Is Constraint Satisfaction Over Two Variables Always Easy?*

Lars Engebretsen[1,**] and Venkatesan Guruswami[2,***]

[1] Department of Numerical Analysis and Computer Science
Royal Institute of Technology
SE-100 44 Stockholm
SWEDEN

[2] Department of Computer Science and Engineering
University of Washington
Seattle, WA 98195
USA

January 2004

Abstract. By the breakthrough work of Håstad (J. ACM 48(4):798–859), several constraint satisfaction problems are now known to have the following *approximation resistance* property: satisfying more clauses than what picking a random assignment would achieve is **NP**-hard. This is the case for example for Max E3-Sat, Max E3-Lin and Max E4-Set Splitting. A notable exception to this extreme hardness is constraint satisfaction over two variables (2-CSP); as a corollary of the celebrated Goemans-Williamson algorithm (J. ACM 42(6):1115–1145), we know that every Boolean 2-CSP has a non-trivial approximation algorithm whose performance ratio is better than that obtained by picking a random assignment to the variables. An intriguing question then is whether this is also the case for 2-CSPs over larger, non-Boolean domains. This question is still open, and is equivalent to whether the generalization of Max 2-SAT to domains of size d, can be approximated to a factor better than $(1 - 1/d^2)$.

In an attempt to make progress towards this question, in this paper we prove, firstly, that a slight restriction of this problem, namely a generalization of linear inequations with two variables per constraint, *is not* approximation resistant, and, secondly, that the Not-All-Equal Sat problem over domain size d with three variables per constraint, *is* approximation resistant, for every $d \geq 3$. In the Boolean case, Not-All-Equal Sat with three variables per constraint is equivalent to Max 2-SAT and thus has a non-trivial approximation algorithm; for larger domain sizes, Max 2-SAT can be reduced to Not-All-Equal Sat with three variables per constraint. Our approximation algorithm implies that a wide class of 2-CSPs called *regular 2-CSPs* can all be approximated beyond their random assignment threshold.

*A preliminary version of this work appears as an extended abstract in the *Proceedings of the 6th International Workshop on Randomization and Approximation Techniques in Computer Science* (RANDOM'02), 13–15 September 2002, Cambridge, Massachusetts.

**Research partly performed while the author was visiting MIT with support from the Marcus Wallenberg Foundation and the Royal Swedish Academy of Sciences.

***Research performed in part when the author was at University of California, Berkeley, supported by a Miller Research Fellowship.

1 Introduction

In a breakthrough paper, Håstad [11] studied the problem of giving approximate solutions to maximization versions of several constraint satisfaction problems. An instance of a such a problem is given as a set of variables and a collection of constraints, i.e., functions from some domain to $\{0, 1\}$, on certain subsets of variables, and the objective is to find an assignment to the variables that satisfies as many constraints as possible. An approximate solution of a constraint satisfaction program is simply an assignment that satisfies roughly as many constraints as possible. In this setting we are interested in proving either that there exists a polynomial time algorithm producing approximate solutions, i.e., solutions that are at most some constant factor worse compared to the optimum, or that no such algorithms exist.

In this paper we will study the common special case where each individual constraint depends on a fixed number k of the variables—this case is usually called the Max k-CSP problem and the size of the instance is given as the total number of variables that appear in the constraints. The complexity of the constraint satisfaction problem (CSP) is determined by the precise set of constraints that may be posed on subsets of k variables, and accordingly we get various families of Max k-CSP problems. For each such CSP, there exists a very naive algorithm that approximates the optimum within a constant factor: The algorithm that just guesses a solution at random. In his paper, Håstad [11] proved the very surprising fact that this algorithm is essentially the best possible efficient algorithm for several constraint satisfaction problems, unless $\mathbf{P} = \mathbf{NP}$. The proofs unify constructions from interactive proof systems with harmonic analysis over finite groups and give a general framework for proving strong impossibility results regarding the approximation of constraint satisfaction programs.

Håstad [11] suggests that predicates with the property that the naive randomized algorithm is the best possible polynomial time approximation algorithm should be called *non-approximable beyond the random assignment threshold*; we also use the phrase *approximation resistant* to refer to the same phenomenon.

Definition 1. *A solution to a maximization problem is* α-approximate *if it is feasible and has weight at least α times the optimum. An approximation algorithm has* performance ratio α *if it delivers α-approximate solutions in polynomial time.*

Definition 2. *A CSP is said to be* approximation resistant *or* non-approximable beyond the random assignment threshold *if, for every constant $\varepsilon > 0$, it is* $\mathbf{NP}$*-hard to compute a $(\rho + \varepsilon)$-approximate solution, where ρ is the expected fraction of constraints satisfied by a solution guessed uniformly at random.*

Clearly, understanding which predicates are approximation resistant is an important pursuit. The current knowledge is that for Boolean CSPs, which understandably have received the most attention so far, there is a precise understanding of which CSPs on *exactly three* variables are approximation resistant: All predicates that are implied by parity have this property [11, 19]. It is known that *no* Boolean CSP over two variables is approximation resistant; this is a corollary of the breakthrough Goemans-Williamson algorithm [9]. For the case of four or more variables, very little is known; therefore it seems to be a good approach to first understand the situation for two and three variables.

Accordingly, we are interested in the situation for CSPs with two and three variables over larger, non-Boolean, domains. In particular, it is a really intriguing question whether every CSP over two variables can be approximated better than random, no matter what the domain size is. The central aim of this paper is to study this question. We are not able to resolve it completely, but we conjecture that the answer to the question is yes.

1.1 Formal definitions of some CSPs

Before discussing our results, we will need to define some of the CSPs that we will be concerned with in this paper. A specific k-CSP problem is defined by the family of constraints that may be imposed on subsets of k variables. Allowing arbitrary constraints gives the most general problem, which we call Max Ek-CSP(d). In this paper, d refers to the domain size from which the variables may take values, with $d = 2$ corresponding to the Boolean case. Over domain size d, a constraint is simply a function $f\colon [d]^k \to \{0, 1\}$, where $[d] = \{0, 1, \ldots, d-1\}$. Equivalently, a constraint f can be viewed as a subset of $[d]^k$ consisting of all inputs which it maps to 1.

The Max Ek-Sat(d) problem is defined by the constraint family $\{f \subseteq [d]^k : |f| = d^k - 1\}$, i.e., the family of all constraints having just one non-satisfying assignment. Max Ek-NAE-Sat(d) is the problem where the constraints assert that the specific variables are not all equal, except that we also allow translates of variables, e.g., for the two variable case, a constraint can assert $x_1+1 \neq x_2+3$ (the addition being done modulo d); this is the analog of complementation of Boolean variables. In the Max Ek-Lin(d) problem, the constraint family is given by all linear constraints: $\{\mathrm{Lin}(\alpha_1, \ldots, \alpha_k, c) : \alpha_i, c \in [d]\}$ where $\mathrm{Lin}(\alpha_1, \ldots, \alpha_k, c) = \{(x_1, \ldots, x_k) : \sum_i \alpha_i x_i = c \bmod d\}$. The Max E$k$-LinInEq($d$) problem is defined by the family of all linear inequations: $\{f \subseteq [d]^k : [d]^k \setminus f \text{ is a linear constraint}\}$.

For the two variable case, we define the constraint satisfaction problems Max BIJ(d) and Max Co-BIJ(d) which are generalizations of Max E2-Lin(d) and Max E2-LinInEq(d) respectively. Let S_d be the set of all bijections from $[d]$ to $[d]$. For each $\pi \in S_d$, define the 2-ary constraint $f_{\pi,d} = \{(a, b) \in [d]^2 : b = \pi(a)\}$. Now define the family $\mathrm{BIJ}(d) = \{f_{\pi,d} : \pi \in S_d\}$; we call the CSP associated with this family Max BIJ(d). The problem Max Co-BIJ(d) is obtained by constraints which are complements of those in BIJ(d), i.e., a constraint is of the form $\pi(x_1) \neq x_2$ for some bijection π defined over $[d]$. It is clear that these problems generalize Max E2-Lin(d) and Max E2-LinInEq(d) respectively.

For the three variable case, we define the problem Max E3-NAE-Sat(G) for finite Abelian groups G. For each triple $(g_1, g_2, g_3) \in G^3$ define the constraint $N_{g_1,g_2,g_3} = \{(x_1, x_2, x_3) \in G^3 : \neg(g_1x_1 = g_2x_2 = g_3x_3)\}$. Now define the family $\mathrm{NAE}(G) = \{N_{g_1,g_2,g_3} : (g_1, g_2, g_3) \in G^3\}$; we denote by Max E3-NAE-Sat(G) the CSP associated with this family of constraints. Note that the group structure is indeed present in the problem since the constraints involve multiplication by elements from G. In fact, we are able to prove in this paper that Max E3-NAE-Sat($\mathbb{Z}_4$) is approximation resistant while we are unable to determine the approximability of Max E3-NAE-Sat($\mathbb{Z}_2 \times \mathbb{Z}_2$).

We remark here that subsequent to our work, Khot [14] proved that even the restricted version of Max E3-NAE-Sat(G) where group multipliers are not allowed is approximation resistant; for this problem the group structure is, of course, not present at all. This result uses the recent multilayer PCP constructions due to Dinur, Guruswami, Khot and Regev [6] in a crucial way to avoid the need for group multipliers (or folding, as it is referred to in PCP jargon).

1.2 Our results

Preliminaries: First, we make explicit the easily seen result that an approximation algorithm for Max E2-Sat(d) with performance ratio better than $1-1/d^2$, i.e., better than the random assignment threshold, implies that any CSP over 2 variables can be approximated to within better than *its respective* random assignment threshold. In other words, Max E2-Sat(d) is the hardest problem in this class, and if there is some Max E2-CSP(d) which is approximation resistant, then Max E2-Sat(d) has to be approximation resistant.

Consequently, our interest is in the approximability of Max E2-Sat(d), specifically to either find a polynomial time approximation algorithm with performance ratio greater than $1 - 1/d^2$ or to prove a tight hardness result that the trivial $1 - 1/d^2$ is the best one can hope for. While we are unable to resolve this question, we consider and prove results for two predicates whose difficulty sandwiches that of solving Max E2-Sat(d): namely Max Co-BIJ(d) and Max E3-NAE-Sat($\mathbb{Z}_d$). The former problem is (in a loose sense) the natural 2-CSP which is next in "easiness" after Max E2-Sat(d) as far as approximating better than the random assignment threshold is concerned. There is an approximation preserving reduction from Max E2-Sat(d) to Max E3-NAE-Sat($\mathbb{Z}_d$), and thus proving the approximation resistance of Max E3-NAE-Sat($\mathbb{Z}_d$) is a natural first step if one tries to prove that Max E2-Sat(d) is approximation resistant.

Algorithms: For the Max Co-BIJ(d) problem, we prove that it is *not* approximation resistant by presenting a polynomial time approximation algorithm with performance ratio $1 - d^{-1} + 0.07d^{-4}$. This result implies that a large class of 2-CSPs, called *regular 2-CSPs* (defined below), are *not* approximation resistant. Viewing a 2-ary constraint C over domain size d as a subset of $[d] \times [d]$, the constraint is said to be *r-regular* if for each $a \in [d]$, $|\{x : (x, a) \in C\}| = |\{y : (a, y) \in C\}| = r$ (the term regular comes from the fact that the bipartite graph defined by C is regular). The constraint is *regular* if it is r-regular for some $1 \leq r < d$. A 2-CSP is regular if all the constraints in the CSP are regular and it is r-regular if all the constraints are r-regular.

Our result for regular 2-CSPs includes as a special case the result of Frieze and Jerrum [8] that Max d-Cut can be approximated to better than its random threshold. Our performance ratio is weaker, but our analysis is simpler and gives a more general result. Another special case is the result for Max E2-Lin(d) where our result actually improves the approximation ratio of Andersson *et al* [3]. Recently, Khot [13] gave a simpler algorithm that beats the random assignment threshold for Max E2-Lin(d) as well as the more general Max BIJ(d) problems—his result is actually more general and can find a near-satisfying assignment given a near-satisfiable instance, i.e., an instance where the optimum solution satisfies a fraction $1 - \varepsilon$ of constraints. Our approximation algorithm for Max Co-BIJ(d) is based on a semidefinite programming relaxation, similar to that used for Max BIJ(d) by Khot [13], combined with a rounding scheme used by Andersson [1, 2] to construct an approximation algorithm for Max d-Section, the generalization of Max Bisection to domains of size d. Technically, we view this algorithmic result as the main contribution of this paper.

Inapproximability results: For the Boolean case, $d = 2$, it is known that Max E3-NAE-Sat can be approximated to better than random. The GW-algorithm [9] for Max E2-Sat essentially gives such an algorithm, and the performance ratio was later improved by Zwick [20]. We prove that for larger domains, the problem becomes approximation resistant; in other words, it is **NP**-hard to approximate Max E3-NAE-Sat($\mathbb{Z}_d$) to better than $(1 - 1/d^2 + \varepsilon)$ for every $d \geq 3$ and every $\varepsilon > 0$. This result rules out the possibility of a non-trivial algorithm for Max E2-Sat(d) that works by reducing it to Max E3-NAE-Sat($\mathbb{Z}_d$). In fact, we prove that for every finite group G which is not of the form $\mathbb{Z}_2 \times \mathbb{Z}_2 \times \cdots \times \mathbb{Z}_2$, Max E3-NAE-Sat($G$) is hard to approximate within a factor $(1 - 1/|G|^2 + \varepsilon)$.

We remark that the above hardness results hold with *perfect completeness*; in other words, the stated approximation factors are hard to obtain even on satisfiable instances of the concerned constraint satisfaction problems.

Conclusions: We are not able to completely resolve the status of the 2-CSP problem over larger domains. Using reductions, we prove a hardness result of $1 - \Omega(1/d^2)$ for Max E2-Sat(d), which compares reasonably well with the $(1-1/d^2)$ random assignment threshold. For satisfiable instances of Max E2-Sat(d), we prove a hardness result of $1-\Omega(1/d^3)$. Nevertheless, we conjecture that there is an approximation algorithm beating the random assignment threshold for Max E2-Sat(d), and hence for all instances of 2-CSP.

Organization: We begin with a brief Section 2 highlighting why Max E2-Sat(d) is the hardest Max E2-CSP(d) problem in terms of beating the random assignment threshold. Next, in Section 3 we prove that every Max Co-BIJ(d) problem admits an algorithm that beats the random assignment threshold, and record some of its consequences. In Section 4, we prove that Max E3-NAE-Sat(G) is approximation resistant for most groups, including $G = \mathbb{Z}_d$ (the case of most interest in the context of Max E2-Sat(d)). Finally, we record results that directly apply to Max E2-Sat(d) in Section 5.

2 The "Universality" of Max E2-Sat(d)

We note that the existence of an approximation algorithm that beats the random threshold for every 2-CSP is equivalent to the existence of such an algorithm for Max E2-Sat(d). Thus, an algorithm for Max E2-Sat(d) with performance ratio better than $(1 - 1/d^2)$ will imply that no 2-CSP is approximation resistant, thus resolving our conjecture that every 2-CSP is "easy".

This claim is seen by a "gadget" reducing an arbitrary k-CSP(d) to Max Ek-Sat(d). Given an instance of any k-CSP(d), construct an instance of Max Ek-Sat(d) by repeating the following for every constraint C in the original k-CSP(d): For every non-satisfying assignment to C, add one Ek-SAT(d) constraint which has precisely this non-satisfying assignment. If an assignment satisfies C then it also satisfies all the Ek-SAT(d) constraints in the gadget, and otherwise it satisfies precisely all but one of the Ek-SAT(d) constraints. Using this fact, it is straightforward to show that if Max Ek-Sat(d) can be approximated beyond the random threshold, the above procedure gives an approximation algorithm that approximates an arbitrary k-CSP(d) beyond the random threshold. Conversely, if any k-CSP(d) at all is approximation resistant, then Max Ek-Sat(d) must be approximation resistant.

3 Approximation algorithm for Max Co-BIJ(d)

To construct an approximation algorithm for Max Co-BIJ(d) we combine a modification of the semidefinite relaxation used by Khot [13] for the Max BIJ(d) problem with a modification of the randomized rounding used by Andersson [1, 2] for the Max d-Section problem. Recall that a specific clause in the Max Co-BIJ(d) problem is of the form (x, x', π), where x and x' are variables in the Max Co-BIJ(d) instance and π is a permutation, and that the clause is satisfied unless $x = j$ and $x' = \pi(j)$ for some j. In our semidefinite relaxation of Max Co-BIJ(d) there are d vectors $\{u_0^x, \ldots, u_{d-1}^x\}$ for every variable x in the Max Co-BIJ(d) instance. Intuitively, the vector u_j^x sets the value of the variable x to j. To prove that our algorithm beats the random assignment threshold, we first establish that the semidefinite program in Fig. 1 is a relaxation of Max Co-BIJ(d), then prove that the rounding scheme proposed in Fig. 2 is well-defined, and finally analyze the expected performance of the rounding scheme using local analysis.

$$\begin{array}{lll}
\text{maximize} & \displaystyle\sum_{x,x',\pi} w_{x,x',\pi}\left(1-\sum_{j=0}^{d-1}\langle u_j^x, u_{\pi(j)}^{x'}\rangle\right) & \\
\text{such that} & \langle u_j^x, u_{j'}^{x'}\rangle \geq 0 & \forall x \in X, x' \in X, j \in \mathbb{Z}_d, j' \in \mathbb{Z}_d \\
& \langle u_j^x, u_{j'}^{x}\rangle = 0 & \forall x \in X, (j,j') \in \mathbb{Z}_d^2 : j \neq j' \\
& \displaystyle\sum_{j=0}^{d-1}\langle u_j^x, u_j^x\rangle = 1 & \forall x \in X \\
& \displaystyle\sum_{j=0}^{d-1}\sum_{j'=0}^{d-1}\langle u_j^x, u_{j'}^{x'}\rangle = 1 & \forall (x,x') \in X^2 : x \neq x'
\end{array}$$

Figure 1. Semidefinite relaxation of Max Co-BIJ(d) with variable set X. A clause in the Max Co-BIJ(d) instance is denoted by (x, x', π) where $x \in X$ and $x' \in X$ are variables and $\pi\colon \mathbb{Z}_d \to \mathbb{Z}_d$ is a permutation. The clause is satisfied unless $x = j$ and $x' = \pi(j)$ for some $j \in \mathbb{Z}_d$. Each clause (x, x', π) has a non-negative weight $w_{x,x',\pi}$ associated with it.

1. Solve the semidefinite program in Fig. 1.
2. Denote by u_j^x the vectors obtained from the solution.
3. For every $(x, j) \in X \times \mathbb{Z}_d$, let $v_j^x = u_j^x - \frac{1}{d}\sum_{j=0}^{d-1} u_j^x$.
4. Select r from a dn-dimensional $\mathrm{N}(0, 1)$ Gaussian distribution.
5. Set $q_{xj} = \frac{1}{d} + K\langle r, v_j^x\rangle$ for all $(x, j) \in X \times \mathbb{Z}_d$.
6. For each $x \in X$,
   – set $p_{xj} = q_{xj}$ if $q_{xj} \in [0, 2/d]$ for all $j \in \mathbb{Z}_d$;
   – set $p_{xj} = 1/d$ for all $j \in \mathbb{Z}_d$ otherwise.
7. For each $x \in X$, let $x = j$ with probability p_{xj}.

Figure 2. Approximation algorithm for Max Co-BIJ(d) with variable set X. The algorithm is parameterized by the positive constant K.

Lemma 1. *The semidefinite program in Fig. 1 is a relaxation of Max Co-BIJ(d).*

Proof. Suppose that we have an instance of Max Co-BIJ(d) with variable set X. Given an assignment ρ to the variables in X consider the following configuration of vectors: Let a be a vector of unit norm and define

$$u_j^x = \begin{cases} a & \text{if } j = \rho(x), \\ 0 & \text{otherwise.} \end{cases}$$

This configuration of vectors is feasible for the semidefinite program in Fig. 1 and the corresponding value of the objective function is exactly the weight of the satisfied equations in the Max Co-BIJ(d) instance. ∎

Lemma 2. *Suppose that $\{u_j^x : (x,j) \in X \times \mathbb{Z}_d\}$ is a feasible solution to the semidefinite program in Fig. 1. Then the barycenter*

$$b_x = \frac{1}{d}\sum_{j=0}^{d-1} u_j^x$$

of the vectors $\{u_j^x : j \in \mathbb{Z}_d\}_{j=0}^{d-1}$ is independent of x.

Proof. The constraints in the semidefinite program imply that $|b_x|^2 = |b_{x'}|^2 = \langle b_x, b_{x'}\rangle = d^{-2}$. ∎

Lemma 3. *for every clause (x, x', π) in the Max Co-BIJ(d) instance, the algorithm in Fig. 2 satisfies (x, x', π) with probability at least*

$$\left(1 - \sum_{j=0}^{d-1} \langle u_j^x, u_{\pi(j)}^{x'}\rangle\right) \int_B \left(1 - \frac{1}{d} + \frac{K^2 r_1^2}{d}\right) dP(r)$$

where K is any positive constant, the vectors u_j^x and $u_{j'}^{x'}$ are as described in the algorithm, $B = \{r \in \mathbb{R}^{2d} : |r| \le 1/Kd\}$, and P is the probability distribution of a $2d$-dimensional Gaussian with mean zero and identity covariance matrix.

Proof. Consider an arbitrary clause (x, x', π) and the corresponding values q_{xj} computed by the algorithm. Let $B = \{r \in \mathbb{R}^{2d} : |r| \le 1/Kd\}$. When $r \in B$, both q_{xj} and $q_{x'j}$ are in the interval $[0, 2/d]$; hence the clause (x, x', π) is satisfied with probability

$$\begin{aligned} 1 - \sum_{j=0}^{d-1} p_{xj} p_{x',\pi(j)} &= 1 - \sum_{j=0}^{d-1} \left(\frac{1}{d} + K\langle r, v_j^x\rangle\right)\left(\frac{1}{d} + K\langle r, v_{\pi(j)}^{x'}\rangle\right) \\ &= 1 - \frac{1}{d} - K^2 \sum_{j=0}^{d-1} \langle r, v_j^x\rangle\langle r, v_{\pi(j)}^{x'}\rangle \end{aligned}$$

given r in this case. By the definition of v_j^x from the algorithm, $\langle r, v_j^x\rangle\langle r, v_{\pi(j)}^{x'}\rangle = \langle r, u_j^x - b\rangle\langle r, u_{\pi(j)}^{x'} - b\rangle = \langle r, u_j^x\rangle\langle r, u_{\pi(j)}^{x'}\rangle - \langle r, u_j^x\rangle\langle r, b\rangle - \langle r, u_{\pi(j)}^{x'}\rangle\langle r, b\rangle + \langle r, b\rangle\langle r, b\rangle$ where b is the barycenter of the vectors $\{u_j^x : j \in \mathbb{Z}_d\}$, which is independent of x according to Lemma 2. Therefore

$$-K^2 \sum_{j=0}^{d-1} \langle r, v_j^x\rangle\langle r, v_{\pi(j)}^{x'}\rangle = K^2\left(d\langle r, b\rangle\langle r, b\rangle - \sum_{j=0}^{d-1} \langle r, u_j^x\rangle\langle r, u_{\pi(j)}^{x'}\rangle\right).$$

To conclude, the probability that the clause is satisfied given r is

$$1 - \frac{1}{d} + K^2\left(d\langle r, b\rangle\langle r, b\rangle - \sum_{j=0}^{d-1} \langle r, u_j^x\rangle\langle r, u_{\pi(j)}^{x'}\rangle\right) \tag{1}$$

when $r \in B$. Integrating over B, we can thus bound the probability that the clause is satisfied from below by

$$\int_B \left(1 - \frac{1}{d} + K^2\left(d\langle r, b\rangle\langle r, b\rangle - \sum_{j=0}^{d-1} \langle r, u_j^x\rangle\langle r, u_{\pi(j)}^{x'}\rangle\right)\right) dP(r).$$

To compute the integral of $\langle r, b\rangle\langle r, b\rangle$, introduce an orthonormal basis $\{e_k\}$ such that $b = e_1/d$ and write $r = \sum_k r_k e_k$ in this basis. Then

$$\int_B \langle r, b\rangle\langle r, b\rangle \, dP(r) = \frac{1}{d^2}\int_B r_1^2 \, dP(r)$$

where the last integral is actually independent of the basis since both P and B are spherically symmetric (r_1 is the first coordinate of the vector r in the basis $\{e_k\}$). To compute the integral of $\langle r, u_j^x\rangle\langle r, u_{\pi(j)}^{x'}\rangle$, we proceed similarly: Introduce an orthonormal basis $\{e_k\}$ such that $u_j^x = x_1 e_1$ and $u_{\pi(j)}^{x'} = y_1 e_1 + y_2 e_2$. Then

$$\int_B \langle r, u_j^x\rangle\langle r, u_{\pi(j)}^{x'}\rangle \, dP(r) = \int_B (r_1^2 x_1 y_1 + r_1 r_2 x_1 y_2) \, dP(r).$$

The integral of the second term vanishes since the integrand is odd and the interval is symmetric around the origin. Therefore

$$\int_B \langle r, u_j^x\rangle\langle r, u_{\pi(j)}^{x'}\rangle \, dP(r) = x_1 y_1 \int_B r_1^2 \, dP(r) = \langle u_j^x, u_{\pi(j)}^{x'}\rangle \int_B r_1^2 \, dP(r).$$

To conclude, we can write the probability that the clause is satisfied as $a - cx$ where

$$\begin{aligned}
a &= \int_B \left(1 - \frac{1}{d} + \frac{K^2 r_1^2}{d}\right) dP(r), \\
c &= K^2 \int_B r_1^2 \, dP(r), \\
x &= \sum_{j=0}^{d-1} \langle u_j^x, u_{\pi(j)}^{x'}\rangle.
\end{aligned}$$

We now find an α such that $a - cx \geq \alpha(1 - x)$. Since the expression (1) is a probability it is non-negative for every r. Hence $a \geq c \geq 0$ and, since the constraints in the semidefinite program imply that $x \geq 0$,

$$a - cx \geq a - ax = \left(1 - \sum_{j=0}^{d-1} \langle u_j^x, u_{\pi(j)}^{x'}\rangle\right) \int_B \left(1 - \frac{1}{d} + \frac{K^2 r_1^2}{d}\right) dP(r). \qquad \blacksquare$$

Theorem 1. *The algorithm in Fig. 2 with $K = 1/\sqrt{13d^3}$ is a randomized polynomial time approximation algorithm for Max Co-BIJ(d) with expected performance ratio $1 - d^{-1} + 0.07d^{-4}$ and thus better than the random assignment threshold.*

1. Select a random assignment ρ to the variables in the instance.
2. Replace each equation $ax + by = c$ with the $d-1$ inequations $ax + by \neq c_i$ for all $c_i \neq c$. Obtain an assignment τ by running the algorithm in Fig. 2 on this instance.
3. Pick the best of the assignments ρ and τ.

Figure 3. Approximation algorithm for Max E2-Lin(d).

Proof. Consider the algorithm in Fig. 2 applied to an instance of Max Co-BIJ(d). By Lemma 1, the semidefinite program in Fig. 1 is a relaxation of Max Co-BIJ(d); therefore the optimum value of the Max Co-BIJ(d) instance can be bounded by

$$\sum_{x,x',\pi} w_{x,x',\pi}\left(1 - \sum_{j=0}^{d-1}\langle u_j^x, u_{\pi(j)}^{x'}\rangle\right)$$

where the vectors u_j^x are the solution to the program. Now consider the rounding. By Lemma 3, the clause (x, x', π) is satisfied with probability

$$\left(1 - \sum_{j=0}^{d-1}\langle u_j^x, u_{\pi(j)}^{x'}\rangle\right)\int_B \left(1 - \frac{1}{d} + \frac{K^2 r_1^2}{d}\right) dP(r)$$

By Lemmas 12 and 13 (see appendix),

$$\int_B \left(1 - \frac{1}{d} + \frac{K^2 r_1^2}{d}\right) dP(r) \geq 1 - \frac{1}{d} + \frac{0.07}{d^4}$$

with the parameter choice $K^2 = d^{-3}/13$. So with this parameter choice, it now follows by linearity of expectation and the above bound on the optimum value of the Max Co-BIJ(d) instance that the expected weight of the solution produced by the algorithm is at least a factor $(1 - d^{-1} + 0.07d^{-4})$ times the optimum value of the Max Co-BIJ(d) instance. ∎

3.1 An approximation algorithm for Max E2-Lin(d)

We can use the above algorithm for Max Co-BIJ(d) to construct an algorithm also for Max E2-Lin(d): Simply replace an equation $ax + by = c$ with the $d-1$ inequations $ax + by \neq c_i$ for all $c_i \neq c$. Then an assignment that satisfies a linear equation satisfies all of the corresponding linear inequations and an assignment that does not satisfy a linear equation satisfies $d-2$ of the $d-1$ corresponding linear equations. This algorithm gives a performance ratio of $1/d + \Omega(1/d^4)$ which improves significantly on the previously best known ratio of $1/d + \Omega(1/d^{14})$ [3].

Theorem 2. *For all $d \geq 4$, the algorithm in Fig. 3 is a randomized polynomial time approximation algorithm for Max E2-Lin(d) with expected performance ratio $d^{-1} + 0.05d^{-4}$.*

Repeat the following for every relation in the Max Co-BIJ(d) instance:

1. Let R be the bipartite graph defined by the relation.
2. Let r be the degree of R.
3. Let R^c be the complement of R.
4. Decompose the edges of R^c into perfect matchings $R_1, \dots, R_{d-r}$.
5. Define the relations π_i such that $d_1 \sim_{\pi_i} d_2$ if $(d_1, d_2) \in R_i$.
6. Add the relations $\pi_1, \pi_2, \dots, \pi_{d-r}$ to the Max Co-BIJ(d) instance.

Figure 4. Construction of an instance of Max Co-BIJ(d) from an instance of a regular CSP.

Proof. If the optimum of the instance is smaller than a fraction $1 - 0.05d^{-3}$ of all equations, the random assignment—which satisfies an expected fraction $1/d$ of all equations—satisfies an expected fraction

$$\frac{1/d}{1 - 0.05d^{-3}} \geq \frac{1}{d} + \frac{0.05}{d^4}$$

of the optimum.

If the optimum is larger than a fraction $1 - 0.05d^{-3}$ of all equations, the optimum of the instance of inequations is at least $1 - 0.05d^{-3}(d-1)^{-1}$. By Theorem 1, our approximation for Max Co-BIJ(d) from Fig. 2 therefore finds a solution with expected weight at least a fraction

$$\left(1 - \frac{0.05}{d^3(d-1)}\right)\left(\frac{d-1}{d} + \frac{0.07}{d^4}\right) = \frac{d-1}{d} + \frac{0.02}{d^4} - \frac{0.0035}{d^7(d-1)}$$

of all inequations. An assignment satisfying a fraction $1/d + \alpha$ of the equations satisfies a fraction

$$\left(\frac{1}{d} + \alpha\right) + \frac{d-2}{d-1}\left(1 - \frac{1}{d} - \alpha\right) = \frac{d-1}{d} + \frac{\alpha}{d-1}$$

of the inequations and vice versa; therefore the assignment constructed above satisfies at least an expected fraction

$$\frac{1}{d} + \frac{0.02(d-1)}{d^4} - \frac{0.0035}{d^7} > \frac{1}{d} + \frac{0.02(d - 1 - 0.0035d^{-3})}{d^4}$$

of the equations. The function $d - 1 - 0.0035d^{-3}$ is increasing in d and for $d = 4$ it is $3 - 0.0035/64 > 2.5$. Therefore, at least an expected fraction $d^{-1} + 0.05d^{-4}$ of all equations are satisfied when $d \geq 4$. ∎

Corollary 1. *For all $d \geq 2$, there is a polynomial time approximation algorithm for Max E2-Lin(d) with expected performance ratio $d^{-1} + 0.05d^{-4}$ and thus better than the random assignment threshold.*

Proof. Algorithms for $d = 2$ and $d = 3$ have been provided by Goemans and Williamson [9, 10], for $d \geq 4$ the result follows by Theorem 2 ∎

1. Select a random assignment ρ to the variables in the instance.
2. Create an instance of Max Co-BIJ(d) as described in Fig. 4. Obtain an assignment τ by running the algorithm in Fig. 2 on this instance.
3. Pick the best of the assignments ρ and τ.

Figure 5. Approximation algorithm for regular 2-CSPs.

3.2 An approximation algorithm for regular 2-CSPs

We can obtain an approximation algorithm for all regular CSPs by a straightforward generalization of the ideas from the previous section. Given an r-regular 2-CSP, we proceed as follows: for every relation R defining the CSP, decompose R^c, the "bipartite complement" of the graph defined by R, into $(d-r)$ perfect matchings $\pi_R^1, \pi_R^2, \ldots, \pi_R^{d-r}$. Then let these matchings define the Max Co-BIJ(d) instance. An assignment that satisfies R satisfies all of the $d-r$ matchings while an assignment that does not satisfy R satisfies $d-r-1$ of them. Run the following two algorithms and take the assignment producing the largest weight as the result: The first algorithms selects a random assignment to the variables; the second algorithm runs the above algorithm for Max Co-BIJ(d).

Theorem 3. *For all $d \geq 2$ and all $1 \leq r \leq d-1$, the algorithm in Fig. 5 is a randomized polynomial time approximation algorithm for r-regular CSPs with expected performance ratio $r/d + \Omega(d^{-4})$.*

Proof. Consider the result of the algorithm on an arbitrary instance. If the optimum of the instance is smaller than a fraction $1-0.05(d-r)d^{-3}(d-1)^{-1}$ of all equations, the random assignment—which satisfies an expected fraction r/d of all equations—satisfies an expected fraction

$$\frac{r/d}{1-0.05(d-r)d^{-3}(d-1)^{-1}} \geq \frac{r}{d} + \frac{0.05r(d-r)}{d^4(d-1)}$$

of the optimum.

If the optimum is larger than a fraction $1-0.05(d-r)d^{-3}(d-1)^{-1}$ of all constraints, the optimum of the corresponding instance of Max Co-BIJ(d) is at least a fraction

$$1 - \frac{0.05(d-r)}{d^3(d-1)}\left(1 - \frac{d-r-1}{d-r}\right) = 1 - \frac{0.05}{d^3(d-1)}$$

By Theorem 1, our approximation for Max Co-BIJ(d) from Fig. 2 therefore finds a solution with expected weight at least a fraction

$$\left(1 - \frac{0.05}{d^3(d-1)}\right)\left(\frac{d-1}{d} + \frac{0.07}{d^4}\right) = \frac{d-1}{d} + \frac{0.02}{d^4} - \frac{0.0035}{d^7(d-1)}$$

of all constraints in the Max Co-BIJ(d) instance. An assignment satisfying a fraction $r/d+\alpha$ of the constraints in the r-regular CSP satisfies a fraction

$$\frac{d-r}{d-r}\left(\frac{r}{d} + \alpha\right) + \frac{d-r-1}{d-r}\left(1 - \frac{r}{d} - \alpha\right) = \frac{d-1}{d} + \frac{\alpha}{d-r}$$

of the constraints in the Max Co-BIJ(d) instance and vice versa; therefore the assignment constructed above satisfies at least a fraction

$$\frac{r}{d} + \frac{0.02(d-r)}{d^4} - \frac{0.0035(d-r)}{d^7(d-1)}$$

of the constraints in the r-regular CSP. ∎

It is not necessary that the various constraints be r-regular for the same r, and a similar argument also shows that every regular 2-CSP can be approximated in polynomial time beyond its random assignment threshold.

4 Hardness results

In this section, our aim is to prove that unlike the Boolean case, for every $d \geq 3$, Max E3-NAE-Sat($\mathbb{Z}_d$) is approximation resistant. We will actually prove that Max E3-NAE-Sat(G) is approximation resistant for pretty much every finite Abelian group.

Theorem 4. *For every constant $\varepsilon > 0$ and every finite Abelian group that is not isomorphic to $\mathbb{Z}_2^m$ for any positive integer m, it is* **NP***-hard to distinguish instances of Max E3-NAE-Sat(G) that are satisfiable from instances where at most a fraction $(1 - |G|^{-2} + \varepsilon)$ of the constraints are satisfiable.*

The proof of this result uses the by now standard PCP-based approach. Since there are many such proofs in the literature, we present a somewhat condensed argument here. For an excellent introduction to the paradigm, see Håstad's paper [11]

The starting point of our argument is the following hardness result, which follows from the celebrated PCP theorem [4] combined with a certain regularization procedure [7] and the parallel repetition theorem [16]:

Theorem 5. *Let V and Φ be two sets of variables. Let R and S, respectively, be the range of the variables in V and Φ, respectively. Let Ψ be a set of triples $(v, \phi, \pi) \in V \times \Phi \times R^S$ with the following properties: Every $v \in V$ occurs in the same number of triples; every $\phi \in \Phi$ occurs in the same number of triples; the functions $\pi: S \to R$ are onto.*

Then there exists a universal constant $\mu > 0$ such that for every large enough constant $|R|$ there exists a constant $|S| > |R|$ such that it is **NP***-hard to distinguish between the following two cases:*

1) There exist assignments $\Pi_V: V \to R$ and $\Pi_\Phi: \Phi \to S$ such that for every $(v, \phi, \pi) \in \Psi$, $\Pi_V(v) = \pi(\Pi_\Phi(\phi))$.

2) There are no assignments $\Pi_V: V \to R$ and $\Pi_\Phi: \Phi \to S$ such that for more than a fraction $|R|^{-\mu}$ of the triples $(v, \phi, \pi) \in \Psi$, $\Pi_V(v) = \pi(\Pi_\Phi(\phi))$.

Our argument then proceeds by constructing a tailored PCP that, in essence, simulates the above problem by posing not-all-equal queries to a proof containing purported assignments Π_V, Π_Φ as defined above. Once we establish that our PCP has certain completeness and soundness properties, Theorem 4 is immediate.

4.1 The proof in our PCP

The proof in our PCP should contain an encoding of assignments Π_V, Π_Φ satisfying property 1 in Theorem 5. Concretely, the assignments are encoded with the *long G-code*. Let G^R denote the set of functions from R to G. The long G-code of some assignment σ to a $v \in V$ is a function $A_{v,\sigma}: G^R \to G$ defined by $A_{v,\sigma}(f) = f(\sigma)$.

Given an instance of the problem defined in Theorem 5, our proof contains for each $v \in V$ a string of length $|G|^{|R|}$, supposedly a function $A_v: G^R \to G$, and for each $\phi \in \Phi$ a string of length $|G|^{|S|}$, supposedly a function $A_\phi: G^S \to G$. The proof is *correct* if there exists assignments Π_V

and Π_Φ satisfying property 1 in Theorem 5 such that A_v is the long code of $\Pi_V(v)$ and A_ϕ is the long G-code of $\Pi_\Phi(\phi)$ for every v and ϕ.

Our analysis requires that all the tables in the PCP are *folded over G*. This means that for $f \in G^R$ and $g \in G$, $A_v(gf) = gA_v(f)$, where the function gf is defined by the map $gf : r \mapsto gf(r)$ where the latter expression denotes multiplication in G of g by $f(r)$. This property can be enforced by employing certain *access conventions* in the verifier; see Håstad's paper [11, §§ 2.5–2.6] for details.

4.2 Tools from harmonic analysis

Our analysis relies heavily on the representation theory for Abelian groups and the associated Fourier transform. We now give a brief account of those concepts—for more details, we refer the reader to Håstad's paper [11, §§ 2.4–2.6] or Terras's book [18].

Let G be an Abelian group and $\hat{G}$ be its dual group, i.e., the group of all homomorphisms from G to $\mathbb{C}$. A central concept in the representation theory is the action of a member of $\hat{G}$ on a member of G. Since the dual groups consists of functions from G to $\mathbb{C}$ (the set of complex numbers), such an action is simply the evaluation of a member of $\hat{G}$ at a member of G.

Example 1. The group $\mathbb{Z}_d$ for some positive integer d is isomorphic to the group consisting of powers of $e^{2\pi i/d}$ with multiplication as the group operation and 1 as identity. The dual group then consists of all functions $x \mapsto x^n$ for the integers n from 0 to $d-1$. The product of the functions $x \mapsto x^n$ and $x \mapsto x^m$ is the function $x \mapsto x^{m+n}$, where addition is defined modulo d. This implies that the trivial homomorphism, that maps every group element to 1 and thus is the function $x \mapsto x^0$, is the identity in the dual group.

The above example motivates the following notation in the general case: For $g \in G$ and $\gamma \in \hat{G}$ we let g^γ denote $\gamma(g)$. We let multiplication be the group operation in G and addition be the group operation in $\hat{G}$ and denote the identities by $\mathbf{1}$ and $\mathbf{0}$, respectively. Then we can write the following identities: $g^{\gamma_1} g^{\gamma_2} = g^{\gamma_1+\gamma_2}$, $g_1^\gamma g_2^\gamma = (g_1 g_2)^\gamma$, $(g^2)^\gamma = g^{2\gamma} = g^\gamma g^\gamma$, $g^{-\gamma} = (g^{-1})^\gamma$, $g^{\mathbf{0}} = 1$, $\mathbf{1}^\gamma = 1$. We also need the following symmetry relations:

$$\sum_{g\in G} g^\gamma = \begin{cases} |G| & \text{if } \gamma = \mathbf{0}, \\ 0 & \text{otherwise,} \end{cases} \qquad \sum_{\gamma\in\hat{G}} g^\gamma = \begin{cases} |G| & \text{if } g = \mathbf{1}, \\ 0 & \text{otherwise.} \end{cases}$$

It turns out that the dual group of a direct product $G \times H$ is (isomorphic to) the direct product $\hat{G} \times \hat{H}$. For $(g,h) \in G \times H$ and $(\gamma,\theta) \in \hat{G} \times \hat{H}$, it then holds that $(g,h)^{(\gamma,\theta)} = g^\gamma h^\theta$.

For readers familiar with the theory of group actions on groups as described by, e.g., Rose [17], we remark that our notation g^γ does not describe such an action since g^γ is a complex number rather than an element of G. Also, in our notation, an expression such as $(g^{\gamma_1})^{\gamma_2}$ is meaningless. Nevertheless, we believe that the exponential notation is indeed a useful one also in our setting.

Now let f be a function from G to $\mathbb{C}$. Then we can write f as a Fourier series $f(g) = \sum_{\gamma\in\hat{G}} \hat{f}_\gamma g^\gamma$ where $\hat{f}_\gamma = |G|^{-1} \sum_{g\in G} f(g) g^{-\gamma}$. Moreover the following version of Plancherel's equality holds in this case: $\sum_{\gamma\in\hat{G}} |\hat{f}_\gamma|^2 = |G|^{-1} \sum_{g\in G} |f(g)|^2$.

In our analysis, we need some identities regarding the Fourier transform of a folded table and the Fourier transform of related tables. All these identities have already been obtained by Håstad [11, § 2.6], we only state them here for easy reference. Since a function from R to G can be identified by a table of $|R|$ values from G, we identify G^R with the group $G^{|R|}$. For an element $f \in G^R$, we let $f(r)$ denote the coordinate in f corresponding to r. Now let A be a function from G^R to $\mathbb{C}$. We

1. Select (v, ϕ, π) uniformly at random from Ψ.
2. Select $f \in G^R$ and $h \in G^S$ uniformly at random.
3. Select e by defining $e(s) = (g_0(s), g_1(s), \dots, g_m(s))$ where
   – $g_0(s)$ is uniformly distributed in $G_0 \setminus \{\mathbf{1}\}$.
   – $g_i(s) = g'g''$ for $i \neq 0$ where g' and g'' are selected as in Lemma 6 with $d = 2^{\alpha_i}$.
   The above random selections for different s are independent.
4. Accept if $A_v(f)$, $A_\phi(h)$ and $A_\phi((f \circ \pi)^{-1}h^2e)$ are not all equal;
   Reject otherwise.

Figure 6. The verifier used to prove optimal approximation hardness for Max E3-NAE-Sat(G) for a finite Abelian group $G \cong G_0 \times \mathbb{Z}_{2^{\alpha_1}} \times \mathbb{Z}_{2^{\alpha_2}} \times \cdots \times \mathbb{Z}_{2^{\alpha_m}}$ where G_0 is a finite Abelian group of odd order and $\alpha_i > 1$ for all i. Above, $(f \circ \pi)^{-1}$ denotes the map $s \mapsto (f(\pi(s)))^{-1}$, the latter inverse being inverse in the group G. Similarly, h^2 denotes the map $s \mapsto (h(s))^2$.

can write this function as a Fourier series $A(f) = \sum_{\alpha \in \hat{G}^R} \hat{A}_\alpha f^\alpha$. To describe the elements of $\hat{G}^R$, it is convenient to view them as functions from R to $\hat{G}$; then we can write $f^\alpha = \prod_{r \in R}(f(r))^{\alpha(r)}$. The latter expression is well-defined since $f(r) \in G$ and $\alpha(r) \in \hat{G}$.

Lemma 4. *Given an Abelian group G and a finite set R, suppose that $A_v\colon G^R \to G$ is folded over G and define $A\colon G^R \to \mathbb{C}$ by the map $f \mapsto (A_v(f))^\gamma$ for some $\gamma \in \hat{G} \setminus \{\mathbf{0}\}$. Then $\hat{A}_\alpha = 0$ unless $\gamma = \sum_{r \in R} \alpha(r)$. In particular, $\hat{A}_{\mathbf{0}} = 0$.*

Lemma 5. *Given an Abelian group G, two finite sets R and S, a function $f\colon R \to G$, an onto function $\pi\colon S \to R$, and a $\beta \in \hat{G}^S$, let $\pi_G(\beta) \in \hat{G}^R$ be defined by $\pi_G(\beta)(r) = \sum_{s \in \pi^{-1}(r)} \beta(s)$. Then $(f \circ \pi)^\beta = f^{\pi_G(\beta)}$.*

Corollary 2. *Given G, R, S, π and π_G as above, suppose that $A_\phi\colon G^S \to G$ is folded over G and define $B\colon G^S \to \mathbb{C}$ by the map $h \mapsto (A_\phi(h))^\gamma$ for some $\gamma \in \hat{G} \setminus \{\mathbf{0}\}$. Then $\pi_G(\beta) = \mathbf{0}$ implies $\hat{B}_\beta = 0$.*

Proof. Since $\pi_G(\beta)(r) = \sum_{s \in \pi^{-1}(r)} \beta(s)$, $\pi_G(\beta) = \mathbf{0}$ implies that $\sum_{s \in S} \beta(s) = \mathbf{0} \neq \gamma$. ∎

4.3 Proof of Theorem 4

Remember that every finite Abelian group is isomorphic to some group of the form

$$G_0 \times \mathbb{Z}_{2^{\alpha_1}} \times \mathbb{Z}_{2^{\alpha_2}} \times \cdots \times \mathbb{Z}_{2^{\alpha_m}} \tag{2}$$

where G_0 is a group of odd order and $|G| = |G_0|2^{\alpha_1+\alpha_2+\cdots+\alpha_m}$. Every $g \in G$ can thus be written as a tuple $(g_0, g_1, g_2, \dots, g_m)$ where $g_0 \in G_0$ and $g_i \in \mathbb{Z}_{2^{\alpha_i}}$ for $i = 1, 2, \dots, m$. Similarly, every $\gamma \in \hat{G}$ can be decomposed as $(\gamma_0, \gamma_1, \gamma_2, \dots, \gamma_m)$ and it then holds that $g^\gamma = g_0^{\gamma_0} g_1^{\gamma_1} g_2^{\gamma_2} \cdots g_m^{\gamma_m}$. Since our PCP in Fig. 6 does not appear to work for groups such that $\alpha_i = 1$ for some i in the above expansion, we assume for now that $\alpha_i > 1$ for all i. Given a hardness result for this restricted case, Lemma 9 then gives us the more general result of Theorem 4.

We represent $\mathbb{Z}_{2^{\alpha_i}}$ as 2^{α_i}th roots of unity with multiplication as the group operator. By representing the elements of the dual group—recall that they are actually the homomorphisms $x \mapsto x^n$ for integers n—as $\{0, 1, 2, \dots, 2^{\alpha_i} - 1\}$ with addition modulo 2^{α_i} as the group operator, we can still use the syntax $g_i^{\gamma_i}$ to denote an element $\gamma_i \in \hat{\mathbb{Z}}_{2^{\alpha_i}}$ acting on some $g_i \in \mathbb{Z}_{2^{\alpha_i}}$. Before embarking on the analysis of the verifier, we need to prove some properties of the "error function" e defined in Fig. 6.

Lemma 6. *For an arbitrary $d \geq 4$ that is a power of two, let $\omega = e^{2\pi i/d}$ and select g' uniformly from $\{\omega^{4i}, \omega^{4i+1}\}_{i=0}^{d/4-1}$ and g'' uniformly from $\{\omega^{4i+1}, \omega^{4i+2}\}_{i=0}^{d/4-1}$. Then, for $0 \leq \gamma \leq d-1$,*

$$g'g'' \neq 1 \quad \textit{with probability } 1, \tag{3}$$

$$\mathrm{E}[(g')^\gamma] = \begin{cases} 1 & \textit{if } \gamma = 0, \\ (1+i)/2 & \textit{if } \gamma = d/4, \\ (1-i)/2 & \textit{if } \gamma = 3d/4, \\ 0 & \textit{otherwise.} \end{cases} \tag{4}$$

$$\mathrm{E}[(g'')^\gamma] = \omega^\gamma \, \mathrm{E}[(g')^\gamma], \tag{5}$$

$$|\,\mathrm{E}[(g'g'')^\gamma]| = \begin{cases} 1 & \textit{if } \gamma = 0, \\ 1/2 & \textit{if } \gamma = d/4 \textit{ or } \gamma = 3d/4, \\ 0 & \textit{otherwise.} \end{cases} \tag{6}$$

Proof. By construction, $g'g''$ can only assume the values $\omega^{4(i+i')+1}, \omega^{4(i+i')+2}, \omega^{4(i+i')+3}$. To prove the equality (4), notice that

$$\mathrm{E}[(g')^\gamma] = \frac{1+\omega^\gamma}{d/2} \sum_{k=0}^{d/4-1} \omega^{4\gamma k} = \frac{(1+\omega^\gamma)(1-\omega^{\gamma d})}{d(1-\omega^{4\gamma})/2} = 0,$$

where the second equality is valid when $\omega^{4\gamma} \neq 1$, i.e., when γ is not an integer multiple of $d/4$. For the other cases,

$$\mathrm{E}[(g')^0] = \mathrm{E}[1] = 1,$$

$$\mathrm{E}[(g')^{d/4}] = \frac{1+\omega^{d/4}}{d/2} \sum_{k=0}^{d/4-1} 1 = \frac{1+i}{2},$$

$$\mathrm{E}[(g')^{d/2}] = \frac{1+\omega^{d/2}}{d/2} \sum_{k=0}^{d/4-1} 1 = 0,$$

$$\mathrm{E}[(g')^{3d/4}] = \frac{1+\omega^{3d/4}}{d/2} \sum_{k=0}^{d/4-1} 1 = \frac{1-i}{2}.$$

The equality (5) follows since g'' is distributed as $\omega g'$ and (6) follows since g' and g'' are selected independently. ∎

Corollary 3. *Let e be selected as in step 3 in Figure 6. Then, for every $\beta \in \hat{G}^S$, $|\,\mathrm{E}[e^\beta]| \leq 2^{-|\beta|}$, where $|\beta|$ denotes the number of $s \in S$ such that $\beta(s) \neq \mathbf{0}$.*

Proof. Since e is selected such that all $e(s)$ are independent, $|\,\mathrm{E}[e^\beta]| = \prod_s |\,\mathrm{E}[(e(s))^{\beta(s)}|$. Factors corresponding to s such that $\beta(s) = \mathbf{0}$ evaluate to 1, for $\beta \neq \mathbf{0}$ we write $e(s) = (g_0, g_1, \ldots, g_m)$ and $\beta(s) = (\gamma_0, \gamma_1, \ldots, \gamma_m)$; then $|\,\mathrm{E}[(e(s))^{\beta(s)}| = \prod_{i=0}^m |\,\mathrm{E}[g_i^{\gamma_i}]|$ where g_0 is selected uniformly from

$G \setminus \{\mathbf{1}\}$ and the other g_i are selected as described in Lemma 6 with $d = 2^{\alpha_i}$. If γ_i is the trivial character, $g_i^{\gamma_i} = 1$. A straightforward calculation shows that if $\gamma_0 \neq \mathbf{0}$,

$$|\mathrm{E}[g_0^{\gamma_0}]| = \left| \frac{1}{|G_0| - 1} \sum_{g_0 \in G_0 \setminus \{1\}} g_0^{\gamma_0} \right| = \left| \frac{-1}{|G_0| - 1} + \frac{1}{|G_0| - 1} \sum_{g_0 \in G_0} g_0^{\gamma_0} \right| = \frac{1}{|G_0| - 1}.$$

For $i > 0$, Lemma 6 shows that $|g_i^{\gamma_i}| \leq 1/2$ if $\gamma_i \neq 0$. ∎

To see that the PCP in Fig. 6 indeed gives the claimed hardness result, we in essence reduce the problem described in Theorem 5 to an instance of Max E3-NAE-Sat(G) using the PCP.

Lemma 7. *Suppose that there exists assignments Π_V and Π_Φ as described in item 1 in Theorem 5. Then, there exists a proof for the verifier in Fig. 6 such that the verifier accepts with probability one.*

Proof. We construct a proof for our PCP as correct long codes of the assignments Π_V and Π_Φ. Then $A_v(f) = f(\Pi_V(v))$, $A_\phi(h) = h(\Pi_\Phi(\phi))$, and

$$\begin{aligned} A_\phi((f \circ \pi)^{-1} h^2 e) &= ((f \circ \pi)(\Pi_\Phi(\phi)))^{-1} (h(\Pi_\Phi(\phi)))^2 e(\Pi_\Phi(\phi)) \\ &= (f(\Pi_V(v)))^{-1} (h(\Pi_\Phi(\phi)))^2 e(\Pi_\Phi(\phi)), \end{aligned}$$

where the last equality follows since $\pi(\Pi_\Phi(\phi)) = \Pi_V(v)$ for all triples (v, ϕ, π) selected by the verifier. It is easy to see that these three values can never be equal. ∎

The soundness is a bit more complicated to analyze. The main idea is to prove that for every constant $\varepsilon > 0$ it is possible to select $|R|$ in such a way that the fact that the test accepts with probability $1 - |G|^{-2} + \varepsilon$ implies that there exist assignments Π_V and Π_Φ as described in item 1 in Theorem 5.

Lemma 8. *Given a constant $\varepsilon > 0$ and an instance Ψ as described in Theorem 5 with $|R| > \varepsilon^{-2/\mu}$, construct a verifier as described in Fig. 6. If the verifier accepts with probability $1 - |G|^{-2} + \varepsilon$ then case 2 in Theorem 5 cannot hold.*

Proof. The test accepts unless all three queries return the same value, therefore we can write the acceptance probability as $1 - \sum_{g \in G} \mathrm{E}[I_g]$ where the expectation is over the verifier's choice of $(v, \phi, \pi), f, h, e$ and I_g denotes the indicator for the event that all three queries return g:

$$I_g = \frac{1}{|G|^3} \Bigl(1 + \sum_{\gamma_1 \neq \mathbf{0}} (g^{-1} A_v(f))^{\gamma_1}\Bigr) \Bigl(1 + \sum_{\gamma_2 \neq \mathbf{0}} (g^{-1} A_\phi(h))^{\gamma_2}\Bigr) \Bigl(1 + \sum_{\gamma_3 \neq \mathbf{0}} (g^{-1} A_\phi((f \circ \pi)^{-1} h^2 e))^{\gamma_3}\Bigr).$$

We now expand the products and get

$$\begin{aligned} I_g = \frac{1}{|G|^3} \Bigl(1 &+ \sum_{\gamma_1 \neq \mathbf{0}} (g^{-1} A_v(f))^{\gamma_1} + \sum_{\gamma_2 \neq \mathbf{0}} (g^{-1} A_\phi(h))^{\gamma_2} + \sum_{\gamma_3 \neq \mathbf{0}} (g^{-1} A_\phi((f \circ \pi)^{-1} h^2 e))^{\gamma_3} \\ &+ \sum_{\gamma_1 \neq \mathbf{0}} \sum_{\gamma_2 \neq \mathbf{0}} (g^{-1} A_v(f))^{\gamma_1} (g^{-1} A_\phi(h))^{\gamma_2} \\ &+ \sum_{\gamma_1 \neq \mathbf{0}} \sum_{\gamma_3 \neq \mathbf{0}} (g^{-1} A_v(f))^{\gamma_1} (g^{-1} A_\phi((f \circ \pi)^{-1} h^2 e))^{\gamma_3} \\ &+ \sum_{\gamma_2 \neq \mathbf{0}} \sum_{\gamma_3 \neq \mathbf{0}} (g^{-1} A_\phi(h))^{\gamma_2} (g^{-1} A_\phi((f \circ \pi)^{-1} h^2 e))^{\gamma_3} \end{aligned}$$

$$+\sum_{\gamma_1\neq\mathbf{0}}\sum_{\gamma_2\neq\mathbf{0}}\sum_{\gamma_3\neq\mathbf{0}}(g^{-1}A_v(f))^{\gamma_1}(g^{-1}A_\phi(h))^{\gamma_2}(g^{-1}A_\phi((f\circ\pi)^{-1}h^2e))^{\gamma_3}\Bigg).$$

Let us consider the contributions of the above terms when we sum I_g over all $g \in G$ and take expectation over f, h, e. The constant term contributes $|G|/|G|^3 = 1/|G|^2$. The first three sums vanish since

$$\sum_{g\in G}\sum_{\gamma\in\hat{G}\setminus\{\mathbf{0}\}}(gh)^\gamma$$

where h is independent of g can be rewritten as

$$\sum_{\gamma\in\hat{G}\setminus\{\mathbf{0}\}}h^\gamma\sum_{g\in G}g^\gamma = 0$$

where the equality follows since $\sum_{g\in G} g^\gamma = 0$ for every $\gamma \neq \mathbf{0}$. The three double-sums also vanish, but the argument is somewhat more involved. Consider the first double-sum:

$$\sum_{g\in G}\sum_{\gamma_1\neq\mathbf{0}}\sum_{\gamma_2\neq\mathbf{0}}(g^{-1}A_v(f))^{\gamma_1}(g^{-1}A_\phi(h))^{\gamma_2} = \sum_{\gamma_1\neq\mathbf{0}}\sum_{\gamma_2\neq\mathbf{0}}(A_v(f))^{\gamma_1}(A_\phi(h))^{\gamma_2}\sum_{g\in G}g^{-\gamma_1-\gamma_2}.$$

For fixed, γ_1 and γ_2 such that $\gamma_1 + \gamma_2 \neq \mathbf{0}$, the inner sum is definitely 0. Therefore, we only have to care about the case when $\gamma_1 = -\gamma_2$. Using the Fourier expansions

$$(A_v(f))^\gamma = \sum_{\alpha\in\hat{G}^R}\hat{A}_\alpha f^\alpha, \qquad (A_\phi(h))^{-\gamma} = \sum_{\beta\in\hat{G}^S}\hat{B}_\beta h^\beta,$$

we obtain the expression

$$\sum_{\gamma\in\hat{G}\setminus\{\mathbf{0}\}}\mathrm{E}\Big[(A_v(f))^\gamma(A_\phi(h))^{-\gamma}\Big] = \sum_{\gamma\in\hat{G}\setminus\{\mathbf{0}\}}\sum_{\alpha\in\hat{G}^R}\sum_{\beta\in\hat{G}^S}\mathrm{E}\Big[\hat{A}_\alpha\hat{B}_\beta\,\mathrm{E}_{f,h}[f^\alpha h^\beta]\Big].$$

The inner expectation above is always zero unless $\alpha = \mathbf{0}$ and $\beta = \mathbf{0}$. Thanks to folding, we always have $\hat{B}_\beta = 0$ when $\beta = \mathbf{0}$, therefore all terms of the above form vanish.

The second double-sum needs an additional argument. Similarly to above, we need to expand $(A_v(\cdot))^\gamma$ and $(A_\phi(\cdot))^{-\gamma}$ in their Fourier series. This gives

$$\sum_{\gamma\in\hat{G}\setminus\{\mathbf{0}\}}\sum_{\alpha\in\hat{G}^R}\sum_{\beta\in\hat{G}^S}\mathrm{E}\Big[\hat{A}_\alpha\hat{B}_\beta\,\mathrm{E}[f^\alpha((f\circ\pi)^{-1}h^2e)^\beta]\Big] = \sum_{\gamma\in\hat{G}\setminus\{\mathbf{0}\}}\sum_{\beta\in\hat{G}^S}\mathrm{E}\Big[\hat{A}_{\pi_G(\beta)}\hat{B}_\beta\,\mathrm{E}[h^{2\beta}]\,\mathrm{E}[e^\beta]\Big],$$

where π_G is the "projected" character defined in Lemma 5. Consider an arbitrary term in the sum. Since h is selected uniformly and e is selected component-wise, the inner expectations can be written as

$$\mathrm{E}[h^{2\beta}]\,\mathrm{E}[e^\beta] = \mathrm{E}[h_0^{2\beta_0}]\,\mathrm{E}[e_0^{\beta_0}]\prod_{i=1}^{m}\mathrm{E}[h_i^{2\beta_i}]\,\mathrm{E}[e_i^{\beta_i}]$$

It is true for groups of odd order that $\beta \neq \mathbf{0}$ implies $2\beta \neq \mathbf{0}$, therefore $\mathrm{E}[h_0^{2\beta_0}]\,\mathrm{E}[e_0^{\beta_0}] = 0$ as soon as $\beta_0 \neq \mathbf{0}$. Similarly, $\mathrm{E}[h_i^{2\beta_i}] = 0$ as soon as $2\beta_i \neq \mathbf{0}$, i.e., as soon as $\beta_i(s) \notin \{0, 2^{\alpha_i}/2\}$ for some i

and s. Suppose that $\beta_i(s) \in \{0, 2^{\alpha_i}/2\}$ for all $s \in S$. If there exists an s_0 such that $\beta_i(s_0) = 2^{\alpha_i}/2$, $\mathrm{E}[(e_i(s_0))^{\beta(s_0)}] = 0$ by Lemma 6, and therefore

$$\mathrm{E}[e_1^{\beta}] = \prod_{s \in S} \mathrm{E}[(e_1(s))^{\beta(s)}] = 0.$$

The only remaining case to consider is that when $\beta(s) = \mathbf{0}$ for all s. However, $\hat{B}_\beta = 0$ as soon as $\beta(s) = 0$ for all s since the tables are folded, therefore the term vanishes also in this case. To conclude, all terms of the above form always vanish, for every β.

For the third double-sum, we proceed as above. We use the Fourier expansions

$$(A_\phi(h))^\gamma = \sum_{\beta_1 \in \hat{G}^S} \hat{B}_{\beta_1} h^{\beta_1},$$
$$(A_\phi((f \circ \pi)^{-1} h^2 e))^{-\gamma} = \sum_{\beta_2 \in \hat{G}^S} \hat{C}_{\beta_2} ((f \circ \pi)^{-1} h^2 e)^{\beta_2},$$

which gives the expression

$$\begin{aligned}
&\sum_{\gamma \in \hat{G} \setminus \{\mathbf{0}\}} \sum_{\beta_1 \in \hat{G}^S} \sum_{\beta_2 \in \hat{G}^S} \mathrm{E}\Big[\hat{B}_{\beta_1} \hat{C}_{\beta_2} \mathrm{E}_{f,h,e}[h^{\beta_1}((f \circ \pi)^{-1} h^2 e)^{\beta_2}]\Big] \\
&\quad = \sum_{\gamma \in \hat{G} \setminus \{\mathbf{0}\}} \sum_{\beta_1 \in \hat{G}^S} \sum_{\beta_2 \in \hat{G}^S} \mathrm{E}\Big[\hat{B}_{\beta_1} \hat{C}_{\beta_2} \mathrm{E}_f[f^{-\pi_G(\beta_2)}] \mathrm{E}_h[h^{\beta_1 + 2\beta_2}] \mathrm{E}_e[e^{\beta_2}]\Big] \\
&\quad = \sum_{\gamma \in \hat{G} \setminus \{\mathbf{0}\}} \sum_{\beta \in \hat{G}^S} \mathrm{E}\Big[\hat{B}_{-2\beta} \hat{C}_\beta \mathrm{E}_f[f^{-\pi_G(\beta)}] \mathrm{E}_e[e^{\beta}]\Big].
\end{aligned}$$

The inner expectation $\mathrm{E}[f^{-\pi_G(\beta)}]$ is zero as soon as $\pi_G(\beta) \neq \mathbf{0}$. Since the tables are folded, $\hat{B}_\beta = 0$ for all β such that $\pi_G(\beta) = \mathbf{0}$ by Corollary 2, therefore also the terms of the above form vanish.

Removing the terms that evaluate to zero, the acceptance probability can now be written

$$1 - \frac{1}{|G|^2} - \frac{1}{|G|^2} \sum_{\substack{\gamma_1, \gamma_2, \gamma_3 \\ \gamma_1 \neq \mathbf{0}, \gamma_2 \neq \mathbf{0}, \gamma_3 \neq \mathbf{0} \\ \gamma_1 + \gamma_2 + \gamma_3 = \mathbf{0}}} \mathrm{E}\Big[(A_v(f))^{\gamma_1} (A_\phi((f \circ \pi)^{-1} h^2 e))^{\gamma_2} (A_\phi(h))^{\gamma_3}\Big].$$

Suppose that the acceptance probability is at least $1 - |G|^{-2} + \varepsilon$ for some $\varepsilon > 0$. Then there exists γ_1, γ_2 and γ_3 such that

$$\Big|\mathrm{E}\Big[(A_v(f))^{\gamma_1} (A_\phi((f \circ \pi)^{-1} h^2 e))^{\gamma_2} (A_\phi(h))^{\gamma_3}\Big]\Big| \geq \varepsilon. \tag{7}$$

Now expand the expression inside $\mathrm{E}[\cdot]$ in a Fourier series for those γ_1, γ_2, γ_3:

$$(A_v(f))^{\gamma_1} = \sum_{\alpha \in \hat{G}^R} \hat{A}_\alpha f^\alpha,$$
$$(A_\phi(h))^{\gamma_3} = \sum_{\beta_1 \in \hat{G}^S} \hat{B}_{\beta_1} h^{\beta_1},$$

$$(A_\phi((f\circ\pi)^{-1}h^2e)))^{\gamma_2} = \sum_{\beta_2\in\hat{G}^S} \hat{C}_{\beta_2}((f\circ\pi)^{-1}h^2e))^{\beta_2}.$$

Substituting the Fourier expansions in (7) gives

$$\begin{aligned}
\varepsilon &\le \left|\mathrm{E}\left[\sum_{\alpha\in\hat{G}^R}\sum_{\beta_1\in\hat{G}^S}\sum_{\beta_2\in\hat{G}^S} \hat{A}_\alpha \hat{B}_{\beta_1}\hat{C}_{\beta_2}\,\mathrm{E}_f[f^{\alpha-\pi_G(\beta_2)}]\,\mathrm{E}_h[h^{\beta_1+2\beta_2}]\,\mathrm{E}_e[e^{\beta_2}]\right]\right| \\
&= \left|\mathrm{E}\left[\sum_{\beta_1\in\hat{G}^S}\sum_{\beta_2\in\hat{G}^S} \hat{A}_{\pi_G(\beta_2)} \hat{B}_{\beta_1}\hat{C}_{\beta_2}\,\mathrm{E}_h[h^{\beta_1+2\beta_2}]\,\mathrm{E}_e[e^{\beta_2}]\right]\right| \\
&= \left|\mathrm{E}\left[\sum_{\beta\in\hat{G}^S} \hat{A}_{\pi_G(\beta)} \hat{B}_{-2\beta}\hat{C}_{\beta}\,\mathrm{E}[e^{\beta}]\right]\right|.
\end{aligned}$$

Putting the pieces together, we have shown that

$$\varepsilon^2 \le \left|\mathrm{E}\left[(A_v(f))^{\gamma_1}(A_\phi((f\circ\pi)^{-1}h^2e))^{\gamma_2}(A_\phi(h))^{\gamma_3}\right]\right|^2 \le \mathrm{E}\left[\left|\sum_{\beta\in\hat{G}^S} \hat{A}_{\pi_G(\beta)} \hat{B}_{-2\beta}\hat{C}_{\beta}\,\mathrm{E}[e^{\beta}]\right|^2\right].$$

By Lemma 6, the only non-zero terms in the last sum above are those corresponding to $\beta\in\hat{G}^S$ such that for all $i\in\{1,2,\dots,s\}$ it holds that $\beta_i(s)\in\{0, 2^{\alpha_i}/4, 3\cdot 2^{\alpha_i}/4\}$ for all s. We now apply Cauchy-Schwartz to rewrite the above bound as

$$\varepsilon^2 \le \mathrm{E}\left[\left(\sum_\beta |\hat{A}_{\pi_G(\beta)}|^2|\hat{C}_\beta|^2|\mathrm{E}[e^\beta]|\right)\left(\sum_\beta |\hat{B}_{-2\beta}|^2|\mathrm{E}[e^\beta]|\right)\right],$$

where, again, all sums over β are over all $\beta\in\hat{G}^S$ such that for all $i\in\{1,2,\dots,m\}$ it holds that $\beta_i(s)\in\{0,2^{\alpha_i}/4, 3\cdot 2^{\alpha_i}/4\}$ for all s. We now want to bound the last factor above using Plancherel's equality. To do this, we gather terms containing the same $\hat{B}_{\beta'}$. Let $|\beta|$ denote the number of $s\in S$ such that $\beta(s)\neq\mathbf{0}$. Since the original sum is only over $\beta\in\hat{G}^S$ such that for all $i\in\{1,2,\dots,m\}$ it holds that $\beta_i(s)\in\{0,2^{\alpha_i}/4,3\cdot 2^{\alpha_i}/4\}$ for all s, there are, for each β', at most $2^{|\beta'|}=2^{|\beta|}$ different β with the property that $-2\beta=\beta'$. Therefore, the last sum can be bounded by

$$\sum_\beta |\hat{B}_{-2\beta}|^2|\,\mathrm{E}[e^\beta]| \le \sum_\beta |\hat{B}_{-2\beta}|^2 2^{-|\beta|} \le \sum_{\beta'\in\hat{G}^S}|\hat{B}_{\beta'}|^2 = \frac{1}{|G^S|}\sum_{h\in G^S}|(A_\phi(h))^{\gamma_3}|^2 = 1,$$

where the first inequality follows from Corollary 3 and the last equality is Plancherel's equality. To conclude, we have shown that

$$\varepsilon^2 \le \mathrm{E}\left[\sum_{\beta\in\hat{G}^S}|\hat{A}_{\pi_G(\beta)}|^2|\hat{C}_\beta|^2|\,\mathrm{E}[e^\beta]|\right] \le \mathrm{E}\left[\sum_{\beta\in\hat{G}^S}|\hat{A}_{\pi_G(\beta)}|^2|\hat{C}_\beta|^2 2^{-|\beta|}\right],$$

where the remaining expectation is over (v,ϕ,π). Consider the following probabilistic procedure to construct an assignment given a randomly selected (v,ϕ,π): Select α and β at random according to the distribution given by $|\hat{A}_\alpha|^2$ and $|\hat{C}_\beta|^2$. This is a well-defined procedure since

$$\sum_{\alpha\in\hat{G}^R}|\hat{A}_\alpha|^2 = \frac{1}{|G^R|}\sum_{f\in G^R}|(A_v(f))^{\gamma_1}|^2 = 1,$$

$$\sum_{\beta\in\hat{G}^S} |\hat{C}_\beta|^2 = \frac{1}{|G^S|}\sum_{h\in G^S} |(A_\phi(h))^{\gamma_2}|^2 = 1,$$

by Plancherel's equality. Then uniformly select an s such that $\beta(s) \neq \mathbf{0}$ and an r such that $\alpha(r) \neq \mathbf{0}$. Since the tables in the PCP are folded, such r and s always exist. It is easy to see that if $\alpha = \pi_G(\beta)$, there is for every r such that $\alpha(r) \neq \mathbf{0}$ at least one $s \in \beta$ such that $r = \pi(s)$. Hence the above strategy selects r and s such that $r = \pi(s)$ with probability at least

$$\sum_{\beta\in\hat{G}^S} |\hat{A}_{\pi_G(\beta)}|^2|\hat{C}_\beta|^2|\beta|^{-1} \geq \sum_{\beta\in\hat{G}^S} |\hat{A}_{\pi_G(\beta)}|^2|\hat{C}_\beta|^2 2^{-|\beta|}$$

and thus a randomly selected (v,ϕ,π) is satisfied with probability at least

$$\mathrm{E}\biggl[\sum_{\beta\in\hat{G}^S} |\hat{A}_{\pi_G(\beta)}|^2|\hat{C}_\beta|^2 2^{-|\beta|}\biggr] \geq \varepsilon^2 > |R|^{-\mu}.$$

It follows that there exists an assignment that satisfies more than a fraction $|R|^{-\mu}$ of the triples in Ψ, i.e., case 2 in Theorem 5 cannot hold. ∎

We are now in a position to prove Theorem 4 for G isomorphic to a product of the form $G_0 \times \mathbb{Z}_{2^{\alpha_1}} \times \mathbb{Z}_{2^{\alpha_2}} \times \cdots \times \mathbb{Z}_{2^{\alpha_m}}$ where G_0 is a finite Abelian group of odd order and $\alpha_i > 1$ for all i. Given an instance Ψ of the problem described in Theorem 5, construct an instance of Max E3-NAE-Sat(G) as follows: For every $(v,\phi,\pi) \in \Psi$, include every possible not-all-equals constraint that can be checked by the verifier in Fig. 6. The weight of each constraint is proportional to the probability that it is chosen by the verifier. The size of this instance is polynomial in $|\Psi|$ and since all weights are rational, it can be converted into an unweighted instance by simply repeating constraints an appropriate number of times. If we can distinguish the case when the so obtained instance of Max E3-NAE-Sat(G) is satisfiable from the case when at most a fraction $(1 - |G|^{-2} + \varepsilon)$ is satisfiable we can, by Lemmas 7 and 8 distinguish between case 1 and case 2 in Theorem 5.

To complete the proof of Theorem 4 we use the following simple induction argument:

Lemma 9. *Suppose that, for every constant $\varepsilon > 0$, it is **NP**-hard to distinguish instances of Max E3-NAE-Sat(G) that are satisfiable from instances where at most a fraction $(1 - |G|^{-2} + \varepsilon)$ of the constraints are simultaneously satisfiable for some finite Abelian group G. Then, for every constant $\varepsilon > 0$, it is **NP**-hard to distinguish instances of Max E3-NAE-Sat($\mathbb{Z}_2 \times G$) that are satisfiable from instances where at most a fraction $(1-(2|G|)^{-2}+\varepsilon)$ of the constraints are simultaneously satisfiable.*

Proof. Given an arbitrary clause $\mathrm{NAE}(g_i x_i, g_j x_j, g_k x_k)$ from the Max E3-NAE-Sat(G) instance construct four clauses

$$\begin{aligned}
&\mathrm{NAE}((1,g_i)y_i, (1,g_j)y_j, (1,g_k)y_k),\\
&\mathrm{NAE}((-1,g_j)y_j, (1,g_j)y_j, (1,g_k)y_k),\\
&\mathrm{NAE}((1,g_i)y_i, (-1,g_j)y_j, (1,g_k)y_k),\\
&\mathrm{NAE}((1,g_i)y_i, (1,g_j)y_j, (-1,g_k)y_k)
\end{aligned}$$

in the Max E3-NAE-Sat($\mathbb{Z}_2 \times G$) instance for the group $\mathbb{Z}_2 \times G$. Above we use multiplication as the group operator and represent $\mathbb{Z}_2$ as $\{-1, 1\}$.

Given a solution to the Max E3-NAE-Sat($\mathbb{Z}_2 \times G$) instance, construct a solution to the Max E3-NAE-Sat(G) in the obvious way. If x, y, and z are not all equal, all four clauses will be satisfied—if they are all equal, the last three clauses will be satisfied. Therefore, it is hard to distinguish the case when all constraints can be satisfied from the case when at most a fraction

$$\frac{4(1-d^{-2}+4\varepsilon)+3(d^{-2}-4\varepsilon)}{4}=\frac{4-d^{-2}+4\varepsilon}{4}=1-(2d)^{-2}+\varepsilon$$

of the constraints can be satisfied. ∎

5 The status of Max E2-Sat(d)

Although we were not able to completely resolve the status of Max E2-Sat(d) as far as approximability is concerned, we have obtained some evidence regarding its hardness. In this paper, we resolved the status of the easier Max Co-BIJ(d) problem—it is *not* approximation resistant according to Theorem 1—and the harder Max E3-NAE-Sat($\mathbb{Z}_d$) problem—it *is* approximation resistant according to Theorem 4.

For the Max E2-Sat(d) problem itself, we present some hardness results below. We first prove a result for domain size 3 and then prove a result for general domains.

Lemma 10. *For every constant $\varepsilon > 0$, the predicate $(x \neq a) \vee (y = b)$ over domains of size 3 is hard to approximate within $(23/24 + \varepsilon)$ with perfect completeness.*

Proof. Consider the following 2P1R interactive proof system for 3SAT: The first prover is given a variable and returns an assignment to that variable, the second prover is given a clause and returns an index of a literal that makes the clause satisfied. The verifier selects a clause at random, then a variable in the clause at random, sends the variable to P_1, the clause to P_2 and accepts unless P_2 returns the index of the variable sent to P_1 and P_1 returned an assignment that does not satisfy the literal. It is known that there are satisfiable instances of 3SAT such that it is **NP**-hard to satisfy more than $7/8+\varepsilon$ of the clauses, for every constant $\varepsilon > 0$ [11]. When the above protocol is applied to such an instance of 3SAT, the test has perfect completeness and soundness $(1-1/24+\varepsilon)$. To obtain the hardness for the claimed constraint satisfaction problem, we just use the following reduction: x specifies the name of a clause, y specifies a variable in this clause, a specifies the location of y in x (encoded as 0,1 or 2), and b specifies a Boolean assignment to y (encoded over 0,1,2, where 2 is meaningless). ∎

Theorem 6. *For every constant $\varepsilon > 0$, it is* **NP***-hard to approximate Max E2-Sat(3) within $47/48+\varepsilon$ with perfect completeness.*

Proof. Follows from Lemma 10 since $(x \neq a) \vee (y = b)$ can be written as two E2-Sat(3) clauses. ∎

Theorem 7. *For every $d \geq 3$ and every constant $\varepsilon > 0$, Max E2-Sat(d) is hard to approximate within a factor of $(1 - d^{-4} + \varepsilon)$ with perfect completeness.*

Proof. We reduce Max E3-Sat(d), which is hard to approximate within $(1 - d^{-3} + d\varepsilon)$ with perfect completeness by Theorem 4 applied to $G = \mathbb{Z}_d$ combined with the discussion in § 2, to Max E2-Sat(d). A constraint $\mathrm{SAT}(x, y, z)$, which requires that at least one of x, y, z does not equal 0, is replaced with the constraints $\mathrm{SAT}(x, t)$, $\mathrm{SAT}(x, t + 1)$, $\mathrm{SAT}(x, t + 2)$, $\ldots$, $\mathrm{SAT}(x, t + d - 3)$, $\mathrm{SAT}(y, t + d - 2)$, $\mathrm{SAT}(z, t + d - 1)$, where t is an auxiliary variable specific to this constraint and the additions are done modulo d. If all d 2SAT clauses are satisfied, the 3SAT clause has to be satisfied; if the 3SAT clause is not satisfied we can satisfy $d - 1$ of the 2SAT clauses. Therefore it is hard to distinguish the case when all the constraints are satisfied from the case when a fraction $\frac{1}{d}(d(1-d^{-3}+d\varepsilon)+(d-1)(d^{-3}-d\varepsilon)) = (1-d^{-4}+\varepsilon)$ of the constraints are satisfied. ∎

Theorem 8. *Max E2-Sat(d) is hard to approximate within a factor $1-\Omega(d^{-2})$ with non-perfect completeness. It is also hard to approximate within a factor $1-\Omega(d^{-3})$ with perfect completeness for all $d \geq 3$.*

Proof. We reduce d-CUT, which is hard to approximate within $1-1/34d+\varepsilon$ with non-perfect completeness [12], to Max E2-Sat(d). In the d-CUT problem, the input is a set of clauses of the form $\mathrm{CUT}(x,y)$ and the objective is to partition the variables into d parts in such a way that the number of clauses $\mathrm{CUT}(x,y)$ where $x \neq y$ is maximized. For the reduction, a clause $\mathrm{CUT}(x,y)$ is replaced with the clauses $\mathrm{SAT}(x+i, y+i)$ for all i from 0 to $d-1$. A d-CUT instance with n constraints corresponds to a 2SAT(d) instance with dn constraints and an assignment satisfying all but k 2SAT(d) constraints satisfies all but k d-CUT constraints.

The hardness result with perfect completeness follows using the above reduction together with the result of Lemma 11 below which shows a factor $1-\Omega(1/d^2)$ inapproximability result for Max d-CUT with perfect completeness, or in other words Max d-CUT on d-colorable graphs. Combining with the above gadget that reduces d-CUT to 2SAT(d), we get the claimed $1-\Omega(1/d^3)$ hardness for satisfiable instances of Max E2-Sat(d) for all $d \geq 3$. ∎

Lemma 11. *There is a constant $\gamma > 0$ such that for all $d \geq 3$, given as input a d-colorable graph, it is* **NP***-hard to find a d-cut that cuts a fraction $1-\gamma/d^2$ of the edges.*

Proof. The reduction is from Max 3-CUT on 3-colorable graphs (i.e., Max 3-CUT with perfect completeness), which is known to be hard to approximate within a factor $(1-\delta)$ for some absolute constant $\delta > 0$ [15, Theorem 3.3]. This already gives the result for $d = 3$, so assume $d \geq 4$. Given a graph $G = (V, E)$ construct a new *weighted* graph H as follows. H will contain G as an induced subgraph. It will further contain $(d-3)$ new vertices $u_1, u_2, \ldots, u_{d-3}$, and each u_i is connected to each $v \in V$ by an edge of weight $d(v)$, the degree of v in G. Also, between u_i and u_j for $1 \leq i < j \leq d-3$, we add an edge of weight $2m$ where m is the number of edges in G. (We remark that it is possible to be more frugal in the size of the weights we assign, but since we are not optimizing for the value of γ in the statement of the lemma, we chose large enough weights to make the proof very simple.)

Clearly, H is d-colorable if G is 3-colorable as one can give $(d-3)$ new colors to the u_i's. Furthermore, given a d-coloring of H that fails to cut a set of edges of total weight W, we can use it to find a 3-coloring of G that fails to cut at most W edges. Indeed, given a d-coloring of H, we can first modify it so that the u_i's receive distinct colors without increasing the weight of the uncut edges. We can do so since if, say, u_1 and u_2 receive the same color, then we can recolor u_2 by a color that is not used by any of the u_i's; under the new coloring the edge (u_1, u_2) of weight $2m$ is cut, and the worst that could happen is that all edges (u_2, v) become uncut and these have a total weight of only $2m$. Once the u_i's all get distinct colors, in the next step, if some $v \in V$ and u_i have the same color, we recolor v using one of the 3 colors not used by $u_1, u_2, \ldots, u_{d-3}$, until no edge of the form (v, u_i) is uncut. These steps do not increase the weight of the uncut edges since the weight of an edge (v, u_i) is at least the total weight of all edges incident upon v in G.

We now have a d-coloring of H in which the vertices in V receive three colors distinct from the $(d-3)$ colors used to color the u_i's and thus the only uncut edges are those of G. Since the weight of uncut edges was originally W and could have only decreased in the process of recoloring H, it follows that there is a 3-coloring of G that leaves at most W uncut (i.e., miscolored) edges.

The total weight of edges of H, say M, is at most d^2m. By the above discussion, if the maximum 3-cut of G cuts at most $(1-\delta)m$ edges, then the maximum d-cut of H cuts edges of total weight at most $M - \delta m \leq (1-\delta/d^2)M$.

Hence, it is **NP**-hard to approximate Max d-cut on *weighted* graphs within a factor of $(1 - \Omega(1/d^2))$ even with perfect completeness. This is almost the result we want except that we only showed hardness for weighted graphs. However, one can deduce inapproximability within the same factor also for unweighted graphs by appealing to a general procedure for getting rid of weights due to Crescenzi, Silvestri, and Trevisan [5]. ∎

The result of Theorem 7 is not entirely subsumed by the result of Theorem 8 for satisfiable instances, since the constant in front of $1/d^3$ implies that the result of Theorem 7 will actually be stronger for small values of d. An interesting question is whether a factor $(1 - \Omega(d^{-2}))$ hardness can be shown for satisfiable instances of Max E2-Sat(d).

Conjecture. Although we did not completely resolve the status of Max E2-Sat(d), we conjecture at this point that the problem is *not* approximation resistant.

Acknowledgments

We would like to thank Subhash Khot for sending us a copy of [13]; his algorithm for Max BIJ(d) directly inspired our algorithm for Max Co-BIJ(d). We also thank the anonymous referees for their good work and many constructive comments that helped us improve the presentation of our results.

References

1. Gunnar Andersson. An approximation algorithm for max p-section. In *Proceedings of the 16th Annual Symposium on Theoretical Aspects of Computer Science*, volume 1563 of *Lecture Notes in Computer Science*, pages 237–247. Trier, 4–6 March 1999.
2. Gunnar Andersson. *Some New Randomized Approximation Algorithms*. Doctoral dissertation, Department of Numerical Analysis and Computer Science, Royal Institute of Technology, May 2000.
3. Gunnar Andersson, Lars Engebretsen, and Johan Håstad. A new way of using semidefinite programming with applications to linear equations mod p. *Journal of Algorithms*, 39(2):162–204, May 2001.
4. Sanjeev Arora, Carsten Lund, Rajeev Motwani, Madhu Sudan, and Márió Szegedy. Proof verification and the hardness of approximation problems. *Journal of the ACM*, 45(3):501–555, May 1998.
5. Pierluigi Crescenzi, Riccardo Silvestri, and Luca Trevisan. On weighted vs unweighted versions of combinatorial optimization problems. *Information and Computation*, 167(1):10-26, May 2001.
6. Irit Dinur, Venkatesan Guruswami, Subhash Khot, and Oded Regev. A new multilayered PCP and the hardness of hypergraph vertex cover. In *Proceedings of the 35th Annual ACM Symposium on Theory of Computing*, pages 595–601, June 2003.
7. Uriel Feige. A threshold of $\ln n$ for approximating set cover. *Journal of the ACM*, 45(4):634–652, July 1998.
8. Alan Frieze and Mark Jerrum. Improved approximation algorithms for MAX k-CUT and MAX BISECTION. *Algorithmica*, 18:67–81, 1997.
9. Michel X. Goemans and David P. Williamson. Improved approximation algorithms for maximum cut and satisfiability problems using semidefinite programming. *Journal of the ACM*, 42(6):1115–1145, November 1995.
10. Michel X. Goemans and David P. Williamson. Approximation algorithms for Max-3-Cut and other problems via complex semidefinite programming. In *Proceedings of the 33rd Annual ACM Symposium on Theory of Computing*, pages 443–452. Hersonissos, Crete, Grece, 6–8 July 2001.

11. Johan Håstad. Some optimal inapproximability results. *Journal of the ACM*, 48(4):798–859, July 2001.
12. Viggo Kann, Sanjeev Khanna, Jens Lagergren, and Alessandro Panconesi. On the hardness of approximating Max k-Cut and its dual. *Chicago Journal of Theoretical Computer Science*, 1997(2), June 1997.
13. Subhash Khot. On the power of unique 2-prover 1-round games. In *Proceedings of the 34th Annual ACM Symposium on Theory of Computing*, pages 767–775. Montréal, Québec, Canada, 19–21 May 2002.
14. Subhash Khot. Hardness results for coloring 3-colorable 3-uniform hypergraphs. In *Proceedings of the 43rd IEEE Symposium on Foundations of Computer Science*, pages 23–32. Vancouver, Canada, 16–19 November 2002.
15. Erez Petrank. The hardness of approximation: Gap location. *Computational Complexity*, 4(2):133–157, 1994.
16. Ran Raz. A parallel repetition theorem. *SIAM Journal on Computing*, 27(3):763–803, June 1998.
17. John S. Rose. *A Course on Group Theory.* Cambridge University Press, Cambridge, 1978.
18. Audrey Terras. *Fourier Analysis on Finite Groups and Applications*, volume 43 of *London Mathematical Society student texts.* Cambridge University Press, Cambridge, 1999.
19. Uri Zwick. Approximation algorithms for constraint satisfaction programs involving at most three variables per constraint. In *Proceedings of the Ninth Annual ACM-SIAM Symposium on Discrete Algorithms*, pages 201–210. San Francisco, California, 25–27 January 1998.
20. Uri Zwick. Outward rotations: a tool for rounding solutions of semidefinite programming relaxations, with applications to MAX CUT and other problems. In *Proceedings of the Thirty-First Annual ACM Symposium on Theory of Computing*, pages 679–687. Atlanta, Georgia, 1–4 May 1999.

A Analysis of the algorithm for Max Co-BIJ(d)

Lemma 12. *Let r be a $2d$-dimensional normal variable with mean zero and identity covariance matrix. Let $B = \{r : |r| \leq 1/Kd\}$. Then,*

$$\int_B \left(1 - \frac{1}{d} + \frac{K^2 r_1^2}{d}\right) dP(r)$$
$$= 1 - \frac{1}{d} + \frac{K^2}{d} - e^{-1/2(Kd)^2} \left(\left(1 - \frac{1}{d}\right) \sum_{k=0}^{d-1} \frac{1}{k! 2^k (Kd)^{2k}} + \frac{K^2}{d} \sum_{k=0}^{d} \frac{1}{k! 2^k (Kd)^{2k}} \right).$$

Proof. We will show that

$$\int_B dP(r) = 1 - e^{-1/2(Kd)^2} \sum_{k=0}^{d-1} \frac{1}{k! 2^k (Kd)^{2k}},$$
$$\int_B r_1^2 \, dP(r) = 1 - e^{-1/2(Kd)^2} \sum_{k=0}^{d} \frac{1}{k! 2^k (Kd)^{2k}}.$$

Let us start by computing the first integral. The probability distribution for a $2d$-dimensional normally distributed random variable with zero mean and identity covariance matrix is $P(r) = (2\pi)^{-d} e^{-|r|^2/2}$, therefore

$$\int_B dP(r) = \frac{1}{(2\pi)^d} \int_0^{1/(Kd)} e^{-\rho^2/2} \int_{r:|r|=\rho} dS(r) \, d\rho.$$

The volume of a $2d$-dimensional hypersphere with radius ρ is $\pi^d \rho^{2d}/d!$ and the area of this sphere is $2d/\rho$ times the volume; thus $\int_{r:|r|=\rho} dS(r) = 2d\pi^d \rho^{2d-1}/d!$ and

$$\int_B dP(r) = \frac{1}{2^{d-1}(d-1)!} \int_0^{1/Kd} \rho^{2d-1} e^{-\rho^2/2} \, d\rho.$$

With the substitution $\rho^2/2 = t$, $\rho\, d\rho = dt$,

$$\int_B dP(r) = \frac{1}{(d-1)!} \int_0^{1/(2K^2d^2)} t^{d-1} e^{-t}\, dt.$$

Using the identity

$$\int t^d e^{-t}\, dt = -e^{-t} \sum_{k=0}^{d} \frac{d!}{k!} t^k,$$

the latter integral can be easily evaluated:

$$\frac{1}{(d-1)!} \int_0^{1/(2K^2d^2)} t^{d-1} e^{-t}\, dt = 1 - e^{-1/2(Kd)^2} \sum_{k=0}^{d-1} \frac{1}{k! 2^k (Kd)^{2k}}\,.$$

The second integral can be computed with similar techniques. Since B is spherically symmetric,

$$\int_B r_1^2\, dP(r) = \frac{1}{2d} \int_B |r|^2\, dP(r).$$

Using the fact that the probability distribution for a $2d$-dimensional normally distributed random variable with zero mean and identity covariance matrix is $P(r) = (2\pi)^{-d} e^{-|r|^2/2}$, the integral of interest is

$$\begin{aligned}
\int_B |r|^2\, dP(r) &= \frac{1}{(2\pi)^d} \int_B |r|^2 e^{-|r|^2/2}\, dV(r) \\
&= \frac{1}{(2\pi)^d} \int_0^{1/(Kd)} \rho^2 e^{-\rho^2/2} \int_{r:|r|=\rho} dS(r)\, d\rho \\
&= \frac{1}{d!} \int_0^{1/2(Kd)^2} t^d e^{-t}\, dt \\
&= \left[-e^{-t} \sum_{k=0}^{d} \frac{t^k}{k!} \right]_0^{t=1/2(Kd)^2}
\end{aligned}$$

To conclude,

$$\int_B r_1^2\, dP(r) = 1 - e^{-1/2(Kd)^2} \sum_{k=0}^{d} \frac{1}{k! 2^k (Kd)^{2k}}\,. \qquad \blacksquare$$

Lemma 13. *Let*

$$S_n = \sum_{k=0}^{n} \frac{1}{k! 2^k (Kd)^{2k}}$$

With the parameter choice $K^2 = d^{-3}/13$,

$$\frac{K^2}{d} - e^{-1/2(Kd)^2} \left(\left(1 - \frac{1}{d}\right) S_{d-1} + \frac{K^2}{d} S_d \right) \geq 0.07 d^{-4}.$$

for all integers $d \geq 2$.

Proof. With the choice $K^2 = d^{-3}/2\alpha$, $e^{-1/2(Kd)^2} = e^{-\alpha d}$ and $S_n = \sum_{k=0}^{d} \frac{(\alpha d)^k}{k!}$. Let $a_k = (\alpha d)^k/k!$. Note that all a_k are positive and that $a_{k+1}/a_k = \alpha d/(k+1) \geq \alpha$ for all $k+1 \leq d$. Therefore, provided that $\alpha \geq 2$, $S_{d-1} \leq 2a_{d-1} \leq a_d$ and $S_d \leq 2a_d$, which implies that

$$\left(1 - \frac{1}{d}\right) S_{d-1} + \frac{K^2}{d} S_d \leq \left(1 - \frac{1}{d}\right) \frac{(\alpha d)^d}{d!} + \frac{1}{\alpha d^4} \frac{(\alpha d)^d}{d!} \leq \frac{(\alpha d)^d}{d!} \leq \frac{(\alpha e)^d}{\sqrt{2\pi d}}$$

where the last inequality follows since $d! \geq d^d e^{-d} \sqrt{2\pi d}$ by Stirling's formula. Therefore

$$\frac{K^2}{d} - e^{-1/2(Kd)^2} \left(\left(1 - \frac{1}{d}\right) S_{n-1} + S_n \right) \geq \frac{K^2}{d} - e^{-\alpha d} \frac{(\alpha e)^d}{\sqrt{2\pi d}} = \beta d^{-4}$$

where

$$\beta = \frac{1}{2\alpha} - (2\pi)^{-1/2} \exp((1 - \alpha + \ln \alpha) d + 3.5 \ln d).$$

All that remains is to select a good α, more precisely, an α such that the exponential function is decreasing in d. A simple calculation shows that $\alpha = 6.5$ suffices for all integers $d \geq 2$ and gives a lower bound of $\beta \geq 0.07$. We remark that it is possible to optimize the value of β for larger d and get $\beta \approx 1/4$ for d large enough. ∎