THE UNIVERSITY OF CHICAGO

GROUP-THEORETIC ASPECTS OF COMPLEXITY THEORY AND CODING THEORY

A DISSERTATION SUBMITTED TO
THE FACULTY OF THE DIVISION OF THE PHYSICAL SCIENCES
IN CANDIDACY FOR THE DEGREE OF
DOCTOR OF PHILOSOPHY

DEPARTMENT OF MATHEMATICS
AND
DEPARTMENT OF COMPUTER SCIENCE

BY
TIMOTHY JAMES FORNELL BLACK

CHICAGO, ILLINOIS
DECEMBER 2019

To my parents.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

ACKNOWLEDGMENTS

Thank you to my advisor, Laci Babai, for introducing me to many new ideas, and for the many hours he spent teaching me about math, computer science, and writing. Thank you to Sasha Razborov for being on my committee, and for his encouragement, especially while working on evasiveness. Thank you to Madhur Tulsiani for being on my committee and for discussions. Thank you to Madhu Sudan for introducing me to the area of list decoding homomorphism codes. Thank you to Angela Wuu for collaborating on the homomorphism codes project, and making the project as fun as it was fruitful. Thank you to my friends in Chicago and beyond, who have been an anchor. Thank you to my parents for encouraging my curiosity and always being supportive.

ABSTRACT

Group theory has long played a role in complexity theory. We explore two of these connections — to decision-tree complexity, and to local list-decoding.

A boolean function in n variables is *weakly evasive* if its decision-tree complexity is $\Omega(n)$. By k-graphs we mean k-uniform hypergraphs. A *k-graph property* on v vertices is a boolean function on $n = \binom{v}{k}$ variables corresponding to the k-subsets of a v-set that is invariant under the $v!$ permutations of the v-set (isomorphisms of k-graphs).

Rivest and Vuillemin (1976) proved that all non-constant monotone graph properties ($k = 2$) are weakly evasive, confirming a conjecture of Aanderaa and Rosenberg (1973). Kulkarni, Qiao, and Sun (2013) proved the analogous result for 3-graphs. We extend these results to k-graphs for every fixed k. From this, we show that monotone boolean functions invariant under the action of a large primitive group are weakly evasive.

While KQS (2013) employ the powerful topological approach of Kahn, Saks, and Sturtevant (1984) combined with heavy number theory, our argument is elementary and self-contained (modulo some basic group theory). Inspired by the outline of the RV and KQS approaches, we formalize the general framework of "orbit augmentation sequences" of sets with group actions. We show that a parameter of such sequences, which we call the "spacing," is a lower bound on the decision-tree complexity for any nontrivial monotone property that is Γ-invariant for all groups Γ involved in the orbit augmentation sequence, assuming all those groups are p-groups. We develop operations on such sequences such as composition and direct product which will provide helpful machinery for our applications. We apply this general technique to k-graphs via certain liftings of k-graphs with wreath product action of p-groups.

The codewords of the *homomorphism code* $\mathrm{aHom}(G, H)$ are the affine homomorphisms between two finite groups, G and H, generalizing Hadamard codes. Following the work of

Goldreich–Levin (1989), Grigorescu et al. (2006), Dinur et al. (2008), and Guo and Sudan (2014), we further expand the range of groups for which local list-decoding is possible up to $\mathsf{mindist}$, the minimum distance of the code. In particular, for the first time, we **do not require either G or H to be solvable**. Specifically, we demonstrate a $\mathrm{poly}(1/\varepsilon)$ bound on the list size, i. e., on the number of codewords within distance $(\mathsf{mindist} - \varepsilon)$ from any received word, when G is either abelian or an alternating group, and H **is an arbitrary (finite or infinite) group**. We conjecture that a similar bound holds for all finite simple groups as domains; the alternating groups serve as the first test case.

The abelian vs. arbitrary result permits us to adapt previous techniques to obtain efficient local list-decoding for this case. In the alternating vs. arbitrary setting, we obtain a local algorithm that produces partial homomorphisms that uniquely extend to the homomorphisms in the list. We introduce this "semi-algorithmic" model, which we call **Certificate List-Decoding**, due to the severe technical difficulties in extending partial homomorphisms to full homomorphisms, even when the extension is unique. A homomorphism extender applied to a list of certificates yields the desired list. Wuu (2018) studied the Homomorphism Extension Problem and solved it for a certain class of pairs of groups. Her result combined with ours gives efficient algorithms when G is an alternating group of degree n, and H is a symmetric group of degree m, for $m < 2^{n-1}/\sqrt{n}$.

The results on evasiveness and homomorphism codes with abelian domain are solo work. All other results on homomorphism codes are joint work with László Babai and Angela Wuu.

CHAPTER 1

INTRODUCTION

1.1 Overview

From the earliest days of computation theory, connections to group theory have played an important role. For instance, the undecidability of questions in group theory have had a major impact on topology, where the fundamental group can be expressed in terms of generators and relations.

In contrast to the computation theory of the early twentieth century where the emphasis was on computability, our emphasis is on efficient computability, where the P versus NP problem serves as a beacon. In algorithmic group theory, complexity of algorithms for classes of represented groups such as permutation groups, matrix groups, and black-box groups have been studied separately from decidability questions.

Efficient computability has been studied in a number of models. In this dissertation, we look at two models: the decision-tree model, and the local computation model. These are both query complexity models — efficiency is measured partially or entirely by the number of queries that must be made to the input. After this, the models diverge.

In the decision-tree model (and specifically, in the study of evasiveness), we concern ourselves with functions that are potentially very sensitive to changes in the input, and where a large number of queries to the input may be needed to evaluate the function. We give an overview and introduce our results on decision-tree complexity and evasiveness in Section 1.3.

In the local computation model, the goal is to use an extremely small number of queries to the input (relative to the size of the input) to get probabilistic answers. We are specifically interested in local algorithms to list-decode error-correcting codes based on group homomorphisms. We discuss the history of this area and introduce our results in Section 1.4.

1.2 Joint work and overlap with Wuu's dissertation

The evasiveness material (Section 1.3 and Chapter 2) is adapted from the solo paper [10].

The homomorphism codes material (Section 1.4 and Chapter 3) is an expanded version of the conference paper [6], which is a joint paper with László Babai and Angela Wuu. However, the {abelian $\to$ arbitrary} result is my solo contribution to that paper; its journal version will be published as a solo paper. The other homomorphism codes results will appear in a joint journal paper with Babai and Wuu. The bulk of the proof of the {abelian $\to$ arbitrary} case appears in Section 3.6.

There is necessarily an overlap between this dissertation and Wuu's dissertation, [36]. Table 1.1 lists the major subjects covered in the two dissertations, and shows the overlap in topics discussed in detail between the two dissertations.

	This dissertation	Wuu's dissertation
Evasiveness	✓	
Hom codes: abelian domain	✓	
Hom codes: alternating domain	✓	✓
Hom codes: misc. results	✓	✓
Homomorphism Extension		✓

Table 1.1: Comparison between this dissertation and [36].

1.3 Evasive properties

1.3.1 History and main results

The study of evasiveness was initiated by a conjecture by Aanderaa and Rosenberg [31] about graph properties; see below.

Suppose we are given a property of subsets of a finite set X, and we are trying to

determine whether a particular subset S of X which is hidden from us has the property by asking membership queries (questions of the form "$x \in S$?"). The maximum number of queries we need to ask over all choices of S, assuming we are using a strategy that minimizes this maximum, is called the *decision-tree complexity* of the property. A property over X is said to be *evasive* if its decision-tree complexity is $|X|$. A collection of properties is said to be *weakly evasive* if each property has decision-tree complexity within a constant factor of the size of its underlying set.

Particular attention has been given to properties that have an amount of symmetry. Rivest and Vuillemin [29] proved that nontrivial monotone properties invariant under the action of a transitive p-group are evasive. Kahn, Saks, and Sturtevant conjectured [24] that nontrivial monotone properties that are invariant under any transitive group action are evasive. The conjecture remains open, although there has been progress on weakened versions of it.

In Chapter 2, we prove that the conjecture holds for large primitive groups, if evasiveness is replaced by weak evasiveness. (See Section 2.1 for the definition of primitive groups.)

Theorem 1.3.1. *For each fixed $\varepsilon > 0$, nontrivial monotone properties over $[n]$ that are invariant under the action of a primitive group of order at least $\exp(n^{\varepsilon})$ are weakly evasive.*

Typical examples of large primitive groups are the induced action of the symmetric group $\mathrm{Sym}(V)$ on $\binom{V}{k}$, the set of k-element subsets of V. Symmetry under this action defines k-uniform hypergraph (k-graph) properties.

Corollary 1.3.2. *For every fixed k, nontrivial monotone k-graph properties are weakly evasive.*

Our estimate for the constant hidden in the definition of weak evasiveness is $\exp(-O(k^2))$ (see Theorem 2.5.3 for a more specific statement).

Interest in weak evasiveness beyond graph properties was recently rekindled by Kulkarni [26]. Corollary 1.3.2 generalizes a recent result of Kulkarni, Qiao, and Sun [27], who proved

it for $k = 3$. Their paper motivated our work on this problem.

The history of this problem goes much further back. The $k = 2$ case is a classical result of Rivest and Vuillemin [29], confirming a 1973 conjecture of Aanderaa and Rosenberg [31] that started this entire subject. Rosenberg [31] credits Karp with the following stronger version of this conjecture.

Conjecture 1.3.3 (Aanderaa–Rosenberg–Karp Conjecture)**.** *Nontrivial monotone graph properties are evasive.*

This conjecture remains open, and has motivated much of the work in this area.

A major result in this direction was obtained by Kahn, Saks, and Sturtevant [24], who introduced a powerful topological technique in the study of evasiveness and confirmed Conjecture 1.3.3 when the number of vertices is a prime-power.

Yao [38], using the KSS method, proved that nontrivial monotone bipartite graph properties are evasive. KQS [27] generalized this to k-partite k-graphs, weakening the conclusion to weak evasiveness. We further generalize this to partwise-uniform hypergraphs.

Definition 1.3.4. For positive integers k, t, $k_1, \ldots, k_t$ such that $k_1 + \cdots + k_t = k$, we say that a k-graph is *t-partite and $(k_1, \ldots, k_t)$-uniform* if its vertices are partitioned into t parts, and each edge has k_i vertices in part i. When $k_1, \ldots, k_t$ are clear from context, we speak of *partwise-uniform* k-graphs.

We shall use Corollary 1.3.2 to derive the following generalization.

Corollary 1.3.5. *For bounded k, and a partition $k = k_1 + \cdots + k_t$, nontrivial monotone properties of t-partite $(k_1, \ldots, k_t)$-uniform k-graphs are weakly evasive.*

Corollary 1.3.2 is the special case where $t = 1$. The KQS result mentioned is the $k_1 = \cdots = k_t = 1$ case of Corollary 1.3.5.

1.3.2 Structure of the chapter

We discuss evasiveness in Chapter 2.

Our main technical result is Corollary 1.3.2, which we prove in Sections 2.2–2.5. In Section 2.6 we derive Corollary 1.3.5 from Corollary 1.3.2 and use Corollary 1.3.5 along with a result of Cameron [12] on large primitive groups to prove Theorem 1.3.1.

We explain the structure of the chapter in greater detail in Section 1.3.4.

1.3.3 Previous methods

A p-group is a group of order a power of the prime p. The connection of the evasiveness problem to p-groups was discovered by Rivest and Vuillemin in their 1976 paper [29] that proved the Aanderaa–Rosenberg Conjecture. They showed that if a property $\mathscr{P}$ over a set X is invariant under a transitive p-group action and $\mathscr{P}$ holds for exactly one of $\emptyset$ and X, then $\mathscr{P}$ is evasive (see Theorem 2.2.1 below). They used this result via *subproperties* to obtain weak evasiveness of properties where the size of the underlying set was not a prime power. (A "subproperty" is obtained from a property over the set X by fixing the membership status of some elements of the underlying set; this corresponds to the notion of *restriction* of boolean functions.)

This basic pattern has been maintained in virtually all subsequent work on the subject, including the seminal 1984 paper by Kahn, Saks, and Sturtevant [24] where it is used through a lemma from [25].

KSS discovered that if a monotone property, viewed as a simplicial complex, is not contractible then the property is evasive. To use this criterion, KSS and all subsequent work relied on powerful results by Oliver [28] on the fixed-point complexes of certain finite groups acting on the complex. Oliver's results apply to a very special class of groups, called "Oliver groups."

Definition 1.3.6. A finite group Γ is an **Oliver group** if it has subgroups Γ_1 and Γ_2 such that $\Gamma_1 \trianglelefteq \Gamma_2$ is a normal subgroup, $\Gamma_2 \trianglelefteq \Gamma$ is a normal subgroup, $|\Gamma/\Gamma_2|$ is a power of a prime, Γ_2/Γ_1 is cyclic, and $|\Gamma_1|$ is a power of a (possibly different) prime.

Note in particular that *all Oliver groups are solvable.* As KSS point out, Oliver's results define the limits of their approach. Indeed it is the very strict constraints on the structure of Oliver groups that makes it unlikely that the method used by Kulkarni, Qiao, and Sun [27] to prove weak evasiveness of nontrivial monotone 3-graph properties could be extended to k-graph properties for $k \geq 4$.

A permutation group is *transitive* if all elements of the permutation domain are equivalent under the group action. A permutation group is *t-transitive* if the action is transitive on the set of ordered t-tuples of distinct elements of the permutation domain.

The KQS approach requires the construction of doubly transitive (2-transitive) Oliver groups acting on a large and well-controlled number of points. Such groups do exist on domains of size a prime power (the groups of affine transformations $x \mapsto ax + b$ over a finite field) and have already been used by KSS to prove that nontrivial monotone graph properties for a prime-power number of vertices are evasive. For proper control of the numbers on which such groups act, KQS invoke a uniform version of Vinogradov's celebrated 3-Primes Theorem [35] (that every sufficiently large odd integer is a sum of three primes that are nearly equal). However, an extension of the KQS approach to k-graphs would require $(k-1)$-transitive Oliver groups; but it is known that there is only a finite number of very small 3-transitive solvable groups; only S_4 is 4-transitive and solvable; and there are no 5-transitive solvable groups. (The last statement is an immediate consequence of the fact that the symmetric group S_5 is not solvable.)

Another concern regarding the use of the topological approach via Oliver's results is that Oliver groups don't combine well. While virtually any reasonable operation on groups (direct product, semidirect product, wreath product, group extension, etc.), when applied to p-groups (for the same prime p) yields a p-group, even a direct product of two Oliver groups is not an Oliver group in general. Of course every p-group is an Oliver group, but to the extent that the class of Oliver groups is richer than just the p-groups, we find this added variety more an obstacle than help in proving weak evasiveness.

Since the KSS paper, virtually all articles in this area have used the topological approach via Oliver's results. The papers [38, 34, 17, 14, 4, 27] are but a small sample. In the light of the concerns mentioned, it seems appropriate that we break this three-decades-old tradition.

1.3.4 Our methods

Rather than using the topological approach, we devise a combination of combinatorial and group theoretic techniques to carry out what on a high level might be called the Rivest–Vuillemin plan: use p-groups through carefully designed subproperties.

Inspired by the outline of the KQS approach (itself inspired by the Rivest–Vuillemin plan), we define a general framework which we call "orbit augmentation sequences" (OA sequences) of sets with group actions. Such a sequence is defined as a sequence of pairs (B_i, A_i) of disjoint subsets of the underlying set X with a sequence Γ_i of subgroups of a group Γ that acts on X such that Γ_i fixes the sets B_i and A_i (setwise) and acts transitively on A_i (so A_i is an orbit of Γ_i) and $B_{i-1} \cup A_{i-1} \supseteq B_i$, hence the name "orbit augmentation sequence." (See Definition 2.2.4.) The minimum size of the A_i is the *spacing* of the sequence. We consider orbit augmentation sequences that start with $B_1 = \emptyset$ and end with $B_m \cup A_m = X$; we call these "full OA sequences."

Our basic observation is that *the spacing of a full OA sequence is a lower bound on the decision-tree complexity* of any nontrivial monotone Γ-invariant property (Proposition 2.2.5), assuming each subgroup involved is a p-group (or more generally, an Oliver group, but we only use p-groups). We develop the theory of orbit augmentation sequences in Section 2.2 where we introduce operations such as composition and direct product that will give us a helpful machinery for our applications. As indicated, Oliver groups are an obstacle to direct product operations; but the machinery works very well when we limit the groups involved to p-groups.

The bulk of Chapter 2 (Sections 2.3–2.5) shows how to implement OA sequences for k-graph properties. One of our combinatorial tools will be a lifting of k-graphs (a special

case of their lexicographic product) with wreath product action of the groups involved; these concepts are discussed in Section 2.3.

Section 2.4 implements such sequences in the context of k-graphs with a prime-power number of vertices. Like KQS, we partition the vertices into equally-sized sets, and classify edges by how many vertices they have in each set. We then build a sequence by adding each class of edges one at a time. KQS added each class in a single step of the sequence, which required more complicated groups, and thus limited the types of edges that could be added. We add each class through an OA sequence. We add the edges that have all vertices in a single set of the partition by inducting on the number of vertices, building up to the next level by taking a lifting of a smaller complete graph. Each other class of edges can be thought of as a product of complete k_i-graphs with $k_i < k$; we add such a class using induction on k and our direct product lemma. Our composition lemma allows us to do each of these additions of classes consecutively in order to build up a full OA sequence for our original k-graph.

In Section 2.5 we extend the result of Section 2.4, constructing orbit augmentation sequences on an arbitrary number of vertices. We build off of the sequences from Section 2.4, once again constructing them inductively using direct products and composition.

The final section, Section 2.6, uses the result of Section 2.5 to prove the main theorem, that nontrivial monotone properties invariant under the action of a large primitive group are weakly evasive. A result of Cameron says that all large permutation groups have a subgroup that looks like the group of symmetries of a hypergraph, raised to some power. We apply the result of Section 2.5, along with our direct product lemma, to this subgroup.

1.4 List Decoding Homomorphism Codes

1.4.1 Brief history

Let G and H be finite groups, to be referred to as the *domain* and the *codomain*, respectively. A map $\psi\colon G \to H$ is an **affine homomorphism** if

$$(\forall a, b, c \in G)(\psi(a)\psi(b)^{-1}\psi(c) = \psi(ab^{-1}c))\,. \tag{1.1}$$

Equivalently, ψ is a translate of a homomorphism, i. e., there exists a homomorphism $\varphi\colon G \to H$ and an element $h \in H$ such that $(\forall g \in G)(\psi(g) = \varphi(g) \cdot h)$. We write $\mathrm{Hom}(G, H)$ and $\mathrm{aHom}(G, H)$ to denote the set of homomorphisms and affine homomorphisms, respectively. Let H^G denote the set of all functions $f\colon G \to H$. We represent an (affine) homomorphism ψ by the list of pairs $(g, \psi(g))$ for $g \in S$ where S is a set of (affine) generators of G.

We view $\mathrm{aHom}(G, H)$ as a (nonlinear) code within the code space H^G (the space of possible "received words") and refer to this class of codes as *homomorphism codes*.

Homomorphism codes are candidates for efficient *local* list-decoding *up to minimum distance* (mindist) and in many cases it is known that their minimum distance is (asymptotically) equal to the list-decoding bound.

This line of work goes back to the celebrated paper by Goldreich and Levin (1989) [18] who found local list-decoders for Hadamard codes, i. e., for homomorphism codes with domain $G = \mathbb{Z}_2^n$ and codomain $H = \mathbb{Z}_2$. This result was extended to homomorphism codes of abelian groups (both the domain and the codomain abelian) by Grigorescu, Kopparty, and Sudan (2006) [19] and Dinur, Grigorescu, Kopparty, and Sudan (2008) [15] and to the case of supersolvable domain and nilpotent codomain by Guo and Sudan (2014) [21], cf. [11].

While homomorphism codes have low (logarithmic or worse) rates, they tend to have remarkable list-decoding properties. In particular, in all cases studied so far (including the present work), for an *arbitrary* received word $f \in H^G$, and any $\varepsilon > 0$, the number of

codewords within radius ($\mathsf{mindist} - \varepsilon$) is bounded by $\mathrm{poly}(1/\varepsilon)$ (as opposed to some faster-growing function of ε, as permitted in the theory of list-decoding). This is an essential feature for the complexity-theoretic application (hard-core predicates) by Goldreich and Levin. Let $\mathcal{L}$ denote the list of codewords within distance ($\mathsf{mindist} - \varepsilon$) of the received word.

We call an $|\mathcal{L}| \leq \mathrm{poly}(1/\varepsilon)$ bound *economical*, and a class of homomorphism codes permitting such a bound **combinatorially economically list-decodable (CombEcon)**. We say the class is CombEcon *of degree* c if the polynomial has degree c. (With some abuse of the language, we shall talk about "a CombEcon code" in reference to members of a CombEcon class of codes defined by the context. We apply the analogous convention to other asymptotic properties to be defined below for classes of codes.)

By *efficient* list-decoding we mean performing $\mathrm{poly}(\log|G|, 1/\varepsilon)$ randomized queries to the received word and performing $\mathrm{poly}(\log|G|, \log|H|, 1/\varepsilon)$ additional work to produce a list of $\leq \mathrm{poly}(1/\varepsilon)$ affine homomorphisms that includes all affine homomorphisms within ($\mathsf{mindist} - \varepsilon$) of the received word.

We call a CombEcon code **AlgEcon (algorithmically economically list-decodable)** if it permits efficient decoding within radius ($\mathsf{mindist} - \varepsilon$) in this sense. So the cited results show that homomorphism codes with abelian domain and codomain, and more generally with supersolvable domain and nilpotent codomain, are CombEcon and AlgEcon.

In all work on the subject, this efficiency depends on the computational representation of the groups used (presentation in terms of generators and relators, black-box access, black-box groups, permutation groups, matrix groups, etc.). We shall make the representation required explicit in all algorithmic results.

1.4.2 Our contributions

Homomorphism codes are the subject of Chapter 3.

Combinatorial bounds

In Chapter 3 we further expand the range of groups for which efficient local list-decoding is possible up to the minimum distance. In particular, for the first time, we **do not require either G or H to be solvable**. In fact, in our combinatorial and semi-algorithmic results (see below), **the codomain is an arbitrary (finite or infinite) group**. We say that a class $\mathfrak{G}$ of finite groups is **universally CombEcon** if for all $G \in \mathfrak{G}$ and arbitrary (finite or infinite) H, the code $\mathrm{aHom}(G, H)$ is CombEcon. This effort is the first to demonstrate the existence of significant universally CombEcon classes.

Convention 1.4.1. When speaking of a homomorphism code $\mathrm{aHom}(G, H)$, the domain G will always be a finite group, but the codomain H will, in general, not be restricted to be finite.

Theorem 1.4.2 (Main combinatorial result)**.** *Finite abelian and alternating groups are universally CombEcon. For abelian groups, the degree is* ≈ 109*, and for alternating groups, the degree is* 8*.*

We explain this result in detail. By *distance* we mean normalized Hamming distance.

(Restatement of Theorem 1.4.2.) *Let the domain G be a finite abelian or alternating group and H an arbitrary (finite or infinite) group. Let* $\mathsf{mindist}$ *denote the minimum distance of the homomorphism code* $\mathrm{aHom}(G, H)$ *and let $\varepsilon > 0$. Let $f \in H^G$ be an arbitrary received word. Then the number of codewords within* $(\mathsf{mindist} - \varepsilon)$ *of f is at most* $\mathrm{poly}(1/\varepsilon)$*, where the degree of this polynomial is* ≈ 109 *when G is abelian, and* 7 *when G is alternating.*

Remark 1.4.3. We give two proofs of the alternating result. The first proof (see Section 3.7) is nonconstructive and is based on a broadly applicable sphere-packing argument (see Sec. 3.5.3). The second proof (see Section 3.8) is more closely based on the structure of the alternating groups and depends on a result about random generation with extremely high probability (see Theorem 3.2.20). This approach yields a very simple semi-algorithmic

result (certificate list-decoding, see Sec. 3.1.5) and leads, using deeper tools [37], to our main algorithmic result, Theorem 1.4.11.

For abelian domains (see Section 3.6) we prove a bound of $O(\varepsilon^{-C-4})$ on the length of the list, i. e., the number of codewords within $(\mathsf{mindist} - \varepsilon)$ of the received word where $O(\varepsilon^{-C})$ is the corresponding {abelian $\to$ abelian} bound. Currently $C \approx 105$ [21]. For alternating domains we prove a bound of $O(\varepsilon^{-7})$ on the length of the list.

Our choice of the alternating groups as the domain is our test case of what we believe is a general phenomenon valid for all finite simple groups.

Conjecture 1.4.4. *The class of finite simple groups is universally CombEcon.*

The following problem is also open.

Problem 1.4.5. Is the class of all finite groups universally CombEcon?

Definition 1.4.6. Let us say that the **depth** of a subgroup K in a group G is the length ℓ of the longest subgroup chain $K = K_0 < K_1 < \cdots < K_\ell = G$. We denote this quantity by $\mathrm{depth}_G(K)$. We say that a subgroup is **shallow** if it has bounded depth.

Theorem 1.4.2 also holds for a hierarchy of wider classes of finite groups we call *shallow random generation* groups or "SRG groups." This hierarchy includes the class of alternating groups. The defining feature of SRG groups is that a bounded number of random elements generate, with extremely high probability, a shallow subgroup.

Our combinatorial tools allow us to play on the relatively well-understood top layers of the subgroup lattice of the (alternating or SRG) domain, avoiding the dependence on the codomain in the combinatorial and semi-algorithmic context.

Remark 1.4.7. Our results list-decode certain classes of codes up to distance $(\mathsf{mindist} - \varepsilon)$ for positive ε. In many cases, $\mathsf{mindist}$ is the list-decoding boundary; examples show that the length of the list may blow up when ε is set to zero. Classes of such examples with abelian domain and codomain were found by Guo and Sudan [21]. We add classes of examples with alternating domains (see Section 3.7.3).

Algorithms and certificate list-decoding

Abelian domains. On the algorithmic front, the combinatorial bound in the {abelian → arbitrary} case permits us to adapt the algorithm of [19] to obtain efficient local list-decoding. We say that a class $\mathfrak{G}$ of finite groups is **universally AlgEcon** if for all $G \in \mathfrak{G}$ and arbitrary finite H, the code $\mathrm{aHom}(G, H)$ is AlgEcon. The validity of such a statement depends not only on the class $\mathfrak{G}$ but also on the representation of the domain and the codomain.

Corollary 1.4.8. *Let G be a finite abelian group and H an arbitrary finite group. Under suitable assumptions on the representation of G and H, the homomorphism code* $\mathrm{aHom}(G, H)$ *is AlgEcon.*

In other words, abelian groups are *universally AlgEcon.*

We need to clarify the "suitable representation." It suffices to have G in its primary decomposition and to have black-box access to H. These concepts, and other options for G, are discussed in Section 3.2.4.

Alternating domains. By a $G \rightharpoonup H$ *partial map* we mean a function $\gamma\colon \mathrm{dom}(\gamma) \to H$ where $\mathrm{dom}(\gamma) \subseteq G$. Algorithmically, we describe a homomorphism φ by its values on a set of generators, that is, by a $G \rightharpoonup H$ partial map γ with $\langle \mathrm{dom}(\gamma) \rangle = G$. In all previous work, and in the case of {abelian → arbitrary} homomorphism codes, the minimum distance corresponded to a subgroup of smallest index in the group G/N where N is the "irrelevant kernel," i. e., the intersection of the kernels of all $G \to H$ homomorphisms (see Sec. 3.3.2). However, this does not always hold in the case of homomorphism codes with alternating domain. Thus, a $G \rightharpoonup H$ partial map γ may be a *certificate* for a homomorphism $\varphi \in \mathrm{Hom}(G, H)$ — that is, φ may be the unique homomorphism that extends γ — even if $\mathrm{dom}(\gamma)$ does not generate G. Moreover, given a certificate γ for a homomorphism φ, there may be not be an efficient algorithm to find the φ on a full set of generators.

The **homomorphism extension problem** HomExt asks whether a $G \rightharpoonup H$ partial map extends to a $G \to H$ homomorphism. The HomExt *search problem* asks to find said

homomorphism. This problem is of interest in its own right, which remains the principal bottleneck for algorithmic progress.

To bypass the HomExt bottleneck, we introduce a new model we call **Certificate List-Decoding**. In this model the output is a short (poly$(1/\varepsilon)$-length) list of $G \to H$ partial maps that includes, for each affine homomorphism φ within $(\mathsf{mindist} - \varepsilon)$ of the received word, a *certificate* of φ, i. e., a partial affine homomorphism that uniquely extends to φ.

We say that a homomorphism code is **economically certificate-list-decodable (CertEcon)** if such a list can be efficiently generated.

Note that, by definition, AlgEcon $\implies$ CertEcon $\implies$ CombEcon.

We say that a class $\mathfrak{G}$ of finite groups is **universally CertEcon** if for all $G \in \mathfrak{G}$ and arbitrary (finite or infinite) H, the code $\mathrm{aHom}(G, H)$ is CertEcon.

Theorem 1.4.9 (Main semi-algorithmic result)**.** *Alternating groups are universally CertEcon.*

In fact we show that SRG groups are universally CertEcon.

For a received word f and an affine homomorphism φ within distance $\mathsf{mindist} - \varepsilon$ of f, a **domain certificate** of φ is a subset S of the domain such that f restricted to S is a certificate of φ. Our semi-algorithmic results will actually produce lists of domain certificates without requiring any access to the codomain.

By the *density* of a partial map γ we mean the density of the subgroup $\langle \mathrm{dom}(\gamma) \rangle$ in G. A λ-certificate list-decoder produces partial maps of density $\geq \lambda$. Our semi-algorithmic results produce $(1 - \mathsf{mindist})$-certificate list-decoders.

The $\textsc{HomExt}_\lambda$ problem asks to solve HomExt for partial maps of density $\geq \lambda$. It is immediate that a λ-certificate list-decoder, combined with a $\textsc{HomExt}_\lambda$ solver, suffices for list-decoding $\mathrm{aHom}(G, H)$. Wuu [37] solves the $\textsc{HomExt}_\lambda$ search problem in the following case.

Theorem 1.4.10 (Wuu)**.** *Let $G = A_n$, $H = S_m$ and $\lambda = 1/\operatorname{poly}(n)$. If $m < 2^{n-1}/\sqrt{n}$, then the $\textsc{HomExt}_\lambda(G, H)$ search problem can be solved in $\operatorname{poly}(n, m)$ time.*

Our $(1-\mathsf{mindist})$-certificate list-decoder combines with this solution to $\textsc{HomExt}_{1-\mathsf{mindist}}$ to give a list decoder, Theorem 1.4.11. This is the main algorithmic result of [6]. A *permutation representation of degree m* of a group G is a homomorphism $G \to S_m$, where the codomain is the symmetric group of degree m. We obtain efficient local list-decoding for the permutation representations of alternating groups under a rather generous restriction on the size of the permutation domain.

Theorem 1.4.11 (Main algorithmic result)**.** *Let $G = A_n$ be the alternating group of degree n and $H = S_m$ the symmetric group of degree m. Then* $\mathrm{aHom}(G, H)$ *is AlgEcon, assuming $m < 2^{n-1}/\sqrt{n}$.*

For details of Theorem 1.4.10, see [37, 36].

Mean-list-decoding and domain extension

We define the (G, H)*-irrelevant kernel* N as the intersection of the kernels of all $G \to H$ homomorphisms. We show that if $\mathrm{aHom}(G/N, H)$ is CombEcon then so is $\mathrm{aHom}(G, H)$.

The corollaries include a CombEcon result for {arbitrary $\to$ abelian} homomorphism codes because of the known CombEcon result for {abelian $\to$ abelian} homomorphism codes [15]. More generally we have a CombEcon result for {arbitrary $\to$ nilpotent} homomorphism codes in view of the CombEcon result for {nilpotent $\to$ nilpotent} homomorphism codes [21, 11]). Analogous results hold for CertEcon and AlgEcon under suitable assumptions on access to the groups.

The main tool underlying these results is the notion of *mean-list-decoding*, where we study not the distance to one received word but the average distance to a family of received words. Our main result in this area establishes the equivalence of CombEcon for list-decoding and mean-list-decoding; and analogous results for CertEcon and AlgEcon.

We discuss these results in Sections 3.4.2–3.4.4. The mean-list-decoding technique was inspired by the concatenated code technique used in [21].

Hom versus aHom

The reader may ask, why we (and all prior work) consider affine homomorphisms rather than homomorphisms. The reason is that affine homomorphisms are the more natural objects in this context. First, this object is more homogeneous. For instance, for finite H, under random affine homomorphisms, the image of any element $g \in G$ is uniformly distributed over H. This uniformity also serves as an inductive tool: when extending the domain from a subgroup G_0 to a group G, the action of any homomorphism $\varphi \in \mathrm{Hom}(G, H)$ can be split into actions on the cosets of G_0 in G. Those actions are affine homomorphisms. On the other hand we also note that list-decoding $\mathrm{Hom}(G, H)$ and $\mathrm{aHom}(G, H)$ are essentially equivalent tasks.

Proposition 1.4.12 (Hom versus aHom)**.** *Let G be a finite group, and H a group.*

(a) [20, Prop. 2.5] If $|\mathrm{Hom}(G, H)| \geq 2$, then $\mathsf{mindist}(\mathrm{Hom}(G, H)) = \mathsf{mindist}(\mathrm{aHom}(G, H))$.

(b) For $X \in \{Comb, Cert, Alg\}$, if $\mathrm{Hom}(G, H)$ is $X\,Econ$ then $\mathrm{aHom}(G, H)$ is $X\,Econ$. For $X \in \{Cert, Alg\}$, this statement requires that nearly uniform random elements of G be available.

Remark 1.4.13. The length of the aHom list for distance $\mathsf{mindist} - \varepsilon$ is not greater than $\frac{1}{1-\mathsf{mindist}+\varepsilon}$ times the length of the Hom list.

We discuss the reduction of affine to ordinary homomorphism codes further in Section 3.4.5.

1.4.3 Structure of the chapter

We discuss homomorphism codes in Chapter 3.

Much of our conceptual framework can be interpreted for codes in general, not just for homomorphism codes. In Section 3.1 we develop the general terminology. This includes the notions of *economy* in local list-decoding as well as the new concepts of *certificate-list decoding*

(Sec. 3.1.5), our semi-algorithmic intermediate concept, and *mean-list decoding*, our main tool for domain relaxation (Sec. 3.1.6), motivated by Guo and Sudan's use of repeated codes [21]. We also introduce *subword extenders*, which constitute the bridge between certificate-list decoding and algorithmic list-decoding (Sec. 3.1.5).

In Section 3.2 we present notation and terminology from group theory and computational group theory, including our *access models*, i. e., computational representations of groups (black-box, generator-relator presentations, etc., Sec. 3.2.4). We interpret general list-decoding concepts in the setting of homomorphism codes. We also present background on the structure of alternating groups (Sec. 3.2.5).

Section 3.3 gives formal statements of our results and occasional minor proofs that contribute to the conceptual development. The section includes a discussion of shallow-random-generation (SRG) groups (Section 3.3.4).

Section 3.4 describes reductions. We connect mean-list size to list size, and we infer our domain relaxation principle. We show the equivalence (both combinatorial and algorithmic) of list-decoding affine homomorphisms and list-decoding ordinary homomorphisms. To aid in our algorithmic proofs, we introduce individual-certificate list-decoding.

Section 3.5 outlines our basic strategy for the combinatorial bounds. It introduces our main tools, a sphere-packing bound and the shallow extension theorem. It indicates the differences between the approach to abelian domains and to alternating (and SRG) domains. For alternating and SRG domains, we discuss how same strategy also produces certificate-list-decoders.

Section 3.6 gives the full technical development of our results for abelian domains.

Sections 3.7 and 3.8 provide the proofs for alternating domains and their generalizations, the SRG groups.

We give two proofs that alternating groups are CombEcon. The first proof, in Section 3.7, is based on a sphere-packing argument and is non-constructive, but the method applies under quite general circumstances. The second, in Section 3.8, depends on structure specific to

the alternating groups (or more generally, to SRG groups), that proof directly translates to a semi-algorithmic result (CertEcon), and under restrictions of the codomain, also provides an algorithmic result (AlgEcon).

In Section 3.9, we describe the maximum agreement for homomorphism codes with solvable domain or codomain, slightly generalizing a result from [20].

CHAPTER 2

MONOTONE K-UNIFORM HYPERGRAPH PROPERTIES ARE WEAKLY EVASIVE

2.1 Preliminaries

2.1.1 Groups

Definition 2.1.1. For X a finite set and $k \geq 0$, let $\binom{X}{k}$ denote the set of k-element subsets of X. Let $\mathcal{P}(X)$ denote the power-set of X, i. e., the set of all subsets of X. For X and Y finite sets, let X^Y denote $\prod_{y \in Y} X$, so that an element of of X^Y is a $|Y|$-tuple of elements of X.

Our general reference to basic group theory is [32].

Definition 2.1.2. For groups Δ and Γ, write $\Delta \leq \Gamma$ to denote that Δ is a subgroup of Γ. For a prime p, a **p-group** is a group whose order is a power of p.

Definition 2.1.3. Let Γ be a group and X a set; let $\mathrm{Sym}(X)$ denote the group of all permutations of X (the symmetric group acting on X). An **action** of Γ on X is a homomorphism $\Gamma \to \mathrm{Sym}(X)$. We denote the image of $x \in X$ under the permutation corresponding to $\gamma \in \Gamma$ by x^γ. The action is **transitive** if $(\forall x, y \in X)(\exists \gamma \in \Gamma)(x^\gamma = y)$. For $S \subseteq X$ and $\gamma \in \Gamma$ we write $S^\gamma = \{s^\gamma \mid s \in S\}$. Thus, an action of Γ on X **induces** an action of Γ on $\binom{X}{k}$, where k is a nonnegative integer, and on $\mathcal{P}(X)$.

Definition 2.1.4. A **permutation group** is a subgroup $\Gamma \leq \mathrm{Sym}(X)$. When $\Gamma \to \mathrm{Sym}(X)$ is injective, we will sometimes find it convenient to identify Γ with its image.

Definition 2.1.5. Let $\Gamma \leq \mathrm{Sym}(V)$ and $\Delta \leq \mathrm{Sym}(W)$ be permutation groups. The **wreath product** $\Gamma \wr \Delta$ is defined as a subgroup of $\mathrm{Sym}(V \times W)$ as follows: we think of the domain $V \times W$ as the union of $|W|$ copies of V; the group Γ^W acts on this set by having each

copy of Γ act on the corresponding copy of V; the group Δ acts by permuting the copies. Formally, the elements of $\Gamma \wr \Delta$ correspond to the Cartesian product $\Gamma^W \times \Delta$; and the element $(\gamma_w : w \in W,\ \delta) \in \Gamma \wr \Delta$ (where $(\forall w \in W)(\gamma_w \in \Gamma)$ and $\delta \in \Delta$) takes $(v, w) \in V \times W$ to $(v^{\gamma_w}, w^{\delta})$. This action of $\Gamma \wr \Delta$ is called the **imprimitive action**. The group $\Gamma \wr \Delta$ also acts on V^W; identify each $(v_w : w \in W) \in V^W$ with the $|W|$-element set $\{(v_w, w) \mid w \in W\}$, then the imprimitive action induces an action on these sets. This is the **product action**.

Note that $|\Gamma \wr \Delta| = |\Gamma|^{|W|}|\Delta|$. So in particular if both Γ and Δ are p-groups (for the same prime p) then their wreath product $\Gamma \wr \Delta$ is also a p-group.

We will introduce one more action of $\Gamma \wr \Delta$ in Section 2.3.

Definition 2.1.6. Let V be a set with $|V| \geq 2$ and $\Gamma \leq \mathrm{Sym}(V)$ a group that acts transitively on V. A **system of imprimitivity** for Γ is a partition of V that is invariant under the action of Γ. If no such system exists other than $\{V\}$ and the discrete partition (consisting of singletons), we say that Γ is **primitive**.

2.1.2 Properties and Evasiveness

Definition 2.1.7. For n a nonnegative integer, a **boolean function on n variables** is a function $f\colon \{0,1\}^n \to \{0,1\}$.

Definition 2.1.8. For X a finite set, a **property over** X is a set $\mathscr{P} \subseteq \mathcal{P}(X)$.

Properties $\mathscr{P}$ over $[n] = \{1, \ldots, n\}$ can be identified with boolean functions $f\colon \{0,1\}^n \to \{0,1\}$ by setting $f(x_1, \ldots, x_n) = 1 \Leftrightarrow \{i \mid x_i = 1\} \in \mathscr{P}$. Here, we find it more convenient to work with properties over $[n]$.

Definition 2.1.9. A property $\mathscr{P}$ over X is **nontrivial** if $\mathscr{P} \neq \emptyset$ and $\mathscr{P} \neq \mathcal{P}(X)$.

Definition 2.1.10. A property $\mathscr{P}$ is **monotone** if for all $T \in \mathscr{P}$ and $S \subseteq T$, we have that $S \in \mathscr{P}$.

Definition 2.1.11. Let $\mathscr{P}$ be a property over X and let B, A be disjoint subsets of X. The **subproperty** of $\mathscr{P}$ defined by the pair (B, A) is the set $\mathscr{Q} = \{S \subseteq A \mid B \cup S \in \mathscr{P}\}$. Note that $\mathscr{Q}$ is a property over A.

Remark 2.1.12. In the language of boolean functions, a subproperty corresponds to a restriction of a function.

Definition 2.1.13. If Γ is a group acting on X and $\mathscr{P}$ is a property over X, we say that $\mathscr{P}$ is **invariant under the action of** Γ, or Γ**-invariant**, if for all $\gamma \in \Gamma$ and $S \in \mathscr{P}$, we have that $S^\gamma \in \mathscr{P}$.

Remark 2.1.14. For $\Gamma \leq \mathrm{Sym}([n])$, a boolean function $f\colon \{0,1\}^n \to \{0,1\}$ is Γ-invariant if for all $\gamma \in \Gamma$ and $(x_1, \ldots, x_n) \in \{0,1\}^n$, we have that $f(x_1, \ldots, x_n) = f(x_{1\gamma}, \ldots, x_{n\gamma})$.

Definition 2.1.15. A decision tree is a binary tree in which each non-leaf node is labeled with an element of a finite set X, and each leaf is labeled with either "true" or "false." Given a decision tree and a subset S of X, we traverse the tree starting at the root node by moving to the left child whenever the label on the node is in S (and to right child otherwise) until reaching a leaf. The decision tree is said to compute $\mathscr{P} \subseteq \mathcal{P}(X)$ if for all $S \subseteq X$, this traversal process ends with "true" exactly when $S \in \mathscr{P}$. For a property $\mathscr{P}$ over X, the **decision-tree complexity** of $\mathscr{P}$, denoted $D(\mathscr{P})$, is the least depth of any decision tree computing $\mathscr{P}$.

Note that $D(\mathscr{P}) \leq |X|$.

Definition 2.1.16. A property $\mathscr{P}$ over X is **evasive** if $D(\mathscr{P}) = |X|$.

Definition 2.1.17. For an infinite index set I, let $\{X_i \mid i \in I\}$ be a collection of sets and let $\mathscr{P}_i$ be a property over X_i. Say that the collection $\{\mathscr{P}_i \mid i \in I\}$ is **weakly evasive** if there is a $c > 0$ such that for all $i \in I$, we have that $D(\mathscr{P}_i) \geq c|X_i|$.

2.2 Orbit Augmentation Sequences

Our main tool for proving weak evasiveness of nontrivial monotone k-graph properties, orbit augmentation sequences, relies on this result of Rivest and Vuillemin.

Theorem 2.2.1 (Rivest and Vuillemin [29])**.** *Let X be a finite set with $|X|$ a power of a prime, Γ a group acting transitively on X, and $\mathscr{P}$ a property over X that is Γ-invariant with exactly one of $\emptyset, X$ in $\mathscr{P}$. Then $\mathscr{P}$ is evasive.*

To show that a property has a large decision-tree complexity, Rivest and Vuillemin and others, including Kulkarni, Qiao, and Sun, looked at subproperties to which they could apply Theorem 2.2.1 (or a related theorem of Kahn, Saks, and Sturtevant). They used the fact that properties pass on some of their symmetry to subproperties. They also used that the decision-tree complexity of a subproperty bounds the decision-tree complexity of the property from which it is induced. These are stated in the next two observations.

Observation 2.2.2. *If the property $\mathscr{P}$ from Definition 2.1.11 is Γ-invariant, and Γ fixes each of B and A, then the subproperty $\mathscr{Q}$ is also Γ-invariant.*

Proof. For all $\gamma \in \Gamma$ and $S \in \mathscr{Q}$ we have that $S^\gamma \subseteq A$ and $B \cup S^\gamma = (B \cup S)^\gamma \in \mathscr{P}$, so $S^\gamma \in \mathscr{Q}$. □

Observation 2.2.3. *If $\mathscr{Q}$ is a subproperty of $\mathscr{P}$, then $D(\mathscr{P}) \geq D(\mathscr{Q})$.* □

We now introduce the central concept of our work: "orbit augmentation sequences." These will enable us to build up ever larger sets with the property, using monotonicity. We continue building these larger sets until we either find that our property has high decision-tree complexity, or we eventually find that the whole set has the property and so the property is trivial. To accomplish this, we use transitive p-group actions on subproperties via the Rivest–Vuillemin Theorem (Theorem 2.2.1).

Definition 2.2.4.

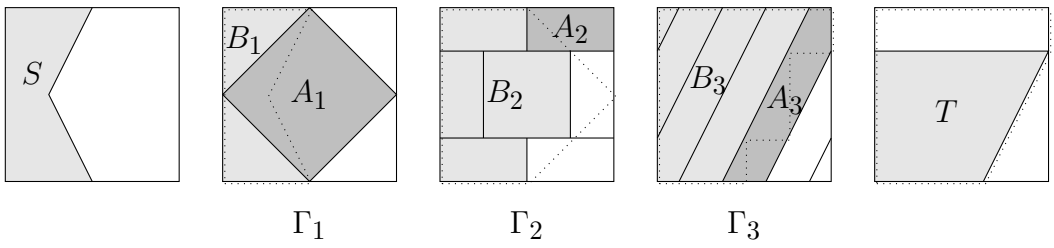


Figure 2.1: A pictorial interpretation of an (S, T) OA sequence. The entire square is the set X, and each region is an orbit of Γ_i. A dotted line outlines $B_{i-1} \cup A_{i-1}$.

- Let X be a finite set, $S, T \subseteq X$, $m \geq 0$;
- for $1 \leq i \leq m$, let B_i, A_i be disjoint subsets of X;
- for each i, let Γ_i be a group acting on X;
- for each i, assume Γ_i fixes B_i and A_i (setwise);
- for each i, assume the Γ_i-action on A_i is transitive;
- for each $i > 1$, assume $B_{i-1} \cup A_{i-1} \supseteq B_i$; and
- assume $S \supseteq B_1$ and $B_m \cup A_m \supseteq T$.

Then we will say that $((B_1, A_1, \Gamma_1), \ldots, (B_m, A_m, \Gamma_m))$ is an (S, T) **orbit augmentation sequence (OA sequence) for** X **using the groups** $\Gamma_1 \ldots, \Gamma_m$. A **full OA sequence for** X is an $(\emptyset, X)$ OA sequence for X.

The **spacing** of the OA sequence is defined to be $\min_{1 \leq i \leq m} |A_i|$ if $m \geq 1$, and ∞ if $m = 0$.

The next result connects decision-tree complexity with orbit augmentation sequences and thus provides our main technical tool.

Proposition 2.2.5 (Spacing Lower Bound)**.** *Let $\mathscr{P}$ be a nontrivial monotone Γ-invariant property over X and suppose there is a full OA sequence for X with spacing d using subgroups of Γ, with the order of each subgroup a power of a prime. Then the decision-tree complexity of $\mathscr{P}$ is $D(\mathscr{P}) \geq d$.*

Remark 2.2.6. It follows from the KSS arguments [24] that we don't need to require the groups Γ_i to have prime-power orders; it would suffice if they are Oliver groups.

Proof of Proposition 2.2.5. Say $((B_1, A_1, \Gamma_1), \ldots, (B_m, A_m, \Gamma_m))$ is a full OA sequence for X. For $1 \leq i \leq m$, let $\mathscr{Q}_i$ be a property over A_i, defined as the subproperty of $\mathscr{P}$ given by $\mathscr{Q}_i = \{S \subseteq A_i \mid B_i \cup S \in \mathscr{P}\}$. By Observation 2.2.2, each $\mathscr{Q}_i$ is Γ_i-invariant. Also, Γ_i acts transitively on A_i and $|\Gamma_i|$ is a power of a prime (so $|A_i|$ is a power of a prime). So, by Theorem 2.2.1, each $\mathscr{Q}_i$ is either trivial or evasive. If any $\mathscr{Q}_i$ is evasive, then by Observation 2.2.3, we have that $D(\mathscr{P}) \geq D(\mathscr{Q}_i) \geq d$, in which case we are done.

The only remaining case is that all $\mathscr{Q}_i$ are trivial, but we will show this does not occur. Let $B_{m+1} = X$. We will show by induction on $1 \leq i \leq m+1$ that $B_i \in \mathscr{P}$. For the base case, $B_1 = \emptyset \in \mathscr{P}$. For the inductive step, consider any $1 \leq i \leq m$, and suppose $B_i \in \mathscr{P}$. Then $\emptyset \in \mathscr{Q}_i$, and since $\mathscr{Q}_i$ is trivial and monotone, this implies $A_i \in \mathscr{Q}_i$. So, $B_i \cup A_i \in \mathscr{P}$. Since $\mathscr{P}$ is monotone and $B_{i+1} \subseteq B_i \cup A_i$, we have $B_{i+1} \in \mathscr{P}$. This completes the inductive step. So, $X = B_{m+1} \in \mathscr{P}$, so $\mathscr{P}$ is trivial. But we assumed that $\mathscr{P}$ was nontrivial, so this case does not occur. □

Next we derive some properties of orbit augmentation sequences.

Observation 2.2.7. *If $T \supseteq S'$, then the concatenation of an (S, T) OA sequence for X with an (S', T') OA sequence for X is an (S, T') OA sequence for X. If these OA sequences have spacing d and d' respectively, the concatenation has spacing* $\min\{d, d'\}$.

Observation 2.2.8. *Suppose that there is an (S, T) OA sequence for X and that the action of each group used in the OA sequence fixes $U \subseteq X$. Then there is a $(U \cup S, U \cup T)$ OA sequence with the same spacing as the original OA sequence and using the same groups.*

Remark 2.2.9. This observation also holds with intersection in place of union.

Proof. Let $((B_1, A_1, \Gamma_1), \ldots, (B_m, A_m, \Gamma_m))$ be an (S, T) OA sequence for X. Then

$$(((U \setminus A_1) \cup B_1, A_1, \Gamma_1), \ldots, ((U \setminus A_m) \cup B_m, A_m, \Gamma_m)) \tag{2.1}$$

is a $(U \cup S, U \cup T)$ OA sequence for X. □

Corollary 2.2.10. *Suppose that there is an $(\emptyset, T)$ OA sequence for X with spacing d and an $(\emptyset, T')$ OA sequence for X with spacing d' using subgroups of Γ'. Suppose the action of Γ' fixes T. Then there is an $(\emptyset, T \cup T')$ OA sequence for X with spacing $\min\{d, d'\}$ using the groups used in the two original OA sequences.*

Proof. By Observation 2.2.8 with $U = T$, there is a $(T, T \cup T')$ OA sequence for X with spacing d' using the same groups used in the $(\emptyset, T')$ OA sequence. We concatenate the $(\emptyset, T)$ OA sequence with the $(T, T \cup T')$ OA sequence. □

Corollary 2.2.11 (Composition Lemma)**.** *Suppose that for each i with $1 \leq i \leq s$, there is an $(\emptyset, T^{(i)})$ OA sequence for X with spacing $d^{(i)}$. Suppose that for $2 \leq i \leq s$, the $(\emptyset, T^{(i)})$ OA sequence uses subgroups of Γ. Suppose that for $1 \leq i \leq s-1$, the action of Γ fixes $T^{(i)}$. Then there is an $(\emptyset, T^{(1)} \cup \cdots \cup T^{(s)})$ OA sequence for X with spacing $\min_{1 \leq i \leq s} d^{(i)}$ using the groups used in the s original OA sequences.*

Proof. By induction on s. The base case $s = 2$ is Corollary 2.2.10. □

Observation 2.2.12. *Let $A \subseteq X$ and let Γ be a group acting transitively on A. Suppose there is a full OA sequence for X' with spacing d' using groups $\Gamma'_1, \ldots, \Gamma'_{m'}$. Then there is an $(\emptyset, A \times X')$ OA sequence for $X \times X'$ with spacing $|A|d'$ using the groups $\Gamma \times \Gamma'_i$, $1 \leq i \leq m'$.*

Proof. Let $((B'_1, A'_1, \Gamma'_1), \ldots, (B'_{m'}, A'_{m'}, \Gamma'_{m'}))$ be a full OA sequence for X'. Then

$$((A \times B'_1, A \times A'_1, \Gamma \times \Gamma'_1), \ldots, (A \times B'_{m'}, A \times A'_{m'}, \Gamma \times \Gamma'_{m'})) \tag{2.2}$$

is an $(\emptyset, A \times T')$ OA sequence for $X \times X'$. □

Lemma 2.2.13 (Direct Product)**.** *Suppose there is a full OA sequence for X with spacing d using groups $\Gamma_1, \ldots, \Gamma_m$, and there is a full OA sequence for X' with spacing d' using groups*

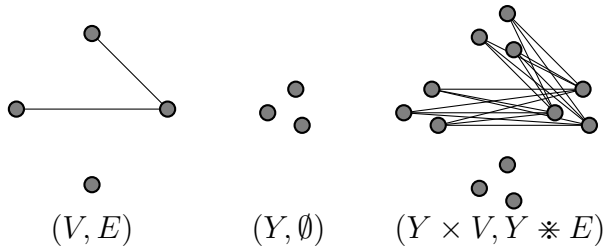


Figure 2.2: An example of a Y-lifting $Y \divideontimes E$

$\Gamma'_1, \ldots, \Gamma'_{m'}$. *Then there is a full OA sequence for* $X \times X'$ *with spacing* dd' *using the groups* $\Gamma_i \times \Gamma'_j$, $1 \leq i \leq m$, $1 \leq j \leq m'$.

Proof. For each $1 \leq i \leq m$, apply Observation 2.2.12 to produce an $(\emptyset, A_i \times X')$ OA sequence for $X \times X'$ with spacing $|A_i|d'$ using the groups $\Gamma_i \times \Gamma'_j$, $1 \leq j \leq m'$. We apply Observation 2.2.8 to the above with $U = B_i \times X'$ to give a $(B_i \times X', (B_i \cup A_i) \times X')$ OA sequence for $X \times X'$ with spacing $|A_i|d'$ using the same groups.

The concatenation of these OA sequences for $1 \leq i \leq m$ is a full OA sequence for $X \times X'$ with spacing dd'. □

2.3 Liftings of Hypergraphs and Wreath Product Action

Definition 2.3.1. Let V be a finite set, and k a positive integer. Let $\mathrm{Sym}(V)$ denote the symmetric group on V. A **property of k-uniform hypergraphs with vertex set V**, or **k-graph property**, is a property over $\binom{V}{k}$ that is $\mathrm{Sym}(V)$-invariant (under the induced action, where $\sigma \in \mathrm{Sym}(V)$ sends $e \in \binom{V}{k}$ to $e^\sigma = \{v^\sigma \mid v \in e\}$).

Next we introduce a technical tool: Y-liftings of a k-graph, where Y is a set.

Definition 2.3.2. Let V and Y be finite sets, $k \geq 1$, and $E \subseteq \binom{V}{k}$. Let the **Y-lifting of E**, $Y \divideontimes E$, be $\{\{(y_1, v_1), \ldots, (y_k, v_k)\} \in \binom{Y \times V}{k} \mid y_1, \ldots, y_k \in Y, \{v_1, \ldots, v_k\} \in E\}$ (in particular, $v_1, \ldots, v_k$ are distinct). Thinking of E as the edge set of a k-graph, $Y \divideontimes E$ is the edge set of the k-graph obtained by replacing each vertex with a copy of Y.

If $\Gamma \leq \mathrm{Sym}(Y)$ and $\Delta \leq \mathrm{Sym}(V)$, then the imprimitive action of $\Gamma \wr \Delta$ on $Y \times V$ induces an action (the **lifting action**) on $Y \circledast \binom{V}{k}$.

Remark 2.3.3. In hypergraph terminology, the k-graph with vertex set $Y \times V$ and edge set $Y \circledast E$ is the anti-lexicographic product of the empty k-graph on vertex set Y and the k-graph (V, E) (cf. [22]).

Observation 2.3.4. *Suppose there is an (S, T) OA sequence for $X \subseteq \binom{V}{k}$ with spacing d using subgroups $\Gamma_1, \ldots, \Gamma_m$ of $\mathrm{Sym}(V)$. Let Y be a finite set, and let $\Gamma' \leq \mathrm{Sym}(Y)$ act transitively. Then there is a $(Y \circledast S, Y \circledast T)$ OA sequence for $Y \circledast X$ with spacing $d|Y|^k$ using the groups $\Gamma' \wr \Gamma_i$, $1 \leq i \leq m$, with the lifting action.*

Proof. Let $((B_1, A_1, \Gamma_1), \ldots, (B_m, A_m, \Gamma_m))$ be an (S, T) OA sequence for X. Then

$$((Y \circledast B_1, Y \circledast A_1, \Gamma' \wr \Gamma_1), \ldots, (Y \circledast B_m, Y \circledast A_m, \Gamma' \wr \Gamma_m)) \tag{2.3}$$

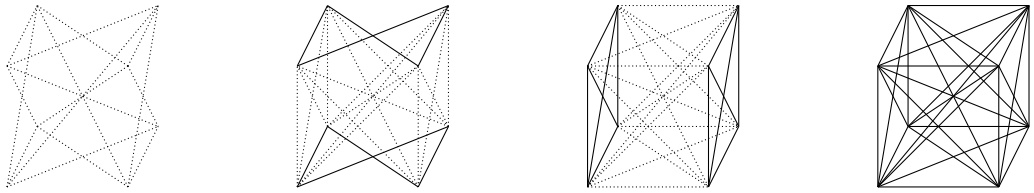
is an $(Y \circledast S, Y \circledast T)$ OA sequence for $Y \circledast X$. □

We find it convenient to deal with Y-liftings in the language of multisets.

Definition 2.3.5. A **multiset** with underlying set I is a function $f \colon I \to \mathbb{Z}_{\geq 0}$. The **cardinality** of f is $|f| = \sum_{i \in I} f(i)$. The **support** of f is $\mathrm{supp}(f) = \{i \in I \mid f(i) \neq 0\}$. To mirror the notation for sets, we may depict a multiset f as $\{\!\{i_1, i_2, \ldots, i_{|f|}\}\!\}$, where $i_1, \ldots, i_{|f|} \in I$, and each $i \in I$ appears with multiplicity $f(i)$. For a positive integer k, we will write $f < k$ to denote that for all i, $f(i) < k$. Let $\binom{I}{k}$ denote the set of multisets with underlying set I and cardinality k.

2.4 Hypergraphs with a Prime-Power Number of Vertices

We can now make our key construction of an OA sequence with large spacing for hypergraphs with a prime-power number of vertices.



$S = B_1 = \emptyset$
$A_1 = [2] \circledast ([2] \circledast \binom{[2]}{2})$
$\Gamma_1 = \mathbb{Z}/2\mathbb{Z}^{\mathrm{a}} \wr (\mathbb{Z}/2\mathbb{Z}^{\mathrm{b}} \wr 1^{\mathrm{c}})$

[a]swaps left and right
[b]swaps top and bottom
[c]fixes front and back

$B_2 = [2] \circledast (\binom{\{1\}\times[2]}{2} \cup \binom{\{2\}\times[2]}{2})$
$A_2 = [2] \circledast (\binom{\{1\}\times[2]}{1} \times \binom{\{2\}\times[2]}{1})$
$\Gamma_2 = \mathbb{Z}/2\mathbb{Z}^{\mathrm{d}} \wr (\mathbb{Z}/2\mathbb{Z}^{\mathrm{e}} \times \mathbb{Z}/2\mathbb{Z}^{\mathrm{f}})$

[d]swaps left and right
[e]swaps bottom front and bottom back
[f]swaps top front and top back

$B_3 = \binom{\{1\}\times[2]^2}{2} \cup \binom{\{2\}\times[2]^2}{2}$
$A_3 = \binom{\{1\}\times[2]^2}{1} \times \binom{\{2\}\times[2]^2}{1}$
$\Gamma_3 = \mathbb{Z}/2^2\mathbb{Z}^{\mathrm{g}} \times \mathbb{Z}/2^2\mathbb{Z}^{\mathrm{h}}$

[g]cycles left four vertices
[h]cycles right four vertices

$T = \binom{[2]^3}{2}$

Figure 2.3: The full OA sequence for $\binom{[2]^3}{2}$ constructed in the proof of Proposition 2.4.2. The sets B_i are shown with solid edges, and the sets A_i are shown with dotted edges.

Lemma 2.4.1. *Let $r_k = \frac{1}{2}k^2 + \frac{1}{2}k - 1$. Then for all $k \geq 2$ and all multisets f with $|f| = k$, $f < k$, we have that $k + \sum_{i \in \mathrm{supp}(f)} r_{f(i)} \leq r_k$.*

Proof. We have that

$$\sum r_{f(i)} = \sum \frac{1}{2}(f(i)+2)(f(i)-1) \tag{2.4}$$
$$\leq \sum \frac{1}{2}(k+1)(f(i)-1) \tag{2.5}$$
$$= \frac{1}{2}(k+1)\,(k - |\mathrm{supp}(f)|) \tag{2.6}$$
$$\leq \frac{1}{2}(k+1)(k-2), \tag{2.7}$$

where the sums are taken over i in the support of f. So $k + \sum r_{f(i)} \leq k + \frac{1}{2}(k+1)(k-2) = r_k$. Equality holds if and only if $f(i) = k-1$ for some i. □

Proposition 2.4.2. *For each positive integer k, there is a real number r_k such that for any prime p and nonnegative integer ℓ, there is a full OA sequence for $\binom{[p^\ell]}{k}$ with spacing at least $p^{\ell k - r_k}$ using p-subgroups of $\mathrm{Sym}([p^\ell])$.*

Note that now our main technical result, Corollary 1.3.2, is immediate in the case when the number of vertices is a power of a prime. We state this as a corollary.

Corollary 2.4.3. *For any fixed prime p and fixed positive integer k, nontrivial monotone*

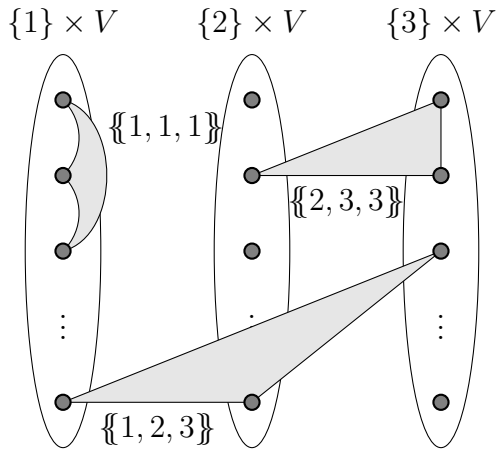


Figure 2.4: Each hyperedge in the figure is labeled with its "type," its image under π. All the hyperedges with the same type are added to the OA sequence in the same batch.

properties of k-uniform hypergraphs with a power of p vertices are weakly evasive.

Proof of Proposition 2.4.2. Let r_k be as in Lemma 2.4.1. We proceed by induction on k.

As a base case, take $k = 1$ with the OA sequence $((\emptyset, \binom{[p^\ell]}{1}), \mathbb{Z}/p^\ell\mathbb{Z}))$.

For the inductive step, consider $k > 1$. Let ℓ_0 be the smallest ℓ such that $p^\ell \geq k$. We proceed by induction on ℓ.

As a base case, for any $\ell < \ell_0$, $\binom{V}{k} = \emptyset$, and an OA sequence of zero length suffices.

For the inductive step, consider $\ell \geq \ell_0$. Let V be a set with $|V| = p^{\ell-1}$. Since $|[p] \times V| = p^\ell$, it suffices to construct a full OA sequence for $\binom{[p]\times V}{k}$ using p-subgroups of $\mathrm{Sym}([p] \times V)$.

Let $\pi\colon \binom{[p]\times V}{k} \to \left(\!\!\binom{[p]}{k}\!\!\right)$ be defined by $\{(i_1, v_1), \ldots, (i_k, v_k)\} \mapsto \{\!\{i_1, \ldots, i_k\}\!\}$. The image of a hyperedge under π is the hyperedge's "type." We will construct an OA sequence in batches, and all hyperedges of the same type will be in the same batch.

Adding edges of the type $\{\!\{i, i, \ldots, i\}\!\}$, $i \in [p]$: By the induction hypothesis for ℓ, there is a full OA sequence for $\binom{V}{k}$ with spacing at least $p^{\ell k - k - r_k}$ using p-subgroups of $\mathrm{Sym}(V)$. By Observation 2.3.4, there is a full OA sequence for $[p] \circledast \binom{V}{k} \subseteq \binom{[p]\times V}{k}$ with spacing $p^{\ell k - r_k}$ using p-subgroups of $\mathbb{Z}/p\mathbb{Z} \wr \mathrm{Sym}(V)$. Since $\bigcup_{i\in[p]} \pi^{-1}(\{\!\{i, i, \ldots, i\}\!\}) = \bigcup_{i\in[p]} \binom{\{i\}\times V}{k} \subseteq [p] \circledast \binom{V}{k}$, this sequence is an $\left(\emptyset, \bigcup_{i\in[p]} \binom{\{i\}\times V}{k}\right)$ OA sequence for $\binom{[p]\times V}{k}$.

Adding hyperedges of other types: Consider any $f \in \left(\!\!\binom{[p]}{k}\!\!\right)$ with $f < k$. For each $i \in$

$\operatorname{supp}(f)$, by the induction hypothesis for k, there is a full OA sequence for $\binom{\{i\}\times V}{f(i)}$ with spacing at least $p^{(\ell-1)f(i)-r_{f(i)}}$ using p-subgroups of $\operatorname{Sym}(\{i\}\times V)$. Let $T_f = \prod_{i\in\operatorname{supp}(f)} \binom{\{i\}\times V}{f(i)}$. By the Direct Product Lemma (Lemma 2.2.13), there is a full OA sequence for T_f with spacing at least $\prod_{i\in\operatorname{supp}(f)} p^{\ell f(i)-f(i)-r_{f(i)}}$ using p-subgroups of $\prod_{i\in\operatorname{supp}(f)} \operatorname{Sym}(\{i\}\times V)$. By Lemma 2.4.1, this spacing is at least $p^{\ell k-r_k}$. This group can be identified with a subgroup of $\operatorname{Sym}([p]\times V)$, and T_f can be identified with $\pi^{-1}(f)$ by $(e_i)_{i\in\operatorname{supp}(f)} \mapsto \bigcup_{i\in\operatorname{supp}(f)} e_i$. These identifications preserve the group action. So, there is a full OA sequence for $\pi^{-1}(f)$ with spacing at least $p^{\ell k-r_k}$ using p-subgroups of $\prod_{i\in[p]} \operatorname{Sym}(\{i\}\times V)$.

Combining the batches: We apply the Composition Lemma (Corollary 2.2.11) with the above $\left(\emptyset, \bigcup_{i\in[p]} \binom{\{i\}\times V}{k}\right)$ OA sequence as the first OA sequence, and the above $(\emptyset, \pi^{-1}(f))$ OA sequences, $f \in \left(\!\!\binom{[p]}{k}\!\!\right)$, $f < k$, as the other OA sequences. Since

$$\left(\bigcup_{i\in[p]} \binom{\{i\}\times V}{k}\right) \cup \bigcup_{f\in\left(\!\!\binom{[p]}{k}\!\!\right), f<k} \pi^{-1}(f) = \binom{[p]\times V}{k}, \tag{2.8}$$

this produces a full OA sequence for $\binom{[p]\times V}{k}$ using p-subgroups of $\operatorname{Sym}([p]\times V)$.

This completes the inductive step for ℓ, and also for k. □

2.5 An Arbitrary Number of Vertices

We can now construct OA sequences with large spacing for hypergraphs with any number of vertices.

Lemma 2.5.1. *Let $q_k = k^2 + 2k - 2$. Then for all $k \geq 2$ and all multisets f with $|f| = k$, $f < k$, we have that $k \leq q_k$, $2k + \sum_{i\in\operatorname{supp}(f)} q_{f(i)} \leq q_k$, and $2k + r_k \leq q_k$, where r_k is as in Lemma 2.4.1.*

Proof. We have

$$\sum q_{f(i)} = \sum((f(i)+3)(f(i)-1)+1) \tag{2.9}$$

$$\leq \sum((k+2)(f(i)-1)+1) \tag{2.10}$$

$$= (k+2)(k-|\mathrm{supp}(f)|)+|\mathrm{supp}(f)| \tag{2.11}$$

$$\leq (k+2)(k-2)+2, \tag{2.12}$$

where the sums are taken over $i \in \mathrm{supp}(f)$. So $2k+\sum q_{f(i)} \leq 2k+(k+2)(k-2)+2 = q_k$. Equality holds if and only if $f(i) = k-1$ for some i.

And, $q_k - (2k+r_k) = \frac{1}{2}(k-2)(k+1) \geq 0$ since $k \geq 2$. □

Theorem 2.5.2. *Let p be a prime. For each positive integer k, there is a real number q_k such that for any nonnegative integer v, and any set V with $|V| = v$, there is a full OA sequence for $\binom{V}{k}$ with spacing at least $p^{-q_k}v^k$ using p-subgroups of* $\mathrm{Sym}(V)$.

Our main technical result, Corollary 1.3.2, is now immediate. □

Proof of Theorem 2.5.2. Let r_k and q_k be as in Lemmas 2.4.1 and 2.5.1. We proceed by induction on k.

As a base case, consider $k = 1$. If $v = 0$, an OA sequence of length zero suffices. Consider any positive integer v. Let ℓ be the largest integer with $p^\ell \leq v$. There are $V_1, \ldots, V_p \subseteq V$, not necessarily disjoint, with $|V_1| = \cdots = |V_p| = p^\ell$, $V_1 \cup \cdots \cup V_p = V$. Let $\Gamma_1, \ldots, \Gamma_p$ be subgroups of $\mathrm{Sym}(V_1), \ldots, \mathrm{Sym}(V_p)$, respectively, each generated by a cycle of length p^ℓ. Then the sequence of length p whose i-th entry is $\left(\binom{(V_1\cup\cdots\cup V_{i-1})\setminus V_i}{1}, \binom{V_i}{1}, \Gamma_i\right)$ is a full OA sequence for $\binom{V}{1}$ of length p using p-subgroups of $\mathrm{Sym}(V)$.

For the inductive step, consider $k > 1$. If $v < k$, an OA sequence of length zero suffices. If $k \leq v < p$, an OA sequence of length $\binom{v}{k}$ using the trivial group suffices, with spacing $1 = p^{-k}p^k \geq p^{-q_k}v^k$. Consider any $v \geq p$. Let ℓ be the largest integer with $p^{\ell+1} \leq v$. Note $p^\ell > p^{-2}v$. Let s be the largest integer such that $sp^\ell \leq v$, and let $V_1, V_2, \ldots, V_s$ be

a partition of V with $|V_1| = \cdots = |V_{s-1}| = p^\ell$ and $p^\ell \le |V_s| < 2p^\ell$. Let $V' \supseteq V_s$ with $|V'| = p^{\ell+1}$. By Proposition 2.4.2, there is an $\left(\emptyset, \binom{V'}{k}\right)$ OA sequence for $\binom{V}{k}$ with spacing $p^{(\ell+1)k-r_k} \ge p^{-k-r_k}v^k$ using p-subgroups of $\mathrm{Sym}(V')$. This OA sequence is an $\left(\emptyset, \binom{V_s}{k}\right)$ OA sequence. By Proposition 2.4.2, for each i with $1 \le i \le s-1$, there is an $\left(\emptyset, \binom{V_i}{k}\right)$ OA sequence for $\binom{V}{k}$ with spacing $p^{\ell k-r_k} \ge p^{-2k-r_k}v^k$ using p-subgroups of $\mathrm{Sym}(V_i)$. By Lemma 2.5.1, the spacing of each of these OA sequences is at least $p^{-q_k}v^k$.

Let $\pi\colon \binom{V}{k} \to \binom{[s]}{k}$ be defined by $\pi(e)(i) = |e \cap V_i|$.

As in the proof of Proposition 2.4.2, consider any $f \in \binom{[s]}{k}$ with $f < k$. For each $i \in \mathrm{supp}(f)$, by the induction hypothesis for induction on k, there is a full OA sequence for $\binom{V_i}{f(i)}$ with spacing at least $p^{-q_{f(i)}}|V_i|^{f(i)} \ge p^{\ell f(i)-q_{f(i)}}$ using p-subgroups of $\mathrm{Sym}(V_i)$. Let $T_f = \prod_{i\in\mathrm{supp}(f)} \binom{V_i}{f(i)}$. By the Direct Product Lemma (Lemma 2.2.13), there is a full OA sequence for T_f with spacing at least $\prod_{i\in\mathrm{supp}(f)} p^{\ell f(i)-q_{f(i)}}$ using p-subgroups of $\prod_{i\in\mathrm{supp}(f)} \mathrm{Sym}(V_i)$. By Lemma 2.5.1, using $p^\ell \ge p^{-2}v$, this spacing is at least $p^{-q_k}v^k$. This group can be identified with a subgroup of $\mathrm{Sym}(V)$, and T_f can be identified with $\pi^{-1}(f)$ by $(e_i)_{i\in\mathrm{supp}(f)} \mapsto \bigcup_{i\in\mathrm{supp}(f)} e_i$. These identifications preserve the group action. So, there is an $(\emptyset, \pi^{-1}(f))$ OA sequence for $\pi^{-1}(f)$ with spacing at least $p^{-q_k}v^k$ using p-subgroups of $\prod_{i\in[s]} \mathrm{Sym}(V_i)$.

We apply the Composition Lemma (Corollary 2.2.11) with the above $\left(\emptyset, \binom{V_s}{k}\right)$ OA sequence as the first OA sequence, the above $\left(\emptyset, \binom{V_i}{k}\right)$ OA sequences, $1 \le i \le s-1$, as the next OA sequences, and the above $(\emptyset, \pi^{-1}(f))$ OA sequences, $f \in \binom{[s]}{k}$, $f < k$, as the other OA sequences. Since

$$\binom{V_s}{k} \cup \bigcup_{1\le i\le s-1} \binom{V_i}{k} \cup \bigcup_{f\in\binom{[s]}{k}, f<k} \pi^{-1}(f) = \binom{V}{k}, \tag{2.13}$$

this produces a full OA sequence for $\binom{V}{k}$ using p-subgroups of $\mathrm{Sym}(V)$.

This completes the inductive step. □

Combining Lemma 2.5.1 and Theorem 2.5.2 with the Spacing Lower Bound (Proposi-

tion 2.2.5) we obtain our main technical result:

Theorem 2.5.3. *The decision-tree complexity of a nontrivial monotone property of k-graphs with v vertices is at least*

$$2^{-(k+1)^2+3}v^k. \tag{2.14}$$

2.6 Monotone Properties Invariant Under Large Primitive Groups

We turn to our main result, that nontrivial monotone properties invariant under large primitive groups are weakly evasive. It relies on a theorem of Cameron from 1981 [12] that such groups have subgroups that look like the group of symmetries that describe a product of hypergraphs.

Definition 2.6.1. Let S_v denote $\mathrm{Sym}([v])$ and $A_v \leq S_v$ the alternating group, consisting of the even permutations on $[v]$. Let $S_v^{(k)} \leq \mathrm{Sym}\left(\binom{[v]}{k}\right)$ be the image of S_v under the induced action on $\binom{[v]}{k}$, and define $A_v^{(k)} \leq \mathrm{Sym}\left(\binom{[v]}{k}\right)$ analogously. Note that $S_v^{(k)} \times \cdots \times S_v^{(k)}$ (t copies) can be identified with the subgroup of $S_v^{(k)} \wr S_t$ corresponding to the identity element in S_t.

Theorem 2.6.2 (Cameron, 1981)**.** *There exists a constant C such that for all $\varepsilon > 0$ and all sufficiently large n, the following holds: For any primitive permutation group $\Gamma \leq S_n$ with order at least $\exp(n^\varepsilon)$, there are positive integers k, t, and v with $n = \binom{v}{k}^t$ such that*

$$\underbrace{A_v^{(k)} \times \cdots \times A_v^{(k)}}_{t} \leq \Gamma \leq S_v^{(k)} \wr S_t, \tag{2.15}$$

where $S_v^{(k)} \wr S_t$ acts in the product action, and $kt \leq C/\varepsilon$.

The results of the previous sections allow us to put a bound on the decision-tree complexity of nontrivial monotone $(A_v^{(k)} \times \cdots \times A_v^{(k)})$-invariant properties.

Theorem 2.6.3. *(a) Let t be a positive integer, let $v_1, \dots, v_t$, $k_1, \dots, k_t$ be positive integers, and let $X = \binom{[v_1]}{k_1} \times \cdots \times \binom{[v_t]}{k_t}$. Note that $\Gamma = A_{v_1}^{(k_1)} \times \cdots \times A_{v_t}^{(k_t)}$ acts on X. Let $q = \sum_{i=1}^{t}(k_i^2 + 2k_i - 2)$. Any nontrivial monotone Γ-invariant property over X has decision-tree complexity at least $3^{-q} v_1^{k_1} \cdots v_t^{k_t}$.*

(b) In particular, nontrivial monotone properties of t-partite partwise-uniform k-graphs are weakly evasive for bounded k, t.

Proof of Theorem 2.6.3. (a) Let q_k be as in Lemma 2.5.1. For each i, since every element of S_{v_i} of odd order is in A_{v_i}, by Theorem 2.5.2 with $p = 3$, there is a full OA sequence for $\binom{[v_i]}{k_i}$ with spacing $3^{-(k_i^2+2k_i-2)} v_i^{k_i}$ using 3-subgroups of $A_{v_i}^{(k_i)}$. By the Direct Product Lemma (Lemma 2.2.13), there is a full OA sequence for X with spacing $\prod_{i=1}^{t} 3^{-(k_i^2+2k_i-2)} v_i^{k_i}$ using 3-subgroups of Γ.

(b) Properties of t-partite partwise-uniform k-graphs are properties that are invariant under $S_{v_1}^{(k_1)} \times \cdots \times S_{v_t}^{(k_t)}$ for some $k_1, \dots, k_t$ with $k_1 + \cdots + k_t = k$, so this follows from (a). □

As a corollary of these two results, we have our main result:

Theorem 2.6.4. *For any constant $\varepsilon > 0$, nontrivial monotone properties over $[n]$ that are invariant under the action of a primitive group of order at least $\exp(n^{\varepsilon})$ are weakly evasive.*

CHAPTER 3

LIST DECODING HOMOMORPHISM CODES WITH ARBITRARY CODOMAIN

3.1 Terminology for general codes

3.1.1 List-decoding

We introduce some terminology that applies to codes in general and not just homomorphism codes.

Let Σ be an alphabet and Ω a set we think of as the set of positions. We view Σ^Ω, the set of $\Omega \to \Sigma$ functions, as our code space; its elements are called *words.* We write $\mathrm{dist}(u, w)$ for the *normalized Hamming distance* between two words $u, w \in \Sigma^\Omega$ (so $0 \leq \mathrm{dist}(u, w) \leq 1$) and refer to it simply as *distance.* Let $\mathcal{C} \subseteq \Sigma^\Omega$ be a code; its elements are called *codewords.* Write $\mathsf{mindist}(\mathcal{C})$ (or simply $\mathsf{mindist}$) for the minimum distance between distinct codewords in $\mathcal{C}$.

Words we wish to decode are referred to in the literature as *received words.* We refer to the set of codewords within a specified distance ρ of a received word $f \in \Sigma^\Omega$ as *"the list"* and denote it by $\mathcal{L} = \mathcal{L}(\mathcal{C}, f, 1-\rho)$. We write $\ell(\mathcal{C}, 1-\rho) \coloneqq \max_f |\mathcal{L}(\mathcal{C}, f, 1-\rho)|$. We use $1-\rho$ as a parameter rather than ρ, as it is a more natural quantity in this context; see Section 3.1.2 below.

3.1.2 Minimum distance versus maximum agreement

Recall that our code space is Σ^Ω, the set of $\Omega \to \Sigma$ functions. In the theory of error-correcting codes, the usual measure of distance between two functions (strings) is the (normalized) Hamming distance, the fraction of symbols on which they differ. Following [19], we find it convenient to consider the measure complementary to normalized Hamming distance, the

(normalized) *agreement*,

$$\operatorname{agr}(f, g) \coloneqq \frac{1}{|\Omega|}|\{\omega \in \Omega \mid f(\omega) = g(\omega)\}|, \tag{3.1}$$

the fraction of positions on which the two functions $f, g\colon \Omega \to \Sigma$ agree.

The list $\mathcal{L}(\mathcal{C}, f, \lambda)$ consists of codewords that have agreement at least λ with f, which are the same as codewords within distance $1 - \lambda$ of f.

Definition 3.1.1. The *maximum agreement* of the code $\mathcal{C}$ is given by

$$\Lambda_{\mathcal{C}} \coloneqq \max_{\substack{\varphi,\psi\in\mathcal{C}\\ \varphi\neq\psi}} \operatorname{agr}(\varphi, \psi). \tag{3.2}$$

Fact 3.1.2. The minimum distance is the complement of the maximum agreement, i. e.,

$$\mathsf{mindist} = 1 - \Lambda_{\mathcal{C}}. \tag{3.3}$$

So, the codewords within distance $(\mathsf{mindist} - \varepsilon)$ of a received word f are the same as the codewords with agreement at least $\Lambda_{\mathcal{C}} + \varepsilon$ with f. We aim to do list-decoding with this level of agreement. Classes of examples for the infeasibility of list-decoding outside this range were provided by Guo and Sudan [21] for abelian domain and codomain, and we provide such classes for alternating domain (see Section 3.7.3), so the list-decoding radius is $\mathsf{mindist}$ for those classes.

3.1.3 Combinatorial list-decoding

The list-decoding problem splits into a combinatorial and an algorithmic part.

The combinatorial problem, to which we refer as *combinatorial list-decoding*, asks to bound the size of the list. Typically, we take $\lambda = \Lambda_{\mathcal{C}} + \varepsilon$ and we wish to obtain a bound $\ell(\mathcal{C}, \lambda) \leq q(\varepsilon)$, that depends only on ε and the class $\mathscr{C}$ of codes under discussion ($\mathcal{C} \in \mathscr{C}$).

We say that a class $\mathscr{C}$ of codes is **CombEcon** ("combinatorially economically list-decodable") if $q(\varepsilon) = \text{poly}(1/\varepsilon)$ for $\mathcal{C} \in \mathcal{F}$. We say that $\mathscr{C}$ is CombEcon *of degree* c if the polynomial $q(\varepsilon)$ has degree c. (With some abuse of terminology, we shall refer to a code $\mathcal{C}$ as a *CombEcon* code if the class $\mathscr{C}$ of codes is understood from the context.)

3.1.4 Algorithmic list-decoding

We shall describe algorithms with certain performance guarantees typically guaranteeing properties of the output with specified probability.

A *list-decoder* is an algorithm that, given the received word $f \in \Sigma^\Omega$ and the agreement λ, lists a superset $\widetilde{\mathcal{L}}$ of the list $\mathcal{L} = \mathcal{L}(\mathcal{C}, f, \lambda)$. Typically, we take $\lambda = \Lambda_{\mathcal{C}} + \varepsilon$, and we usually wish to produce a list of size $|\widetilde{\mathcal{L}}| \le \widetilde{q}(\varepsilon)$ for some $\widetilde{q}(\varepsilon)$ that depends only on ε and the class $\mathscr{C}$ of codes under discussion ($\mathcal{C} \in \mathscr{C}$).

Adapting the terminology of [19] and [15], we say that a *local algorithm* is a probabilistic algorithm that has oracle access to the received word f.

We say that $\mathscr{C}$ is an *AlgEcon* ("algorithmically economically list-decodable") class of codes if there exists a local list-decoder with the following features.

Input: $\Lambda_{\mathcal{C}}$, $\varepsilon > 0$, oracle access to $f \in \Sigma^\Omega$.

Notation: $\mathcal{L} = \mathcal{L}(\mathcal{C}, f, \Lambda_{\mathcal{C}} + \varepsilon)$.

Output: A list $\widetilde{\mathcal{L}}$ of codewords in $\mathcal{C}$ of length $|\widetilde{\mathcal{L}}| = \text{poly}(1/\varepsilon)$.

Guarantee: With probability $\ge 3/4$, we have $\widetilde{\mathcal{L}} \supseteq \mathcal{L}$.

Cost:

(i) $\text{poly}(\log|\Omega|, 1/\varepsilon)$ queries to the received word f.

(ii) $\text{poly}(\log|\Omega|, \log|\Sigma|, 1/\varepsilon)$ amount of work.

Access: The meaning of this definition depends also on the access model to Σ and Ω. We shall clarify this in each application.

Strong AlgEcon

In the **unit-cost model** for Σ, we charge unit cost to name an element of Σ.

Definition 3.1.3. We say that $\mathcal{C}$ is a **strong AlgEcon** code if there exists a list-decoder satisfying the conditions of AlgEcon, except with (ii) replaced by the following.

(ii') poly$(\log|\Omega|, 1/\varepsilon)$ amount of work in the unit-cost model for Σ.

Typically, elements of Σ are encoded by strings of length $\log|\Sigma|$ and therefore (ii') implies (ii) with linear dependence on $\log|\Sigma|$. The AlgEcon results proved in prior work [15, 21, 11] are actually strong AlgEcon results for those classes of pairs of groups. Our AlgEcon result for abelian domain is also strong AlgEcon (see Section 3.3.3). On the other hand, our AlgEcon result for alternating domain does not meet the "strong" requirement.

Remark 3.1.4. The unit-cost model can also be used in the case of infinite Σ. In fact, our AlgEcon result for abelian domains holds even for infinite codomains in the unit-cost model; i. e., it satisfies (ii').

3.1.5 Certificate list-decoding

In the light of technical difficulties arising from algorithmic list-decoding, we introduce a new type of list-decoding that is intermediate between the combinatorial and algorithmic. We call it "certificate list-decoding." We shall refer to results of this type as "semi-algorithmic."

An $\Omega \rightharpoonup \Sigma$ *partial map* is a map $\gamma\colon\ \mathrm{dom}(\gamma) \to \Sigma$ where $\mathrm{dom}(\gamma) \subseteq \Omega$.

Definition 3.1.5 (Certificate)**.** We say that an $\Omega \rightharpoonup \Sigma$ partial map γ is a *certificate for the codeword* $\varphi \in \mathcal{C}$ if $\gamma = \varphi|_{\mathrm{dom}(\gamma)}$ (the restriction of φ to domain $\mathrm{dom}(\gamma)$) and φ is the unique codeword in $\mathcal{C}$ that extends γ. A *certificate* for the code $\mathcal{C}$ is a certificate for some codeword in $\mathcal{C}$.

Definition 3.1.6 (Certificate-list)**.** We say that a list Γ of $\Omega \rightharpoonup \Sigma$ partial maps is a *certificate-list* for the set $\mathcal{K} \subseteq \mathcal{C}$ of codewords if Γ contains a certificate for each codeword in $\mathcal{K}$. A

certificate-list for $\mathcal{C}$ *up to agreement* λ *of* the received word $f\colon \Omega \to \Sigma$ is a certificate-list for the list $\mathcal{L} = \mathcal{L}(\mathcal{C}, f, \lambda)$.

Remark 3.1.7. Note that we permit the certificate-list Γ to contain redundancies (more than one certificate for the same codeword) and irrelevant items (partial functions that are not certificates of any codeword in $\mathcal{K}$, or not even certificates of any codeword at all).

Definition 3.1.8. A *certificate-list-decoder* is an algorithm that, given the received word $f \in \Sigma^{\Omega}$ and the agreement λ, constructs a certificate-list of $\mathcal{C}$ up to agreement λ of f.

Definition 3.1.9. We say that $\mathcal{C}$ is a **CertEcon** ("certificate-economically list-decodable") code if there exists a local certificate-list-decoder with the following features.

Input: $\varepsilon > 0$, oracle access to $f \in \Sigma^{\Omega}$.

Notation: Again, let $\mathcal{L} = \mathcal{L}(\mathcal{C}, f, \Lambda_{\mathcal{C}} + \varepsilon)$.

Output: A list Γ of $\Omega \rightharpoonup \Sigma$ partial maps of length $|\Gamma| = \mathrm{poly}(1/\varepsilon)$.

Guarantee: With probability $\geq 3/4$, the output list Γ is a certificate-list for $\mathcal{L}$.

Cost:

(i) $\mathrm{poly}(\log|\Omega|, 1/\varepsilon)$ queries to the received word f.

(ii) $\mathrm{poly}(\log|\Omega|, \log|\Sigma|, 1/\varepsilon)$ amount of work.

Access: The meaning of this definition depends also on the access model to Σ and Ω. We shall clarify this in each application.

Remark 3.1.10. Note that $\Lambda_{\mathcal{C}}$ is not part of the input. In our results, we are likely to find a certificate of $\mathcal{C}$ up to agreement $\Lambda_{\mathcal{C}}$ of the received word f, regardless of the actual value of $\Lambda_{\mathcal{C}}$. We note that, depending on the access model, we may not be able to find $\Lambda_{\mathcal{C}}$.

Remark 3.1.11. CertEcon is intermediate between AlgEcon and CombEcon. Indeed, CertEcon implies CombEcon, by the length bound of the Output and the Guarantee. Moreover, AlgEcon implies CertEcon, as the AlgEcon Output $\widetilde{\mathcal{L}}$ satisfies the definition of a certificate, under the same Guarantee and Cost bound.

Strong CertEcon

We define *strong CertEcon* exactly analogously to *strong AlgEcon*.

Definition 3.1.12. We say that $\mathcal{C}$ is a **strong CertEcon** code if there exists a certificate-list-decoder satisfying the conditions of CertEcon, except with (ii) replaced by the following.

(ii') poly$(\log|\Omega|, 1/\varepsilon)$ amount of work in the unit-cost model for Σ.

All CertEcon results in this paper are actually strong CertEcon results.

Remark 3.1.13. Strong CertEcon does not follow from AlgEcon, though it does follow from strong AlgEcon.

Remark 3.1.14. As in the AlgEcon context, the unit-cost model can also be used in the case of infinite Σ. In fact, all our CertEcon results hold for infinite codomain in the unit-cost model; i. e., they satisfy (ii').

$\mathcal{W}$-certificates

We sometimes want to impose further conditions on the certificates that we allow in certificate lists. In particular, we will want certificates that are compatible with subword extenders; see Section 3.1.5.

Let $\mathcal{W}$ be a set of $\Omega \rightharpoonup \Sigma$ partial maps.

Definition 3.1.15 ($\mathcal{W}$-certificate)**.** A *$\mathcal{W}$-certificate* is a certificate that belongs to $\mathcal{W}$.

Definition 3.1.16 ($\mathcal{W}$-certificate-list)**.** We say that a list Γ of $\Omega \rightharpoonup \Sigma$ partial maps is a *$\mathcal{W}$-certificate-list* for the set $\mathcal{K} \subseteq \mathcal{C}$ of codewords if Γ contains a $\mathcal{W}$-certificate for each codeword in $\mathcal{K}$. A *$\mathcal{W}$-certificate-list* for $\mathcal{C}$ *up to agreement* λ of the received word $f\colon \Omega \to \Sigma$ is a $\mathcal{W}$-certificate-list for the list $\mathcal{L} = \mathcal{L}(\mathcal{C}, f, \lambda)$.

Remark 3.1.17. Note that, as mentioned in Remark 3.1.7, we permit the $\mathcal{W}$-certificate-list Γ to contain redundancies and irrelevant items, including partial functions γ that do not belong to $\mathcal{W}$.

Definition 3.1.18. A $\mathcal{W}$-certificate-list-decoder is an algorithm that, given the received word $f \in \Sigma^\Omega$ and the agreement λ, constructs a $\mathcal{W}$-certificate-list of $\mathcal{C}$ up to agreement λ with f.

Definition 3.1.19. We say that $\mathcal{C}$ is a **$\mathcal{W}$-CertEcon** ("$\mathcal{W}$-certificate-economically list-decodable") code if there exists a local $\mathcal{W}$-certificate-list-decoder with the features listed in Definition 3.1.9.

Definition 3.1.20. We say that $\mathcal{C}$ is a **strong $\mathcal{W}$-CertEcon** code if there exists a strong $\mathcal{W}$-certificate-list-decoder, i. e., a $\mathcal{W}$-certificate-list-decoder that is a strong certificate-list-decoder (see Definition 3.1.12).

Domain certificates

A natural idea for producing a certificate γ is to select a small subset $S \subseteq \Omega$, query f at the elements of S, and let γ be the restriction of f to S. This is the approach we take for list decoding homomorphism codes with alternating domains and (more generally) SRG domains in Section 3.8. We call such a restriction a "domain certificate."

Recall that the restriction $f|_S$ is the map $S \to \Sigma$ defined by $f|_S(x) = f(x)$ for $x \in S$.

Definition 3.1.21 (Domain certificate)**.** When the code $\mathcal{C} \subseteq \Sigma^\Omega$ and the received word f are understood, a *domain certificate* is a restriction $\gamma = f|_S$ of f to a subset $S \subseteq \Omega$ such that γ is a certificate for $\mathcal{C}$.

For a set $\mathcal{W}$ of $\Omega \rightharpoonup \Sigma$ partial maps, a *domain $\mathcal{W}$-certificate* is a domain certificate that is a $\mathcal{W}$-certificate for $\mathcal{C}$.

Computationally, we represent domain certificates $f|_S$ by their domain S.

Definition 3.1.22 (Domain-certificate-list)**.** A *domain-certificate-list* is a certificate-list consisting of domain certificates. We define *domain-$\mathcal{W}$-certificate-list* similarly.

Computationally, a domain-certificate-list can be represented as a list of subsets S of the domain, Ω.

Remark 3.1.23. In most cases in this paper, our certificates are domain certificates. For these cases, we do not even need access to the codomain. The exception is the case of homomorphism codes with abelian domain, where our algorithm is adapted from Dinur et al. [15]. In this case, we do need access to the codomain, and do not always return domain certificates.

Individual-certificate list-decoding

A natural way to produce a certificate-list is to run a process that randomly produces a single certificate, and then repeat this process many times. In light of this strategy, we define *individual-certificate list-decoding*, and *individual-certificate CertEcon*. We will see in Section 3.4.1 that individual-certificate CertEcon is an equivalent concept to (ordinary) CertEcon.

An *individual-certificate list-decoder* is an algorithm that, like a list-decoder, takes in a received word f and the agreement λ. Instead of returning a superset of the list $\mathcal{L}(\mathcal{C}, f, \lambda)$, it returns a single word γ. It must return each word in the list with a probability bounded away from zero.

Definition 3.1.24. We say that $\mathcal{C}$ is an **individual-certificate CertEcon** ("certificate-economically list-decodable") code if there exists an individual-certificate list-decoder with the following features.

Input: $\varepsilon > 0$, oracle access to $f \in \Sigma^{\Omega}$.

Notation: Again, let $\mathcal{L} = \mathcal{L}(\mathcal{C}, f, \Lambda_{\mathcal{C}} + \varepsilon)$.

Output: An $\Omega \rightharpoonup \Sigma$ partial map γ.

Guarantee: For every $\varphi \in \mathcal{L}$, with probability $\geq \varepsilon^{O(1)}$, the output γ is a certificate for φ.

Cost:

(i) $\text{poly}(\log|\Omega|, 1/\varepsilon)$ queries to the received word f.

(ii) $\mathrm{poly}(\log|\Omega|, \log|\Sigma|, 1/\varepsilon)$ amount of work.

Access: The meaning of this definition depends also on the access model to Σ and Ω. We shall clarify this in each application.

Definition 3.1.25. We say that $\mathcal{C}$ is a **strong individual-certificate CertEcon** code if there exists an algorithm satisfying the conditions of individual-certificate CertEcon, except with (ii) replaced by the following.

(ii') $\mathrm{poly}(\log|\Omega|, 1/\varepsilon)$ amount of work in the unit-cost model for Σ.

One can similarly define individual-certificate versions of list-decoding for domain certificates, $\mathcal{W}$-certificates, and domain-cW-certificates. Each such variation on individual-certificate CertEcon is equivalent to the corresponding (ordinary) CertEcon concept; see Section 3.4.1.

Subword extension

In this section we discuss the relationship between algorithmic and certificate list-decoding.

Definition 3.1.26 (Subword extension problem)**.** Let $\mathcal{C}$ be a code. The *subword extension problem* asks, given a partial map $\gamma\colon \Omega \rightharpoonup \Sigma$, whether γ extends to a codeword in $\mathcal{C}$.

A *subword extender* is an algorithm that answers this question and returns a codeword in $\mathcal{C}$ extending γ, if one exists.

For $\mathcal{W}$ a set of $\Omega \rightharpoonup \Sigma$ partial maps, the *$\mathcal{W}$-subword extension problem* asks to solve the subword extension problem on inputs from $\mathcal{W}$. A *$\mathcal{W}$-subword extender* is a subword extender that is only required to be correct on inputs from $\mathcal{W}$.

Observation 3.1.27. *A certificate-list-decoder and a subword extender combine to a list-decoder. A $\mathcal{W}$-certificate-list-decoder and a $\mathcal{W}$-subword extender combine to a list-decoder.*

Remark 3.1.28. This observation describes the two-phase plan to prove Theorem 1.4.11 in [6]. In the case of homomorphism codes, the subword extension problem corresponds to the *homomorphism extension problem* (see Section 1.4.2). The algorithmic difficulty of the homomorphism extension problem is a major bottleneck to further progress.

Note that restricting $\mathcal{W}$ to be a smaller set of partial maps makes $\mathcal{W}$-certificate-list-decoding more difficult, but makes $\mathcal{W}$-subword extension easier.

3.1.6 Mean-list-decoding

In this section we discuss *mean-list-decoding*, a tool for extending our economical list-decoding results to wider classes of domain groups. The metric used in mean-list-decoding is not distance to a single received word, but rather the average distance to a family of received words. Although apparently more general than list-decoding, mean-list-decoding is actually equivalent to list-decoding (in the CombEcon, CertEcon, and AlgEcon sense).

Let $\mathscr{F} = \{f_i : i \in I\}$ be a family of received words $f_i \in \Sigma^\Omega$. By the size $|\mathscr{F}|$ of $\mathscr{F}$ we mean the size $|I|$ of the index set I. We define the *average agreement* $\operatorname{agr}(w, \mathscr{F})$ of a word $w \in \Sigma^\Omega$ and $\mathscr{F}$ to be

$$\operatorname{agr}(w, \mathscr{F}) \coloneqq \mathop{\mathbb{E}}_{i \in I}[\operatorname{agr}(w, f_i)], \tag{3.4}$$

where the expectation $\mathbb{E}$ is taken with respect to the uniform distribution over I.

Definition 3.1.29 (Mean-lists)**.** For a code $\mathcal{C}$, we define the *mean list* $\mathcal{L}$ to be the set of codewords with at least a specified average agreement λ with the family $\mathscr{F}$ of received words, i. e.,

$$\mathcal{L} = \mathcal{L}(\mathcal{C}, \mathscr{F}, \lambda) \coloneqq \{w \in \mathcal{C} : \operatorname{agr}(w, \mathscr{F}) \geq \lambda\}. \tag{3.5}$$

The *maximum mean-list size* is

$$\mathfrak{m}\ell(\mathcal{C}, \lambda) \coloneqq \max_{\mathscr{F}} |\mathcal{L}(\mathcal{C}, \mathscr{F}, \lambda)| \tag{3.6}$$

For a positive integer r, the *maximum r-mean-list size* is the maximum taken only over families of size r, i. e.,

$$\mathfrak{m}_r\ell(\mathcal{C},\lambda) \coloneqq \max_{\mathscr{F}:|\mathscr{F}|=r} |\mathcal{L}(\mathcal{C},\mathscr{F},\lambda)| \tag{3.7}$$

We remark that $\ell(\mathcal{C},\lambda) = \mathfrak{m}_1\ell(\mathcal{C},\lambda)$, and $\mathfrak{m}\ell(\mathcal{C},\lambda) = \max_r \mathfrak{m}_r\ell(\mathcal{C},\lambda)$.

Mean-list-decoding and repeated codes

This concept of mean list was inspired by the use of repeated codes by Guo and Sudan [21].

Notation 3.1.30. For a word w, we denote by $w * r \coloneqq (\overbrace{w \dots w}^{r})$ the word found by concatenating r copies of w. For a set $\mathcal{S}$ of words, we write $\mathcal{S} * r \coloneqq \{w * r : w \in \mathcal{S}\}$.

Remark 3.1.31 (Mean-list-decoding versus repeated codes). Let $\mathscr{F} = \{f_i : i \in [r]\}$ be a family of r received words. Notice that $\mathcal{L}(\mathcal{C} * r, (f_1, \dots, f_r), \lambda)$ is the r-fold repetition of $\mathcal{L}(\mathcal{C},\mathscr{F},\lambda)$, i. e.,

$$\mathcal{L}(\mathcal{C},\mathscr{F},\lambda) * r = \mathcal{L}(\mathcal{C} * r, (f_1, \dots, f_r), \lambda). \tag{3.8}$$

It follows that $\mathfrak{m}_r\ell(\mathcal{C},\lambda) = \ell(\mathcal{C} * r, \lambda)$. In this way, mean-list-decoding can be viewed as list-decoding repeated codes.

Economical mean-list-decoding

We will define *CombEconM*, *CertEconM*, and *AlgEconM* by replacing the received word f with a family of received words $\mathscr{F}$ and replacing "the list" with "the mean-list" in the definitions of CombEcon, CertEcon, and AlgEcon, respectively. However, as we shall see the –EconM concepts turn out to be equivalent to the –Econ, concepts. The mean list-decoding concept helps expand the scope of our results, without making them more difficult to prove. We now make these terms precise.

Combinatorial. We wish to bound mean-list size by $|\mathcal{L}(\mathcal{C},\mathscr{F},\Lambda_{\mathcal{C}} + \varepsilon)| \le q'(\varepsilon)$ for some $q'(\varepsilon)$ that depends only on ε and the class $\mathscr{C}$ of codes under discussion ($\mathcal{C} \in \mathscr{C}$). We say that

the class $\mathscr{C}$ of codes is **CombEconM** ("combinatorially economically mean-list-decodable") if $q'(\varepsilon) = \text{poly}(1/\varepsilon)$ for $\mathcal{C} \in \mathscr{C}$.

Algorithmic. We say that the class $\mathscr{C}$ of codes is *AlgEconM* ("algorithmically economically mean-list-decodable") if it satisfies the definition of AlgEcon classes of codes, with the following modifications.

For each $\mathcal{C} \in \mathscr{C}$, the received word f is replaced by a family $\mathscr{F}$ of received words and the list $\mathcal{L}$ becomes $\mathcal{L} = \mathcal{L}(\mathcal{C}, \mathscr{F}, \lambda)$. Oracle access to $\mathscr{F}$ means that, given $i \in I$ and $\omega \in \Omega$, the oracle returns $f_i(\omega)$. Condition (ii) is replaced by the following.

(ii-M) $\text{poly}(\log|\Omega|, \log|\Sigma|, \log|\mathscr{F}|, 1/\varepsilon)$ amount of work.

Note that the number of queries to the family $\mathscr{F}$ remains $\text{poly}(\log|\Omega|, 1/\varepsilon)$.

Certificate. We say that a class $\mathscr{C}$ of codes is *CertEconM* ("certificate economically mean-list-decodable") if it satisfies the definition of CertEcon, with the same modifications as AlgEconM.

We will see that, for $X \in \{\text{Comb}, \text{Alg}, \text{Cert}\}$, a class $\mathscr{C}$ of codes is X-EconM if and only if it is X-Econ. For a more precise theorem statement, with proof, see Section 3.4.2.

Remark 3.1.32 (Significance of mean-list-decoding). Dinur et al. show the CombEcon and AlgEcon list-decodability of {abelian $\to$ abelian} homomorphism codes [15]. We shall see that Theorem 3.4.3 quickly leads to the conclusion of CombEcon list-decodability of {arbitrary $\to$ abelian} homomorphism codes. The same inference can be made about AlgEcon list-decodability, assuming natural conditions about representation of the domain group. See Section 3.4.4 for details.

Strong mean-list-decoding

We say that $\mathcal{C}$ is a *strong AlgEconM* code if it satisfies the definition of AlgEconM, except with (ii-M) replaced by (ii'-M) below. Similarly, we say that $\mathcal{C}$ is a *strong CertEconM* code if it satisfies the definition of CertEconM, except with (ii-M) replaced by (ii'-M) below.

(ii'-M) poly$(\log|\Omega|, 1/\varepsilon)$ amount of work in the unit-cost model for Σ and unit-sampling-cost model for $\mathscr{F}$.

In the *unit-sampling-cost model* for $\mathscr{F} = \{f_i : i \in I\}$, we charge unit cost for naming any $i \in I$ and for generating a uniform random $i \in I$.

3.2 Preliminaries

Let G be a set. For any subset $S \subseteq G$, define the *density* of S in G by $\mu_G(S) = \frac{|S|}{|G|}$. Call G the "ambient set" and write $\mu(S) = \mu_G(S)$ when G is understood. The ambient set will generally be a group G.

We use the notation $[n] = \{1, \ldots, n\}$.

3.2.1 Group-theoretic notation

Our group theory reference is [30]. We review some definitions and facts.

For G and M groups, we write $M \leq G$ to indicate that M is a subgroup of G. We write $M \trianglelefteq G$ to indicate that M is a normal subgroup of G. For a group H and $T \subseteq H$, the *centralizer* $C_H(T)$ consists of those elements of H that commute with all elements of T. For $T \subseteq H$ we write $\langle T \rangle$ to denote the subgroup generated by T (the smallest subgroup of G containing T).

For $M \leq G$, a *coset* of M is a set $aM \coloneqq \{ag : g \in M\}$, where $a \in G$. We refer to cosets of subgroups of G as *subcosets of* G. Let $|G : aM| \coloneqq |G : M|$ denote the index of M in G. A subset $K \subseteq G$ is *affine-closed* if $(\forall a, b, c \in K)(ab^{-1}c \in K)$. An affine-closed subset is either empty or it is a subcoset. The intersection of affine-closed subsets is affine-closed. The *affine*

closure $\langle S \rangle_{\mathsf{aff}}$, affinely generated by S, is the the smallest affine-closed subset containing S. Note that the affine closure of the empty set is empty. The affine closure of a nonempty set is a subcoset; indeed, for any $q \in S$, we have that $\langle S \rangle_{\mathsf{aff}} = q \cdot \langle q^{-1} r \mid r \in S \rangle$.

For a set Ω, the *symmetric group* $\mathrm{Sym}(\Omega)$ consists of all permutations of Ω. We write $S_n = \mathrm{Sym}([n])$. *Permutation groups acting on* Ω are subgroups of $\mathrm{Sym}(\Omega)$; their *degree* is $|\Omega|$. The set Ω is the *permutation domain*. The *alternating group* $A_n \leq S_n$ consists of the even permutations.

Let $G \leq \mathrm{Sym}(\Omega)$. For $\pi \in G$ and $x \in \Omega$, we denote by x^π the action of π on x. For $x \in \Omega$, denote by $G_x \coloneqq \{\pi \in G : x^\pi = x\}$ the point stabilizer of x. Let $\Delta \subseteq \Omega$. Denote by $G_{(\Delta)} \coloneqq \{\pi \in G : (\forall x \in \Delta)(x^\pi = x)\}$ the pointwise stabilizer of Δ. Denote by $G_{\{\Delta\}} \coloneqq \{\pi \in G : \Delta^\pi = \Delta\}$ the setwise stabilizer of Δ, where $\Delta^\pi \coloneqq \{x^\pi : x \in \Delta\}$.

In this paper we will denote the class of finite groups by $\mathfrak{Groups}_{\text{finite}}$, and the class of all groups (finite or infinite) by $\mathfrak{Groups}_{\text{finite or infinite}}$. We write $\mathfrak{Abel}$ to denote the class of finite abelian groups and $\mathfrak{Alt}$ for the class of (finite) alternating groups.

3.2.2 Homomorphism codes — Affine homomorphisms as codewords

Let G be a finite group and H a group. Denote the set of homomorphisms from G to H by $\mathrm{Hom}(G, H)$.

Definition 3.2.1. Let G_1 and H_1 be affine-closed subsets of G and H, resp. A function $\varphi\colon G_1 \to H_1$ is an *affine homomorphism* if

$$(\forall a, b, c \in G_1)(\varphi(a)\varphi(b)^{-1}\varphi(c) = \varphi(ab^{-1}c)) .$$

We write $\mathrm{aHom}(G_1, H_1)$ to denote the set of affine homomorphisms from G_1 to H_1.

Fact 3.2.2. Let $G_0 \leq G$ and $H_0 \leq H$. Let $a \in G$ and $b \in H$. A function $\varphi\colon aG_0 \to bH_0$ is

an affine homomorphism if and only if there exists $h \in H$ and $\varphi_0 \in \mathrm{Hom}(G_0, H_0)$ such that

$$\varphi(ag) = h \cdot \varphi_0(g) \tag{3.9}$$

for every $g \in G_0$. The element h and the homomorphism φ_0 are unique.

The analogous statement also holds with h on the right of $\varphi_0(g)$.

Definition 3.2.3. In Fact 3.2.2, call φ_0 the *base homomorphism* of φ.

Definition 3.2.4. For sets G, H and functions $f, g\colon G \to H$, the *equalizer* $\mathrm{Eq}(f, g)$ is the subset of G on which f and g agree, i. e.,

$$\mathrm{Eq}(f, g) \coloneqq \{x \in G \mid f(x) = g(x)\}. \tag{3.10}$$

More generally, if Φ is a collection of functions from G to H, then the *equalizer* $\mathrm{Eq}(\Phi)$ is the set

$$\mathrm{Eq}(\Phi) \coloneqq \{x \in G \mid (\forall f, g \in \Phi)(f(x) = g(x))\}. \tag{3.11}$$

Fact 3.2.5. (a) If $\varphi, \psi \in \mathrm{Hom}(G, H)$ then $\mathrm{Eq}(\varphi, \psi) \leq G$.

(b) If $\varphi, \psi \in \mathrm{aHom}(G, H)$ then $\mathrm{Eq}(\varphi, \psi)$ is affine-closed. Moreover, if $\varphi_0, \psi_0 \in \mathrm{Hom}(G, H)$ are the corresponding base homomorphisms (see Def. 3.2.3) then either $\mathrm{Eq}(\varphi, \psi)$ is empty or $\mathrm{Eq}(\varphi, \psi) = g \cdot \mathrm{Eq}(\varphi_0, \psi_0)$ for any $g \in \mathrm{Eq}(\varphi, \psi)$.

Recall that the (normalized) *agreement* $\mathrm{agr}(f, g)$ between two functions $f, g\colon G \to H$ is given by

$$\mathrm{agr}(f, g) \coloneqq \frac{|\mathrm{Eq}(f, g)|}{|G|}. \tag{3.12}$$

Specializing Def. 3.1.1 to homomorphism codes, we follow established notation and write

$$\Lambda_{G,H} \coloneqq \Lambda_{\mathrm{aHom}(G,H)} \tag{3.13}$$

for the *maximum agreement* between distinct elements of of $\mathrm{aHom}(G, H)$. In other words,

$$\Lambda_{G,H} = \max\{\mathrm{agr}(\varphi, \psi) \mid \varphi, \psi \in \mathrm{aHom}(G, H), \varphi \neq \psi\}. \tag{3.14}$$

When the groups G and H are understood, we often write Λ in place of $\Lambda_{G,H}$. Using this notation, the minimum distance of the homomorphism code $\mathrm{aHom}(G, H)$ is $(1 - \Lambda_{G,H})$.

The following statement appears in [20, Prop. 3.5]. We include the proof for completeness.

Proposition 3.2.6 (Guo)**.** *Let G, H be groups. The maximum agreement $\Lambda_{G,H}$ can equivalently be defined with* aHom *replaced by* Hom*, i. e.,*

$$\Lambda_{\mathrm{Hom}(G,H)} = \Lambda_{\mathrm{aHom}(G,H)}. \tag{3.15}$$

Here we use the convention that the maximum of the empty set (of nonnegative numbers) is zero. Otherwise we would need to make the additional assumption $|\mathrm{Hom}(G, H)| > 1$.

Proof. Let $\Lambda'_{G,H} = \Lambda_{\mathrm{Hom}(G,H)}$. Certainly $\Lambda_{G,H} \geq \Lambda'_{G,H}$. Now let $\varphi, \psi \in \mathrm{aHom}(G, H)$. Let $\varphi_0, \psi_0 \in \mathrm{Hom}(G, H)$ be the corresponding base homomorphisms (see Def. 3.2.3). By Fact 3.2.5, if $g \in \mathrm{Eq}(\varphi, \psi)$ then $\mathrm{Eq}(\varphi, \psi) = g \cdot \mathrm{Eq}(\varphi_0, \psi_0)$. Hence $\mathrm{agr}(\varphi, \psi)$ is either zero or equal to $\mathrm{agr}(\varphi_0, \psi_0)$, proving that $\Lambda_{G,H} \leq \Lambda'_{G,H}$. □

Corollary 3.2.7. *Let G be a finite group and H a group. Then, $\Lambda \leq \max\{\mu(K) \mid K \lneq G\}$, the largest density of a proper subgroup of G.*

Fact 3.2.8. Let G and H be groups and $S \subseteq G$ a subset. If $\varphi, \psi \in \mathrm{aHom}(G, H)$ and $\varphi(x) = \psi(x)$ for all $x \in S$, then $\langle S \rangle_{\mathsf{aff}} \subseteq \mathrm{Eq}(\varphi, \psi)$. □

Corollary 3.2.9. *Let G be a finite group, H a group, and $S \subseteq G$, such that $\mu(\langle S \rangle_{\mathsf{aff}}) > \Lambda$. If $\varphi, \psi \in \mathrm{aHom}(G, H)$ are such that $\varphi(x) = \psi(x)$ for all $x \in S$, then $\varphi = \psi$.* □

Remark 3.2.10. Fact 3.2.8 and Corollary 3.2.9 also hold if aHom is replaced by Hom and $\langle . \rangle_{\mathsf{aff}}$ is replaced by $\langle . \rangle$.

3.2.3 Certificate list-decoding for homomorphism codes

First we translate the concepts associated with certificate list-decoding (Section 3.1.5) to the context of homomorphism codes. A *certificate* γ is a $G \rightharpoonup H$ partial map that extends uniquely to an affine homomorphism $\varphi \in \mathrm{aHom}(G, H)$.

A *subword extender* is an algorithm that extends a $G \rightharpoonup H$ partial map to a full (affine) homomorphism if possible.

Recall that for a subset $S \subseteq G$, we denote by $\mu_G(S) \coloneqq |S|/|G|$ the *density* of S in G. Recall that for notational simplicity, we write Λ for $\Lambda_{G,H}$.

Notation 3.2.11. For $0 \leq \lambda \leq 1$, let $\mathcal{W}_\lambda$ (resp. $\mathcal{W}^{\mathrm{a}}_\lambda$) be the set of $G \rightharpoonup H$ partial maps γ such that $\mu(\langle \mathrm{dom}(\gamma) \rangle) > \lambda$ (resp. $\mu(\langle \mathrm{dom}(\gamma) \rangle_{\mathsf{aff}}) > \lambda$).

Observation 3.2.12. *If a partial map $\gamma \colon G \rightharpoonup H$ belongs to $\mathcal{W}^{\mathrm{a}}_\Lambda$, then γ extends to at most one affine homomorphism in* $\mathrm{aHom}(G, H)$.

Recall that we have introduced certificate list-decoding as an intermediate step towards algorithmic list-decoding, to address technical difficulties that arise in algorithmic list-decoding in the alternating case. The proof of Theorem 1.4.11 proceeds by applying Observation 3.1.27 with $\mathcal{W} = \mathcal{W}^{\mathrm{a}}_\Lambda$.

We find $\mathcal{W}_\Lambda$-certificates for homomorphism codes with SRG (and in particular, alternating) domain in Section 3.8.5.

3.2.4 Computational representations of groups and homomorphisms

In this section we discuss the models of access to groups required by our algorithms. The choice of the model significantly impacts the running time and even the feasibility of an algorithm.

The models include oracle models (black-box access, black-box groups), generator-relator presentations, and various explicit models. Commonly used explicit representations include

permutation groups, matrix groups, and representations of abelian groups such as the primary decomposition and the canonical form.

Recall that our domain groups are always finite but the codomain may be infinite (Convention 1.4.1).

Recall also that **homomorphisms** will be represented by the list of their values on a set of generators.

Our general reference to algorithmic group theory is [33].

Black-box models

If the codomain is infinite, and even if it is finite but very large, the black-box-group model with its fixed-length encoding [7] is not appropriate (see "encoded groups" below). We start with an extension of that model.

Definition 3.2.13 (Black-box access)**.** An *unencoded black-box representation* of a (finite or infinite) group K is an ordered 5-tuple

$$(U, r, \mathsf{mult}, \mathsf{inv}, \mathsf{id}) \tag{3.16}$$

where

- U is a (possibly infinite) set;
- $r\colon U \to K \cup \{*\}$ with $r(U) \supseteq K$;
- $\mathsf{mult}\colon r^{-1}(K) \times r^{-1}(K) \to r^{-1}(K)$ with $r(\mathsf{mult}(x, y)) = r(x)r(y)$ for all $x, y \in r^{-1}(K)$;
- $\mathsf{inv}\colon r^{-1}(K) \to r^{-1}(K)$ with $r(\mathsf{inv}(x)) = r(x)^{-1}$ for $x \in r^{-1}(K)$; and
- $\mathsf{id}\colon r^{-1}(K) \to \{\mathsf{yes}, \mathsf{no}\}$ with $\mathsf{id}(x) = \mathsf{yes}$ if and only if $r(x)$ is the identity in K.

We say that an algorithm has **black-box access** to the group K if the algorithm can store elements of U and query the functions (oracles) $\mathsf{mult}, \mathsf{inv}, \mathsf{id}$. We say that K is given as an

(unencoded) black-box group if in addition a list of generators of K is given.

Remark 3.2.14. We emphasize that the difference between *black-box access* to a group G and the group G being given as a *black-box group* is that in the latter model, a list of generators of G is given, whereas no elements of G may be a priori known in the former.

If $U = \{0,1\}^n$ then we talk about an **encoded group**, of **encoding length** n. This immediately implies that K is finite, namely, $|K| \leq 2^n$. (This is the model introduced in [7].)

In an abuse of notation, when black-box access to a group K is given, we may refer to elements of $r^{-1}(K)$ by their images under r, we may write gh in place of $\mathsf{mult}(g,h)$, we may write g^{-1} in place of $\mathsf{inv}(g)$, and we may write $g = 1$ in place of $\mathsf{id}(g) = \mathsf{yes}$.

Access to domain and codomain. In general we shall not need generators of the codomain, H, just black-box access. This is for instance the case for the {abelian $\rightarrow$ arbitrary} algorithmic result. For the purposes of producing domain-certificate-lists, we may not need access to H at all. Indeed, for SRG groups, we produce a domain-certificate list by choosing a list of random subsets of the domain G without ever looking at H. Of course, these subsets of the domain will have to be evaluated by the f-oracle.

On the other hand, we do need generators of the domain, G; homomorphisms will be defined by their values on a set of generators. So our access to the domain will be assumed to be at least as strong as an (encoded) black-box group.

The black-box unit cost model. The (unencoded) black-box access model is particularly well suited to the **unit-cost model** where we assume that we can copy and store an element of U and query an oracle at unit cost. We shall analyze our algorithms in the unit-cost model for the codomain H. This essentially counts the operations performed in H, so its bit-cost will incur an additional factor of $O(\log |H|)$ (if H is finite and nearly optimally encoded).

Random generation. In encoded black-box groups, independent nearly uniform random elements can be generated in time that is polynomial in the encoding length [3].

Remark 3.2.15. Black-box groups have been studied in a substantial body of literature, both in the theory of computing and in computational group theory (see the references in [5]). It is common to make additional access assumptions to a black-box group (assume additional oracles) such as an oracle for the order of the elements.

Given a black-box group H, we cannot determine the order $|H|$ or the order of a given element $h \in H$. In fact, even with an oracle for the order of elements, $\mathbb{Z}_p$ and $\mathbb{Z}_p \times \mathbb{Z}_p$ cannot be distinguished in fewer than $\Omega(\sqrt{p})$ randomized black-box identity queries. To avoid such obstacles, it is common to assume additional information beyond black-box access. In finding $\Lambda_{G,H}$ for abelian domain G one needs to decide if a given prime divides $|H|$. To accomplish this, we assume additional information about the group H such as the order $|H|$ or the list of primes dividing $|H|$.

Generator-relator presentation, homomorphism checking

By "presentation" of a group we mean *generator-relator presentation*.

For a group given by a presentation, basic questions, such as whether the group has order 1, are undecidable. However, special types of presentations, such polycyclic presentations of finite solvable groups, are often helpful. Note, however, that it is not known how to efficiently perform group operations in a finite solvable group given by a polycyclic presentation, so such presentations cannot answer basic black-box queries.

Any presentation, however, can be used for homomorphism checking, a critical operation in decoding homomorphism codes.

Proposition 3.2.16 (Homomorphism checking)**.** *Let $S \subseteq G$ be a list of generators of G. Assume a presentation of G is given in terms of S. Let $\varphi\colon S \to H$ be a function. Then φ extends to a homomorphism $\widetilde{\varphi}\colon G \to H$ if and only if the list $(\varphi(s) \mid s \in S)$ satisfies the relations.*

Note that this gives an efficient way to check whether φ extends to a homomorphism if

the relators are short or are given as short *straight-line programs*, assuming black-box access to the codomain.

Definition 3.2.17. Let G be a group and $S = (s_1, \ldots, s_\ell)$ a list of elements of G. A *straight-line program* in G from S to $g \in G$ is a sequence $P = (x_1, \ldots, x_m)$ of elements of G such that $x_m = g$, and each x_i is either a member of S, a product of the form $x_j x_k$ for some $j, k < i$, or x_j^{-1} for some $j < i$. We say that g is given in terms of S by the straight-line program P.

The following is well known.

Proposition 3.2.18. *Let $G \leq S_n$ be a permutation group and S a set of generators of G. Given S, a presentation of G in terms of S can be computed in polynomial time, where the relators returned are represented as straight-line programs.*

Abelian groups

For finite abelian groups, the representation used by prior authors is the *primary decomposition*, i. e., the representation as the direct product of cyclic groups of prime-power order.

The *canonical form* of finite abelian groups is the representation as the direct product of cyclic groups of orders $n_1, \ldots, n_k$ where $n_i \mid n_{i+1}$. Note that any *abelian presentation* in terms of generators and relations can be converted, in polynomial time, to the canonical form using the Smith normal form of integer matrices.

The canonical form alone will not suffice for the algorithms in prior work nor those in the present paper; one needs be able to factor the n_i in order to convert this to primary representation. This can be done, for instance, if *a superset of the prime divisors of the order of the finite abelian group G is available.*

3.2.5 Subgroup structure of alternating groups

We present a few useful structural results for alternating and symmetric groups. We use these to prove that alternating groups are universally CombEcon and CertEcon. We give two different CombEcon proofs, in Section 3.7.2 and Section 3.8.1. See Section 3.5 for a comparison of these proofs.

Subgroups of polynomial index in alternating groups are well understood; they are described by a result known as the Jordan–Liebeck Theorem (JLT), see [16, Theorem 5.2A].

Theorem 3.2.19 (Jordan-Liebeck)**.** *Let $n \geq 10$ and let r be an integer with $1 \leq r < n/2$. Suppose that $K \leq A_n$ has index $|A_n : K| < \binom{n}{r}$. Then, for some $\Delta \subseteq [n]$ with $|\Delta| < r$, we have $(A_n)_{(\Delta)} \leq K \leq (A_n)_{\{\Delta\}}$.*

We use Jordan-Liebeck in our first proof that alternating groups are universally CombEcon, Theorem 3.7.5. We also use it to bound the maximum agreement, Λ, for alternating domain in Lemma 3.7.4.

The following result [2, Theorem 1.5] says that two random elements of the symmetric group generate a "large" subgroup with high probability. This event is denoted $E(n, k)$ in the statement.

Theorem 3.2.20 (Babai)**.** *Let π, σ be a pair of independent uniform random elements from S_n. For $0 \leq k \leq n/3$, let $E(n, k)$ denote the following event: The subgroup $K = \langle \pi, \sigma \rangle$ acts as S_r or A_r on r elements of the permutation domain for some $r \geq n - k$. Then,*

$$\Pr(E(n,k)) = 1 - \binom{n}{k+1}^{-1} + O\left(\binom{n}{k+2}^{-1}\right). \tag{3.17}$$

The constant implied by the big-O notation is absolute.

We use Theorem 3.2.20 as part of the second proof that alternating groups are universally CombEcon, in Theorem 3.8.1. This also helps us show that alternating groups are universally CertEcon.

Remark 3.2.21. Suppose that we choose π and σ from A_n (instead of S_n) in Theorem 3.2.20. The same conclusion is still true. However, using only Theorem 3.2.20 as justification, the conclusion is slightly weaker – there will be a coefficient of 4 in front of $\binom{n}{k+1}^{-1}$. In our application, this coefficient makes no difference to our argument.

Remark 3.2.22. If $n \geq 7$ and $E(n,k)$ occurs in Theorem 3.2.20, then K has as a subgroup an alternating group acting on r elements of the permutation domain.

Both Jordan-Liebeck (Theorem 3.2.19) and Theorem 3.2.20 are statements about subgroups A_r or S_r of A_n or S_n. For our list-decoding results, we need to know that these subgroups have bounded depth.

Notation 3.2.23. For G a group, let $\mathfrak{l}(G)$ denote the length of the longest subgroup chain in G. Note that $\mathfrak{l}(G) = \mathrm{depth}_G(1)$ (see Definition 1.4.6).

Notation 3.2.24. For $n \in \mathbb{N}$, let b_n denote the number of ones in the binary expansion of n.

The following is known [13].

Theorem 3.2.25 (Cameron–Solomon–Turull)**.** $\mathfrak{l}(S_n) = \left\lfloor \dfrac{3n-1}{2} \right\rfloor - b_n.$

Proposition 3.2.26 (Babai)**.** *For $k < n/2$, we have that* $\mathrm{depth}_{S_n}(A_{n-k}) = \mathfrak{l}(S_k) + 2$.

Proof. See forthcoming note by Babai. □

Corollary 3.2.27. *For $k < n/2$, we have that* $\mathrm{depth}_{S_n}(A_{n-k}) = \left\lfloor \dfrac{3k+3}{2} \right\rfloor - b_k$. *Note that for $k \geq 2$, the right-hand side is at most $3k/2$.*

We use Corollary 3.2.27 in the proof of Theorem 3.7.5 and Theorem 3.8.1.

Remark 3.2.28. The slightly weaker bound $\mathrm{depth}_{S_n}(A_{n-k}) \leq 2k-1$ follows from Proposition 3.2.26 by the easier bound $\mathfrak{l}(S_n) \leq 2n-3$ [1]. This bound would suffice for our purposes.

3.3 Formal statements

3.3.1 List-decoding homomorphism codes

Let $\mathfrak{D}$ be a class of pairs (G, H) of groups. We say that $\mathfrak{D}$ is CombEcon if the class $\{\mathrm{aHom}(G, H) \mid (G, H) \in \mathfrak{D}\}$ of codes is CombEcon. We define CertEcon and AlgEcon classes of pairs of groups analogously.

Recall that we denote by $\mathfrak{Groups}_{\text{finite or infinite}}$ the class of all groups, finite or infinite. Recall that we say that a class $\mathfrak{G}$ of finite groups is *universally CombEcon* if $\mathfrak{G} \times \mathfrak{Groups}_{\text{finite or infinite}}$ is CombEcon. We define universally CertEcon and universally AlgEcon analogously, under access models to be specified.

A common feature of the prior work reviewed in Section 1.4.1 is that each class of pairs of groups considered was CombEcon and CertEcon. The present work continues to maintain this feature. In prior work, each class was also AlgEcon (which was almost equivalent to CertEcon in those cases). The present work shows AlgEcon in some cases.

All previously existing results put structural restrictions both on the domain and the codomain. In particular, they were restricted to subclasses of the solvable groups. In this paper we extend the economical list-decodability (both combinatorial and algorithmic) in the following three directions.

1. We give a general principle for removing certain types of constraints on the domain (see Section 3.3.2). It will follow that the previously known results extend to arbitrary domains.

2. We find universally economically list-decodable classes of groups. Specifically, abelian and alternating groups are universally CombEcon. Moreover, abelian groups are universally AlgEcon, and alternating groups are universally CertEcon, under modest access assumptions.

3. We exhibit the first (nontrivial) classes of examples where the domain is not solvable.

We note that no CombEcon bounds appear to be known for the much-studied classical linear codes (Reed–Solomon, Reed–Muller, BCH) (cf., e.g., [9]). The poly$(1/\varepsilon)$ CombEcon bound for Hadamard codes is quadratic [18]. For abelian and nilpotent groups, it currently has degree 105 [15, 21].

3.3.2 Extending the domain: the irrelevant kernel

In the prior work reviewed, both the domain and the codomain was abelian or close to abelian (nilpotent or supersolvable). It is natural to ask how to further relax the structural constraints on the groups involved.

We point out that structural constraints such as nilpotence or solvability (or any other hereditary property) play a very different role if imposed on the domain as on the codomain. For instance, a combinatorial list-decoding bound on {abelian $\to$ abelian} homomorphism codes implies the same bound for {arbitrary $\to$ abelian} homomorphism codes. This is shown by reducing list-decoding $\mathrm{aHom}(G, H)$ for arbitrary G and abelian H to mean-list-decoding $\mathrm{aHom}(G/G', H)$, where G' is the commutator subgroup of G, so G/G' is the largest abelian quotient of G. A similar argument extends the bounds for {nilpotent $\to$ nilpotent} homomorphism codes to {arbitrary $\to$ nilpotent} working through the largest nilpotent quotient of G. Similar results hold for certificate and algorithmic list-decoding.

In general, we can replace G by its *relevant quotient* G/N, where N is the irrelevant kernel (intersection of the kernels of all $G \to H$ homomorphisms), see Sec. 3.4.3.

While this observation extends the reach of the results of Dinur et al. [15] and Guo and Sudan [21], it also shows that, in a sense, the gains by extending the class of groups serving as the domains, without relaxing the structural constraints on the codomains, are *virtual*, and the main impediment to extending these results to wider classes of pairs of groups is the structural constraints on the *codomain*.

Our main contribution is the **elimination of all constraints on the codomain.**

This also opens up the question of meaningfully (as opposed to "virtually") **removing**

structural constraints on the domain side. Of particular interest becomes the case where the domain is a *finite simple group* and the codomain is arbitrary. We initiate this direction by studying the class of **alternating groups as domains.**

Definition 3.3.1 (Irrelevant kernel)**.** Let G and H be groups. The (G, H)-*irrelevant kernel* (or "irrelevant kernel" if G and H are clear) is the intersection of the kernels of all $G \to H$ homomorphisms, i. e.,

$$\bigcap_{\varphi \in \mathrm{Hom}(G,H)} \ker(\varphi). \tag{3.18}$$

We call elements and subgroups of the irrelevant kernel *irrelevant*.

For instance, if H is abelian, then the commutator subgroup G' is irrelevant.

Theorem 3.3.2. *Let N be an irrelevant normal subgroup of G. Then, $\Lambda_{G/N,H} = \Lambda_{G,H}$. Moreover,*

(i) if $\mathrm{aHom}(G/N, H)$ *is CombEcon then* $\mathrm{aHom}(G, H)$ *is CombEcon;*

(ii) if $\mathrm{aHom}(G/N, H)$ *is CertEcon then* $\mathrm{aHom}(G, H)$ *is CertEcon;*

(iii) if $\mathrm{aHom}(G/N, H)$ *is AlgEcon then* $\mathrm{aHom}(G, H)$ *is AlgEcon.*

For items (ii) and (iii) we need to make suitable assumptions on access to the domain.

For the proofs and discussion, see Section 3.4.3. The proofs rely on mean-list-decoding (Theorem 3.4.3).

Corollary 3.3.3. *The code* $\mathrm{aHom}(G, H)$ *is AlgEcon for any finite group G and any finite nilpotent group H.*

Proof. Combine Theorem 3.3.2 and the main result of [21]. For abelian H, use [15] instead. □

3.3.3 List-decoding: abelian → arbitrary

We state our main result about abelian domains.

Theorem 3.3.4. *If G is a finite abelian group, then G is universally CombEcon and universally strong AlgEcon.*

The degree of the poly$(1/\varepsilon)$ list-size bound is $C+4$ where C is the bound for {abelian → abelian} homomorphism codes (currently $C \approx 105$ [21]).

The proof of the CombEcon bound is based on the following structural result that says the range of all relevant homomorphisms is covered by a small number of finite abelian subgroups of H.

Theorem 3.3.5 (Structure of range)**.** *Let G be a finite abelian group, H an arbitrary group (finite or infinite), $f \in H^G$ a received word, and $\varepsilon > 0$. Then there exists a set $\mathcal{A}$ of finite abelian subgroups of H with $|\mathcal{A}| < \dfrac{1}{4(2\Lambda+\varepsilon)\varepsilon^2} + \dfrac{1}{\varepsilon}$ such that for all $\varphi \in \mathcal{L}(\mathrm{Hom}(G,H), f, \Lambda+\varepsilon)$, there is $M \in \mathcal{A}$ such that $\varphi(G) \le M$.*

Access model. We need to clarify how the algorithm accesses the domain and codomain. Following [15, 21, 11], we assume the domain is given explicitly in the primary decomposition (that is, as a direct product of cyclic groups of prime-power order). We remark that representing the domain in terms of a presentation by generators and abelian relations would suffice, if we are also given a superset of the prime divisors of the order of the domain. Without that additional information, factoring would be required (see Section 3.2.4). — We only require black-box access to the codomain (see Definition 3.2.13).

Pointer. We prove Theorems 3.3.4 and 3.3.5 in Section 3.6. The essential new result is the CombEcon bound, proved in Section 3.6.5. The algorithm is an adaptation of the algorithm of [15, 21], based on our CombEcon bound. This adaptation is discussed in Section 3.6.6.

3.3.4 Shallow random generation and list-decodability

We shall consider groups with the property that a bounded number of random elements tend to generate a subgroup of bounded depth (see Definitions 3.3.9 and 3.3.10 below). This class includes the alternating groups. We show that groups in this class are CombEcon, and under minimal assumptions on access they are also CertEcon.

It will be useful to consider an H-independent lower bound on the quantity $\Lambda_{G,H}$.

Definition 3.3.6. We define $\Lambda^*_G = \min\{\Lambda_{G,H} : \Lambda_{G,H} \neq 0, H \in \mathfrak{Groups}_{\text{finite or infinite}}\}$.

Observation 3.3.7. *For simple groups the following three quantities are equal: (a) Λ^*_G, (b) $\Lambda_{G,G}$, and (c) the largest fraction of elements of G fixed by an automorphism.*

Observation 3.3.8. *For $G = A_n$, $n \geq 5$, we have $\Lambda^*_G = 1/\binom{n}{2}$.*

Recall (Definition 1.4.6) that the *depth* of a subgroup M in a group G is the length d of the longest subgroup chain $M = M_0 < M_1 < \cdots < M_d = G$. We say that a subgroup is "shallow" if its depth is bounded. It follows from Theorem 3.2.20 [2] that a pair of random elements of A_n generates a subgroup of depth at most 6 with probability greater than $1 - O(n^{-5})$. This is the property that we generalize.

Definition 3.3.9 (Shallow random generation)**.** Let $k, d \in \mathbb{N}$. We say that a finite group G is (k, d)-*shallow generating* if

$$\Pr_{g_1,\ldots,g_k \in G}[\text{depth}_G(\langle g_1, \ldots, g_k\rangle) > d] < (\Lambda^*_G)^k. \tag{3.19}$$

Definition 3.3.10 (SRG groups)**.** We say that a class $\mathfrak{G}$ of finite groups has *shallow random generation* ($\mathfrak{G}$ is SRG) if there exist $k, d \in \mathbb{N}$ such that all $G \in \mathfrak{G}$ are (k, d)-shallow generating.

Lemma 3.3.11. *The alternating groups are SRG groups. In particular, for sufficiently large n, the alternating group A_n is $(2, 5)$-shallow generating.*

We prove this lemma in Section 3.8.1. We note that certain classes of Lie type simple groups are also SRG. We shall elaborate on this in a separate paper.

Now we can state one of the main results of this paper.

Theorem 3.3.12. *If G is an SRG group, then G is universally CombEcon.*

For the case of alternating groups, we show that the degree of the $\mathrm{poly}(1/\varepsilon)$ list-size bound is at most 8.

Theorem 3.3.13. *If G is an SRG group, then G is universally strong CertEcon.*

Recall that $\mathcal{W}^{\mathrm{a}}_{\Lambda}$ denotes the set of $G \rightharpoonup H$ partial maps γ such that $\mu(\langle \mathrm{dom}(\gamma)\rangle_{\mathsf{aff}}) > \Lambda_{G,H}$; see Section 3.2.3. In the above theorem, we can actually guarantee that the certificates come from the set $\mathcal{W}^{\mathrm{a}}_{\Lambda}$. Moreover, we can guarantee that all certificates are domain certificates (that is, the certificates are restrictions of the received word; see Section 3.1.5). Thus, we have the following.

Theorem 3.3.14 (SRG certificate, abridged)**.** *If G is an SRG group, then G is universally strong $\mathcal{W}^{\mathrm{a}}_{\Lambda}$-CertEcon via domain certificates.*

More precisely, we have the following.

Theorem 3.3.15 (SRG certificate, unabridged)**.** *Let G be a (k,d)-shallow generating group and H an arbitrary group. We have a local algorithm with the following features.*

Input: Values $\varepsilon, \eta > 0$.

Output: A set $\Pi \subseteq G^{k+d+1}$ of $(k+d+1)$-tuples in G, where

$$|\Pi| \leq \left\lceil \frac{1}{\varepsilon^{k+d+1}} \ln\left(\frac{1}{\eta\varepsilon^{k+d+1}}\right)\right\rceil. \tag{3.20}$$

Cost: $\mathrm{poly}(1/\varepsilon, \log(1/\eta))$ amount of work.

Performance guarantee: For every received word $f \in H^G$, with probability at least $(1-\eta)$, the set $\Gamma \coloneqq \{f|_S : S \in \Pi\}$ is a domain-$\mathcal{W}^{\mathrm{a}}_{\Lambda}$-certificate-list for $\mathrm{aHom}(G,H)$ up to distance $(\mathsf{mindist} - \varepsilon)$ of f.

Access model. For the CertEcon results, we assume access to (nearly) uniform random elements of the domain. We do not multiply elements of the domain, so we do not need black-box access to the domain. However, representing the domain as an encoded black-box group suffices for random generation (see Sec. 3.2.4).

We need no access to the codomain. We get ahold of elements of the codomain by querying the received word. We shall not perform any group operations in the codomain.

Pointers. We prove the CombEcon result in Section 3.8.4 and the CertEcon result in Section 3.8.5. For alternating groups we also give another, non-algorithmic, proof of the CombEcon result in Section 3.7. That proof relies on a generic sphere-packing argument to split the sphere into more tractable bins (see Lemma 3.5.7 and Section 3.7.1).

Remark 3.3.16. Given that A_n is $(2, 5)$-shallow generating (Lemma 3.3.11), Theorem 3.3.15 applies to A_n with $k + d + 1 = 8$. We think of A_n being given in its natural permutation representation. We note that a representation of A_n as a black-box group would suffice, because the natural permutation representation of an alternating group can be efficiently extracted from a black-box group representation [8].

3.4 Reductions

In [15], Dinur et al. show that if one can algorithmically list-decode $\mathrm{aHom}(G, \mathbb{Z}_{p^r})$ where G is a finite abelian group, p is prime, and r is an integer, then one can list decode $\mathrm{aHom}(G, H)$, where G and H are both abelian groups. They proceed to the more involved task of showing how to list decode $\mathrm{aHom}(G, \mathbb{Z}_{p^r})$.

In this section we prove three reductions: we give general settings where a list-decoding result immediately implies a stronger result in a broader setting. These principles apply to both combinatorial and algorithmic list-decoding. These principles are as follows.

(1) The *mean-list to list reduction*: If a homomorphism code is list-decodable (in the combinatorial, certificate, or algorithmic sense), then it is mean-list-decodable.

(2) The *domain relaxation principle*: If $\mathfrak{G}$ is a class of finite groups closed under subgroups and direct products (a *quasivariety of finite groups*), and $\mathfrak{G} \times \mathfrak{G}$ is list-decodable, then $\mathfrak{Groups}_{\text{finite}} \times \mathfrak{G}$ is list-decodable.

(3) The *reduction of affine to ordinary homomorphisms*: If $\text{Hom}(G, H)$ is list-decodable, then $\text{aHom}(G, H)$ is list-decodable.

The (semi-)algorithmic versions of these principles are proven using the equivalence of CertEcon and individual-certificate CertEcon (Corollary 3.4.2, proven in Section 3.4.1). This lemma will be used again in our certificate list-decodability proof for SRG groups. In Section 3.4.2, we prove the mean-list to list reduction. In Section 3.4.3, we use the mean-list to list reduction to prove a theorem about irrelevant normal subgroups. In Section 3.4.4, we use this theorem prove the domain relaxation principle. In Section 3.4.5, we prove the reduction of affine to ordinary homomorphisms.

Another reduction principle, the small Λ lemma (Lemma 3.5.9), is proven in Section 3.5.3 using the technique of strong negative correlation, which is introduced in that section.

3.4.1 Equivalence of CertEcon and individual-certificate CertEcon

In this Section, we show that a class of codes is individual-certificate CertEcon (see Section 3.1.5 for the definition) if and only if it is (ordinary) CertEcon. Similar equivalences hold for many variations on CertEcon.

The following lemma allows us to construct a certificate-list by repeatedly running an algorithm that outputs a single partial map.

Lemma 3.4.1. *Let $\delta > 0$. Let $0 < \eta < 1$. Let A be a set. Let $N = |A|$. Let $s = \lceil (1/\delta) \ln(N/\eta) \rceil$. Let $X = \{x_1, \ldots, x_s\}$ such that the x_i are chosen independently at random from $A \cup \{\star\}$. Suppose that for each $a \in A$ and each $1 \leq i \leq s$, we have that $x_i = a$ with probability at least δ. Then, the probability that $X \supseteq A$ is at least η.*

Proof. For each $a \in A$, the probability that $a \notin X$ is $(1-\delta)^s$, so by the union bound, the probability that $X \not\supseteq S$ is at most

$$N(1-\delta)^s \leq Ne^{-\delta s} \leq \eta. \tag{3.21}$$

□

Corollary 3.4.2. *A class of codes is CertEcon if and only if it is individual-certificate CertEcon. The access model is the same for both algorithms. A class of codes is strong-CertEcon if and only if it is strong individual-CertEcon.*

If the one algorithm returns domain certificates, $\mathcal{W}$-certificates, or domain $\mathcal{W}$-certificates, then so does the other algorithm. Moreover, if one algorithm returns not just certificates but full codewords, then so does the other.

Proof. Let $\mathscr{C}$ be a class of codes.

Suppose $\mathscr{C}$ is CertEcon. An individual-certificate CertEcon algorithm is as follows: Run the CertEcon algorithm to get a list of partial maps. Return a random member of this list.

Suppose $\mathscr{C}$ is individual-certificate CertEcon, and let $c \geq 0$ such that guaranteed probability lower bound is ε^c. Let $s = \left\lceil \left(\frac{1}{\varepsilon}\right)^c \left(c \ln \frac{1}{\varepsilon} + \ln 4\right) \right\rceil$. A CertEcon algorithm is to run the individual-CertEcon algorithm s times, and return the length-s list of partial maps produced. This is in fact a CertEcon algorithm by Lemma 3.4.1 with $A = \mathcal{L} = \mathcal{L}(\mathcal{C}, f, \Lambda + \varepsilon)$, with $\eta = 3/4$, and with $\delta = \varepsilon^c$. There, $x_1, \ldots, x_s$ represent the elements of $\mathcal{L}$ for which we have certificates. □

In the remainder of this section, when we give proofs regarding CertEcon, we will use the individual-certificate CertEcon formulation.

3.4.2 Reduction of mean-list decoding to list decoding

In this section we discuss results that apply to all codes, not just to homomorphism codes.

In this section, we discuss mean-list-decoding. Refer to Section 3.1.6 for the relevant definitions.

The main result of this section is the mean-list to list reduction, Theorem 3.4.3, which says that –EconM concepts are equivalent to the corresponding –Econ concepts. For more precise statements, see Lemma 3.4.7 and Corollary 3.4.8. For a code $\mathcal{C}$, a family $\mathscr{F}$ of received words, and $\lambda, \tau > 0$, one can recover the mean-list $\mathcal{L}(\mathcal{C}, \mathscr{F}, \lambda + \tau)$ by finding the list $\mathcal{L}(\mathcal{C}, \mathscr{F}, \lambda)$ for a random selection of $f \in \mathscr{F}$, and taking the union of these lists.

Theorem 3.4.3 (Mean-list to list reduction)**.** *Let $\mathscr{C}$ be a class of codes.*

(i) $\mathscr{C}$ is CombEconM if and only if it is CombEcon.

(ii) Under suitable access assumptions (Access 3.4.4), $\mathscr{C}$ is CertEconM if and only if it is CertEcon.

(iii) Under suitable access assumptions (Access 3.4.4), $\mathscr{C}$ is AlgEconM if and only if it is AlgEcon.

The same results hold for the strong versions of these concepts.

The deterioration in cost is as described in Remark 3.4.5.

Access 3.4.4. An oracle provides uniform random elements of the index set I of the family $\mathscr{F} = \{f_i : i \in I\}$ of received words.

Remark 3.4.5. That –EconM implies –Econ is immediate, with no deterioration in the cost.

The bounds on cost deteriorate as follows when going from –Econ to –EconM.

- An $O(1/\varepsilon)$ multiplicative factor in list size.
- An $O(\frac{1}{\varepsilon}\log(1/\varepsilon))$ multiplicative factor in queries to the received word f.
- An $O(\frac{1}{\varepsilon}\log(1/\varepsilon))$ multiplicative factor in amount of work.

The mean-list to list reduction is a consequence of Lemma 3.4.7 and Corollary 3.4.8, along with Lemma 3.4.1.

The proof of Lemma 3.4.7 uses the following observation.

Lemma 3.4.6 (Markov degradation)**.** *Fix a codeword φ. Let $\mathscr{F} = \{f_i : i \in I\}$ be a family of received words in the code space. Assume $\mathbb{E}_i[\mathrm{agr}(\varphi, f_i)] \geq \lambda + \delta$. Then $\Pr_i[\mathrm{agr}(\varphi, f_i) > \lambda] > \delta$.*

Proof. Let $x_i = \mathrm{dist}(\varphi, f_i) = 1 - \mathrm{agr}(\varphi, f_i)$. Then $\mathbb{E}_i[x_i] \leq 1 - \lambda - \delta$. Therefore, by Markov's inequality, $\Pr[\mathrm{agr}(\varphi, f_i) \leq \lambda] = \Pr[\mathrm{dist}(\varphi, f_i) \geq 1 - \lambda] \leq \frac{1-\lambda-\delta}{1-\lambda} = 1 - \frac{\delta}{1-\lambda} < 1 - \delta$. □

The following lemma shows how to turn a certificate list-decoder into a mean-certificate list-decoder.

Lemma 3.4.7. *Let $\mathcal{C}$ be a code, $\lambda, \tau, \delta > 0$. Let $\mathscr{F} = \{f_i : i \in I\}$ be a family of received words.*

Let i be a uniform random element of I. Let γ be a random partial map such that for each $\varphi \in \mathcal{L}(\mathcal{C}, f_i, \lambda)$, we have that γ is a certificate for φ with probability $\geq \delta$.

Then, for each $\varphi \in \mathcal{L}(\mathcal{C}, \mathscr{F}, \lambda + \tau)$, we have that γ is a certificate for φ with probability $\geq \tau\delta$.

Proof. By Markov degradation (Lemma 3.4.6), γ is a certificate for φ with probability

$$\Pr[\gamma \text{ certif. for } \varphi] \geq \Pr[\gamma \text{ certif. for } \varphi \mid \mathrm{agr}(f, \varphi) \geq \lambda + \tau] \cdot \Pr[\mathrm{agr}(f, \varphi) \geq \lambda + \tau] \tag{3.22}$$

$$\geq \delta \cdot \tau. \tag{3.23}$$

□

As a corollary, we get a bound on mean-list size.

Corollary 3.4.8. *Let $\mathcal{C}$ be a code, $\lambda, \tau > 0$. Let $\mathscr{F} = \{f_i \mid i \in I\}$ be a family of received words. Let $\mathcal{L} = \mathcal{L}(\mathcal{C}, \mathscr{F}, \lambda + \delta)$. Then,*

$$|\mathcal{L}| \leq \ell(\mathcal{C}, \lambda)/\tau. \tag{3.24}$$

Proof. Apply Lemma 3.4.7, with γ chosen uniformly from among the $\leq \ell(\mathcal{C}, \lambda)$ elements of $\mathcal{L}(\mathcal{C}, f_i, \lambda)$. Note that γ can be a certificate for at most one $\varphi \in \mathcal{L}$. □

Proof of Theorem 3.4.3. A code that is –EconM is –Econ. We only need to show the other direction.

(i) Apply Corollary 3.4.8 with $\lambda = \Lambda_{\mathcal{C}} + \varepsilon/2$ and $\tau = \varepsilon/2$.

(ii) Given an individual-certificate CertEcon algorithm $\mathcal{A}$ for $\mathcal{C}$, we construct an individual-certificate CertEconM algorithm for $\mathcal{C}$. This is sufficient by Corollary 3.4.2.

On input $\mathscr{F} = \{f_i : i \in I\}$, ε, let i be a uniform random element of I, and return $\mathcal{A}(f, \varepsilon/2)$. This is an individual-certificate CertEconM algorithm by Lemma 3.4.7.

(iii) Similar to (ii). □

Remark 3.4.9. Knowledge of $\Lambda_{\mathcal{C}}$ is not needed in the conversion from CertEcon to CertEconM. Even in the AlgEcon case, $\Lambda_{\mathcal{C}}$ is only needed by the mean-list-decoder if required by the list-decoder. The crucial knowledge for this conversion is ε, so that the deterioration factor (denoted τ above) can be controlled. This deterioration factor is set to $\tau = \varepsilon/2$ in our proofs.

3.4.3 Irrelevant normal subgroups

In Section 3.4.4 we present the domain relaxation principle (Theorem 3.4.21), which allows us to lift constraints on the domain when list-decoding many classes of homomorphism codes. An example is the automatic extension of {abelian → abelian} results to the {arbitrary → abelian} context (Corollary 3.4.23; see the discussion in Remark 3.1.32.)

The key concept is the (G, H)-*irrelevant kernel* N for a pair (G, H) of groups, defined as the intersection of the kernels of all $G \to H$ homomorphisms (see Definition 3.3.1). We shall find that economical list-decodability results with domain group G/N extend to results with domain G.

In this section, we show that $\mathrm{aHom}(G, H)$ is X-Econ (for $X \in \{\mathrm{Comb}, \mathrm{Alg}, \mathrm{Cert}\}$) if and

only if aHom$(G/N, H)$ is, where N is the (G, H)-irrelevant kernel, or any (G, H)-irrelevant normal subgroup. We state this as Theorem 3.4.10. For more precise statements, see Corollary 3.4.19 and Lemma 3.4.20. This will be the main tool in proving the domain relaxation principle, Theorem 3.4.21.

Theorem 3.4.10. *Let G, H and N be finite groups such that N is a (G, H)-irrelevant normal subgroup of G.*

(i) If aHom$(G/N, H)$ *is CombEcon, then* aHom(G, H) *is CombEcon.*

(ii) Under suitable access assumptions (Access 3.4.11 (ii)), if aHom$(G/N, H)$ *is CertEcon, then* aHom(G, H) *is CertEcon.*

(iii) Under suitable access assumptions (Access 3.4.11 (iii)), if aHom$(G/N, H)$ *is AlgEcon, then* aHom(G, H) *is AlgEcon.*

The deterioration in cost is as described in Remark 3.4.5.

Access 3.4.11. (ii) (a) Elements of N can be generated uniformly. (b) A transversal, i. e., an injection $\tau\colon G/N \to G$ that assigns a representative element to each coset, is given. (c) G/N is known well enough to satisfy the CertEcon access assumptions on aHom$(G/N, H)$.

(iii) (a') Elements of N can be generated uniformly and generators for N are given. (b') Same as (b). (c') Same as (c), for AlgEcon.

Remark 3.4.12. If the access assumptions on N are at least as strong as having N as a black-box group, then generating (nearly) uniform elements and being given a set of generators are equivalent. If N is a black-box group, generators are given by definition. Nearly uniform random elements in black-box groups can be generated in polynomial time (polynomial in the encoding length of group elements).

Remark 3.4.13. For the proof of this theorem, we will actually need the –EconM versions of the assumptions, which we may assume as a consequence of Theorem 3.4.3.

Let G be a finite group, H a group, and N a (G,H)-irrelevant normal subgroup of G. Recall that N is (G,H)-irrelevant if N is contained in the kernel of every $G \to H$ homomorphism (see Definition 3.3.1). We note two bijective correspondences, each relating maps $G \to H$ to maps $G/N \to H$. Both of the correspondences preserve the notion of agreement. These two correspondences, along with the mean-list to list reduction (Theorem 3.4.3), are used to prove Theorem 3.4.10.

- There is a bijective correspondence between affine homomorphisms $G \to H$ and affine homomorphisms G/N to H (see Observation 3.4.14).
- There is a bijective correspondence between functions $G \to H$, and size-$|N|$ families of functions $G/N \to H$ (see Observation 3.4.16).

We now describe these correspondences more precisely.

Let $\pi\colon G \to G/N$ be the projection onto cosets.

Observation 3.4.14. *The map* $\mathrm{aHom}(G/N,H) \to \mathrm{aHom}(G,H)$ *given by* $\varphi \mapsto \varphi \circ \pi$ *is a bijection. And, for* $\varphi, \psi \in \mathrm{aHom}(G/N)$*, we have that* $\mathrm{agr}(\varphi,\psi) = \mathrm{agr}(\varphi \circ \pi, \psi \circ \pi)$. □

Corollary 3.4.15. $\Lambda_{G/N,H} = \Lambda_{G,H}$. □

Recall that H^G is the set of all maps $G \to H$. Let $\mathscr{X}$ be the set of all families $\mathscr{F} = \{q_n : n \in N\}$ where $q_n\colon G/N \to H$. Let S be a set of coset representatives of N in G.

We define a map $\beta\colon H^G \to \mathscr{X}$. Consider any $f\colon G \to H$. For $n \in N$, let $\beta_n f\colon G/N \to H$ by $\beta_n f(sN) = f(sn)$ for all $s \in S$. Let $\beta f = \{\beta_n f : n \in N\}$.

Observation 3.4.16. *The map* β *defined above is a bijection.* □

Theorem 3.4.17. *(a) For all* $q\colon G/N \to H$*, we have that* $\beta(q \circ \pi) = \{q : n \in N\}$.

(b) For all $q\colon G/N \to H$ *and* $f\colon G \to H$*, we have that* $\mathrm{agr}(q \circ \pi, f) = \mathrm{agr}(q, \beta f)$.

Proof. (a) Let $q_n = \beta_n(q \circ \pi)$. For $s \in S$, we have that $q_n(sN) = q(sn) = (q \circ \pi)(sn) = q(sN)$, so $q_n = q$. So, $\beta(q \circ \pi) = \{q : n \in N\}$.

(b) Let $f_n = \beta_n f$. We have that

$$\begin{aligned}
\operatorname{agr}(q, \beta f) &= \mathop{\mathbb{E}}_{n\in N}[\operatorname{agr}(q, \beta_n f)] = \mathop{\mathbb{E}}_{n\in N}\mathop{\Pr}_{s\in S}[q(sN) = \beta_n f(sN)] && (3.25)\\
&= \mathop{\mathbb{E}}_{n\in N}\mathop{\Pr}_{s\in S}[(q\circ\pi)(sn) = f(sn)] = \mathop{\Pr}_{n\in N, s\in S}[(q\circ\pi)(sn) = f(sn)] && (3.26)\\
&= \mathop{\Pr}_{g\in G}[(q\circ\pi)(g) = f(g)] = \operatorname{agr}(q\circ\pi, f). && (3.27)
\end{aligned}$$

□

Corollary 3.4.18. *Let λ be a real number. Then,*

$$\ell(\mathrm{aHom}(G, H), \lambda) = \mathfrak{m}_{|N|}\ell(\mathrm{aHom}(G/N, H), \lambda). \tag{3.28}$$

Proof. Let $f\colon G \to H$ be a received word. For $\varphi \in \mathrm{aHom}(G/N, H)$,

$$\begin{aligned}
\varphi\circ\pi \in \mathcal{L}(\mathrm{aHom}(G, H), f, \lambda) &\iff \operatorname{agr}(\varphi\circ\pi, f) > \lambda \iff \operatorname{agr}(\varphi, \beta f) > \lambda && (3.29)\\
&\iff \varphi \in \mathcal{L}(\mathrm{aHom}(G/N, H), \beta f, \lambda). && (3.30)
\end{aligned}$$

To finish the proof, we note that every affine homomorphism in $\mathrm{aHom}(G, H)$ can be written (uniquely) as $\varphi\circ\pi$ for some $\varphi \in \mathrm{aHom}(G/N, H)$, and every size-$|N|$ family of maps $G/N \to H$ can be written as βf for some $f\colon G \to H$. □

Corollary 3.4.19 (Irrelevant normal subgroup lemma)**.** *For $\varepsilon > 0$,*

$$\ell(\mathrm{aHom}(G, H), \Lambda_{G,H} + \varepsilon) \le \frac{2}{\varepsilon}\cdot\ell(\mathrm{aHom}(G/N, H), \Lambda_{G,H} + \varepsilon/2). \tag{3.31}$$

Proof. We calculate

$$\ell(\mathrm{aHom}(G,H),\Lambda+\varepsilon) = \mathfrak{m}_{|N|}\ell(\mathrm{aHom}(G/N,H),\Lambda+\varepsilon) \qquad \text{Corollary 3.4.18} \qquad (3.32)$$

$$\leq \mathfrak{m}\ell(\mathrm{aHom}(G/N,H),\Lambda+\varepsilon) \qquad \text{Definition of } \mathfrak{m}\ell \qquad (3.33)$$

$$\leq \frac{2}{\varepsilon}\ell(\mathrm{aHom}(G/N,H),\Lambda+\varepsilon/2). \qquad \text{Corollary 3.4.8} \qquad (3.34)$$

□

The following lemma shows how to turn a certificate mean-list-decoder for $\mathrm{aHom}(G/N,H)$ into a certificate list-decoder for $\mathrm{aHom}(G,H)$.

Lemma 3.4.20. *Let G, H and N be finite groups such that N is a (G,H)-irrelevant normal subgroup of G. Let $\mathcal{C}$ be a code, $\lambda,\delta>0$. Let $f\colon G\to H$ be a received word.*

Let γ be a random $G/N \rightharpoonup H$ partial map such that for each $\varphi\in\mathcal{L}(\mathrm{aHom}(G/N,H),\beta f,\lambda)$, we have that γ is a certificate for φ with probability $\geq\delta$. Let $\widetilde{\gamma}$ be the $G\rightharpoonup H$ partial map defined by $\widetilde{\gamma}(\tau(g))=\gamma(g)$ for $g\in\mathrm{dom}(\gamma)$.

Then, for each $\varphi\in\mathcal{L}(\mathrm{aHom}(G,H),f,\lambda)$, we have that $\widetilde{\gamma}$ is a certificate for φ with probability $\geq\delta$.

Proof. Apply Observation 3.4.14 and Theorem 3.4.17 (b). □

We can now prove Theorem 3.4.10.

Proof of Theorem 3.4.10. Let $\Lambda=\Lambda_{G,H}=\Lambda_{G/N,H}$. Let $\tau\colon G/N\to G$ be the transversal from Access 3.4.11.

(i) Follows from Corollary 3.4.19.

(ii) Given an individual-certificate CertEconM algorithm $\mathcal{A}$ for $\mathrm{aHom}(G/N,H)$, we construct an individual-certificate CertEcon algorithm for $\mathrm{aHom}(G,H)$. This is sufficient by Corollary 3.4.2 and Theorem 3.4.3.

On input $f\colon G\to H$, $\varepsilon>0$, let $\gamma=\mathcal{A}(\beta f,\varepsilon)$. Let $\widetilde{\gamma}$ be as in Lemma 3.4.20. Return $\widetilde{\gamma}$. This is an individual-certificate CertEcon algorithm by Lemma 3.4.20.

(iii) We apply the same strategy as in part (ii), but need to change the definition of $\widetilde{\gamma}$ to make sure that the affine closure of its domain is G. Let X be the set of generators for N guaranteed by Access 3.4.11. Let $a \in \mathrm{dom}(\gamma)$. We define the $G \rightharpoonup H$ partial map $\widetilde{\gamma}$ by letting $\widetilde{\gamma}(\tau(g)) = \gamma(g)$ for $g \in \mathrm{dom}(\gamma)$, and letting $\widetilde{\gamma}(ax) = \gamma(a)$ for $x \in X$. □

3.4.4 The domain relaxation principle

We say that a class $\mathfrak{G}$ of finite groups is a *quasivariety of finite groups* if it is closed under subgroups and direct products.[1] Examples of quasivarieties include abelian groups, nilpotent groups, supersolvable groups, and solvable groups.

Theorem 3.4.21 (Domain relaxation principle)**.** *Let $\mathfrak{G}$ be a quasivariety of finite groups.*

(i) Suppose $\mathrm{aHom}(G, H)$ *is CombEcon for* $G, H \in \mathfrak{G}$*. Then,* $\mathrm{aHom}(G, H)$ *is CombEcon for* $G \in \mathfrak{Groups}_{\mathrm{finite}}$ *and* $H \in \mathfrak{G}$*.*

(ii) Under suitable access assumptions (Access 3.4.22 (ii)), if $\mathrm{aHom}(G, H)$ *is CertEcon for* $G, H \in \mathfrak{G}$*, then* $\mathrm{aHom}(G, H)$ *is CertEcon for* $G \in \mathfrak{Groups}_{\mathrm{finite}}$ *and* $H \in \mathfrak{G}$*.*

(iii) Under suitable access assumptions (Access 3.4.22 (ii)), if $\mathrm{aHom}(G, H)$ *is AlgEcon for* $G, H \in \mathfrak{G}$*, then* $\mathrm{aHom}(G, H)$ *is AlgEcon for* $G \in \mathfrak{Groups}_{\mathrm{finite}}$ *and* $H \in \mathfrak{G}$*.*

The deterioration in cost is as described in Remark 3.4.5.

The access assumptions mirror those of Access 3.4.11 of Theorem 3.4.10.

Access 3.4.22. For every $G \in \mathfrak{Groups}_{\mathrm{finite}}$ and $H \in \mathfrak{G}$, denote by N the (G, H)-irrelevant kernel and assume we have access to N and G/N as follows.

1. According to the standard definition, a *quasivariety of groups* is a class of groups that is closed under subgroups and reduced products. One can show that a class of finite groups is a "quasivariety of finite groups" in our sense if and only if it consists of the finite groups in some quasivariety of groups, justifying our terminology.

(ii) (a) Random elements of N can be generated uniformly. (b) A transversal of G/N in G can be found. (c) G/N can be found well enough to satisfy the access model of the assumed CertEcon list-decodability of pairs in $\mathfrak{G} \times \mathfrak{G}$.

(iii) (a') Random elements of N can be generated uniformly and a set of generators for N can be found. (b') Same as (b). (c') Same as (c) but for AlgEcon list-decodability.

Proof. Fix $G \in \mathfrak{Groups}_{\text{finite}}$ and $H \in \mathfrak{G}$. Let N be the (G, H)-irrelevant kernel. By Theorem 3.4.10, it suffices to show that $G/N \in \mathfrak{G}$.

Let

$$\widetilde{H} = \prod_{\varphi \in \mathrm{aHom}(G,H)} H. \tag{3.35}$$

Define the map $\psi\colon G \to \widetilde{H}$ by $\psi(g) = (\varphi(g))_\varphi$. Notice that $\psi(G)$ is subgroup of $\widetilde{H}$, which is a direct product of copies of H. Since $\mathfrak{G}$ is closed under subgroups and direct products, it follows that $\psi(G) \in \mathfrak{G}$.

Since $\ker(\psi) = N$, we have $\psi(G) \cong G/N$, so $G/N \in \mathfrak{G}$. □

The AlgEcon list-decodability of {abelian → abelian} and {nilpotent → nilpotent} homomorphism codes is shown in [15] and [21], respectively. As the class of abelian groups and the class of nilpotent groups are each quasivarieties of finite groups, we conclude the following using the mentioned results and Theorem 3.4.21.

Corollary 3.4.23. *If G is a finite group and H an abelian (or, more generally, nilpotent) group, then* $\mathrm{aHom}(G, H)$ *is AlgEcon (and therefore CombEcon).*

3.4.5 Reduction of affine to ordinary homomorphism codes

We show that the code $\mathrm{Hom}(G, H)$ is CombEcon if and only if $\mathrm{aHom}(G, H)$ is CombEcon, and similarly for CertEcon and AlgEcon under modest assumptions of the representation of the groups. Therefore we can use these two types of codes interchangeably. This reflects a phenomenon similar to our results on mean-list-decoding.

Recall Proposition 3.2.6 [20] says that $\Lambda_{\mathrm{Hom}(G,H)} = \Lambda_{\mathrm{aHom}(G,H)}$.

We state the central result of this section, that every aHom list is contained within a small number of translated Hom lists. It is similar in spirit to the mean-list to list reduction, Theorem 3.4.3. For more precise statements, see Lemma 3.4.30 and Corollary 3.4.31.

Theorem 3.4.24 (Reduction of affine to ordinary homomorphisms)**.** *Let G and H be finite groups.*

(i) If $\mathrm{Hom}(G, H)$ *is CombEcon, then* $\mathrm{aHom}(G, H)$ *is CombEcon.*

(ii) Under suitable access assumptions (Access 3.4.25), if $\mathrm{Hom}(G, H)$ *is CertEcon, then* $\mathrm{aHom}(G, H)$ *is CertEcon.*

(iii) Under suitable access assumptions (Access 3.4.25), if $\mathrm{Hom}(G, H)$ *is AlgEcon, then* $\mathrm{aHom}(G, H)$ *is AlgEcon.*

The same results hold for the strong versions of these concepts.

The deterioration in cost is as described in Remark 3.4.26.

Access 3.4.25. An oracle provides uniform random elements of G.

Remark 3.4.26 (Deterioration)*.* The bounds on cost in the result above deteriorate as follows.

- A $\frac{1}{\Lambda+\varepsilon}$ multiplicative factor in list size.
- An $O(\frac{1}{\Lambda+\varepsilon}\log(1/\varepsilon))$ multiplicative factor in queries to the received word f.
- An $O(\frac{1}{\Lambda+\varepsilon}\log(1/\varepsilon))$ multiplicative factor in amount of work.

We set notation for this section. For an affine homomorphism $\varphi \in \mathrm{Hom}(G, H)$, let φ_0 denote its base homomorphism (the unique homomorphism satisfying $\varphi = h\varphi_0$ for some $h \in H$; see Def. 3.2.3). For an element $a \in G$ and function $f\colon G \to H$, let f^a denote the function $f^a\colon G \to H$ given by $f^a(g) = f(a)^{-1}f(ag)$.

Towards proving the reduction of affine to ordinary homomorphisms, we state a few facts relating affine homomorphisms to their base homomorphisms.

Observation 3.4.27. *Let G and H be groups and $\varphi \in \mathrm{aHom}(G,H)$. Then,*

$$\varphi(a)^{-1}\varphi(ag) = \varphi_0(g) \quad \forall a, g \in G. \quad \square \tag{3.36}$$

Corollary 3.4.28. *Let G and H be groups, $f\colon G \to H$, and $\varphi \in \mathrm{aHom}(G,H)$. If $f(a) = \varphi(a)$, then*

$$f(ag) = \varphi(ag) \iff f(a)^{-1}f(ag) = \varphi_0(g) \iff f^a(g) = \varphi_0(g). \tag{3.37}$$

It follows that $\mathrm{agr}(f,\varphi) = \mathrm{agr}(f^a, \varphi_0)$. $\square$

Lemma 3.4.29. *Let G be a finite group, H a group. Let $\varphi \in \mathrm{aHom}(G,H)$. Let γ be a certificate for φ_0 in $\mathrm{Hom}(G,H)$ with $\gamma(1) = 1$. Let $a \in G$. Then $\widetilde{\gamma}\colon a \cdot \mathrm{dom}(\gamma) \to H$ given by $\widetilde{\gamma}(g) = \varphi(a)\gamma(a^{-1}g)$ is a certificate for φ in $\mathrm{aHom}(G,H)$.*

Proof. First, $\widetilde{\gamma}$ is a restriction of φ.

Suppose $\widetilde{\gamma}$ also a restriction of $\psi \in \mathrm{aHom}(G,H)$. For $g \in \mathrm{dom}(\gamma)$, we have that $\psi_0(g) = \psi(a)^{-1}\psi(ag) = \varphi(a)^{-1}\varphi(a)\gamma(g) = \gamma(g)$. So, γ is a restriction of ψ_0. So, $\psi_0 = \varphi_0$. Also, $\psi(a) = \widetilde{\gamma}(a) = \varphi(a)$. So, $\psi = \varphi$.

Thus, $\widetilde{\gamma}$ is a certificate for φ in $\mathrm{aHom}(G,H)$. $\square$

The following lemma shows how to turn a certificate list-decoder for $\mathrm{Hom}(G,H)$ into a certificate list-decoder for $\mathrm{aHom}(G,H)$.

Lemma 3.4.30. *Let G be a finite group, H a group, $\lambda, \delta > 0$. Let $f\colon G \to H$ be a received word.*

Let a be a uniform random element of G. Let γ be a random partial map such that for each $\varphi \in \mathcal{L}(\mathrm{Hom}(G,H), f^a, \lambda)$, we have that γ is a certificate for φ with probability $\geq \delta$. Assume that $\gamma(1) = 1$. Let $\widetilde{\gamma}\colon a \cdot \mathrm{dom}(\gamma) \to H$ by $\widetilde{\gamma}(g) = f(a)\gamma(a^{-1}g)$ for $g \in a\,\mathrm{dom}(\gamma)$.

Then, for each $\varphi \in \mathcal{L}(\mathrm{aHom}(G,H), f, \lambda)$, we have that $\widetilde{\gamma}$ is a certificate for φ with probability $\geq \lambda\delta$.

Proof. Let φ_0 be the base homomorphism of φ. If $a \in \mathrm{Eq}(f, \varphi)$, which happens with probability $\geq \lambda$, then $\mathrm{agr}(f^a, \varphi_0) = \mathrm{agr}(f, \varphi) \geq \lambda$ by Corollary 3.4.28. If so, then γ is a certificate for φ_0 with probability $\geq \delta$. In this case, $\widetilde{\gamma}$ is a certificate for φ by Lemma 3.4.29.

So, $\mathcal{B}$ returns φ with probability $\geq \lambda\delta$. □

Corollary 3.4.31. *Let G be a finite group, H a group, $\lambda > 0$. Then,*

$$\ell(\mathrm{aHom}(G, H), \lambda) \leq \frac{1}{\lambda}\ell(\mathrm{Hom}(G, H), \lambda). \tag{3.38}$$

Proof. Let $f\colon G \to H$. We bound the size of $\mathcal{L} = \mathcal{L}(\mathrm{aHom}(G, H), f, \lambda)$. Apply Lemma 3.4.30, with γ chosen uniformly from among the $\leq \ell(\mathrm{Hom}(G, H), \lambda)$ elements of $\mathcal{L}(\mathrm{Hom}(G, H), f^a, \lambda)$. Note that $\widetilde{\gamma}$ can be a certificate for at most one $\varphi \in \mathcal{L}$. □

Proof of Theorem 3.4.24. (i) Apply Corollary 3.4.31 with $\lambda = \Lambda + \varepsilon$.

(ii) Given an individual-certificate CertEcon algorithm $\mathcal{A}$ for $\mathrm{Hom}(G, H)$, we construct an individual-certificate CertEcon algorithm for $\mathrm{aHom}(G, H)$. This is sufficient by Corollary 3.4.2.

On input $f\colon G \to H$, $\varepsilon > 0$, let a be a uniform random element of G, and let $\gamma = \mathcal{A}(f^a, \varepsilon)$. Let $\widetilde{\gamma}$ be as in Lemma 3.4.30. Return $\widetilde{\gamma}$. This is an individual-certificate CertEcon algorithm by Lemma 3.4.30.

(iii) Similar to (ii). □

3.5 Strategy

Throughout this section, let G be a finite, group, H a group, $f \in \mathrm{Hom}(G, H)$ a received word, $\varepsilon > 0$. Let $\mathcal{L} = \mathcal{L}(\mathrm{Hom}(G, H), f, \Lambda_{G,H} + \varepsilon)$ be the list of codewords within distance $(\mathsf{mindist} - \varepsilon)$ of f. The combinatorial problem is to find a bound of the form $|\mathcal{L}| \leq \mathrm{poly}(1/\varepsilon)$. We work with homomorphisms, but by the reduction of affine to ordinary homomorphisms, Theorem 3.4.24, our results also hold for affine homomorphisms.

In this paper, we prove CombEcon bounds for these kinds of domain groups G: abelian, alternating, and SRG (see Section 3.3.4 for the definition of SRG). We allow the codomain group H to be an arbitrary group (finite or infinite). We shall give two separate proofs for the alternating case, a nonconstructive one in Section 3.7 and a constructive one in Section 3.8 inferable from the fact that alternating groups are SRG.

3.5.1 Overview of strategy

Our core strategy for counting the elements of $\mathcal{L}$ proceeds in two steps (which are modified for each of the three kinds of domains). Say that a $G \rightharpoonup H$ partial map is *substantial* if its domain is a coset of a bounded-depth subgroup of G. The two steps are as follows.

(1) Find a small (poly$(1/\varepsilon)$-size) set Γ of substantial $G \rightharpoonup H$ partial maps such that every element of $\mathcal{L}$ is an extension of some $\gamma \in \Gamma$.

(2) Show that a substantial $G \rightharpoonup H$ partial map extends to a poly$(1/\varepsilon)$ number of elements of $\mathcal{L}$.

If we can carry out these two steps for a class of homomorphism codes, then that class is CombEcon.

Our main tools for carrying out these steps (or, at least, nearly carrying out these steps) are the following.

(i) The *bucket-splitting lemma* (Lemma 3.5.11): There is a set $\Psi \subseteq \mathcal{L}$ of size $O(1/\varepsilon^2)$ such that every $\varphi \in \mathcal{L}$ has agreement $> \Lambda^2$ with some $\psi \in \Psi$.

(ii) The *shallow extension theorem* (Theorem 3.5.1): If γ is a $G \rightharpoonup H$ partial map whose domain is a subgroup of G of depth d, then there are at most $1/\varepsilon^d$ extensions of γ in $\mathcal{L}$.

We prove the shallow extension theorem (ii) in Section 3.5.2. We give a sphere-packing argument in Section 3.5.3, which allows us to prove the bucket-splitting lemma (i) in Section 3.5.4.

The shallow extension theorem (ii) faithfully carries out step (2) of the strategy. Furthermore, the proof of this theorem is constructive — it is useful not only for combinatorial results, but also for (semi-)algorithmic results. However, the bucket-splitting lemma (i) does not entirely fulfill step (1) of the strategy. It has the following shortcomings.

- If we knew that $\varphi \in \mathcal{L}$ agreed with a $\psi \in \Psi$ on a *particular* large subgroup K, then we could say that φ extends the partial map $\psi|_K$. However, while the bucket-splitting lemma tells us that every element of $\mathcal{L}$ agrees with some $\psi \in \Psi$ on *some* large subgroup of G, it does not say *where* that agreement occurs.

- Even though we know that each $\varphi \in \mathcal{L}$ agrees with some $\psi \in \Psi$ on some large subgroup, "large" in this context means density $> \Lambda^2$, rather than bounded-depth. While in our contexts, the former essentially implies the latter, it fails, for instance, for the class of linear groups $\mathrm{SL}(n, q)$, the next natural target for this line of work.

- The proof of the bucket-splitting lemma is non-constructive, and therefore does not directly suggest a path to an algorithmic solution. There are cases, however, where the mere existence of a combinatorial bound is the core of the analysis of the algorithm (for instance, for the {abelian $\to$ abelian} algorithm of [15] which is adapted into the {abelian $\to$ arbitrary} algorithm in this paper in Section 3.6).

To overcome these shortcomings, we modify the strategy above, in a different way for each of our three kinds of domain groups.

Alternating domains offer the most direct application of our core strategy. In this case, we carry out step (1) of the strategy by applying the bucket-splitting lemma, and then analyzing the possible subgroups on which $\varphi \in \mathcal{L}$ and $\psi \in \Psi$ might agree. We use the subgroup structure of the alternating group to show that agreement always occurs on one of a small ($\mathrm{poly}(1/\Lambda)$) number of bounded-depth subgroups. We can then apply the shallow extension theorem. We elaborate on this strategy in Section 3.5.6.

In the case of SRG domain (which includes alternating groups as a particular case), we take a different approach to achieve step (1) of our strategy. We produce substantial partial maps by evaluating the received word f on a random bounded-size set $S \subseteq G$ of inputs, and taking the restriction $f|_S$. This step is algorithmically efficient. We then apply the random subgroup density lemma (Lemma 3.5.3), a relative of the shallow extension theorem, to produce certificates. We describe this strategy further in Section 3.5.7.

In the case of abelian domain, we use both the bucket-splitting lemma and the random subgroup density lemma, but we find a way to sidestep the issue of not knowing where the agreement between $\varphi \in \mathcal{L}$ and $\psi \in \Psi$ occurs. To do this, we focus on the image of each homomorphism instead of the map itself, reducing the {abelian $\to$ arbitrary} problem to {abelian $\to$ abelian}. To achieve this reduction we introduce a new group-theoretic tool we call an *abelian enlargement* (see Section 3.6.3).

3.5.2 Shallow extension theorem

In all three of our universal CombEcon proofs — for abelian groups, for alternating groups, and for SRG groups — we use either the shallow extension theorem (Theorem 3.5.1, used for alternating groups) or the lemma that underlies it, the random subgroup density lemma (Lemma 3.5.3, used for abelian and SRG groups). The theorem allows us to bound the number of codewords in the list that extend a particular partial map.

Theorem 3.5.1 (Shallow extension)**.** *Let G be a finite group, H a group, and $S \subseteq G$ a subset. Let $f\colon G \to H$. Let $\gamma\colon S \to G$. Let $\varepsilon > 0$. Then there are at most*

$$1/\varepsilon^{\operatorname{depth}_G(\langle S \rangle)}. \tag{3.39}$$

extensions of γ to an element of $\mathcal{L}(\operatorname{Hom}(G,H), f, \Lambda + \varepsilon)$.

Remark 3.5.2. (a) Theorem 3.5.1 also holds if f is replaced with a family $\mathscr{F}$ of maps $G \to H$.

(b) Theorem 3.5.1 also holds if Hom is replaced with aHom, and $\langle S \rangle$ is replaced with the subgroup $K \leq G$ such that $\langle S \rangle_{\mathsf{aff}}$ is a coset of K.

The proof of Theorem 3.5.1 relies on the following lemma.

The set S in the lemma should be thought of as $\mathrm{Eq}(f, \varphi)$ for some homomorphism φ of interest.

Lemma 3.5.3 (Random subgroup density lemma)**.** *Let $0 \leq \lambda < 1$. Let G be a finite group, $K \leq G$ a subgroup, and $S \subseteq G$ a subset. Suppose that $\mu_G(S) > \lambda$. Let $\varepsilon = \mu(S) - \lambda$ and $d = \mathrm{depth}_G(K)$. Then,*

$$\Pr_{s_1,\ldots,s_d \in S}[\mu(\langle K, s_1, \ldots, s_d \rangle) > \lambda] \geq \left(\frac{\varepsilon}{\lambda + \varepsilon}\right)^d. \tag{3.40}$$

It follows that

$$\Pr_{g_1,\ldots,g_d \in G}[g_1, \ldots, g_d \in S \text{ and } \mu(\langle K, g_1, \ldots, g_d \rangle) > \lambda] \geq \varepsilon^d. \tag{3.41}$$

Proof of Lemma 3.5.3. Pick $s_1, s_2, s_3, \ldots$ independently and uniformly from S. We proceed by induction on $|G : K|$.

Suppose $\mu(K) > \lambda$. Then, $\Pr[\mu(\langle K, s_1, \ldots, s_d \rangle) > \lambda] = 1$.

Suppose $\mu(K) \leq \lambda$. Then, with probability $\dfrac{\mu(S \smallsetminus K)}{\mu(S)} \geq \dfrac{\varepsilon}{\lambda + \varepsilon}$, we have that $s_1 \notin K$, so $\langle K, s_1 \rangle > K$, and $\mathrm{depth}_G \langle K, s_1 \rangle \leq d - 1$. Then, by the induction hypothesis,

$$\Pr[\mu(\langle K, s_1, \ldots, s_d \rangle)] \geq \Pr[\mu(\langle K, s_1, \ldots, s_d \rangle) \mid s_1 \notin K] \cdot \Pr[s_1 \notin K] \tag{3.42}$$

$$\geq \left(\frac{\varepsilon}{\lambda + \varepsilon}\right)^{d-1} \cdot \left(\frac{\varepsilon}{\lambda + \varepsilon}\right). \tag{3.43}$$

This completes the inductive step. □

Remark 3.5.4. With a little more care in the proof (separating out the case $\lambda/2 < \mu(K) \leq \lambda$),

one can prove the stronger conclusion

$$\Pr[\mu(\langle K, s_1, \ldots, s_d\rangle) > \lambda] \geq \left(\frac{\frac{1}{2}\lambda + \varepsilon}{\lambda + \varepsilon}\right)^{d-1} \cdot \frac{\varepsilon}{\lambda + \varepsilon}. \tag{3.44}$$

Proof of Theorem 3.5.1. Let $d = \operatorname{depth}_G(\langle S\rangle)$. Randomly construct a partial map $\widetilde{\gamma}$ by choosing $g_1, \ldots, g_d \in G$ uniformly and independently, and defining $\widetilde{\gamma}\colon S \cup \{g_1, \ldots, g_k\} \to H$ by

$$\widetilde{\gamma}(x) = \begin{cases} \gamma(x) & \text{if } x \in S, \\ f(x) & \text{if } x \in \{g_1, \ldots, g_k\} \setminus S. \end{cases} \tag{3.45}$$

Let $\mathcal{L} = \mathcal{L}(\operatorname{Hom}(G, H), f, \Lambda + \varepsilon)$. Consider any $\varphi \in \mathcal{L}$ that extends γ. If $g_1, \ldots, g_k \in \operatorname{Eq}(f, \varphi)$, then φ extends $\widetilde{\gamma}$. If furthermore $\mu(\langle S, g_1, \ldots, g_k\rangle) > \Lambda$, then $\widetilde{\gamma}$ is a certificate for φ. By Lemma 3.5.3 with $K = \langle S\rangle$, since $\operatorname{agr}(f, \varphi) \geq \Lambda + \varepsilon$, this happens with probability at least ε^d.

Since a partial map can be a certificate for at most one homomorphism, this implies that there are at most $1/\varepsilon^d$ homomorphisms in $\mathcal{L}(\operatorname{Hom}(G, H), f, \Lambda + \varepsilon)$ that extend γ. □

3.5.3 Strong negative correlation and a sphere packing argument

We shall use a sphere-packing argument to split the list into more manageable parts. We will use the sphere-packing bound (Lemma 3.5.7) to prove the bucket-splitting lemma (Lemma 3.5.11) in Section 3.5.4. We also show in this section that in the definition of CombEcon, we may replace the bound $\operatorname{poly}(1/\varepsilon)$ by $\operatorname{poly}(1/\Lambda, 1/\varepsilon)$ without changing the meaning.

We begin with a strong negative correlation inequality.

Definition 3.5.5 (Strong negative correlation)**.** Let $\tau > 0$. Let $A_1, \ldots, A_k$ be events in a probability space P. We say that $A_1, \ldots, A_k$ are **τ-strongly negatively correlated** if $\Pr(A_i \cap A_j) \leq \Pr(A_i)\Pr(A_j) - \tau$ for all $i \neq j$.

Lemma 3.5.6 (Strong negative correlation bound)**.** *Let $\tau > 0$. Let $A_1, \ldots, A_k$ be τ-strongly negatively correlated events in a probability space. Then $k \leq \frac{1}{4\tau} + 1$.*

Proof. For $1 \leq i \leq k$, let Z_i be the indicator random variable (characteristic function) of the event A_i; so $\mathbb{E}(Z_i) = \Pr(A_i)$ and $\operatorname{Var}(Z_i) = \Pr(A_i)(1 - \Pr(A_i)) \leq \frac{1}{4}$. For the covariances ($i \neq j$) we have $\operatorname{Cov}(Z_i, Z_j) = \mathbb{E}[Z_i Z_j] - \mathbb{E}[Z_i]\,\mathbb{E}[Z_j] \leq -\tau$. So,

$$0 \leq \operatorname{Var}\left(\sum_i Z_i\right) = \sum_i \operatorname{Var}(Z_i) + \sum_{i \neq j} \operatorname{Cov}(Z_i, Z_j) \leq \frac{k}{4} - k(k-1)\tau. \tag{3.46}$$

Solving for k gives the bound as claimed. □

In our applications, P will be the uniform distribution μ over a finite set and we shall always have $\Pr(A_i) \geq \Lambda + \varepsilon$.

Lemma 3.5.7 (Sphere-packing bound)**.** *Let G be a finite group, H a group, and $\varepsilon > 0$. Let $f\colon G \to H$ be a received word. Let $\Psi \subseteq \mathcal{L} = \mathcal{L}(\mathrm{aHom}(G, H), f, \Lambda + \varepsilon)$ be a subset of the list that is maximal under the constraint that $\operatorname{agr}(\psi_1, \psi_2) \leq \Lambda^2$ for all distinct $\psi_1, \psi_2 \in \Psi$. Then*

$$|\Psi| \leq \frac{1}{4(2\Lambda + \varepsilon)\varepsilon} + 1 \leq \frac{1}{4\varepsilon^2} + 1. \tag{3.47}$$

Proof. Observe that the sets $\operatorname{Eq}(\psi, f)$ for $\psi \in \Psi$ have density $\geq \Lambda + \varepsilon$ and they are $\varepsilon(2\Lambda + \varepsilon)$-strongly negatively correlated. Apply Lemma 3.5.6. □

Remark 3.5.8. While the sphere-packing bound (Lemma 3.5.7) applies to all groups, it is an existential result. We cannot algorithmically find Ψ. So, although we use this lemma in proving combinatorial results, we cannot directly translate those proofs into algorithms.

The way we overcome this varies by setting. The list-decoder for {abelian → arbitrary} is indifferent to how CombEcon is proved; the AlgEcon proof relies on already having proved that abelian groups are universally CombEcon, but the algorithm is is unconnected to the method of proof. In other cases, including {alternating → arbitrary}, we only get a com-

binatorial result using the sphere-packing bound, but alternating groups are SRG, and for SRG groups we can avoid the need for the sphere-packing bound (see Section 3.8).

As another consequence of the strong negative correlation bound, we get the following reduction.

Lemma 3.5.9 (Small Λ lemma)**.** *Let $\mathcal{L} = \mathcal{L}(\mathrm{aHom}(G,H), f, \Lambda+\varepsilon)$. If $|\mathcal{L}| \le p(1/\Lambda, 1/\varepsilon)$ for some monotone function $p(.,.)$ then $|\mathcal{L}| \le p(2/\varepsilon^2, 1/\varepsilon) + 1/(2\varepsilon^2)$. In particular, in the definition of CombEcon, we may replace the bound* $\mathrm{poly}(1/\varepsilon)$ *by* $\mathrm{poly}(1/\Lambda, 1/\varepsilon)$ *without changing the meaning.*

Lemma 3.5.7 and Lemma 3.5.9 also hold when $\mathrm{aHom}(G,H)$ is replaced with $\mathrm{Hom}(G,H)$, or, in fact, any other code $\mathcal{C}$.

Proof. For $\Lambda > \varepsilon^2/2$, we are done. For $\Lambda \le \varepsilon^2/2$ we have $|\mathcal{L}| \le 1 + 1/(2\varepsilon^2)$ by Lemma 3.5.6 because the sets $\mathrm{Eq}(f,\varphi)$ for $\varphi \in \mathcal{L}$ are $(\varepsilon^2/2)$-strongly negatively correlated. □

3.5.4 Bucket splitting

In this section, we prove the bucket-splitting lemma (Lemma 3.5.11), which we use in the proof of universal CombEcon for abelian groups, and the first proof of universal CombEcon for alternating groups.

Let G be a finite group, H a group (finite or infinite), $\Lambda = \Lambda_{G,H}$ the maximum agreement, $f\colon G \to H$ a received word, $\varepsilon > 0$, and $\mathcal{L} = \mathcal{L}(\mathrm{aHom}(G,H), f, \Lambda+\varepsilon)$ the list. Let Ψ be as in the sphere-packing bound (Lemma 3.5.7).

We split $\mathcal{L}(\mathrm{aHom}(G,H), f, \Lambda+\varepsilon)$ into sets that we call *buckets*, which we label with elements of Ψ. Each bucket, denoted $\mathcal{L}_\psi$, will contain the sphere centered at the homomorphism $\psi \in \Psi$ with radius $(1-\Lambda^2)$.

Definition 3.5.10 (Bucket $\mathcal{L}_\psi$)**.** For $\psi \in \Psi$, we define the *bucket* $\mathcal{L}_\psi$ by

$$\mathcal{L}_\psi \coloneqq \{\varphi \in \mathcal{L} \mid \mathrm{agr}(\varphi,\psi) > \Lambda^2\}. \tag{3.48}$$

The union of the buckets includes the list $\mathcal{L}$. Since the number of buckets is $|\Psi| \leq 1/(4\varepsilon^2) + 1$, to get a poly$(1/\varepsilon)$-bound on list size, we only need to bound the size of each bucket by poly$(1/\varepsilon)$.

Lemma 3.5.11 (Bucket-splitting lemma)**.** *Let G be a finite group, H a group, $f\colon G \to H$, $\psi \in \mathrm{aHom}(G, H)$, and $\varepsilon > 0$. Then, there exists a subset $\Psi \subseteq \mathcal{L}(\mathrm{aHom}(G, H), f, \Lambda + \varepsilon)$, with size $|\Psi| \leq \frac{1}{4(2\Lambda+\varepsilon)\varepsilon} + 1$, such that*

$$\mathcal{L}(\mathrm{aHom}(G, H), f, \Lambda + \varepsilon) \subseteq \bigcup_{\psi \in \Psi} \mathcal{L}_\psi. \tag{3.49}$$

Proof. Let Ψ be as in Lemma 3.5.7, that is, a subset of $\mathcal{L}(\mathrm{aHom}(G, H), f, \Lambda+\varepsilon)$ that is maximal under the conditions that distinct $\psi_1, \psi_2 \in \Psi$ have small agreement $\mathrm{agr}(\psi_1, \psi_2) \leq \Lambda^2$. By the maximality of Ψ, every $\varphi \in \mathcal{L}(\mathrm{aHom}(G, H), f, \Lambda+\varepsilon)$ has high agreement $\mathrm{agr}(\varphi, \psi) > \Lambda^2$ with some homomorphism $\psi \in \Psi$. □

3.5.5 Bounding the list size for abelian groups

To prove that abelian groups are universally CombEcon, we prove that the codomain has a small number of abelian subgroups such that each homomorphism in the list $\mathcal{L}$ maps the domain G into one of those abelian subgroups. This reduces the problem to showing CombEcon for {abelian → abelian} homomorphism codes, which was done by Dinur, Grigorescu, Kopparty, and Sudan [15].

We elaborate on this strategy in Section 3.6.

3.5.6 Bounding the list size for alternating groups

In this section, we outline our first proof that alternating groups are universally CombEcon. For details, see Section 3.7. We work with homomorphisms instead of affine homomorphisms in this section; this is sufficient for showing CombEcon by the reduction of affine to ordinary homomorphisms (Theorem 3.4.24).

Recall that subgroups of polynomial index in alternating groups are described by a the Jordan–Liebeck Theorem (JLT), Theorem 3.2.19. in A_n and $(A_n)_\Delta$ the setwise stabilizer of Δ. We use JLT multiple times in this section.

Ignoring the trivial case $\Lambda = 0$, it is easy to show that for $G = A_n$ with $n \geq 5$ we have $\Lambda \geq 1/\binom{n}{2}$. It then follows from JLT (Theorem 3.2.19) that for $n \geq 10$ we have $\Lambda = 1/\binom{n}{s}$ for $s \in \{1, 2\}$. In the light of Lemma 3.5.9 it suffices to find a $\text{poly}(n, 1/\varepsilon)$ bound on the size of each bucket $\mathcal{L}_\psi$.

Let $\mathcal{K}$ denote the set of all subgroups that are the pointwise stabilizer of $2s$ points:

$$\mathcal{K} = \{(A_n)_{(\Delta)} \mid \Delta \subseteq [n], |\Delta| = 2s\} \tag{3.50}$$

where $s \in \{1, 2\}$ and $\Lambda = 1/\binom{n}{s}$ (see above).

We shall refer to the elements of $\mathcal{K}$ as *label subgroups*. We have $|\mathcal{K}| = \binom{n}{2s}$. By JLT for $n \geq 11$, every subgroup of A_n of index $< \binom{n}{2s+1}$ contains a member of $\mathcal{K}$. By Corollary 3.2.27, the depth of any $K \in \mathcal{K}$ in A_n is 5 if $s = 2$ and 2 if $s = 1$. (All we need is that this depth is bounded.)

All homomorphisms φ in the bucket $\mathcal{L}_\psi$ have agreement $> \Lambda^2$ with one representative homomorphism ψ; so φ and ψ agree on a subgroup of index $< 1/\Lambda^2 < \binom{n}{2s+1}$ (for $n \geq 40$) and therefore, by JLT, they agree on some label subgroup. So we can split each bucket $\mathcal{L}_\psi$ further into $\binom{n}{2s}$ *sub-buckets* $\mathcal{L}_{\psi,K}$, where the homomorphisms in $\mathcal{L}_{\psi,K}$ agree with ψ on K.

Since $K \in \mathcal{K}$ has depth $d \leq 2s + 1$, we can apply the shallow extension theorem, Theorem 3.5.1, to $\gamma = \psi|_K$ to find that the size of a sub-bucket is $|\mathcal{L}_{\psi,K}| \leq 1/\varepsilon^d$.

Combining our bounds on the number of buckets, the number of sub-buckets per bucket, and size of each sub-bucket, we conclude that $|\mathcal{L}| = O(\varepsilon^{-2s-3}\Lambda^{-2})$ and therefore $|\mathcal{L}| = O(\varepsilon^{-2s-7}) = O(\varepsilon^{-11})$ by Lemma 3.5.9.

This concludes our first proof that the alternating groups are universally CombEcon. The proof is non-constructive because it relies on the sphere packing argument. We elaborate on

this proof in Section 3.7.

3.5.7 Bounding the list size for SRG groups

Our second strategy for list decoding {alternating → arbitrary} exploits a property of the alternating groups which we call *shallow random generation* (SRG). We outline this strategy in this section; for details, see Section 3.8.

Recall that a class $\mathfrak{G}$ of groups G is SRG if, roughly, a small number of random elements of G are extremely likely to generate a shallow subgroup; see Definitions 3.3.9 and 3.3.10.

Alternating groups are an example of a class of SRG groups.

Theorem 3.5.12. *The class of alternating groups is SRG. Specifically, for sufficiently large n, the group A_n is $(2, 5)$-shallow generating.*

The proof uses Theorem 3.2.20 [2], which says that two random elements of an alternating group are extremely likely to act as an alternating or symmetric group on a large subset of the permutation domain.

To prove Theorem 3.5.12 we use Theorem 3.2.20 with $t = 4$, noting that $\Lambda^*_{A_n} = 1/\binom{n}{2}$ and $\mathrm{depth}_{A_n}(A_{n-4}) = 5$.

We then prove that SRG classes of groups are CombEcon using the shallow extension theorem, Theorem 3.5.1. This view allows us not only to combinatorially list-decode {SRG → arbitrary}, but also (via the random subgroup density lemma, Lemma 3.5.3) to certificate list-decode via domain certificates. Domain certificates are given by restricting the received word to small number of random elements of G (see Section 3.1.5 for the definition of domain certificate). If $\mathfrak{G}$ is an SRG class of groups, then $\mathfrak{G}$ is universally Λ-CertEcon. For the class of alternating groups, this algorithm produces a list of length $\widetilde{O}(\varepsilon^{-8})$.

3.6 Abelian domain, combinatorial and algorithmic list-decoding

In this section we describe the details of the proof that finite abelian groups are universally combinatorially and algorithmically economically list-decodable. The key technical result is Theorem 3.6.3, which says that there are a small number of abelian subgroups of the codomain such that every homomorphism in the list maps into one of these subgroups. To prove this result, we introduce a new concept that we call an *abelian enlargement*.

We did not succeed in generalizing the result to nilpotent domains, the technical reason being that the abelian enlargement concept does not seem to have an analog. Therefore the following problem remains open.

Open Problem 3.6.1. Are nilpotent groups universally CombEcon?

We remark that {abelian $\rightarrow$ abelian} homomorphism codes usually cannot be list-decoded beyond radius $1 - (\Lambda_{G,H} + \varepsilon)$ (see Remark 1.4.7).

3.6.1 Structure of the section

In Section 3.6.2, we state characterizations of $\Lambda_{G,H}$ when G is abelian. In Section 3.6.3, we state Theorem 3.6.3 (the key result mentioned in the previous paragraph). In Section 3.6.4 we introduce abelian enlargements, the concept central to the proof of Theorem 3.6.3. Using this tool, in Section 3.6.5 we prove Theorem 3.6.3 and infer that abelian groups are universally CombEcon. In Section 3.6.6 we adapt the algorithm of [15, 21], to give an algorithm to locally list-decode these codes.

3.6.2 $\Lambda_{G,H}$ when G is abelian

The following characterization of $\Lambda_{G,H}$ for G abelian is straightforward.

Fact 3.6.2. Let G be a finite abelian group and H a group (finite or infinite). The following are equivalent for any prime p.

(a) $\Lambda_{G,H} = 1/p$.

(b) p is the smallest prime number such that p divides $|G|$ and H has an element of order p.

(c) p is the smallest prime number dividing $|G : N|$, where N is the (G, H)-irrelevant kernel.

If no such p exists in (b) or (c), then $|\mathrm{Hom}(G, H)| = 1$ and $\Lambda_{G,H} = 0$.

Guo [20, Theorem 1.1] gave a characterization of $\Lambda_{G,H}$ when G and H are finite groups with G solvable or H nilpotent. In Section 3.9 we generalize this to the case where G is a finite group, H is a group (finite or infinite), and at least one of G or H is solvable.

3.6.3 Structure theorem for abelian domains

To show that abelian groups are universally CombEcon, we prove a structure theorem, Theorem 3.6.3. Specifically, we show that there is a poly$(1/\varepsilon)$-size set of abelian subgroups of the codomain H such that every homomorphism in the list $\mathcal{L}$ maps the domain G into one of these abelian subgroups. We combine this with the {abelian $\to$ abelian} CombEcon result of Dinur et al. [15] to conclude that {abelian $\to$ arbitrary} homomorphism codes are CombEcon.

We work now with homomorphisms instead of affine homomorphisms; we will appeal to the reduction of affine to ordinary homomorphisms, Theorem 3.4.24, to obtain affine results.

Theorem 3.6.3. *Let G be a finite abelian group and H an arbitrary group. Let $f \in H^G$ be a received word. Let $\varepsilon > 0$. Then there exists a set $\mathcal{A}$ of finite abelian subgroups of the codomain H with $|\mathcal{A}| \le \dfrac{1}{4(2\Lambda+\varepsilon)\varepsilon^2} + \dfrac{1}{\varepsilon}$ such that for all $\varphi \in \mathcal{L} = \mathcal{L}(\mathrm{Hom}(G, H), f, \Lambda + \varepsilon)$, there is $M \in \mathcal{A}$ such that $\varphi(G) \le M$.*

We will find $\mathcal{A}$ by working separately on each bucket. We define, for each $\psi \in \Psi$, a set $\mathcal{A}_\psi$ of finite abelian subgroups of H such that

(i) for all $\varphi \in \mathcal{L}_\psi$, there is $M \in \mathcal{A}_\psi$ such that $\varphi(G) \le M$,

(ii) $|\mathcal{A}_\psi| \leq 1/\varepsilon$.

It follows from (i) that we can set $\mathcal{A} = \bigcup_{\psi \in \Psi} \mathcal{A}_\psi$, so Theorem 3.6.3 follows from (ii) and the sphere packing bound (Lemma 3.5.7).

To construct the sets $\mathcal{A}_\psi$ and prove properties (i) and (ii), we introduce a concept that we call an *abelian enlargement*.

3.6.4 Abelian enlargements

To define the set $\mathcal{A}_\psi$, we introduce the following concept.

Definition 3.6.4. Let H be a group, $B \leq H$ and $T \subseteq H$. The *abelian enlargement* of T by B is the group generated by T and the elements of B that commute with all elements of T, i. e.,

$$\mathrm{enl}_B(T) = \langle T, C_H(T) \cap B \rangle. \tag{3.51}$$

Note that if both $\langle T \rangle$ and B are finite abelian groups then so is $\mathrm{enl}_B(T)$; this is the only case in which we shall be interested. When $T = \{h\}$ is a singleton, we write $\mathrm{enl}_B(h)$ for $\mathrm{enl}_B(T)$.

Fix $\psi \in \Psi$. Let $\mathcal{A}_\psi$ be the set of all subgroups $M \leq H$ that occur as $M = \mathrm{enl}_{\psi(G)}(\varphi(G))$ for some φ in the bucket $\mathcal{L}_\psi$. We shall show that every $M \in \mathcal{A}_\psi$ is equal to $\mathrm{enl}_{\psi(G)}(f(g))$ for at least an ε proportion of $g \in G$. The idea is that since φ and ψ have large agreement, most of $\varphi(G)$ is contained in $\psi(G)$. So even if we take a single random element $g \in G$, it is likely that the enlargement of $\varphi(g)$ by $\psi(G)$ already contains all of $\varphi(G)$. Specifically, we show the following.

Proposition 3.6.5. *Let $\varphi, \psi \in \mathrm{Hom}(G, H)$ and $g \in G$ such that $\langle g, \mathrm{Eq}(\psi, \varphi) \rangle = G$. Then $\varphi(G) \leq \mathrm{enl}_{\psi(G)}(\varphi(G)) = \mathrm{enl}_{\psi(G)}(\varphi(g))$.*

And, since f and φ have high agreement, it is likely that $\varphi(g) = f(g)$.

In this section, we prove Proposition 3.6.5.

Throughout this section, let G be a finite abelian group, and H a group (finite or infinite). We prove facts about abelian enlargements, which were defined in Section 3.6.3.

Lemma 3.6.6. *Let $B \leq H$ be a finite abelian subgroup and $T \subseteq H$ a set such that $\langle T \rangle$ is a finite abelian group. Then, $\mathrm{enl}_B(T)$ is a finite abelian group.*

Proof. Every element of T commutes with every element of $C_H(T)$ by the definition of the centralizer. So, $\mathrm{enl}_B(T)$ is the direct product of $\langle T \rangle$ and $C_H(T) \cap B$. The group $\langle T \rangle$ is finite abelian by assumption, and $C_H(T) \cap B$ is finite abelian because it is a subgroup of B. So, $\mathrm{enl}_B(T)$ is a finite abelian group. □

Lemma 3.6.7. *Let $B \leq H$ be a finite abelian subgroup, and $T \subseteq H$ a subset such that $\langle T \rangle$ is a finite abelian group. For $U \subseteq \mathrm{enl}_B(T)$, we have that $\mathrm{enl}_B(T) = \mathrm{enl}_B(T \cup U)$.*

Proof. First, we show that $\mathrm{enl}_B(T) \leq \mathrm{enl}_B(T \cup U)$. Since $\mathrm{enl}_B(T)$ is abelian (Lemma 3.6.6), we have that

$$C_H(T) \cap B \leq \mathrm{enl}_B(T) \leq C_H(\mathrm{enl}_B(T)) \leq C_H(U). \tag{3.52}$$

So,

$$C_H(T) \cap B \leq C_H(T) \cap C_H(U) \cap B = C_H(T \cup U) \cap B \leq \mathrm{enl}_B(T \cup U). \tag{3.53}$$

Since also $T \subseteq \mathrm{enl}_B(T \cup U)$, we have that $\mathrm{enl}_B(T) = \langle T, C_H(T) \cap B \rangle \leq \mathrm{enl}_B(T \cup U)$.

Next, we show that $\mathrm{enl}_B(T \cup U) \leq \mathrm{enl}_B(T)$. We have that $T \subseteq \mathrm{enl}_B(T)$, that $U \subseteq \mathrm{enl}_B(T)$, and $C_H(T \cup U) \cap B \leq C_H(T) \cap B \leq \mathrm{enl}_B(T)$. So, $\mathrm{enl}_B(T \cup U) = \langle T \cup U, C_H(T \cup U) \cap B \rangle \leq \mathrm{enl}_B(T)$. □

Proposition 3.6.8. *Let $\varphi, \psi \in \mathrm{Hom}(G, H)$ and $A \subseteq G$ such that $\langle A, \mathrm{Eq}(\psi, \varphi), \ker \varphi \rangle = G$. Then $\mathrm{enl}_{\psi(G)}(\varphi(A)) = \mathrm{enl}_{\psi(G)}(\varphi(G))$.*

Proof. Since G is finite abelian, so are $\varphi(G)$ and $\psi(G)$. Let $B = \psi(G)$. Let $T = \varphi(A)$. Let $U = \varphi(\mathrm{Eq}(\psi, \varphi))$. Since $T, U \subseteq \varphi(G)$, and $\varphi(G)$ is abelian, $U \leq C_H(T)$. And, since $U = \psi(\mathrm{Eq}(\psi, \varphi))$, we have that $U \leq \psi(T) = B$. Thus, $U \leq C_H(T) \cap B \leq \mathrm{enl}_B(T)$.

Also, $\langle T \cup U \rangle = \langle T, U, 1 \rangle = \langle \varphi(A), \varphi(\mathrm{Eq}(\psi, \varphi)), \varphi(\ker \varphi) \rangle = \varphi(\langle A, \mathrm{Eq}(\psi, \varphi), \ker \varphi \rangle) = \varphi(G)$.

Therefore, by Lemma 3.6.7,

$$\mathrm{enl}_{\psi(G)}(\varphi(A)) = \mathrm{enl}_B(T) = \mathrm{enl}_B(T \cup U) = \mathrm{enl}_B(\langle T \cup U \rangle) = \mathrm{enl}_{\psi(G)}(\varphi(G)). \tag{3.54}$$

□

Corollary 3.6.9. *Let φ, ψ, and A be as above. Then $\varphi(G) \leq \mathrm{enl}_{\psi(G)}(\varphi(A))$.*

Proposition 3.6.5 is a special case of Proposition 3.6.8.

3.6.5 Combinatorial list-decodability, finite abelian to arbitrary

In this section, we establish that the class $\mathfrak{Abel}$ of finite abelian groups is universally CombEcon.

Throughout this section, let G be a finite abelian group, and H an arbitrary group (finite or infinite). Let $f \colon G \to H$ be a received word. Let $\varepsilon > 0$. Let $\mathcal{L} = \mathcal{L}(\mathrm{Hom}(G, H), f, \Lambda + \varepsilon)$ be the list (note that in this section we deal with the code of homomorphisms, rather than affine homomorphisms; however, we can convert between the two; see Section 3.4.5). The list $\mathcal{L}$ is divided into buckets $\mathcal{L}_\psi$ for $\psi \in \Psi$, where Ψ is as in Lemma 3.5.7.

We will see that there is a small set of abelian subgroups $M \leq H$ such that every $\varphi \in \mathcal{L}$ has its image in some M. Dinur, Grigorescu, Kopparty, and Sudan [15] proved that $\mathrm{aHom}(G, H)$ is CombEcon (and in fact, AlgEcon) for all finite abelian groups G and H.

Theorem 3.6.10 (DGKS 2008)**.** *The class $\mathfrak{Abel} \times \mathfrak{Abel}$ of pairs of abelian groups is CombEcon.*

Theorem 3.6.3 (which we prove in this section), combined with the DGKS result, lets us conclude that $\mathrm{Hom}(G, H)$ (and thus $\mathrm{aHom}(G, H)$) is CombEcon.

Corollary 3.6.11. *Finite abelian groups are universally CombEcon. Specifically, let C be a constant such that $\ell(\mathrm{aHom}(G, H), \Lambda + \varepsilon) \leq (\frac{1}{\varepsilon})^C$ for G, H finite abelian groups. Then $\ell(\mathrm{aHom}(G, H), \Lambda + \varepsilon) \leq O((\frac{1}{\varepsilon})^{C+4})$ for G a finite abelian group and H an arbitrary group.*

By [21, 11], the constant C currently stands at ≈ 105.

Proof of Corollary 3.6.11, assuming Theorem 3.6.3. Let $\mathcal{A}$ be the collection of subgroups of H guaranteed by Theorem 3.6.3; $|\mathcal{A}| \leq \frac{1}{4(2\Lambda+\varepsilon)\varepsilon^2} + \frac{1}{\varepsilon}$. Then, $\mathcal{L} \subseteq \bigcup_{M \in \mathcal{A}} \mathcal{L}(\mathrm{Hom}(G, M), f, \Lambda + \varepsilon)$ (on the right hand side we let f be redefined arbitrarily at points in its domain that do not map to M). So,

$$|\mathcal{L}| \leq \sum_{M \in \mathcal{A}} \ell(\mathrm{Hom}(G, M), \Lambda + \varepsilon) \leq \left(\frac{1}{4(2\Lambda + \varepsilon)\varepsilon^2} + \frac{1}{\varepsilon} \right) \left(\frac{1}{\varepsilon} \right)^C. \tag{3.55}$$

We then apply the reduction of affine to ordinary homomorphisms (Theorem 3.4.24). □

In the remainder of this subsection, we prove Theorem 3.6.3.

Let Ψ be as in the sphere-packing bound, Lemma 3.5.7. Recall our strategy from Section 3.5.4 of dividing the list $\mathcal{L}$ into buckets $\mathcal{L}_\psi$ for $\psi \in \Psi$. We prove the following.

Lemma 3.6.12. *Let $\psi \in \Psi$. There is a set $\mathcal{A}_\psi$ of finite abelian subgroups of H with $|\mathcal{A}_\psi| \leq \frac{1}{\varepsilon}$ such that for all $\varphi \in \mathcal{L}_\psi$, there is $M \in \mathcal{A}_\psi$ for which $\varphi(G) \leq M$.*

From this, Theorem 3.6.3 follows by taking $\mathcal{A} = \bigcup_{\psi \in \Psi} \mathcal{A}_\psi$.

The proof of Lemma 3.6.12 uses the following lemma.

Lemma 3.6.13. *Let $\psi \in \Psi$ and $\varphi \in \mathcal{L}_\psi$. Then* $\mathrm{depth}_G(\mathrm{Eq}(\psi, \varphi)) \leq 1$.

Proof. Let $p = 1/\Lambda$. Let N be the (G, H)-irrelevant kernel. By Fact 3.6.2, the smallest prime factor of $|G : N|$ is p. Since $N \leq \mathrm{Eq}(\psi, \varphi)$ this also means that $|G : \mathrm{Eq}(\psi, \varphi)|$ has

no prime factor smaller than p. On the other hand, since $\mathrm{agr}(\psi, \varphi) > \Lambda^2$, we have that $|G : \mathrm{Eq}(\psi, \varphi)| < p^2$. So, $|G : \mathrm{Eq}(\psi, \varphi)|$ has at most one prime factor. The result follows. □

Proof of Lemma 3.6.12. Let $\mathcal{A}_\psi = \{\mathrm{enl}_{\psi(G)}(\varphi(G)) \mid \varphi \in \mathcal{L}_\psi\}$. Then $\mathcal{A}_\psi$ is a set of finite abelian subgroups of H. And, for all $\varphi \in \mathcal{L}_\psi$, we have that $\varphi(G) \leq \mathrm{enl}_{\psi(G)}(\varphi(G)) \in \mathcal{A}_\psi$.

Let a be a uniform random element of G. For each $M \in \mathcal{A}_\psi$, let E_M be the event that $\mathrm{enl}_{\psi(G)}(f(a)) = M$. We will show that $\Pr[E_M] \geq \varepsilon$. Since the events E_M for $M \in \mathcal{A}_\psi$ are pairwise disjoint, this will imply that $|\mathcal{A}_\psi| \leq \frac{1}{\varepsilon}$.

Consider any $M \in \mathcal{A}_\psi$. There exists $\varphi \in \mathcal{L}_\psi$ such that $\mathrm{enl}_{\psi(G)}(\varphi(G)) = M$. If $a \in \mathrm{Eq}(f, \varphi)$ and $\langle \mathrm{Eq}(\psi, \varphi), a\rangle > \Lambda$, then in fact $\langle \mathrm{Eq}(\psi, \varphi), a\rangle = G$, and by Proposition 3.6.5,

$$\mathrm{enl}_{\psi(G)}(f(a)) = \mathrm{enl}_{\psi(G)}(\varphi(a)) = \mathrm{enl}_{\psi(G)}(\varphi(G)) = M. \tag{3.56}$$

By Lemma 3.6.13, $\mathrm{depth}_G(\mathrm{Eq}(\psi, \varphi)) \leq 1$. Note also that $\mu(\mathrm{Eq}(f, \varphi)) \geq \Lambda + \varepsilon$. So, by the random subgroup density lemma, Lemma 3.5.3 (or a direct calculation),

$$\Pr[E_M] \geq \Pr[a \in \mathrm{Eq}(f, \varphi) \text{ and } \langle \mathrm{Eq}(\psi, \varphi), a\rangle > \Lambda] \geq \varepsilon. \tag{3.57}$$

We conclude that $|\mathcal{A}_\psi| \leq \frac{1}{\varepsilon}$. □

Remark 3.6.14. Strictly speaking, we did not need to invoke the random subgroup density lemma in the proof of Lemma 3.6.12. We could have instead done a relatively short direct calculation. We reference Lemma 3.5.3 to emphasize the similarity between this step and the corresponding steps in in the first proof that alternating groups are universally CombEcon (specifically, in Lemma 3.7.3) and the proof that SRG groups are universally CombEcon (specifically, in Theorem 3.8.5).

3.6.6 Algorithm

For G a finite abelian group and H an arbitrary group, we can locally list-decode $\mathrm{aHom}(G, H)$. Based on our CombEcon bound for {abelian $\to$ arbitrary}, we adapt the {abelian $\to$ abelian} algorithm of Dinur, Grigorescu, Kopparty, and Sudan from [15, Sec. 5]. Thus, such codes are AlgEcon. Like [15], we assume that G is given explicitly by a primary decomposition (that is, as a direct product of cyclic groups of prime-power order; see Section 3.2.4).

Theorem 3.6.15. *Let $\mathfrak{D}$ be the class of pairs (G, H) where G is a finite abelian group given explicitly by an primary decomposition, and H is a group with black-box access. Then there is an algorithm to locally list-decode $\mathfrak{D}$ in time* $\mathrm{poly}(\log|G| \cdot \frac{1}{\varepsilon})$.

Here we assume the unit-cost model of naming elements of H (see Section 3.2.4).

We indicate how our our adaptation of the algorithm of Dinur et al. [15] differs from the original.

First, [15] reduces to the case where $H = \mathbb{Z}_{p^r}$. We do not make such a reduction. We let p be the prime such that $\Lambda = \frac{1}{p}$. Every mention of $\mathbb{Z}_{p^r}$ should be replaced by H. As in their algorithm, we take $G = G_1, \ldots, G_k$, with each $G_i = \mathbb{Z}_{p_i^{r_i}}$. We order the G_i such that $p_1 = p$. For them, the only important coordinates are the ones where $p_i = p$, but for our purposes, instances of $\mathbb{Z}_{p^{r_i}}$ should be replaced with $\mathbb{Z}_{p_i^{r_i}}$.

In the algorithm Extend of [15], the statement "If $c_1 - c_2$ is not divisible by p" should be replaced with "If $c_1 - c_2$ is not divisible by p_i, and if $f(y_1, c_1, s)$ and $f(y_2, c_2, s)$ commute with each other and with $\varphi(e_1), \ldots, \varphi(e_{i-1})$." Here e_j denotes a generator of G_j. The system of equations that follows should be solved under the assumption that the order of a divides $p_i^{r_i}$.

We note that when solving the system of equations in Extend,we are working in an abelian subgroup of H. Actually, even this does not matter; we can solve the given system of equations without assuming the elements of H commute.

In the algorithm as stated, we assume that the value $\Lambda_{G,H}$ is known. This assumption

can actually be discarded.

3.7 Alternating domain, combinatorial list-decoding

In this section, we will find that homomorphism codes with alternating domain are CombEcon. The exact constant is stated in Theorem 3.7.5. We remark that the degree of the poly$(1/\varepsilon)$-bound on list size is improved by the SRG methods of Section 3.8. The proof here uses the sphere-packing bound (Lemma 3.5.7), the shallow extension theorem (Theorem 3.5.1), Jordan-Liebeck (Theorem 3.2.19), and the length of of subgroup chains in symmetric groups (Corollary 3.2.27). For the latter two tools, refer back to Section 3.2.5 on the subgroup structure of the alternating groups.

In Section 3.7.1, we begin to carry out the strategy from Section 3.5.6. We present a method to prove CombEcon (Lemma 3.7.3) that bridges the gap (see Section 3.5.1) between the sphere-packing bound and the shallow extension theorem. In Section 3.7.2 we prove that alternating groups are universally CombEcon. Section 3.7.3 addresses the list-decoding radius of homomorphism codes with alternating domain, by exhibiting a "blowup" in list size when agreement is exactly Λ, or when radius is $(1 - \Lambda)$.

Throughout this section, we work with $\mathrm{Hom}(G, H)$ instead of $\mathrm{aHom}(G, H)$; by the reduction of affine to ordinary homomorphisms (Theorem 3.4.24), this suffices for showing CombEcon, and it does not worsen our degree bounds.

3.7.1 Sphere packing by shallow subgroups

In this section, we give the details of the sphere packing with label subgroups described in Section 3.5.6. Recall our core strategy from Section 3.5. We combine the sphere-packing bound (Lemma 3.5.7) with the shallow extension theorem (Theorem 3.5.1). The main impediment to simply composing these two tools to show CombEcon is the matter of *where* agreement occurs. The sphere-packing bound shows the existence of an $O(1/\varepsilon)$-size set Ψ

such that every φ in the list $\mathcal{L}$ has agreement $> \Lambda^2$ with some $\psi \in \Psi$. If we knew that the agreement occurred on a particular subgroup $K \leq G$, then we could apply the shallow extension theorem to the partial map $\gamma = \psi|_K$. As long as γ is a substantial partial map, this gives a good bound on the number of extensions of γ, and therefore bounds the size of the list $\mathcal{L}$. Unfortunately, the sphere-packing bound does not give the location of the agreement. However, in the case of alternating groups, there is a small (poly$(1/\varepsilon)$) set of "label subgroups" on which the agreement can occur. In this section, in Lemma 3.7.3, we bound the list size in terms of a set of label subgroups, and the depth of these subgroups. This approach depends very little on the codomain H.

Throughout this section, we let G be a finite group, H a group (finite or infinite), $\Lambda = \Lambda_{G,H}$ the maximum agreement, $f\colon G \to H$ a received word, $\varepsilon > 0$ a real, $\mathcal{L} = \mathcal{L}(\mathrm{Hom}(G,H), f, \Lambda + \varepsilon)$ the list, and $\Psi \subseteq \mathrm{Hom}(G,H)$ as defined in Lemma 3.5.7 (but with Hom in place of aHom).

Recall our strategy for proving CombEcon using the sphere-packing argument — We divided the list $\mathcal{L}$ into buckets $\mathcal{L}_\psi$ for $\psi \in \Psi$, where

$$\mathcal{L}_\psi = \{\varphi \in \mathcal{L}(\mathrm{Hom}(G,H), f, \Lambda + \varepsilon) \mid \mathrm{agr}(\psi, \varphi) > \Lambda^2\}. \tag{3.58}$$

We then split the bucket $\mathcal{L}_\psi$ into further *sub-buckets* according to the location of agreement with ψ. Each sub-bucket of $\mathcal{L}_\psi$ is labeled by a subgroup K of G which we call the *label subgroup*. We defined $\mathcal{L}_{\psi,K} \subseteq \mathcal{L}_\psi$ to be the subset of homomorphisms whose equalizer with ψ contains K; that is,

$$\mathcal{L}_{\psi,K} = \{\varphi \in \mathcal{L}_\psi \mid K \leq \mathrm{Eq}(\varphi, \psi)\}. \tag{3.59}$$

We concern ourselves now with the the set of label subgroups; we call such a set a *starting set*. Intuitively, a set $\mathcal{K}$ of subgroups is a starting set if the upper range of the subgroup lattice of G contains only supergroups of elements in $\mathcal{K}$. With an appropriate notion of "upper range," these starting sets form a sufficient set of label subgroups so that the sub-buckets

$\mathcal{L}_{\psi,K}$ cover the bucket $\mathcal{L}_\psi$ (see Remark 3.7.2).

Definition 3.7.1 ((G,λ)-starting-set)**.** Let $\mathcal{K}$ be a set of subgroups of G. Let $\lambda \in (0,1)$. We say that $\mathcal{K}$ is a (G,λ)-*starting-set* if

$$(\forall K \leq G)(\mu_G(K) > \lambda \Rightarrow (\exists S \in \mathcal{K})(S \leq K)). \tag{3.60}$$

Remark 3.7.2. Suppose that $\mathcal{K}$ is a (G,Λ^2)-starting set. Then, for any $f\colon G \to H$ and $\psi \in \mathrm{Hom}(G,H)$,

$$\mathcal{L}_\psi = \bigcup_{K\in\mathcal{K}} \mathcal{L}_{\psi,K}. \tag{3.61}$$

Combining this with the bucket-splitting lemma (Lemma 3.5.11),

$$\mathcal{L} = \bigcup_{\psi\in\Psi} \bigcup_{K\in\mathcal{K}} \mathcal{L}_{\psi,K}. \tag{3.62}$$

We can use Theorem 3.5.1 to bound the size of $\mathcal{L}_{\psi,K}$, so this allows us to bound the size of the list.

Lemma 3.7.3 (Sphere packing via shallow subgroups)**.** *Let G be a finite group, H a group, and $\varepsilon > 0$. Let $\mathcal{K}$ be a (G,Λ^2)-starting-set. Then,*

$$\ell(\mathrm{Hom}(G,H),\Lambda+\varepsilon) \leq \left(\frac{1}{4(2\Lambda+\varepsilon)\varepsilon}+1\right)\cdot \sum_{K\in\mathcal{K}} 1/\varepsilon^{\mathrm{depth}_G(K)}. \tag{3.63}$$

Proof. By Remark 3.7.2, we have that $|\mathcal{L}| \leq \sum_{\psi\in\Psi}\sum_{K\in\mathcal{K}}|\mathcal{L}_{\psi,K}|$. By the sphere-packing bound (Lemma 3.5.7), $|\Psi| \leq \frac{1}{4(2\Lambda+\varepsilon)\varepsilon}+1$. Since every element of $\mathcal{L}_{\psi,K}$ extends $\psi|_K$, by the shallow extension theorem (Theorem 3.5.1), $|\mathcal{L}_{\psi,K}| \leq 1/\varepsilon^{\mathrm{depth}_G(K)}$. □

We will use Lemma 3.7.3 in the proof that alternating groups are universally CombEcon.

3.7.2 Proof A_n is universally CombEcon

We prove that A_n is CombEcon by proving Theorem 3.7.5 below. First, we discuss the maximum agreement for alternating domains.

Lemma 3.7.4. *Let $n \geq 10$. Let $G = A_n$ and let H be a group. If $\Lambda_{G,H} \neq 0$, then either $\Lambda_{G,H} = 1/\binom{n}{2}$ or $\Lambda_{G,H} = 1/n$.*

Proof. We note that $\Lambda_{G,G} \geq 1/\binom{n}{2}$, since the identity automorphism of G and the automorphism that sends g to its conjugation by the transposition $(1\,2)$ agree on $G_{\{1,2\}}$ which has index $\binom{n}{2}$ (In fact, $\Lambda_{G,G} = 1/\binom{n}{2}$).

Suppose $\Lambda_{G,H} \neq 0$, so $\mathrm{Hom}(G,H)$ is nontrivial. Since A_n is simple, H contains an isomorphic copy of A_n (The image of a nontrivial homomorphism is isomorphic to A_n). So, $\Lambda_{G,H} \geq \Lambda_{G,G} \geq 1/\binom{n}{2}$. By Fact 3.2.5 and the Jordan-Liebeck Theorem (Theorem 3.2.19), $\Lambda_{G,H} = 1/\binom{n}{2}$ or $1/n$. □

We remark that Guo [20, Proposition 6.1] proved that $1/\binom{n}{2} \leq \Lambda_{A_n,A_n} \leq 1/n$ for $n \geq 5$.

Theorem 3.7.5. *For every group H, integer $n \geq 38$ and $\varepsilon > 0$, we find that*

$$\ell(\mathrm{Hom}(A_n, H), \Lambda_{A_n,H} + \varepsilon) \leq 1/\varepsilon^{11}. \tag{3.64}$$

Proof of Theorem 3.7.5. By Lemma 3.7.4, $\Lambda = \binom{n}{s}$ for $s \in \{1,2\}$. So, $\Lambda^2 \geq \binom{n}{2s+1}$.

We will use Lemma 3.7.3. We first define a starting set

$$\mathcal{K} = \{(A_n)_{(\Delta)} : \Delta \subseteq [n], |\Delta| = 2s\}. \tag{3.65}$$

That $\mathcal{K}$ is an (A_n, Λ^2)-starting-set follows by Jordan-Liebeck, Theorem 3.2.19. By Corollary 3.2.27, we find that $\mathrm{depth}_{A_n}(K) \leq \mathrm{depth}_{S_n}(A_{n-2s}) - 1 \leq 5$ for all $K \in \mathcal{K}$. Furthermore, $|\mathcal{K}| = \binom{n}{2s} < \binom{n}{s}^2/2 = \frac{1}{2\Lambda^2}$.

By Lemma 3.7.3, $\ell(\mathrm{Hom}(G,H), \Lambda + \varepsilon) \leq (\frac{1}{4(2\Lambda+\varepsilon)\varepsilon} + 1)(\frac{1}{2\Lambda^2})(\frac{1}{\varepsilon})^5$. By Lemma 3.5.9, this implies $\ell(\mathrm{Hom}(G,H), \Lambda + \varepsilon) \leq \frac{1}{2\varepsilon^{11}} + \frac{1}{\varepsilon^9} + \frac{1}{2\varepsilon^2}$. □

Corollary 3.7.6. *Alternating groups are universally CombEcon; in particular,* $\mathrm{aHom}(A_n, H)$ *is CombEcon of degree* 12.

The degree is improved by the methods of SRG groups; see Corollary 3.8.9.

3.7.3 Upper bound on list-decoding radius

We showed in Section 3.7.2 that $\mathfrak{Alt} \times \mathfrak{Groups}_{\text{finite or infinite}}$, and all of its subclasses, have list-decoding radius greater than $1 - (\Lambda + \varepsilon)$ for all $\varepsilon > 0$.

In contrast, $\mathfrak{Alt} \times \mathfrak{Groups}_{\text{finite or infinite}}$ and many of its subclasses have list-decoding radius at most $1 - \Lambda$. In this section, we demonstrate such a subclass. The number of homomorphisms within a closed ball of radius $1 - \Lambda$ of a received word will be exponential in $\log|G|$ and $\log|H|$. We note that $|H| \geq |G|$ unless $\Lambda = 0$.

Proposition 3.7.7. *For any* n*, and* $\lambda \in \{1/n, 1/\binom{n}{2}\}$*, there exists a finite group* H_n *such that* $\Lambda_{A_n,H_n} = \lambda$ *and*

$$\ell(\mathrm{Hom}(A_n, H_n), \Lambda) = 2^{\Omega(n)} \geq 2^{\Omega\left(\sqrt[3]{\log|H_n|}\right)}. \tag{3.66}$$

Moreover, for any fixed $n \geq 10$*, and any integer* M*, there is a finite group* H *such that*

$$\ell(\mathrm{Hom}(A_n, H), \Lambda) \geq M. \tag{3.67}$$

Proof. We use the same construction for both parts. To prove the first claim, let $k = n$. To prove the second claim, let $k \geq \log_2 M$.

Suppose $\lambda = 1/n$. Let $H_n = A_{n+1}^k$, the direct product of k copies of A_{n+1}. Then $\Lambda_{A_n,H_n} = 1/n$. Let $f\colon A_n \to H_n$ by $f(g) = (g, \ldots, g)$, the diagonal identity map, where A_n is embedded in A_{n+1}. For nonempty $S \subseteq [n]$ and $j \in [n]$, let $h = h(S, j) = (h_1, \ldots, h_k) \in H_n$, where h_i is the transposition $(j, n+1)$ if $i \in S$ and 1 otherwise. For each such h, let $\varphi_h \in \mathrm{Hom}(A_n, H_n)$ be given by $\varphi_h(g) = h^{-1}f(g)h$. Each φ_h has agreement $\mathrm{agr}(\varphi_h, f) = 1/n = \Lambda$

with f. There are $n(2^k-1)$ such h, so $\ell(\mathrm{Hom}(A_n, H_n), \Lambda) \geq n(2^k-1)$.

Suppose $\lambda = 1/\binom{n}{2}$. Let $H_n = A_n^k$. Then, $\Lambda_{A_n,H_n} = 1/\binom{n}{2}$. Let $f\colon A_n \to H_n$ by $f(g) = (g, \ldots, g)$, the diagonal identity map. For nonempty $S \subseteq [n]$ and $\tau \in S_n$ is a transposition, let $h = h_{S,\tau} = (h_1, \ldots, h_k) \in A_n^k$, where $h_i = \tau$ if $i \in S$ and 1 otherwise. For each such h, let $\varphi_h \in \mathrm{Hom}(A_n, H_n)$ be given by $\varphi_h(g) = h^{-1}f(g)h$. Each φ_h has agreement $\mathrm{agr}(\varphi_h, f) = 1/\binom{n}{2}$ with f. There are $\binom{n}{2}(2^k-1)$ such h, so $\ell(\mathrm{Hom}(A_n, H_n), \Lambda) \geq \binom{n}{2}(2^k-1)$. □

We remark that $\ell(\mathrm{Hom}(A_n, H), \Lambda_{A_n,H})$ is not bounded as a function of n for a wide variety of classes of H.

3.8 Shallow random generation

In this section, we prove results about SRG groups, defined in Section 3.3.4), for which few random elements tend to generate a shallow (low-depth) subgroup. In Section 3.8.1, we will show that alternating groups are SRG. In Section 3.8.3 we will prove that SRG groups are also "KLC" groups (another generation property, defined in Section 3.8.2). In Section 3.8.4, we prove that KLC groups (and thus SRG groups, and thus alternating groups) are universally CombEcon. In Section 3.8.5 we prove KLC groups are universally CertEcon.

Recall that by the reduction of affine to ordinary homomorphisms (Theorem 3.4.24), the code $\mathrm{Hom}(G, H)$ is CombEcon if and only if $\mathrm{aHom}(G, H)$ is CombEcon, and similarly for CertEcon and AlgEcon under modest assumptions of the representation of the groups. All our results about SRG groups (universal CombEcon and universal CertEcon) will take advantage of this equivalence, as our proofs will argue about $\mathrm{Hom}(G, H)$ instead of $\mathrm{aHom}(G, H)$. To reflect this, concepts in this section are defined in terms of subgroup generation using $\langle\cdot\rangle$ instead of affine generation using $\langle\cdot\rangle_{\mathsf{aff}}$. If we were to instead reason directly about aHom, it would not improve the degrees of our CombEcon and CertEcon results.

3.8.1 Alternating groups are SRG

In this subsection, we prove that the class $\mathfrak{Alt}$ of alternating groups is SRG.

Theorem 3.8.1. *The class of alternating groups is SRG. In fact, for all $k \geq 2$ there is an integer n_k such that for all $n \geq n_k$, the alternating group A_n is $(k, 3k-1)$-shallow generating.*

Moreover, there is an integer n_0 such that for all $k \geq 2$ and all $n \geq \max\{n_0, (3k)^3\}$, the alternating group A_n is $(k, (9/2)k - 1)$-shallow generating.

In particular, for $k = 2$, the alternating group A_n is $(2, 5)$-shallow generating, justifying Lemma 3.3.11.

The proof uses facts about the subgroup structure of the alternating groups; see Section 3.2.5.

Proof. We prove the first part.

Let π, σ be a pair of independent uniform random elements from A_n. Let $E(n, k)$ be the event defined in Theorem 3.2.20. By Theorem 3.2.20 and the remarks that follow it, if $E(n, 2k)$ occurs, then $\langle \pi, \sigma \rangle$ has as a subgroup the alternating group on $n - 2k$ elements of the permutation domain. In this case, then by Corollary 3.2.27, $\mathrm{depth}_{A_n}(\langle \pi, \sigma \rangle) \leq \mathrm{depth}_{S_n}(A_{n-2k}) - 1 \leq 3k - 1$.

So, for sufficiently large n,

$$\Pr_{\pi,\sigma \in A_n}[\mathrm{depth}_{A_n}(\langle \pi, \sigma \rangle) > 3k-1] \leq \Pr_{\pi,\sigma \in A_n}[\neg E(n, 2k)] \leq \frac{5}{\binom{n}{2k+1}} \leq \frac{1}{\binom{n}{2}^k} = (\Lambda^*_{A_n})^k. \quad (3.68)$$

It follows that A_n is $(k, 3k - 1)$-shallow generating.

The proof of the second part is similar, replacing $E(n, 2k)$ by $E(n, 3k)$. By Corollary 3.2.27, the bound on $\mathrm{depth}_{A_n}(\langle \pi, \sigma \rangle)$ changes from $3k - 1$ to $(9/2)k - 1$. The probability in Equation (3.68) changes from $5/\binom{n}{2k+1}$ to $5/\binom{n}{3k+1}$ (this bound holds for $n \geq \max\{n_0, k^2\}$), and the latter quantity is $\leq 1/\binom{n}{2}^k$ if $n \geq \max\{n_0, (3k)^3\}$. □

Consequences. In the remainder of Section 3.8, we prove facts that hold for all SRG groups. Here are the implications of those facts for alternating groups.

From Theorem 3.8.5 and Theorem 3.8.7, setting $k = 2$, we find that $\mathrm{Hom}(A_n, H)$ is CombEcon with degree 7; i. e., $\ell(\mathrm{Hom}(A_n, H), \Lambda + \varepsilon) < 1/\varepsilon^7$ for all groups H (finite or infinite). By the reduction of affine to ordinary homomorphisms (Theorem 3.4.24, Corollary 3.4.31), $\mathrm{aHom}(A_n, H)$ is CombEcon with degree 8 (see Corollary 3.8.9). We remark that the degree 8 can be improved to 7. This is by going through the proof with depth K redefined to be the maximal length of a subgroup chain from K to a subgroup of density $> \Lambda$.

By Theorem 3.8.11, alternating groups are universally CertEcon. More specifically, the class of alternating groups is universally strong certificate-list-decodable using $O(\log(1/\varepsilon)/\varepsilon^8)$ queries and computation time. Certificates are generated by querying the received word on sets of uniform size (see Section 3.8.5).

3.8.2 Definition of KLC, subset-generation

In this section we define a useful technical condition on classes of groups which we call "KLC". All SRG classes of groups are KLC (shown in the next section). We can more directly prove that KLC classes of groups are universally CombEcon and CertEcon, and our "SRG implies universally CombEcon and CertEcon" results are proved by way of the KLC property (Sections 3.8.4 and 3.8.5).

Definition 3.8.2 ((k, λ, c)-subset-generated)**.** Let G be a finite group, k a nonnegative integer, $0 \leq \lambda < 1$, and $c \geq 0$. We say that G is (k, λ, c)-*subset-generated* if, for all subsets $S \subseteq G$ with $\mu(S) > \lambda$, we have that

$$\Pr_{s_1,\ldots,s_k \in S}[\mu(\langle s_1, \ldots, s_k \rangle) > \lambda] \geq \left(1 - \frac{\lambda}{\mu(S)}\right)^c, \tag{3.69}$$

where $s_1, \ldots, s_k$ are chosen independently and uniformly from S.

Note that, if we define $\varepsilon = \mu(S) - \lambda$, then $1 - \frac{\lambda}{\mu(S)} = \frac{\varepsilon}{\lambda+\varepsilon}$, so Equation (3.69) mirrors the expression of the random subgroup density lemma (Lemma 3.5.3).

We say that G is (k, λ, c)-*affine-generated* if it satisfies Definition 3.8.2 but with $\langle s_1, \ldots, s_k \rangle$ replaced by $\langle s_1, \ldots, s_k \rangle_{\mathsf{aff}}$.

Definition 3.8.3 (KLC)**.** Let $\mathfrak{G}$ be a class of finite groups. We say that $\mathfrak{G}$ is *KLC* if there exists a positive integer k and a constant $c > 0$ such that, for all $G \in \mathfrak{G}$ and for all groups H, we have that G is $(k, \Lambda_{G,H}, c)$-subset-generated.

The notion of "KLC-affine" can be defined analogously. But, the two notions are equivalent; see Remark 3.8.4 (c) below.

We make a few remarks on (k, λ, c)-subset-generated groups.

Remark 3.8.4. (a) For every $k \geq 1$ and $c \geq 0$, the class $\mathfrak{Groups}_{\text{finite}}$ of all finite groups is $(k, 0, c)$-subset-generated.

(b) The property of being a (k, λ, c)-subset-generated class of groups is monotone in both k and c. Specifically, for $k' > k$ and $c' > c$, if G is (k, λ, c)-subset-generated, then G is also (k', λ, c)-subset-generated and (k, λ, c')-subset-generated.

(c) If G is (k, λ, c)-affine-generated, then it is (k, λ, c)-subset-generated. If G is (k, λ, c)-affine-generated, then it is $(k + 1, \lambda, c)$-subset-generated.

3.8.3 SRG implies KLC

Theorem 3.8.5 (SRG implies KLC)**.** *If a class $\mathfrak{G}$ of groups is SRG, then $\mathfrak{G}$ is KLC.*

In particular, let $k, d \in \mathbb{N}$, and let G be a (k, d)-shallow generating finite group. Then G is $(k + d, \lambda, d + 1)$-subset-generated for all $\lambda \geq \Lambda^*_G$.

After rearranging the definition of (k, d)-shallow generating (which we do in Lemma 3.8.6), we prove Theorem 3.8.5 by applying the random subgroup density lemma (Lemma 3.5.3).

For a k-tuple $\mathbf{g} = (g_1, \ldots, g_k)$, we write $\langle \mathbf{g} \rangle$ to mean $\langle g_1, \ldots, g_k \rangle$.

Lemma 3.8.6. *Let $k, d \in \mathbb{N}$. Let G be a (k,d)-shallow generating finite group. Then, for all $\lambda \geq \Lambda_G^*$ and all $S \subseteq G$ with $\mu(S) > \lambda$,*

$$\Pr_{\mathbf{s} \in S^k}[\mathrm{depth}_G(\langle \mathbf{s} \rangle) \leq d] > \frac{\mu(S) - \lambda}{\mu(S)}. \tag{3.70}$$

Proof of Lemma 3.8.6. By the definition of (k,d)-shallow generating,

$$\Pr_{\mathbf{g} \in G^k}[\mathrm{depth}_G(\langle \mathbf{g} \rangle) > d] < \lambda^k. \tag{3.71}$$

So (taking probabilities to be uniform over $\mathbf{g} \in G^k$ when not otherwise specified),

$$\Pr_{\mathbf{s} \in S^k}[\mathrm{depth}_G(\langle \mathbf{s} \rangle) > d] = \frac{\Pr[\mathbf{g} \in S^k \text{ and } \mathrm{depth}_G(\langle \mathbf{g} \rangle) > d]}{\Pr[\mathbf{g} \in S^k]} \tag{3.72}$$

$$\leq \frac{\Pr[\mathrm{depth}_G(\langle \mathbf{g} \rangle) > d]}{\Pr[\mathbf{g} \in S^k]} \tag{3.73}$$

$$< \frac{\lambda^k}{\mu(S)^k} \leq \frac{\lambda}{\mu(S)}. \tag{3.74}$$

□

Proof of Theorem 3.8.5. We check the definition of $(k+d, \lambda, d+1)$-subset-generated. Let $S \subseteq G$ such that $\mu(S) > \lambda$ and let $\varepsilon = \mu(S) - \lambda$.

Observe that

$$\Pr_{\mathbf{s} \in S^k, \mathbf{t} \in S^d}[\mu(\langle \mathbf{s}, \mathbf{t} \rangle) > \lambda] \geq \Pr_{\mathbf{s} \in S^k, \mathbf{t} \in S^d}[\mu(\langle \mathbf{s}, \mathbf{t} \rangle) > \lambda \mid \mathrm{depth}_G(\langle \mathbf{s} \rangle) \leq d] \cdot \Pr_{\mathbf{s} \in S^k}[\mathrm{depth}_G(\langle \mathbf{s} \rangle) \leq d].$$

By the random subgroup density lemma (Lemma 3.5.3) with $K = \langle \mathbf{s} \rangle$, the first factor on the right hand side is bounded by

$$\Pr_{\mathbf{s} \in S^k, \mathbf{t} \in S^d}[\mu(\langle \mathbf{s}, \mathbf{t} \rangle) > \lambda \mid \mathrm{depth}_G(\langle \mathbf{s} \rangle) \leq d] > \left(\frac{\varepsilon}{\lambda + \varepsilon}\right)^d. \tag{3.75}$$

By Lemma 3.8.6, the second factor is bounded by

$$\Pr_{\mathbf{s}\in S^k}[\mathrm{depth}_G(\langle \mathbf{s}\rangle) \le d] > \frac{\varepsilon}{\lambda+\varepsilon}. \tag{3.76}$$

Thus,

$$\Pr_{\mathbf{s}\in S^k,\mathbf{t}\in S^d}\left[\mu\left(\langle \mathbf{s},\mathbf{t}\rangle\right) > \lambda\right] > \left(\frac{\varepsilon}{\lambda+\varepsilon}\right)^{d+1}, \tag{3.77}$$

so G is $(k+d, \lambda, d+1)$-subset-generated. □

3.8.4 KLC implies CombEcon

Theorem 3.8.7. *If $\mathfrak{G}$ is a KLC class of finite groups, then $\mathfrak{G}$ is universally CombEcon.*

More precisely, let $k \in \mathbb{N}$, let $c > 0$, let $G \in \mathfrak{G}$, and let H be a group. If G is $(k, \Lambda_{G,H}, c)$-subset-generated, then $\ell(\mathrm{Hom}(G,H), \Lambda_{G,H}+\varepsilon) \le 1/\varepsilon^{\max\{c,k\}}$ for all $\varepsilon > 0$.

Since SRG classes of groups are KLC, this implies the following.

Corollary 3.8.8. *If $\mathfrak{G}$ is an SRG class of finite groups, then $\mathfrak{G}$ is universally CombEcon.*

Corollary 3.8.9. *Alternating groups are universally CombEcon; in particular,* $\mathrm{aHom}(A_n, H)$ *is CombEcon of degree* 8.

Recall from Section 3.1.5 that a *domain certificate* is a certificate that is a restriction $f|_S$ of the received word f to a subset S of the domain. Computationally, we represent a domain certificate $f|_S$ by its domain, S. Recall from Section 3.1.5 that a $\mathcal{W}_\Lambda$*-certificate* is a certificate whose domain generates a group of density $> \Lambda$. Theorem 3.8.7 follows from the following lemma.

Lemma 3.8.10. *Let G be a finite group, H a group, $f\colon G \to H$ a received word, $\varepsilon > 0$, and $\varphi \in \mathcal{L}(\mathrm{Hom}(G,H), f, \Lambda+\varepsilon)$. If G is $(k, \Lambda_{G,H}, c)$-subset-generated, and S is a set of k uniform, independent random elements of G, then $f|_S$ is a domain $\mathcal{W}_\Lambda$-certificate for φ with probability $\ge \varepsilon^{\max\{c,k\}}$.*

Proof of Lemma 3.8.10. Let $g_1, \ldots, g_k$ be uniform, independent random elements of G. Let $S = \{g_1, \ldots, g_k\}$. If $S \subseteq \mathrm{Eq}(f, \varphi)$ and $\mu(\langle S \rangle) > \Lambda$, then $f|_S$ is a domain $\mathcal{W}_\Lambda$-certificate for φ. This happens with probability at least

$$\begin{aligned}
\Pr\left[S \subseteq \mathrm{Eq}(\psi, f) \text{ and } \mu\left(\langle S \rangle\right) > \Lambda\right] &= \Pr\left[\mu\left(\langle S \rangle\right) > \Lambda \,\middle|\, S \subseteq \mathrm{Eq}(\psi, f)\right] \cdot \Pr[S \subseteq \mathrm{Eq}(\psi, f)] \\
&> \left(\frac{\varepsilon}{\Lambda + \varepsilon}\right)^c \cdot (\Lambda + \varepsilon)^k \\
&\geq \varepsilon^{\max\{c,k\}}.
\end{aligned}$$

□

Since a set is a domain certificate for at most one homomorphism, Lemma 3.8.10 implies Theorem 3.8.7.

3.8.5 KLC implies CertEcon

Our CertEcon result for KLC classes of groups builds on our CombEcon proof.

Theorem 3.8.11. *If $\mathfrak{G}$ is a KLC class of finite groups, then $\mathfrak{G}$ is universally strong $\mathcal{W}_\Lambda$-CertEcon via domain certificates.*

Access 3.8.12. We assume that all groups in $\mathfrak{G}$ are encoded groups, and that (nearly) uniform elements of G are provided. To generate the domain-$\mathcal{W}_\Lambda$-certificate-list, because we represent a domain certificate by its domain, we need access only the domain, only in the ability to generate random elements. No knowledge of H is required. The dependence on H appears only in the the assumption that G is $(k, \Lambda_{G,H}, c)$-subset generated. Knowledge of $\Lambda_{G,H}$ is also not required.

To evaluate the certificates (instead of simply representing them as a subset of the domain), we assume that we have oracle access to the entries of the received word.

Again, since SRG classes of groups are KLC, this implies the following.

Corollary 3.8.13. *If $\mathfrak{G}$ is an SRG class of finite groups, then $\mathfrak{G}$ is universally CertEcon (with access as described in Theorem 3.8.11).*

Theorem 3.8.11 follows from Lemma 3.8.10 together with the equivalence of CertEcon and individual-certificate CertEcon (Corollary 3.4.2). These two lemmas imply the following CertEcon algorithm for aHom(G, H).

Algorithm 3.8.14. Let $b = \max\{c, k\} + 1$. To generate the domain-$\mathcal{W}_\Lambda$-certificate-list, independently choose $\left\lceil \frac{1}{\varepsilon^b} \ln\left(\frac{4}{\varepsilon^b}\right) \right\rceil$ subsets of G, where each subset consists of b uniform, independent random elements of G.

To then evaluate these $\mathcal{W}_\Lambda$-certificates, query f on the domain $\mathcal{W}$-certificates, an additional poly$(1/\varepsilon)$ queries to f.

Remark 3.8.15 (Amount of work). For a class of codes to be strong CertEcon, the amount of work is required to be poly$(\log|G|, 1/\varepsilon)$ in the unit cost model, but in fact, Algorithm 3.8.14 only requires poly$(1/\varepsilon)$ amount of work (no dependency on $|G|$).[2]

3.9 $\Lambda_{G,H}$ when G or H is solvable

We give a combinatorial description of $\Lambda_{G,H}$ when G is a finite abelian group and H is an arbitrary group.

Proposition 3.9.1. *Let G be a finite abelian group and H a group. Then $\Lambda_{G,H} = 1/p$, where p is the smallest prime number such that p divides $|G|$ and H has an element of order p. If no such p exists, then $|\mathrm{Hom}(G, H)| = 1$ and $\Lambda_{G,H} = 0$.*

This proposition is a special case of the following theorem, which describes $\Lambda_{G,H}$ when G or H is a solvable group. This is a slight generalization of a result of Guo [20, Theorem 1.1].

Theorem 3.9.2. *Let G be a finite group and H a group, such that at least one of G or H is solvable. Then $\Lambda_{G,H} = 1/p$, where p is the smallest prime number such that G has*

2. Two incomparable sufficient conditions for the access model to G are black-box access and polycyclic presentations. In a black-box group, ε-uniform elements can be generated in polynomial time [3]. Given a polycyclic presentation, exactly uniform elements can be generated.

a normal subgroup of index p and H has an element of order p. If no such p exists, then $|\mathrm{Hom}(G,H)| = 1$ *and* $\Lambda_{G,H} = 0$.

We will prove Theorem 3.9.2 in this subsection. Guo proved Theorem 3.9.2 in the case where H is finite, and either G is solvable or H is nilpotent. Our proof relies on the following lemma about $\bigcap_{\varphi\in\mathrm{Hom}(G,H)} \ker\varphi$.

Lemma 3.9.3. *Let G be a finite group and H a group. Let $K = \bigcap_{\varphi\in\mathrm{Hom}(G,H)} \ker\varphi$. Then every prime factor of $|G:K|$ is the order of an element of H.*

Proof. Consider any prime factor p of $|G:K|$. Then there is $g \in G$ such that gK has order p in G/K. Since $g \notin K$, there is $\varphi \in \mathrm{Hom}(G,H)$ such that $g \notin \ker\varphi$. We have $g^p \in K$, so $\varphi(g)^p \in \varphi(K) = 1$, so $|\varphi(g)|$ divides p. Since $\varphi(g) \neq 1$, we have $|\varphi(g)| = p$. □

We also use the following well-known fact and a theorem by Berkovich (see [23, Problem 3B.15]).

Fact 3.9.4. In a solvable group, a normal subgroup is a maximal normal subgroup if and only if it has prime index.

Theorem 3.9.5 (Berkovich)**.** *Let G be a finite solvable group and K a proper subgroup of smallest index. Then $K \trianglelefteq G$.*

Next, we prove Theorem 3.9.2 in the case when G is solvable. Guo [20, Theorem 5.5] proved this in the case that also H is finite. Guo's proof can be modified slightly to also accommodate infinite groups. We include a compact proof for completeness.

Proof of Theorem 3.9.2 in the case where G is solvable. Let $K = \bigcap_{\varphi\in\mathrm{Hom}(G,H)} \ker\varphi$. If $\Lambda > 0$, then there is a nontrivial homomorphism $\varphi \in \mathrm{Hom}(G,H)$. Then $\ker\varphi$ is a proper normal subgroup of G, so is contained in a maximal normal subgroup M. By Fact 3.9.4, $|G:M|$ is prime, and since $M \geq \ker\varphi \geq K$, we have that $|G:M|$ divides $|G:K|$. So, by Lemma 3.9.3, we have that H has an element of order $|G:M|$. So, if $\Lambda > 0$, then p exists.

Henceforth, assume p exists. Let N be a normal subgroup of G of index p, and h an element of H of order p. We show that $\Lambda \geq 1/p$ by exhibiting a pair of homomorphisms that achieve this agreement. Let $\varphi_1 \colon G \to H$ be the trivial homomorphism. Since $|G/N| = |\langle h \rangle| = p$ prime, there is a group isomorphism $G/N \to \langle h \rangle$. This lifts to a group homomorphism $\varphi_2 \colon G \to \langle h \rangle$. Then $\mathrm{Eq}(\varphi_1, \varphi_2) = N$, so $\mathrm{agr}(\varphi_1, \varphi_2) = 1/p$. So, $\Lambda_{G,H} \geq 1/p$.

We next show that $\Lambda \leq 1/p$. Since $N = \ker \varphi_2 \geq K$, we have that $N/K \trianglelefteq G/K$ and $|G/K : N/K| = p$. Furthermore, we claim that N/K is a proper normal subgroup of smallest prime index in G/K — if there were a proper normal $\hat{N}/K$ of prime index $q < p$ (with $K \leq \hat{N} \leq G$), then $\hat{N}$ would be a normal subgroup of G of index q, and H would have an element of order q by Lemma 3.9.3, which would contradict the definition of p. By Fact 3.9.4, N/K is in fact a proper normal subgroup of smallest index in G/K (removing "prime"). By Theorem 3.9.5, we further have that N/K is a proper subgroup of smallest index in G/K (removing "normal"). Thus, N is the has the smallest index of any subgroup of G that contains K. Any equalizer of two homomorphisms in $\mathrm{Hom}(G, H)$ contains K, so no equalizer can have smaller index than N. So, $\Lambda \leq \mu(N) = 1/p$. □

To prove Theorem 3.9.2 in the case where H is solvable, we use the following fact.

Lemma 3.9.6. *Let G be a group and H a solvable group. Let $K = \bigcap_{\varphi \in \mathrm{Hom}(G,H)} \ker \varphi$. Then G/K is solvable.*

Proof. For each $\varphi \in \mathrm{Hom}(G, H)$, we have that $\varphi(G)$ is solvable with derived length at most the derived length of H. Also, $G/\ker \varphi \cong \varphi(G)$. So, $\prod_{\varphi \in \mathrm{Hom}(G,H)} G/\ker \varphi$ is a direct product of solvable subgroups with bounded derived length, so is solvable. Let $\psi \colon G \to \prod_{\varphi \in \mathrm{Hom}(G,H)} G/\ker \varphi$ be the projection onto each coordinate. Then $\ker \psi = K$. And, $\psi(G)$ is a subgroup of a solvable group, so is solvable. Thus, $G/K = G/\ker \psi \cong \psi(G)$ is solvable. □

We can now prove Theorem 3.9.2 in the case where H is solvable.

Proof of Theorem 3.9.2 in the case where H is solvable. Let $K = \bigcap_{\varphi \in \mathrm{Hom}(G,H)} \ker \varphi$, the (G, H)-irrelevant kernel. Let p be the smallest prime divisor of $|G|$ such that G has a normal subgroup of index p and H has an element of order p. If N is a normal subgroup of prime index and H contains an element h of order $|G : N|$, then $K \leq N$, since the isomorphism $G/N \to \langle h \rangle$ lifts to a homomorphism $G \to \langle h \rangle$ with kernel N. So, p is the smallest prime index of a normal subgroup of G that contains K. So, p is the smallest prime index of a normal subgroup of G/K.

We have that G/K is solvable by Lemma 3.9.6. So, the case of Theorem 3.9.2 in which the domain is solvable, we have that $\Lambda_{G/K,H} = 1/p$. Then, by Corollary 3.4.15, we have $\Lambda_{G,H} = \Lambda_{G/K,H} = 1/p$. □

REFERENCES

[1] László Babai. On the length of subgroup chains in the symmetric group. *Communications in Algebra*, 14(9):1729–1736, 1986. doi:10.1080/00927878608823393.

[2] László Babai. The probability of generating the symmetric group. *J. Combinat. Theory, Series A*, 52(1):148 – 153, 1989. doi:10.1016/0097-3165(89)90068-X.

[3] László Babai. Local expansion of vertex-transitive graphs and random generation in finite groups. In *23rd STOC*, pages 164–174. ACM, 1991. doi:10.1145/103418.103440.

[4] László Babai, Anandam Banerjee, Raghav Kulkarni, and Vipul Naik. Evasiveness and the distribution of prime numbers. In *STACS 2010: 27th Internat. Symp. on Theoretical Aspects of Comp. Sci.*, pages 71–82. Schloss Dagstuhl. Leibniz-Zent. Inform., 2010. doi:10.4230/LIPIcs.STACS.2010.2445.

[5] László Babai, Robert Beals, and Ákos Seress. Polynomial-time theory of matrix groups. In *41st STOC*, pages 55–64. ACM, 2009. doi:10.1145/1536414.1536425.

[6] László Babai, Timothy J. F. Black, and Angela Wuu. List-decoding homomorphism codes with arbitrary codomains. In *Approximation, Randomization, and Combinatorial Optimization. Algorithms and Techniques, APPROX/RANDOM 2018*, pages 29:1–29:18. Schloss Dagstuhl. Leibniz-Zent. Inform., 2018. doi:10.4230/LIPIcs.APPROX-RANDOM.2018.29.

[7] László Babai and Endre Szemerédi. On the complexity of matrix group problems I. In *25th FOCS*, pages 229–240. IEEE Computer Soc., 1984. doi:10.1109/SFCS.1984.715919.

[8] Robert Beals, Charles R. Leedham-Green, Alice C. Niemeyer, Cheryl E. Praeger, and Ákos Seress. Constructive recognition of finite alternating and symmetric groups acting as matrix groups on their natural permutation modules. *J. Algebra*, 292(1):4 – 46, 2005. doi:10.1016/j.jalgebra.2005.01.035.

[9] Abhishek Bhowmick and Shachar Lovett. The list decoding radius of Reed-Muller codes over small fields. In *47th STOC*, pages 277–285. ACM, 2015. doi:10.1145/2746539.2746543.

[10] Timothy Black. Monotone properties of k-uniform hypergraphs are weakly evasive. *ACM Trans. Comput. Theory*, 11(3):14:1–14:14, April 2019. doi:10.1145/3313908.

[11] Timothy Black, Alan Guo, Madhu Sudan, and Angela Wuu. List decoding group homomorphisms to nilpotent groups. In preparation, 2019.

[12] Peter J. Cameron. Finite permutation groups and finite simple groups. *Bulletin of the London Mathematical Society*, 13(1):1–22, 1981. doi:10.1112/blms/13.1.1.

[13] Peter J Cameron, Ron Solomon, and Alexandre Turull. Chains of subgroups in symmetric groups. *Journal of Algebra*, 127(2):340 – 352, 1989. doi:10.1016/0021-8693(89)90256-1.

[14] Amit Chakrabarti, Subhash Khot, and Yaoyun Shi. Evasiveness of subgraph containment and related properties. *SIAM J. Comput.*, 31(3):866–875, 2002. doi:10.1137/S0097539700382005.

[15] Irit Dinur, Elena Grigorescu, Swastik Kopparty, and Madhu Sudan. Decodability of group homomorphisms beyond the Johnson bound. In *40th STOC*, pages 275–284. ACM, 2008. doi:10.1145/1374376.1374418.

[16] John D. Dixon and Brian Mortimer. *Permutation Groups*. Graduate Texts in Math. Springer New York, 1996. doi:10.1007/978-1-4612-0731-3.

[17] Suixiang Gao and Guohui Lin. Decision tree complexity of graph properties with dimension at most 5. *J. Comput. Sci. Tech.*, 15(5):416–422, 2000. doi:10.1007/BF02950404.

[18] Oded Goldreich and Leonid A. Levin. A hard-core predicate for all one-way functions. In *21st STOC*, pages 25–32. ACM, 1989. doi:10.1145/73007.73010.

[19] Elena Grigorescu, Swastik Kopparty, and Madhu Sudan. Local decoding and testing for homomorphisms. In *Approximation, Randomization, and Combinatorial Optimization. Algorithms and Techniques, APPROX/RANDOM 2006*, pages 375–385, 2006. doi:10.1007/11830924_35.

[20] Alan Guo. Group homomorphisms as error correcting codes. *Electronic J. Combinatorics*, 22(1):P1.4, 2015.

[21] Alan Guo and Madhu Sudan. List decoding group homomorphisms between supersolvable groups. In *Approximation, Randomization, and Combinatorial Optimization. Algorithms and Techniques, APPROX/RANDOM 2014*, pages 737–747. Schloss Dagstuhl. Leibniz-Zent. Inform., 2014. doi:10.4230/LIPIcs.APPROX-RANDOM.2014.737.

[22] Marc Hellmuth, Lydia Ostermeier, and Peter F. Stadler. A survey on hypergraph products. *Math. Comput. Sci.*, 6(1):1–32, 2012. doi:10.1007/s11786-012-0109-6.

[23] I. Martin Isaacs. *Finite Group Theory*. Graduate studies in mathematics. Amer. Math. Soc., 2008.

[24] Jeff Kahn, Michael Saks, and Dean Sturtevant. A topological approach to evasiveness. *Combinatorica*, 4(4):297–306, 1984. doi:10.1007/BF02579140.

[25] Daniel J. Kleitman and David J. Kwiatkowski. Further results on the Aanderaa-Rosenberg conjecture. *J. Combin. Theory Ser. B*, 28(1):85–95, 1980. doi:10.1016/0095-8956(80)90057-X.

[26] Raghav Kulkarni. Evasiveness through a circuit lens. In *Innovations in Theoret. Comp. Sci., ITCS '13*, pages 139–144, 2013. doi:10.1145/2422436.2422454.

[27] Raghav Kulkarni, Youming Qiao, and Xiaoming Sun. Any monotone property of 3-uniform hypergraphs is weakly evasive. In *Theory and Appl. of Models of Computation*, volume 7876 of *Lect. Notes in Comp. Sci.*, pages 224–235. Springer Berlin Heidelberg, 2013. doi:10.1007/978-3-642-38236-9_21.

[28] Robert Oliver. Fixed-point sets of group actions on finite acyclic complexes. *Comment. Math. Helv.*, 50:155–177, 1975. doi:10.1007/BF02565743.

[29] Ronald L. Rivest and Jean Vuillemin. On recognizing graph properties from adjacency matrices. *Theoret. Comput. Sci.*, 3(3):371–384, 1976/77. doi:10.1016/0304-3975(76)90053-0.

[30] Derek J. S. Robinson. *A Course in the Theory of Groups*. Springer, 2nd edition, 1995. doi:10.1007/978-1-4419-8594-1.

[31] Arnold L. Rosenberg. On the time required to recognize properties of graphs: A problem. *SIGACT News*, 5(4):15–16, October 1973. doi:10.1145/1008299.1008302.

[32] Joseph J. Rotman. *An Introduction to the Theory of Groups*. Springer-Verlag, New York, 1995. doi:10.1007/978-1-4612-4176-8.

[33] Ákos Seress. *Permutation Group Algorithms*. Cambridge Tracts in Math. Cambridge Univ. Press, 2003. doi:10.1017/cbo9780511546549.

[34] Eberhard Triesch. On the recognition complexity of some graph properties. *Combinatorica*, 16(2):259–268, 1996. doi:10.1007/BF01844851.

[35] Ivan Matveevich Vinogradov. *The Method of Trigonometrical Sums in the Theory of Numbers (Russian). Trav. Inst. Math. Stekloff*, 10, 1937.

[36] Angela Wuu. *List-Decoding Homomorphism Codes*. PhD thesis, University of Chicago, 2018. doi:10.6082/m0h1-7108.

[37] Angela Wuu. Homomorphism extension. *arXiv*, arXiv:1802.08656, 2018.

[38] Andrew Chi-Chih Yao. Monotone bipartite graph properties are evasive. *SIAM J. Comput.*, 17(3):517–520, 1988. doi:10.1137/0217031.