

# List-Decoding Homomorphism Codes with Arbitrary Codomains

László Babai<sup>1</sup>

University of Chicago, Chicago IL, USA

laci@cs.uchicago.edu

 <https://orcid.org/0000-0002-2058-685X>

Timothy J. F. Black<sup>2</sup>

University of Chicago, Chicago IL, USA

timblack@math.uchicago.edu

 <https://orcid.org/0000-0003-2469-9867>

Angela Wu<sup>3</sup>

University of Chicago, Chicago IL, USA

wu@math.uchicago.edu

---

## Abstract

---

The codewords of the *homomorphism code*  $\text{aHom}(G, H)$  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  $\text{mindist}$ , 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  $\text{poly}(1/\varepsilon)$  bound on the list size, i. e., on the number of codewords within distance  $(\text{mindist} - \varepsilon)$  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. We also obtain efficient local list-decoding for the permutation representations of alternating groups (the codomain is a symmetric group) under the restriction that the domain  $G = A_n$  is paired with codomain  $H = S_m$  satisfying  $m < 2^{n-1}/\sqrt{n}$ .

The limitations on the codomain in the latter case arise from severe technical difficulties stemming from the need to solve the *homomorphism extension* (HOMEXT) *problem* in certain cases; these are addressed in a separate paper (Wuu 2018).

We introduce an intermediate “semi-algorithmic” model we call **Certificate List-Decoding** that bypasses the HOMEXT bottleneck and works in the alternating vs. arbitrary setting. A certificate list-decoder produces partial homomorphisms that uniquely extend to the homomorphisms in the list. A homomorphism extender applied to a list of certificates yields the desired list.

**2012 ACM Subject Classification** Mathematics of computing → Coding theory, Mathematics of computing → Probabilistic algorithms

**Keywords and phrases** Error-correcting codes, Local algorithms, Local list-decoding, Finite groups, Homomorphism codes

**Digital Object Identifier** 10.4230/LIPIcs.APPROX-RANDOM.2018.29

**Related Version** A full version of this paper appears on arXiv [4], <https://arxiv.org/abs/1806.02969>.

---

<sup>1</sup> Partially supported by NSF Grants CCF 1423309 and CCF 1718902. The views expressed in the paper are those of the authors and do not necessarily reflect the views of the NSF.

<sup>2</sup> Partially supported by L. Babai’s cited NSF grants.

<sup>3</sup> Partially supported by L. Babai’s cited NSF grants.

## 1 Introduction

### 1.1 Brief history

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

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

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

We view  $\text{aHom}(G, H)$  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) [12] 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) [14] and Dinur, Grigorescu, Kopparty, and Sudan (2008) [10] and to the case of supersolvable domain and nilpotent codomain by Guo and Sudan (2014) [16], cf. [9].

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 paper), for an *arbitrary* received word  $f \in H^G$ , and any  $\varepsilon > 0$ , the number of codewords within radius (**mindist**  $- \varepsilon$ ) is bounded by  $\text{poly}(1/\varepsilon)$  (as opposed to some faster-growing function of  $\varepsilon$ , 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 (**mindist**  $- \varepsilon$ ) of the received word.

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

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

We call a CombEcon code **AlgEcon (algorithmically economically list-decodable)** if it permits efficient decoding within radius (**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.

## 1.2 Our contributions

### 1.2.1 Combinatorial bounds

In this paper 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  $\text{aHom}(G, H)$  is CombEcon. This paper is the first to demonstrate the existence of significant universally CombEcon classes.

► **Convention 1.** When speaking of a homomorphism code  $\text{aHom}(G, H)$ , the domain  $G$  will always be a finite group, but the codomain  $H$  will, in general, not be restricted to be finite.

► **Theorem 2** (Main combinatorial result). *Finite abelian and alternating groups are universally CombEcon.*

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

(Restatement of Theorem 2.) *Let the domain  $G$  be a finite abelian or alternating group and  $H$  an arbitrary (finite or infinite) group. Let  $\text{mindist}$  denote the minimum distance of the homomorphism code  $\text{aHom}(G, H)$  and let  $\varepsilon > 0$ . Let  $f \in H^G$  be an arbitrary received word. Then the number of codewords within  $(\text{mindist} - \varepsilon)$  of  $f$  is at most  $\text{poly}(1/\varepsilon)$ .*

► **Remark.** We give two proofs of this result. The first proof is nonconstructive and is based on a broadly applicable sphere packing argument (see Sec. 3.2). The second proof 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 20). This approach yields a very simple semi-algorithmic result (certificate list-decoding, see Sec. 1.2.3) and leads, using deeper tools [21], to our main algorithmic result, Theorem 6.

For abelian domains we prove a bound of  $O(\varepsilon^{-C-5})$  on the length of the list, i. e., the number of codewords within  $(\text{mindist} - \varepsilon)$  of the received word where  $O(\varepsilon^{-C})$  is the degree in the corresponding  $\{\text{abelian} \rightarrow \text{abelian}\}$  bound. (Currently  $C \approx 105$  [16].) For alternating domains we prove a bound of  $\tilde{O}(\varepsilon^{-7})$  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 3.** *The class of finite simple groups is universally CombEcon.*

The following problem is also open.

► **Problem 4.** *Is the class of all finite groups universally CombEcon?*

We suspect the answer to be negative.

Let us say that the **depth** of a subgroup  $M$  in a group  $G$  is the length  $\ell$  of the longest subgroup chain  $M = M_0 < M_1 < \dots < M_\ell = G$ . We say that a subgroup is **shallow** if it has bounded depth.

Theorem 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.** Our results list-decode certain classes of codes up to distance  $(\text{mindist} - \varepsilon)$  for positive  $\varepsilon$ . In many cases,  $\text{mindist}$  is the list-decoding boundary; examples show that the length of the list may blow up when  $\varepsilon$  is set to zero. Classes of such examples with abelian domain and codomain were found by Guo and Sudan [16]. We add classes of examples with alternating domains (see Appendix B).

### 1.2.2 Algorithms

On the algorithmic front, the combinatorial bound in the  $\{\text{abelian} \rightarrow \text{arbitrary}\}$  case permits us to adapt the algorithm of [14] 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  $\text{aHom}(G, H)$  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 5.** *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  $\text{aHom}(G, H)$  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 Appendix A.

A *permutation representation of degree  $m$*  of a group  $G$  is a homomorphism  $G \rightarrow S_m$ , 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 6** (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  $\text{aHom}(G, H)$  is AlgEcon, assuming  $m < 2^{n-1}/\sqrt{n}$ .*

The limitations on the codomain arise from the severe technical difficulties encountered.

In contrast to all previous work, in the alternating case the minimum distance does not necessarily correspond 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  $G \rightarrow H$  homomorphisms (see Sec. C). This necessitates the introduction of the *homomorphism extension (HOMEXT) problem*, a problem of interest in its own right, which remains the principal bottleneck for algorithmic progress.

By a  $G \rightarrow H$  *partial map* we mean a function  $\gamma: \text{dom}(\gamma) \rightarrow H$  where  $\text{dom}(\gamma) \subseteq G$ . The **homomorphism extension problem** HOMEXT asks whether a  $G \rightarrow H$  partial map extends to a  $G \rightarrow H$  homomorphism. The HOMEXT $_\lambda$  problem asks this only for maps  $\gamma$  whose domain generates a subgroup of density  $\mu(\langle \text{dom } \gamma \rangle) > \lambda$  in  $G$ .

The HOMEXT $_\lambda$  problem was solved by Wu [21] in the special case required for Theorem 6.

### 1.2.3 Certificate list-decoding

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

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

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

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  $\text{aHom}(G, H)$  is CertEcon.

► **Theorem 7** (Main semi-algorithmic result). *Alternating groups are universally CertEcon.*

In fact we show that SRG groups are universally CertEcon.

By the *density* of a partial map  $\gamma$  we mean the density of the subgroup  $\langle \text{dom}(\gamma) \rangle$  in  $G$ . A  $\lambda$ -certificate list-decoder produces partial maps of density  $\geq \lambda$ . The  $\text{HOMEXT}_\lambda$  problem asks to solve  $\text{HOMEXT}$  for partial maps of density  $\geq \lambda$ .

It is immediate that a  $\lambda$ -certificate list-decoder, combined with a  $\text{HOMEXT}_\lambda$  solver, suffices for list-decoding  $\text{aHom}(G, H)$ . This is the route we take to proving Theorem 6.

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

### 1.2.4 Mean-list-decoding and domain extension

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

The corollaries include a CombEcon result for {arbitrary  $\rightarrow$  abelian} homomorphism codes because of the known CombEcon result for {abelian  $\rightarrow$  abelian} homomorphism codes [10]. More generally we have a CombEcon result for {arbitrary  $\rightarrow$  nilpotent} homomorphism codes in view of the CombEcon result for {nilpotent  $\rightarrow$  nilpotent} homomorphism codes [16, 9]). 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 some detail in Appendix C. The mean-list-decoding technique was inspired by the concatenated code technique used in [16].

### 1.2.5 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  $\varphi \in \text{Hom}(G, H)$  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  $\text{Hom}(G, H)$  and  $\text{aHom}(G, H)$  are essentially equivalent tasks.

- **Proposition 8** (Hom versus aHom). *Let  $G$  be a finite group, and  $H$  a group.*
- (a) [15, Prop. 2.5] *If  $|\text{Hom}(G, H)| \geq 2$ , then  $\text{mindist}(\text{Hom}(G, H)) = \text{mindist}(\text{aHom}(G, H))$ .*
- (b) *For  $X \in \{\text{Comb}, \text{Cert}, \text{Alg}\}$ , if  $\text{Hom}(G, H)$  is  $X$  Econ then  $\text{aHom}(G, H)$  is  $X$  Econ. For  $X \in \{\text{Cert}, \text{Alg}\}$ , this statement requires that nearly uniform random elements of  $G$  be available.*

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

## 2 Notation and terminology

### 2.1 Group theoretic notation

Our general group theory reference is [19]. For the theory of permutation groups we refer to [11].

For finite sets  $B \subseteq A$  where  $A \neq \emptyset$ , we write  $\mu(B) = \mu_A(B) = |B|/|A|$  for the *density* of  $B$  in  $A$ . We use the notation  $[n] = \{1, \dots, n\}$ .

For  $G$  and  $M$  groups, we write  $M \leq G$  to indicate that  $M$  is a 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  $T$ .

For a set  $\Omega$ , the *symmetric group*  $\text{Sym}(\Omega)$  consists of all permutations of  $\Omega$ . We write  $S_n = \text{Sym}([n])$ . Permutation groups acting on  $\Omega$  are subgroups of  $\text{Sym}(\Omega)$ ; their *degree* is  $|\Omega|$ . The *alternating group*  $A_n \leq S_n$  consists of the even permutations. For  $G \leq \text{Sym}(\Omega)$  and  $\Delta \subseteq \Omega$ , the *pointwise stabilizer*  $G_{(\Delta)}$  consists of those  $\sigma \in G$  that fix  $\Delta$  pointwise. The setwise stabilizer  $G_\Delta$  is defined analogously.

### 2.2 Computational representation of groups

Our general reference to algorithmic group theory is [20].

Commonly used explicit representations include permutation groups, matrix groups, various representations of abelian groups such as the primary decomposition and the canonical form, etc. The latter are explained in Appendix A.1.

Black-box access is a general concept of oracle access to group operations. A black-box group is a finite group with (1) elements encoded by strings of uniform length, (2) black-box access, and (3) a given list of names of generators.

These concepts are explained in Appendix A.2.

## 3 Strategy

Let  $f \in \text{aHom}(G, H)$  be a received word and let  $\mathcal{L}$  be the list of codewords within distance  $(\text{mindist} - \varepsilon)$  of  $f$ . The combinatorial problem is to find a bound of the form  $|\mathcal{L}| \leq \text{poly}(1/\varepsilon)$ . First we use a sphere-packing argument to split  $\mathcal{L}$  into a moderate number of *buckets* (more manageable subsets).

### 3.1 Notation: agreement, equalizer

The *agreement*  $\text{agr}(f, g)$  of two functions  $f, g$  in the code space  $H^G$  is the proportion of inputs on which  $f$  and  $g$  agree, i. e.,

$$\text{agr}(f, g) = \frac{|\text{Eq}(f, g)|}{|G|}, \quad (2)$$

where  $\text{Eq}(f, g) = \{x \in G \mid f(x) = g(x)\}$  is the *equalizer* (agreements set) of  $f$  and  $g$ . So, the distance between  $f$  and  $g$  is  $1 - \text{agr}(f, g)$ . Following established notation, we write  $\Lambda_{G,H}$  to denote the maximum agreement between pairs of distinct elements of  $\text{aHom}(G, H)$ ,

$$\Lambda_{G,H} = \max\{\text{agr}(\varphi, \psi) \mid \varphi, \psi \in \text{Hom}(G, H), \varphi \neq \psi\}, \quad (3)$$

or  $\Lambda_{G,H} = 0$  if the only  $G \rightarrow H$  homomorphism is the trivial one. So, the minimum distance of the code  $\text{Hom}(G, H)$  is  $1 - \Lambda_{G,H}$ . By Proposition 8(a), the minimum distance of  $\text{aHom}(G, H)$  is also  $1 - \Lambda_{G,H}$ . When  $G$  and  $H$  are clear from context, we write  $\Lambda = \Lambda_{G,H}$ .

For a homomorphism code  $\mathcal{C}$  (either  $\text{Hom}(G, H)$  or  $\text{aHom}(G, H)$ ) and  $\lambda > 0$ , we write

$$\mathcal{L}(\mathcal{C}, f, \lambda) = \{\varphi \in \mathcal{C} \mid \text{agr}(f, \varphi) \geq \lambda\}. \quad (4)$$

So the list  $\mathcal{L}$  defined in the preamble of Section 3 is  $\mathcal{L} = \mathcal{L}(\text{aHom}(G, H), f, \Lambda + \epsilon)$ .

### 3.2 A sphere packing argument

We shall use a sphere packing argument to split the list into more manageable parts.

We begin with a strong negative correlation inequality.

► **Definition 9** (Strong negative correlation). Let  $\tau > 0$ . Let  $A_1, \dots, A_k$  be events in a probability space. We say that  $A_1, \dots, A_k$  are  **$\tau$ -strongly negatively correlated** if  $\Pr(A_i \cap A_j) \leq \Pr(A_i)\Pr(A_j) - \tau$  for all  $i \neq j$ .

► **Lemma 10** (Strong negative correlation bound). Let  $\tau > 0$ . Let  $A_1, \dots, A_k$  be  $\tau$ -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  $\text{Var}(Z_i) = \Pr(A_i)(1 - \Pr(A_i)) \leq \frac{1}{4}$ . For the covariances ( $i \neq j$ ) we have  $\text{Cov}(Z_i, Z_j) = \mathbb{E}[Z_i Z_j] - \mathbb{E}[Z_i]\mathbb{E}[Z_j] \leq -\tau$ . So,

$$0 \leq \text{Var}\left(\sum_i Z_i\right) = \sum_i \text{Var}(Z_i) + \sum_{i \neq j} \text{Cov}(Z_i, Z_j) \leq \frac{k}{4} - k(k-1)\tau. \quad (5)$$

Solving for  $k$  gives the bound as claimed. ◀

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

► **Lemma 11** (Sphere packing bound). Let  $G$  be a finite group,  $H$  a group, and  $\epsilon > 0$ . Let  $f: G \rightarrow H$  be a received word. Let  $\Psi \subseteq \mathcal{L} = \mathcal{L}(\text{aHom}(G, H), f, \Lambda + \epsilon)$  be a subset of the list that is maximal under the constraint that  $\text{agr}(\psi_1, \psi_2) \leq \Lambda^2$  for all distinct  $\psi_1, \psi_2 \in \Psi$ . Then

$$|\Psi| \leq \frac{1}{4(2\Lambda + \epsilon)\epsilon} + 1 \leq \frac{1}{4\epsilon^2} + 1. \quad (6)$$

**Proof.** Observe that the sets  $\text{Eq}(\psi, f)$  for  $\psi \in \Psi$  have density  $\geq \Lambda + \epsilon$  and they are  $\epsilon(2\Lambda + \epsilon)$ -strongly negatively correlated. Apply Lemma 10. ◀

► **Lemma 12.** Let  $\mathcal{L} = \mathcal{L}(\text{aHom}(G, H), \Lambda + \epsilon)$ . If  $|\mathcal{L}| \leq p(1/\Lambda, 1/\epsilon)$  for some monotone function  $p(\cdot, \cdot)$  then  $|\mathcal{L}| \leq p(2/\epsilon^2, 1/\epsilon) + 1/(2\epsilon^2)$ . In particular, in the definition of *CombEcon*, we may replace the bound  $\text{poly}(1/\epsilon)$  by  $\text{poly}(1/\Lambda, 1/\epsilon)$  without changing the meaning.

**Proof.** For  $\Lambda > \epsilon^2/2$ , we are done. For  $\Lambda \leq \epsilon^2/2$  we have  $|\mathcal{L}| \leq 1 + 1/(2\epsilon^2)$  by Lemma 10 because the sets  $\text{Eq}(f, \varphi)$  for  $\varphi \in \mathcal{L}$  are  $(\epsilon^2/2)$ -strongly negatively correlated. ◀

For  $\psi \in \Psi$ , we define the bucket  $\mathcal{L}_\psi$  by

$$\mathcal{L}_\psi = \{\varphi \in \mathcal{L} \mid \text{agr}(\varphi, \psi) > \Lambda^2\} \quad (7)$$

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

Our strategy to bound bucket size differs depending on the type of the domain  $G$ .

### 3.3 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  $\{\text{abelian} \rightarrow \text{abelian}\}$  homomorphism codes, which was done by Dinur, Grigorescu, Kopparty, and Sudan [10].

We work now with homomorphisms instead of affine homomorphisms; we will appeal to Proposition 8 to obtain affine results.

► **Theorem 13.** *Let  $G$  be a finite abelian group and  $H$  an arbitrary group. Let  $f \in H^G$  be a received word. Then there exists a set  $\mathcal{A}$  of finite abelian subgroups of the codomain  $H$  with  $|\mathcal{A}| \leq \frac{1}{4(2\Lambda + \varepsilon)\varepsilon^2} + \frac{1}{\varepsilon}$  such that for all  $\varphi \in \mathcal{L} = \mathcal{L}(\text{Hom}(G, H), f, \Lambda + \varepsilon)$ , there is  $M \in \mathcal{A}$  such that  $\varphi(G) \leq M$ .*

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  $\varphi(G) \leq M$ ,
- (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 13 follows from (ii) and the sphere packing bound (Lemma 11).

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

$$\text{enl}_B(T) = \langle T, C_H(T) \cap B \rangle. \quad (8)$$

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

Fix  $\psi \in \Psi$ . Let  $\mathcal{A}_\psi$  be the set of all subgroups  $M \leq H$  that occur as  $M = \text{enl}_{\psi(G)}(\varphi(G))$  for some  $\varphi$  in the bucket  $\mathcal{L}_\psi$ . We shall show that every  $M \in \mathcal{A}_\psi$  is equal to  $\text{enl}_{\psi(G)}(f(g))$  for at least an  $\varepsilon$  proportion of  $g \in G$ . The idea is that since  $\varphi$  and  $\psi$  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 14.** *Let  $\varphi, \psi \in \text{Hom}(G, H)$  and  $g \in G$  such that  $\langle g, \text{Eq}(\psi, \varphi) \rangle = G$ . Then  $\varphi(G) \leq \text{enl}_{\psi(G)}(\varphi(G)) = \text{enl}_{\psi(G)}(\varphi(g))$ .*

And, since  $f$  and  $\varphi$  have high agreement, it is likely that  $\varphi(g) = f(g)$ . We elaborate on this strategy in Section 4.

### 3.4 Bucket estimation for alternating groups

Subgroups of polynomial index in alternating groups are well understood; they are described by a result known as the Jordan–Liebeck Theorem (JLT), see [11, Theorem 5.2A].

► **Theorem 15** (Jordan–Liebeck). *Let  $n \geq 10$  and let  $r$  be an integer with  $1 \leq r < n/2$ . Suppose  $K \leq A_n$  has index  $|A_n : K| < \binom{n}{r}$ . Then for some  $\Delta \subseteq [n]$  with  $|\Delta| < r$  we have  $(A_n)_{(\Delta)} \leq K \leq (A_n)_\Delta$ .*

Here  $(A_n)_{(\Delta)}$  denotes the pointwise stabilizer of  $\Delta$  in  $A_n$  and  $(A_n)_\Delta$  the setwise stabilizer of  $\Delta$ .

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 that for  $n \geq 10$  we have  $\Lambda = 1/\binom{n}{s}$  for  $s \in \{1, 2\}$ . In the light of Lemma 12 it suffices to find a  $\text{poly}(n, 1/\varepsilon)$  bound on the size of each bucket.

#### 3.4.1 Nonconstructive proof

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

$$\mathcal{K} = \{(A_n)_{(\Delta)} \mid \Delta \subseteq [n], |\Delta| = 2s\} \tag{9}$$

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  $\mathcal{K}$ . It is not difficult to see that the depth of any  $K \in \mathcal{K}$  in  $A_n$  is 5 if  $s = 2$  (cf. [1]) and 2 if  $s = 1$ . (All we need is that this depth is bounded, which is obvious.)

All homomorphisms  $\varphi$  in the bucket  $\mathcal{L}_\psi$  have agreement  $> \Lambda^2$  with one representative homomorphism  $\psi$ ; so  $\varphi$  and  $\psi$  agree on a subgroup of index  $< 1/\Lambda^2 \leq \binom{n}{2s+1}$  (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  $\psi$  on  $K$ .

**Bound on the size of sub-buckets.** To bound the size of a sub-bucket  $\mathcal{L}_{\psi,K}$ , we describe a process for choosing a random homomorphism in the sub-bucket. For a positive integer  $d$ , we choose  $d$  random elements of  $G$ . If there is a unique homomorphism  $\varphi$  that agrees with  $f$  on the  $d$  random inputs, and agrees with  $\psi$  on  $K$ , we choose this homomorphism.

Now we combine the following two straightforward observations.

► **Observation 16.** *Let  $K \leq G$  be a subgroup,  $\psi \in \text{Hom}(G, H)$  a homomorphism,  $d$  a nonnegative integer, and  $g_1, \dots, g_d \in G$ . If  $\mu(\langle K, g_1, \dots, g_d \rangle) > \Lambda$ , then there is at most one homomorphism  $\varphi \in \text{Hom}(G, H)$  such that  $K \leq \text{Eq}(\psi, \varphi)$  and  $g_1, \dots, g_d \in \text{Eq}(f, \varphi)$ .*

► **Observation 17.** *Let  $0 \leq \lambda < 1$ . Let  $G$  be a finite group,  $K \leq G$  a subgroup, and  $S \subseteq G$  a subset. Suppose  $\mu(S) > \lambda$ . Let  $\varepsilon = \mu(S) - \lambda$  and  $d = \text{depth}_G(K)$ . Then,*

$$\Pr_{g_1, \dots, g_d \in G} [g_1, \dots, g_d \in S \text{ and } \mu(\langle K, g_1, \dots, g_d \rangle) > \lambda] \geq \varepsilon^d. \tag{10}$$

It follows that if  $d \geq \text{depth}_{A_n} K$  ( $K \in \mathcal{K}$ ) then each homomorphism in the sub-bucket  $\mathcal{L}_{\psi,K}$  gets chosen with probability at least  $\varepsilon^d$  and therefore  $|\mathcal{L}_{\psi,K}| \leq 1/\varepsilon^d$ . By the foregoing, we may choose  $d = 2s + 1$ , so  $|\mathcal{L}_{\psi,K}| \leq 1/\varepsilon^{2s+1}$ .

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 12.

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.

### 3.4.2 Constructive proof

Our second strategy for list decoding {alternating  $\rightarrow$  arbitrary} exploits a property of the alternating groups which we call *shallow random generation* (SRG).

A group  $G$  is SRG if, roughly, a small number of random elements of  $G$  are extremely likely to generate a low-depth subgroup.

► **Definition 18** (Shallow random generation). Let  $k, d \in \mathbb{N}$ . We say that a finite group  $G$  is  $(k, d)$ -shallow generating if

$$\Pr_{g_1, \dots, g_k \in G}[\text{depth}(\langle g_1, \dots, g_k \rangle) > d] < (\Lambda_G^*)^k, \quad (11)$$

where  $\Lambda_G^* = \min_H \{\Lambda_{G,H} \mid \Lambda_{G,H} \neq 0\}$ .

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.

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

► **Theorem 19.** *The class of alternating groups is SRG. Specifically, for sufficiently large  $n$ , the group  $A_n$  is  $(2, 5)$ -shallow generating.*

The proof uses the following result that 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 [6].

► **Theorem 20** (Babai). *Let  $\pi, \sigma$  be a pair of independent uniform random elements from  $S_n$ . For  $0 \leq t \leq n/3$ , let  $E(n, t)$  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  $r \geq n - t$ . Then,*

$$\Pr(E(n, t)) = 1 - \binom{n}{t+1}^{-1} + O\left(\binom{n}{t+2}^{-1}\right). \quad (12)$$

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

To prove Theorem 19 we use Theorem 20 with  $t = 4$ , noting that  $\Lambda_{A_n}^* = 1/\binom{n}{2}$  and  $\text{depth}_{A_n}(A_{n-4}) = 5$ .

From Observations 16 and 17 one can infer that SRG classes of groups are CombEcon. This view allows us not only to combinatorially list-decode {SRG  $\rightarrow$  arbitrary}, but also to certificate list-decode; certificates are given by  $f$  restricted to a small number of random elements of  $G$ .

► **Definition 21.** A domain certificate  $S \subseteq G$  is a *domain- $\Lambda$ -certificate* if  $\mu(\langle S \rangle) > \Lambda$ .

► **Theorem 22** (SRG implies CertEcon, via domain certificates). *Let  $k \in \mathbb{N}$  and  $c > 0$ . Let  $G$  be a  $(k, d)$ -shallow generating group and  $H$  a group. Let  $f: G \rightarrow H$  and  $\varepsilon > 0$ . Let  $\Upsilon$  be a list of  $\lceil \frac{1}{\varepsilon^{k+d}} \ln(\frac{4}{\varepsilon^{k+d}}) \rceil$  independently chosen subsets of  $G$ , each of size  $k + d$ . Then, with probability at least  $3/4$ ,  $\Upsilon$  is a domain- $\Lambda$ -certificate-list of  $\mathcal{L}(\text{aHom}(G, H), f, \Lambda + \varepsilon)$ .*

► **Remark.** From Theorem 19 with  $k = 2, d = 5$  we infer that for  $\{A_n \rightarrow \text{arbitrary}\}$ , the length of the list is  $\tilde{O}(\varepsilon^{-7})$ .

We say that a class  $\mathfrak{G}$  is  $\lambda$ -*CertEcon* if it is *CertEcon* with the additional requirement that for all certificates  $\gamma$  in our certificate list, we must have  $\mu(\langle \text{dom } \gamma \rangle) > \lambda$ .

► **Theorem 23.** *If  $\mathfrak{G}$  is an SRG class of groups, then  $\mathfrak{G}$  is universally  $\Lambda$ -CertEcon.*

Here we assume that the domain  $G$  is given as a black-box group. In order to obtain domain-certificates, no access to the codomain  $H$  is needed. To obtain actual certificates ( $G \rightarrow H$  partial functions), we only need black-box access to  $H$ .

### 3.5 Cert + HomExt = Alg

A  $\Lambda$ -certificate list-decoder and a  $\text{HOMEXT}_\Lambda$  solver combine to an algorithmic list decoder. Wuu [21] solved the homomorphism extension problem in the following case.

► **Theorem 24** (Wuu). *Assume  $m < 2^{n-1}/\sqrt{n}$  and  $\lambda = 1/\text{poly}(n)$ . Then the  $\text{HOMEXT}_\lambda(A_n, S_m)$  search problem can be solved in  $\text{poly}(n, m)$  time.*

Combining the previous two theorems, we obtain our main algorithmic result.

► **Theorem 25** (Main algorithmic result).  *$\text{aHom}(A_n, S_m)$  is AlgEcon, assuming  $m < 2^{n-1}/\sqrt{n}$ .*

## 4 Homomorphism Codes with finite abelian domain and arbitrary codomain

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 13, 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.

In Section 4.1, we state characterizations of  $\Lambda_{G,H}$  when  $G$  is abelian. In Section 4.2 we state facts about abelian enlargements (see definition in Section 3.3). Using this tool, in Section 4.3 we prove Theorem 13 (the key result mentioned in the previous paragraph) and infer that abelian groups are universally CombEcon. In Section 4.4 we adapt the algorithm of [10, 16], to give an algorithm to locally list-decode these codes.

We remark that these codes usually cannot be list-decoded beyond radius  $1 - (\Lambda_{G,H} + \varepsilon)$  (see Remark at the end of Section 1.2.1).

### 4.1 $\Lambda_{G,H}$ when $G$ is abelian

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

► **Fact 26.** Let  $G$  be a finite abelian group and  $H$  a group. 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  $|\text{Hom}(G, H)| = 1$  and  $\Lambda_{G,H} = 0$ .

Guo [15, Theorem 1.1] gave a characterization of  $\Lambda_{G,H}$  when  $G$  and  $H$  are finite groups with  $G$  solvable or  $H$  nilpotent.

## 4.2 Abelian enlargements

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 3.3.

► **Lemma 27.** *Let  $B \leq H$  a finite abelian subgroup, and  $T \subseteq H$  a subset such that  $\langle T \rangle$  is a finite abelian group. For  $U \subseteq \text{enl}_B(T)$ , we have that  $\text{enl}_B(T) = \text{enl}_B(T \cup U)$ .*

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

$$C_H(T) \cap B \leq \text{enl}_B(T) \leq C_H(\text{enl}_B(T)) \leq C_H(U). \quad (13)$$

So,

$$C_H(T) \cap B \leq C_H(T) \cap C_H(U) \cap B = C_H(T \cup U) \cap B \leq \text{enl}_B(T \cup U). \quad (14)$$

Since also  $T \subseteq \text{enl}_B(T \cup U)$ , we have that  $\text{enl}_B(T) \leq \text{enl}_B(T \cup U)$ .

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

► **Proposition 28.** *Let  $\varphi, \psi \in \text{Hom}(G, H)$  and  $A \subseteq G$  such that  $\langle A, \text{Eq}(\psi, \varphi), \ker \varphi \rangle = G$ . Then  $\text{enl}_{\psi(G)}(\varphi(A)) = \text{enl}_{\psi(G)}(\varphi(G))$ .*

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

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

Therefore, by Lemma 27,

$$\text{enl}_{\psi(G)}(\varphi(A)) = \text{enl}_B(T) = \text{enl}_B(T \cup U) = \text{enl}_B(\langle T \cup U \rangle) = \text{enl}_{\psi(G)}(\varphi(G)). \quad (15)$$

► **Corollary 29.** *Let  $\varphi, \psi$ , and  $A$  be as above. Then  $\varphi(G) \leq \text{enl}_{\psi(G)}(\varphi(A))$ .*

Proposition 14 is a special case of Proposition 28.

## 4.3 Combinatorial list-decodability, finite abelian to arbitrary

In this section, we establish that finite abelian groups are universally CombEcon.

Throughout this section, let  $G$  be a finite abelian group, and  $H$  an arbitrary group (finite or infinite). Let  $f: G \rightarrow H$  be a received word. Let  $\varepsilon > 0$ . Let  $\mathcal{L} = \mathcal{L}(\text{Hom}(G, H), f, \Lambda + \varepsilon)$  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 1.2.5). The list  $\mathcal{L}$  is divided into buckets  $\mathcal{L}_\psi$  for  $\psi \in \Psi$ , where  $\Psi$  is as in Lemma 11.

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 [10] proved that  $\text{aHom}(G, H)$  is CombEcon (and in fact, AlgEcon) for all finite abelian groups  $G$  and  $H$ . Theorem 13, combined with the DGKS result, lets us conclude that  $\text{Hom}(G, H)$  (and thus  $\text{aHom}(G, H)$ ) is CombEcon.

► **Corollary 30.** *Finite abelian groups are universally CombEcon. Specifically, let  $C$  be a constant such that  $|\mathcal{L}(\text{aHom}(G, H), f, \Lambda + \varepsilon)| \leq (\frac{1}{\varepsilon})^C$  for  $G, H$  finite abelian groups. Then  $|\mathcal{L}(\text{aHom}(G, H), f, \Lambda + \varepsilon)| \leq O((\frac{1}{\varepsilon})^{C+4})$  for  $G$  a finite abelian group and  $H$  an arbitrary group.*

By [16, 9], the constant  $C$  currently stands at  $\approx 105$ .

**Proof of Corollary 30.** Let  $\mathcal{A}$  be the collection of subgroups of  $H$  guaranteed by Theorem 13. Then,  $\mathcal{L} \subseteq \bigcup_{M \in \mathcal{A}} \mathcal{L}(\text{Hom}(G, M), f, \Lambda + \varepsilon)$  (on the right hand side we let  $f$  be redefined arbitrarily at points in its domain that do not map to  $M$ ). So,

$$|\mathcal{L}| \leq \sum_{M \in \mathcal{A}} \ell(\text{Hom}(G, M), \Lambda + \varepsilon) \leq \left( \frac{1}{4(2\Lambda + \varepsilon)\varepsilon^2} + \frac{1}{\varepsilon} \right) \left( \frac{1}{\varepsilon} \right)^C. \quad (16)$$

We then apply the Remark after Proposition 8. ◀

In the remainder of this subsection, we prove Theorem 13.

Let  $\Psi$  be as in Lemma 11. Recall our strategy from Section 3.2 of dividing the list  $\mathcal{L}$  into buckets  $\mathcal{L}_\psi$  for  $\psi \in \Psi$ . We will prove the following.

► **Lemma 31.** *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 13 follows by taking  $\mathcal{A} = \bigcup_{\psi \in \Psi} \mathcal{A}_\psi$ .

**Proof of Lemma 31.** Let  $\mathcal{A}_\psi = \{\text{enl}_{\psi(G)}(\varphi(G)) \mid \varphi \in \mathcal{L}_\psi\}$ . Then  $\mathcal{A}_\psi$  is a set of finite abelian subgroups of  $H$ . And, for all  $\varphi \in \mathcal{L}_\psi$ , we have that  $\varphi(G) \leq \text{enl}_{\psi(G)}(\varphi(G)) \in \mathcal{A}_\psi$ .

Let  $a$  be a uniform random element of  $G$ . For each  $M \in \mathcal{A}_\psi$ , let  $E_M$  be the event that  $\text{enl}_{\psi(G)}(f(a)) = M$ . 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  $\text{enl}_{\psi(G)}(\varphi(G)) = M$ . Let  $N$  be the  $(G, H)$ -irrelevant kernel. Since  $N \leq \text{Eq}(\varphi, \psi)$ , we have that  $|G : \text{Eq}(\psi, \varphi)|$  divides  $|G : N|$ , whose smallest prime factor is  $\frac{1}{\Lambda}$  by Fact 26.

If  $a \in \text{Eq}(f, \varphi) \setminus \text{Eq}(\psi, \varphi)$ , then  $\mu(\langle a, \text{Eq}(\psi, \varphi) \rangle) \geq \frac{1}{\Lambda} \mu(\text{Eq}(\psi, \varphi)) > \frac{1}{\Lambda} \Lambda^2 = \Lambda$ , so  $\langle a, \text{Eq}(\psi, \varphi) \rangle = G$ . In this case, by Proposition 14,

$$\text{enl}_{\psi(G)}(f(a)) = \text{enl}_{\psi(G)}(\varphi(a)) = \text{enl}_{\psi(G)}(\varphi(G)) = M. \quad (17)$$

Therefore,

$$\Pr[E_M] \geq \Pr[a \in \text{Eq}(f, \varphi) \setminus \text{Eq}(\psi, \varphi)] \geq \text{agr}(f, \varphi) - \text{agr}(\psi, \varphi) \geq \varepsilon. \quad (18)$$

We conclude that  $|\mathcal{A}_\psi| \leq \frac{1}{\varepsilon}$ . ◀

## 4.4 Algorithm

► **Definition 32.** A **primary decomposition** of a finite abelian group  $G$  is a representation as a direct product of cyclic groups of prime-power order.

For  $G$  a finite abelian group and  $H$  an arbitrary group, we can locally list-decode  $\text{aHom}(G, H)$ . Based on our CombEcon bound for this class of pairs of groups, we adapt the algorithm of Dinur, Grigorescu, Kopparty, and Sudan from [10, Sec. 5]. Thus, such codes are AlgEcon. Like [10], we assume that  $G$  is given explicitly by an primary decomposition.

► **Theorem 33.** *Let  $\mathcal{D}$  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  $\mathcal{D}$  in time  $\text{poly}(\log|G| \cdot \frac{1}{\epsilon})$ .*

Here we assume the unit-cost model of naming elements of  $H$  (cf. Def. 34).

Details of our adaptation of the algorithm of Dinur et al. [10] are given in Appendix D.

---

## 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. 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.
- 3 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.
- 4 László Babai, Timothy J. F. Black, and Angela Wu. List-decoding homomorphism codes with arbitrary codomains. *arXiv*, 2018. (full version of this paper). arXiv:1806.02969.
- 5 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.
- 6 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.
- 7 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.
- 8 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.
- 9 Timothy Black, Alan Guo, Madhu Sudan, and Angela Wu. List decoding nilpotent groups. In preparation, 2018.
- 10 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.
- 11 John D. Dixon and Brian Mortimer. *Permutation Groups*. Graduate Texts in Math. Springer New York, 1996.
- 12 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.
- 13 Parikshit Gopalan, Adam R. Klivans, and David Zuckerman. List-decoding Reed-Muller codes over small fields. In *40th STOC*, pages 265–274. ACM, 2008. doi:10.1145/1374376.1374417.
- 14 Elena Grigorescu, Swastik Kopparty, and Madhu Sudan. Local decoding and testing for homomorphisms. In *APPROX-RANDOM*, volume 4110 of *Lecture Notes in Computer Science*, pages 375–385. Springer, 2006. doi:10.1007/11830924\_35.
- 15 Alan Guo. Group homomorphisms as error correcting codes. *Electronic J. Combinatorics*, 22(1):P1.4, 2015.
- 16 Alan Guo and Madhu Sudan. List decoding group homomorphisms between supersolvable groups. In *APPROX/RANDOM*, volume 28 of *LIPICs*, pages 737–747. Schloss Dagstuhl - Leibniz-Zentrum fuer Informatik, 2014. doi:10.4230/LIPICs.APPROX-RANDOM.2014.737.
- 17 William Cary Huffman and Vera Pless. *Fundamentals of Error-Correcting Codes*. Cambridge Univ. Press, 2003.

- 18 I. Martin Isaacs. *Finite Group Theory*. Graduate studies in mathematics. Amer. Math. Soc., 2008.
- 19 Derek J. S. Robinson. *A Course in the Theory of Groups*. Springer, 2nd edition, 1995.
- 20 Ákos Seress. *Permutation Group Algorithms*. Cambridge Tracts in Math. Cambridge Univ. Press, 2003.
- 21 Angela Wuu. Homomorphism extension. *arXiv*, 2018. [arXiv:1802.08656](https://arxiv.org/abs/1802.08656).

## A Computational representation of groups

### A.1 Finite 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  $n_1, \dots, n_k$  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 representation alone will not suffice for the algorithms in prior work; we need 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*.

### A.2 Black-box access, black-box groups

Our most general model of access to a group is *black-box access*. In this model, a “universe”  $U$  is a collection of potential names of group elements. Not all elements of the universe encode group elements and a group element may have multiple names.

► **Definition 34.** We say that we have **black-box access** to a group  $G$  if the following holds. There is a set  $U$  of “names” and a surjection  $r: U \rightarrow G \cup *$  where  $*$  is a special symbol. For  $u \in r^{-1}(U)$  we say that  $u$  is a name of  $r(u) \in G$ . Given the names of two group elements, an oracle gives a name of their product/quotient, and recognizes the names of the identity element. We assume a tape that can hold an element of  $U$  in each cell, accessible at unit cost (“unit-cost model”).

Black-box access does not assume we know the name of any element of  $G$ . Nevertheless, such access may be sufficient in the case of the codomain; in this case we assume the received word consists of names of elements of the codomain. “Black-box groups” (introduced in [5]) require in addition that a list of generators be known, and that the names be words of uniform length over a fixed finite alphabet.

► **Definition 35.** We say that a *finite group  $G$*  is given as a **black-box group** if the following hold.

- (a) The universe  $U$  is  $\Sigma^n$  where  $\Sigma$  is a fixed finite alphabet. We call  $n$  the *encoding length* of the group elements.
- (b)  $G$  is given black-box access over the universe  $U$ .
- (c) The names of a list of generators of  $G$  are given.

It follows in particular that  $|G| \leq |\Sigma|^n$ .

An important consequence of representation as a black-box group is the availability of nearly uniform random elements. We say that we sample elements of a finite set  $\Omega$   $\epsilon$ -nearly uniformly if the probability of each element to be selected is between  $(1 \pm \epsilon)/|\Omega|$ .

► **Theorem 36** ([2]). *Given a finite group  $G$  as a black-box group, one can generate names of independent, nearly uniformly distributed elements of  $G$ ; the per element cost is polynomial in the encoding length of the elements and  $|\log(\epsilon)|$  where  $\epsilon$  is the near-uniformity parameter.*

In our applications, setting  $\epsilon = 1/2$  will suffice.

## B Upper bound on the list-decoding radius

We have shown that the class of homomorphism codes with alternating domain and arbitrary codomain have list-decoding radius greater than  $1 - (\Lambda + \epsilon)$  for all  $\epsilon > 0$ .

On the other hand, we now show that in many cases, a blowup of the list size occurs at radius  $1 - \Lambda$ , demonstrating that the list-decoding radius is at most  $1 - \Lambda$  in these cases.

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 37.** *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*

$$\ell(\text{Hom}(A_n, H_n), \Lambda) = 2^{\Omega(n)} \geq 2^{\Omega(\sqrt[3]{\log|H|})}. \quad (19)$$

Moreover, for any fixed  $n \geq 10$ , and any integer  $M$ , there is a finite group  $H$  such that

$$\ell(\text{Hom}(A_n, H), \Lambda) \geq M. \quad (20)$$

**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  $f: A_n \rightarrow H_n$  by  $f(g) = (g, \dots, g)$ , the diagonal identity map, where  $A_n$  is embedded in  $A_{n+1}$ . For nonempty  $S \subseteq [n]$  and  $j \in [n]$ , let  $h = h(S, j) = (h_1, \dots, h_k) \in H_n$ , where  $h_i$  is the transposition  $(j, n+1)$  if  $i \in S$  and 1 otherwise. For each such  $h$ , let  $\varphi_h \in \text{Hom}(A_n, H_n)$  be given by  $\varphi_h(g) = h^{-1}f(g)h$ . Each  $\varphi_h$  has agreement  $\text{agr}(\varphi_h, f) = 1/n = \Lambda$  with  $f$ . There are  $n(2^k - 1)$  such  $h$ , so  $\ell(\text{Hom}(A_n, H_n), \Lambda) \geq n(2^k - 1)$ .

Suppose  $\lambda = 1/\binom{n}{2}$ . Let  $H_n = A_n^k$ . Then,  $\Lambda_{A_n, H_n} = 1/\binom{n}{2}$ . Let  $f: A_n \rightarrow H_n$  by  $f(g) = (g, \dots, g)$ , the diagonal identity map. For nonempty  $S \subseteq [n]$  and  $\tau \in S_n$  is a transposition, let  $h = h_{S, \tau} = (h_1, \dots, h_k) \in A_n^k$ , where  $h_i = \tau$  if  $i \in S$  and 1 otherwise. For each such  $h$ , let  $\varphi_h \in \text{Hom}(A_n, H_n)$  be given by  $\varphi_h(g) = h^{-1}f(g)h$ . Each such  $\varphi_h$  has agreement  $\text{agr}(\varphi_h, f) = 1/\binom{n}{2}$ . There are  $\binom{n}{2}(2^k - 1)$  such  $h$ , so  $\ell(\text{Hom}(A_n, H_n), \Lambda) \geq \binom{n}{2}(2^k - 1)$ . ◀

We remark that  $\ell(\text{Hom}(A_n, H), \Lambda_{A_n, H})$  is not bounded as a function of  $n$  for a wide variety of classes of  $H$ .

## C Mean-list-decoding, irrelevant kernel, and domain relaxation

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). An important implication is that we can extend our list-decoding results to a wider class of domain groups. Specifically, define the  $(G, H)$ -irrelevant kernel to be the intersection of the kernels of all  $G \rightarrow H$  homomorphisms. We show (Theorem 40) that for  $\text{aHom}(G, H)$  to be economically list-decodable (CombEcon, CertEcon, or AlgEcon), it suffices to show that  $\text{aHom}(G/N, H)$  is. As an example, the {abelian  $\rightarrow$  abelian} list-decoding results automatically extend to {arbitrary  $\rightarrow$  abelian} results.

For a family  $\mathcal{F} = \{f_i \mid i \in I\}$  of received words and a word  $f \in H^G$ , we define the **average agreement**  $\text{agr}(w, \mathcal{F})$  of  $w$  and  $\mathcal{F}$  by

$$\text{agr}(w, \mathcal{F}) = \frac{1}{|\mathcal{F}|} \sum_{i \in I} \text{agr}(w, f_i). \quad (21)$$

For a homomorphism code  $\mathcal{C}$  (either  $\text{Hom}(G, H)$  or  $\text{aHom}(G, H)$ ), the **mean-list**  $\mathcal{L}$  is the set of codewords whose agreement with  $\mathcal{F}$  is at least a specified quantity  $\rho$ ; i. e.,

$$\mathcal{L} = \mathcal{L}(\mathcal{C}, \mathcal{F}, \rho) = \{w \in \mathcal{C} \mid \text{agr}(w, \mathcal{F}) \geq \rho\}. \quad (22)$$

We define **CombEconM**, **CertEconM**, and **AlgEconM** by replacing the received word  $f$  with a family of received words  $\mathcal{F}$  and replacing “the list” with “the mean-list” in the definitions of CombEcon, CertEcon, and AlgEcon, respectively. However, the -M concepts turn out to be equivalent to the non-M, concepts.

► **Theorem 38.** *For a class  $\mathcal{C}$  of homomorphism codes,  $\mathcal{C}$  is CombEconM if and only if it is CombEcon.*

Under suitable access assumptions, analogous results hold for CertEcon and AlgEcon.

Theorem 38 follows from the following observation.

► **Lemma 39.** *For all homomorphism codes  $\mathcal{C}$  (either  $\text{Hom}(G, H)$  or  $\text{aHom}(G, H)$ ), for all families  $\mathcal{F}$  of received words, for all  $\rho, \delta > 0$ ,*

$$|\mathcal{L}(\mathcal{C}, \mathcal{F}, \rho)| \leq \frac{1}{\delta} \max_{f \in \mathcal{F}} |\mathcal{L}(\mathcal{C}, f, \rho + \delta)|. \quad (23)$$

For groups  $G$  and  $H$ , and  $N \trianglelefteq G$  a subgroup of the  $(G, H)$ -irrelevant kernel, we note that any homomorphism (or affine homomorphism)  $G \rightarrow H$  is the composition of a homomorphism (or affine homomorphism)  $G/N \rightarrow H$  with the projection map  $G \rightarrow G/N$ . Thus,  $\Lambda_{G,H} = \Lambda_{G/N,H}$ .

► **Theorem 40.** *Let  $G, H$  be groups and  $N \trianglelefteq G$  a subgroup of the  $(G, H)$ -irrelevant kernel. If  $\text{aHom}(G/N, H)$  is CombEcon, then  $\text{aHom}(G, H)$  is CombEcon.*

► **Remark.** Under suitable access assumptions, analogous results hold for CertEcon and AlgEcon. These assumptions involve (a) uniform random generation of elements of  $N$ , and (b) oracle access to a transversal, i. e., an injection  $G/N \rightarrow G$  that assigns a representative element to each coset.

**Proof of Theorem 40.** Let  $\Lambda = \Lambda_{G,H} = \Lambda_{G/N,H}$ .

We prove the theorem in the combinatorial setting. Fix a set  $S$  of coset representatives of  $N$  in  $G$ . To the function  $f: G \rightarrow H$  we associate the family  $\mathcal{F} = \{f_n: G/N \rightarrow H \mid n \in N\}$  where  $f_n(sN) = f(sn)$  for all  $n \in N$  and  $s \in S$ . Then,  $\text{agr}(\varphi \circ \pi, f) = \text{agr}(\varphi, \mathcal{F})$  for all  $\varphi \in \text{aHom}(G/N, H)$ , where  $\pi: G \rightarrow G/N$  is the projection map. If we identify  $\varphi \in \text{aHom}(G/N, H)$  with  $\varphi \circ \pi \in \text{aHom}(G, H)$ , then, for any  $\epsilon > 0$ , we have

$$\mathcal{L}(\text{aHom}(G, H), f, \Lambda + \epsilon) = \mathcal{L}(\text{aHom}(G/N, H), \mathcal{F}, \Lambda + \epsilon). \quad (24)$$

Now Theorem 40 follows by an application of Theorem 38. ◀

**D** Adaptation of the DGKS algorithm to abelian  $\rightarrow$  arbitrary

In Section 4.4 we mentioned that to prove the {abelian $\rightarrow$ arbitrary} AlgEcon result, we combine our CombEcon result for this class of codes with an adaptation of the {abelian  $\rightarrow$  arbitrary} algorithm from [10]. Here we indicate how our version differs from that algorithm.

First, [10] 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  $G = G_1, \dots, G_k$ , 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  $\mathbb{Z}_{p_i^{r_i}}$  should be replaced with  $\mathbb{Z}_{p_i^{r_i}}$ .

In the algorithm EXTEND of [10], 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  $\varphi(e_1), \dots, \varphi(e_{i-1})$ .” 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.