



UNIVERSITY OF LEEDS

This is a repository copy of *Parameterized Approximability for Modular Linear Equations*.

White Rose Research Online URL for this paper:

<https://eprints.whiterose.ac.uk/228989/>

Version: Accepted Version

Proceedings Paper:

Dawbrowski, K. K., Jonsson, P., Ordyniak, S. orcid.org/0000-0003-1935-651X et al. (2 more authors) (Accepted: 2025) *Parameterized Approximability for Modular Linear Equations*. In: *Leibniz International Proceedings in Informatics. European Symposium on Algorithms (ESA 2025)*, 15-17 Sep 2025, Warsaw, Poland. Schloss Dagstuhl -- Leibniz-Zentrum fuer Informatik (In Press)

This is an author produced version of a proceedings paper accepted for publication in *Leibniz International Proceedings in Informatics*, made available under the terms of the Creative Commons Attribution License (CC-BY), which permits unrestricted use, distribution and reproduction in any medium, provided the original work is properly cited.

Reuse

This article is distributed under the terms of the Creative Commons Attribution (CC BY) licence. This licence allows you to distribute, remix, tweak, and build upon the work, even commercially, as long as you credit the authors for the original work. More information and the full terms of the licence here:

<https://creativecommons.org/licenses/>

Takedown

If you consider content in White Rose Research Online to be in breach of UK law, please notify us by emailing eprints@whiterose.ac.uk including the URL of the record and the reason for the withdrawal request.





eprints@whiterose.ac.uk
<https://eprints.whiterose.ac.uk/>

Parameterized Approximability for Modular Linear Equations

Konrad K. Dabrowski  

Newcastle University, UK

Peter Jonsson  

Linköping University, Sweden

Sebastian Ordyniak  

University of Leeds, Leeds, United Kingdom

George Osipov  

Linköping University, Sweden, and University of Oxford, UK

Magnus Wahlström  

Royal Holloway, University of London, UK

Abstract

We consider the $\text{MIN-}r\text{-LIN}(\mathbb{Z}_m)$ problem: given a system S of length- r linear equations modulo m , find $Z \subseteq S$ of minimum cardinality such that $S - Z$ is satisfiable. The problem is NP-hard and UGC-hard to approximate in polynomial time within any constant factor even when $r = m = 2$. We focus on parameterized approximation with solution size as the parameter. Dabrowski, Jonsson, Ordyniak, Osipov and Wahlström [SODA-2023] showed that $\text{MIN-2-LIN}(\mathbb{Z}_m)$ is in FPT if m is prime (i.e. \mathbb{Z}_m is a field), and it is W[1]-hard if m is not a prime power. We show that $\text{MIN-2-LIN}(\mathbb{Z}_{p^n})$ is FPT-approximable within a factor of 2 for every prime p and integer $n \geq 2$. This implies that $\text{MIN-2-LIN}(\mathbb{Z}_m)$, $m \in \mathbb{Z}^+$, is FPT-approximable within a factor of $2\omega(m)$ where $\omega(m)$ counts the number of distinct prime divisors of m . The high-level idea behind the algorithm is to solve tighter and tighter relaxations of the problem, decreasing the set of possible values for the variables at each step. When working over \mathbb{Z}_{p^n} and viewing the values in base- p , one can roughly think of a relaxation as fixing the number of trailing zeros and the least significant nonzero digits of the values assigned to the variables. To solve the relaxed problem, we construct a certain graph where solutions can be identified with a particular collection of cuts. The relaxation may hide obstructions that will only become visible in the next iteration of the algorithm, which makes it difficult to find optimal solutions. To deal with this, we use a strategy based on shadow removal [Marx & Razgon, STOC-2011] to compute solutions that (1) cost at most twice as much as the optimum and (2) allow us to reduce the set of values for all variables simultaneously. We complement the algorithmic result with two lower bounds, ruling out constant-factor FPT-approximation for $\text{MIN-3-LIN}(R)$ over any nontrivial ring R and for $\text{MIN-2-LIN}(R)$ over some finite commutative rings R .

2012 ACM Subject Classification Theory of computation \rightarrow Parameterized complexity and exact algorithms

Keywords and phrases parameterized complexity, approximation algorithms, linear equations

Digital Object Identifier 10.4230/LIPIcs.ESA.2025.86

Funding *Peter Jonsson*: Supported by the Swedish Research Council (VR) under grant 2021-04371. *Sebastian Ordyniak*: Supported by the Engineering and Physical Sciences Research Council (EPSRC), project EP/V00252X/1).

George Osipov: Supported by the Swedish Research Council (VR) under grant 2024-00274.

1 Introduction

Systems of linear equations are ubiquitous in computer science and mathematics [16] and methods like Gaussian elimination can efficiently solve linear systems over various rings.



© Konrad K. Dabrowski, Peter Jonsson, Sebastian Ordyniak, George Osipov, Magnus Wahlström; licensed under Creative Commons License CC-BY 4.0

33rd Annual European Symposium on Algorithms (ESA 2025).

Editors: Anne Benoit, Haim Kaplan, Sebastian Wild, and Grzegorz Herman; Article No. 86; pp. 86:1–86:14



Leibniz International Proceedings in Informatics

LIPICs Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany

Equations and congruences over the ring of integers modulo m (\mathbb{Z}_m) are of central importance in number theory, but also have many applications in computer science, including complexity theory, coding theory, cryptography, hash functions and pseudorandom generators, see e.g. [3, 9, 11, 33]. Linear equations over modular rings can be solved in polynomial time, but the methods are not suited to dealing with inconsistent systems of equations. We consider the $\text{MIN-}r\text{-LIN}(R)$ problem which asks to find an assignment to a system of linear equation over the ring R that violates the minimum number of equations, and where each equations contains at most r distinct variables. This problem is NP-hard even when $r = 2$ and R is the simplest nontrivial ring \mathbb{Z}_2 [23]. We note that $\text{MIN-}r\text{-LIN}(R)$ for $r \in \mathbb{N}$ and finite ring R is a special case of $\text{MINCSP}(\Gamma)$ for a finite constraint language Γ . This, and the more general VALUED CSP, have been widely studied from many different perspectives, e.g. [4, 6, 22, 23, 31, 32].

Two common ways of coping with NP-hardness are approximation and parameterized algorithms, but neither of them seems sufficient in isolation to deal with $\text{MIN-}r\text{-LIN}(\mathbb{Z}_m)$. Even MIN-2-LIN over finite fields such as \mathbb{Z}_2 is conjectured to be NP-hard to approximate within any constant factor under the Unique Games Conjecture (UGC) [20]: see Definition 3 in [21] and the discussion that follows. The natural parameter for $\text{MIN-}r\text{-LIN}(\mathbb{Z}_m)$ is the cost of the optimal solution (i.e. the number of equations not satisfied by it), which we denote by k . Under this parameterization, $\text{MIN-2-LIN}(\mathbb{Z}_m)$ is fixed-parameter tractable when m is a prime, i.e. \mathbb{Z}_m is a field, but W[1]-hard when m is not a prime power. Moreover, the problem MIN-3-LIN is W[1]-hard for every nontrivial ring [8]. This motivates us to study *parameterized approximation* algorithms [14, 27]. This approach has received rapidly increasing interest (see, for instance, [13, 17, 18, 25, 26, 30]). Let $c \geq 1$ be a constant. A *factor- c FPT-approximation algorithm* takes an instance (I, k) , runs in $O^*(f(k))$ ¹ time for an arbitrary computable function f , either returns that there is no solution of size at most k or returns that there is a solution of size at most $c \cdot k$. Thus, there is more time to compute the solution (compared to polynomial-time approximation) and the algorithm may output an oversized solution (unlike an exact FPT algorithm).² Our main result is the following. Let $\omega(m)$ be the number of distinct prime factors of m .

► **Theorem 1.** *For every $m \in \mathbb{Z}_+$, $\text{MIN-2-LIN}(\mathbb{Z}_m)$ is FPT-approximable within $2\omega(m)$.*

We complement the result with two lower bounds. First, we show that allowing three or more variables per equation leads to W[1]-hardness of constant-factor approximation.

► **Theorem 2.** *$\text{MIN-3-LIN}(R)$ over every nontrivial ring R is W[1]-hard to approximate within any constant factor.*

This result strengthens two previously known hardness results: (i) $\text{MIN-3-LIN}(R)$ is W[1]-hard [8] and (ii) $\text{MIN-3-LIN}(R)$ is NP-hard to approximate within any constant (which can easily be derived from [19]). While we focus on rings of the form \mathbb{Z}_m , the result of Theorem 1 begs the questions whether $\text{MIN-2-LIN}(R)$ is FPT-approximable within a constant factor for every finite commutative ring R . We answer this question in the negative.

¹ The notation $O^*(\cdot)$ hides polynomial factors in the input size.

² A decision c -approximation procedure for $\text{MIN-2-LIN}(\mathbb{Z}_m)$ can be turned into an algorithm that returns a c -approximate solution using self-reducibility: if (S, k) is a yes-instance, then there exists a subset $S' \subseteq S$, $|S'| \leq c$, such that $(S - S', k - 1)$ is a yes-instance; moreover, such S' can be found by iterating over all subsets of size at most c , which incurs a polynomial overhead on the running time of the algorithm.

► **Theorem 3** (See Theorem 17 for a more detailed statement.). *There exist finite commutative rings R such that $\text{MIN-2-LIN}(R)$ is $\text{W}[1]$ -hard to approximate within any constant factor.*

Theorems 5.2 and 6.2 in [8] leave open the question of whether $\text{MIN-2-LIN}(\mathbb{Z}_{p^n})$ is FPT or $\text{W}[1]$ -hard for a prime p and $n \geq 2$. The answer is unknown even for the smallest such ring – \mathbb{Z}_4 . While our result implies that $\text{MIN-2-LIN}(\mathbb{Z}_{p^n})$ is FPT-approximable within a factor of 2, its exact parameterized complexity remains an intriguing open problem.

Full proofs for the results marked with a \star can be found in the full version of the paper. Another version of this paper is available on arXiv [7]; it considers a broader class of rings, but gives worse approximation factors.

2 Preliminaries

For the basics of graph theory and parameterized complexity, we refer to [10, 12, 15, 29].

An expression $c_1 \cdot x_1 + \dots + c_r \cdot x_r = c$ is a (*linear*) *equation over R* if $c_1, \dots, c_r, c \in R$ and x_1, \dots, x_r are variables with domain R . This equation is *homogeneous* if $c = 0$. Let S be a set (or equivalently a system) of equations over R . Let $V(S)$ denote the variables in S , and we say that S is *consistent* if there is an assignment $\varphi : V(S) \rightarrow R$ satisfying all equations in S . An instance of the computational problem $r\text{-LIN}(R)$ is a system S of equations in at most r variables over R , and the question is whether S is consistent. Linear equation systems over \mathbb{Z}_m are solvable in polynomial time and the well-known procedure is outlined, for instance, in [1, p. 473]. We now define the computational problem when we allow some equations in an instance to be soft (i.e. deletable at unit cost) and crisp (i.e. undeletable).

$\text{MIN-}r\text{-LIN}(R)$	
INSTANCE:	A (multi)set S of equations over R with at most r variables per equation, a subset $S^\infty \subseteq S$ of crisp equations and an integer k .
PARAMETER:	k .
QUESTION:	Is there a set $Z \subseteq S \setminus S^\infty$ such that $S - Z$ is consistent and $ Z \leq k$?

We use crisp equations for convenience since they can be modelled by $k + 1$ copies of the same soft equation. For an assignment $\alpha : V(S) \rightarrow R$, let $\text{cost}_S(\alpha)$ be ∞ if α does not satisfy a crisp equation and the number of unsatisfied soft equations otherwise. We drop the subscript S when it is clear from context. We write $\text{mincost}(S)$ to denote the minimum cost of an assignment to S .

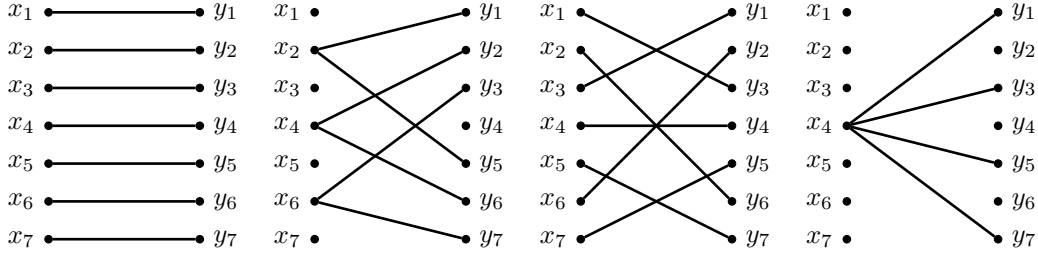
3 FPT-Approximation Algorithm

3.1 Algorithm Summary

Let $p_1^{n_1} \dots p_\ell^{n_\ell}$ be the prime factorization of $m \in \mathbb{Z}_+$. It is well known that \mathbb{Z}_m is isomorphic to the direct sum $\bigoplus_{i=1}^\ell \mathbb{Z}_{p_i^{n_i}}$, and we can reduce the problem to the prime power case.

► **Proposition 4** (\star). *Suppose that the ring R is isomorphic to a direct sum $\bigoplus_{i=1}^\ell R_i$. If $\text{MIN-2-LIN}(R_i)$ is FPT-approximable within a factor c_i for all $i \in [\ell]$, then $\text{MIN-2-LIN}(R)$ is FPT-approximable within a factor $\sum_{i=1}^\ell c_i$.*

Now, consider the ring \mathbb{Z}_{p^n} for a prime p and positive integer n . We start with a simplification step. An equation over a ring R is *simple* if it is either a binary equation of the form $u = rv$ for some $r \in R$ or a crisp unary equation $u = r$ for some $r \in R$. An instance S of $2\text{-LIN}(R)$ is simple if every equation in S is simple.



■ **Figure 1** Graphs B_e corresponding to equations $x = 1 \cdot y$, $x = 2 \cdot y$, $x = 3 \cdot y$ and $x = 4 \cdot y$.

► **Lemma 5** (*). *Let m be positive integer and c be a constant. If $\text{MIN-2-LIN}(\mathbb{Z}_m)$ restricted to simple instances is FPT-approximable within a factor c , then $\text{MIN-2-LIN}(\mathbb{Z}_m)$ on general instances is FPT-approximable within a factor c .*

Now, suppose $n = 1$, i.e. we are working over a field \mathbb{Z}_p . $\text{MIN-2-LIN}(\mathbb{Z}_p)$ can be solved exactly in FPT time (see. [8]); we will show a 2-approximation because it is illustrative of our approach. For an instance S of $\text{2-LIN}(\mathbb{Z}_p)$, we construct a graph $G = G(S)$ with vertices x_i for every $x \in V(G)$ and $1 \leq i \leq p-1$ and special vertices s and t . For unary equations $x = r$ in S , add crisp edges sx_r if $r \neq 0$, and crisp edges x_rt for every $1 \leq r \leq p-1$ if $r = 0$. For every binary equation e of the form $x = r \cdot y$, construct an edge bundle $B_e = \{x_{ri}y_i : 1 \leq i \leq p-1\}$ and add these edges to G . This completes the construction. We establish a correspondence between conformal st -cuts in $G(S)$ and assignments to S . Formally, for a set of vertices $X \subseteq V(G)$, let $\delta(X)$ be the set of edges with exactly one endpoint in X , i.e. $\delta(X)$ is the cut separating X and \bar{X} . If $U \subseteq V(G)$ is such that $s \in U$, $t \notin U$ and there is at most one vertex $x_i \in U$ for any $x \in V(S)$, we say that $\delta(U)$ is a *conformal st -cut*. To construct an assignment from a conformal cut U , map $x \mapsto i$ whenever $x_i \in U$ for some $i \in \{1, \dots, p-1\}$, and otherwise $x \mapsto 0$; the reverse direction of the correspondence is obvious.

Let $b(U)$ be the number of edge bundles B_e intersected by $\delta(U)$. Then R being a field implies that $b(U)$ is exactly the cost of the assignment corresponding to U . More specifically, consider an equation $e = (x = r \cdot y)$, suppose $x_i, y_j \in U$ and $B_e \cap \delta(U) = \emptyset$. Then $i = rj$ because the edge $x_{rj}y_j$ is uncut and both its endpoints are reachable from s , so the assignment corresponding to U satisfies e . This guarantee is represented by B_e being a matching: view an edge x_iy_j as encoding “ $x = i$ if and only if $y = j$ ”. To complete the algorithm for fields, we compute a conformal cut $\delta(U)$ with $b(U) \leq k$. Observe that $b(U) \leq k$ implies $\delta(U) \leq 2k$ because U contains at most one vertex per variable so $\delta(U)$ may intersect at most two edges of any bundle. Thus, for 2-approximation it suffices to compute a conformal cut of size $2k$, which is guaranteed to exist for yes-instances. However, when translating it back into a set of equations, we may delete $2k$ equations because all edges of our conformal cut may intersect distinct bundles. Finally, to compute a conformal cut in FPT time, we may use branching in the style of DIGRAPH PAIR CUT [24]: compute the closest st -cut (which is unique by submodularity of cuts), and if it is not conformal, then branch.

If we use the same approach naïvely over \mathbb{Z}_{p^n} , $n \geq 2$, then the bundles B_e stop being matchings. Consider the equation $e = (x = 2y)$ over \mathbb{Z}_8 (second graph from the left in Figure 1). Note that both y_2 and y_6 are adjacent to x_4 in B_e . Moreover, if $x = 4$ then either $y = 2$ or $y = 6$ so the dependencies cannot be captured by binary edges (even if one were to use directed graphs). Thus, we lose the connection between the number of bundles intersected by a conformal cut and the cost of the corresponding assignment. One idea for solving $\text{MIN-2-LIN}(\mathbb{Z}_{p^n})$ is to retain the “if and only if” semantics of an edge by matching

sets of values rather than individual values. More specifically, let us partition $\{1, \dots, p^n - 1\}$ into classes C_1, \dots, C_ℓ and build $G(S)$ with vertices s, t and $x_{C_1}, \dots, x_{C_\ell}$ for all $x \in V(S)$. To keep the matching structure for an equation e , we want an edge $x_{C_i} y_{C_j}$ in B_e to mean “ $x \in C_i$ if and only if $y \in C_j$ ”. For fields, we have used the most refined partition (every nonzero element is a class of its own). For other rings, a coarser partition is needed: e.g. if $R = \mathbb{Z}_8$, then 2 and 6 have to be in the same class (think of $x = 2y$). However, simply taking a coarser partition is not sufficient: indeed, the coarsest partition (putting all non-zero elements in the same class) has the required structure, but it only distinguishes between zero and nonzero values, and is not very useful algorithmically. Intuitively, we want a partition such that a class assignment over \mathbb{Z}_{p^n} allows us to rewrite our input as a set of equations over $\mathbb{Z}_{p^{n-1}}$ without increasing the cost too much.

A useful partition is obtained by viewing the elements of \mathbb{Z}_{p^n} represented in base- p . Formally, every element $a \in \mathbb{Z}_{p^n}$ equals $\sum_{i=0}^{p^n-1} a_i p^i$, where the coefficients $a_0, \dots, a_{p-1} \in \mathbb{Z}_p$ uniquely define a . Let $\vec{a} = (a_0, \dots, a_{p-1})$, and for every $a \neq 0$, define $\text{ord}(a) = \min\{i : a_i \neq 0\}$ to be the index of the first nonzero coordinate in \vec{a} , and $\text{lsu}(a) = r_{\text{ord}(a)}$ to be the least significant unit in \vec{a} . For completeness, let $\text{ord}(0) = \text{lsu}(0) = 0$. Let

$$a \equiv b \iff \text{ord}(a) = \text{ord}(b) \text{ and } \text{lsu}(a) = \text{lsu}(b).$$

This equivalence relation has two important properties. First, it is *matching*, i.e. $\{0\}$ is an equivalence class, and for every $i, j \in \mathbb{Z}_{p^n}$ and $r \in \mathbb{Z}_{p^n}$,

- if $i \equiv j$ then $ri \equiv rj$,
- if $i \not\equiv j$, then either $ri \not\equiv rj$ or $ri = rj = 0$.

Moreover, it is *absorbing*, meaning that

- $i \equiv j \implies p$ divides $i - j$ for all $i, j \in \mathbb{Z}_{p^n}$.

Let Γ_{p^n} denote the set of equivalence classes of \equiv , and $\Gamma_{p^n}^{\neq 0} = \Gamma_{p^n} \setminus \{\{0\}\}$. We will drop the subscript when it is clear from the context. The name “matching” comes from considering bipartite graphs $G_r^{\neq 0}$ defined by binary equations $u = rv$ for every $r \in R$ as follows: let $V(G_r) = \Gamma^{\neq 0} \uplus \Gamma^{\neq 0}$ and let there be an edge between two classes C_1 on the left and C_2 on the right if and only if $i = rj$ for some $i \in C_1$ and $j \in C_2$. Then \equiv being matching implies that $G_r^{\neq 0}$ is a matching for every r , i.e. every vertex has degree at most 1.

► **Example 6.** Partition Γ_{3^2} (with elements written in base-3) has classes

$$\{01_3, 11_3, 21_3\}, \{02_3, 12_3, 22_3\}, \{10_3\}, \{20_3\}, \{00_3\}.$$

For another example, Γ_{2^3} has classes

$$\{001_2, 011_2, 101_2, 111_2\}, \{010_2, 110_2\}, \{100_2\}, \{000_2\}.$$

For a non-example, a coarser partition of \mathbb{Z}_{2^3} into three classes

$$\{001_2, 011_2, 101_2, 111_2\}, \{010_2, 110_2, 100_2\}, \{000_2\}.$$

lacks the matching property (as is evident from the equation $x = 2y$ in Figure 1).

The matching property of \equiv is crucial for the main algorithmic lemma, which we state below and prove in Section 3.2. A value assignment $\alpha : V(S) \rightarrow \mathbb{Z}_{p^n}$ agrees with a class assignment $\tau : V(S) \rightarrow \Gamma$ if $\alpha(v) \in \tau(v)$ for all $v \in V(S)$. A class assignment τ respects an equation e if it admits a satisfying assignment that agrees with τ , otherwise we say that τ violates e . Define the cost of a class assignment τ to be the number of equations in S

that it violates. Note that every value assignment α uniquely defines a class assignment $\tau_\alpha : V(S) \rightarrow \Gamma_{p^n}$, so we say that α *strongly violates* an equation e if τ_α violates e . Clearly, an optimal assignment can violate at most k equations in S , and we can guess the number $q \in \{0, \dots, k\}$ of strongly violated equations.

► **Lemma 7.** *Let p be a prime, n be a positive integer, and let k and q be integers with $k \geq q$. There is a randomized algorithm that takes a simple instance S of $2\text{-LIN}(\mathbb{Z}_{p^n})$ and integers k and q as input, and in $O^*(2^{O(k \log k)})$ time returns a class assignment $\tau : V(S) \rightarrow \Gamma_{p^n}$ such that the following holds. Let Y be the set of equations in S violated by τ . If S admits an assignment that strongly violates q equations and has cost k , then with probability at least $2^{-O(q^2)}$, $|Y| \leq 2q$ and $S - Y$ admits an assignment of cost at most $k - q$ that agrees with τ .*

While technical details are deferred to Section 3.2, we remark that we can no longer use branching in the style of DIGRAPH PAIR CUT when working over non-fields since some conformal cuts of low cost correspond to class assignments of high cost. For an extreme example over \mathbb{Z}_8 , consider the system of equations $\Delta = \{x = 4, 2a = x, 3a = b, 3b = c, 3c = a\}$. By the construction of $G(\Delta)$, the vertex t is isolated and the connected component U of s contains vertices $s, x_{\{4\}}, a_{\{2,6\}}, b_{\{2,6\}}$ and $c_{\{2,6\}}$. Hence, $\delta(U)$ is empty and conformal, but the cost of Δ is at least 1 because the system is inconsistent (this follows from considering both possible values for a , which are 2 and 6). In fact, the cost is exactly 1 because it is sufficient to delete $2a = x$. To mitigate this issue of “invisible” future costs, we use shadow removal in the class assignment graph followed by branching on the shadow components.

Now, to explain how the absorbing property of \equiv is used, we need some definitions. Choose an arbitrary representative element C^\sim from every equivalence class $C \in \Gamma$. Consider a simple equation e and a class assignment $\tau : V(e) \rightarrow \Gamma$ to its variables that respects e . For unary equations $e = (u = r)$, define $e' = \text{next}(e, \tau)$ to be

$$u' = \frac{r - \tau(u)^\sim}{p}.$$

For binary equations $e = (u = rv)$, define $e' = \text{next}(e, \tau)$ to be

$$u' = rv' + \frac{r\tau(v)^\sim - \tau(u)^\sim}{p}.$$

The absorbing property implies that $\text{next}(e, \tau)$ is defined in both cases. Indeed, if $e = (u = r)$ and τ respects e , then $r \in \tau(u)$ and $r \equiv \tau(u)^\sim$, so p divides $r - \tau(u)^\sim$. If $e = (u = rv)$ and τ respects e , then there is an assignment $\alpha : \{u, v\} \rightarrow \mathbb{Z}_{p^n}$ such that $\alpha(u) = r\alpha(v)$, $\alpha(u) \in \tau(u)$ and $\alpha(v) \in \tau(v)$. Equivalently, $\alpha(u) - r\alpha(v) = 0$ and p divides both $\tau(u)^\sim - \alpha(u)$ and $\tau(v)^\sim - \alpha(v)$, hence p also divides any linear combination of these two values, particularly

$$r(\tau(v)^\sim - \alpha(v)) - (\tau(u)^\sim - \alpha(u)) = r\tau(v)^\sim - \tau(u)^\sim + (\alpha(u) - r\alpha(v)) = r\tau(u)^\sim - \tau(v)^\sim.$$

► **Lemma 8 (*)**. *Let p be a prime and $n \in \mathbb{Z}_+$. Let e be a simple equation over \mathbb{Z}_{p^n} , and $\tau : V(e) \rightarrow \Gamma$ be a class assignment. Then τ respects e if and only if $\text{next}(e, \tau)$ is satisfiable over $\mathbb{Z}_{p^{n-1}}$.*

Now we combine all ingredients to prove the main theorem.

► **Theorem 1.** *For every $m \in \mathbb{Z}_+$, $\text{MIN-2-LIN}(\mathbb{Z}_m)$ is FPT-approximable within $2\omega(m)$.*

Proof sketch. Let $m = p_1^{n_1} \cdots p_\ell^{n_\ell}$ be the prime factorization of m . Note that $\omega(m) = \ell$, so by Proposition 4, it suffices to show that $\text{MIN-2-LIN}(\mathbb{Z}_{p^n})$ is FPT-approximable within

factor 2 for every prime p and positive integer n . We proceed by induction on n . If $n = 1$, then we can use the algorithm of [8] for MIN-2-LIN over fields to solve the problem exactly. Otherwise, let (S, k) be an instance of MIN-2-LIN(\mathbb{Z}_{p^n}). By Lemma 5, we may assume without loss of generality that S is simple.

Guess $q \in \{0, \dots, k\}$ and run the algorithm from Lemma 7 on (S, k, q) to produce a class assignment $\tau : V(S) \rightarrow \Gamma_{p^n}$. Let Y be the set of equations in S violated by τ . If $|Y| > 2q$, then reject (S, k) . Otherwise, create an instance S' of 2-LIN($\mathbb{Z}_{p^{n-1}}$) with $V(S') = \{v' : v \in V(S)\}$ and $S' = \{\text{next}(e, \tau) : e \in S - Y\}$. Set $k' = k - q$ and pass (S', k') as input to the algorithm for MIN-2-LIN($\mathbb{Z}_{p^{n-1}}$), and return the same answer.

Correctness of the algorithm follows from Lemmas 7 and 8. On the one hand, if (S, k) is a yes-instance admitting an optimal assignment that strongly violates q equations, then our guess for q is correct with probability $1/(k+1)$, and the instance S' we produce from $S - Y$ has cost at most $k - q$. On the other hand, if we obtain a 2-approximate solution for S' , then translating it back into a solution for $S - Y$ of size $\leq 2(k - q)$ and combining with Y , which is of size $\leq 2q$, yields a solution for S of size $\leq 2k$. ◀

3.2 Computing Class Assignments

This subsection is devoted to a proof of Lemma 7. To achieve this we introduce the class assignment graph (Section 3.2.1) and show that certain cuts in this graph correspond to class assignments (Section 3.2.2), which themselves correspond to solutions of MIN-2-LIN(\mathbb{Z}_{p^n}). We then use shadow removal and branching to compute these cuts (Section 3.2.3). Throughout this section, we consider the ring \mathbb{Z}_{p^n} for prime number p and integer n .

3.2.1 The Class Assignment Graph

In what follows, let (S, k) be a simple instance of MIN-2-LIN(\mathbb{Z}_{p^n}) and let \equiv be the matching and absorbing equivalence relation on \mathbb{Z}_{p^n} defined in Section 3.1. Without loss of generality, we assume that every equation of S is consistent (otherwise we can remove the equation and decrease k by 1) and non-trivial (otherwise we can remove the equation). We first use the matching property of \equiv to define the mapping $\pi_e : \Gamma_{p^n} \rightarrow \Gamma_{p^n}^{\neq 0}$ between equivalence classes for any equation $e = (ax = y)$ with $a \in \mathbb{Z}_{p^n} \setminus \{0\}$, as follows. For every $C \in \Gamma_{p^n}$, we set:

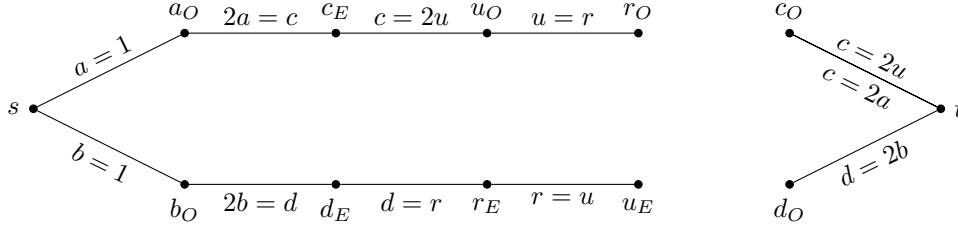
- $\pi_e(C) = 0$ if $ar = 0$ for every $r \in C$ and
- $\pi_e(C) = D$ otherwise, where D is the unique equivalence class such that e maps C to D .

This is uniquely defined since \equiv has the matching property.

For an equation $e = (x = a)$ with $a \in \mathbb{Z}_{p^n}$, we let π_e be the unique equivalence class $\Gamma_{p^n}(a)$ consistent with e . We can now define the *class assignment* graph $G = G(S)$; see Figure 2 for an illustration. The graph G has two distinguished vertices s and t together with vertices x_C for every $x \in V(S)$ and every non-zero class $C \in \Gamma_{p^n}^{\neq 0}$. Moreover, G contains the following edges for each equation.

- For an equation $e = (ax = y)$ do the following. For every $C \in \Gamma_{p^n}$, add the edge $x_C y_{\pi_e(C)}$ if $\pi_e(C) \neq 0$. For every $D \in \Gamma_{p^n}^{\neq 0}$ such that $\pi_e^{-1}(D)$ is undefined, add the edge $y_D t$.
- For a crisp equation $e = (x = 0)$, add the crisp edge $x_C t$ for every class $C \in \Gamma_{p^n}^{\neq 0}$.
- For a crisp equation $e = (x = b)$, where $b \neq 0$, add the crisp edge $s x_{\pi_e}$.

Intuitively, every node of G corresponds to a Boolean variable and every edge e of G corresponds to an “if and only if” between the two Boolean variables connected by e . Moreover, every assignment φ of the variables of S naturally corresponds to a Boolean assignment, denoted by φ_G of the vertices in G by setting $s = 1$, $t = 0$, and:



■ **Figure 2** Let S be the instance of $2\text{-LIN}(\mathbb{Z}_4)$ with variables a, b, c, d, u, r and equations $a = 1$, $b = 1$, $2a = c$, $c = 2u$, $u = r$, $2b = d$, and $d = r$. The figure illustrates the class assignment graph $G = G(S)$. Note that \equiv has only two non-zero equivalence classes, namely, $E = \{2\}$ and $O = \{1, 3\}$. Every edge of G is annotated with the equation that implies it. A minimum conformal st -cut is given by the two edges that correspond to the equation $u = r$ and corresponds to the class assignment $a = O$, $b = O$, $c = E$, $d = E$, $u = O$, and $r = E$. Note that G has only one minimal conformal st -cut closest to s , namely $\{a_O c_E, b_O d_E\}$. This st -cut corresponds to the class assignment $a = O$, $b = O$, and $c = d = u = r = 0$. Therefore, the optimum solution for S only removes the equation $u = r$, however, any solution that corresponds to a minimum conformal st -cut closest to s has to remove the equations $2a = c$ and $2b = d$.

- if $\varphi(x)$ belongs to the non-zero class C , then we set $x_C = 1$ and $x_{C'} = 0$ for every non-zero class C' not equal to C ,
- if $\varphi(x) = 0$, we set $x_C = 0$ for every non-zero class C .

We say that an edge e of G is satisfied by φ if φ_G satisfies the “if and only if” Boolean constraint represented by that edge.

► **Observation 9.** Let S be a simple instance of $2\text{-LIN}(\mathbb{Z}_{p^n})$, let φ be an assignment of S and pick $e \in S$. If φ satisfies e , then φ_G satisfies all edges corresponding to e in $G(S)$.

3.2.2 Cuts in the Class Assignment Graph

In this section, we introduce *conformal cuts* and show how they relate to class assignments and solutions to $\text{MIN-2-LIN}(\mathbb{Z}_{p^n})$ instances. Let S be a simple instance of $2\text{-LIN}(\mathbb{Z}_{p^n})$ and $G = G(S)$. An st -cut Y in G is *conformal* if for every variable $x \in V(S)$ at most one vertex x_C for some $C \in \Gamma_{p^n}^{\neq 0}$ is connected to s in $G - Y$. Please refer to Figure 2 for an illustration of conformal cuts in the class assignment graph. If Y is a conformal st -cut in G , then we say that a variable x is *decided* with respect to Y if (exactly) one vertex x_C is reachable from s in $G - Y$; and otherwise we say that x is *undecided* with respect to Y . Moreover, we denote by τ_Y the assignment of variables of S to classes in Γ_{p^n} implied by Y , i.e. $\tau_Y(x) = 0$ if x is undecided and otherwise $\tau_Y(x) = C$, where C is the unique non-zero class in Γ_{p^n} such that x_C is reachable from s in $G - Y$. We say that an assignment φ of S *agrees with* Y if $\varphi(x)$ is in the class $\tau_Y(x)$ for every variable x of S . Note that if some assignment agrees with Y , then Y is conformal. The following auxiliary lemma characterizes which edges of G are satisfied by an assignment φ of S after removing a set Y of edges from G .

► **Lemma 10.** Let Y be a set of edges of G and let φ be an assignment of S . Then, φ_G satisfies all edges reachable from s in $G - Y$ if and only if φ_G sets all Boolean variables reachable from s in $G - Y$ to 1. Similarly, φ_G satisfies all edges reachable from t in $G - Y$ if and only if φ_G sets all Boolean variables reachable from t in $G - Y$ to 0.

Proof. This follows because $\varphi_G(s) = 1$ and $\varphi_G(t) = 0$ for any φ and every edge of G corresponds to an “if and only if” between the variables corresponding to its two endpoints. ◀

For a set Z of equations of S , we let $\text{ed}(Z)$ denote the set of all edges of G corresponding to an equation of Z . Conversely, for a set Y of edges of G , we let $\text{eqn}(Y)$ denote all equations of S having a corresponding edge in Y . Moreover, if Y is an st -cut in G , we let $\text{sep}(Y)$ denote the unique minimal st -cut contained in Y that is *closest* to s in G . Finally, for an optimal solution Z of S , we let \bar{Z} be the set of equations $Z \setminus \text{eqn}(\text{sep}(\text{ed}(Z)))$, i.e. all equations in Z that do not have an edge in $\text{sep}(\text{ed}(Z))$. We can now establish the connection between solutions of $\text{MIN-2-LIN}(\mathbb{Z}_{p^n})$ instances and conformal st -cuts in the class assignment graph.

► **Lemma 11** (*). *Let Z be a set of equations such that $S - Z$ is satisfiable and let $Y = \text{ed}(Z)$. Then, $Y' = \text{sep}(Y)$ satisfies:*

1. Y' is a conformal st -cut.
2. $|Y'| \leq 2|\text{eqn}(Y')| = 2|Z \setminus \bar{Z}|$.
3. *There is a satisfying assignment for $S - Z$ that agrees with Y' .*

Proof Sketch. Let φ be a satisfying assignment of $S - Z$. Observation 9 implies that φ_G satisfies all edges of $G - Y$. Therefore, it follows from Lemma 10 that φ_G sets all vertices reachable from s in $G - Y$ to 1 and all vertices reachable from t in $G - Y$ to 0. Thus, Y is an st -cut, because otherwise t would have to be set to 1 by φ_G since it would be reachable from s in $G - Y$. Therefore, $Y' = \text{sep}(Y)$ exists. Because Y' is closest to s , it holds that a vertex is reachable from s in $G - Y$ if and only if it is reachable from s in $G - Y'$. Therefore, if at least two vertices x_C and $x_{C'}$ for some distinct non-zero classes C and C' are reachable from s in $G - Y'$ for some variable x , then all of them must be set to 1 by φ_G , which is not possible due to the definition of φ_G . We conclude that Y' is conformal.

Towards showing that $|Y'| \leq 2|\text{eqn}(Y')|$, it suffices to show that $|Y' \cap \text{ed}(e)| \leq 2$ for every equation $e \in Z$. Note that because Y' is a minimal st -cut, it holds that one of the endpoints of every $y \in Y'$ is reachable from s in $G - Y'$. Therefore, because Y' is conformal, Y' can contain at most two edges in $\text{ed}(e) = \{x_C y_{\pi_e(C)} \mid C \in \Gamma_{p^n}^{\neq 0} \wedge \pi_e(C) \neq 0\} \cup \{y_{Dt} \mid D \in \Gamma_{p^n}^{\neq 0} \wedge \pi_e^{-1}(D) \text{ is undefined}\}$ for every binary equation e of the form $ax = y$. Similarly, Y' can contain at most one edge in $\text{ed}(e) = \{x_C t \mid C \in \Gamma_{p^n}^{\neq 0}\}$ for every unary equation e of the form $x = 0$. Finally, $|\text{ed}(e)| = |\{sx_{\pi_e}\}| = 1$ for every unary equation e of the form $x = b$. Therefore, $|Y'| \leq 2|\text{eqn}(Y')|$, and $Y' = Z \setminus \bar{Z}$ by definition.

Let D be the set of all variables of S such that no vertex x_C is reachable from s in $G - Y'$. Let φ' be the assignment for S such that $\varphi'(x) = 0$ if $x \in D$ and $\varphi'(x) = \varphi(x)$ otherwise. Clearly, φ' agrees with Y' , because φ agrees with all variables not in D and all other variables are correctly set to 0 by φ' . It therefore only remains to show that φ' still satisfies $S - Z$, which we leave to the full version of the paper. ◀

3.2.3 Shadow Removal

We show how shadow removal (introduced in [28] and improved in [5]) can be used for computing conformal cuts that correspond to solutions of a $\text{MIN-2-LIN}(\mathbb{Z}_{p^n})$ instance. We follow [5] and begin by importing some definitions, which we translate from directed graphs to undirected graphs to fit our setting; to get back to directed graphs one simply has to think of an undirected graph as the directed graph obtained after replacing each undirected edge with two directed arcs in both directions. Let G be an undirected graph. Let \mathcal{F} be a set of connected subgraphs of G . A set $T \subseteq V(G)$ is an \mathcal{F} -*transversal* if T intersects every subgraph in \mathcal{F} . Conversely, if T is an \mathcal{F} -transversal, we say that \mathcal{F} is T -*connected*.

► **Theorem 12** ([5]). *Let G be an undirected graph, $T \subseteq V(G)$ and $k \in \mathbb{N}$. There is a randomized algorithm that takes (G, T, k) as input and returns in $O^*(4^k)$ time a set*

$W \subseteq V(G) \setminus T$ such that the following holds with probability $2^{-O(k^2)}$. For every T -connected family of connected subgraphs \mathcal{F} in G , if there is an \mathcal{F} -transversal of size at most k in $V(G) \setminus T$, then there is an \mathcal{F} -transversal $Y \subseteq V(G) \setminus (W \cup T)$ of size at most k such that every vertex $v \notin W \cup Y$ is connected to T in $G - Y$.

The following lemma is crucial for the application of shadow removal (Theorem 12). Informally, it shows that if Z is a solution, i.e. a set of equations such that $S - Z$ is satisfiable, then we can obtain a (not too large) new solution $Z' = (\bar{Z} \cup \text{eqn}(Y'))$ by replacing the corresponding conformal minimal sA -cut $Y = \text{sep}(\text{ed}(Z))$, where A is the set of vertices in G not reachable from s in $G - Y$, by any minimal sA -cut Y' .

► **Lemma 13.** *Let S be a simple instance of $2\text{-LIN}(\mathbb{Z}_{p^n})$ and $G = G(S)$. Moreover, let Z be a set of equations such that $S - Z$ is satisfiable, $Y = \text{sep}(\text{ed}(Z))$, A be the set of all vertices in G that are not reachable from s in $G - Y$, and let Y' be an sA -cut in G . Then, there is an assignment $\varphi : V(S) \rightarrow \mathbb{Z}_{p^n}$ of S that satisfies $S - Z'$ and agrees with Y' , where $Z' = (\bar{Z} \cup \text{eqn}(Y'))$.*

Proof. Lemma 11 implies that Y is conformal and there is a satisfying assignment φ for $S - Z$ that agrees with Y . Because Y' is also an sA -cut in G , if no vertex x_C is reachable from s in $G - Y$ for some variable x of S , then the same applies in $G - Y'$. Let D be the set of all variables x of S such that some vertex x_C is reachable from s in $G - Y$ but that is not the case in $G - Y'$. Let φ' be the assignment obtained from φ by setting all variables in D to 0. Then, φ' agrees with Y' . We claim that φ' also satisfies $S - Z'$, where $Z' = (\bar{Z} \cup \text{eqn}(Y'))$.

Consider a unary equation e of $S - Z'$ on variable x . If $x \notin D$, then $\varphi'(x) = \varphi(x)$ and therefore φ' satisfies e (because e is crisp and therefore $e \notin Z$). So suppose that $x \in D$. If e is of the form $x = 0$, then φ' satisfies e . Otherwise, e is of the form $x = b$ for some $b \neq 0$ and $G - Y'$ contains the edge sx_{π_e} . Therefore, x_{π_e} is reachable from s in $G - Y'$ contradicting our assumption that $x \in D$.

Now, consider a binary equation $e = (ax = y)$ of $S - Z'$ on variables x and y and first consider the case when $e \in Z$. Clearly, if neither a vertex x_C nor a vertex y_C is reachable from s in $G - Y'$, then $\varphi'(x) = \varphi'(y) = 0$, so e is satisfied by φ' . We next show that either no vertex x_C or no vertex y_C is reachable from s in $S - Y'$. Suppose for a contradiction that x_{C_x} and y_{C_y} are reachable from s in $S - Y'$. Let h be an arbitrary edge in $\text{ed}(e) \cap Y$; such an edge h exists because $e \in Z \setminus Z'$. Because Y is a minimal st -cut, it follows that exactly one endpoint of h is reachable from s in $G - Y$ and either x_C or y_C (endpoint of h) for some $C \in \Gamma_{p^n}$ must be reachable from s in $G - Y$. We assume without loss of generality that x_C is reachable from s in $G - Y$. Because Y' is an sA -cut and Y' does not contain h , x_C is not reachable from s in $G - Y'$. But then $C \neq C_x$ and both x_C and x_{C_x} are reachable from s in $G - Y$, which contradicts that Y is conformal. It remains to consider the case when there is a vertex x_C that is reachable from s in $G - Y'$ but no vertex y_C is reachable from s in $G - Y'$; the case when there is a vertex y_C reachable from s in $G - Y'$ but no vertex x_C reachable from s in $G - Y'$ is analogous. Since $Y' \cap \text{ed}(e) = \emptyset$, we obtain that $\pi_e(C) = 0$ since otherwise either t or some $y_{C'}$ would be reachable from s in $G - Y'$. Because $\varphi'(y) = 0$, it follows that e is satisfied by φ' . This completes the proof for the case when $e \in Z$.

Suppose instead that $e \notin Z$. In this case φ satisfies e and therefore φ' also satisfies e unless exactly one of x and y is not in D . We distinguish the following cases:

- $x \notin D$ and $y \in D$. If there is no vertex x_C that is reachable from s in $G - Y'$, then the same holds in $G - Y$ so $\varphi'(x) = \varphi(x) = \varphi'(y) = 0$, which shows that φ' satisfies e . Otherwise, let x_C be reachable from s in $G - Y'$. Then, $\pi_e(C) = 0$ since otherwise either t or $y_{\pi_e(C)}$ is also reachable from s in $G - Y'$ (because $Y' \cap \text{ed}(e) = \emptyset$), which in the

former case contradicts our assumption that Y' is an st -cut and which in the latter case contradicts our assumption that $y \in D$. Therefore, φ' satisfies e (because $\varphi'(y) = 0$).

- $x \in D$ and $y \notin D$. We first show that there is no vertex y_C that is reachable from s in $G - Y'$. Suppose there is such a vertex y_C . Then, $\pi_e^{-1}(C)$ is undefined since otherwise $x_{\pi_e^{-1}(C)}$ is reachable from s in $G - Y'$ (because $Y' \cap \text{ed}(e) = \emptyset$), which contradicts our assumption that $x \in D$. But then $y_C t \in E(G - Y')$ and t is reachable from s in $G - Y'$, which contradicts our assumption that Y' is an st -cut. Hence, there is no vertex y_C that is reachable from s in $G - Y'$, which implies that the same holds in $G - Y$ so $\varphi'(y) = \varphi(y) = \varphi'(x) = 0$, which shows that φ' satisfies e . ◀

Let $G = G(S)$ and for a set $W \subseteq V(G)$, let $\delta(W)$ be the set of edges incident to a vertex in W and a vertex in $V(G) \setminus W$. The forthcoming Lemma 14 provides a version of shadow removal adopted to our problem. Informally, it provides us with a set $W \subseteq V(G)$ such that we only have to look for conformal st -cuts that are subsets of $\delta(W)$ to obtain our class assignment; in fact it even shows that for every component C of $G[W]$ either all edges in $\delta(C)$ are part of the cut or no edge of $\delta(C)$ is part of the cut. We will use this fact in Lemma 15 to find a conformal st -cut by branching on which components of $G[W]$ are reachable from s .

More formally, if Z is a set of equations such that $S - Z$ is satisfiable and A is the set of vertices not reachable from s in G minus the conformal st -cut $\text{sep}(\text{ed}(Z))$ (see Lemma 11), then the lemma provides us with a set $W \subseteq V(G)$ such that there is a conformal sA -cut Y' within $\delta(W)$ of size at most $2|Z \setminus \bar{Z}|$ such that there is an assignment $\varphi : V(S) \rightarrow \mathbb{Z}_{p^n}$ for the variables in S that satisfies $S - (\bar{Z} \cup \text{eqn}(Y'))$ and agrees with Y' . The main idea behind the proof is the application of Theorem 12 to the set of all walks from s to A in G to obtain the set W and to employ Lemma 13 to obtain the new solution that corresponds to the minimum sA -cut $Y' \subseteq \delta(W)$.

► **Lemma 14** (★). *Let S be a simple instance of $2\text{-LIN}(\mathbb{Z}_{p^n})$ and let $G = G(S)$. Moreover, let Z be a set of equations such that $S - Z$ is satisfiable, $Y = \text{sep}(\text{ed}(Z))$, let A be the set of all vertices in G that are not reachable from s in $G - Y$, and let $q = |Z \setminus \bar{Z}|$. There is a randomized algorithm that in $\mathcal{O}^*(4^{2q})$ time takes (G, q) as input and returns a set $W \subseteq V(G) \setminus \{s\}$ such that the following holds with probability $2^{-\mathcal{O}(q^2)}$. There is a (minimal) sA -cut Y' of size at most $2q$ satisfying:*

1. *every vertex $v \notin W$ is connected to s in $G - Y'$,*
2. *$Y' \subseteq \delta(W)$,*
3. *there is satisfying assignment for $S - (\bar{Z} \cup \text{eqn}(Y'))$ that agrees with Y' .*

Moreover, for every component C of $G[W]$ the following holds:

resume *either $Y' \cap \delta(C) = \emptyset$ or $\delta(C) \subseteq Y'$,*

resume *if $t \in C$, then $\delta(C) \subseteq Y'$,*

resume *if $x_\alpha, x_{\alpha'} \in C$ for some variable x and $\alpha \neq \alpha'$, then $\delta(C) \subseteq Y'$,*

resume *if C contains some x_α for some decided variable x w.r.t. Y' , then $\delta(C) \subseteq Y'$.*

The following lemma now uses the set $W \subseteq V(G)$ computed in Lemma 14 to compute a set of at most $2^{\mathcal{O}(k \log k)}$ conformal cuts \mathcal{Y} each of size at most $2q$ such that if S has a solution Z of size at most k such that $|Z \setminus \bar{Z}| = q$, then there is a cut $Y \in \mathcal{Y}$ of size at most $2q$ together with an assignment satisfying $S - (\bar{Z} \cup \text{eqn}(Y))$ that agrees with Y . Note that Lemma 7 is now an immediate consequence of Lemma 15, i.e. instead of returning the set \mathcal{Y} of conformal cuts, we choose one conformal cut $Y \in \mathcal{Y}$ uniformly at random and output the class assignment corresponding to Y . The idea behind computing \mathcal{Y} is that we only need to consider conformal cuts that are within $\delta(W)$ and this allows us to branch on which components of $G[W]$ are reachable from s (see also Property 4. in Lemma 14).

► **Lemma 15** (\star). *Let S be a simple instance of $2\text{-LIN}(\mathbb{Z}_{p^n})$, $G = G(S)$, and let k and q with $k \geq q$ be integers. There is a randomized algorithm that takes (G, k, q) as input and returns in $\mathcal{O}^*(2^{\mathcal{O}(k \log k)})$ time a set \mathcal{Y} of at most $2^{\mathcal{O}(k \log k)}$ conformal cuts (each of size at most $2q$), such that with probability at least $2^{-\mathcal{O}(q^2)}$ there is a cut $Y \in \mathcal{Y}$ with the following property: if S has a solution Z of size at most k such that $q = |Z \setminus \bar{Z}|$, then $|Y| \leq 2q$ and there is an assignment that satisfies $S - (\bar{Z} \cup \text{eqn}(Y))$ and agrees with Y .*

4 Hardness of FPT-Approximation

We complement the approximation algorithm with hardness results: we prove (1) that for finite, commutative, non-trivial rings R , $\text{MIN-}r\text{-LIN}(R)$ is $\text{W}[1]$ -hard to FPT-approximate within any constant when $r \geq 3$ and (2) the existence of finite commutative rings R such that $\text{MIN-}2\text{-LIN}(R)$ is $\text{W}[1]$ -hard to FPT-approximate within any constant.

Let G denote an arbitrary Abelian group. An expression $x_1 + \dots + x_r = c$ is an *equation over G* if $c \in G$ and x_1, \dots, x_r are either variables or inverted variables with domain G . We say that it is an *r -variable equation* if it contains at most r distinct variables. A *cyclic* group is generated by a single element and every finite cyclic group C_n of order n is isomorphic to the additive group of \mathbb{Z}_n . Cyclic groups are the building blocks of more complex Abelian groups: the fundamental theorem of finite Abelian groups asserts that every finite Abelian group is a direct sum of cyclic groups whose orders are prime powers.

We consider the natural group-based variant $\text{MIN-}r\text{-LIN}(G)$ of the $\text{MIN-}r\text{-LIN}(R)$ problems in what follows. We first prove that $\text{MIN-}3\text{-LIN}(C_p)$, with p a prime, is not FPT-approximable within any constant if $\text{FPT} \neq \text{W}[1]$. Our result is based on a reduction from a fundamental problem in coding theory: the $\text{MAXIMUM LIKELIHOOD DECODING}$ problem over \mathbb{Z}_p with p prime. Here we are given a matrix $A \in \mathbb{Z}_p^{n \times m}$, a vector $b \in \mathbb{Z}_p^m$, and the goal is to find $x \in \mathbb{Z}_p^n$ such that $Ax = b$ with minimum Hamming weight, i.e. the one that minimizes $k = |\{i \in [n] : x_i \neq 0\}|$. The parameter is k . Theorem 5.1 in [2] proves that for every prime p , the problem MLD_p is $\text{W}[1]$ -hard to approximate within any constant factor.

Intuitively, row i in $Ax = b$ is a linear equation $\sum_{j=1}^n a_{ij}x_j = b_j$, where $a_{ij}, b_j \in \mathbb{Z}_p$ are coefficients and x_j are variables. There is a straightforward way to subdivide long equations into ternary equations: for example, if we have an equation $x_1 + x_2 + x_3 + x_4 = 1$, we can introduce auxiliary variables y_1, y_2, y_3 and write

$$x_1 + x_2 - y_1 = 0, \quad y_1 + x_3 - y_2 = 0, \quad y_2 + x_4 - y_3 = 0 \quad \text{and} \quad y_3 = 1.$$

When summing up these equations, auxiliary variables cancel out and we obtain $x_1 + x_2 + x_3 + x_4 = 1$. Using this trick, we encode the constraints implied by the row equations of $Ax = b$ as crisp ternary and unary equations. To encode the objective function, i.e. the fact that we are minimizing the Hamming weight of x , we add soft equations $x_j = 0$ for all $j \in [n]$. This way, breaking a soft equation corresponds to increasing the Hamming weight by 1. This hardness result for C_p , with p prime, can be lifted into the general case via two simple steps: algebraic manipulations allow us to show hardness for C_{p^l} , $l \geq 1$, and this implies hardness for $\text{MIN-}r\text{-LIN}(G)$ by recalling that G is a direct sum of cyclic groups of prime power order. We obtain the following due to the additive group of every ring being Abelian.

► **Theorem 16** (\star). *Let R be a non-trivial finite ring. $\text{MIN-}r\text{-LIN}(R)$ is $\text{W}[1]$ -hard to FPT-approximate within any constant factor when $r \geq 3$.*

We use Theorem 16 to demonstrate that there exist finite commutative rings such that $\text{MIN-}2\text{-LIN}(R)$ is $\text{W}[1]$ -hard to FPT-approximate within any constant factor. Let R

denote the 16-element polynomial ring $\mathbb{Z}_2[x, y]/(x^2, y^2)$, i.e. the ring with coefficients from \mathbb{Z}_2 and indeterminates x, y with x^2 and y^2 factored out. An element $r \in R$ is thus a sum $r_{\text{unit}} + r_x x + r_y y + r_{xy} xy$, where $r_{\text{unit}}, r_x, r_y, r_{xy} \in \mathbb{Z}_2$. The idea is to express equations of length 3 over \mathbb{Z}_2 using equations of length 2 over R . We illustrate this by considering an equation $a + b + c = 0$ over \mathbb{Z}_2 . To express it using binary equations over R , we introduce a fresh variable v and three equations (1) $xv = xyb$, (2) $yv = xya$, and (3) $(x + y)v = -xyc$. Summing up the first two equations, we obtain $(x + y)v = xy(a + b)$. Together with the third one, this implies $xy(a + b + c) = 0$. On the other hand, any assignment that satisfies $xy(a + b + c) = 0$ can be extended as $v = xa + yb$ to satisfy all three binary equations. With this in mind, it is not too difficult to prove the following with the aid of Theorem 16.

► **Theorem 17** (*). *MIN-2-LIN(R) is $W[1]$ -hard to FPT-approximate within any constant factor when $R = \mathbb{Z}_p[x_1, \dots, x_k]/(x_1^2, \dots, x_k^2)$, p prime, and $k \geq 2$.*

References

- 1 Vikraman Arvind and T. C. Vijayaraghavan. The complexity of solving linear equations over a finite ring. In *Proc. 22nd Annual Symposium on Theoretical Aspects of Computer Science (STACS-2005)*, pages 472–484, 2005.
- 2 Arnab Bhattacharyya, Édouard Bonnet, László Egri, Suprovat Ghoshal, Bingkai Lin, Pasin Manurangsi, and Dániel Marx. Parameterized intractability of even set and shortest vector problem. *Journal of the ACM*, 68(3):1–40, 2021.
- 3 Ian F. Blake. Codes over certain rings. *Information and Control*, 20(4):396–404, 1972.
- 4 Édouard Bonnet, László Egri, and Dániel Marx. Fixed-parameter approximability of Boolean MinCSPs. In *Proc. 24th Annual European Symposium on Algorithms (ESA-2016)*, pages 18:1–18:18, 2016.
- 5 Rajesh Chitnis, Marek Cygan, MohammadTaghi Hajiaghayi, and Dániel Marx. Directed subset feedback vertex set is fixed-parameter tractable. *ACM Transactions on Algorithms*, 11(4):1–28, 2015.
- 6 Konrad K. Dabrowski, Peter Jonsson, Sebastian Ordyniak, George Osipov, Marcin Pilipczuk, and Roohani Sharma. Parameterized complexity classification for interval constraints. In *Proc. 18th International Symposium on Parameterized and Exact Computation (IPEC-2023)*, pages 11:1–11:19, 2023.
- 7 Konrad K. Dabrowski, Peter Jonsson, Sebastian Ordyniak, George Osipov, and Magnus Wahlström. Towards a parameterized approximation dichotomy of mincsp for linear equations over finite commutative rings. *CoRR*, abs/2410.09932, 2024. URL: <https://doi.org/10.48550/arXiv.2410.09932>, doi:10.48550/ARXIV.2410.09932.
- 8 Konrad K. Dabrowski, Peter Jonsson, Sebastian Ordyniak, George Osipov, and Magnus Wahlström. Almost consistent systems of linear equations. In *Proc. 34th Annual ACM-SIAM Symposium on Discrete Algorithms (SODA-2023)*, pages 3179–3217, 2023.
- 9 Anuj Dawar, Erich Grädel, Bjarki Holm, Eryk Kopczynski, and Wied Pakusa. Definability of linear equation systems over groups and rings. *Logical Methods in Computer Science*, 9(4), 2013.
- 10 Reinhard Diestel. *Graph Theory*. Springer, Berlin, Heidelberg, 6th edition, 2025.
- 11 Cunsheng Ding, Dingyi Pei, and Arto Salomaa. *Chinese Remainder Theorem: Applications in Computing, Coding, Cryptography*. World Scientific, 1996.
- 12 Rodney G. Downey and Michael R. Fellows. *Parameterized Complexity*. Springer, 1999.
- 13 Eduard Eiben, Clément Rambaud, and Magnus Wahlström. On the parameterized complexity of symmetric directed multicut. In *Proc. 17th International Symposium on Parameterized and Exact Computation (IPEC-2022)*, pages 11:1–11:17, 2022.

- 14 Andreas Emil Feldmann, Karthik C. S., Euiwoong Lee, and Pasin Manurangsi. A survey on approximation in parameterized complexity: Hardness and algorithms. *Algorithms*, 13(6):146, 2020.
- 15 Jörg Flum and Martin Grohe. *Parameterized Complexity Theory*. Springer, 2006.
- 16 Joseph F. Grcar. How ordinary elimination became Gaussian elimination. *Historia Mathematica*, 38(2):163–218, 2011.
- 17 Venkatesan Guruswami, Bingkai Lin, Xuandi Ren, Yican Sun, and Kewen Wu. Parameterized inapproximability hypothesis under exponential time hypothesis. In *Proc. 56th Annual ACM Symposium on Theory of Computing (STOC-2024)*, pages 24–35, 2024.
- 18 Venkatesan Guruswami, Xuandi Ren, and Sai Sandeep. Baby PIH: parameterized inapproximability of MinCSP. In *Proc. 39th Computational Complexity Conference (CCC-2024)*, pages 27:1–27:17, 2024.
- 19 Johan Håstad. Some optimal inapproximability results. *Journal of the ACM*, 48(4):798–859, 2001.
- 20 Subhash Khot. On the power of unique 2-prover 1-round games. In *Proc. 24th Annual ACM Symposium on Theory of Computing (STOC-2002)*, pages 767–775, 2002.
- 21 Subhash Khot and Dana Moshkovitz. Candidate hard unique game. In *Proc. 48th Annual ACM Symposium on Theory of Computing (STOC-2016)*, pages 63–76, 2016.
- 22 Eun Jung Kim, Stefan Kratsch, Marcin Pilipczuk, and Magnus Wahlström. Flow-augmentation III: complexity dichotomy for boolean CSPs parameterized by the number of unsatisfied constraints. In *Proc. 2023 ACM-SIAM Symposium on Discrete Algorithms (SODA-2023)*, pages 3218–3228, 2023.
- 23 Vladimir Kolmogorov, Andrei A. Krokhin, and Michal Rolínek. The complexity of general-valued CSPs. *SIAM Journal on Computing*, 46(3):1087–1110, 2017.
- 24 Stefan Kratsch and Magnus Wahlström. Representative sets and irrelevant vertices: New tools for kernelization. *Journal of the ACM*, 67(3):1–50, 2020.
- 25 Daniel Lokshtanov, Pranabendu Misra, M. S. Ramanujan, Saket Saurabh, and Meirav Zehavi. FPT-approximation for FPT problems. In *Proc. 2021 ACM-SIAM Symposium on Discrete Algorithms (SODA-2021)*, pages 199–218, 2021.
- 26 Daniel Lokshtanov, M. S. Ramanujan, Saket Saurabh, and Meirav Zehavi. Parameterized complexity and approximability of directed odd cycle transversal. In *Proc. 40th Annual ACM-SIAM Symposium on Discrete Algorithms (SODA-2020)*, pages 2181–2200, 2020.
- 27 Dániel Marx. Parameterized complexity and approximation algorithms. *The Computer Journal*, 51(1):60–78, 2008.
- 28 Dániel Marx and Igor Razgon. Fixed-parameter tractability of multicut parameterized by the size of the cutset. *SIAM Journal on Computing*, 43(2):355–388, 2014.
- 29 Rolf Niedermeier. *Invitation to Fixed-Parameter Algorithms*. Oxford University Press, 2006.
- 30 George Osipov, Marcin Pilipczuk, and Magnus Wahlström. Parameterized complexity of MinCSP over the point algebra. In *Proc. 32nd Annual European Symposium on Algorithms (ESA-2024)*, volume 308, pages 93:1–93:15, 2024.
- 31 George Osipov and Magnus Wahlström. Parameterized complexity of equality MinCSP. In *Proc. 31st Annual European Symposium on Algorithms (ESA-2023)*, pages 86:1–86:17, 2023.
- 32 Prasad Raghavendra. Optimal algorithms and inapproximability results for every CSP? In *Proc. 40th Annual ACM Symposium on Theory of Computing (STOC-2008)*, pages 245–254, 2008.
- 33 Song Yan. *Number Theory for Computing*. Springer, 2002.