2018 IEEE 59th Annual Symposium on Foundations of Computer Science

A Faster Isomorphism Test for Graphs of Small Degree

Martin Grohe, Daniel Neuen
RWTH Aachen University
Aachen, Germany
Email: {grohe,neuen}@informatik.rwth-aachen.de

Pascal Schweitzer
TU Kaiserslautern
Kaiserslautern, Germany
Email: schweitzer@cs.uni-kl.de

***Abstract*—In a recent breakthrough, Babai (STOC 2016) gave quasipolynomial graph isomorphism test. In this work, we give an improved isomorphism test for graphs of small degree: our algorithms runs in time $n^{\mathcal{O}((\log d)^c)}$, where n is the number of vertices of the input graphs, d is the maximum degree of the input graphs, and c is an absolute constant. The best previous isomorphism test for graphs of maximum degree d due to Babai, Kantor and Luks (FOCS 1983) runs in time $n^{\mathcal{O}(d/\log d)}$.**

***Keywords*-graph isomorphism, bounded degree graphs, group theory, groups with restricted composition factors**

I. Introduction

Luks's polynomial time isomorphism test for graphs of bounded degree [1] is one of the cornerstones of the algorithmic theory of graph isomorphism. With a slight improvement given later [2], it tests in time $n^{\mathcal{O}(d/\log d)}$ whether two n-vertex graphs of maximum degree d are isomorphic. Over the past decades Luks's algorithm and its algorithmic framework have been used as a building block for many isomorphism algorithms (see e.g. [2], [3], [4], [5], [6], [7], [8]). More importantly, it also forms the basis for Babai's recent isomorphism test for general graphs [9], [10] which runs in quasipolynomial time (i.e., the running time is bounded by $n^{\mathrm{polylog}(n)}$). Indeed, Babai's algorithm follows Luks's algorithm, but attacks the obstacle cases for which the recursion performed by Luks's framework does not lead to the desired running time. Graphs whose maximum degree d is at most polylogarithmic in the number n of vertices are not a critical case for Babai's algorithm, because for such graphs no large alternating or symmetric groups appear as factors of the automorphism group, and therefore the running time of Babai's algorithm on the class of all these graphs is still quasipolynomial. Hence graphs of polylogarithmic maximum degree form one of the obstacle cases towards improving Babai's algorithm. This alone is a strong motivation for trying to improve Luks's algorithm. In view of Babai's quasipolynomial time algorithm, it is natural to ask whether there is an $n^{\mathrm{polylog}(d)}$-isomorphism test for graphs of maximum degree d. In this paper we answer this question affirmatively.

Theorem I.1. *The Graph Isomorphism Problem for graphs of maximum degree d can be solved in time $n^{\mathcal{O}((\log d)^c)}$, for an absolute constant c.*

To prove the result we follow the standard route of considering the *String Isomorphism Problem*, which is an abstraction of the graph isomorphism problem that has been introduced by Luks in order to facilitate a recursive isomorphism test based on the structure of the permutation groups involved. Here a *string* is simply a mapping $\mathfrak{x} : \Omega \to \Sigma$, where the *domain* Ω and *alphabet* Σ are just finite sets. Given two strings $\mathfrak{x}, \mathfrak{y} : \Omega \to \Sigma$ and a permutation group $G \leq \mathrm{Sym}(\Omega)$ (given by a set of generators), the objective of the string isomorphism problem is to compute the set $\mathrm{Iso}_G(\mathfrak{x}, \mathfrak{y})$ of all *G-isomorphisms* from $\mathfrak{x}$ to $\mathfrak{y}$, that is, all permutations $g \in G$ mapping $\mathfrak{x}$ to $\mathfrak{y}$. We study the string isomorphism problem for groups G in the class $\widehat{\Gamma}_d$ of groups all of whose composition factors are isomorphic to subgroups of S_d, the symmetric group acting on d points. Luks introduced this class because he observed that, after fixing a single vertex, the automorphism group of a connected graph of maximum degree d is in $\widehat{\Gamma}_d$[1]. Our main technical result, Theorem VII.3, states that we can solve the string isomorphism problem for groups $G \in \widehat{\Gamma}_d$ in time $n^{\mathrm{polylog}(d)}$, where $n = |\Omega|$ is the length of the input strings. This implies Theorem I.1 (as outlined in Section VIII).

To prove this result, we introduce the new concept of an *almost d-ary sequence* of invariant partitions. More precisely, we exploit for the group G a sequence $\{\Omega\} = \mathfrak{B}_0 \succ \cdots \succ \mathfrak{B}_m = \{\{\alpha\} \mid \alpha \in \Omega\}$ of G-invariant partitions $\mathfrak{B}_i$ of Ω, where $\mathfrak{B}_{i-1} \succ \mathfrak{B}_i$ means that $\mathfrak{B}_i$ refines $\mathfrak{B}_{i-1}$. For this sequence we require that for all i the induced group of permutations of the subclasses in $\mathfrak{B}_i$ of a given class in $\mathfrak{B}_{i-1}$ is isomorphic to a subgroup of the symmetric group S_d or semi-regular (i.e., only the identity has fixed points). Our algorithm that exploits such a sequence is heavily based on techniques introduced by Babai for his quasipolynomial time isomorphism test. We even use Babai's algorithm as a black box in one case. One of our technical contributions is an adaptation of Babai's Unaffected Stabilizers Theorem [10, Theorem 6] to groups constrained by an almost d-ary sequence of invariant partitions. In [10], the Unaffected

[1] In [1], the class $\widehat{\Gamma}_d$ is denoted by Γ_d. However, in the more recent literature Γ_d typically refers to a larger class of groups [11] (see Subsection II-B).

2575-8454/18/$31.00 ©2018 IEEE
DOI 10.1109/FOCS.2018.00018

Stabilizers Theorem lays the groundwork for the group theoretic algorithms (the Local Certificates routine), and it plays a similar role here. However, we need a more refined running time analysis. Based on this we can then adapt the Local Certificates routine to our setting.

However, not every group in $\widehat{\Gamma}_d$ has such an almost d-ary sequence required by our technique. We remedy this by changing the operation of the group while preserving string isomorphisms. The structural and algorithmic results enabling such a change of operation form the second technical contribution of our work. For this we employ some heavy group theoretic results. First, applying the classification of finite simple groups via the O'Nan Scott Theorem and several other group theoretic characterizations, we obtain a structure theorem for primitive permutation groups in $\widehat{\Gamma}_d$ showing that they are either small (of size at most $n^{\mathrm{polylog}(d)}$) or have a specific structure. More precisely, large primitive groups in $\widehat{\Gamma}_d$ are composed, in a well defined manner, of Johnson schemes. Second, to construct the almost d-ary sequence of partitions, we exploit the existence of these Johnson schemes and introduce subset lattices which are unfolded yielding the desired group operation.

With Luks's framework being used as a subroutine in various other algorithms, one can ask for the impact of the improved running time in such contexts. As a first, simple application we obtain an improved isomorphism test for relational structures (Theorem VIII.3) and hypergraphs (Corollary VIII.4). A deeper application is a new fixed-parameter tractable algorithm for graph isomorphism of graphs parameterized by tree width [12], which substantially improves the algorithm from [13].

II. Preliminaries

A. Graphs and other structures

A *graph* is a pair $\Gamma = (V, E)$ with vertex set $V = V(\Gamma)$ and edge relation $E = E(\Gamma)$. In this paper all graphs are finite simple, undirected graphs. The *neighborhood* of $v \in V(\Gamma)$ is denoted $N(v)$. A *path* of length k is a sequence $v_0, \dots, v_k$ of distinct vertices such that $v_{i-1}v_i \in E(\Gamma)$ for all $i \in [k]$ (where $[k] := \{1, \dots, k\}$). The *distance* between two vertices $v, w \in V(\Gamma)$, denoted by $\mathrm{dist}(v, w)$, is the length of the shortest path from v to w.

An *isomorphism* from a graph Γ_1 to another graph Γ_2 is a bijective mapping $\varphi\colon V(\Gamma_1) \to V(\Gamma_2)$ which preserves the edge relation, that is $vw \in E(\Gamma_1)$ if and only if $\varphi(v)\varphi(w) \in E(\Gamma_2)$ for all $v, w \in V(\Gamma_1)$. Two graphs Γ_1 and Γ_2 are *isomorphic* ($\Gamma_1 \cong \Gamma_2$) if there is an isomorphism from Γ_1 to Γ_2. An *automorphism* of a graph Γ is an isomorphism from Γ to itself. By $\mathrm{Aut}(\Gamma)$ we denote the group of automorphisms of Γ. The *Graph Isomorphism Problem* asks, given two graphs Γ_1 and Γ_2, whether they are isomorphic.

More generally, a *t-ary relational structure* is a tuple $\mathfrak{A} = (D, R_1, \dots, R_k)$ with domain D and t-ary relations $R_i \subseteq D^t$ for $i \in [k]$. An *isomorphism* from a structure $\mathfrak{A}_1 = (D_1, R_1, \dots, R_k)$ to another structure $\mathfrak{A}_2 = (D_2, S_1, \dots, S_k)$ is a bijective mapping $\varphi\colon D_1 \to D_2$ such that $(v_1, \dots, v_t) \in R_i$ if and only if $(\varphi(v_1), \dots, \varphi(v_t)) \in S_i$ for all $v_1, \dots, v_t \in D_1$ and $i \in [k]$. As before, $\mathrm{Aut}(\mathfrak{A})$ denotes the automorphism group of $\mathfrak{A}$.

Let $\mathfrak{B}_1, \mathfrak{B}_2$ be two partitions of the same set Ω. We say $\mathfrak{B}_1$ *refines* $\mathfrak{B}_2$, denoted by $\mathfrak{B}_1 \preceq \mathfrak{B}_2$, if for every $B_1 \in \mathfrak{B}_1$ there is some $B_2 \in \mathfrak{B}_2$ such that $B_1 \subseteq B_2$. If additionally $\mathfrak{B}_1$ and $\mathfrak{B}_2$ are distinct we say $\mathfrak{B}_1$ *strictly refines* $\mathfrak{B}_2$ ($\mathfrak{B}_1 \prec \mathfrak{B}_2$). The *index* of $\mathfrak{B}_1$ in $\mathfrak{B}_2$ is $|\mathfrak{B}_2 : \mathfrak{B}_1| = \max_{B_2 \in \mathfrak{B}_2} |\{B_1 \in \mathfrak{B}_1 \mid B_1 \subseteq B_2\}|$. A partition $\mathfrak{B}$ (of the set Ω) is an *equipartition* if all elements $B \in \mathfrak{B}$ have the same size. For $S \subseteq \Omega$ we define the *induced partition* $\mathfrak{B}[S] = \{B \cap S \mid B \in \mathfrak{B} \text{ such that } B \cap S \neq \emptyset\}$. Note that $\mathfrak{B}[S]$ forms a partition of the set S.

For a set M and a natural number $t \leq |M|$ we denote by $\binom{M}{t}$ the set of all t-element subsets of M, that is, $\binom{M}{t} = \{X \subseteq M \mid |X| = t\}$. Note that the number of elements in $\binom{M}{t}$ is exactly $\binom{|M|}{t}$. Moreover, $\binom{M}{\leq t}$ denotes the set of all subsets of M of cardinality at most t.

B. Group Theory

In this section we introduce the group theoretic notions required in this work. For a general background on group theory we refer to [14] whereas background on permutation groups can be found in [15].

A *permutation group* acting on a set Ω is a subgroup $G \leq \mathrm{Sym}(\Omega)$ of the symmetric group. The size of the permutation domain Ω is called the *degree* of G and, throughout this work, is denoted by $n = |\Omega|$. If $\Omega = [n]$ then we also write S_n instead of $\mathrm{Sym}(\Omega)$. For $g \in G$ and $\alpha \in \Omega$ we denote by α^g the image of α under the permutation g. The set $\alpha^G = \{\alpha^g \mid g \in G\}$ is the *orbit* of α. The group G is *transitive* if $\alpha^G = \Omega$ for some (and therefore every) $\alpha \in \Omega$.

For $\alpha \in \Omega$ the group $G_\alpha = \{g \in G \mid \alpha^g = \alpha\} \leq G$ is the *stabilizer* of α in G. For $\Delta \subseteq \Omega$ and $g \in G$ let $\Delta^g = \{\alpha^g \mid \alpha \in \Delta\}$. The *pointwise stabilizer* of Δ is the subgroup $G_{(\Delta)} = \{g \in G \mid \forall \alpha \in \Delta\colon \alpha^g = \alpha\}$. The *setwise stabilizer* of Δ is the subgroup $G_\Delta = \{g \in G \mid \Delta^g = \Delta\}$. Observe that $G_{(\Delta)} \leq G_\Delta$.

Let $G \leq \mathrm{Sym}(\Omega)$ be a transitive group. A *block* of G is a nonempty subset $B \subseteq \Omega$ such that $B^g = B$ or $B^g \cap B = \emptyset$ for all $g \in G$. The trivial blocks are Ω and the singletons $\{\alpha\}$ for $\alpha \in \Omega$. The group G is said to be *primitive* if there are no non-trivial blocks. If $B \subseteq \Omega$ is a block of G then $\mathfrak{B} = \{B^g \mid g \in G\}$ forms a *block system* of G. The group $G_{(\mathfrak{B})} = \{g \in G \mid \forall B \in \mathfrak{B}\colon B^g = B\}$ denotes the subgroup stabilizing each block $B \in \mathfrak{B}$ setwise. Observe that $G_{(\mathfrak{B})}$ is a normal subgroup of G. We denote by $G^{\mathfrak{B}} \leq \mathrm{Sym}(\mathfrak{B})$ the natural action of G on the block system $\mathfrak{B}$. More generally, if A is a set of objects on which G acts naturally, we denote by $G^A \leq \mathrm{Sym}(A)$ the action of G on the set A. A block system $\mathfrak{B}$ is *minimal* if there is no non-trivial block system

$\mathfrak{B}'$ such that $\mathfrak{B} \prec \mathfrak{B}'$. A block system $\mathfrak{B}$ is minimal if and only if $G^{\mathfrak{B}}$ is primitive.

Let $G \leq \mathrm{Sym}(\Omega)$ and $G' \leq \mathrm{Sym}(\Omega')$. A *homomorphism* is a mapping $\varphi\colon G \to G'$ such that $\varphi(g)\varphi(h) = \varphi(gh)$ for all $g, h \in G$. For $g \in G$ we denote by g^φ the φ-image of g. Similarly, for $H \leq G$ we denote by H^φ the φ-image of H (note that H^φ is a subgroup of G').

A *permutational isomorphism* from G to G' is a bijective mapping $f : \Omega \to \Omega'$ such that $G' = \{f^{-1}gf \mid g \in G\}$ where $f^{-1}gf\colon \Omega' \to \Omega'\colon f(\alpha) \mapsto f(\alpha^g)$. If there is a permutational isomorphism from G to G', we call G and G' *permutationally equivalent*.

In this work we shall be interested in a particular subclass of permutation groups, namely groups with restricted composition factors. Let G be a group. A *normal series* is a sequence of subgroups $G = G_0 \trianglerighteq G_1 \trianglerighteq \cdots \trianglerighteq G_k = \{1\}$. The length of the series is k and the groups G_{i-1}/G_i are the factor groups of the series, $i \in [k]$. A *composition series* is a strictly decreasing normal series of maximal length. For every finite group G all composition series have the same family of factor groups considered as a multi-set (cf. [14]). A *composition factor* of a finite group G is a factor group of a composition series of G.

Definition II.1. For $d \geq 2$ let $\widehat{\Gamma}_d$ denote the class of all groups G for which every composition factor of G is isomorphic to a subgroup of S_d.

We want to stress the fact that there are two similar classes of groups that have been used in the literature both typically denoted by Γ_d. One of these is the class we define as $\widehat{\Gamma}_d$ introduced by Luks [1] while the other one used in [11] in particular allows composition factors that are simple groups of Lie type of bounded dimension.

Lemma II.2 (Luks [1])**.** *Let $G \in \widehat{\Gamma}_d$. Then*

1) *$H \in \widehat{\Gamma}_d$ for every subgroup $H \leq G$, and*
2) *$G^\varphi \in \widehat{\Gamma}_d$ for every homomorphism $\varphi\colon G \to H$.*

C. String Isomorphism and Luks's algorithm

In the following we give an outline of Luks's algorithm [1]. Our description of the algorithm as well as the notation mainly follows [10].

Let $\mathfrak{x}, \mathfrak{y}\colon \Omega \to \Sigma$ be two strings over a finite alphabet Σ and let $G \leq \mathrm{Sym}(\Omega)$ be a group. For $\sigma \in \mathrm{Sym}(\Omega)$ the string $\mathfrak{x}^\sigma$ is defined by

$$\mathfrak{x}^\sigma(\alpha) = \mathfrak{x}(\alpha^{\sigma^{-1}})$$

for all $\alpha \in \Omega$. A permutation $\sigma \in \mathrm{Sym}(\Omega)$ is a *G-isomorphism* from $\mathfrak{x}$ to $\mathfrak{y}$ if $\sigma \in G$ and $\mathfrak{x}^\sigma = \mathfrak{y}$. The *String Isomorphism Problem* asks, given $\mathfrak{x}, \mathfrak{y}\colon \Omega \to \Sigma$ and a group $G \leq \mathrm{Sym}(\Omega)$ given as a set of generators, whether there is a G-isomorphism from $\mathfrak{x}$ to $\mathfrak{y}$. The set of G-isomorphisms is denoted by $\mathrm{Iso}_G(\mathfrak{x}, \mathfrak{y}) := \{g \in G \mid \mathfrak{x}^g = \mathfrak{y}\}$.

More generally, for $K \subseteq \mathrm{Sym}(\Omega)$ and $W \subseteq \Omega$ we define

$$\mathrm{Iso}_K^W(\mathfrak{x}, \mathfrak{y}) = \{g \in K \mid \forall \alpha \in W\colon \mathfrak{x}(\alpha) = \mathfrak{y}(\alpha^g)\}. \quad (1)$$

In this work $K = Gg$ will always be a coset where $G \leq \mathrm{Sym}(\Omega)$ and $g \in \mathrm{Sym}(\Omega)$ and the set W will be G-invariant. In this case $\mathrm{Iso}_K^W(\mathfrak{x}, \mathfrak{y})$ is either empty or a coset of the group $\mathrm{Aut}_G^W(\mathfrak{x}) := \mathrm{Iso}_G^W(\mathfrak{x}, \mathfrak{x})$, that is, $\mathrm{Iso}_K^W(\mathfrak{x}, \mathfrak{y}) = \mathrm{Aut}_G^W(\mathfrak{x})\sigma$ where $\sigma \in \mathrm{Iso}_K^W(\mathfrak{x}, \mathfrak{y})$ is arbitrary. Hence, the set $\mathrm{Iso}_K^W(\mathfrak{x}, \mathfrak{y})$ can be represented by a generating set for $\mathrm{Aut}_G^W(\mathfrak{x})$ and an element σ. Moreover, using the identity

$$\mathrm{Iso}_{Gg}^W(\mathfrak{x}, \mathfrak{y}) = \mathrm{Iso}_G^W(\mathfrak{x}, \mathfrak{y}^{g^{-1}})g, \quad (2)$$

it is actually possible to restrict ourselves to the case where K is a group.

We now describe the two main recursive steps used in Luks's algorithm [1]. First suppose $G \leq \mathrm{Sym}(\Omega)$ is not transitive and let $\Omega_1, \ldots, \Omega_s$ be the orbits of G. Then the strings are processed orbit by orbit.

```
K := G
for all i = 1 to s do
    K := Iso_K^{Ω_i}(𝔵, 𝔶)
end for
```

Note that the set $\mathrm{Iso}_K^{\Omega_i}(\mathfrak{x}, \mathfrak{y})$ can be computed making one call to String Isomorphism over domain size $n_i = |\Omega_i|$. Indeed, using Equation (2), it can be assumed that $K \leq \mathrm{Sym}(\Omega)$ is a group and Ω_i is K-invariant. Then

$$\mathrm{Iso}_K^{\Omega_i}(\mathfrak{x}, \mathfrak{y}) = \left\{k \in K \mid k^{\Omega_i} \in \mathrm{Iso}_{K^{\Omega_i}}(\mathfrak{x}^{\Omega_i}, \mathfrak{y}^{\Omega_i})\right\}.$$

Here, $\mathfrak{x}^{\Omega_i}$ (respectively $\mathfrak{y}^{\Omega_i}$) denotes the restriction of the string $\mathfrak{x}$ (respectively $\mathfrak{y}$) to the set Ω_i. Having computed the set $\mathrm{Iso}_{K^{\Omega_i}}(\mathfrak{x}^{\Omega_i}, \mathfrak{y}^{\Omega_i})$ making one recursive call to String Isomorphism over domain size $n_i = |\Omega_i|$, the set $\mathrm{Iso}_K^{\Omega_i}(\mathfrak{x}, \mathfrak{y})$ can be computed in polynomial time. So overall the algorithm needs to make s recursive calls to String Isomorphism over domain sizes $n_1, \ldots, n_s$.

For the second type of recursion let $H \leq G$ be a subgroup and let $T = \{g_1, \ldots, g_t\}$ be a transversal for H. Then

$$\mathrm{Iso}_G(\mathfrak{x}, \mathfrak{y}) = \bigcup_{i \in [t]} \mathrm{Iso}_{Hg_i}(\mathfrak{x}, \mathfrak{y}). \quad (3)$$

In Luks's algorithm this type of recursion is applied when G is a transitive group, $\mathfrak{B}$ is a minimal block system and $H = G_{(\mathfrak{B})}$. Observe that $G^{\mathfrak{B}}$ is a primitive group and $t = |G^{\mathfrak{B}}|$. Also note that H is not transitive. Indeed, each orbit of H has size n/b where $b = |\mathfrak{B}|$. Hence, combining both types of recursion the computation of $\mathrm{Iso}_G(\mathfrak{x}, \mathfrak{y})$ is reduced to $t \cdot b$ instances of String Isomorphism over domain size n/b. We refer to this as the *standard Luks reduction*.

Now suppose $G \in \widehat{\Gamma}_d$. The crucial step to analyze Luks's algorithm is to determine the size of primitive groups occurring in the recursion.

Theorem II.3 ([11])**.** *There exists a function f such that every primitive $\widehat{\Gamma}_d$-group $G \leq \mathrm{Sym}(\Omega)$ has order $|G| \leq n^{f(d)}$.*

Indeed, the function f can be chosen to be linear in d (cf. [16]). As a result, Luks's algorithm runs in time $n^{\mathcal{O}(d)}$ for all groups $G \in \widehat{\Gamma}_d$.

D. Recursion

For the purpose of later analyzing our recursion, we record some bounds.

Lemma II.4. *Let $k \in \mathbb{N}$ and $t\colon \mathbb{N} \to \mathbb{N}$ be a function such that $t(1) = 1$. Suppose that for every $n \in \mathbb{N}$ there are natural numbers $n_1, \dots, n_\ell$ for which one of the following holds:*

1) *$t(n) \le \sum_{i=1}^{\ell} t(n_i)$ where $\sum_{i=1}^{\ell} n_i \le 2^k n$ and $n_i \le n/2$ for all $i \in [\ell]$, or*
2) *$t(n) \le \sum_{i=1}^{\ell} t(n_i)$ where $\sum_{i=1}^{\ell} n_i \le n$ and $\ell \ge 2$.*

Then $t(n) \le n^{k+1}$.

Lemma II.5. *Let $m, k \ge 1$ and suppose $k \le \frac{m}{2}$. Then*

$$\binom{m}{k}^{\log m} \ge m^k. \tag{4}$$

Proof: It holds that

$$\binom{m}{k}^{\log m} \ge \left(\frac{m}{k}\right)^{k \log m} \ge 2^{k \log m} = m^k.$$

■

III. The structure of primitive groups in $\widehat{\Gamma}_d$

We give a structure theorem for primitive $\widehat{\Gamma}_d$-groups saying that they are either small or essentially composed of Johnson groups in a well defined manner.

For $t \le m$ we denote by $A_m^{(t)}$ the action of the alternating group A_m on the set of t-element subsets of $[m]$. Also, for $G \le \mathrm{Sym}(\Omega)$ and $\mathfrak{B}, \mathfrak{B}'$ two G-invariant partitions such that $\mathfrak{B} \prec \mathfrak{B}'$, we denote by $G_B^{\mathfrak{B}[B]}$ the natural induced action of G_B on the set $\mathfrak{B}[B]$ for all $B \in \mathfrak{B}'$. Finally, recall that the socle of G, denoted by $\mathrm{Soc}(G)$, is the group generated by all minimal normal subgroups of G.

Theorem III.1. *Let $G \le \mathrm{Sym}(\Omega)$ be a primitive $\widehat{\Gamma}_d$-group. Then one of the following holds:*

1) *$|G| = n^{\mathcal{O}(\log d)}$, or*
2) *for the normal subgroup $N = \mathrm{Soc}(G) \le G$ there is a sequence of partitions $\{\Omega\} = \mathfrak{B}_1 \succ \cdots \succ \mathfrak{B}_k = \{\{\alpha\} \mid \alpha \in \Omega\}$ such that the following holds:*
   a) *$|G : N| \le n^{1+\log d}$,*
   b) *$\mathfrak{B}_i$ is N-invariant for every $i \in [k]$, and*
   c) *there are $m \le d$ and $t \le \frac{m}{2}$ with $m > 4 \log s$ where $s = \binom{m}{t}$ such that for all $i \in [k-1]$ and $B \in \mathfrak{B}_i$ the group $N_B^{\mathfrak{B}_{i+1}[B]}$ is permutationally equivalent to $A_m^{(t)}$.*

Moreover, there is a polynomial-time algorithm that determines one of the options that is satisfied and in case of the second option computes N and the partitions $\mathfrak{B}_1, \dots, \mathfrak{B}_k$.

The proof is based on the well-known O'Nan-Scott Theorem saying a primitive group G has to be one of five types. For each of those types we separately analyze the size of primitive groups of this type and give a precise description of the large groups. The proofs for the different types are based on several further group theoretic statements [17], [18], [19], [20], [21], [22], [23], [24] several of which are based on the classification of finite simple groups.

Remark III.2. Let Γ_d denote the family of groups G with the property that G has no alternating composition factors of degree greater than d and no classical composition factors of rank greater than d. (There is no restriction on the cyclic, exceptional, and sporadic composition factors of G.) While the class $\widehat{\Gamma}_d$ considered in this paper follows the original definition of Luks [1], most of the recent literature is concerned with the more general class of groups Γ_d [11], [21]. The reason is that many results that can be proved for the class $\widehat{\Gamma}_d$ indeed carry over to the more general class of groups Γ_d. We want to stress the fact that this is not the case for the theorem presented in this section. Indeed, consider the affine general linear group $G = \mathrm{AGL}(d, p)$ of dimension d (with its natural action on the corresponding vector space). Then G is a primitive group of affine type and $|G| = n^{\Omega(d)}$ where $n = p^d$ is the size of the vector space. The group G is contained in the class Γ_d, but it is not contained in $\widehat{\Gamma}_d$.

IV. Almost d-ary block system sequences

In this section we describe a reduction from the String Isomorphism Problem for $\widehat{\Gamma}_d$-groups to a more restricted version of this problem. In this restricted version, the group is equipped with a sequence of block systems satisfying a particular property defined as follows. (Recall that a permutation group $G \le \mathrm{Sym}(\Omega)$ is semi-regular if $G_\alpha = \{1\}$ for every $\alpha \in \Omega$. Also remember that, for G-invariant partitions $\mathfrak{B} \prec \mathfrak{B}'$ and $B \in \mathfrak{B}'$, we denote by $G_B^{\mathfrak{B}[B]}$ the natural induced action of G_B on the set $\mathfrak{B}[B]$.)

Definition IV.1. Let $G \le \mathrm{Sym}(\Omega)$ be a permutation group. A G-invariant sequence of partitions $\{\Omega\} = \mathfrak{B}_0 \succ \cdots \succ \mathfrak{B}_k = \{\{\alpha\} \mid \alpha \in \Omega\}$ is called *almost d-ary* if for every $i \in [k]$ and $B \in \mathfrak{B}_{i-1}$ it holds that

1) $G_B^{\mathfrak{B}_i[B]}$ is semi-regular, or
2) $|\mathfrak{B}_i[B]| \le d$.

A simple, but crucial observation is that the property, that such a sequence exists, is closed under taking subgroups and under restricting the group to an invariant subset of the domain.

Observation IV.2. *Let $G \le \mathrm{Sym}(\Omega)$ be a group and suppose there is an almost d-ary sequence of G-invariant partitions $\{\Omega\} = \mathfrak{B}_0 \succ \cdots \succ \mathfrak{B}_m = \{\{\alpha\} \mid \alpha \in \Omega\}$. Moreover, let $H \le G$. Then $\mathfrak{B}_0 \succ \cdots \succ \mathfrak{B}_m$ also forms an almost d-ary sequence of H-invariant partitions. Additionally, for an H-invariant subset $\Delta \subseteq \Omega$ it holds that*

$\mathfrak{B}_0[\Delta] \succeq \cdots \succeq \mathfrak{B}_m[\Delta]$ *forms an almost* d*-ary sequence of* H^{Δ}*-invariant partitions.*

The goal of this section is to describe a reduction that, given an instance of String Isomorphism for $\widehat{\Gamma}_d$-groups, computes a new equivalent instance, in which the permutation group is equipped with an almost d-*ary* G-invariant sequence of partitions. This reduction runs in time $n^{\mathrm{polylog}(d)}$. We shall then see in subsequent sections that the String Isomorphism Problem for groups equipped with such a sequence can be solved in time $n^{\mathrm{polylog}(d)}$.

High Level Idea.: The central idea for the reduction is to change the action of the permutation group G. Let us first illustrate this on a high level for the special case that G is a primitive group. Using the characterization of primitive $\widehat{\Gamma}_d$-groups given in the previous section we have to distinguish two cases. First suppose that $|G| \leq n^{c_1 \log d + c_2}$ for some appropriate absolute constants c_1, c_2. Now define $\Omega^* = G \times \Omega$. Then $g \in G$ acts on Ω^* via

$$(h, \alpha)^g = (hg, \alpha^g).$$

Let $G^* \leq \mathrm{Sym}(\Omega^*)$ be the permutation group obtained from the action of G on the set Ω^*. It is easy to check that G^* is semi-regular. Also note that $|\Omega^*| \leq n^{\mathcal{O}(\log d)}$. Of course we also need to transform the strings. For a string $\mathfrak{x} \colon \Omega \to \Sigma$ define $\mathfrak{x}^* \colon \Omega^* \to \Sigma \colon (h, \alpha) \mapsto \mathfrak{x}(\alpha)$. Note that no information is lost in this transformation. Indeed, it can be verified that two strings $\mathfrak{x}, \mathfrak{y}$ are G-isomorphic if and only if $\mathfrak{x}^*$ is G^*-isomorphic to $\mathfrak{y}^*$. So this gives us the desired reduction.

Next, let us consider the more interesting case that G satisfies Property 2 of Theorem III.1. Let $N = \mathrm{Soc}(G)$. Then, in a first step, we consider the set $\Omega^* = G/N \times \Omega$. An element $g \in G$ acts on Ω^* via

$$(Nh, \alpha)^g = (Nhg, \alpha^g).$$

Let $G^* = G^{\Omega^*} \leq \mathrm{Sym}(\Omega^*)$ denote the permutation group corresponding to the action of G on Ω^*. Now the crucial observation is that $\mathfrak{B} = \{\{(Nh, \alpha) \mid \alpha \in \Omega\} \mid h \in G\}$ is a G^*-invariant partition. For every $B \in \mathfrak{B}$, it holds that $(G^*)_B^B$ is permutationally equivalent to N, and the group $(G^*)^{\mathfrak{B}}$ is regular. Note that again $|\Omega^*| \leq n^{\mathcal{O}(\log d)}$. Also, the strings can be transformed in the same way as before. Hence, it remains to consider only the group N.

Finally, for an intuition on how the group N is transformed suppose for simplicity that $N = A_m^{(t)}$. The group A_m has another action closely related to the action $A_m^{(t)}$ on the t-element subsets of $[m]$, namely the action on the set $[m]^{\langle t \rangle}$ of all t-tuples with distinct entries. A crucial difference between these actions is that the action on the tuples is not primitive. Indeed, fixing more and more coordinates, we get the following sequence of partitions. For $i = 0, \ldots, t$ let

$$\mathfrak{B}_i^* = \big\{\{(a_1, \ldots, a_t) \in [m]^{\langle t \rangle} \mid \forall j \leq i \colon a_j = b_j\} \mid (b_1, \ldots, b_i) \in [m]^{\langle i \rangle}\big\}.$$

Let N^* be the action of N on the set of ordered t-tuples with distinct entries. For every $i \in [t]$ the partition $\mathfrak{B}_i^*$ is N^*-invariant and for every $B \in \mathfrak{B}_{i-1}^*$ it holds that $|\mathfrak{B}_i^*[B]| \leq m \leq d$. Moreover, with every element $\bar{a} \in [m]^{\langle t \rangle}$ we can associate the underlying unordered set of elements. This way, we can also transform the strings in a way similar to before. Also note that the set $[m]^{\langle t \rangle}$ is only slightly larger than $\binom{m}{t}$ (cf. Lemma II.5).

Theorem IV.3. *Let* $G \leq \mathrm{Sym}(\Omega)$ *be a transitive* $\widehat{\Gamma}_d$*-group and let* $\mathfrak{x}, \mathfrak{y} \colon \Omega \to \Sigma$ *be two strings. Then there is a set* Ω^**, a* $\widehat{\Gamma}_d$*-group* $G^* \leq \mathrm{Sym}(\Omega^*)$*, two strings* $\mathfrak{x}^*, \mathfrak{y}^* \colon \Omega^* \to \Sigma$ *and a* G^**-invariant almost* d-ary *sequence of partitions* $\{\Omega^*\} = \mathfrak{B}_0^* \succ \cdots \succ \mathfrak{B}_k^* = \{\{\alpha^*\} \mid \alpha^* \in \Omega^*\}$ *of the set* Ω^* *such that the following holds:*

1) $|\Omega^*| \leq n^{(c_1 \log d + c_2 + 1) \log d}$*, and*
2) $\mathfrak{x} \cong_G \mathfrak{y}$ *if and only if* $\mathfrak{x}^* \cong_{G^*} \mathfrak{y}^*$.

Moreover, one can compute all objects in time polynomial in the size of Ω^*.

While the proof follows the high level idea presented above there are several intricacies that need to be considered when dealing with non-primitive groups.

For the proof we consider a maximal sequence of G-invariant partitions $\{\Omega\} = \mathfrak{B}_0 \succ \cdots \succ \mathfrak{B}_k = \{\{\alpha\} \mid \alpha \in \Omega\}$ and change the action of the group along this sequence. A main additional challenge comes from the fact that, starting from the second partition in the sequence, one has to change the action in several blocks in parallel in a consistent manner. For example, suppose $|\mathfrak{B}_1| \leq d$ and thus, the action on the block system $\mathfrak{B}_1$ remains unchanged. In the next iteration we wish to modify the actions $G_B^{\mathfrak{B}_2[B]}$ for every $B \in \mathfrak{B}_1$. To achieve this, we need to change the actions of all these groups at the same time in a consistent manner to obtain an action of the complete group G. In the proof, we actually split this task into two separate steps similar to the description given above. First, we introduce the semi-regular actions and reduce to the case where we only have to deal with the Johnson groups $A_m^{(t)}$. Then, in a second step, we deal with the Johnson groups separately using tree unfoldings of graphs composed of subset lattices (that correspond to the Johnson schemes) to obtain the desired group action and the sequence of partitions.

More formally, in the first step we proof the following theorem.

Theorem IV.4. *Let* $G \leq \mathrm{Sym}(\Omega)$ *be a transitive* $\widehat{\Gamma}_d$*-group and let* $\mathfrak{x}, \mathfrak{y} \colon \Omega \to \Sigma$ *be two strings. Then there is a set* Ω^**, a transitive* $\widehat{\Gamma}_d$*-group* $G^* \leq \mathrm{Sym}(\Omega^*)$*, two strings* $\mathfrak{x}^*, \mathfrak{y}^* \colon \Omega^* \to \Sigma$ *and a sequence of* G^**-invariant partitions*

$\{\Omega^*\} = \mathfrak{B}_0^* \succ \cdots \succ \mathfrak{B}_k^* = \{\{\alpha^*\} \mid \alpha^* \in \Omega^*\}$ *of the set* Ω^* *such that the following holds:*

1) $|\Omega^*| \leq n^{c_1 \log d + c_2 + 1}$ *for some absolute constants* c_1, c_2 *where* $n = |\Omega|$,
2) $\mathfrak{x} \cong_G \mathfrak{y}$ *if and only if* $\mathfrak{x}^* \cong_{G^*} \mathfrak{y}^*$, *and*
3) *for every* $i \in [k]$ *and* $B \in \mathfrak{B}_{i-1}^*$ *it holds that*
   a) $(G^*)_B^{\mathfrak{B}_i^*[B]}$ *is semi-regular, or*
   b) $(G^*)_B^{\mathfrak{B}_i^*[B]}$ *is permutationally equivalent to* $A_m^{(t)}$ *for some* $m \leq d$ *and* $t \leq \frac{m}{2}$ *where* $m > 4 \log s$ *for* $s = \binom{m}{t}$.

Moreover, one can compute all objects in time polynomial in the size of Ω^*.

With this theorem we can prove the main reduction theorem by elimating the Johnson groups. In the following we outline the proof of this second step by describing the construction of the new instance more formally.

Proof Idea of Theorem IV.3: By Theorem IV.4 we can assume that there is a sequence of G-invariant partitions $\{\Omega\} = \mathfrak{B}_0 \succ \cdots \succ \mathfrak{B}_\ell = \{\{\alpha\} \mid \alpha \in \Omega\}$ such that for every $i \in [\ell]$ and $B \in \mathfrak{B}_{i-1}$ it holds that

(A) $G_B^{\mathfrak{B}_i[B]}$ is semi-regular, or
(B) $G_B^{\mathfrak{B}_i[B]}$ is permutationally equivalent to $A_m^{(t)}$ for some $m \leq d$ and $t \leq \frac{m}{2}$ where $m > 4 \log s$ for $s = \binom{m}{t}$.

(Actually, using Theorem IV.4, the above condition can only be achieved by increasing the size of the set Ω as described in Theorem IV.4, Property 1. We argue that under the above assumption the set Ω^* constructed in this proof has size at most $n^{\log d}$ which in combination with Theorem IV.4 results in the desired bound given in 1.)

In order to get almost d-arity, we need to worry about those blocks that satisfy item (B). Let

$$I = \big\{ i \in [\ell] \mid \exists B \in \mathfrak{B}_{i-1} \colon G_B^{\mathfrak{B}_i[B]} \text{ is permutationally equivalent to } A_{m_i}^{(t_i)} \big\}.$$

Note that for $B, B' \in \mathfrak{B}_{i-1}$ the groups $G_B^{\mathfrak{B}_i[B]}$ and $G_{B'}^{\mathfrak{B}_i[B']}$ are permutationally equivalent. So the existential quantifier in the definition of the set I can also be replaced by a universal quantifier.

For $i \in I$ and $B \in \mathfrak{B}_{i-1}$ let $\rho_{i,B} \colon \mathfrak{B}_i[B] \to \binom{[m_i]}{t_i}$ be a permutational isomorphism from $G_B^{\mathfrak{B}_i[B]}$ to $A_{m_i}^{(t_i)}$. Such a $\rho_{i,B}$ can be computed in polynomial time (see e.g. [25]). Let $\Gamma = (V(\Gamma), E(\Gamma))$ be the graph with

$$V(\Gamma) = \bigcup_{i \in \{0,\dots,\ell\}} \mathfrak{B}_i \cup \left\{ (i, B, X) \mid i \in I, B \in \mathfrak{B}_{i-1}, X \in \binom{[m_i]}{\leq t_i} \right\}$$

and

$$\begin{aligned}
&\{(i,B,X),(i',B',X')\} \in E(\Gamma) \\
&\qquad :\Leftrightarrow \quad i = i' \wedge B = B' \wedge X \subseteq X' \wedge |X' \setminus X| = 1, \\
&\{B,(i,B',X)\} \in E(\Gamma) \\
&\qquad :\Leftrightarrow \quad B = B' \wedge X = \emptyset \\
&\qquad\qquad \text{or } |X| = t_i \wedge B \in \mathfrak{B}_i \wedge B \subseteq B' \wedge \rho_{i,B'}(B) = X, \\
&\{B,B'\} \in E(\Gamma) \\
&\qquad :\Leftrightarrow \quad \exists i \in [\ell] \setminus I \colon B \in \mathfrak{B}_{i-1} \wedge B' \in \mathfrak{B}_i \wedge B' \subseteq B.
\end{aligned}$$

Let $v_0 = \Omega$ be the root of Γ (note that $\Omega \in \mathfrak{B}_0$). A *branch of* (Γ, v_0) is a path $(v_0, v_1, \dots, v_p)$ such that $\operatorname{dist}(v_0, v_i) = i$ for all $i \in [p]$. A *maximal branch of* (Γ, v_0) is a branch of maximal length. Observe that for every maximal branch $(v_0, v_1, \dots, v_p)$ it holds that $v_p = \{\alpha\}$ for some $\alpha \in \Omega$. Let M be the set of maximal branches of (Γ, v_0).

Claim 1. $|M| \leq n^{\log d}$.

Proof. We can view the sequence of partitions $\mathfrak{B}_i$ as a tree of height ℓ. Each leaf of this tree corresponds to an element $\alpha \in \Omega$.

The graph Γ is obtained from the partition tree by squeezing subset-lattices of the $(\leq t_i)$-element subsets of $[m_i]$ between some internal node of the tree and its $\binom{m_i}{t_i}$ children. Counting the number of branches in Γ amounts to counting the number of leaves in the tree unfolding of Γ. To obtain the tree unfolding, we replace each of the subset lattices of size $\binom{m_i}{t_i}$ by a tree of size $m_i^{t_i}$. For a fixed subset lattice every element $X \subseteq [m_i]$ of size t_i corresponds to $m_i^{t_i} / \binom{m_i}{t_i}$ many tuples in the tree unfolding. Hence,

$$\begin{aligned}
|M| &\leq n \cdot \prod_{i \in I} \left(m_i^{t_i} / \binom{m_i}{t_i} \right) \\
&\leq n \cdot \prod_{i \in I} \binom{m_i}{t_i}^{\log d - 1} && \text{by Lemma II.5} \\
&\leq n \cdot \left(\prod_{i \in I} \binom{m_i}{t_i} \right)^{\log d - 1} \\
&\leq n \cdot \left(\prod_{i \in I} |\mathfrak{B}_{i-1} : \mathfrak{B}_i| \right)^{\log d - 1} \\
&\leq n^{\log d}.
\end{aligned}$$

⌟

For every maximal branch $\bar{v} = (v_0, \dots, v_p) \in M$ define $\sigma(\bar{v}) = \alpha$ for the unique $\alpha \in \Omega$ such that $v_p = \{\alpha\}$. Now let $\Omega^* = \{(\alpha, \bar{v}) \mid \alpha \in \Omega, \bar{v} \in M, \alpha = \sigma(\bar{v})\}$. Clearly, $|\Omega^*| = |M| \leq n^{\log d}$ by Claim 1. Let $\mathfrak{x}^* \colon \Omega^* \to \Sigma \colon (\alpha, \bar{v}) \mapsto \mathfrak{x}(\alpha)$ and $\mathfrak{y}^* \colon \Omega^* \to \Sigma \colon (\alpha, \bar{v}) \mapsto \mathfrak{y}(\alpha)$.

For $g \in G$ define $g^\Gamma \in \operatorname{Sym}(V(\Gamma))$ to be the permutation defined by

$$B^{(g^\Gamma)} = B^g$$

and

$$(i, B, X)^{(g^\Gamma)} = (i, B^g, X')$$

where X' is defined as follows. Let $g^{\mathfrak{B}_i[B]} \colon \mathfrak{B}_i[B] \to \mathfrak{B}_i[B^g] \colon B' \mapsto (B')^g$ and define

$$f \colon \binom{[m_i]}{t_i} \to \binom{[m_i]}{t_i} \colon Y \mapsto Y^{\rho_{i,B}^{-1} \cdot g^{\mathfrak{B}_i[B]} \cdot \rho_{i,B^g}}$$

The bijection $f \in \mathrm{Sym}(\binom{[m_i]}{t_i})$ is induced by a unique permutation $\pi \in S_{m_i}$. Now define $X' = X^\pi$.

We can show that for every $g \in G$ we have $g^\Gamma \in \mathrm{Aut}(\Gamma, v_0)$. With this we can define an action of the group G on the set of maximal branches and thus, on the set Ω^*. For $g \in G$ define $g^* \in \mathrm{Sym}(\Omega^*)$ via

$$(\alpha, (v_0, \ldots, v_p))^{g^*} = \left(\alpha^g, \left(v_0^{(g^\Gamma)}, \ldots, v_p^{(g^\Gamma)}\right)\right)$$

and let $G^* = \{g^* \mid g \in G\}$.

Form this point it is not difficult to prove that the constructed instance of the String Isomorphism Problem satisfies the desired properties. In particular, the almost d-ary sequence of partitions is naturally defined via the set of maximal branches coming from the tree unfolding of the graph Γ. ∎

The previous theorem states that there is an $n^{\mathrm{polylog}(d)}$-reduction from the String Isomorphism Problem for $\widehat{\Gamma}_d$-groups to the String Isomorphism Problem for groups where we are additionally given an almost d-ary sequence of invariant partitions. Hence, in the remainder of this work, we shall be concerned with solving the latter problem. The basic approach to do this is to adapt the Local Certificates Routine developed by Babai for his quasipolynomial time isomorphism test [10].

V. Affected Orbits

The basis of Babai's Local Certificates algorithm is a group theoretic statement, the Unaffected Stabilizers Theorem (see [10, Theorem 6]). In the following we generalize this theorem to our setting.

Definition V.1 ([10])**.** Let $G \leq \mathrm{Sym}(\Omega)$. A homomorphism $\varphi \colon G \to S_k$ is a *giant representation* if $G^\varphi \geq A_k$. In this case an element $\alpha \in \Omega$ is *affected by* φ if $G_\alpha^\varphi \not\geq A_k$.

Remark V.2. Let $\varphi \colon G \to S_k$ be a giant representation and suppose $\alpha \in \Omega$ is affected by φ. Then every element in the orbit α^G is affected by φ. We call α^G an *affected orbit* (with respect to φ).

With this definition we can state the generalization of the Unaffected Stabilizers Theorem (see [10, Theorem 6]).

Theorem V.3. *Let $G \leq \mathrm{Sym}(\Omega)$ be a permutation group and suppose there is an almost d-ary sequence of G-invariant partitions $\{\Omega\} = \mathfrak{B}_0 \succ \cdots \succ \mathfrak{B}_m = \{\{\alpha\} \mid \alpha \in \Omega\}$. Furthermore let $k > \max\{8, 2 + \log_2 d\}$ and $\varphi \colon G \to S_k$ be a giant representation. Let $D \subseteq \Omega$ be the set of elements not affected by φ. Then $G_{(D)}^\varphi \geq A_k$.*

For the proof we roughly follow the argumentation from [9]. However, on the technical level, several details need to be changed to allow for the treatment of the semi-regular operations allowed in our setting. A crucial part of the proof is to show the following Lemma which is similar in nature to [9, Lemma 8.3.1].

Lemma V.4. *Let $G \leq \mathrm{Sym}(\Omega)$ be a transitive group and suppose there is an almost d-ary sequence of invariant partitions $\{\Omega\} = \mathfrak{B}_0 \succ \cdots \succ \mathfrak{B}_m = \{\{\alpha\} \mid \alpha \in \Omega\}$. Furthermore let $k > \max\{8, 2+\log_2 d\}$, and let $\varphi \colon G \to A_k$ be an epimorphism. Then $G_\alpha^\varphi \neq A_k$ for all $\alpha \in \Omega$.*

Actually, the proof of the previous lemma even builds on [9, Lemma 8.3.1].

Lemma V.5 ([9], Lemma 8.3.1)**.** *Let $G \leq S_d$ be a transitive group and $\varphi \colon G \to A_k$ an epimorphism where $k > \max\{8, 2 + \log_2 d\}$. Then $G_\alpha^\varphi \neq A_k$ for all $\alpha \in [d]$.*

Moreover, we shall need the following two lemmas.

Lemma V.6 (cf. [9], [26])**.** *Let $G \leq K_1 \times \cdots \times K_\ell$ be a subdirect product and let $\varphi \colon G \to S$ be an epimorphism where S is a non-abelian simple group. Furthermore let $\pi_i \colon G \to K_i$ be the projection to the i-th component and $M_i = \ker(\pi_i)$. Then there is some $i^* \in [\ell]$ such that $M_{i^*} \leq \ker(\varphi)$.*

Lemma V.7. *Let G be a group, $H, K \trianglelefteq G$ and suppose $\varphi \colon G \to S$ is an epimorphism where S is a non-abelian simple group. Furthermore suppose that $H^\varphi = K^\varphi = S$. Then $(H \cap K)^\varphi = S$.*

Proof: Let $N = \ker(\varphi)$. Suppose that $(H \cap K)^\varphi \neq S$. Since $H \cap K \trianglelefteq G$ and S is a simple group we conclude that $(H \cap K)^\varphi = \{1\}$, that is, $H \cap K \leq N$.

Now let $s_1, s_2 \in S$ be two arbitrary elements. Then there are $h \in H$, $k \in K$ such that $\varphi(h) = s_1$ and $\varphi(k) = s_2$. Moreover, $h^{-1}k^{-1}hk \in H \cap K$ since $H \trianglelefteq G$ and $K \trianglelefteq G$. Hence, there is some $n \in (H \cap K) \leq N$ such that $hk = khn$. But then $s_1 s_2 = \varphi(h)\varphi(k) = \varphi(hk) = \varphi(khn) = \varphi(k)\varphi(h)\varphi(n) = s_2 s_1$. Since $s_1, s_2 \in S$ were chosen arbitrarily it follows that S is abelian. ∎

Proof of Lemma V.4: We prove the statement by induction on the cardinality of G. Let $K = G_{(\mathfrak{B}_1)}$ be the normal subgroup stabilizing the block system $\mathfrak{B}_1$ and $N = \ker(\varphi)$. Observe that N is a maximal normal subgroup of G ($N \trianglelefteq G$ is a maximal normal subgroup of G if and only if the quotient group G/N is simple; here G/N is isomorphic to $G^\varphi = A_k$). Hence, it holds that $K \leq N$ or $\langle K, N \rangle = KN = G$.

First suppose $K \leq N$. Then φ factors across $G \to G^{\mathfrak{B}_1} \xrightarrow{\psi} A_k$. Observe that ψ is an epimorphism since φ is an epimorphism. Suppose $|\mathfrak{B}_1| \leq d$. Then, by Lemma

V.5, for every $B \in \mathfrak{B}_1$ it holds that $(G^{\mathfrak{B}_1})^{\psi}_B \neq A_k$. Hence, $G^{\varphi}_{\alpha} \leq G^{\varphi}_B \neq A_k$ where $B \in \mathfrak{B}_1$ is the unique set such that $\alpha \in B$. Otherwise $G^{\mathfrak{B}_1}$ is semi-regular and hence, $(G^{\mathfrak{B}_1})^{\psi}_B = \{1\} \neq A_k$ for all $B \in \mathfrak{B}_1$. Again, $G^{\varphi}_{\alpha} \leq G^{\varphi}_B \neq A_k$ where $B \in \mathfrak{B}_1$ is the unique set such that $\alpha \in B$.

So consider the case that $KN = G$, that is, $K^{\varphi} = A_k$. Suppose towards a contradiction that there is some $\alpha \in \Omega$ such that $G^{\varphi}_{\alpha} = A_k$. Pick $B \in \mathfrak{B}_1$ such that $\alpha \in B$. In particular, $G^{\varphi}_B = A_k$.

Claim 1. $G^{\varphi}_{(B)} \neq A_k$.

Proof. Assume towards a contradiction that $G^{\varphi}_{(B)} = A_k$. Then, by Lemma V.7, $K^{\varphi}_{(B)} = (G_{(B)} \cap K)^{\varphi} = A_k$ since $G_{(B)} \trianglelefteq G_B$, $K \trianglelefteq G_B$ and $K^{\varphi} = A_k$.

On the other hand, let $\Omega_1, \ldots, \Omega_\ell$ be the orbits of K. Let $\pi_i \colon K \to \mathrm{Sym}(\Omega_i)$ be the restriction of K to Ω_i, $K_i = \mathrm{im}(\pi_i)$ and $M_i = \ker(\pi_i)$. By Lemma V.6 there is some $i \in [\ell]$ such that $M_i \leq N$. Since G acts transitively on the blocks $\{\Omega_1, \ldots, \Omega_\ell\}$ the groups M_i, $i \in [\ell]$, are conjugate subgroups in G and therefore $M_i \leq N$ for all $i \in [\ell]$. Pick $i^* \in [\ell]$ such that $\alpha \in \Omega_{i^*}$. Since $M_{i^*} \leq N$ the epimorphism $\varphi|_K \colon K \to A_k$ factors across K_{i^*} as $K \xrightarrow{\pi_{i^*}} K_{i^*} \xrightarrow{\psi} A_k$. Hence, $K^{\psi}_{i^*} = A_k$. Moreover, $\mathfrak{B}_1[\Omega_{i^*}] \succ \cdots \succ \mathfrak{B}_m[\Omega_{i^*}]$ is an almost d-ary sequence of partitions for K_{i^*}. By the induction hypothesis it follows that $(K_{i^*})^{\psi}_{\alpha} \neq A_k$ and thus, $K^{\varphi}_{\alpha} \neq A_k$. But this is a contradiction since $K^{\varphi}_{(B)} \leq K^{\varphi}_{\alpha}$. ⌟

Since $G^{\varphi}_{(B)} \trianglelefteq G^{\varphi}_B$ it follows $G^{\varphi}_{(B)} = \{1\}$. So $\varphi|_{G_B}$ factors across $G_B \to G^B_B \xrightarrow{\psi} A_k$. Moreover, $\varphi|_{G_\alpha}$ factors across $G_\alpha \to G^B_\alpha \xrightarrow{\psi'} A_k$, where $\psi' = \psi|_{G^B_\alpha}$. Overall this means $(G^B_B)^{\psi} = A_k$ and $(G^B_B)^{\psi}_{\alpha} = ((G_\alpha)^B)^{\psi'} = A_k$. But this contradicts the induction hypothesis since $\mathfrak{B}_1[B] \succ \cdots \succ \mathfrak{B}_m[B]$ is an almost d-ary sequence of G^B_B-invariant partitions and G^B_B is transitive. ∎

We also use Babai's Affected Orbit Lemma, which does not need to be adapted to our setting.

Theorem V.8 ([10, Theorem 6(b)])**.** *Let $G \leq \mathrm{Sym}(\Omega)$ be a permutation group and suppose $\varphi \colon G \to S_k$ is a giant representation for $k \geq 5$. Suppose $\Delta \subseteq \Omega$ is an affected orbit of G (with respect to φ). Then every orbit of $\ker(\varphi)$ in Δ has length at most $|\Delta|/k$.*

VI. Local Certificates

Based on the generalization to the Unaffected Stabilizers Theorem presented in the previous section we can adapt the Local Certificates technique developed in [10] to our setting. Besides the adaptation to our setting, the main difference is a more precise analysis of the running time which is required for our overall analysis. Before doing so, we need to introduce some notation, which follows the one in [9].

Let $G \leq \mathrm{Sym}(\Omega)$ be a permutation group and let $\mathfrak{x} \colon \Omega \to \Sigma$ be a string. Furthermore let $\varphi \colon G \to S_k$ be a giant representation. For a set $T \subseteq [k]$ let $G_T = \varphi^{-1}((G^{\varphi})_T)$. Similarly we define $G_{(T)} = \varphi^{-1}((G^{\varphi})_{(T)})$.

For a set Δ we denote by $\mathrm{Alt}(\Delta)$ the alternating group acting with its standard action on the set Δ. Moreover, we refer to the groups $\mathrm{Alt}(\Delta)$ and $\mathrm{Sym}(\Delta)$ as the *giants* where Δ is an arbitrary finite set.

Definition VI.1. A set $T \subseteq [k]$ is *full* if $((\mathrm{Aut}_{G_T}(\mathfrak{x}))^{\varphi})^T \geq \mathrm{Alt}(T)$. A *certificate of fullness* is a subgroup $K \leq \mathrm{Aut}_{G_T}(\mathfrak{x})$ such that $(K^{\varphi})^T \geq \mathrm{Alt}(T)$. A *certificate of non-fullness* is a non-giant $M \leq \mathrm{Sym}(T)$ such that $((\mathrm{Aut}_{G_T}(\mathfrak{x}))^{\varphi})^T \leq M$.

Let $W \subseteq \Omega$ be G-invariant and let $\mathfrak{y} \colon \Omega \to \Sigma$ be a second string. Recall that $\mathrm{Iso}^W_G(\mathfrak{x}, \mathfrak{y}) = \{g \in G \mid \forall \alpha \in W \colon \mathfrak{x}(\alpha) = \mathfrak{y}(\alpha^g)\}$ and $\mathrm{Aut}^W_G(\mathfrak{x}) = \mathrm{Iso}^W_G(\mathfrak{x}, \mathfrak{x})$.

For $H \leq G$ we define $\mathrm{Aff}(H, \varphi) = \{\alpha \in \Omega \mid H^{\varphi}_{\alpha} \not\geq A_k\}$. Note that for $H_1 \leq H_2 \leq G$ it holds that $\mathrm{Aff}(H_1, \varphi) \supseteq \mathrm{Aff}(H_2, \varphi)$.

The Local Certificates algorithm is used to determine whether a given test set T is full and to produce a corresponding certificate.

Lemma VI.2. *Let $\mathfrak{x} \colon \Omega \to \Sigma$ be a string, $G \leq \mathrm{Sym}(\Omega)$ be a group and suppose there is an almost d-ary sequence of G-invariant partitions $\{\Omega\} = \mathfrak{B}_0 \succ \cdots \succ \mathfrak{B}_m = \{\{\alpha\} \mid \alpha \in \Omega\}$. Furthermore suppose there is a giant representation $\varphi \colon G \to S_k$ and let $T \subseteq [k]$ be a set of size $|T| = t > \max\{8, 2 + \log_2 d\}$.*

Then there are natural numbers $n_1, \ldots, n_\ell \leq n/2$ such that $\sum_{i=1}^{\ell} n_i \leq n$ and, for each $i \in [\ell]$ using at most $t!$ recursive calls to String Isomorphism over domain size n_i and $\mathcal{O}(t! \cdot n^c)$ additional computation, one can decide whether T is full or not and generate a corresponding certificate.

Proof: Without loss of generality assume $T = [k]$. Otherwise one can compute the group G_T and restrict the image of φ to the set T.

Consider the algorithm given in Figure 1. The algorithm computes, for increasing windows $W_0 \subseteq W_1 \subseteq W_2 \subseteq \ldots$, the group G_i of permutations that respect the input string $\mathfrak{x}$ on the window W_i, that is, $G_i = \mathrm{Aut}^{W_i}_G(\mathfrak{x}) = \mathrm{Aut}^{W_i^*}_{G_{i-1}}(\mathfrak{x})$. Note that $G_i \leq G_{i-1}$ and therefore $W_{i+1} \supseteq W_i$ for $i \geq 1$ (initially $W_1 \neq \emptyset$ since at least one point has to be affected). The algorithm stops when the current group G^{φ}_i is not a giant or the window stops growing.

Let i^* be the value of the variable i at the end of while-loop. Furthermore let $W = W_{i^*}$. Note that $\{W^*_j \mid 1 \leq j \leq i^*\}$ forms a partition of the set W.

We first show the correctness of the algorithm. For every $0 \leq j \leq i^*$ it holds that $\mathrm{Aut}_G(\mathfrak{x}) \leq G_j \leq G$. We distinguish two cases. First suppose that $G^{\varphi}_{i^*} \not\geq A_k$. Then $G^{\varphi}_{i^*}$ forms a certificate of non-fullness. Otherwise $G^{\varphi}_{i^*} \geq A_k$ and $W = \mathrm{Aff}(G_{i^*}, \varphi)$. Note that $\mathfrak{B}_0, \ldots, \mathfrak{B}_m$ forms an almost d-ary sequence of invariant partitions for the group

Input: $G \leq \mathrm{Sym}(\Omega)$, $\mathfrak{x}\colon \Omega \to \Sigma$, and $\varphi\colon G \to S_k$ with $k > \max\{8, 2 + \log_2 d\}$. There exists an almost d-ary sequence of G-invariant partitions $\{\Omega\} = \mathfrak{B}_0 \succ \cdots \succ \mathfrak{B}_m = \{\{\alpha\} \mid \alpha \in \Omega\}$.

Output: non-giant $M \leq S_k$ with $(\mathrm{Aut}_G(\mathfrak{x}))^\varphi \leq M$ or $K \leq \mathrm{Aut}_G(\mathfrak{x})$ with $K^\varphi \geq A_k$.

1: $G_0 := G$
2: $W_0 := \emptyset$
3: $i := 0$
4: **while** $G_i^\varphi \geq A_k$ **and** $W_i \neq \mathrm{Aff}(G_i, \varphi)$ **do**
5: $\quad W_{i+1} := \mathrm{Aff}(G_i, \varphi)$
6: $\quad W_{i+1}^* := W_{i+1} \setminus W_i$
7: $\quad$ **if** $|W_{i+1}^*| \leq \frac{1}{2}|\Omega|$ **then**
8: $\quad\quad G_{i+1} := \mathrm{Aut}_{G_i}^{W_{i+1}^*}(\mathfrak{x})$
9: $\quad$ **else**
10: $\quad\quad G_{i+1} := \emptyset$
11: $\quad\quad N := \ker(\varphi|_{G_i})$
12: $\quad\quad$ **for** $g \in G_i^\varphi$ **do**
13: $\quad\quad\quad$ compute $\bar{g} \in \varphi^{-1}(g)$
14: $\quad\quad\quad G_{i+1} := G_{i+1} \cup \mathrm{Aut}_{N\bar{g}}^{W_{i+1}^*}(\mathfrak{x})$
15: $\quad\quad$ **end for**
16: $\quad$ **end if**
17: $\quad i := i + 1$
18: **end while**
19: **if** $G_i^\varphi \not\geq A_k$ **then**
20: $\quad$ **return** G_i^φ
21: **else**
22: $\quad$ **return** $(G_i)_{(\Omega \setminus W_i)}$
23: **end if**

Figure 1. The LocalCertificates algorithm

G_{i^*} (cf. Observation IV.2). So $((G_{i^*})_{(\Omega\setminus W)})^\varphi \geq A_k$ by Theorem V.3. Furthermore, it easy to check that G_{i^*} respects the string $\mathfrak{x}$ on all positions in W_j for all $0 \leq j \leq i^*$. Hence, $(G_{i^*})_{(\Omega\setminus W)} \leq \mathrm{Aut}_G(\mathfrak{x})$ because it respects all positions within W and fixes all other positions.

It remains to analyze the running time of the algorithm. Again we distinguish two cases. First suppose $|W_j^*| \leq n/2$ for all $j \in [i^*]$. Then, for each $j \in [i^*]$, the algorithm makes one recursive call to String Isomorphism over domain size $|W_i^*| \leq n/2$ (Line 8) and $\sum_{j\in[i^*]} |W_j^*| \leq |W| \leq n$. Otherwise there is a unique $j^* \in \{0, \ldots, i^* - 1\}$ such that $|W_{j^*+1}^*| > n/2$. Let $N = \ker(\varphi|_{G_{j^*}})$. Since all elements in $W_{j^*+1}^*$ are affected by φ with respect to G_{j^*} it holds that every orbit of N in $W_{j^*+1}^*$ has size at most $|W_{j^*+1}^*|/k$ by Theorem V.8. For each orbit the algorithm makes $k!$ calls to String Isomorphism where the domain is restricted to exactly this orbit (Line 14). Additionally, for every $j \in [i^*], j \neq j^* + 1$ there is one recursive call to String Isomorphism over domain size $|W_j^*|$. ∎

Similar to the previous lemma we can also adapt the algorithms to compare local certificates and to aggregate the local certificates (cf. [10]). From the aggregation algorithm we obtain the following lemma.

The *symmetry defect* of a group $G \leq \mathrm{Sym}(\Omega)$ is the minimal $t \in [n]$ such that there is a set $\Delta \subseteq \Omega$ of size $|\Delta| = n - t$ such that $\mathrm{Alt}(\Delta) \leq G$ (the group $\mathrm{Alt}(\Delta)$ fixes all elements of $\Omega \setminus \Delta$). In this case the *relative symmetry defect* is t/n. For a relational structure $\mathfrak{A}$ we define the *(relative) symmetry defect* of $\mathfrak{A}$ to be the (relative) symmetry defect of its automorphism group $\mathrm{Aut}(\mathfrak{A})$.

Lemma VI.3. *Let $\mathfrak{x}_1, \mathfrak{x}_2\colon \Omega \to \Sigma$ be two strings, $G \leq \mathrm{Sym}(\Omega)$ be a group and suppose there is an almost d-ary sequence of G-invariant partitions $\{\Omega\} = \mathfrak{B}_0 \succ \cdots \succ \mathfrak{B}_m = \{\{\alpha\} \mid \alpha \in \Omega\}$. Furthermore suppose there is a giant representation $\varphi\colon G \to S_k$. Let $\max\{8, 2 + \log_2 d\} < t < k/10$.*

Then there are natural numbers $\ell \in \mathbb{N}$ and $n_1, \ldots, n_\ell \leq n/2$ such that $\sum_{i=1}^{\ell} n_i \leq k^{\mathcal{O}(t)} n$ and, for each $i \in [\ell]$ using a recursive call to String Isomorphism over domain size n_i, and $k^{\mathcal{O}(t)} n^c$ additional computation, one obtains for $i = 1, 2$ one of the following:

1) *a family of $r \leq k^6$ many t-ary relational structures $\mathfrak{A}_{i,j}$, for $j \in [r]$, associated with $\mathfrak{x}_i$, each with domain $D_{i,j} \subseteq [k]$ of size $|D_{i,j}| \geq \frac{3}{4}k$ and with relative symmetry defect at least $\frac{1}{4}$ such that*

$$\{\mathfrak{A}_{1,1}, \ldots, \mathfrak{A}_{1,r}\}^{\varphi(g)} = \{\mathfrak{A}_{2,1}, \ldots, \mathfrak{A}_{2,r}\}$$

for every $g \in \mathrm{Iso}_G(\mathfrak{x}_1, \mathfrak{x}_2)$, or

2) *a subset $\Delta_i \subseteq [k]$ associated with $\mathfrak{x}_i$ of size $|\Delta_i| \geq \frac{3}{4}k$ and $K_i \leq \mathrm{Aut}_{G_{\Delta_i}}(\mathfrak{x}_i)$ such that $(K_i^\varphi)^{\Delta_i} \geq \mathrm{Alt}(\Delta_i)$ and*

$$\Delta_1^{\varphi(g)} = \Delta_2$$

for every $g \in \mathrm{Iso}_G(\mathfrak{x}_1, \mathfrak{x}_2)$.

The aggregation algorithm either results in a small family of t-ary relational structures (where $t = \Theta(\log d)$) or finds many G-automorphisms of the input strings. In the former case we use Babai's algorithm as a black box to decide isomorphism of the relational structures to significantly reduce the size of the input group G. More precisely, we obtain the following statement.

Lemma VI.4. *Suppose Option 1 of Lemma VI.3 is satisfied, yielding a number $r \leq k^6$ and relational structures $\mathfrak{A}_{i,j}$ for $i \in [2], j \in [r]$. Then there are subgroups $H_j \leq G$ and elements $h_j \in \mathrm{Sym}(\Omega)$ for $j \in [r]$ such that*

1) *$|G^\varphi : H_j^\varphi| \geq (4/3)^k$ for all $j \in [r]$, and*
2) *$\mathfrak{x}_1 \cong_G \mathfrak{x}_2$ if and only if $\mathfrak{x}_1 \cong_{H_j h_j} \mathfrak{x}_2$ for some $j \in [r]$, and given representations for the sets $\mathrm{Iso}_{H_j h_j}(\mathfrak{x}_1, \mathfrak{x}_2)$ for all $j \in [r]$ one can compute in polynomial time a representation for $\mathrm{Iso}_G(\mathfrak{x}_1, \mathfrak{x}_2)$.*

Moreover, given the relational structures $\mathfrak{A}_{i,j}$ for all $i \in [2]$ and $j \in [r]$, the groups H_j and elements h_j can be computed in time $k^{\mathcal{O}(t^c (\log k)^c)} n^c$ for some constant c.

Remark VI.5. The proof of the previous lemma is the only place where we use Babai's quasipolynomial time isomorphism test [10] as a black box.

In the other case we utilize the symmetries of the input strings to make significant progress.

Lemma VI.6. *Suppose Option 2 of Lemma VI.3 is satisfied. Then there is a number $r \in \{1,2\}$, a subgroup $H \leq G$ and elements $h_j \in \mathrm{Sym}(\Omega)$ for $j \in [r]$ such that*

1) *$|G^\varphi : H^\varphi| \geq (4/3)^k$, and*
2) *$\mathfrak{x}_1 \cong_G \mathfrak{x}_2$ if and only if $\mathfrak{x}_1 \cong_{Hh_j} \mathfrak{x}_2$ for some $j \in [r]$, and given representations for the sets $\mathrm{Iso}_{Hh_j}(\mathfrak{x}_1, \mathfrak{x}_2)$ for all $j \in [r]$ and a generating set for K_1 one can compute in polynomial time a representation for $\mathrm{Iso}_G(\mathfrak{x}_1, \mathfrak{x}_2)$.*

Moreover, given the sets Δ_i for all $i \in [2]$, the group H and the elements h_i can be computed in polynomial time.

Proof: We let $H = G_{(\Delta_1)}$ (recall that $G_{(T)} = \varphi^{-1}((G^\varphi)_{(T)})$ for $T \subseteq [k]$). Let $g \in G$ such that $\Delta_1^{g^\varphi} = \Delta_2$ and $\tau \in G_{\Delta_1}$ such that $(\tau^\varphi)^{\Delta_1}$ is a transposition. Now define $h_1 = g$ and $h_2 = \tau g$. Then $\mathfrak{x}_1 \cong_G \mathfrak{x}_2$ if and only if $\mathfrak{x}_1 \cong_{Hh_j} \mathfrak{x}_2$ since $(K_1^\varphi)^{\Delta_1} \geq \mathrm{Alt}(\Delta_1)$. Moreover, if $G_j g_j = \mathrm{Iso}_{Hh_j}(\mathfrak{x}_1, \mathfrak{x}_2)$ then $\mathrm{Iso}_G(\mathfrak{x}_1, \mathfrak{x}_2) = \bigcup_{j=1,2} \langle K_1, G_j \rangle g_j$. Finally, $|G^\varphi : H^\varphi| \geq |\mathrm{Alt}(\Delta_1)| \geq (4/3)^k$. ∎

VII. String Isomorphism

With the adaption of Babai's techniques to our setting we overall obtain the following tool to build our recursive algorithm.

Lemma VII.1. *Let $G \leq \mathrm{Sym}(\Omega)$ be transitive and let $\mathfrak{x}, \mathfrak{y} \colon \Omega \to \Sigma$ be two strings. Moreover, suppose there is an almost d-ary sequence of G-invariant partitions $\{\Omega\} = \mathfrak{B}_0 \succ \cdots \succ \mathfrak{B}_m = \{\{\alpha\} \mid \alpha \in \Omega\}$ such that $|\mathfrak{B}_1| \leq d$. Then there are natural numbers $\ell \in \mathbb{N}$ and $n_1, \ldots, n_\ell \leq n/2$ such that $\sum_{i=1}^{\ell} n_i \leq 2^{\mathcal{O}((\log d)^3)} n$ and, for each $i \in [\ell]$ using a recursive call to String Isomorphism over domain size at most n_i and $d^{\mathcal{O}((\log d)^c)} n^c$ additional computation, one can compute $\mathrm{Iso}_G(\mathfrak{x}, \mathfrak{y})$.*

Combining the type of recursion described in the previous lemma with standard Luks reduction we obtain an algorithm solving string isomorphism in case the group is equipped with an almost d-ary sequence of invariant partitions.

Theorem VII.2. *Let $G \leq \mathrm{Sym}(\Omega)$ be a permutation group and let $\mathfrak{x}, \mathfrak{y} \colon \Omega \to \Sigma$ be two strings. Moreover, suppose there is an almost d-ary sequence of G-invariant partitions $\{\Omega\} = \mathfrak{B}_0 \succ \cdots \succ \mathfrak{B}_m = \{\{\alpha\} \mid \alpha \in \Omega\}$. Then one can compute $\mathrm{Iso}_G(\mathfrak{x}, \mathfrak{y})$ in time $n^{\mathcal{O}((\log d)^c)}$, for an absolute constant c.*

Proof: Consider the algorithm described in Figure 2. The algorithm essentially distinguishes between two cases. If the input group G is not transitive or the action of G on the block system $\mathfrak{B}_1$ is semi-regular, the algorithm follows

Input: $G \leq \mathrm{Sym}(\Omega)$ a $\widehat{\Gamma}_d$-group, $\mathfrak{x}, \mathfrak{y} \colon \Omega \to \Sigma$ two strings and an almost d-ary sequence of G-invariant partitions $\{\Omega\} = \mathfrak{B}_0 \succ \cdots \succ \mathfrak{B}_m = \{\{\alpha\} \mid \alpha \in \Omega\}$.
Output: $\mathrm{Iso}_G(\mathfrak{x}, \mathfrak{y})$

```
if G is not transitive then
    compute orbits Ω_1, ..., Ω_s
    recursively process group orbit by orbit
    return Iso_G(x, y)
else
    if G^{B_1} is semi-regular then
        apply standard Luks reduction
        return Iso_G(x, y)
    else
        apply Lemma VII.1
        return Iso_G(x, y)
    end if
end if
```

Figure 2. Algorithm for String Isomorphism

Luks algorithm recursively computing the set $\mathrm{Iso}_G(\mathfrak{x}, \mathfrak{y})$. In the other case G is transitive and $|\mathfrak{B}_1| \leq d$ and hence, we can apply Lemma VII.1 to recursively compute $\mathrm{Iso}_G(\mathfrak{x}, \mathfrak{y})$.

Clearly, it computes the desired set of isomorphisms. The bound on the running follows from Lemma II.4. Note that the bottleneck is the type of recursion used in Lemma VII.1. Also every group H, for which the algorithm performs a recursive call, is the projection of a subgroup of G to an invariant subset of the domain. Hence, by restricting the partitions $\mathfrak{B}_0, \ldots, \mathfrak{B}_m$ to the domain of H one obtains a sequence of partitions for the group H with the desired properties (cf. Observation IV.2). ∎

Combining Theorem IV.3 and Theorem VII.2 we obtain the main technical result of this work.

Theorem VII.3. *Let $G \leq \mathrm{Sym}(\Omega)$ be a $\widehat{\Gamma}_d$-group and let $\mathfrak{x}, \mathfrak{y} \colon \Omega \to \Sigma$ be two strings. Then there is an algorithm deciding whether $\mathfrak{x} \cong_G \mathfrak{y}$ in time $n^{\mathcal{O}((\log d)^c)}$, for an absolute constant c.*

Proof: Using orbit-by-orbit processing we can assume that the group G is transitive. For a transitive group the statement follows by first applying Theorem IV.3 and then Theorem VII.2. ∎

VIII. Applications

Using the improved algorithm for string isomorphism we can now prove the main result of this work using the following well-known reduction.

Theorem VIII.1 ([1], [3])**.** *There is a polynomial-time Turing-reduction from the Graph Isomorphism Problem for graphs of maximum degree d to the String Isomorphism Problem for $\widehat{\Gamma}_d$-groups (the running time of the reduction does not depend on d).*

The reduction follows [1] using an additional trick presented in [3, Section 4.2] to remove the dependence of the running time on d.

Combining this reduction with the improved algorithm for string isomorphism, we get the desired algorithm for isomorphism tests of bounded degree graphs.

Theorem VIII.2 (Theorem I.1 restated)**.** *The Graph Isomorphism Problem for graphs of maximum degree d can be solved in time $n^{\mathcal{O}((\log d)^c)}$, for an absolute constant c.*

Proof: This follows from Theorem VII.3 and VIII.1. ■

For a second application of Theorem VII.3 consider the isomorphism problem for relational structures.

Theorem VIII.3. *Let $\mathfrak{A} = (D, R)$, $\mathfrak{A}' = (D, R')$ be relational structures where $R, R' \subseteq D^t$ are t-ary relations. Then one can decide whether $\mathfrak{A}$ is isomorphic to $\mathfrak{A}'$ in time $n^{\mathcal{O}(t\cdot(\log n)^c)}$ where $n = |D|$.*

In many cases this leads to a better running time than first translating the structure into a graph and than applying Babai's algorithm to test whether the two resulting graphs are isomorphic. In particular, in case the arity t is large and also the size of the relation is large our method gives a much better worst case complexity than the other approach.

Also note that as a special case the same running time can be obtained for hypergraphs if t is the maximal hyperedge size. This also improves on previous results (see e.g. [27]).

Corollary VIII.4. *Let $\mathcal{H} = (V, \mathcal{E})$, $\mathcal{H}' = (V, \mathcal{E}')$ be two hypergraphs such that every hyperedge $E \in \mathcal{E}$ has size $|E| \leq t$. Then one can decide whether $\mathcal{H}$ is isomorphic to $\mathcal{H}'$ in time $n^{\mathcal{O}(t\cdot(\log n)^c)}$ where $n = |V|$.*

IX. Concluding Remarks

We have obtained a new graph isomorphism test with a running time bounded by a polynomial of degree polylogarithmic in the maximum degree of the input graphs. Technically, this result relies on some heavy group theory, new combinatorial tricks that allow us to reduce the string isomorphism problem for $\widehat{\Gamma}_d$ groups to a setting where we have an "almost d-ary" sequence of invariant partitions controlling the operation of the groups, and a refinement of the techniques introduced by Babai [10] for his quasipolynomial time isomorphism test.

We hope that the machinery we have developed here will have further applications and ultimately even lead to an improvement of Babai's isomorphism test. More immediate applications may be obtained for the isomorphism problem under restrictions of other parameters than the maximum degree. For example, we conjecture that there also is an isomorphism test running in time $n^{\mathcal{O}((\log k)^c)}$, where k is the tree width of the input graphs.

Another related problem that we leave open is whether the graph isomorphism problem parameterized by the maximum degree of the input graphs is fixed-parameter tractable.

References

[1] E. M. Luks, "Isomorphism of graphs of bounded valence can be tested in polynomial time," *Journal of Computer and System Sciences*, vol. 25, no. 1, pp. 42 – 65, 1982. [Online]. Available: http://www.sciencedirect.com/science/article/pii/0022000082900095

[2] L. Babai, W. M. Kantor, and E. M. Luks, "Computational complexity and the classification of finite simple groups," in *24th Annual Symposium on Foundations of Computer Science, Tucson, Arizona, USA, 7-9 November 1983*. IEEE Computer Society, 1983, pp. 162–171. [Online]. Available: https://doi.org/10.1109/SFCS.1983.10

[3] L. Babai and E. M. Luks, "Canonical labeling of graphs," in *Proceedings of the 15th Annual ACM Symposium on Theory of Computing, 25-27 April, 1983, Boston, Massachusetts, USA*, D. S. Johnson, R. Fagin, M. L. Fredman, D. Harel, R. M. Karp, N. A. Lynch, C. H. Papadimitriou, R. L. Rivest, W. L. Ruzzo, and J. I. Seiferas, Eds. ACM, 1983, pp. 171–183. [Online]. Available: http://doi.acm.org/10.1145/800061.808746

[4] M. Grohe and D. Marx, "Structure theorem and isomorphism test for graphs with excluded topological subgraphs," *SIAM J. Comput.*, vol. 44, no. 1, pp. 114–159, 2015. [Online]. Available: https://doi.org/10.1137/120892234

[5] S. Kratsch and P. Schweitzer, "Graph isomorphism for graph classes characterized by two forbidden induced subgraphs," *Discrete Applied Mathematics*, vol. 216, pp. 240–253, 2017. [Online]. Available: https://doi.org/10.1016/j.dam.2014.10.026

[6] E. M. Luks, "Permutation groups and polynomial-time computation," in *Groups And Computation*, ser. DIMACS Series in Discrete Mathematics and Theoretical Computer Science, vol. 11. DIMACS/AMS, 1991, pp. 139–176.

[7] I. N. Ponomarenko, "The isomorphism problem for classes of graphs closed under contraction," *Journal of Soviet Mathematics*, vol. 55, no. 2, pp. 1621–1643, Jun 1991. [Online]. Available: https://doi.org/10.1007/BF01098279

[8] Á. Seress, *Permutation Group Algorithms*, ser. Cambridge Tracts in Mathematics. Cambridge University Press, 2003. [Online]. Available: https://books.google.de/books?id=hxFqdbfc_CMC

[9] L. Babai, "Graph isomorphism in quasipolynomial time," *CoRR*, vol. abs/1512.03547v2, 2015. [Online]. Available: http://arxiv.org/abs/1512.03547

[10] ——, "Graph isomorphism in quasipolynomial time [extended abstract]," in *Proceedings of the 48th Annual ACM SIGACT Symposium on Theory of Computing, STOC 2016, Cambridge, MA, USA, June 18-21, 2016*, D. Wichs and Y. Mansour, Eds. ACM, 2016, pp. 684–697. [Online]. Available: http://doi.acm.org/10.1145/2897518.2897542

[11] L. Babai, P. Cameron, and P. Plfy, "On the orders of primitive groups with restricted nonabelian composition factors," *Journal of Algebra*, vol. 79, no. 1, pp. 161 – 168, 1982. [Online]. Available: http://www.sciencedirect.com/science/article/pii/0021869382903234

[12] M. Grohe, D. Neuen, P. Schweitzer, and D. Wiebking, "An improved isomorphism test for bounded-tree-width graphs," in *45th International Colloquium on Automata, Languages, and Programming, ICALP 2018, July 9-13, 2018, Prague, Czech Republic*, ser. LIPIcs, I. Chatzigiannakis, C. Kaklamanis, D. Marx, and D. Sannella, Eds., vol. 107. Schloss Dagstuhl - Leibniz-Zentrum fuer Informatik, 2018, pp. 67:1–67:14. [Online]. Available: https://doi.org/10.4230/LIPIcs.ICALP.2018.67

[13] D. Lokshtanov, M. Pilipczuk, M. Pilipczuk, and S. Saurabh, "Fixed-parameter tractable canonization and isomorphism test for graphs of bounded treewidth," in *55th IEEE Annual Symposium on Foundations of Computer Science, FOCS 2014, Philadelphia, PA, USA, October 18-21, 2014*. IEEE Computer Society, 2014, pp. 186–195. [Online]. Available: https://doi.org/10.1109/FOCS.2014.28

[14] J. Rotman, *An Introduction to the Theory of Groups*, ser. Graduate Texts in Mathematics. Springer New York, 1999. [Online]. Available: https://books.google.de/books?id=lYrsiaHSHKcC

[15] J. D. Dixon and B. Mortimer, *Permutation groups*, ser. Graduate Texts in Mathematics. Springer-Verlag, New York, 1996, vol. 163. [Online]. Available: http://dx.doi.org/10.1007/978-1-4612-0731-3

[16] M. W. Liebeck and A. Shalev, "Simple groups, permutation groups, and probability," *J. Amer. Math. Soc.*, vol. 12, no. 2, pp. 497–520, 1999. [Online]. Available: http://dx.doi.org/10.1090/S0894-0347-99-00288-X

[17] ——, "Bases of primitive linear groups," *Journal of Algebra*, vol. 252, no. 1, pp. 95 – 113, 2002. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0021869302000017

[18] ——, "Bases of primitive linear groups II," *Journal of Algebra*, vol. 403, pp. 223 – 228, 2014. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0021869314000490

[19] B. N. Cooperstein, "Minimal degree for a permutation representation of a classical group," *Israel Journal of Mathematics*, vol. 30, no. 3, pp. 213–235, Sep 1978. [Online]. Available: https://doi.org/10.1007/BF02761072

[20] P. Kleidman and M. Liebeck, *The subgroup structure of the finite classical groups*, ser. London Mathematical Society Lecture Note Series. Cambridge University Press, Cambridge, 1990, vol. 129. [Online]. Available: http://dx.doi.org/10.1017/CBO9780511629235

[21] D. Gluck, kos Seress, and A. Shalev, "Bases for primitive permutation groups and a conjecture of Babai," *Journal of Algebra*, vol. 199, no. 2, pp. 367 – 378, 1998. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0021869397971490

[22] M. W. Liebeck, "On minimal degrees and base sizes of primitive permutation-groups," *Archiv der Mathematik*, vol. 43, pp. 11–15, 1984. [Online]. Available: http://dx.doi.org/10.1007/BF01193603

[23] P. J. Cameron, "Finite permutation groups and finite simple groups," *Bulletin of the London Mathematical Society*, vol. 13, no. 1, pp. 1–22, 1981. [Online]. Available: http://dx.doi.org/10.1112/blms/13.1.1

[24] A. Marti, "On the orders of primitive groups," *Journal of Algebra*, vol. 258, no. 2, pp. 631 – 640, 2002. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0021869302006464

[25] L. Babai, E. M. Luks, and Á. Seress, "Permutation groups in NC," in *Proceedings of the 19th Annual ACM Symposium on Theory of Computing, 1987, New York, New York, USA*, A. V. Aho, Ed. ACM, 1987, pp. 409–420. [Online]. Available: http://doi.acm.org/10.1145/28395.28439

[26] U. Meierfrankenfeld, *Non-Finitary Locally Finite Simple Groups*. Dordrecht: Springer Netherlands, 1995, pp. 189–212. [Online]. Available: https://doi.org/10.1007/978-94-011-0329-9_7

[27] L. Babai and P. Codenotti, "Isomorhism of hypergraphs of low rank in moderately exponential time," in *49th Annual IEEE Symposium on Foundations of Computer Science, FOCS 2008, October 25-28, 2008, Philadelphia, PA, USA*. IEEE Computer Society, 2008, pp. 667–676. [Online]. Available: https://doi.org/10.1109/FOCS.2008.80