer [DiM] as Theorem 4.1A. We further compress the result to suit our purposes.

Theorem 10.2.4 (Extracted from [DiM, Thm 4.1A])**.** *Let $G \leq \mathfrak{S}_n$ be a primitive group with socle $\mathrm{Soc}(G) \cong R^s$ where R is a nonabelian simple group. If $s \geq 2$ then either (a) $n \geq |R|^{s-1}$ or (b) there exists a proper divisor $d \mid s$ and a primitive group U with socle R^d such that $n = \ell^{s/d}$ where ℓ is the degree of U.*

Proof of Prop. 10.2.3. We use the elementary fact that the smallest nonabelian simple group is $\mathfrak{A}_5$. We proceed by induction on s.

For $s = 1$ we need to prove $n \geq 5$. This follows from the solvability of $\mathfrak{S}_4$.

Now assume $s \geq 2$. We have $|R|^{s-1} \geq |R|^{s/2} \geq 60^{s/2} > 5^s$, so in case (a) of Theorem 10.2.4 we are done. Assume we are in case (b) and let $U \leq \mathfrak{S}_\ell$ be the primitive group provided in this case. Since $d < s$, by induction we have $\ell \geq 5^d$ and therefore $n = \ell^{s/d} \geq 5^s$. □

Now we state our result.

Lemma 10.2.5. *Let $G \leq \mathfrak{S}(\Omega)$ be primitive. Assume $\varphi : G \to \mathfrak{A}_k$ is an epimorphism where $k > \max\{8, 2 + \log_2 n\}$. Then φ is an isomorphism; hence $G \cong \mathfrak{A}_k$.*

The proof depends on the CFSG through "Schreier's Hypothesis" (a known consequence of the CFSG) which states that the outer automorphism group of every finite simple group is solvable.

Proof. Let $N = \mathrm{Soc}(G)$. By Fact 10.2.2 N can be written as $N = R_1 \times \cdots \times R_s$ where the R_i are isomorphic simple groups. Case 1. N is abelian (the "affine case") and therefore regular, i. e., $n = |N|$. In this case $N \cong \mathbb{Z}_p^s$ and $G/N \leq \mathrm{GL}(s, p)$ for some prime p, so $n = p^s$. Moreover $\mathfrak{A}_k$ is involved in $\mathrm{GL}(s, p)$. It is shown in [BaPS, Prop. 1.22] that if $\mathfrak{A}_k$ is involved in $\mathrm{GL}(s, p)$ then, combining a result of Feit and Tits [FeT] with [KlL, Prop. 5.3.7], for $k \geq 9$ it follows that $k \leq s + 2$. But we have $s + 2 \leq 2 + \log_p n < k$, a contradiction, so this case cannot occur.

Case 2. N is nonabelian. By Prop. 10.2.3 we have $s \leq \log_5 n$. In particular, $k > s$.

Following [BaB][9], let $\mathrm{Pker}(G)$ ("permutation kernel") denote the kernel of the induced permutation action $G \to \mathfrak{S}_s$ (permuting the copies of R by conjugation by elements of G). Then $\mathrm{Pker}(G) \leq \mathrm{Aut}(R_1) \times \cdots \times \mathrm{Aut}(R_s)$. It follows that $\mathrm{Pker}(G)/N \leq \mathrm{Out}(R_1) \times \cdots \times \mathrm{Out}(R_s)$ is solvable by Schreier's Hypothesis.

Now $G/\mathrm{Pker}\, G \leq \mathfrak{S}_s$ and $s < k$ so $G/\mathrm{Pker}\, G$ cannot involve $\mathfrak{A}_k$. The solvable group $\mathrm{Pker}(G)/N$ also does not involve $\mathfrak{A}_k$. It follows that G/N does not involve $\mathfrak{A}_k$ and therefore $\ker\varphi \not\geq N$.

Let M be a minimal normal subgroup of G.

Case 2a. $M \neq N$. Then there is a unique other minimal normal subgroup, M^*, the centralizer of M, which is isomorphic to M. It follows that M is regular, so $n = |M|$. Moreover, s is even and $|M| = |\mathfrak{A}_k|^{s/2}$. Hence, $n \geq |\mathfrak{A}_k| > 2^k > n$, a contradiction. So this case cannot occur.

Case 2b. $M = N$ is the unique minimal normal subgroup of G. Since $N \not\leq \ker\varphi$, it follows that $\ker\varphi = 1$. □

Remark 10.2.6. The assumption $k > 2 + \log_2 n$ is tight infinitely often, as shown by the affine case of even dimension in characteristic 2. In this case $G = \mathbb{Z}_2^{k-2} \rtimes \mathfrak{A}_k$ acts primitively

[9]Introduced in [BaB] (1999), this notation was subsequently adopted in computational group theory (see. e. g., [HoS]).

on $n = 2^{k-2}$ elements as follows: $\mathfrak{A}_k$ acts on $\mathbb{Z}_2^k$ by permuting the coordinates; restrict this action to the zero-weight subspace $\sum x_i = 0$, and then to the quotient space by the one-dimensional subspace $x_1 = \cdots = x_k$ (this is contained in the zero-weight subspace when the dimension is even). In this case, $k = 2 + \log_2 n$, and $\mathfrak{A}_k$ is a proper quotient of G.

Remark 10.2.7. Under the stronger assumption $k > (\log n)^c$ for some constant c, Pyber [Py17] gave an elementary proof of the conclusion of Lemma 10.2.5.

10.3 Alternating quotients versus stabilizers: The Unaffected Stabilizers Lemma

Lemma 10.3.1. *Let $G \leq \mathfrak{S}(\Omega)$ be a transitive permutation group and $\varphi : G \to \mathfrak{A}_k$ an epimorphism where $k > \max\{8, 2 + \log_2 n\}$. Then $G_x^\varphi \neq \mathfrak{A}_k$ for any $x \in \Omega$.*

Proof. We proceed by induction on the order of G. Let $N = \ker \varphi$. Assume for a contradiction that $G_x^\varphi = \mathfrak{A}_k$, i. e., $NG_x = G$.

Let B be a maximal block of imprimitivity containing x (so $|B| < |\Omega|$). (If G is primitive then $B = \{x\}$.) So $G_B \geq G_x$ and therefore $NG_B = G$.

Let Ω' be the set of G-images of B. This is a system of imprimitivity on which G acts as a primitive group; let K be the kernel of this action.

Since N is a maximal normal subgroup of G, we have $K \leq N$ or $KN = G$.

If $K \leq N$ then φ maps the primitive group G/K onto $\mathfrak{A}_k$ and therefore by Lemma 10.2.5, $G/K \cong \mathfrak{A}_k$, hence $K = N$ and therefore $KG_B = G$. But obviously $G_B \geq K$, so $G = KG_B = G_B$ and therefore $|\Omega'| = 1$, i. e., $B = \Omega$, a contradiction.

So we have $KN = G$, i. e., $K^\varphi = \mathfrak{A}_k$. Let $\Omega_1, \dots, \Omega_m$ denote the orbits of K ($m \geq 2$). Let K_i denote the restriction of K to Ω_i and $M_i \triangleleft K$ the kernel of the $K \to K_i$ epimorphism. By Lemma 10.1.1, $(\exists i)(M_i \leq N)$. The set $(\Omega_1, \dots, \Omega_m)$ is a system of imprimitivity for G on which G acts transitively, so the M_i are conjugate subgroups in G and therefore $M_i \leq N$ for all i. Let $x \in \Omega_i$. It follows from $M_i \leq N$ that the epimorphism $K \to \mathfrak{A}_k$ (restriction of φ to K) factors across K_i as $K \to K_i \xrightarrow{\psi} \mathfrak{A}_k$, so $K_i^\psi = \mathfrak{A}_k$. By the inductive hypothesis, applied to K_i, we infer that $(K_i)_x^\psi \neq \mathfrak{A}_k$. On the other hand, $(K_i)_x^\psi = K_x^\varphi \triangleleft G_x^\varphi = \mathfrak{A}_k$. We conclude that $|K_x^\varphi| = 1$ and therefore $n > |x^K| = |K : K_x| \geq |K^\varphi : K_x^\varphi| = |K^\varphi| = k!/2 > 2^{k-2} > n$, a contradiction. ◻

Remark 10.3.2. Again, the assumption $k > 2 + \log_2 n$ is tight; the Lemma fails infinitely often if $k = 2 + \log_2 n$ is permitted. This is shown by the same examples as in Remark 10.2.6.

Next we extend Lemma 10.3.1 to not necessarily transitive groups.

Lemma 10.3.3. *Let $G \leq \mathfrak{S}(\Omega)$ be a permutation group and $\varphi : G \to \mathfrak{A}_k$ an epimorphism. Assume $k > \max\{8, 2 + \log_2 n_0\}$ where $n_0 = n_0(G)$ denotes the length of the largest orbit of G. Then $G_x^\varphi \neq \mathfrak{A}_k$ for some $x \in \Omega$.*

Proof. Let $\Omega_1, \dots, \Omega_m$ be the orbits of G and let G_i be the restriction of G to Ω_i. So G is a subdirect product of the G_i. Let M_i denote the kernel of the $G \to G_i$ epimorphism. By

Lemma 10.1.1, $(\exists i)(M_i \leq \ker\varphi)$, so φ factors across the restriction $G \to G_i$ as $G \to G_i \xrightarrow{\psi} \mathfrak{A}_k$. So $G_i^{\psi} = \mathfrak{A}_k$.

Let $x \in \Omega_i$. We apply Lemma 10.3.1 to G_i and notice that $G_x^{\varphi} = (G_i)_x^{\psi} \neq \mathfrak{A}_k$. □

The following result, Theorem 10.3.5, along with a companion observation, Cor. 10.3.7, will be the principal tools for our central algorithm, the LocalCertificates procedure. Recall that $G_{(D)}$ denotes the pointwise stabilizer of D in G ($D \subseteq \Omega$).

Definition 10.3.4 (Affected)**.** We say that the homomorphism $\varphi : G \to \mathfrak{S}_k$ is a *giant representation* if $G^{\varphi} \geq \mathfrak{A}_k$. We say that $x \in \Omega$ is *affected* by φ if $G_x^{\varphi} \ngeq \mathfrak{A}_k$.

Theorem 10.3.5 (Unaffected Stabilizers Lemma)**.** *Let $G \leq \mathfrak{S}(\Omega)$ be a permutation group and $\varphi : G \to \mathfrak{S}_k$ a giant representation. Assume $k > \max\{8, 2 + \log_2 n_0\}$ where $n_0 = n_0(G)$ denotes the length of the largest orbit of G. Let D be the set of elements of Ω not affected by φ. Then $G_{(D)}^{\varphi} \geq \mathfrak{A}_k$.*

Proof. First assume $G^{\varphi} = \mathfrak{A}_k$. The set D is G-invariant and $G_{(D)}$ is the kernel of the restriction map $G \to \mathfrak{S}(D)$. Let $P \leq \mathfrak{S}(D)$ be the image of this map (restriction of G to D), so $P \cong G/G_{(D)}$. Since $G_{(D)} \triangleleft G$, we have $G_{(D)}^{\varphi} \triangleleft G^{\varphi} = \mathfrak{A}_k$. Assume for a contradiction that $G_{(D)}^{\varphi} \neq \mathfrak{A}_k$; it follows that $|G_{(D)}^{\varphi}| = 1$, i. e., $G_{(D)} \leq \ker(\varphi)$. Hence φ factors across P as $G \to P \xrightarrow{\psi} \mathfrak{A}_k$. It follows that $P^{\psi} = G^{\varphi} = \mathfrak{A}_k$ so ψ is an epimorphism. By Lemma 10.3.3 we have $P_x^{\psi} \neq \mathfrak{A}_k$ for some $x \in D$. But $P_x^{\psi} = G_x^{\varphi} = \mathfrak{A}_k$ (because $x \in D$ is not affected by φ), a contradiction.

Now if $G^{\varphi} = \mathfrak{S}_k$ then let $G_1 = \varphi^{-1}(\mathfrak{A}_k)$. Let φ_1 be the restriction of φ to G_1, so $\varphi_1 : G_1 \to \mathfrak{A}_k$ is an epimorphism. Moreover, $x \in \Omega$ is affected by φ if and only if x is affected by φ_1 (because $\mathfrak{A}_k$ has no subgroup of index 2). An application of the previous case to (G_1, φ_1) completes the proof. □

Proposition 10.3.6. *Let $G \leq \mathfrak{S}(\Omega)$ be a permutation group and $\varphi : G \to H$ an epimorphism. Let $\Delta \subseteq \Omega$ be an orbit of G and $x \in \Delta$. Let $L = G_x^{\varphi}$. Then each orbit of* $\ker(\varphi)$ *in Δ has length $|\Delta|/k$ where $k = |H : L|$.*

Proof. First we note that k only depends on Δ, not on the specific element $x \in \Delta$ because if $y \in \Delta$ then G_x and G_y are conjugates in G. Now let $N = \ker(\varphi)$ and $|\Delta| = d$. So $d = |G : G_x|$. The length of the N-orbit x^N is $|N : N_x|$. We have $|G : NG_x| = |G^{\varphi} : G_x^{\varphi}| = |H : L| = k$. Therefore $|NG_x : G_x| = d/k$. But $|N : N_x| = |N : (N \cap G_x)| = |NG_x : G_x| = d/k$. □

Corollary 10.3.7 (Affected Orbit Lemma)**.** *Let $G \leq \mathfrak{S}(\Omega)$ be a permutation group and $\varphi : G \to \mathfrak{S}_k$ a giant representation. Assume $k \geq 5$. Then, if Δ is an affected G-orbit, i. e., $\Delta \cap D = \emptyset$, then* $\ker(\varphi)$ *is not transitive on Δ; in fact, each orbit of* $\ker(\varphi)$ *in Δ has length $\leq |\Delta|/k$.*

Proof. For $k \geq 5$, the largest proper subgroup of $\mathfrak{A}_k$ has index k, and the largest subgroup of $\mathfrak{S}_k$ not containing $\mathfrak{A}_k$ also has index k. So the statement follows from Prop. 10.3.6. □

Remark 10.3.8. If $k \geq \max\{9, 2\log_2 n_0\}$ then we can use Theorem 10.5.1 to make a more detailed statement. We observe that $\ker(\varphi)$ fixes each standard block (setwise) (see item (e) in Theorem 10.5.1) so the length of each orbit of $\ker(\varphi)$ contained in Δ is $\leq |\Delta|/\binom{k}{t_\Delta}$.

10.4 Subgroups of small index in $\mathfrak{S}_n$

Observation 10.4.1. Let $T, U \subset \Omega$, $|T|, |U| < n/2$, where $n = |\Omega| \geq 5$. Assume $\mathfrak{A}(\Omega)_{(T)} \leq \mathfrak{S}(\Omega)_U$. Then $U \subseteq T$.

Proof. By assumption, $|\Omega \setminus T| \geq 3$ and therefore $\Omega \setminus T$ is an orbit of $\mathfrak{A}(\Omega)_{(T)}$ so it must be part of an orbit of $\mathfrak{S}(\Omega)_U$. Since $|\Omega \setminus T| > n/2 > |U|$, we must have $\Omega \setminus T \subseteq \Omega \setminus U$, as claimed. □

According to Dixon and Mortimer [DiM, p. 176], the following result goes back to Jordan (1870) [Jor, pp. 68–75]; a modern treatment was given by Liebeck [Lie83, Lemma 1.1]. We cite from the version given in [DiM, Thm. 5.2A,B]. Uniqueness follows from Observation 10.4.1.

Theorem 10.4.2 (Jordan–Liebeck)**.** *Let $\mathfrak{A}(\Omega) \leq K \leq \mathfrak{S}(\Omega)$. Let $H \leq K$ and $1 \leq r < n/2$ where $n = |\Omega| \geq 9$. Assume $|K : H| < \binom{n}{r}$. Then there exists a unique $T \subset \Omega$ with $|T| < n/2$ such that $\mathfrak{A}(\Omega)_{(T)} \leq H \leq \mathfrak{S}(\Omega)_T$. This unique T satisfies $|T| < r$.*

Notation 10.4.3. Under the assumptions of Theorem 10.4.2 we write $T(H)$ for the unique subset $T \subset \Omega$ guaranteed by the Theorem. So we have

$$\mathfrak{A}(\Omega)_{(T(H))} \leq H \leq \mathfrak{S}(\Omega)_{T(H)}. \tag{51}$$

Remark 10.4.4. $T(H) = \emptyset$ if and only if $\mathfrak{A}(\Omega) \leq H \leq \mathfrak{S}(\Omega)$.

10.5 Large alternating quotient acts as a Johnson group on blocks: The Main Structure Theorem

Recall that the homomorphism $\varphi : G \to \mathfrak{S}_k$ is a *giant representation* if $G^\varphi \geq \mathfrak{A}_k$.

Theorem 10.5.1 (Main Structure Theorem)**.** *Let $G \leq \mathfrak{S}(\Omega)$ be a permutation group and $\varphi : G \to \mathfrak{S}_k$ a giant representation. Assume $k \geq \max\{9, 2\log_2 n_0\}$ where $n_0 = n_0(G)$ denotes the length of the largest orbit of G.*

(a) For every $x \in \Omega$ there exists a unique subset $T(x) \subset [k]$ such that $|T(x)| < k/4$ and

$$(\mathfrak{A}_k)_{(T(x))} \leq G_x^\varphi \leq (\mathfrak{S}_k)_{T(x)}. \tag{52}$$

(b) The element $x \in \Omega$ is affected by φ if and only if $|T(x)| \geq 1$.

(c) For each orbit Δ there is an integer $t_\Delta \geq 0$ such that $|T(x)| = t_\Delta$ for every $x \in \Delta$. We say that Δ is affected by φ *if $t_\Delta \geq 1$, i. e., the elements of Δ are affected.*

(d) At least one orbit is affected. In fact, if D is the union of the unaffected blocks then $G^\varphi_{(D)} \geq \mathfrak{A}_k$.

(e) (Johnson group action on blocks in affected orbits) *For every orbit Δ the equivalence relation $T(x) = T(y)$ $(x, y \in \Delta)$ splits Δ into $\binom{k}{t_\Delta}$ blocks of imprimitivity, labeled by the t_Δ-subsets of $[k]$. We refer to these blocks as the* standard blocks *for φ. The action of G on the set of standard blocks in Δ is $\mathfrak{A}_k^{(t_\Delta)}$ or $\mathfrak{S}_k^{(t_\Delta)}$. If $t_\Delta \geq 1$ then this is a Johnson group and the kernel of this action is* $\ker\varphi$*; if $t_\Delta = 0$ then the action is trivial (its kernel is G, there is just one block, namely Δ). — In particular, by item* (d)*, we have a Johnson group action on the set of standard blocks in at least one orbit.*

(f) If $B \subseteq \Delta$ is a standard block and $x \in B$ then $G_B^\varphi = (G^\varphi)_{T(x)}$ (so it is either $(\mathfrak{S}_k)_{T(x)}$ or $(\mathfrak{A}_k)_{T(x)}$).

(g) If $\Psi = \{C_1, \ldots, C_r\}$ is another system of imprimitivity on the orbit Δ such that the kernel of the action $G \to \mathfrak{S}(\Psi)$ is $\ker(\varphi)$ *then $r = \binom{k}{t'}$ for some $t' < t_\Delta$ and the G-action on Ψ is $\mathfrak{S}_k^{(t')}$ or $\mathfrak{A}_k^{(t')}$. In particular, the standard blocks form the unique largest system of imprimitivity on which the kernel of G-action is* $\ker(\varphi)$*. Moreover, if $x \in C_i$ then $|T(G_{C_i})| = t'$ and $T(G_{C_i}) \subset T(x)$.*

Proof. Item (a) follows from the Jordan–Liebeck theorem (Thm. 10.4.2), setting $K = G^\varphi$ and $H = G_x^\varphi$ (so $T(x) = T(G_x^\varphi)$) and noting that

$$\binom{k}{\lfloor k/4 \rfloor} > 2^{k/2} \geq n_0 \geq |x^G| = |G : G_x| \geq |G^\varphi : G_x^\varphi|. \tag{53}$$

Item (b) is immediate from Eq. (52) and the definition of being "affected."

Item (c) follows from the observation that for $x \in \Omega$ and $\sigma \in G$ we have

$$G_{x^\sigma} = G_x^\sigma \quad \text{and therefore} \quad T(x^\sigma) = T(x)^{\sigma^\varphi}. \tag{54}$$

Item (d) is of greatest importance; it is the content of the "Unaffected Stabilizers Lemma" (Thm. 10.3.5).

To see Item (e), let Δ be an orbit and let $[x]$ denote the equivalence class (block) of $x \in \Delta$ under the equivalence relation stated. By Eq. (54), this equivalence relation is G-invariant and G acts transitively on the blocks. We also infer from Eq. (54) that the blocks in Δ are in 1-to-1 correspondence with the t_Δ-subsets of $[k]$ (noting that $\mathfrak{A}_k$ acts transitively on $\binom{[k]}{t_\Delta}$). Moreover, through this bijection, the G-action on the blocks in Δ is equivalent to the action of $\mathfrak{A}_k$ on $\binom{[k]}{t_\Delta}$. This bijection also proves item (f).

To see item (g), first we note that $r \geq 3$ (in fact, $r \geq k$) because the kernel of the action on Ψ has index $\geq k!/2$ and therefore $r! \geq k!/2$. Let $x \in C_i$ and $H = G_{C_i}$. So $G_x \leq H$ and H is a maximal subgroup of G of index ≥ 3. Let $N = \ker(\varphi)$; so $N \leq H$ and $3 \leq |G : H| = |G^\varphi : H^\varphi|$. Moreover, H^φ is a maximal subgroup of $\mathfrak{S}_k$ or $\mathfrak{A}_k$ containing $G_x^\varphi \geq (\mathfrak{A}_k)_{(T(x))}$. For $T \subset [k]$ with $|T| < k/2$, the only maximal subgroups of $\mathfrak{S}_k$ containing $(\mathfrak{A}_k)_{(T)}$ are of the form $(\mathfrak{S}_k)_U$ for $U \subseteq T$. Intersecting these with $\mathfrak{A}_k$ we obtain the maximal subgroups of $\mathfrak{A}_k$ containing $(\mathfrak{A}_k)_{(T)}$. This proves that $T(G_{C_i}) \subset T(x)$. Setting $t' = |T(G_{C_i})|$, the corresponding Johnson group action on Ψ follows the lines of the proof of item (e). $\square$

Remark 10.5.2 (Tight bound for k)**.** The actual condition on k, sufficient for most conclusions of the theorem, is that $k > \max\{8, 2 + \log_2 n_0\}$ and $\frac{1}{2}\binom{k}{\lfloor k/2 \rfloor} > n_0$. The latter translates to $k > \log_2 n_0 + (1/2 + o(1)) \log_2 \log_2 n_0$. The only difference would be that instead of $|T(x)| < k/4$ we would only get $|T(x)| < k/2$, sufficient for our goals.

Our assumption $k \geq \max\{9, 2\log_2 n_0\}$ is generously sufficient for both conditions above. Under this condition we shall not only have $|T(x)| < k/4$ but $|T(x)| < H^{-1}(1/2)(1+o(1))k < k/9$ (for large k). Here $H(x)$ is the binary entropy function, so $H^{-1}(1/2) \approx 0.11003 < 1/9$. — We note that any bound of the form $k > c \log n_0$ would work for the purposes of this paper; the actual value of c will not affect our complexity estimate.

Remark 10.5.3 (Multiple systems of imprimitivity)**.** The presence of multiple systems of imprimitivity with the same kernel as discussed in Item (g) is a real possibility. Consider for instance the action $\mathfrak{S}_k \to \mathfrak{S}_{k(k-1)}$ defined by the action of $\mathfrak{S}_k$ on the $k(k-1)$ ordered pairs; let $G \leq \mathfrak{S}_{k(k-1)}$ be the image of this action. Then G has two systems of imprimitivity on which $\mathfrak{S}_k$ acts in its natural action (there are k blocks in each system), and there is a unique system of imprimitivity with $\binom{k}{2}$ blocks on which the action is $\mathfrak{S}_k^{(2)}$. The latter are the *standard blocks*; in this case each standard block has 2 elements. Each of the three actions is faithful, so their kernel is the same, namely, the idenity.

Finally, and algorithmic observation.

Proposition 10.5.4. *Given a giant representation $\varphi : G \to \mathfrak{S}_k$, we can find the standard blocks in each G-orbit in polynomial time.*

Proof. Standard. ☐

11 Algorithmic setup

11.1 Luks's framework

In this section we review Luks's framework using notation and terminology that better suits our purposes.

Let $G \leq \mathfrak{S}(\Omega)$ be a permutation group acting on the domain Ω. G will be represented concisely by a list of generators; if $|\Omega| = n$ then every minimal set of generators has $\leq 2n$ elements [Ba86].

Let Σ be a finite alphabet. We consider the set of strings $\mathfrak{x}$ over the alphabet Σ indexed by Ω, i. e., mappings $\mathfrak{x} : \Omega \to \Sigma$. For $\tau \in \mathfrak{S}(\Omega)$ and $\mathfrak{x} : \Omega \to \Sigma$ we define the string $\mathfrak{x}^\tau$ by setting $\mathfrak{x}^\tau(u) = \mathfrak{x}(u^{\tau^{-1}})$ for all $u \in \Omega$. In other words, for all $u \in \Omega$ and $\tau \in \mathfrak{S}(\Omega)$,

$$\mathfrak{x}^\tau(u^\tau) = \mathfrak{x}(u). \tag{55}$$

(The purpose of the inversion is to ensure that $\mathfrak{x}^{\sigma\tau} = (\mathfrak{x}^\sigma)^\tau$ for $\sigma, \tau \in \mathfrak{S}(\Omega)$.)

For $K \subseteq \mathfrak{S}(\Omega)$ we say that τ is a *K-isomorphism* of strings $\mathfrak{x}$ and $\mathfrak{y}$ if $\tau \in K$ and $\mathfrak{x}^\tau = \mathfrak{y}$. Let $\mathrm{Iso}_K(\mathfrak{x}, \mathfrak{y})$ denote the set of K-isomorphisms of $\mathfrak{x}$ and $\mathfrak{y}$:

$$\mathrm{Iso}_K(\mathfrak{x}, \mathfrak{y}) = \{\tau \in K \mid \mathfrak{x}^\tau = \mathfrak{y}\} = \{\tau \in K \mid (\forall u \in \Omega)(\mathfrak{x}(u) = \mathfrak{y}(u^\tau)\} \tag{56}$$

and let $\mathrm{Aut}_K(\mathfrak{x}) = \mathrm{Iso}_K(\mathfrak{x}, \mathfrak{x})$ denote the set of K-automorphisms of $\mathfrak{x}$.

Remark 11.1.1. The only context in which we use this concept is when K is a coset. However, the general principles are more transparent in this more general context.

In the Introduction we stated the String Isomorphism decision problem: "Is $\mathrm{Iso}_G(\mathfrak{x}, \mathfrak{y})$ not empty?" In the rest of the paper we shall use the term "String Isomorphism problem" for the computation version (compute the set $\mathrm{Iso}(\mathfrak{x}, \mathfrak{y})$). The decision and computation versions are polynomial-time equivalent (under Cook reductions).

Definition 11.1.2 (String Isomorphism Problem)**.**

Input: a set Ω, a finite alphabet Σ, two strings $\mathfrak{x}, \mathfrak{y} : \Omega \to \Sigma$,
a permutation group $G \leq \mathfrak{S}(\Omega)$ (given by a list of generators)

Output: the set $\mathrm{Iso}_G(\mathfrak{x}, \mathfrak{y})$. If this set is nonempty, it is represented by a list of generators of the group $\mathrm{Aut}_G(\mathfrak{x})$ and a coset representative $\sigma \in \mathrm{Iso}_G(\mathfrak{x}, \mathfrak{y})$.

For $K \subseteq \mathfrak{S}(\Omega)$ and $\sigma \in \mathfrak{S}(\Omega)$ we note the *shift identity*

$$\mathrm{Iso}_{K\sigma}(\mathfrak{x}, \mathfrak{y}) = \mathrm{Iso}_K(\mathfrak{x}, \mathfrak{y}^{\sigma^{-1}})\sigma. \tag{57}$$

For the purposes of recursion we need to introduce one more variable, a subset $\Delta \subseteq \Omega$ to which we shall refer as the *window*.

Definition 11.1.3 (Window isomorphism)**.** Let $\Delta \subseteq \Omega$ and $K \subset \mathfrak{S}(\Omega)$. Let

$$\mathrm{Iso}_K^{\Delta}(\mathfrak{x}, \mathfrak{y}) = \{\tau \in K \mid (\forall u \in \Delta)(\mathfrak{x}(u) = \mathfrak{y}(u^\tau))\}. \tag{58}$$

For $K \subseteq \mathfrak{S}(\Omega)$ and $\sigma \in \mathfrak{S}(\Omega)$ we again have the *shift identity:*

$$\mathrm{Iso}_{K\sigma}^{\Delta}(\mathfrak{x}, \mathfrak{y}) = \mathrm{Iso}_K^{\Delta}(\mathfrak{x}, \mathfrak{y}^{\sigma^{-1}})\sigma. \tag{59}$$

Remark 11.1.4 (Alignment)**.** Applying Eq. (57) to a *subgroup* $K = G \leq \mathfrak{S}(\Omega)$, we see that the isomorphism problem for the pair $(\mathfrak{x}, \mathfrak{y})$ of strings with respect to a coset $G\sigma$ is the same as the isomorphism problem for $(\mathfrak{x}, \mathfrak{y}^{\sigma^{-1}})$ with respect to the group G. In view of Eq. (59), the same holds for window-isomorphism. The shift $\mathfrak{y} \leftarrow \mathfrak{y}^{\sigma^{-1}}$ is an important **alignment step** that will accompany every reduction of the ambient group G.

Remark 11.1.5. When applying the concept of window-isomorphism, we shall always assume that the window is *invariant* under the group $G \leq \mathfrak{S}(\Omega)$, and K is a coset, $K = G\sigma$ for some $\sigma \in \mathfrak{S}(\Omega)$. Under these circumstances we make the following observations.

(i) $\mathrm{Aut}_G^{\Delta}(\mathfrak{x})$ is a subgroup of G

(ii) $\mathrm{Iso}_{G\sigma}^{\Delta}(\mathfrak{x}, \mathfrak{y})$ is either empty or a right coset of $\mathrm{Aut}_G^{\Delta}(\mathfrak{x})$, namely,

$$\mathrm{Iso}_{G\sigma}^{\Delta}(\mathfrak{x}, \mathfrak{y}) = \mathrm{Aut}_G^{\Delta}(\mathfrak{x})\sigma' \quad \text{for any} \quad \sigma' \in \mathrm{Iso}_{G\sigma}^{\Delta}(\mathfrak{x}, \mathfrak{y})\,. \tag{60}$$

However, again, the general principles are more transparent in the more general context where K is an arbitrary subset of $\mathfrak{S}(\Omega)$ and Δ is an arbitrary subset of Ω.

The following straighforward identity plays a central role in Luks's method. Let $K, L \subseteq \mathfrak{S}(\Omega)$ and $\Delta \subseteq \Omega$. Then

$$\mathrm{Iso}^{\Delta}_{K\cup L}(\mathfrak{x}, \mathfrak{y}) = \mathrm{Iso}^{\Delta}_{K}(\mathfrak{x}, \mathfrak{y}) \cup \mathrm{Iso}^{\Delta}_{L}(\mathfrak{x}, \mathfrak{y}) \tag{61}$$

Next we describe Luks's group-theoretic divide-and-conquer strategies.

Proposition 11.1.6 (Descent)**.** *Let $H \le G$. Then finding $\mathrm{Iso}^{\Delta}_{G}(\mathfrak{x}, \mathfrak{y})$ reduces to $|G : H|$ instances of finding $\mathrm{Iso}^{\Delta}_{H}(\mathfrak{x}, \mathfrak{y}^{\sigma})$ for various $\sigma \in G$.*

Proof. We can write $G = \bigcup_{\sigma} H\sigma$ where σ ranges over a set of right coset representatives of H in G. Applying Eq. (61) to this decomposition, we obtain

$$\mathrm{Iso}^{\Delta}_{G}(\mathfrak{x}, \mathfrak{y}) = \bigcup_{\sigma} \mathrm{Iso}^{\Delta}_{H\sigma}(\mathfrak{x}, \mathfrak{y}) = \bigcup_{\sigma} \mathrm{Iso}^{\Delta}_{H}(\mathfrak{x}, \mathfrak{y}^{\sigma^{-1}})\sigma \tag{62}$$

where we also employed the shift identity, Eq. (59). □

The following identity describes Luks's basic recurrence for sequential processing of windows.

Proposition 11.1.7 (Chain Rule)**.** *Let Δ_1 and Δ_2 be G-invariant subsets of Ω and let $\mathrm{Iso}^{\Delta_1}_{G}(\mathfrak{x}, \mathfrak{y}) = G_1\sigma$, where $\sigma \in G$ and $G_1 \le G$. Then*

$$\mathrm{Iso}^{\Delta_1\cup\Delta_2}_{G}(\mathfrak{x}, \mathfrak{y}) = \mathrm{Iso}^{\Delta_2}_{G_1\sigma}(\mathfrak{x}, \mathfrak{y}) = \mathrm{Iso}^{\Delta_2}_{G_1}(\mathfrak{x}, \mathfrak{y}^{\sigma^{-1}})\sigma. \tag{63}$$

Proof. The first equation is immediate from the definitions. The second equation uses the shift identity, Eq. (59). □

We can now describe what we call "strong descent." We assume G is transitive. "Strong descent" begins with descent to an intransitive subgroup $N \le G$ followed by an application of the Chain rule to the orbit partition of N. We shall apply this to normal subgroups N.

Fact 11.1.8. If G is a transitive group and $N \triangleleft G$ then the orbits of N have equal length.

It follows that if N is intransitive then each orbit of N has length n/m for some $m \ge 2$, so we obtain a recurrence of the form

$$T(n) \le |G : N| m T(n/m) \tag{64}$$

for some $m \ge 2$.

Luks applied this combination to the case when G is transitive, imprimitive, and N is the kernel of the G-action on the set of blocks. We next describe this case in detail.

Recall the restriction notation G^{Δ} (Notation 2.2.1).

Theorem 11.1.9 (Imprimitive Luks reduction)**.** *Let $G \le \mathfrak{S}(\Omega)$ and let $\Delta \subseteq \Omega$ be a G-invariant subset. Let $\{B_1, \dots, B_m\}$ be a G-invariant partition of Δ. Let $\psi : G \to \overline{G} \le \mathfrak{S}_m$ be the induced action of G on the set of blocks and let $N = \ker(\psi)$. Then finding $\mathrm{Iso}^{\Delta}_{G}(\mathfrak{x}, \mathfrak{y})$ reduces to $m|\overline{G}| = m|G/N|$ instances of finding $\mathrm{Iso}^{B_i}_{M_i}(\mathfrak{x}, \mathfrak{y}^{\sigma_i})$ for the blocks B_i and certain subgroups $M_i \le N$ and $\sigma_i \in G$.*

(The cost of the reduction is polynomial per instance.)

Proof. First descend to $N = \ker \psi$. Then consider each B_i to be the window in succession, reducing the group at each round, following the Chain Rule. In the end, combine all the results into a single coset. $\square$

Luks applied this reduction with great effect to minimal systems of imprimitivity (systems with at least two blocks that cannot be made coarser, i. e., the blocks are maximal) so $\overline{G}$ is a primitive group. Therefore the order of primitive groups involved in G (action of subgroups on a system of blocks of imprimitivity of the subgroup) is a critical parameter of the performance of Luks reduction.

We note that in our core "Local certificates" algorithm we shall employ strong descent in a context other than the imprimitive Luks reduction (see Procedure Recompute $H(W)$ in Section 13.1).

A final observation: when trying to determine $\mathrm{Iso}_G^{\Delta}(\mathfrak{x}, \mathfrak{y})$, it suffices to consider the case $\Delta = \Omega$ (Obs. 11.1.11 below).

Definition 11.1.10 (Straight-line program)**.** Given a group G by a list S of generators, a *straight-line program* of length ℓ in G is a sequence of length ℓ of elements of G such that each element in the sequence is either one of the generators or is a product of two elements earlier in the sequence or is the inverse of an element earlier in the sequence. We say that the straight-line program *computes* a set T of elements if the elements of T appear in the sequence and are marked as belonging to T. A subgroup is computed if a set of generators of the subgroup is computed. A coset is computed if the corresponding subgroup and a coset representative are computed.

Observation 11.1.11 (Reducing to the window)**.** Let $G \le \mathfrak{S}(\Omega)$ and let Δ be a G-invariant subset of Ω. Let $\mathfrak{x}^{\Delta}$ and $\mathfrak{y}^{\Delta}$ be the restriction of $\mathfrak{x}$ and $\mathfrak{y}$ to Δ, respectively. Given a straight-line program of length ℓ that computes $\mathrm{Iso}_{G^{\Delta}}(\mathfrak{x}^{\Delta}, \mathfrak{y}^{\Delta})$, we can, in time $O(n\ell) + \mathrm{poly}(n)$, compute $\mathrm{Iso}_G^{\Delta}(\mathfrak{x}, \mathfrak{y})$ (where $n = |\Omega|$).

Proof. While we concentrate on the action of the elements of G on the window, we maintain their "tails" – their action on the rest of the permutation domain. The set $\mathrm{Iso}_{G^{\Delta}}(\mathfrak{x}^{\Delta}, \mathfrak{y}^{\Delta})$ is empty if and only if $\mathrm{Iso}_G^{\Delta}(\mathfrak{x}, \mathfrak{y})$ is empty. If $\mathrm{Iso}_{G^{\Delta}}(\mathfrak{x}^{\Delta}, \mathfrak{y}^{\Delta})$ is not empty, in the end we obtain a subset $S \subseteq G$ and an element $\sigma \in G$ such that the restiction of the elements of S to Δ generates $\mathrm{Aut}_{G^{\Delta}}^{\Delta}(\mathfrak{x})$ and the restiction of σ to Δ belongs to $\mathrm{Iso}_{G^{\Delta}}(\mathfrak{x}^{\Delta}, \mathfrak{y}^{\Delta})$. Adding to S a set of generators of the kernel of the G-action on Δ we obtain a set of generators of $\mathrm{Aut}_G^{\Delta}(\mathfrak{x})$; and $\mathrm{Iso}_G^{\Delta}(\mathfrak{x}, \mathfrak{y}) = \mathrm{Aut}_G^{\Delta}(\mathfrak{x})\sigma$. $\square$

Once again we stress that everything in this section was a review of Luks's work.

11.2 Johnson groups are the only barrier

The barriers to efficient application of Luks's reductions are large primitive groups involved in G.

The following result reduces the Luks barriers to the class of Johnson groups at a multiplicative cost of $\le n$.

Theorem 11.2.1. *Let $G \leq \mathfrak{S}_n$ be a primitive group of order $|G| \geq n^{1+\log_2 n}$ where n is greater than some absolute constant. Then G has a normal subgroup N of index $\leq n$ such that N has a system of imprimitivity on which N acts as a Johnson group $\mathfrak{A}_k^{(t)}$ with $k \geq \log_2 n$. Moreover, N and the system of imprimitivity in question can be found in polynomial time.*

The mathematical part of this result is an immediate consequence of Cameron's classification of large primitive groups which we state below.

The *socle* $\mathrm{Soc}(G)$ of the group G is defined as the product of its minimal normal subgroups. $\mathrm{Soc}(G)$ can be written as $\mathrm{Soc}(G) = R_1 \times \cdots \times R_s$ where the R_i are isomorphic simple groups.

Definition 11.2.2. $G \leq \mathfrak{S}_n$ is a *Cameron group* with parameters $s, t \geq 1$ and $k \geq \max(2t, 5)$ if for some $s, t \geq 1$ and $k > 2t$ we have $n = \binom{k}{t}^s$, the socle of G is isomorphic to $\mathfrak{A}_k^s$, and $(\mathfrak{A}_k^{(t)})^s \leq G \leq \mathfrak{S}_k^{(t)} \wr \mathfrak{S}_s$ (wreath product, product action), moreover the induced action $G \to \mathfrak{S}_s$ on the direct factors of the socle is transitive.

Note that for $k \geq 5$ the Johnson groups $\mathfrak{S}_k^{(t)}$ and $\mathfrak{A}_k^{(t)}$ are exactly the Cameron groups with $s = 1$.

Theorem 11.2.3 (Cameron [Cam81], Maróti [Mar])**.** *For $n \geq 25$, if G is primitive and $|G| \geq n^{1+\log_2 n}$ then G is a Cameron group.*

We can further reduce Cameron groups to Johnson groups.

Proposition 11.2.4. *If $G \leq \mathfrak{S}_n$ is a Cameron group with parameters k, t, s then $ts \leq \log_2 n$. Moreover, $s \leq \log n / \log k \leq \log n / \log 5$.*

Proof. We have $n = \binom{k}{t}^s \geq (k/t)^{ts} \geq 2^{ts}$. Moreover, $n = \binom{k}{t}^s \geq k^s$. □

Proposition 11.2.5. *If $G \leq \mathfrak{S}_n$ is a Cameron group with parameters k, t, s and $|G| \geq n^{1+\log_2 n}$ then $k \geq \log_2 n$ and $s! < n$, assuming n is greater than an absolute constant.*

Proof. As before, we have $n \geq k^s$. On the other hand $n^{1+\log_2 n} \leq |G| \leq (k!)^s s! < k^{ks} s! \leq n^k s! < n^k (\log_2 n)^{\log_2 n} = n^{k+\log_2 \log_2 n}$. Therefore $k > \log_2 n - \log_2 \log_2 n > \log_2 n / \log 5 \geq s$. Hence, $s! < s^s \leq k^s \leq n$. Moreover, $n^{1+\log_2 n} < n^k s! < n^{k+1}$, hence $k \geq \log_2 n$. □

This completes the proof of the mathematical part of Theorem 11.2.1. The algorithmic part is well known: Cameron groups can be recognized and their structure mapped out in polynomial time (and even in NC [BaLS]).

11.3 Reduction to Johnson groups

We summarize the reduction to Johnson groups.

Procedure Reduce-to-Johnson

Input: group $G \leq \mathfrak{S}(\Omega)$, strings $\mathfrak{x}, \mathfrak{y} : \Omega \to \Sigma$

Output: $\mathrm{Iso}_G(\mathfrak{x}, \mathfrak{y})$ or updated $\Omega, G, \mathfrak{x}, \mathfrak{y}$, G transitive, with set $\mathcal{B}$ of blocks on which G acts as Johnson group $\mathfrak{G} \leq \mathfrak{S}(\mathcal{B})$

1. **if** $G \le \mathrm{Aut}(\mathfrak{x})$ **then**

   if $\mathfrak{x} = \mathfrak{y}$ **then return** $\mathrm{Iso}_G(\mathfrak{x}, \mathfrak{y}) = G$, **exit**

   else return $\mathrm{Iso}_G(\mathfrak{x}, \mathfrak{y}) = \emptyset$, **exit**

2. **if** $|G| < C_0$ for some absolute constant C_0 **then** compute $\mathrm{Iso}_G(\mathfrak{x}, \mathfrak{y})$ by brute force, **exit**

3. **if** G intransitive **then** apply Chain Rule

4. (: G transitive :) Find minimal block system $\mathcal{B}$. Let $m = |\mathcal{B}|$. Let $\mathfrak{G} \le \mathfrak{S}(\mathcal{B})$ be the induced G-action on $\mathcal{B}$ and N the kernel of the $G \to \mathfrak{G}$ epimorphism (: $\mathfrak{G}$ is a primitive group :)

5. **if** $|\mathfrak{G}| < m^{1+\log_2 m}$ then reduce G to N via imprimitive Luks reduction

6. **else** (: $\mathfrak{G}$ a Cameron group of order $\ge m^{1+\log_2 m}$:) reduce $\mathfrak{G}$ to Johnson group via Luks descent (: Theorem 11.2.1, multiplicative cost $\le m$:)

7. (: $\mathfrak{G}$ a Johnson group :)
   return $\Omega, G, \mathcal{B}, \mathfrak{G}$ (Johnson group), $\mathfrak{x}, \mathfrak{y}$

Our contribution is a ProcessJohnsonAction routine that takes the output of the last line as input. The paper is devoted to this algorithm; it is summarized in the Master Algorithm, starting with line 2 of that algorithm (Sec. 15).

11.4 Cost estimate

We describe the recurrent estimate of the cost.

By the cost of the algorithm we mean the number of group operations performed on the domain Ω.

For a real number $x \ge 1$, let $T(x)$ denote the worst-case cost of solving String Isomorphism for strings of length $\le x$. Let $T_{\mathrm{trans}}(x)$ denote the same quantity restricted to transitive groups and $T_{\mathrm{Jh}}(x)$ the same quantity further restricted to the case when G acts on a minimal system of imprimitivity as a Johnson group of order $\ge m^{1+\log_2 m}$ where m is the number of blocks ($2 \le m \le x$). We obtain the following recurrences. Here $p(x)$ denotes a polynomial, representing the overhead incurred in the reductions. C_1 is an absolute constant. For $x < 2$ we set $T(x) = T_{\mathrm{trans}}(x) = 1$. For $x \ge C_0$ (an absolute constant), Luks reductions yield the following recurrences:

(i) $T(x) \le \max\{\sum T_{\mathrm{trans}}(n_i) + p(x)\}$, where the maximum is taken over all partitions of $\lfloor x \rfloor$ as $\lfloor x \rfloor = \sum_i n_i$ into positive integers n_i, including the trivial partition $n_1 = \lfloor x \rfloor$ (Chain Rule)

(ii) $T_{\mathrm{trans}}(x) \le \max\{m^{2+\log_2 m}(T(x/m) + p(x)), m(T_{\mathrm{Jh}}(x) + p(x))\}$, where the maximum is taken over all m where $2 \le m \le x$ (imprimitive Luks reduction; $m = n \le x$ covers the case when G is primitive)

Assume we are looking for an upper bound $T_1(x)$ on $T(x)$ that satisfies $T_1(x) \geq x^{c \log_2 x}$ for some constant $c > 1$ and is a "nice" function in the sense that $\log\log T_1(x)/\log\log x$ is monotone nondecreasing for sufficiently large x. In this case we can replace item (i) by

(i') $T(x) \leq 1.1 T_{\mathrm{trans}}(x)$.

(The factor 1.1 absorbs the additive polynomial term.) Moreover, we can ignore the first part of the right-hand side of Eq. (ii) since $T_1(x)$ automatically satisfies $T_1(x) \geq m^{2+\log_2 n}(T_1(x/m) + p(x))$ (for all m, $2 \leq m \leq x$, assuming x is sufficently large), so we only need to assume

(ii') $T_{\mathrm{trans}}(x) \leq 1.1 x T_{\mathrm{Jh}}(x)$.

(Again, the factor 1.1 absorbs the additive polynomial term.) Combining inequalities (i') and (ii') we obtain

(iii) $T(x) \leq 2x T_{\mathrm{Jh}}(x)$.

Our contribution is an inequality of the form

$$T_{\mathrm{Jh}}(x) \leq q(x) T(4x/5), \tag{65}$$

where $q(x)$ is a quasipolynomial function. Combining with item (iii) we obtain

$$T(x) \leq 2xq(x)T(4x/5) < q(x)^2 T(4x/5) \tag{66}$$

which resolves to $T(x) = q(x)^{O(\log x)}$, yielding the desired quasipolynomial bound on $T(x)$.

Definition 11.4.1. We refer to $(G, \mathcal{B})$ as the *Johnson case* if G is a transitive group with a system $\mathcal{B}$ of imprimitivity such that G acts on $\mathcal{B}$ as a Johnson group $\mathfrak{S}_k^{(t)}$ or $\mathfrak{A}_k^{(t)}$. We refer to k as the *Johnson parameter*.

To prove Eq. (65), we define a finer complexity estimate that involves the Johnson parameter.

For real numbers $x \geq y \geq 5$, let $T_{\mathrm{Jh}}(x, y)$ denote the maximum cost of solving all Johnson cases with $n \leq x$ and Johnson parameter $\ell(x) \leq k \leq y$ for some specific polylogarithmic function $\ell(x)$. For $y < \max\{5, \ell(x)\}$ we set $T(x, y) = 0$. We obtain recurrences of the form

(iv) $T_{\mathrm{Jh}}(x) = T_{\mathrm{Jh}}(x, x)$

(v) $T_{\mathrm{Jh}}(x, y) \leq q_1(x)\,(T(4x/5) + T_{\mathrm{Jh}}(x, 0.9y))$

where $q_1(x)$ is a quasipolynomial function. An upper bound of the form $T_{\mathrm{Jh}}(x, y) \leq T(4x/5) q_1(x)^{O(\log y)}$ follows, hence Eq. (65) with $q(x) = q_1(x)^{O(\log x)}$ and therefore

$$T(x) = q_1(x)^{O(\log^2 x)}. \tag{67}$$

Explanation of item (v): we shall either reduce the domain (window) size n by a positive fraction, or reduce the Johnson parameter k by a positive fraction while not increasing n, at quasipolynomial multiplicative cost. These reductions are covered under our concept of "symmetry breaking."

12 Verification of top action

In this section we show that if $\varphi : G \to \mathfrak{S}(\Gamma)$ is a giant representation then we can recognize whether φ maps $\mathrm{Aut}_G(\mathfrak{x})$ onto a giant and if so can find $\mathrm{Iso}_G(\mathfrak{x}, \mathfrak{y})$, all this at the cost of $O(m)$ calls to String Isomorphism on windows of size $\leq n/m$, where $m = |\Gamma|$. Note that the solution to the recurrence $f(n) = O(mf(n/m) + n^c)$ is $f(n) \leq n^{c+1+o(1)}$ assuming $m \to \infty$ as $n \to \infty$ and c is a constant.

Proposition 12.0.1 (Lifting)**.** *Let $G \leq \mathfrak{S}(\Omega)$ and $H \leq \mathfrak{S}(\Gamma)$ be permutation groups, $\varphi : G \to H$ a homomorphism, and $N = \ker(\varphi)$. Given these data, the strings $\mathfrak{x}, \mathfrak{y} : \Omega \to \Sigma$ and $\overline{\sigma} \in H$, one can reduce, in polynomial time, the computation of the set $\varphi^{-1}(\overline{\sigma}) \cap \mathrm{Iso}_G(\mathfrak{x}, \mathfrak{y})$ (set of liftings of $\overline{\sigma}$ to isomorphisms) to a single call to $\mathrm{Iso}_N(\mathfrak{x}', \mathfrak{y})$ for some string $\mathfrak{x}'$.*

Proof. If $\overline{\sigma} \notin G^\varphi$ then return "empty." Otherwise let $\sigma \in \varphi^{-1}(\overline{\sigma})$. (We can find such a σ in polynomial time by Prop. 2.2.6.) Let $\mathfrak{x}' = \mathfrak{x}^\sigma$. Then $\varphi^{-1}(\overline{\sigma}) = \sigma N$ and $\mathrm{Iso}_G(\mathfrak{x}, \mathfrak{y}) = \sigma \, \mathrm{Iso}_G(\mathfrak{x}', \mathfrak{y})$. Consequently, $\varphi^{-1}(\overline{\sigma}) \cap \mathrm{Iso}_G(\mathfrak{x}, \mathfrak{y}) = \sigma \left(N \cap \mathrm{Iso}_G(\mathfrak{x}', \mathfrak{y}) \right) = \sigma \, \mathrm{Iso}_N(\mathfrak{x}', \mathfrak{y})$. ☐

Remark 12.0.2. H does not need to be a permutation group. What we need is that H permit constructive membership testing, i. e., for any list of elements $\tau_1, \ldots, \tau_k, \rho \in H$ we should be able to efficiently decide whether $\rho \in K$ where K is the subgroup generated by the τ_i, and if the answer is affirmative, to produce a straight-line program that constructs ρ from the τ_i (see Def. 11.1.10). Constructive membership testing can be done, for instance, for matrix groups over finite fields of odd characteristic in quantum polynomial time [BaBS].

Definition 12.0.3. A subcoset of a group G is a coset of a subgroup. Let $H \leq G$ be groups and $\tau \in G$. We say that the subset $S \subseteq H\tau$ is a set of *coset generators* of $H\tau$ if $H\tau$ is the smallest subcoset of G containing S. (Note that any intersection of subcosets of G is either empty or a subcoset; so every subset of G generates a subcoset of G.)

Observation 12.0.4. Let S be a set of generators of the group G. Then $S \cup \{1\}$ is a set of coset generators of G, i. e., no proper subcoset of G contains $S \cup \{1\}$.

Proposition 12.0.5 (TopAction1)**.** *Let $G \leq \mathfrak{S}(\Omega)$ and $H \leq \mathfrak{S}(\Gamma)$ be permutation groups, $\varphi : G \to H$ a homomorphism, and $N = \ker(\varphi)$. Let S be the given set of generators of H. Given these data and the strings $\mathfrak{x}, \mathfrak{y} : \Omega \to \Sigma$, we can achieve the following by recursively calling $|S| + 1$ instances of String Isomorphism with respect to N, at polynomial cost per instance:*

- *(i) decide whether φ maps $\mathrm{Iso}_G(\mathfrak{x}, \mathfrak{y})$ onto H;*
- *(ii) if the answer is affirmative, find $\mathrm{Iso}_G(\mathfrak{x}, \mathfrak{y})$.*

Proof. Let $S' = S \cup \{1\}$; so S' is a set of coset generators of H. Apply Prop. 12.0.1 to each $\overline{\sigma} \in S'$. If there is a $\overline{\sigma} \in S'$ for which the algorithm returns the empty set ($\overline{\sigma}$ does not lift to an isomorphism), return the answer "no" to item (i). Else, return the answer "yes" to item (i) and observe that $\mathrm{Iso}_G(\mathfrak{x}, \mathfrak{y})$ is the right subcoset of G generated by the subcosets $\varphi^{-1}(\overline{\sigma}) \cap \mathrm{Iso}_G(\mathfrak{x}, \mathfrak{y})$ ($\overline{\sigma} \in S'$) found by Prop. 12.0.1. ☐

Corollary 12.0.6 (TopAction2)**.** *Let $G \le \mathfrak{S}(\Omega)$ be a transitive permutation group and $\varphi : G \to \mathfrak{S}(\Gamma)$ a giant representation, where $|\Gamma| = m > \max\{8, 2+\log_2 n\}$. Given these data and the strings $\mathfrak{x}, \mathfrak{y} : \Omega \to \Sigma$, we can achieve the following by recursively calling $\le 6k$ instances of String Isomorphism with window size $\le n/k$ for some $m \le k \le n$, at polynomial cost per instance:*

(i) decide whether φ maps $\mathrm{Iso}_G(\mathfrak{x}, \mathfrak{y})$ onto a giant coset, i. e., $\mathrm{Iso}_G(\mathfrak{x}, \mathfrak{y})^\varphi \ge \mathfrak{A}(\Gamma)\tau$ for some $\tau \in \mathfrak{S}(\Gamma)$;

(ii) if the answer is affirmative, find $\mathrm{Iso}_G(\mathfrak{x}, \mathfrak{y})$.

Proof. First assume $G^\varphi = \mathfrak{A}(\Gamma)$. Apply Prop. 12.0.5 to $H := \mathfrak{A}(\Gamma)$ with S a pair of generators of H. This reduces our questions to three instances of N-isomorphism where $N = \ker(\varphi)$. Now N is intransitive with k orbits for some $k \le n$. Each orbit has equal length (because $N \triangleleft G$) so Luks's Chain Rule performs the desired reduction, calling $3k$ instances of window size n/k. We need to justify the inequality $k \ge m$. Lemma 10.3.1 (our first lemma toward the Unaffected Stabilizers Lemma, Theorem 10.3.5) says that Ω is affected. Then the Affected Orbit Lemma (Cor. 10.3.7) asserts that each orbit of N has length $\le n/m$.

Now if $G^\varphi = \mathfrak{S}(\Gamma)$ then apply Luks descent, reducing G-isomorphism to two instances of G_1-isomorphism where $G_1 = \varphi^{-1}(\mathfrak{A}(\Gamma))$. □

Remark 12.0.7. If $m \ge \max\{9, 2\log_2 n\}$ then $n/k = \binom{m}{t}$ for some $1 \le t < m/4$ by item (e) of the Main Structure Theorem (Theorem 10.5.1).

Corollary 12.0.8 (TopAction3)**.** *Let $G \le \mathfrak{S}(\Omega)$ be a transitive permutation group and $\varphi : G \to \mathfrak{S}(\Gamma)$ a giant representation, where $|\Gamma| = m > \max\{8, 2+\log_2 n\}$. Given these data and the strings $\mathfrak{x}, \mathfrak{y} : \Omega \to \Sigma$, we can achieve the following by recursively calling $\le 6k$ instances of String Isomorphism with window size $\le n/k$ for some $m \le k \le n$, at polynomial cost per instance:*

(i) *decide whether φ maps $\mathrm{Aut}_G(\mathfrak{x})$ onto a giant, i. e., $\mathrm{Aut}_G(\mathfrak{x})^\varphi \ge \mathfrak{A}(\Gamma)$;*

(i) *if the answer is affirmative, find $\mathrm{Iso}_G(\mathfrak{x}, \mathfrak{y})$.*

Proof. To answer (i), apply Cor. cor:topaction with $\mathfrak{y} = \mathfrak{x}$. Assume the answer is affirmative. If $\mathrm{Iso}_G(\mathfrak{x}, \mathfrak{y})^\varphi$ is a giant coset, we are done by Cor. 12.0.6. We claim that if $\mathrm{Iso}_G(\mathfrak{x}, \mathfrak{y})^\varphi$ is not a giant coset then $\mathfrak{x}$ and $\mathfrak{y}$ are not G-isomorphic. Indeed, if $\mathfrak{x} \cong_G y$ then $\mathrm{Iso}_G(\mathfrak{x}, \mathfrak{y}) = \mathrm{Aut}_G(\mathfrak{x})\sigma$ where σ is any element of $\mathrm{Iso}_G(\mathfrak{x}, \mathfrak{y})$. It follows that $\mathrm{Iso}_G(\mathfrak{x}, \mathfrak{y})^\varphi = \mathrm{Aut}_G(\mathfrak{x})^\varphi \overline{\sigma}$ is a giant coset (where $\overline{\sigma} = \sigma^\varphi$). □

Proposition 12.0.9. *Let $G \le \mathfrak{S}(\Omega)$ be a permutation group and $\varphi : G \to \mathfrak{S}(\Gamma)$ a giant representation. Let $C \subseteq \Gamma$. Then the setwise stabilizer $G_C = \{\sigma \in G \mid C^{\sigma^\varphi} = C\}$ can be found in polynomial time.*

Proof. Let $H = (G^\varphi)_C$. Given that G^φ is a giant, finding H is straightforward. Now $G_C = \varphi^{-1}(H)$. □

Corollary 12.0.10 (TopAction4)**.** *Let $G \le \mathfrak{S}(\Omega)$ be a transitive permutation group and $\varphi : G \to \mathfrak{S}(\Gamma)$ a giant representation where $|\Gamma| = m \ge \max\{16, 4+2\log_2 n\}$. Let $\mathfrak{x}, \mathfrak{y} : \Omega \to \Sigma$ be strings. Assume Γ has a canonical coloring with respect to $\mathfrak{x}$ with a color class C of size $|C| > m/2$ such that the restriction of $\mathrm{Aut}_G(\mathfrak{x})^\varphi$ to C is a giant (includes $\mathfrak{A}(C)$). Then we can find $\mathrm{Iso}_G(\mathfrak{x}, \mathfrak{y})$ by recursively calling $\le 6k$ instances of String Isomorphism with window size $\le n/k$ for some $|C| \le k \le n$, plus a number of instances of total size $\le n$ and maximum size $\le 2n/3$.*

Proof. Since $m \ge \max\{9, 2\log_2 n\}$, by the Main Structure Theorem (Theorem 10.5.1) Ω can be divided into standard blocks on which G acts as a Johnson group. The standard blocks are labeled by $\binom{\Gamma}{t}$ for some $t \ge 1$; and $\Omega(C)$ denotes the union of the standard blocks labeled by the elements of the set $\binom{C}{t}$.

Let $C_\mathfrak{x} = C$. By canonicity, there is a corresponding color class $C_\mathfrak{y} \subseteq \Gamma$ (which may be empty). Apply items 1 to 6 of Procedure Align (Sec. 14.1) with $\mathfrak{X}(\mathfrak{x}) := C_\mathfrak{x}$ and $\mathfrak{X}(\mathfrak{y}) := C_\mathfrak{y}$. The result is that

- if $|C_\mathfrak{x}| \ne |C_\mathfrak{y}|$ then isomorphism is rejected
- else $\mathfrak{y}$ is updated so now $C_\mathfrak{x} = C_\mathfrak{y} = C$
- from the coloring $(C, \Gamma \setminus C)$ of Γ we infer a canonical coloring of Ω; one of the color classes is $\Omega(C)$; and we begin the application of the Chain Rule with this color class.

Now we process $\Omega(C)$ via Cor. 12.0.8. This can be done because $|C| > m/2 \ge \max\{8, 2 + \log_2 n\}$. Then proceed to the remaining color classes in accordance with the Chain Rule.

The bound $2n/3$ on the length of the remaining color classes comes from Lemma 7.2.1. □

Remark 12.0.11. The cost of this procedure can generously be overestimated by $6T(2n/3)$ where $T(n)$ is the maximum cost of instances of size $\le n$.

13 The method of local certificates

13.1 Local Certificates: the core algorithm

In this section we present the group-theoretic “Local certificates” algorithm. **This is the core algorithm of the entire paper.**

The situation we consider is as follows.

The input is a transitive permutation group $G \le \mathfrak{S}(\Omega)$ along with a giant representation $\varphi : G \to \mathfrak{S}(\Gamma)$ (i. e., a homomorphism such that $G^\varphi \ge \mathfrak{A}(\Gamma)$) and two strings $\mathfrak{x}, \mathfrak{y} : \Omega \to \Sigma$ (Σ is a finite alphabet).

Notation: $n = |\Omega|$, $m = |\Gamma|$. We shall assume $m \ge 10\log_2 n$.

Notation 13.1.1. Recall that for a subgroup $L \le G$ and a subset $A \subseteq \Gamma$ we write L_A to denote the setwise stabilizer of A in L with respect to the representation $\varphi_{|L} : L \to \mathfrak{S}(\Gamma)$. We say that A is L-invariant if $L_A = L$. We write $\psi_A : G_A \to \mathfrak{S}(A)$ for the map that restricts the G^φ-action to A. If A is L-invariant then $L^A := L^{\psi_A}$ is the restriction of L^φ to A. In particular, $\psi_\Gamma = \varphi$ and $L^\Gamma = L^\varphi$.

We note that the group $(G^\varphi)_A$ can be trivially computed because φ is a giant represetation. Therefore G_A can be computed in polynomial time as $G_A = \varphi^{-1}((G^\varphi)_A)$.

We note further that if $|A| \le |\Gamma| - 2$ then $\psi_A : G_A \to \mathfrak{S}(A)$ is an epimorphism. Indeed, in this case the setwise stabilizer of A in $\mathfrak{A}(\Gamma)$ acts on A as $\mathfrak{S}(A)$.

We fix a value t and refer to subsets $T \subset \Gamma$ of size $|T| = t$ as *test sets.* For now we only assume $t \le m - 2$ (where $m = |\Gamma|$) but later we further restrict the value of t.

Definition 13.1.2 (Fullness of test set)**.** Let $T \in \binom{\Gamma}{t}$ be a test set. We say that T is *full* with respect to $\mathfrak{x}$ if $\mathrm{Aut}_G(\mathfrak{x})_T^T \ge \mathfrak{A}(T)$, i.e., the G-automorphisms of $\mathfrak{x}$ induce a giant on T. Notation: $\mathcal{F}(\mathfrak{x}) = \{T \in \binom{\Gamma}{t} \mid T \text{ is full}\}$ and $\overline{\mathcal{F}}(\mathfrak{x}) = \binom{\Gamma}{t} \setminus \mathcal{F}(\mathfrak{x})$.

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 $t!\,\mathrm{poly}(n)$) be reduced to the String Isomorphism problem on inputs of size $\le n/t$ where $t = |T|$ is the size of our test set; we shall choose $t = O(\log n)$.

Next we define the types of certificates we seek.

Certificate of non-fullness. A certificate of non-fullness of the test set $T \subset \Gamma$ is a permutation group $M(T) \le \mathfrak{S}(T)$ such that

(i) $M(T) \ngeq \mathfrak{A}(T)$ and

(ii) $M(T) \ge \mathrm{Aut}_G(\mathfrak{x})_T^T$ ($M(T)$ is guaranteed to contain the projection of the G-automorphism group of $\mathfrak{x}$).

Such a group $M(T)$ constitutes a constructive refutation of fullness.

Certificate of fullness. A certificate of fullness of the test set $T \subset \Gamma$ is a permutation group $K(T) \le \mathfrak{S}(\Omega)$ such that

(i) $K(T)^T \ge \mathfrak{A}(T)$ and

(ii) $K(T) \le \mathrm{Aut}_{G_T}(\mathfrak{x})$.

Note that $K(T)$ represents an easily (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" (Thm. 10.3.5).

Theorem 13.1.3 (Local certificates)**.** *Let $T \subseteq \Gamma$ where $|T| = t$. We refer to T as our "test set." Assume* $\max\{8, 2 + \log_2 n\} < t \le m/10$*. By making $\le t!n^2$ calls to String Isomorphism problems on domains of size $\le n/t$ and performing $t!\,\mathrm{poly}(n)$ computation we can decide whether T is full and*

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

(b) if T is not full, find a certificate $M(T) \le \mathfrak{S}(T)$ of non-fullness.

The families $\{(T, K(T)) : T \in \mathcal{F}(\mathfrak{x})\}$ *and* $\{(T, M(T)) : T \in \overline{\mathcal{F}}(\mathfrak{x})\}$ *are canonical.*

Definition 13.1.4 (Affected)**.** Let $G \leq \mathfrak{S}(\Omega)$ be a permutation group and and $\varphi : G \to \mathfrak{S}(\Gamma)$ a homomorphism. Consistently with previous usage, for a subgroup $H \leq G$ we say that $x \in \Omega$ is *affected by* (H, φ) if $H_x^\varphi \not\geq \mathfrak{A}(\Gamma)$. Let $\mathrm{Aff}(H, \varphi)$ denote the set of elements affected by (H, φ), i. e.,

$$\mathrm{Aff}(H, \varphi) = \{x \in \Omega \mid H_x^\varphi \not\geq \mathfrak{A}(\Gamma)\}. \tag{68}$$

Note that if φ restricted to H is not a giant representation then all of Ω is affected by (H, φ).

If $x \in \Omega$ is affected by (H, φ) then all elements of the orbit x^H are affected by (H, φ). In other words, $\mathrm{Aff}(H, \varphi)$ is an H-invariant set. So we can speak of *affected orbits* of H (of which all elements are affected).

We observe the dual monotonicity of the Aff operator.

Observation 13.1.5. If $H_1 \leq H_2 \leq G$ then $\mathrm{Aff}(H_1, \varphi) \supseteq \mathrm{Aff}(H_2, \varphi)$.

The algorithm will consider the input in an increasing sequence of *windows* $W \subseteq \Omega$; in each round, the part of the input outside the window will be ignored. The group $H(W)$ will be the subgroup of G_T that respects the string $\mathfrak{x}^W$, the restriction of $\mathfrak{x}$ to W.

The initial window is the empty set (the input is wholly ignored), so the initial group is G_T. Then in each round we add to W the set of elements of Ω affected by the current group $H(W)$ (we enlarge the window). By the second round $W \neq \emptyset$ because $\mathrm{Aff}(G_T, \psi_T)$ cannot be empty (by the Unaffected Stabilizer Theorem).

As an increasing segment of $\mathfrak{x}$ is taken into account, the group $H(W)$ (the automorphism group of this segment) decreases, and thereby the set of elements affected by $H(W)$ increases. (Previous windows will always be invariant under $H(W)$.)

We stop when one of two things happens: either ψ_T restricted to $H(W)$ is no longer a giant homomorphism, or the window stops growing: no element outside W is affected by $H(W)$.

In the former case we declare that our test set T is *not full* (witnessed by a non-giant group $M(T) := H(W)^T \leq \mathfrak{S}(T)$). Note that the reason $M(T)$ is not a giant is still "local," it only depends on the restriction of $\mathfrak{x}$ to the current window.

In the latter case we declare that T is *full*, and bring as witness the group $K(T) = H(W)_{(\overline{W})}$, the pointwise stabilizer of $\overline{W} = \Omega \setminus W$ in $H(W)$. We claim two things about $K(T)$. First, $K(T)^\varphi \geq \mathfrak{A}(\Gamma)$. This follows from the Unaffected Stabilizers Lemma (Thm. 10.3.5) since none of the elements of $\overline{W}$ is affected. (This is why the window stopped growing.) Second, we observe that $K(T) \leq \mathrm{Aut}_G(\mathfrak{x})$. Indeed, $K(T)$ respects the letters of the string $\mathfrak{x}$ on W (this is an invariant of the algorithm); and it fixes all elements outside W, so the letters of the string restricted to $\overline{W}$ are automatically respected[10].

Here is the algorithm in pseudocode, with a more formal proof.

[10]This observation was the culmination of a long struggle to construct global automorphisms from local information. It amounted to the realization of the decisive role the affected/unaffected dichotomy was to play in the algorithm; indeed this was the moment when the concept of this dichotomy crystallized. It was the "eureka moment" of this long quest. It occurred around noon on September 14, 2015.

Proof of Theorem 13.1.3. For $W \subseteq \Omega$ let $H(W) = \mathrm{Aut}_{G_T}^{W}(\mathfrak{x})$.

All sets denoted T, T', and T_i below will be subsets of Γ of size t (the test sets). An invariant of the **while** loop will be that T is invariant under the action of the group $H(W)$, i. e., $H(W) \le G_T$.

Procedure LocalCertificates

Input: $G \le \mathfrak{S}(\Omega)$, epimorphism $\psi_T : G_T \to \mathfrak{S}(\Gamma)$, test set $T \in \binom{\Gamma}{t}$
Output: decision: "T full/not full," group $K(T)$ (if full) or $M(T)$ (if not full), set $W(T) \subseteq \Omega$
Notation: $H(W) := \mathrm{Aut}_{G_T}^{W}(\mathfrak{x})$ (to be updated as W is updated)

01 $W := \emptyset$ (: so $H(W) = G_T$:)
02 **while** $H(W)^T \ge \mathfrak{A}(T)$ and $\mathrm{Aff}(H(W), \psi_T) \not\subseteq W$
03 $\quad W \leftarrow \mathrm{Aff}(H(W), \psi_T)$ (: enlarging the window :)
04 $\quad$ recompute $H(W)$
05 **end(while)**
06 $W(T) \leftarrow W$
07 **if** $H(W)^T \ge \mathfrak{A}(T)$ (: so $\mathrm{Aff}(H(W), \psi_T) \subseteq W$:)
08 $\quad$ **then** $K(T) \leftarrow H(W)_{(\overline{W})}$ where $\overline{W} = \Omega \setminus W$
09 $\quad$ **return** $W(T)$, $K(T)$, "T full," **exit** (: certificate of fullness found :)
10 **else** $M(T) \leftarrow H(W)^T$
11 $\quad$ **return** $W(T)$, $M(T)$, "T not full," **exit** (: certificate of non-fullness found :)

We need to show how to recompute $H(W)$ on line 4. We write W_{old} for the value of W before the execution of line 03 and W_{new} after.

Procedure Recompute $H(W)$

04a $N \leftarrow H(W_{\text{old}})^T_{(T)}$ (: kernel of $H(W_{\text{old}}) \to \mathfrak{S}(T)$ map :)
04b $L \leftarrow \emptyset$ (: L will collect elements of $H(W_{\text{new}})$:)
04c **for** $\overline{\sigma} \in H(W_{\text{old}})^T$ (: $H(W_{\text{old}})^T = \mathfrak{A}(T)$ or $\mathfrak{S}(T)$:)
04d $\quad$ select $\sigma \in H(W_{\text{old}})$ such that $\sigma^T = \overline{\sigma}$ (: lifting $\overline{\sigma}$ to Ω :)
04e $\quad L(\overline{\sigma}) \leftarrow \mathrm{Aut}_{N\sigma}^{W_{\text{new}}}(\mathfrak{x})$ (: performing strong descent to N :)
04f $\quad L \leftarrow L \cup L(\overline{\sigma})$
04g **end(for)**
04h **return** $H(W_{\text{new}}) \leftarrow L$

Justification. First we observe that on each iteration of the **while** loop on lines 02–05, $H(W_{\text{new}}) \le H(W_{\text{old}})$ and $W_{\text{new}} \supseteq W_{\text{old}}$. In fact, these inclusions are proper or else we exit on line 02. In particular, T is invariant under $H(W)$ throughout the process because it is invariant in line 01. It also follows that on line 07 we actually have $\mathrm{Aff}(H(W), \psi_T) = W$. We also note that the **while** loop will be executed at least once (by the comment on line 01).

Claim 13.1.6. *On line 08,* $K(T)^T \ge \mathfrak{A}(T)$ *and* $K(T) \le \mathrm{Aut}_G(\mathfrak{x})$. *In particular, T is full.*

Proof. $K(T) \geq \mathfrak{A}(T)$ is the crucial consequence of Theorem 10.3.5, applied to the giant representation $\overline{\psi}_T : H(W_{\text{old}}) \to \mathfrak{S}(T)$. ($\overline{\psi}_T$ denotes the restriction of ψ_T to $H(W_{\text{old}})$.)

To show that $K(T) \leq \text{Aut}_G(\mathfrak{x})$ let $\sigma \in K(T)$ and $u \in \Omega$. We need to show that $\mathfrak{x}(u^\sigma) = \mathfrak{x}(u)$. If $u \in W$ then this follows because $\sigma \in H(W) = \text{Aut}_G^W(\mathfrak{x})$. If $u \in \overline{W}$ then $u^\sigma = u$. $\square$

Claim 13.1.7. *If T is not full then we reach line 10 with $M(T) \not\geq \mathfrak{A}(T)$ and $\text{Aut}_G(\mathfrak{x})_T^T \leq M(T)$.*

Proof. We reach line 10 by Claim 13.1.6. We then have $\text{Aut}_G(\mathfrak{x})_T^T \leq M(T)$ because the relation $\text{Aut}_G(\mathfrak{x})_T^T \leq H(W)$ is an invariant of the process. $\square$

Next we justify procedure Recompute $H(W)$. This is immediate from the observation

$$H(W_{\text{old}}) = \bigcup_{\overline{\sigma}} N\overline{\sigma} \tag{69}$$

where the union extends over $\overline{\sigma} \in H(W_{\text{old}})$. So we can use strong descent (over the orbits of N in W_{new}) to compute $\text{Aut}_{H(W_{\text{old}})}^{W_{\text{new}}}(\mathfrak{x})$. But this group is $H(W_{\text{new}})$ because $W_{\text{new}} \supseteq W_{\text{old}}$.

Finally we need to justify the complexity assertion. This is where Cor. 10.3.7 ("Affected Orbit Lemma") plays a critical role.

The **while** loop is executed at most n times (because W strictly increases in each round; we exit on line 02 when the window stops growing), so the dominant component of the complexity is in recomputing $H(W)$. We have reduced this to $\leq t!$ instances of string N-isomorphism on the window W_{new}.

By Cor. 10.3.7 ("Affected Orbit Lemma"), each orbit of N in W_{new} has length $\leq n/t$.

We conclude that strong Luks reduction reduces the recomputation of $H(W)$ to $\leq n \cdot t!$ instances of String Isomorphism on windows of size $\leq n/t$, justifying the stated complexity estimate. $\square$

Our procedure does more than stated in Theorem 13.1.3. It also returns the set $W(T)$. We summarize key properties of this assignment.

Proposition 13.1.8. *As in Theorem 13.1.3, let a* test set *be a subset $T \subseteq \Gamma$ with $|T| = t$ elements where $\max\{8, 2 + \log_2 n\} < t \leq m/10$. For all test sets T we have*

(i) $\Omega(T) \subseteq W(T) \subseteq \Omega$

(ii) $W(T)$ *is invariant under* $\text{Aut}_{G_T}(\mathfrak{x})$

(iii) *if T is full then* $W(T) = \text{Aff}(\text{Aut}_{G_T}^{W(T)}(\mathfrak{x}))$

(iv) *if T is full then $K(T)^T$ fixes all elements of $\Omega \setminus W(T)$*

(v) *the assignment $T \mapsto W(T)$ is canonical.*

Proof. Evident from the algorithm. $\square$

We need to highlight one more fact about the structures we obtained.

Notation 13.1.9 (Truncation of strings)**.** Let $*$ be a special symbol not in the alphabet Σ. For the string $\mathfrak{x} : \Omega \to \Sigma$ and "window" $W \subseteq \Omega$ we define the string $\mathfrak{x}^W : \Omega \to (\Sigma \cup \{*\})$ by setting $\mathfrak{x}^W(u) = \mathfrak{x}(u)$ for $u \in W$ and $\mathfrak{x}^W(u) = *$ for $u \in \Omega \setminus W$.

Notation 13.1.10 (Coloring of strings)**.** For the string $\mathfrak{x} : \Omega \to \Sigma$ and the test set $T \subseteq \Gamma$ we define the string $\mathfrak{x}_T : \Omega \to (\Sigma \times \{0,1\})$ by setting $\mathfrak{x}_T(u) = (\mathfrak{x}(u), 1)$ if $u \in \Omega(T)$ and $\mathfrak{x}_T(u) = (\mathfrak{x}(u), 0)$ if $u \notin \Omega(T)$.

Proposition 13.1.11 (Comparing local certificates)**.** *For all test sets $T, T' \subseteq \Gamma$ with $|T| = |T'| = t$ and all strings $\mathfrak{x}, \mathfrak{x}' : \Omega \to \Sigma$ we can compute $\mathrm{Iso}_G\left(\mathfrak{x}_T^{W(T)}, (\mathfrak{x}')_{T'}^{W(T')}\right)$ by making $\le t!n^2$ calls to String Isomorphism problems on domains of size $\le n/t$ and performing $t!\,\mathrm{poly}(n)$ computation.*

Proof. Run procedure LocalCertificates simultaneously on $(\mathfrak{x}, T)$ and on $(\mathfrak{x}', T')$, maintaining the variable W for (x, T) and the variable W' for $(\mathfrak{x}', T')$. Further maintain the set $Q = \mathrm{Iso}_G(\mathfrak{x}_T^W, (\mathfrak{x}')_{T'}^{W'})$. On line 01 we shall have $Q = G_T\sigma$ for any $\sigma \in G$ that takes T to T'.

Change line 04 to "recompute $H(W)$ and Q."
Here is the modified "Recompute" code.

Procedure Recompute $H(W)$ and Q

04a	$N \leftarrow H(W_{\mathrm{old}})_{(T)}^T$	(: kernel of $H(W_{\mathrm{old}}) \to \mathfrak{S}(T)$ map :)
04b1	$L \leftarrow \emptyset$	(: L will collect elements of $H(W_{\mathrm{new}})$:)
04b2	$R \leftarrow \emptyset$	(: R will collect elements of Q_{new} :)
04c0	fix $\pi_0 \in Q_{\mathrm{old}}$	
04c1	**for** $\overline{\sigma} \in H(W_{\mathrm{old}})^T$	(: $H(W_{\mathrm{old}})^T = \mathfrak{A}(T)$ or $\mathfrak{S}(T)$:)
04d1	select $\sigma \in H(W_{\mathrm{old}})$ such that $\sigma^T = \overline{\sigma}$	(: lifting $\overline{\sigma}$ to Ω :)
04d2	$\pi \leftarrow \sigma\pi_0$	(: $\pi \in Q_{\mathrm{old}}$:)
04e1	$L(\overline{\sigma}) \leftarrow \mathrm{Aut}_{N\sigma}^{W_{\mathrm{new}}}(\mathfrak{x})$	
04e2	$R(\overline{\sigma}) \leftarrow \mathrm{Iso}_{N\pi}(\mathfrak{x}_T^{W_{\mathrm{new}}}, (\mathfrak{x}')_{T'}^{W'_{\mathrm{new}}})$	(: performing strong descent to N :)
04f1	$L \leftarrow L \cup L(\overline{\sigma})$	(: collecting automorphisms :)
04f2	$R \leftarrow R \cup R(\overline{\pi})$	(: collecting isomorphisms :)
04g	**end(for)**	
04x	**if** $R = \emptyset$ **then** reject isomorphism, **exit**	
04h	**else return** $H(W_{\mathrm{new}}) \leftarrow L$ and $Q \leftarrow R$	

The analysis is analogous with the analysis of the Recompute $H(W)$ routine. □

13.2 Aggregating the local certificates

We continue the notation of the previous section.

Theorem 13.2.1 (AggregateCertificates)**.** *Let $\varphi : G \to \mathfrak{S}(\Gamma)$ be a giant representation, where $G \le \mathfrak{S}(\Omega)$, $|\Omega| = n$, and $|\Gamma| = m$. Let $\max\{8, 2 + \log_2 n\} < t < m/10$. Then, at a multiplicative cost of $m^{O(t)}$, we can either find a canonical colored $4/5$-partition of Γ or find* *a canonically embedded t-ary relational structure with relative symmetry defect $\geq 1/2$ on Γ, or reduce the determination of $\mathrm{Iso}_G(\mathfrak{x},\mathfrak{y})$ to $n^{O(1)}$ instances of size $\leq 2n/3$.*

Proof. We describe the procedure, interspersed with the justification.

Run the LocalCertificates routine for both inputs $\mathfrak{x},\mathfrak{y}$ and all test sets $T \in \binom{\Gamma}{t}$.

Run the CompareLocalCertificates routine for all pairs $((\mathfrak{x},T),(\mathfrak{x}',T'))$ where $\mathfrak{x}$ is fixed, $\mathfrak{x}' \in \{\mathfrak{x},\mathfrak{y}\}$, and $T,T' \in \binom{\Gamma}{t}$ are test sets (a total of $2\binom{m}{t}^2$ runs).

Let $F(\mathfrak{x})$ be the subgroup generated by the groups $K(T)$ for all full subsets $T \in \binom{\Gamma}{t}$ with reference to input string $\mathfrak{x}$. So $F(\mathfrak{x})$, and with it $F(\mathfrak{x})^\Gamma$, are canonically associated with $\mathfrak{x}$. In particular, if $F(\mathfrak{y})$ is analogously defined for $\mathfrak{y}$, then $F(\mathfrak{x})^\Gamma$ is permutationally isomorphic to $F(\mathfrak{y})^\Gamma$, i. e., there exists a permutation $\alpha \in \mathfrak{S}(\Gamma)$ such that $F(\mathfrak{y})^\Gamma = \alpha^{-1}F(\mathfrak{x})^\Gamma\alpha$.

Below we ignore $\mathfrak{y}$ and focus on $\mathfrak{x}$, omitting it from the notation, so we write $F = F(\mathfrak{x})$. But our guide is the above consequence of canonicity.

(1) **if** the nontrivial orbits (orbits of length ≥ 2) of F^Γ cover at least $m/5$ elements of Γ and no orbit of F^Γ has length $> 4m/5$ we found a colored 4/5-partition of Γ, **exit**

(2) **else if** F^Γ has an orbit $C \subseteq \Gamma$ of length $|C| > 4m/5$ (: since $|C| > m/2$, this orbit is canonical. :)

2a **if** $F^C \geq \mathfrak{A}(C)$ then apply Cor. 12.0.10

2b **else** let d be the degree of transitivity[11] of F^C (see Def. 2.2.3)
(: so $1 \leq d \leq 5$ by Theorem 2.2.4 :)
individualize the elements of a set $S \in \binom{C}{d-1}$
(: so $F^C_{(S)}$ is transitive but not doubly transitive on $C' := C \setminus S$:)
Let $\mathfrak{X} = (C'; R_1, \ldots, R_r)$ be the orbital configuration of $F^C_{(S)}$ on C' (the R_i are the orbits of $F^C_{(S)}$ on C'). This is a non-clique homogeneous coherent configuration, so $3 \leq r \leq m$. (: Warning: the numbering of the R_i is not canonical; isomorphisms may permute the R_i :)
Let $R_1 = \mathrm{diag}(C')$ be the diagonal
(: so for $i \geq 2$ the constituents $X_i = (C', R_i)$ are nontrivial biregular digraphs :)
Individualize one of the X_i ($i \geq 2$) (: multiplicative cost $r - 1 \leq m - 1$:)
return X_i, **exit**
(: Note: X_i has relative symmetry defect $\geq 1/2$ by Cor. 2.4.12 because X_i is an irreflexive, biregular, nontrivial digraph. :)

(3) **else** $|D| \geq 4m/5$ where $D \subseteq \Gamma$ is the set of fixed points of F^Γ. So in the remaining case we have Note that in this case, if $T \subset D$ then T is not full. (In fact even if $T \cap D \neq \emptyset$ then T is not full.)
Claim (Turning local asymmetry into global irregularity)
In time $m^{O(t)}$ we can construct a canonical t-ary relational structure on D with symmetry defect (much) greater than 1/2.

[11] The case when F^C is doubly transitive but not a giant was handled by a different method, using an 1897 gem of asymptotic group theory by Bochert [Bo97], see [DiM, Thm. 5.4A].

Proof. We apply Prop. **??** (Local guides). To do so, we need to define the relevant categories. Let $\mathfrak{x}_1$ and $\mathfrak{x}_2$ (rather than $\mathfrak{x}$ and $\mathfrak{y}$) denote our two input strings. Let D_i be the subset D derived from input $\mathfrak{x}_i$. We apply Prop. **??** with the assignment $\Omega_i \leftarrow D_i$ of variables.

The objects of the category $\mathcal{L}$ correspond to the pairs (T, i) where $T \in \binom{D_i}{t}$ is a test set. The set of morphisms $(T, i) \to (T', j)$ are the bijections $T \to T'$ corresponding to the set $\mathrm{Iso}_G\left((\mathfrak{x}_i)_T^{W_i(T)}, (\mathfrak{x}_j)_{T'}^{W_j(T')}\right)$ for all $T, T' \in \binom{\Gamma}{t}$, where W_i corresponds to W under input $\mathfrak{x}_i$.

The two abstract objects of category $\mathcal{C}$ are denoted $\mathfrak{X}_1$ and $\mathfrak{X}_2$. The underlying set of $\mathfrak{X}_i$ is $\square(\mathfrak{X}_i) = D_i$. The morphisms are the bijections $D_1 \to D_2$ induced by the G-isomorphisms $\mathfrak{x}_1 \to \mathfrak{x}_2$.

Our current assumption is that the objects in $\mathcal{L}$ are not full in our sense, i. e., $\mathrm{Aut}(T, i) \leq M_i(T)$ where $M_i(T) \not\geq \mathfrak{A}(T)$. In particular it follows that the objects in $\mathcal{L}$ are not full in the sense of Prop. **??**, i. e., $\mathrm{Aut}(T, i) \neq \mathfrak{S}(T)$.

Thus the assumptions of Prop. **??** are satisfied. The algorithm of Prop. **??** returns canonical t-ary relational structures on D_i with strong symmetry defect $\geq |D_i| - t + 1 > m/2$. $\square$

Now **return** this canonical t-ary relational structure, **exit**

This completes the procedure and the proof. $\square$

14 Effect of discovery of canonical structures

Situation: We have a transitive group $G \leq \mathfrak{S}(\Omega)$ of degree $n = |\Omega|$ and a giant representation $\varphi : G \to \mathfrak{S}(\Gamma)$ (i. e., $G^{\varphi} \geq \mathfrak{A}(\Gamma)$). Assume $m := |\Gamma| \geq 10 \log_2 n$. Let Φ be the set of standard blocks for φ (see the Main Structure Theorem, Thm. 10.5.1, so $\Phi = \{B_T : T \in \binom{\Gamma}{t}\}$. The B_T partition Ω and form a system of imprimitivity for G.

In this section we study the effect of canonical structures embedded in Γ.

Both our group-theoretic partitioning algorithm (AggregateCertificates, Theorem 13.2.1) and our combinatorial partitioning algorithm (the Extended Design Lemma, Theorem 9.2.3) produce a canonical coloring of Γ with an additional canonical structure on some of the color classes. The additional structure can be an equipartition or a Johnson scheme. (We note that canonicity in each case is relative to arbitrary choices previously made and correspondigly came at a multiplicative cost.)

14.1 Alignment of input strings, reduction of group

A common feature of the categories of these types of structures is that their G^{φ}-isomorphisms are easy to find (where G^{φ} is either $\mathfrak{S}(\Gamma)$ or $\mathfrak{A}(\Gamma)$). (This is trivial in linear time for colored equipartitions, and polynomial time for Johnson schemes.)

We use these structures to align the input strings $\mathfrak{x}$ and $\mathfrak{y}$ and reduce the group G.

Let $\mathfrak{X}(\mathfrak{z})$ be the canonical structure associated with the input string $\mathfrak{z} \in \{\mathfrak{x}, \mathfrak{y}\}$. Alignment means that $\mathfrak{X}(\mathfrak{x}) = \mathfrak{X}(\mathfrak{y}')$ for a G-shifted copy $\mathfrak{y}'$ of $\mathfrak{y}$.

Procedure Align

Input: canonical structures $\mathfrak{X}(\mathfrak{x})$, $\mathfrak{X}(\mathfrak{y})$ on Γ
Output: string $\mathfrak{y}'$, permutation $\sigma \in G$, and group $G_1 \le G$ such that

$$\mathrm{Iso}_G(\mathfrak{x}, \mathfrak{y}) = \mathrm{Iso}_{G_1}(\mathfrak{x}, \mathfrak{y}')\sigma \quad \text{and} \quad G_1^{\varphi} = \mathrm{Aut}(\mathfrak{X}(\mathfrak{x})) \tag{70}$$

(: Note that it follows that $\mathfrak{X}(\mathfrak{x}) = \mathfrak{X}(\mathfrak{y}')$:)

Additional output if $\mathfrak{X}$ has a dominant color class $\Delta \subseteq \Gamma$ ($|\Delta| > m/2$) and $\mathfrak{X}$ involves an equipartition of Δ or a Johnson scheme on Δ: reduced set Γ' and giant representation $G \to \mathfrak{S}(\Gamma')$ for recursive processing of the corresponding window $\Omega(\Delta)$.

1. If $\mathfrak{X}(\mathfrak{x})$ and $\mathfrak{X}(\mathfrak{y})$ are not G^{φ}-isomorphic then reject isomorphism, exit
2. Else, let
   (i) $\overline{\sigma} \in \mathrm{Iso}_{G^{\varphi}}(\mathfrak{X}(\mathfrak{x}), \mathfrak{X}(\mathfrak{y}))$ (: aligning in Γ :)
   (ii) $\sigma \in \varphi^{-1}(\overline{\sigma})$ (: lifting :)
   (iii) $\mathfrak{y}' = \mathfrak{y}^{\sigma^{-1}}$ (: aligning the inputs :)
   (iv) $G_1 = \varphi^{-1}(\mathrm{Aut}(\mathfrak{X}(\mathfrak{x})))$ (: reducing the group :)

   (: Alignment as stated in Eq. (70) achieved :)
3. Update: $\mathfrak{y} \leftarrow \mathfrak{y}'$, $G \leftarrow G_1$.
4. (: Each of our structures has an underlying coloring – possibly trivial :)

   Let $(\Delta_1, \dots, \Delta_k)$ be the coloring of $\mathfrak{X}(\mathfrak{x})$ (the Δ_j are the color classes); so Γ is the disjoin union of the Δ_j.

   This coloring induces a canonical coloring of $\Phi = \binom{\Gamma}{t}$ as described in Lemma 7.2.1; let $\Phi_1, \dots, \Phi_s$ be the color classes. This coloring in turn lifts to a canonical coloring of Ω with corresponding color classes $\Omega_1, \dots, \Omega_s$ where $\Omega_i = \bigcup_{T \in \Phi_i} B_T$. For $A \subseteq \Gamma$ recall the notation $\Phi(A) = \binom{A}{t}$ and $\Omega(A) = \bigcup_{T \in \Phi(A)} B_T$.
5. Apply the Chain Rule to the color classes Ω_i.
6. If $(\exists j)(|\Delta_j| > m/2)$ ("dominant color") then start the application of the Chain Rule with the window $\Omega(\Delta_j) = \bigcup_{T \in \binom{\Delta_j}{t}} B_T$.
7. While processing window $\Omega(\Delta_j)$

   (A) if $\mathfrak{X}$ gives a nontrivial equipartition of Δ_j then let $\mathfrak{Y}$ be this equivalence relation on Δ_j and Γ^* the set of blocks

(B) if Δ_j is the vertex set of a Johnson scheme $\mathfrak{J}(m^*, t^*)$ ($t^* \geq 2$) then identify Δ_j with $\Delta_j = \binom{\Gamma^*}{t^*}$ where $|\Gamma^*| = m^*$ and let $\mathfrak{Y}$ denote this Johnson scheme on Δ_j

8. let $H = \mathrm{Aut}(\mathfrak{Y})$ and $\psi : H \to \mathfrak{S}(\Gamma^*)$ be the natural epimorphism

9. let $G^* = \varphi^{-1}(H)$

10. let $\varphi^* : G^* \to \mathfrak{S}(\Gamma^*)$ be the composition of φ restricted to G^* and ψ
(: this is a giant representation :)

11. update: $G \leftarrow G^*$, $\Gamma \leftarrow \Gamma^*$, $\varphi \leftarrow \varphi^*$

end(procedure)

14.2 Cost analysis

We are assuming that isomorphism of our canonical structures $\mathfrak{X}$ is testable in polynomial time (which is certainly true for the types of structures considered), so Line 2 is executed in polynomial (in m) time.

We need to examine the efficiency of the application of the Chain rule in Lines (5), (6).

We measure complexity in terms of the number of group operations. We assume G and a giant representation $\varphi : G \to \mathfrak{S}(\Gamma)$ are given where $G \leq \mathfrak{S}(\Omega)$ with $|\Omega| = n$ and $|\Gamma| = m$. Let $T(G, \varphi)$ be the maximum cost over all input strings for the pair (G, φ).

We use the notation of Section 11.4. So $T_{\mathrm{Jh}}(x, y)$ is the maximum of $T(G, \varphi)$ over all G and φ with the parameters $n \leq x$ and $m \leq y$. Moreover, $T_{\mathrm{Jh}}(x)$ is defined as $T_{\mathrm{Jh}}(x) = T_{\mathrm{Jh}}(x, x)$. $T(x)$ is the upper bound for all groups G of degree $n \leq x$. (Note that n is the "window size.")

We are looking a function $T(x)$ that is "nice" in the sense that $\log\log T(x)/\log\log x$ is monotone nondecreasing for sufficiently large x. (For the function $\exp((\log x)^c)$, this quantity is constant.)

In analyzing the complexity, we need to take into account the potentially quasipolynomial (in terms of m), say $q(m)$, multiplicative cost of reaching our canonical structures $\mathfrak{X}$: we need to compare not one but $q(m)$ instances of $\mathfrak{X}(\mathfrak{y})$ with $\mathfrak{X}(\mathfrak{x})$). So the overall cost, including the application of the Chain rule, will be

$$T(G, \varphi) \leq q(m) \sum_i T(|\Omega_i|) \tag{71}$$

If $(\forall i)(|\Omega_i| \leq 2n/3)$ then this yields (generously) the inequality

$$T(G, \varphi) \leq m \cdot q(m) T(2n/3), \tag{72}$$

justifying Inequality (65). (In fact, for "nice" functions as postulated, we obtain $T(G, \varphi) \leq q(m)(T(n/3) + T(2n/3))$. But this gain of a factor of m will make no difference.)

If $(\exists i)(|\Omega_i| > 2n/3)$ then by Lemma 7.2.1, for this $i = i_0$ we must have $\Omega_{i_0} = \Omega(\Delta_j)$ where $|\Delta_j| > 2m/3$. The total contribution of all other Ω_i to the right-hand side of Eq. (71) is at most $q(m)T(n/3)$.

Our progress on $\Omega(\Delta_j)$ is measured in terms of the reduced Γ. In the case of an equipartition, Γ' is the set of blocks of the partition, so $|\Gamma'| \le m/2$. In case of a Johnson scheme $\mathfrak{J}(m', t')$ $(t' \ge 2)$ with vertex set $\Delta_j = \binom{\Gamma'}{t'}$, we have $m \ge |\Delta_j| = \binom{m'}{t'} \ge \binom{m'}{2} > (m'-1)^2/2$, so $m' < 1 + \sqrt{2m} < m/2$ (for $m \ge 12$). So in each case we obtain the inequality

$$T(G, \varphi) \le q(m)(T(n/3) + T_{\mathrm{Jh}}(n, m/2)) \tag{73}$$

justifying Eq. (v) in Sec. 11.4 and yielding the conclusion

$$T(n) \le q(n)^{O(\log^2 n)} \tag{74}$$

as in Eq. (67).

15 The Master Algorithm

The algorithm will refer to a polylogarithmic function $\ell(x)$ to be specified later.

Whenever a subroutine in the algorithm exits and returns a good color-partition of Ω, the algorithm starts over (recursively). If it returns a structure such as a UPCC, we move to the next line. If the subroutine returns isomorphism rejection, that branch of the recursion terminates and the algorithm backtracks.

Procedure String-Isomorphism

Input: group $G \le \mathfrak{S}(\Omega)$, strings $\mathfrak{x}, \mathfrak{y} : \Omega \to \Sigma$

Output: $\mathrm{Iso}_G(\mathfrak{x}, \mathfrak{y})$

1. Apply Procedure Reduce-to-Johnson (Luks reductions, Sec. 11.3)
   (: The rest of this algorithm constitutes the ProcessJohnsonAction routine announced in Sec. 11.3)

2. (: G is transitive, G-action $\mathfrak{G}$ on blocks is Johnson group isomorphic to $\mathfrak{S}_m$ or $\mathfrak{A}_m$:)
   set $\ell = (\log n)^3$
   if $m \le \ell$ **then** apply imprimitive Luks reduction to reduce to kernel of the G-action on the blocks (brute force on small primitive group $\mathfrak{G}$, multiplicative cost $\ell!$:)

3. (: G-action on blocks is isomorphic to $\mathfrak{S}(\Gamma)$ or $\mathfrak{A}(\Gamma)$, $|\Gamma| = m > \ell$:)
   Let $\varphi : G \to \mathfrak{S}(\Gamma)$ be a giant representation (inferred from $\mathfrak{G}$)
   Let $N = \ker(\varphi)$ and let $\Phi = \{B_T \mid T \in \binom{\Gamma}{t}\}$ be the set of standard blocks (Thm. 10.5.1)
   (: the B_T partition Ω and G acts on Φ as $\mathfrak{S}^{(t)}(\Gamma)$ or $\mathfrak{A}^{(t)}(\Gamma)$:)

4. **if** G primitive (: i. e., $\Omega = \Phi$:)

4a. **if** $t = 1$ **then** find $\mathrm{Iso}_G(\mathfrak{x}, \mathfrak{y})$, **exit** (: trivial case: $\Omega = \Gamma$, $G \ge \mathfrak{A}(\Omega)$;
isomorphism only depends on the multiplicity of each letter in the strings $\mathfrak{x}, \mathfrak{y}$:)

4b. **else** (: $t \geq 2$:) view $\mathfrak{x}, \mathfrak{y}$ as edge-colored t-uniform hypergraphs $\mathcal{H}(\mathfrak{x})$ and $\mathcal{H}(\mathfrak{y})$ on vertex set Γ
if relative symmetry defect of $\mathcal{H}(\mathfrak{x})$ is $< 1/2$ **then** apply Cor. 12.0.10

4c. **else** (: now their relative symmetry defect is $\geq 1/2$:)
(: view these hypergraphs as t-ary relational structures :)
apply Extended Design Lemma (Theorem 9.2.3)

4d. (: canonical structure $\mathfrak{X}$ on Γ found: colored equipartition or Johnson scheme :)
apply Procedure Align to $\mathfrak{X}$ (Sec. 14.1)

5. **else** (: G imprimitive, i. e., $|\Phi| \leq (1/2)|\Omega|$:)
apply AggregateCertificates (Theorem 13.2.1)
(: Note: this is where our main group-theoretic algorithm,
Procedure LocalCertificates (Theorem 13.1.3), is used :)

6. **if** AggregateCertificates returns canonically embedded k-ary relational structure on Γ with relative symmetry defect $\geq 1/2$ **then**

6a. apply Extended Design Lemma (Theorem 9.2.3)
$\mathfrak{X} \leftarrow$ canonical structure on Γ returned
(: $\mathfrak{X}$ is a colored equipartition of Γ or a Johnson scheme embedded in Γ :)

7. **else** (: AggregateCertificates returns canonical colored equipartition on Γ :)
$\mathfrak{X} \leftarrow$ colored equipartition returned

7a. apply Procedure Align to $\mathfrak{X}$ (Sec. 14.1)

The essence of the analysis is in the analysis of Procedure Align given in Section 14.1.

16 Concluding remarks

16.1 Dependence on the Classification of Finite Simple Groups

As mentioned in the Introduction, the analysis of the algorithm, as stated, depends on the Classification of Finite Simple Groups (CFSG) via Cameron's classification of large primitive permutation groups. There is one other instance in which we rely on CFSG; we employ "Schreier's Hypothesis" in the proof of Lemma 10.2.5.

We are, however, able to considerably reduce the dependence of the analysis on CFSG; we are able to do *without Cameron's result* by one more application of the Procedure UPCC Split-or-Johnson (Theorem 9.2.1) and some 80-year-old group theory.

Cameron's result guaranteed that if G acted as a large primitive group $\mathfrak{G} \leq \mathfrak{S}(\Phi)$ on the set Φ of blocks of a minimal system of imprimitivity (the blocks are maximal), then $\mathfrak{G}$ was a Cameron group, which in turn either had a transitive, imprimitive subgroup of small index (so Luks reduction was applicable) or a Johnson group. This reduction was done in Procedure Reduce-to-Johnson (Sec. 11.3).

We are able to replace this procedure by one that does not rely on Cameron's result; we locate this Johnson group combinatorially. Here is an outline.

Let $k = |\Phi|$ be the number of blocks.

If $\mathfrak{G}$ is uniprimitive (primitive but not doubly transitive) then let $\mathfrak{X}$ be the orbital configuration of $\mathfrak{G}$, defined as the coherent configuration on Φ where the color classes are the orbitals of $\mathfrak{G}$, i. e., the orbits of $\mathfrak{G}$ on $\Phi \times \Phi$. Now $\mathfrak{X}$ is uniprimitive because $\mathfrak{G}$ is uniprimitive, and the color classes are by definition G-invariant. Apply Procedure UPCC Split-or-Johnson (Theorem 9.2.1) to $\mathfrak{X}$. The procedure either returns a canonical colored 3/4-partition of Φ, representing significant progress, or returns a canonically embedded Johnson scheme $\mathfrak{J}(m,t)$ on a subset J of Φ of size $|J| = \binom{m}{t} \geq 3k/4$. After breaking up Φ via the Chain rule, we shall be left with J (Lemma 7.2.1). The G-action on $\mathfrak{J}(m,t)$ is a subgroup of $\mathfrak{S}_m^{(t)}$ and can be represented on the set $[m]$ which is much smaller than Φ ($k \geq \binom{m}{2}$, so $m < 1 + \sqrt{2k}$). If this is a giant action (the image contains $\mathfrak{A}_m$), we are in the same situation as if we had used Cameron's theorem. If the action is not giant, we recurse (find orbits and minimal block system for the action on $[m]$, etc.).

This completes the case when $\mathfrak{G}$ is uniprimitive.

If $\mathfrak{G}$ is not uniprimitive then $\mathfrak{G}$ is doubly transitive. Giants are the $t = 1$ case of Johnson groups, so if $\mathfrak{G}$ is a giant, we are done. So we may now assume that $\mathfrak{G}$ is doubly transitive but not a giant. We could conclude now by applying strong Luks reduction to the kernel of the $G \to \mathfrak{G}$ epimorphism (brute force on $\mathfrak{G}$), with reference to an elementary result by Pyber [Py93] that the order of $\mathfrak{G}$ is quasipolynomially bounded (as a function of k). But we can make our algorithm even more efficient and the analysis even more elementary by limiting the group theory used to an old reference.

Let d be the degree of transitivity of $\mathfrak{G}$ (see Def. 2.2.3). By Wielandt's 1934 result (Thm. 2.2.5) we have $d < 3 \ln n$.

Pick $S \subset \Phi$ with $|S| = d - 1$. Individualize the elements of S. Now the group $\mathfrak{G}_{(S)}$ (pointwise stabilizer of S) is transitive but not doubly transitive in its action on $\Phi \setminus S$. If $\mathfrak{G}_{(S)}$ is imprimitive on $\Phi \setminus S$ then we reduce Φ to the set Φ' corresponding to the blocks of imprimitivity of $\mathfrak{G}_{(S)}$; so we now have $k' := |\Phi'| \leq k/2$ blocks, significant progress. Otherwise, $\mathfrak{G}_{(S)}$ is uniprimitive on $\Phi \setminus S$, so we are back to the case already discussed.

Remark 16.1.1. Stronger bounds hold on the degree of transitivity. Under CFSG, we have $t \leq 5$, and in fact $t \leq 3$ if $k \geq 25$. Moreover, Wielandt [Wi2] (see [DiM, Thm. 7.3A]) has shown that assuming only Schreier's Hypothesis, one can prove $t \leq 7$. So 6 individualizations (rather than $3 \ln k$ individualizations) suffice in the above argument if we are willing to assume Schreier's Hypothesis. (Note also that if we do encounter a 7-transitive group that is not a giant, we shall have found an explicit counterexample to Schreier's Hypothesis and thereby to CFSG, an impressive by-product.)

Remark 16.1.2. A failure of Schreier's Hypothesis would not cause a hidden error in the algorithm: the algorithm would produce and explicit counterexample to the Unaffected Stabilizers Lemma (Thm. 10.3.5) and thereby to Schreier's Hypothesis, and would therefore exhibit a hitherto unknown finite simple group. This would be a rather remarkable by-product.

16.2 How easy is Graph Isomorphism?

The first theoretical evidence against the possibility of NP-completeness of GI was the equivalence of existence and counting [Ba77, Mat], not observed in any NP-complete problem. The second, stronger evidence came from the early theory of interactive proofs: graph isomorphism is in coAM, and therefore if GI is NP-complete then the polynomial-time hierarchy collapses to the second level (Goldreich–Micali–Wigderson 1987 [GoMW]). Our result provides a third piece of evidence: GI is not NP-complete unless all of NP can be solved in quasipolynomial time.

A number of questions remain. The first one is of course whether GI is in P. Such expectations should be tempered by the status of the *Group Isomorphism* problem[12]: given two groups by their Cayley tables, are they isomorphic? It is easy to reduce this problem to GI. In fact, Group Isomorphism seems much easier than GI; it can trivially be solved in time $n^{O(\log n)}$ where n is the order of the group. But in spite of considerable effort and the availability of powerful algebraic machinery, Group Isomorphism is still not known to be in P. We are not even able to decide Group Isomorphism[13] in time $n^{o(\log n)}$.

A closely related challenge that deserves attention is the String Isomorphism problem on $n = p^k$ points, with respect to the linear group $\mathrm{GL}(k, p)$. The order of this group is about $p^{k^2} = n^{\log_p n}$; the question is, can this problem be solved in time $p^{o(k^2)}$ (or perhaps even in $\mathrm{poly}(n)$ time). I note that this problem can be encoded as a GI problem for graphs with $\mathrm{poly}(n)$ vertices so if $\mathrm{GI} \in \mathrm{P}$ then this problem is in P as well.

The result of the present paper amplifies the significance of the Group Isomorphism problem (and the challenge problem stated) as a barrier to placing GI in P. It is quite possible that the intermediate status of GI (neither NP-complete, nor polynomial time) will persist.

In fact, even putting GI in coNP faces the same obstacle: Group Isomorphism is not known to be in coNP.

16.3 How hard is Graph Isomorphism?

Paradoxically, from a structural complexity point of view, GI (still) seems harder than factoring integers. The decision version of Factoring (given positive integers x, y, does x have divisor d in the interval $2 \leq d \leq y$?) is in $\mathrm{NP} \cap \mathrm{coNP}$ while the best we can say about GI

[12]In complexity theory, the "Group Isomorphism Problem" refers to groups given by Cayley tables; in other words, complexity is compared to the order of the group. From the point of view of applications, this complexity measure is of little use; in computational group theory, groups are usually given in compact representations (permutation groups, matrix groups given by lists of generators, p-groups given by power commutator presentation, etc.). But the fact remains that even in the unreasonably redundant representation by Cayley tables, we are unable to solve the problem is polynomial time.

[13]A simple algorithm, proposed by Tim Gowers on Dick Lipton's blog in November 2011, has a chance of running in $n^{O(\sqrt{\log n})}$. Let the *k-profile* of a finite group G be the function f on isomorphism types of k-generated groups where $f(H)$ counts those k-subsets of G that generate a subgroup isomorphic to H. For what k do k-profiles discriminate between nonisomorphic groups of order n? It is known that $k < (1/2)\sqrt{\log_2 n}$ is insufficient for infinitely many values of n (Glauberman, Grabowski [GlG]). Whether some k that is not much greater than $\sqrt{\log n}$ suffices is an open question that I think would deserve attention. The test case is p-groups of class 2; the Glauberman–Grabowski examples belong to this class.

is NP $\cap$ coAM. Factoring can be solved in polynomial time on a quantum computer, but no quantum advantage has yet been found for GI. On the other hand, apparently hard instances of factoring abound, whereas we don't know how to construct hard instances of GI. Could this be an indication that in structural complexity maybe we are not asking the right questions?

Even more baffling is another complexity arena, where GI is provably hard, on par with many NP-hard problems: relaxation hierarchies in proof complexity theory (Lovász–Schrijver, Sherali–Adams, Sum-of-Squares hierarchies). Building on the seminal paper by Cai, Furer, and Immerman [CaiFI], increasingly powerful hierarchies have recently been shown to be unable to refute isomorphism of graphs on sublinear levels [AtM, OWWZ, SnSC], showing that GI tests based on these hierarchies necessarily have exponential (even factorial) complexity. However, hard-to-distinguish CFI pairs of graphs and the related pairs of which isomorphism is hard to refute in these hierarchies are vertex-colored graphs with bounded color classes. Testing isomorphism of such pairs of graphs was shown to be in polynomial time via the first application of group theory (1979/80) that used hardly more than Lagrange's Theorem from group theory [Ba79a, FuHL]. One lesson is that these hierarchies have difficulty capturing the power of even the most naive applications of group theory. Given that hardness with respect to these hierarchies can now be proved by reduction from GI, this raises the question, in what sense these hierarchies indicate hardness.

16.4 Outlook

On the bright side, a number of GI-related questions may look a bit more hopeful now. While GI is complete over the isomorphism problems of *explicit structures*, there are interesting classes of non-explicit structures where progress may be possible. Two important examples are *equivalence of linear codes* and *conjugacy (permutational equivalence) of permutation groups.* The former easily reduces to the latter. Both of these problems belong[14] to NP $\cap$ coAM and therefore they are not NP-complete unless the polynomial-time hierarchy collapses. In spite of this complexity status, no moderately exponential ($\exp(n^{1-c})$) algorithm is known for either problem. GI reduces to each of these problems [Lu93][15]. Regarding both problems, see also [BaCGQ, BaCQ].

The present paper does not address the question of *canonical forms.* Do graphs permit quasipolynomial-time computable canonical forms?

It would be of great interest to find stronger structural results to better correspond to the "local $\to$ global symmetry" philosophy. This raises difficult mathematical questions that our algorithmic techniques bypass, but results of this flavor could make the algorithm more elegant and more efficient.

Finally a more concrete question. Let $\mathfrak{X} = (V; \mathcal{R})$ be a homogeneous coherent configuration with n vertices. Let $W \subseteq V$, $|W| \geq \alpha n$. Suppose that the induced configuration $\mathfrak{X}[W]$ is a Johnson scheme. Is there a constant $\alpha < 1$ such that this implies that $\mathfrak{X}$ itself is a Johnson scheme?

[14] To see that these problems belong to coAM, one can adapt the GMW protocol [GoMW] by conjugating the group by a random permutation and choosing a uniform random set of $O(n)$ generators.

[15] Luks's reduction is explained by Miyazaki in a post on The Math Forum, Sep. 29, 1996.

A result in this direction could be a step toward an elementary characterization of the Cameron groups as the only primitive groups of large order, or somewhat less ambitiously, an elementary characterization of the Johnson groups as the only primitive groups of large order, without an imprimitive subgroup of small index. Steps toward these goals have previously been made in [Ba81] for the case $|G| > \exp(n^{1/2+\epsilon})$ and in a remarkable recent paper by Sun and Wilmes [SuW] for the case $|G| > \exp(n^{1/3+\epsilon})$.

16.5 Analyze this!

The purpose of the present paper is to give a guaranteed upper bound (worst-case analysis); it does not contribute to practical solutions. It seems, for all practical purposes, the Graph Isomorphism problem is solved; a suite of remarkably efficient programs is available (nauty, saucy, Bliss, conauto, Traces). The article by McKay and Piperno [McP] gives a detailed comparison of methods and performance. Piperno's article [Pi] gives a detailed description of *Traces*, possibly the most successful program for large, difficult graphs.

These algorithms provide ingenious shortcuts in backtrack search. One of the most important questions facing the theorist in this area is to analyze these algorithms. While Miyazaki's graphs provide hard cases for the early version of *nauty*, the recent update overcomes that difficulty.

The question is, does there exist an infinite family of pairs of graphs on which these heuristic algorithms fail to perform efficiently? The search for such pairs might turn up interesting families of graphs.

Alternatively, can one prove strong worst-case upper bounds on the performance of any of these algorithms?

The comparison charts in [McP] seem to suggest that we lack true benchmarks – difficult classes of graphs on which to compare the algorithms. Encoding class-2 p-groups as graphs could provide quasipolynomially difficult examples, but right now we have no guarantee that the heuristics could not be tricked into much worse, (moderately?) exponential behavior.

17 Acknowledgments

17.1 May 2017

I am grateful to my colleagues Jin-Yi Cai, Gábor Tardos, and Harald Helfgott for their careful reading of (parts of) the paper and their comments. Jin-Yi found a mistake in the Design Lemma (fixed in the present version). Gábor reviewed the revised presentation of higher coherent configurations, including the Design Lemma, and made a long list of comments that helped greatly improve the presentation. My special gratitude is due to Harald, probably the only person in the world who carefully read the entire paper. His many questions helped significantly improve the presentation. Most importantly, he found a gap in the analysis of the Split-or-Johnson algorithm; this is fixed in the present version. The fix has also led to considerable simplification of the proof of that result.

Improvements of the exposition unrelated to the fixes above include the elimination of the "weak twin" relation, a simplified and more complete introduction to both the classical and

the higher coherent configurations, and a simplified analysis of the aggregation of positive certificates (Sec. 13.2, item 2b).

WARNING. The revisions of some sections caused notational, conceptual, and organizational inconsistencies with other sections; not all of these have been eliminated yet. A more detailed and more specific timing analysis is yet to be added. The present version is work in progress; I am posting it because it already includes the most needed fixes and improvements.

17.2 January 2016

I am happy to acknowledge the inspiration gained from my recent collaboration on the structure, automorphism group, and isomorphism problem for highly regular combinatorial structures with my student John Wilmes as well as with Xi Chen, Xiaorui Sun, and Shang-Hua Teng [BaW1, BaCh+, BaW2]. The recent breakthrough on primitive coherent configurations by Sun and Wilmes [SuW] was particularly encouraging; at one point during the weeks before the completion of the present work, it served as a tool to breaking the decades-old $\exp(\widetilde{O}(\sqrt{n}))$ barrier (see Remark 8.1.4).

The most direct forerunner of this paper was my joint work with Paolo Codenotti on hypergraph isomorphism [BaCo]; that paper combined the group theory method with a web of combinatorial partitioning techniques. In particular, I found an early version of the Design Lemma in the wake of that work. (A much simpler observation is called "Design Lemma" in that paper.)

Some of the group theory used in the present paper was inspired by my joint work with Péter Pál Pálfy and Jan Saxl [BaPS]; in particular, the rendering of a result of Feit and Tits [FeT] in that paper turned out to be particularly handy in the proof of the main group theoretic lemma of this paper ("Unaffected Stabilizers Lemma," Theorem 10.3.5).

I am grateful to three long-time friends who helped me verify critical parts of this paper: Péter Pál Pálfy and László Pyber the proof of various versions of the group-theoretic "Main structure theorem" (Theorem 10.5.1) that includes the crucial "Unaffected Stabilizers Lemma," and Gene Luks the LocalCertificates procedure (Sec. 13), the core algorithm of the paper. Their comments helped improve the presentation, and, more significantly, raised my confidence that these items actually work. All other parts of the paper seem quite "fault-tolerant," with multiple solutions, and a bag of tricks to rely on, should any gaps be found. Naturally, any errors that may remain in these items (or any other part of the paper) are my sole responsibility.

I wish to thank several colleagues, and especially Thomas Klimpel and Péter P. Pálfy, for their careful reading of parts of the first arXiv version and pointing out a large number of typos and some inaccuracies. The second arXiv version corrected these and added an occasional clarification.

New content added in the present (third) arXiv version icludes a

References

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

[AtM] Albert Atserias and Elitza Maneva: Graph Isomorphism, Sherali–Adams Relaxations and Indistinguishability in Counting Logics. *SIAM J. Comp.* **42(1)**, 2013, 112–137.

[Ba77] László Babai: On the isomorphism problem. Manuscript, 1977. Cited in [Mat]

[Ba79a] László Babai: Monte Carlo algorithms in graph isomorphism testing. Tech. Rep. 79–10, Dép. Math. et Stat., Université de Montréal, 1979 (pp. 42) `http://people.cs.uchicago.edu/~laci/lasvegas79.pdf`

[Ba79b] László Babai: Lectures on Graph Isomorphism. University of Toronto, Department of Computer Science. Mimeographed lecture notes, October 1979

[Ba80] László Babai: Almost all Steiner triple systems are asymmetric. In: *Topics on Steiner Systems* (C. C. Lindner and A. Rosa, eds.), *Annals of Discrete Math.* **7** (1980), 37-39.

[Ba81] László Babai: On the order of uniprimitive permutation groups. *Annals of Math.* **113(3)** (1981) 553–568. Updated version on author's website

[Ba83] László Babai: *Permutation Groups, Coherent Configurations and Graph Isomorphism.* D.Sc. Thesis (Hungarian), Hungarian Academy of Sciences, April 1983.

[Ba86] László Babai: On the length of subgroup chains in the symmetric group. *Communications in Algebra* **14** (1986) 1729–1736.

[Ba–] László Babai: Coset intersection in moderately exponential time. Manuscript, 2008. `http://people.cs.uchicago.edu/~laci/int.pdf`

[BaB] László Babai and Robert Beals: A polynomial-time theory of black box groups I. In: *Groups St Andrews 1997 in Bath, I* (C. M. Campbell *et al.*, eds.), pp. 30–64, London Math. Soc. Lecture Note Ser. Vol. 260, Cambridge U. Press, 1999.

[BaBS] László Babai, Robert Beals, Ákos Seress: Polynomial-Time Theory of Mtrix Groups (Extended Abstract). *In: Proc. 41st ACM STOC*, 2009, pp. 55–64.

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

[BaCh+] 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.

[BaCo] László Babai, Paolo Codenotti: Isomorphism of hypergraphs of low rank in moderately exponential time. *In: Proc. 49th IEEE FOCS*, 2008, pp. 667–676.

[BaCGQ] László Babai, Paolo Codenotti, Joshua A. Grochow, Youming Qiao: Code Equivalence and Group Isomorphism. *In: Proc. 22nd Ann. Symp. on Discrete Algorithms (SODA'11)*, ACM-SIAM, 2011, pp. 1395–1408.

[BaCQ] László Babai, Paolo Codenotti, Youming Qiao: Polynomial-time Isomorphism Test for Groups with no Abelian Normal Subgroups (Extended Abstract). *In: Proc. 39th Internat. Colloq. on Automata, Languages and Programming (ICALP'12)*, Springer LNCS 7391, 2012, pp. 51–62.

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

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

[BaLS] László Babai, Eugene M. Luks, Ákos Seress: Permutation groups in NC. *In: Proc. 19th ACM STOC*, 1987, pp. 409–420.

[BaPS] László Babai, Péter P. Pálfy, Jan Saxl: On the number of p-regular elements in finite simple groups. *LMS J. Comput. and Math.*, **12** (2009) 82–119.

[BaS] László Babai, Ákos Seress: On the degree of transitivity of permutation groups: a short proof. *J. Combinatorial Theory—A* **45** (1987) 310–315.

[BaW1] László Babai, John Wilmes: Quasipolynomial-time canonical form for Steiner designs. *In: Proc. 45th ACM STOC*, 2013, pp. 261–270.

[BaW2] László Babai, John Wilmes: Asymptotic Delsarte cliques in distance-regular graphs. *J. Algebraic Combinatorics,* to appear. See arXiv:1503.02746

[Bo89] Alfred Bochert: Über die Zahl verschiedener Werthe, die eine Funktion gegebener Buchstaben durch Vertauschung derselben erlangen kann. *Math. Ann.* **33** (1889) 584–590.

[Bo92] Alfred Bochert: *Math. Ann.* **40** (1892) 176–193

[Bo97] Alfred Bochert: Über die Classe der transitiven Substitutionengruppen II. *Math. Ann.* **49** (1897) 133–144.

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

[Cam80] Peter J. Cameron: Almost all Latin squares and Steiner Triple Systems are asymmetric. Unpublished manuscript, 1981.

[Cam81] Peter J. Cameron: Finite permutation groups and finite simple groups, *Bull. London Math Soc.* **13** (1981) 1–22.

[Cam11] PETER J. CAMERON: The symmetric group, 12. *Blog article*, 21/04/2011, http://cameroncounts.wordpress.com/2011/04/21/the-symmetric-group-12/

[ChST] 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.

[CuKS] CHARLES W. CURTIS, WILLIAM M. KANTOR, AND GARY M. SEITZ: The 2-transitive permutation representations of the finite Chevalley groups. *Trans. Amer. Math. Soc.* **218** (1976), 1–59

[DiM] JOHN D. DIXON, BRIAN MORTIMER: *Permutation Groups.* Springer Grad. Texts in Math. vol. 163, 1996

[FeT] WALTER FEIT AND JACQUES TITS: Projective representations of minimum degree of group extensions. *Canad. J. Math.* **30** (1978) 1092–1102.

[FuHL] MERRICK FURST, JOHN HOPCROFT, EUGENE LUKS: Polynomial-time algorithms for permutation groups. *In: Proc. 21st IEEE FOCS*, 1980, pp. 36–41.

[GlG] GEORGE GLAUBERMAN AND ŁUKASZ GRABOWSKI: Groups with identical k-profiles. Manuscript, 2015.

[GoMW] ODED GOLDREICH, SILVIO MICALI, AND AVI WIGDERSON: Proofs that yield nothing but their validity and a methodology of cryptographic protocol design. *In: Proc. 27th IEEE FOCS*, 1986, pp. 174–187.

[HoS] DEREK F. HOLT, MARK J. STATHER: Computing chief series and the soluble radical of a matrix group over a finite field. *LMS J. Computation and Mathematics* **11** (2008) pp. 223–251.

[HoT] JOHN HOPCROFT AND ROBERT E. TARJAN: Isomorphism of planar graphs. In *Complexity of Computer Computations* (R. Miller and J. W. Thatcher, eds.), Plenum Press 1972, pp. 131–152.

[ImL] NEIL IMMERMAN, ERIC S. LANDER: Describing graphs: a first-order approach to graph canonization. *In: Complexity Theory Retrospective — in honor of Juris Hartmanis on the occasion of his 60th birthday, July 5, 1988* (Alan Selman, ed.), Springer 1990, pp. 59–81.

[Jor] CAMILLE JORDAN: *Traité des substitutions et des equations algébriques.* Gauthier–Villars, 1870. (Reprinted 1957, Paris, Albert Blanchard)

[Jor2] CAMILLE JORDAN: Nouvelles recherches sur la limite de transivité des groupes qui ne contiennent pas le groupe alterné. *J. de Math. Pures et Appliquées* **1** (1895), 35–60. URL : http://eudml.org/doc/234202

[KlL] Peter Kleidman and Martin Liebeck: *The Subgroup Structure of the Finite Classical Groups.* London Math. Soc. Lecture Note Ser. Vol. 129, Cambridge Univ. Press, 1990.

[Kn] Donald E. Knuth: Efficient representation of perm groups. *Combinatorica* **11** (1991) 57–68.

[Lie83] Martin W. Liebeck: On graphs whose full automorphism group is an alternating group or a finite classical group. *Proc. London Math. Soc. (3)* **47** (1983) 337-362

[LiePS] Martin W. Liebeck, Cheryl E. Praeger, Jan Saxl: On the O'Nan–Scott theorem for finite primitive permutation groups. *J. Austral. Math. Soc. (A)* **44** (1988) 389–396

[Lu82] Eugene M. Luks: Isomorphism of graphs of bounded valence can be tested in polynomial time. *J. Comput. Syst. Sci.* **25(1)** (1982) 42–65.

[Lu87] Eugene M. Luks: Computing the composition factors of a permutation group in polynomial time. *Combinatorica* **7** (1987) 87–99.

[Lu93] Eugene M. Luks: Permutation groups and polynomial-time computation. *In: Groups and Computation,* DIMACS Ser. in Discr. Math. and Theor. Computer Sci. **11** (1993) 139–175.

[Lu99] Eugene M. Luks: Hypergraph Isomorphism and Structural Equivalence of Boolean Functions. *In: 31st ACM STOC*, 1999, pp. 652-658.

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

[Mat] Rudi Mathon: A note on the graph isomorphism counting problem. *Info. Proc. Lett.* **8** pp. 131–132.

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

[Me] Ulrich Meierfrankefeld: Non-finitary locally finite simple groups. *In: Finite and Locally Finite Groups.* B. Hartley et al., eds., Kluwer 1995, pp. 189–212.

[Mi] Takunari Miyazaki: Luks's reduction of Graph isomorphism to code equivalence. Comment on The Math Forum, Sep. 29, 1996. `http://mathforum.org/kb/thread.jspa?forumID=253&threadID=561418&messageID=1681072#1681072`

[OWWZ] Ryan O'Donnell, John Wright, Chenggang Wu, Yuan Zhou: Hardness of robust graph isomorphism, Lasserre gaps, and asymmetry of random graphs. *In: Proc. 25th ACM–SIAM Symp. Disr. Alg. (SODA'14)*, 2014, pp. 1659–1677.

[Pi] Adolfo Piperno: Search Space Contraction in Canonical Labeling of Graphs. arXiv:0804.4881, 2008, v2 2011.

[Py93] László Pyber: On the orders of doubly transitive permutation groups, elementary estimates. *J. Combinatorial Theory, Ser A* **62(2)** (1993) 361–366.

[Py17] László Pyber: A CFSG-free analysis of Babai's quasipolynomial GI-algorithm. arXiv:1605.08266, 2016, v2 2017

[Sch] Issai Schur: Zur Theorie der einfach transitiven Permutationsgruppen. *Sitzungsberichte der Preussischen Akademie der Wissenschaften, Physikalisch–Mathematische Klasse*, 1933, pp. 598–623.

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

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

[Si1] Charles C. Sims: Computation with Permutation Groups. *In: Proc. 2^{nd} Symp. Symb. Algeb. Manip.* (S. R. Petrick,ed.), ACM, New York, 1971, pp. 23–28.

[Si2] Charles C. Sims: Some group theoretic algorithms. *In: Lecture Notes in Math.* Vol. 697, Springer, 1978, pp. 108-124.

[SnSC] Aaron Snook, Grant Schoenebeck, Paolo Codenotti: Graph Isomorphism and the Lasserre Hierarchy. arXiv:1401.0758

[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: Proc. 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.

[Wi1] Helmut Wielandt: Abschätzungen für den Grad einer Permutationsgruppe von vorgeschriebenem Transitivitätsgrad. Dissertation, Berlin, 1934. *Schriften Math. Seminars Inst. Angew. Math. Univ. Berlin* **2** (1934) 151–174.

[Wi2] Helmut Wielandt: Über den Transitivitätsgrad von Permutationsgruppen. *Math. Z.* **74** (1960) 297–298.

[Wi3] Helmut Wielandt: *Finite Permutation Groups.* Acad. Press, New York 1964.

[ZKT] VIKTOR N. ZEMLYACHENKO, NIKOLAI M. KORNEENKO, REGINA I. TYSHKEVICH: Graph isomorphism problem. *Zapiski Nauchnykh Seminarov LOMI* **118** (1982) 83–158, 215.