Parameterized Inapproximability of the Minimum Distance Problem over all Fields and the Shortest Vector Problem in all ℓ_p Norms*

Huck Bennett† Mahdi Cheraghchi‡ Venkatesan Guruswami§ João Ribeiro¶

Abstract

We prove that the Minimum Distance Problem (MDP) on linear codes over any fixed finite field and parameterized by the input distance bound is W[1]-hard to approximate within any constant factor. We also prove analogous results for the parameterized Shortest Vector Problem (SVP) on integer lattices. Specifically, we prove that SVP in the ℓ_p norm is W[1]-hard to approximate within any constant factor for any fixed $p > 1$ and W[1]-hard to approximate within a factor approaching 2 for $p = 1$. (We show hardness under randomized reductions in each case.)

These results answer the main questions left open (and explicitly posed) by Bhattacharyya, Bonnet, Egri, Ghoshal, Karthik C. S., Lin, Manurangsi, and Marx (Journal of the ACM, 2021) on the complexity of parameterized MDP and SVP. For MDP, they established similar hardness for *binary* linear codes and left the case of general fields open. For SVP in ℓ_p norms with $p > 1$, they showed inapproximability within *some* constant factor (depending on p) and left open showing such hardness for arbitrary constant factors. They also left open showing W[1]-hardness even of exact SVP in the ℓ_1 norm.

*A preliminary version of this work appeared at STOC 2023 [BCGR23].

†University of Colorado Boulder. huck.bennett@oregonstate.edu. This work was mainly done while the author was at Oregon State University.

‡University of Michigan, Ann Arbor. mahdich@umich.edu.

§University of California, Berkeley. venkatg@berkeley.edu

¶NOVA LINCS and NOVA School of Science and Technology – Universidade Nova de Lisboa. joao.ribeiro@fct.unl.pt. This work was mainly done while the author was at Carnegie Mellon University.

Contents

1 Introduction

Error correcting codes and point lattices are fundamental mathematical objects, and computational problems on them have a wide range of applications in computer science including to robust communication, cryptography, optimization, complexity theory, and more. Indeed, because computational problems on codes and lattices are so ubiquitous, a highly active line of research spanning decades has worked to understand the complexity of the problems themselves. In particular, a great deal of work has studied the complexity of the Minimum Distance Problem (MDP) (and its affine version, the Nearest Codeword Problem (NCP)) on linear error correcting codes [BMvT78, ABSS97, Var97, DMS03, CW12]. Similarly, a large amount of work has studied the complexity of the analogous problems on lattices, the Shortest Vector Problem (SVP) (and its affine version, the Closest Vector Problem (CVP)) [vEB81, ABSS97, Ajt98, Mic00, Kho05, HR12].

In MDP_q, the goal is, given a linear error correcting code $\mathcal{C}$ over a finite field $\mathbb{F}_q$ and a distance bound k as input, to determine whether or not the minimum Hamming weight of a nonzero codeword in $\mathcal{C}$ is at most k. Similarly, in SVP_p the goal is, given a lattice $\mathcal{L}$ and a distance bound k as input, to determine whether or not the ℓ_p norm of some nonzero vector in $\mathcal{L}$ is at most k.[1] One may also consider γ-approximate versions of these problems for $\gamma \geqslant 1$, which we denote by γ-MDP_q and γ-SVP_p, respectively. (In what follows we refer to linear error correcting codes over finite fields simply as "codes." We define codes, lattices, and computational problems on them formally in Sections 2.3 and 2.4.)

In the 1990s, the field of *parameterized complexity*, in which the running time of an algorithm for a given computational problem is considered not just as a function of the problem's input size n but also in terms of some parameter k, developed and matured. The fundamental notion of efficiency in the study of parameterized algorithms is *fixed-parameter tractability*, which means that the algorithm runs in time $f(k) \cdot \mathrm{poly}(n)$ for some (possibly fast-growing) function $f(k)$ depending on the parameter k but not the input length. A computational problem (formally, problem-parameter pair) with such an algorithm is called *fixed-parameter tractable* (FPT), and the set of all such problems forms the complexity class FPT. On the other hand, the canonical notion of inefficiency for parameterized problems is $\mathsf{W}[1]$-hardness, which is analogous to NP-hardness in the non-parameterized setting. To show $\mathsf{W}[1]$-hardness of a given problem, it suffices to give an *FPT reduction* from a known $\mathsf{W}[1]$-hard problem to that problem. Giving such a reduction in particular implies that the problem reduced to is not in FPT unless $\mathsf{W}[1] = \mathsf{FPT}$, which is widely conjectured not to be the case. (Determining whether FPT is equal to $\mathsf{W}[1]$ is a major open question, and is the analog of the P versus NP question in the parameterized world.) See the books by Downey and Fellows [DF99, DF13] for comprehensive references on parameterized complexity.

Parameterized complexity of MDP **and** SVP**.** As part of the development of parameterized complexity as a whole, substantial interest arose in the parameterized complexity (specifically, $\mathsf{W}[1]$-hardness) of computational problems on codes and lattices. This was especially true for MDP and SVP, where in each case the parameter k of interest is the input distance bound.[2] Indeed, until recently, one of the major unresolved questions in parameterized complexity theory was to determine whether the Minimum Distance Problem on binary codes was $\mathsf{W}[1]$-hard. It was one of the few remaining open problems from [DF99], and Downey and Fellows called it one of the "most

[1] The ℓ_p norm used is fixed and independent of the input. One may also consider SVP with respect to arbitrary norms, but it is most commonly considered with respect to ℓ_p norms (and especially with respect to the Euclidean norm ℓ_2) as is the case in this work.

[2] In the parameterized setting, we consider SVP only on *integer* lattices; otherwise the distance bound is not meaningful.

infamous" such open problems in their follow-up book [DF13].[3] Similarly, the fixed-parameter (in)tractability of the Shortest Vector Problem in the ℓ_2 norm was mentioned as an important unresolved question in [DF99, DF13]. Interestingly, it is not known whether γ-MDP$_q$ nor γ-SVP$_p$ are in W[1].

In recent seminal work, Bhattacharyya, Bonnet, Egri, Ghoshal, Karthik C. S., Lin, Manurangsi, and Marx [BBE+21], building on work of Lin [Lin18], resolved both of these questions in the affirmative. They in fact even showed that both parameterized MDP and SVP are hard to approximate. Specifically, they showed that for any constant $\gamma \geqslant 1$, γ-MDP$_2$ is W[1]-hard to approximate under randomized reductions, and that for any $p > 1$ and constant γ satisfying $1 \leqslant \gamma < (1/2 + 1/2^p)^{-1/p}$, γ-SVP$_p$ is W[1]-hard to approximate under randomized reductions.

However, despite its major achievements, [BBE+21] still fell short of providing a complete understanding of the parameterized hardness of approximate MDP and SVP. To that end, they gave several open questions. Specifically, the authors asked whether it was possible to show W[1]-hardness of MDP over *all* finite fields $\mathbb{F}_q$ (and not just for the binary case of $\mathbb{F}_2$). They also asked about showing W[1]-hardness of SVP in *all* ℓ_p norms (specifically, they asked about γ-SVP$_1$, for which they did not show hardness even in the exact case of $\gamma = 1$), and about showing W[1]-hardness of γ-SVP$_p$ with arbitrarily large constant γ for (some) p (they did not obtain such a result for any p).[4] The first two of these three questions from [BBE+21] were also asked as Open Questions 2 and 3, respectively, in a recent survey on approximation in parameterized complexity by Feldmann, Karthik C. S., Lee, and Manurangsi [FKLM20], which discussed important open problems in the field as a whole.

1.1 Our contributions

In this work, we answer all three of the open questions of [BBE+21] discussed above, and provide a nearly complete picture of the parameterized inapproximability of MDP and SVP. In each of the three theorems below (i.e., Theorems 1.1 to 1.3) the parameter of interest is the input distance bound k.

We first give our hardness result for MDP, which resolves the first open question from [BBE+21] (also asked as [FKLM20, Open Question 2]).

Theorem 1.1. *For any fixed prime power q and constant $\gamma \geqslant 1$, γ-MDP$_q$ is W[1]-hard under randomized FPT reductions with two-sided error.*

Second, we settle the second open question from [BBE+21] (also asked as [FKLM20, Open Question 3]) by showing the following hardness result for parameterized γ-SVP$_p$ for all (finite) $p \geqslant 1$ and some $\gamma = \gamma(p)$.[5] Indeed, in particular applies to the ℓ_1 norm. It also shows hardness of approximation for larger factors $\gamma(p)$ for $p > 1$ than [BBE+21] does.

Theorem 1.2. *For any fixed $p \in [1, \infty)$ and constant $\gamma \in [1, 2^{1/p})$, γ-SVP$_p$ is W[1]-hard under randomized FPT reductions with two-sided error.*

[3] More precisely, [DF99, DF13] asked about the complexity of the Even Set Problem, which is equivalent to the dual formulation of the Minimum Distance Problem over $\mathbb{F}_2$.

[4] In fact, [BBE+21] asked about such a result for $p \neq 2$ and claimed such a result in passing for $p = 2$. However, the claim was referring to a result from prior work (specifically, [BGKM18]) that showed hardness only under stronger hypotheses. See Remark 1.4.

[5] We do not consider the case of $p = \infty$ because, as [BBE+21] notes, SVP in the ℓ_∞ norm is NP-hard even when $k = 1$.

Finally, we establish the parameterized inapproximability of SVP with an arbitrary constant approximation factor in the ℓ_p norm for all fixed $p > 1$. This resolves the third question from [BBE+21] mentioned above.

Theorem 1.3. *For any fixed $p \in (1, \infty)$ and constant $\gamma \geqslant 1$, γ-SVP$_p$ is $\mathsf{W}[1]$-hard under randomized FPT reductions with two-sided error.*

Remark 1.4. We note that [BBE+21] erroneously claimed in a passing remark that the important Euclidean (i.e., $p = 2$) special case of Theorem 1.3 was already known. However, that remark was in fact referring to a result from an earlier version of [BBE+21] (i.e., [BGKM18]) that shows parameterized hardness of γ-SVP$_2$ for arbitrary constant $\gamma \geqslant 1$, *but only under the (randomized) Gap Exponential Time Hypothesis (Gap-ETH) or the Parameterized Inapproximability Hypothesis (PIH)*, which are stronger assumptions than $\mathsf{W}[1] \neq \mathsf{FPT}$.[6] In particular, the result in Theorem 1.3 was previously unknown for any p. We thank Pasin Manurangsi [Man22] for clarifying this for us.

We provide a technical overview of our arguments in Section 1.2 and provide formal proofs of Theorems 1.1 to 1.3 in Sections 3 to 5, respectively.

Fine-grained hardness of parameterized MDP **and** SVP**.** Our reductions also directly yield improved results concerning the fine-grained hardness of γ-MDP$_q$ and γ-SVP$_p$ under Gap-ETH. Leveraging results from [BGKM18, BBE+21], Manurangsi [Man20] showed that there are no (possibly randomized) algorithms running in time $f(k) \cdot n^{o(k)}$ for γ-NCP$_q$ (respectively, time $f(k) \cdot n^{o(k^p)}$ for γ-CVP$_p$) with any function f, $\gamma \geqslant 1$, and prime power q (respectively, $p \geqslant 1$), where n is the dimension of the input code (respectively, the rank of the input lattice) and k is the input distance bound (in each case) assuming randomized Gap-ETH.

By inspection, our FPT reductions from approximate NCP$_q$ to approximate MDP$_q$ and from approximate CVP$_p$ to approximate SVP$_p$ in Sections 3 and 4, respectively, transform the distance parameter k into $k' = O(k)$ (for formal statements, see Theorems 3.1 and 4.1). Therefore, we can combine these reductions with the results from [Man20] to immediately obtain the following results on the parameterized fine-grained hardness of MDP and SVP, which imply that the brute force solution to each of these problems is essentially optimal under randomized Gap-ETH.[7]

Theorem 1.5. *For any function f, fixed prime power q, and every $\gamma \in \left[1, \frac{2q}{2q-1}\right)$ it holds that, assuming randomized Gap-ETH, there is no randomized algorithm running in time $f(k) \cdot n^{o(k)}$ for deciding γ-MDP$_q$, where n is the dimension of the input code and k is the input distance bound.*

[6] Randomized Gap-ETH [Din16, MR17] states that there exist constants $\varepsilon, c > 0$ such that no randomized algorithm which is given as input a 3-CNF formula F with m clauses and runs in time $O(2^{cm})$ can distinguish with probability at least $2/3$ between the cases where F is satisfiable and where only at most a $(1 - \varepsilon)$-fraction of clauses in F are satisfiable.
PIH [LRSZ20] states that there exists a constant $\varepsilon > 0$ such that it is $\mathsf{W}[1]$-hard to approximate the Multicolored Densest Subgraph problem to within a $\gamma = 1 + \varepsilon$ approximation factor. This corresponds to the problem where we are given as input a graph $G = (V, E)$ with the vertex set partitioned into k sets $V_1, \ldots, V_k$, and the goal is to select vertices $v_1 \in V_1, \ldots, v_k \in V_k$ that induce as many edges as possible in G.

[7] We note that we work with the standard (in the non-parameterized setting) formulation of γ-SVP$_p$ throughout the paper, where the goal is to decide whether the input lattice has a nonzero vector $\boldsymbol{x}$ with $\|\boldsymbol{x}\|_p \leqslant k$ or if all such vectors have ℓ_p norm greater than γk. On the other hand, [BBE+21, Man20] work with an equivalent but different parameterization of the problem, which asks whether the input lattice has a nonzero vector $\boldsymbol{x}$ with $\|\boldsymbol{x}\|_p^p \leqslant k$ or if the pth power of the ℓ_p norm of all such vectors is greater than γk. This discrepancy leads to certain runtimes and approximation factors in our work being off by a power of p from [BBE+21, Man20].

Theorem 1.6. *For any function f, fixed real number $p \geqslant 1$, and every $\gamma \in [1, 2^{1/p})$ it holds that, assuming randomized Gap-ETH, there is no randomized algorithm running in time $f(k) \cdot n^{o(k^p)}$ for deciding γ-SVP$_p$, where n is the rank of the input lattice and k is the input distance bound.*

Previously, Theorem 1.5 was only known to hold for $q = 2$, and Theorem 1.6 was only known to hold for $p > 1$ and with approximation factors $\gamma = \gamma(p) < (1/2 + 1/2^p)^{-1/p}$ that are smaller than those achieved by Theorem 1.6; see [BGKM18, Man20].

Interestingly, the standard technique of tensoring instances of MDP or SVP to boost the approximation factor cannot be used to prove fine-grained hardness results as above, because the distance parameter k is mapped to $k' = k^c$ for $c > 1$. This motivates the search for FPT reductions that preserve the parameter k up to a linear factor (i.e., for which $k' = O(k)$) while simultaneously showing hardness for as large an approximation factor γ as possible. Our pre-tensoring hardness reductions for MDP$_q$ and SVP$_p$ in Sections 3 and 4 also yield $k' = O(k)$, and it would be interesting to come up with improved reductions that achieve larger approximation factors. Moreover, we note that although we obtain better W[1]-hardness of approximation for SVP$_p$ with $p > 1$ from the reduction in Section 5, we in fact get better fine-grained hardness from our reduction in Section 4. (The reduction in Section 4 also has the advantage of showing hardness of SVP in the ℓ_1 norm.)

1.2 Technical overview

1.2.1 Parameterized inapproximability of γ-MDP$_q$

Inapproximability results for MDP and SVP follow the blueprint originally pioneered by Ajtai [Ajt98], Micciancio [Mic00], and Khot [Kho05] for lattices and Dumer, Micciancio, and Sudan [DMS03] for codes. In each case, the idea is to reduce the affine versions of the problems (NCP and CVP, respectively), for which NP-hardness results were long known, to the linear versions (MDP and SVP, respectively).

The DMS reduction from NCP **to** MDP**.** We start by illustrating the Dumer-Micciancio-Sudan (DMS) reduction from NCP to MDP, which is based on analogous reductions of Ajtai [Ajt98] and Micciancio [Mic00] from CVP to SVP. An instance of NCP consists of a linear code $\mathcal{C} = \mathcal{C}(G) \subset \mathbb{F}_q^m$ generated by a matrix $G \in \mathbb{F}_q^{m \times n}$ and a target $\boldsymbol{t} \in \mathbb{F}_q^m$, and the goal is to minimize the distance $\mathrm{dist}(\boldsymbol{t}, \mathcal{C})$ of $\boldsymbol{t}$ to its closest codeword, i.e., the minimum Hamming weight of $G\boldsymbol{x} - \boldsymbol{t}$ over all $\boldsymbol{x} \in \mathbb{F}_q^n$, where the Hamming weight of a vector $\boldsymbol{v}$ is $\|\boldsymbol{v}\|_0 = |\{i : \boldsymbol{v}_i \neq 0\}|$. A natural reduction to MDP will produce the instance $\mathcal{C}' = \mathrm{span}(\mathcal{C}, \boldsymbol{t})$ generated by $G' = (G \mid \boldsymbol{t}) \in \mathbb{F}_q^{m \times (n+1)}$. If we restrict to codewords $G\boldsymbol{x} + \beta\boldsymbol{t}$ of $\mathcal{C}'$ that use the target in the combination, i.e., have $\beta \neq 0$, then the minimum distance of such a codeword equals the Hamming distance $\mathrm{dist}(\boldsymbol{t}, \mathcal{C})$ of $\boldsymbol{t}$ to $\mathcal{C}$. Under this (unreasonable) restriction we have a reduction that preserves the objective value. The obvious trouble though is that $\mathcal{C}$ (and hence $\mathcal{C}'$) might have short codewords of weight much smaller than $\mathrm{dist}(\boldsymbol{t}, \mathcal{C})$. In this case, the minimum distance of $\mathcal{C}'$ will equal the distance of $\mathcal{C}$, and have nothing to do with $\boldsymbol{t}$. Note, however, that this reduction *does* work if $\lambda(\mathcal{C}) > k$, where $\lambda(\mathcal{C}) = \min_{\boldsymbol{c} \in \mathcal{C} \setminus \{\boldsymbol{0}\}} \|\boldsymbol{c}\|_0$ is the minimum Hamming weight of $\mathcal{C}$. Further, starting from a gap-γ version of NCP asking if $\mathrm{dist}(\boldsymbol{t}, \mathcal{C}) \leqslant k$ or $\mathrm{dist}(\boldsymbol{t}, \mathcal{C}) > \gamma k$, we would get hardness of a gap-γ version of MDP if $\lambda(\mathcal{C}) > \gamma k$.

A natural goal is therefore to increase the distance of $\mathcal{C}$ without increasing the proximity parameter in NCP by too much. This was achieved in [DMS03] by encoding the message according to $\mathcal{C}$ as well as a second code $\tilde{\mathcal{C}} \subset \mathbb{F}_q^{m'}$ with generator matrix $\tilde{G} \in \mathbb{F}_q^{m' \times n'}$ with large distance, say D. Further, $\tilde{\mathcal{C}}$ will be a *locally dense code* in the sense that one can find a "bad list decoding configuration" comprising a center $\boldsymbol{s} \in \mathbb{F}_q^{m'}$ that has a large number of codewords of $\tilde{\mathcal{C}}$ within

distance αD for some $\alpha < 1$; we call α the *relative radius* of the locally dense code. (One can in fact efficiently construct such locally dense codes with any constant relative radius $\alpha > 1/2$ using randomness [DMS03].) The number of codewords will be so large that one can sample a linear map T that with high probability projects these codewords *onto* $\mathbb{F}_q^n$. If $\tilde{G}$ is the generator matrix of the locally dense code $\tilde{\mathcal{C}}$, the reduction, which will use randomness to pick both the center $\boldsymbol{s}$ and the projection T, will produce the instance of MDP generated by

$$\begin{pmatrix} GT\tilde{G} & \boldsymbol{t} \\ \tilde{G} & \boldsymbol{s} \end{pmatrix}. \tag{1}$$

The completeness of the reduction follows because for any $\boldsymbol{x} \in \mathbb{F}_q^n$ that might satisfy $\|G\boldsymbol{x} - \boldsymbol{t}\|_0 \leqslant k$, there will be a codeword $\tilde{G}\boldsymbol{y} \in \tilde{\mathcal{C}}$ within distance αD from $\boldsymbol{s}$ that projects to $\boldsymbol{x}$ under T. Thus multiplying the generator matrix in Equation (1) by $(\boldsymbol{y}^T, -1)^T$ will yield a nonzero codeword of weight at most $k + \alpha D$. Since the distance of $\tilde{\mathcal{C}}$ is D, codewords which don't use the last column of Equation (1) will have Hamming weight at least D. If $\alpha D + k < D$, which is possible to ensure provided $\alpha < 1$, we get a gap.

Challenges in the FPT setting. It is reasonable to wonder whether the DMS reduction above works directly in the FPT setting. However, as already pointed out in [BBE+21, Section 2.1], one quickly runs into some obstacles. Indeed, the locally dense codes used in [DMS03] have minimum distance D which depends on the input code dimension, and this is necessary to ensure that we can sample the linear map T with the desired properties. This is because the existence of T implies that there must be at least $|\mathbb{F}_q^n| = q^n$ codewords in $\tilde{\mathcal{C}}$ of Hamming weight at most αD. Since the distance threshold of the resulting MDP instance is $k' = \alpha D + k$, it follows that k' depends on the input code dimension n, and so the DMS reduction is not FPT.

To overcome these issues, [BBE+21] modify both the problem they reduce from as well as the reduction itself. First, instead of reducing from NCP to MDP, they reduce from a variant of NCP they call the *Sparse Nearest Codeword Problem* (SNCP), where the Hamming weight of the coefficient vector realizing the nearest codeword is also taken into account. More precisely, the objective function $\mathrm{dist}(\boldsymbol{t}, \mathcal{C})$ of NCP is replaced by

$$\min_{\boldsymbol{x} \in \mathbb{F}_q^n} (\|G\boldsymbol{x} - \boldsymbol{t}\|_0 + \|\boldsymbol{x}\|_0),$$

where $\mathcal{C} = \mathcal{C}(G)$. It is not hard to reduce NCP to SNCP in the FPT setting, and this allows [BBE+21] to avoid having to sample the linear map T. Second, they replace locally dense codes by another variant which they call *locally suffix dense codes* (LSDCs). These are codes $\tilde{\mathcal{C}} \subseteq \mathbb{F}_q^{m'}$ with minimum distance D and generator matrix

$$\begin{pmatrix} I_n & 0 \\ \tilde{G}_1 & \tilde{G}_2 \end{pmatrix} \in \mathbb{F}_q^{m' \times n'}$$

which have the property that, given any prefix $\boldsymbol{p} \in \mathbb{F}_q^n$, for most "suffix centers" $\boldsymbol{s} \in \mathbb{F}_q^{m'-n}$ there is a suffix $\boldsymbol{u} \in \mathbb{F}_q^{m'-n}$ within Hamming distance αD of $\boldsymbol{s}$ such that $(\boldsymbol{p}, \boldsymbol{u}) \in \tilde{\mathcal{C}}$. With the help of these notions, [BBE+21] consider the MDP instance generated by

$$\begin{pmatrix} G & 0 & \boldsymbol{t} \\ I_n & 0 & 0 \\ \tilde{G}_1 & \tilde{G}_2 & \boldsymbol{s} \end{pmatrix},$$

where $\boldsymbol{s}$ is sampled uniformly at random from $\mathbb{F}_q^{m'-n}$. The proof that this reduction works is similar to the one sketched above for the DMS reduction. The main challenge is to efficiently construct LSDCs with appropriate parameters, in particular with minimum distance D *independent* of m', n', and n. Unfortunately, known constructions of locally dense codes do not yield LSDCs with the desired properties. In the binary setting $q = 2$, [BBE+21] showed that one can take $\tilde{C}$ to be a binary BCH code [Hoc59, BR60] with design minimum distance D. This ingenious approach allows them to prove that γ-MDP$_2$ is W[1]-hard.

It is instructive to discuss more precisely why the choice of binary BCH codes as LSDCs works, and why it cannot be extended to other finite fields. Binary BCH codes with minimum distance D have codimension $\approx \lfloor \frac{D-1}{2} \rfloor \log(m'+1)$. The crucial fact that makes the counting analysis of [BBE+21] go through is that $\lfloor \frac{D-1}{2} \rfloor$ is also the *unique decoding radius* for the binary BCH code, i.e., Hamming balls of this radius centered on BCH codewords are disjoint. In other words, binary BCH codes almost meet the sphere packing bound. One would hope that replacing binary BCH codes with q-ary BCH codes would suffice to show W[1]-hardness of γ-MDP$_q$ more generally. However, q-ary BCH codes with minimum distance D have codimension $\approx \lfloor (D-1)(1-1/q) \rfloor \log_q(m'+1)$ (see Theorem 3.4), *while the unique decoding radius remains* $\lfloor \frac{D-1}{2} \rfloor$. Put differently, q-ary BCH codes for $q > 2$ are no longer close to the sphere packing bound, which breaks the analysis from [BBE+21]. In fact, for $q > 2$, it is not known if there exist q-ary codes with rate vs. distance trade-off close to the sphere packing bound. Therefore, it seems challenging to make the approach from [BBE+21] work as is over $\mathbb{F}_q$, for $q > 2$.

Our approach: Khot for codes. We succeed in overcoming the barriers that [BBE+21] faced and establish the W[1]-hardness of γ-MDP$_q$ for arbitrary finite fields $\mathbb{F}_q$ via a different and arguably simpler (direct) reduction from NCP to MDP. Our key insight is to adapt Khot's reduction [Kho05] from CVP to SVP to the coding-theoretic setting. We are able to meet the requirements of such a reduction with locally dense codes constructed from q-ary BCH codes.

This approach is quite natural. In fact, early lecture notes of Khot [Kho04] showed how to use this strategy to reduce from NCP to MDP in the special case of binary codes. Our reduction is more general and requires more careful analysis in that it works with arbitrary locally dense codes with constant relative radius $\alpha < 1$, works over $\mathbb{F}_q$ and not just $\mathbb{F}_2$, and requires a more careful analysis of the the distance bound k' in the output MDP instance as a function of the distance bound k in the input NCP instance. However, the core idea is the same.

More precisely, given an instance $(G, \boldsymbol{t}, k)$ of γ-NCP$_q$ with $G \in \mathbb{F}_q^{m \times n}$ and $\boldsymbol{t} \in \mathbb{F}_q^m$ and an appropriate locally dense code $(\tilde{G}, \boldsymbol{s})$ with $\tilde{G} \in \mathbb{F}_q^{m' \times n'}$ and $\boldsymbol{s} \in \mathbb{F}_q^{m'}$, we consider the intermediate code $\mathcal{C}_{\text{int}}$ spanned by the generator matrix

$$G_{\text{int}} = \begin{pmatrix} G & 0 & -\boldsymbol{t} \\ 0 & \tilde{G} & -\boldsymbol{s} \end{pmatrix}.$$

This is analogous to the intermediate lattice introduced in Khot's reduction [Kho05] from CVP to SVP, with the difference being that we replace the CVP instance by an NCP instance and the locally dense *lattice* by a locally dense *code*. Note that it may happen that $\mathcal{C}_{\text{int}}$ contains low weight vectors even when $(G, \boldsymbol{t}, k)$ is a NO instance of γ-NCP$_q$. This is, however, not a show-stopper, as it in fact suffices to show that there are *many more* low weight vectors in $\mathcal{C}_{\text{int}}$ when $(G, \boldsymbol{t}, k)$ is a YES instance of γ-NCP$_q$ than when $(G, \boldsymbol{t}, k)$ is a NO instance. Indeed, if this holds then we can *sparsify* $\mathcal{C}_{\text{int}}$ by intersecting it with an appropriate random code $\mathcal{C}_{\text{rand}}$ so that, with high probability, all low weight vectors are eliminated in the NO case, but at least one low weight vector survives in the YES case. Again, this is analogous to the lattice sparsfication performed in Khot's

reduction [Kho05], Finally, the γ'-MDP$_q$ instance is obtained by computing a generator matrix G_{final} of $\mathcal{C}_{\text{final}} = \mathcal{C}_{\text{int}} \cap \mathcal{C}_{\text{rand}}$ and outputting (G_{final}, k') for some appropriate k'.

To guarantee that the reduction is FPT, we need to ensure that $k' \leqslant f(k)$ for some function f. In fact, in our reduction k only increases by a linear factor, i.e., we get $k' \leqslant f(k) = O(k)$. We briefly sketch how to establish the desired properties of $\mathcal{C}_{\text{int}}$ and choose k'. Suppose that $(\tilde{G}, \boldsymbol{s})$ is a locally dense code with minimum distance D and such that there are at least N vectors $\boldsymbol{y}$ satsifying $\|\tilde{G}\boldsymbol{y} - \boldsymbol{s}\|_0 \leqslant \alpha D$ for some $\alpha \in (1/2, 1)$. If $(G, \boldsymbol{t}, k)$ is a YES instance of γ-NCP$_q$, i.e., there exists $\boldsymbol{x}$ such that $\|G\boldsymbol{x} - \boldsymbol{t}\|_0 \leqslant k$, then multiplying G_{int} by $(\boldsymbol{x}, \boldsymbol{y}, 1)^T$ yields a codeword of weight at most $k' = \alpha D + k$. As a result, there are at least N vectors in $\mathcal{C}_{\text{int}}$ of weight at most k', which we call *good*. On the other hand, if $(G, \boldsymbol{t}, k)$ is a NO instance of γ-NCP$_q$ and $D > \gamma k$, then it is not hard to see that every codeword of weight at most γk in $\mathcal{C}_{\text{int}}$ is of the form $G_{\text{int}}(\boldsymbol{x}, \boldsymbol{0}, 0)^T$, and so there are at most q^n such *annoying* vectors in $\mathcal{C}_{\text{int}}$, where $n = \dim(\mathcal{C}(G))$. (The "good" versus "annoying" vectors terminology was also introduced in [Kho05].)

We conclude that the reduction works and is FPT if we are able to construct a q-ary locally dense code $(\tilde{G}, \boldsymbol{s})$ as above under the constraints that (i) $D = g(k) > \gamma k$, (ii) $k' = \alpha D + k \ll \gamma k$, and (iii) $N \gg q^n$ (so that there many more good vectors in YES instances than annoying vectors in NO instances and the sparsification step works). While the approach of [BBE+21] described above required q-ary codes of codimension $\approx \frac{D}{2} \log_q(m')$, which are not known to exist for $q > 2$, we show that to construct locally dense codes satisfying our constraints it is enough to consider q-ary codes with minimum distance $D \approx \gamma k$ and codimension $\approx \beta D \log_q(m')$ for *any* $\beta < 1$! Therefore, we can use q-ary BCH codes of length $m' = \text{poly}(m)$ with design minimum distance $D \approx \gamma k$, which have codimension $\approx D(1 - 1/q) \log_q(m')$ for any prime power q.

This approach shows W[1]-hardness of γ'-MDP$_q$ for some approximation factor $\gamma' > 1$; in fact, we get hardness with $\gamma' \approx 1/\beta \approx 1/\alpha$. We can then amplify this approximation factor γ' in a standard manner via *tensoring* to obtain W[1]-hardness of γ''-MDP$_q$ for every $\gamma'' \geqslant 1$. For more details, see Section 3.

1.2.2 Parameterized inapproximability of γ-SVP$_p$

We first define locally dense lattices, which are analogous objects to locally dense codes, and which are important both for understanding the issues with [BBE+21] and our ways of handling them. A *locally dense lattice* (with respect to the ℓ_p norm) is a lattice $\mathcal{L} \subset \mathbb{R}^m$ together with a shift $\boldsymbol{s} \in \mathbb{R}^m$ such that $\mathcal{L} - \boldsymbol{s}$ contains many vectors of ℓ_p norm at most $\alpha\lambda_1^{(p)}(\mathcal{L})$ for some constant $\alpha = \alpha(p) \in (1/2, 1)$, where $\lambda_1^{(p)}(\mathcal{L}) = \min_{\boldsymbol{v} \in \mathcal{L} \setminus \{\boldsymbol{0}\}} \|\boldsymbol{v}\|_p$. As is the case for locally dense codes, we call α the *relative radius* of the corresponding locally dense lattice.

As in Section 1.2.1, we start by explaining the original approach from [BBE+21] towards showing W[1]-hardness of γ-SVP$_p$ for $p > 1$ and some approximation factor $\gamma > 1$, and why it fails to resolve the problems we tackle. To recall, [BBE+21] proved that γ-CVP$_p$ is W[1]-hard for every fixed $p, \gamma \geqslant 1$. Then, they simply noted that Khot's initial reduction [Kho05] from CVP$_p$ to SVP$_p$ (which is similar to our FPT reduction from NCP to MDP discussed in Section 1.2.1) is itself an FPT reduction if the parameters of the locally dense lattice from [Kho05] (which is based on binary BCH codes) are chosen appropriately. Combining this observation with the W[1]-hardness of γ-CVP$_p$ immediately yields that γ'-SVP$_p$ is W[1]-hard for some approximation factor $\gamma' = \gamma'(p) > 1$. However, despite achieving this nice result, the approach of [BBE+21] has two significant shortcomings.

Showing inapproximability of SVP in all ℓ_p norms (including ℓ_1). The first limitation of the

approach in [BBE+21] is the use of Khot's locally dense lattices, which do not suffice to show either NP- or W[1]-hardness of SVP in the ℓ_1 norm. More specifically, Khot's locally dense lattices have relative radius $\alpha = \alpha(p) > (1/2 + 1/2^p)^{1/p}$, which suffices to show W[1]-hardness of parameterized γ-SVP$_p$ (and NP-hardness of non-parameterized γ-SVP$_p$) for any

$$\gamma' = \gamma'(p) < 1/\alpha(p) < (1/2 + 1/2^p)^{-1/p} ,$$

and no better. Plugging $p = 1$ into the right-hand side of this equation shows that Khot's reduction does not yield hardness even for exact SVP$_1$ (i.e., for γ'-SVP$_1$ with $\gamma' = 1$). Indeed, this issue is what kept Khot's reduction from showing NP-hardness of SVP$_1$ in [Kho05] and what kept [BBE+21] from showing W[1]-hardness of SVP$_1$.

Despite Khot's reduction not working, other reductions nevertheless showed NP-hardness of (even approximating) SVP$_1$. Unfortunately, as [BBE+21] notes, these reductions fail both because of their use of non-integral lattices and the fact that rounding real-valued lattice bases to integral ones in a black-box way amounts to a non-FPT reduction, since the minimum distance of the resulting lattices will depend on their dimension. (Multiplying rational lattice bases by the least common multiple of their entries' denominators causes a similar problem.) First, Micciancio [Mic00] showed hardness of γ-SVP$_1$ for any $\gamma < 2$ using locally dense lattices constructed from prime number lattices. However, these locally dense lattices are non-integral and even non-rational.[8] Second, Regev and Rosen [RR06] showed how to use efficiently computable linear norm embeddings to reduce γ-SVP$_2$ to γ'-SVP$_p$ for any $p \geqslant 1$ and any constant $\gamma' < \gamma$. Combined with Khot's work [Kho05], which showed NP-hardness of approximating SVP$_2$ to within any constant factor γ, [RR06] implies that SVP$_p$ for any p (and in particular, SVP$_1$) is NP-hard to approximate within any constant factor as well. However, the norm embeddings given in [RR06] use random Gaussian projection matrices, and therefore output non-integral lattices. Moreover, using different, integral distributions for the projection matrices also does not obviously work.

We overcome this first issue of Khot's locally dense lattices not working in the ℓ_1 norm by instantiating Khot's reduction with different locally dense lattices. Specifically, we instantiate Khot's reduction with the locally dense lattices constructed in recent work of Bennett and Peikert [BP23], which are built from Reed-Solomon codes. These locally dense lattices meet all of the requirements necessary for the proof of Theorem 1.2. Namely, they are efficiently constructible; their base lattices $\mathcal{L}$ are integral; they can be constructed so that $\lambda_1^{(p)}(\mathcal{L})$ does not depend on the dimension of the input lattice $\mathcal{L}$; and for $p \in [1, \infty)$ they have ℓ_p relative radius $\alpha(p) \approx 1/2^p < 1$. In particular, they have ℓ_1 relative radius $\alpha(1) \approx 1/2$ (which is essentially optimal by the triangle inequality), and so Khot's reduction shows hardness of γ-SVP$_1$ for any constant $\gamma < 2$. (We again note that the largest approximation factor $\gamma = \gamma(p)$ for which Khot's reduction shows parameterized hardness of γ-SVP$_p$ is $\gamma \approx 1/\alpha$, where $\alpha = \alpha(p)$ is the relative radius of the locally dense lattice used, and this is where the bound on the approximation factor $\gamma = \gamma(p)$ in Theorem 1.2 comes from.)

Showing inapproximability of γ-SVP$_p$ for all $p > 1$ and all γ. The second main shortcoming of the approach in [BBE+21] is that it is not clear how to amplify the approximation factor $\gamma > 1$ for which they get W[1]-hardness of γ-SVP$_p$ (for $p > 1$), to an arbitrary constant. As in the case of codes, the natural thing to try for amplifying hardness is to take the *tensor product* of the input SVP instance with itself. The idea of tensoring is, given an instance (B, k) of SVP as input, to output the SVP instance $(B \otimes B, k^2)$, where $B \otimes B$ is the Kronecker product of the input basis

[8] We note in passing that Micciancio did in fact carefully analyze rounding these locally dense lattices to get integral ones, but emphasize again that this rounding causes the minimum distance of the resulting lattices to depend on their dimension.

matrix B with itself. Unfortunately, unlike for codes, tensoring does not work in general for lattices. Indeed, although it always holds that $\lambda_1(\mathcal{L}(B) \otimes \mathcal{L}(B)) \leqslant \lambda_1(\mathcal{L}(B))^2$, the converse is not always true or even "close to true"; see, e.g., [HR12, Lemma 2.3].

Although Haviv and Regev [HR12] showed that Khot's original SVP_2 instances have properties that *do* in fact allow them to tensor nicely, this is *not the case* for the SVP instances obtained in [BBE+21]. Indeed, the (crucial!) subtlety is that the standard NP-hardness proof for approximate CVP_p proceeds via a reduction from approximate *Exact Set Cover* [ABSS97, Kho05, HR12], and the resulting CVP_p instances enjoy important additional properties that are then inherited by the SVP_p instances in [Kho05]. Parameterized inapproximability of Exact Set Cover is known under the (randomized) Gap-ETH and PIH assumptions, and this is what allowed [BGKM18] to show parameterized hardness of γ-SVP_2 for any constant $\gamma \geqslant 1$; see also Remark 1.4. However, it is not currently known whether approximate Exact Set Cover is W[1]-hard, and so [BBE+21] generate their CVP_p instances via a different reduction from (the dual version of) NCP_q with a suitably large prime q instead.[9] As a result, important properties no longer hold when [BBE+21] use these alternative CVP_p instances to create SVP_p instances via Khot's reduction. Namely, it is no longer true that every lattice vector with at least one odd coordinate has large Hamming weight, a property that is needed to ensure that the SVP_p instance tensors nicely in [HR12].

It is also sensible to wonder whether Khot's *augmented tensor product* [Kho05], which he introduced in his original work to overcome issues with tensoring, can nevertheless be used to boost the approximation factor of the SVP_p instances generated in [BBE+21]. However, the augmented tensor product cannot be applied in the FPT setting unless the short lattice vectors in the base SVP instances also have have short coefficient vectors (i.e., coefficient vectors whose ℓ_p norm is independent of lattice dimension). The SVP instances in [BBE+21] do not seem to have this property.

Our solution. In order to construct W[1]-hard SVP instances that tensor nicely and thereby prove Theorem 1.3, we give a reduction directly from approximate NCP_2 to approximate SVP_p for any $p > 1$.[10] Our reduction is a variant of the reductions in Khot [Kho05] and Haviv and Regev [HR12], and again we instantiate the reduction with locally dense lattices constructed from binary BCH codes similar to those used by Khot [Kho05, BBE+21]. We emphasize that although the proofs of Theorems 1.2 and 1.3 both use variants of Khot's reduction, the key to proving Theorem 1.2 was to instantiate Khot's reduction with different locally dense lattices and the key to Theorem 1.3 was to reduce from a different W[1]-hard problem. Moreover, ensuring that the characteristic of the underlying codes in the NCP instances that we reduce from matches that of the underlying BCH codes in the locally dense lattices that we use seems essential for our analysis. Indeed, our NCP instances and locally dense lattices both use codes over $\mathbb{F}_2$, whereas [BBE+21] reduced from NCP instances over $\mathbb{F}_q$ for larger prime q.

Our reduction allows us to construct γ-SVP_2 instances with some constant $\gamma > 1$ that meet the sufficient conditions given in [HR12] to be amplified to γ'-SVP_2 instances for arbitrarily large constant γ'. These conditions roughly say that the base lattices $\mathcal{L}$ in the SVP instance must be such that all vectors $\boldsymbol{v} \in \mathcal{L} \subseteq \mathbb{Z}^n$ satisfy at least one of the following: (1) $\boldsymbol{v}$ has Hamming weight at least d for some distance bound d, (2) $\boldsymbol{v} \in 2\mathbb{Z}^n$ and $\boldsymbol{v}$ has Hamming weight at least $d/4$, or (3) $\boldsymbol{v} \in 2\mathbb{Z}^n$ and $\boldsymbol{v}$ has very high ℓ_2 norm. The minimum distances of lattices in which all vectors satisfy either conditions (1) or (2) behave nicely under tensoring, but condition (3) makes the analysis subtle. However, we are essentially able to rely on the analysis in [HR12]. Moreover, modifying the analysis

[9] The exact version of this problem is known to be W[1]-hard, see [CFK+15, Section 13.6.3].

[10] We could also use CVP as an intermediate problem in the reduction as is done in [BBE+21], but that does not obviously make the reduction simpler or more modular.

in [HR12] a bit additionally allows us to extend our result to ℓ_p norms for $p > 1$. (The omission of $p = 1$ is yet again for the same reason as in [Kho05, BBE+21]; it is because of the binary BCH-code-based locally dense lattices that we use.)

1.3 Additional related work

Interest in the complexity of computational problems on codes and lattices more broadly goes back several decades. We survey the most closely related work here.

Complexity of NCP **and** MDP**.** Berlekamp, McEliece, and van Tilborg [BMvT78] showed that certain problems related to linear codes, such as the exact version of NCP, are NP-hard. They also conjectured that the exact version of MDP is NP-hard. This conjecture remained open until groundbreaking work of Vardy [Var97], who showed that exact MDP is indeed NP-hard. Not long after, Dumer, Micciancio, and Sudan [DMS03] showed that approximate MDP is NP-hard under randomized reductions. Follow-up work by Cheng and Wan [CW12], Austrin and Khot [AK14], and Micciancio [Mic14] showed that approximate MDP is NP-hard under deterministic reductions. The unparameterized fine-grained hardness of NCP and MDP was recently studied by Stephens-Davidowitz and Vaikuntanathan [SV19].

On the parameterized front, Downey, Fellows, Vardy, and Whittle [DFVW99] showed, among other things, that the exact version of NCP is W[1]-hard, and infamously conjectured that MDP is W[1]-hard. As discussed above, the status of this conjecture did not budge until the seminal work [BBE+21], where it was shown that γ-NCP$_q$ is W[1]-hard for every $\gamma \geqslant 1$ and prime power q, and that γ-MDP$_2$ is W[1]-hard for every $\gamma \geqslant 1$. Finally, by establishing the parameterized fine-grained hardness of Exact Set Cover and invoking results from [BGKM18, BBE+21], Manurangsi [Man20] showed that there are no algorithms running in time $n^{o(k)}$ for deciding γ-NCP$_q$ (for all constant $\gamma \geqslant 1$) and γ-MDP$_2$ (for some constant $\gamma > 1$) assuming Gap-ETH.

Complexity of CVP **and** SVP**.** The study of the complexity of lattice problems was initiated by van Emde Boas [vEB81], who showed that CVP$_2$ was NP-hard. He also showed that SVP$_\infty$ is NP-hard and conjectured that SVP$_2$ was NP-hard. This result remained the state-of-the-art until Ajtai [Ajt98] extended it to the ℓ_2 norm under randomized reductions, and a deep line of work soon followed showing progressively stronger hardness of approximation results for SVP$_p$ in different ℓ_p norms [CN98, Mic00, Din02, Kho05, HR12, Mic12]. A recent line of work has also focused on the (unparameterized) fine-grained hardness of approximate CVP and SVP [BGS17, AS18, ABGS21, BPT22].

In terms of parameterized hardness, Downey, Fellows, Vardy, and Whittle [DFVW99] showed that exact CVP is W[1]-hard, and asked whether SVP is W[1]-hard. As was the case for MDP, this question was only settled in [BBE+21], where it was shown that γ-CVP$_p$ is W[1]-hard for all $p \geqslant 1$ and $\gamma \geqslant 1$, and that γ-SVP$_p$ is W[1]-hard for $p > 1$ with some $\gamma = \gamma(p) > 1$. From a fine-grained perspective, it was shown by Manurangsi [Man20] that, assuming Gap-ETH, there are no algorithms running in time $n^{o(k^p)}$, where n is the rank of the input lattice, for deciding γ-CVP$_p$ with any $\gamma \geqslant 1$ for $p \geqslant 1$ and deciding γ-SVP$_p$ with some $\gamma > 1$ for all $p > 1$.

1.4 Open problems

We highlight two interesting directions for future research:

- The reductions that we use to prove all of our main theorems are randomized and have two-sided error due to our randomized constructions of locally dense codes and lattices and

due to our use of sparsification. It would be a groundbreaking contribution to find ways to derandomize these reductions and obtain deterministic parameterized hardness results for MDP and SVP. We note that when it comes to showing NP-hardness (instead of W[1]-hardness), we know deterministic reductions from NCP to MDP [CW12, AK14, Mic14] and randomized reductions with one-sided error from CVP to SVP [Mic12]. Additionally, we note that showing deterministic (NP-)hardness of SVP in the non-parameterized setting is a major open question.

- We have shown that γ-SVP$_p$ is W[1]-hard for any fixed $p > 1$ and $\gamma \geqslant 1$. When $p = 1$, we showed that γ-SVP$_p$ is W[1]-hard when $\gamma \in [1, 2)$. We leave it as a fascinating open problem to extend our W[1]-hardness result for all $\gamma \geqslant 1$ to $p = 1$ as well. This is an important missing piece of our understanding of the parameterized hardness of approximate SVP in ℓ_p norms.

1.5 Acknowledgements

We thank Arnab Bhattacharyya, Ishay Haviv, Bingkai Lin, Pasin Manurangsi, Xuandi Ren, and Noah Stephens-Davidowitz for helpful comments and answers to our questions. In particular, we would like to thank Ishay Haviv for sketching how to extend the tensoring-based hardness amplification for SVP in [HR12] to general ℓ_p norms, and would like to thank Pasin Manurangsi for clarifying the status of W[1]-hardness of approximation results for SVP in the ℓ_2 norm.

H. Bennett's research was partially supported by NSF Award No. CCF-2312297. M. Cheraghchi's research was partially supported by the National Science Foundation under Grants No. CCF-2006455, CCF-2107345, and a CAREER Award CCF-2236931. V. Guruswami's research was supported in part by NSF grants CCF-2228287 and CCF-2210823, a Simons Investigator award, and a UC Noyce Initiative award. J. Ribeiro's research was supported by NOVA LINCS (UIDB/04516/2020) with the financial support of FCT - Fundação para a Ciência e a Tecnologia and by the NSF grants CCF-1814603 and CCF-2107347 and the following grants of Vipul Goyal: the NSF award 1916939, DARPA SIEVE program, a gift from Ripple, a DoE NETL award, a JP Morgan Faculty Fellowship, a PNC center for financial services innovation award, and a Cylab seed funding award.

2 Preliminaries

Throughout we use boldface, lower-case letters like $\boldsymbol{v}, \boldsymbol{x}, \boldsymbol{s}, \boldsymbol{t}$ to denote column vectors.

2.1 Probability theory

We denote random variables by uppercase letters such as X, Y, and Z. Throughout this work we consider only discrete random variables supported on finite sets. Given a random variable X, we denote its expected value by $\mathrm{E}[X]$ and its variance by $\mathrm{Var}[X]$. We write the indicator random variable for an event E as $\mathbf{1}_{\{E\}}$.

We will make use of the following standard corollary of Chebyshev's inequality. For completeness, we provide a short proof.

Lemma 2.1. *Let $X_1, \ldots, X_N$ be pairwise independent random variables over $\{0, 1\}$ such that $\Pr[X_i = 1] = p > 0$ for $i = 1, \ldots, N$. Then, it holds that*

$$\Pr\left[\forall i \in [N], X_i = 0\right] \leqslant \frac{1}{pN}.$$

Proof. Let $X = \sum_{i=1}^{N} X_i$ and note that $\mathrm{E}[X] = pN$. We have

$$\Pr[X = 0] \leqslant \Pr[|X - \mathrm{E}[X]| \geqslant pN] \leqslant \frac{\mathrm{Var}[X]}{(pN)^2} = \frac{(1-p)pN}{(pN)^2} \leqslant \frac{1}{pN},$$

where the second inequality follows from Chebyshev's inequality and the equality holds due to the pairwise independence of the X_i's. □

2.2 Parameterized promise problems and FPT reductions

We recall basic definitions related to parameterized promise (decision) problems and Fixed-Parameter Tractable (FPT) reductions between such problems. We refer the reader to [DF99] for an excellent discussion on parameterized algorithms and reductions.

Definition 2.2 (Parameterized language)**.** A set $\mathcal{S} \subseteq \Sigma^* \times \mathbb{N}$ is said to be a *parameterized language* (with respect to the rightmost coordinate).

Definition 2.3 (Parameterized promise problem)**.** The tuple of parameterized languages $\Pi = (\Pi_{\mathrm{YES}}, \Pi_{\mathrm{NO}})$ is said to be a *parameterized promise problem* if $\{x : (x,k) \in \Pi_{\mathrm{YES}}\} \cap \{x : (x,k) \in \Pi_{\mathrm{NO}}\} = \emptyset$ for every parameter choice $k \in \mathbb{N}$.

Definition 2.4 (Randomized FPT reductions with two-sided error)**.** We say that a randomized algorithm is a *randomized FPT reduction with two-sided error* from the parameterized promise problem Π to the parameterized promise problem Π' if the following properties hold:

- On input (x,k), the algorithm runs in time at most $T(k) \cdot |x|^c$ for some computable function $T(\cdot)$ and some absolute constant $c > 0$ and outputs a tuple (x', k');
- It holds that $k' \leqslant g(k)$ for some computable function $g(\cdot)$;
- If $(x,k) \in \Pi_{\mathrm{YES}}$, it holds that $\Pr[(x',k') \in \Pi'_{\mathrm{YES}}] \geqslant 2/3$, where the probability is taken over the randomness of the algorithm;
- If $(x,k) \in \Pi_{\mathrm{NO}}$, it holds that $\Pr[(x',k') \in \Pi'_{\mathrm{NO}}] \geqslant 2/3$, where the probability is taken over the randomness of the algorithm.

Note that if there is a randomized FPT reduction with two-sided error from Π to Π', it follows that there is a randomized FPT algorithm (i.e., an algorithm running in time $T(k) \cdot |x|^c$ for some computable function $T(\cdot)$ on input an instance (x,k)) for deciding Π with two-sided error whenever there is such an algorithm for deciding Π'. The success probability of any such algorithm can be amplified in a standard manner.

In this work we focus on the parameterized complexity class $\mathsf{W}[1]$. It is well-known that the parameterized Clique problem, in which we are given as input a graph G and a positive integer k (with k being the parameter of interest) and must decide whether G contains a clique of size k, is $\mathsf{W}[1]$-complete. That is, parameterized Clique is in $\mathsf{W}[1]$ and it is $\mathsf{W}[1]$-hard, i.e., there is an FPT reduction from every problem in $\mathsf{W}[1]$ to it (see, e.g., [CFK+15, Theorem 13.18]). Therefore, one may *define* $\mathsf{W}[1]$ to be the class of all parameterized problems with FPT reductions to Clique. We refrain from discussing $\mathsf{W}[1]$ in more detail; for an extensive discussion, see [DF99, Chapters 9 to 11].

It is widely believed that $\mathsf{W}[1]$ problems cannot be decided by FPT algorithms, even if randomness with two-sided error is allowed. We say that a parameterized promise problem Π' is $\mathsf{W}[1]$-*hard*

under randomized reductions if there is a randomized FPT reduction with two-sided error from a W[1]-hard problem Π to Π'. The existence of such a reduction shows that Π' is likely intractable from a parameterized perspective.

2.3 Coding problems

Let $\mathcal{C}(G) := \{G\boldsymbol{x} : \boldsymbol{x} \in \mathbb{F}_q^n\}$ denote the code generated by the generator matrix $G \in \mathbb{F}_q^{m\times n}$ (note that here $\mathcal{C}(G)$ is the $\mathbb{F}_q$-span of the *columns* of G). Alternatively, we may see $\mathcal{C}(G)$ as the kernel of the parity-check matrix $H \in \mathbb{F}_q^{(m-n)\times m}$ which spans the dual subspace of $\mathcal{C}(G)$, i.e., $\mathcal{C}(G) = \{\boldsymbol{y} \in \mathbb{F}_q^m : H\boldsymbol{y} = \boldsymbol{0}\}$. We call sets of the form $\boldsymbol{u} + \mathcal{C} = \{\boldsymbol{u} + \boldsymbol{c} : \boldsymbol{c} \in \mathcal{C}\}$ with $\boldsymbol{u} \in \mathbb{F}_q^m$ the cosets of $\mathcal{C}$. We write $\|\boldsymbol{x}\|_0 = |\{i \in [m] : \boldsymbol{x}_i \neq 0\}$ for the Hamming weight of a vector $\boldsymbol{x} \in \mathbb{F}_q^m$ and call $\|\boldsymbol{x} - \boldsymbol{y}\|_0$ the Hamming distance between $\boldsymbol{x}$ and $\boldsymbol{y}$. For a code $\mathcal{C} \subseteq \mathbb{F}_q^m$, let $\lambda(\mathcal{C}) = \min_{\boldsymbol{c}\in\mathcal{C}\setminus\{\boldsymbol{0}\}} \|\boldsymbol{c}\|_0$ be the Hamming minimum distance of $\mathcal{C}$, and let $\operatorname{dist}(\boldsymbol{y}, \mathcal{C}) := \min_{\boldsymbol{c}\in\mathcal{C}} \|\boldsymbol{y} - \boldsymbol{c}\|_0$ denote the Hamming distance between a vector $\boldsymbol{y} \in \mathbb{F}_q^m$ and $\mathcal{C}$. Let $\mathcal{B}_{q,m}(r) = \{\boldsymbol{x} \in \mathbb{F}_q^m : \|\boldsymbol{x}\|_0 \leqslant r\}$ denote the Hamming ball of radius r in $\mathbb{F}_q^m$.

We define two fundamental promise problems from coding theory.

Definition 2.5 (Nearest Codeword Problem)**.** The *γ-approximate Nearest Codeword Problem over $\mathbb{F}_q$ (γ-NCP$_q$)* is the decisional promise problem defined as follows. On input a generator matrix $G \in \mathbb{F}_q^{m\times n}$, target $\boldsymbol{t} \in \mathbb{F}_q^m$, and distance parameter $k \in \mathbb{Z}^+$, the goal is to decide between the following two cases when one is guaranteed to hold:

- (YES) $\operatorname{dist}(\mathcal{C}(G), \boldsymbol{t}) \leqslant k$,
- (NO) $\operatorname{dist}(\mathcal{C}(G), \boldsymbol{t}) > \gamma k$.

The parameter of interest is k.

Remark 2.6. A scaling argument shows that the NO case in Definition 2.5 is equivalent to

- (NO) $\operatorname{dist}(\mathcal{C}(G), \alpha\boldsymbol{t}) > \gamma k$ for any $\alpha \in \mathbb{F}_q \setminus \{0\}$.

The following results establish the W[1]-hardness and parameterized fine-grained hardness of NCP.

Theorem 2.7 ([BBE+21, Theorem 5.1, adapted])**.** *For any prime power $q \geqslant 2$ and real number $\gamma \geqslant 1$ it holds that γ-NCP$_q$ is W[1]-hard.*

Theorem 2.8 ([Man20, Corollary 5, adapted])**.** *For any fixed prime power q and $\gamma \geqslant 1$ and any function T, assuming randomized Gap-ETH, there is no randomized algorithm running in time $T(k)n^{o(k)}$ which decides γ-NCP$_q$ with probability at least $2/3$, where n is the dimension of the input code.*

We remark that [BBE+21] states Theorem 2.7 as the W[1]-hardness of the "γ-MLD$_q$" problem (where "MLD" stands for "Maximum Likelihood Decoding" and the parameter of interest is again the input distance k), which is equivalent to the γ-NCP$_q$ problem. More precisely, the input to the γ-MLD$_q$ problem consists of a parity-check matrix $H \in \mathbb{F}_q^{h\times m}$, a target $\boldsymbol{t} \in \mathbb{F}_q^m$, and a distance bound k, and we must decide whether there exists a vector $\boldsymbol{e}$ with $\|\boldsymbol{e}\|_0 \leqslant k$ such that $H\boldsymbol{e} = H\boldsymbol{t}$ (i.e., $\boldsymbol{t}$ and $\boldsymbol{e}$ have the same syndrome), or whether all such vectors $\boldsymbol{e}$ have Hamming weight larger than γk. This is equivalent to γ-NCP$_q$ because we can efficiently compute the generator matrix

G of the code with parity-check matrix H and vice-versa, and because $H\boldsymbol{e} = H\boldsymbol{t}$ if and only if $\boldsymbol{t} = \boldsymbol{c} + \boldsymbol{e}$ for some codeword $\boldsymbol{c} \in \mathcal{C}(G)$. Moreover, [BBE+21] only stated the result for prime q. However, direct inspection of [BBE+21, Section 5.2] shows that their proof also yields the more general version stated in Theorem 2.7. In particular, [BBE+21, Definition 5.3], including the two observations there, generalizes to arbitrary finite fields.

Definition 2.9 (Minimum Distance Problem)**.** The *γ-approximate Minimum Distance Problem over $\mathbb{F}_q$ (γ-MDP$_q$)* is the decisional promise problem defined as follows. On input a generator matrix $G \in \mathbb{F}_q^{m \times n}$ and distance parameter $k \in \mathbb{Z}^+$, the goal is to decide between the following two cases when one is guaranteed to hold:

- (YES) $\lambda(\mathcal{C}(G)) \leqslant k$,
- (NO) $\lambda(\mathcal{C}(G)) > \gamma k$.

The parameter of interest is k.

2.3.1 Tensoring codes

The tensor product of linear codes is an important operation for building new codes with interesting properties by combining two linear codes. In particular, tensoring can be used to boost the approximation factor in W[1]-hardness results for NCP and MDP from some constant $\gamma > 1$ to an arbitrary constant.

Given two linear codes $\mathcal{C}(G_1)$ and $\mathcal{C}(G_2)$ with $G_i \in \mathbb{F}_q^{m_i \times n_i}$ and minimum distance d_i for $i = 1, 2$, we define the associated tensor product code as

$$\mathcal{C}(G_1) \otimes \mathcal{C}(G_2) := \mathcal{C}(G_1 \otimes G_2) \ ,$$

where $G_1 \otimes G_2 \in \mathbb{F}_q^{m_1 m_2 \times n_1 n_2}$ is the Kronecker product of G_1 and G_2. Furthermore, we have

$$\lambda(\mathcal{C}(G_1) \otimes \mathcal{C}(G_2)) = d_1 \cdot d_2 \ . \tag{2}$$

See, e.g., [DMS03, Section V.B] for a proof.

Suppose that we know that γ-MDP$_q$ is W[1]-hard (under randomized reductions) for *some* $\gamma > 1$. Then, using Equation (2), we can immediately conclude that for any integer $c \geqslant 1$, γ^c-MDP$_q$ is W[1]-hard (under randomized reductions) by considering the tensored MDP instances $(B^{\otimes c}, k^c)$, where $B^{\otimes c}$ denotes the c-fold Kronecker product of B with itself. In particular, constructing tensored MDP instances in this way gives an FPT self-reduction from γ-MDP$_q$ to γ^c-MDP$_q$.

2.4 Lattice problems

Let $\mathcal{L}(B) = \{B\boldsymbol{x} : \boldsymbol{x} \in \mathbb{Z}^n\}$ denote the lattice generated by the matrix $B \in \mathbb{R}^{m \times n}$. We call n the rank of $\mathcal{L}(B)$ and write $\det(\mathcal{L}(B)) = \sqrt{\det(B^T B)}$ for the determinant of $\mathcal{L}(B)$, where B^T denotes the transpose of B. For $p \in [1, \infty)$, we write $\|\boldsymbol{x}\|_p = (\sum_{i=1}^m |x_i|^p)^{1/p}$ for the ℓ_p norm of a vector $x \in \mathbb{R}^m$. We use $\lambda_1^{(p)}(\mathcal{L})$ to denote the ℓ_p norm of a shortest nonzero vector in $\mathcal{L}$ and set $\mathrm{dist}_p(\mathcal{L}, \boldsymbol{t}) := \min_{\boldsymbol{v} \in \mathcal{L}} \|\boldsymbol{v} - \boldsymbol{t}\|_p$. We write $\mathcal{B}_m^{(p)}(r) = \{\boldsymbol{x} \in \mathbb{R}^m : \|\boldsymbol{x}\|_p \leqslant r\}$ for the closed, centered ℓ_p ball of radius r in $\mathbb{R}^m$.

We define two fundamental promise problems related to lattices.

Definition 2.10 (Closest Vector Problem)**.** The *γ-approximate Closest Vector Problem with respect to the p-norm (γ-CVP$_p$)* is the decisional promise problem defined as follows. On input a generator matrix $B \in \mathbb{Z}^{m \times n}$, a target $\boldsymbol{t} \in \mathbb{Z}^m$, and a distance parameter $k \in \mathbb{Z}^+$, the goal is to decide between the following two cases when one is guaranteed to hold:

- (YES) $\mathrm{dist}_p(\mathcal{L}(B), \boldsymbol{t}) \leqslant k$,
- (NO) $\mathrm{dist}_p(\mathcal{L}(B), \alpha\boldsymbol{t}) > \gamma k$ for any $\alpha \in \mathbb{Z} \setminus \{0\}$.

The parameter of interest is k.

The definition above is a slight (but widely used) variant of the original Closest Vector Problem, since in the NO case we also require all multiples $\alpha\boldsymbol{t}$ with $\alpha \in \mathbb{Z} \setminus \{0\}$ to be far from $\mathcal{L}(B)$. The following results about the $\mathsf{W}[1]$-hardness and parameterized fine-grained hardness of this variant of CVP are known to hold.

Theorem 2.11 ([BBE+21, Theorem 7.2])**.** *For any real numbers $\gamma, p \geqslant 1$ it holds that γ-CVP$_p$ is $\mathsf{W}[1]$-hard.*

Theorem 2.12 ([Man20, Corollary 6, adapted])**.** *For any fixed $p, \gamma \geqslant 1$ and any function T, assuming randomized Gap-ETH, there is no randomized algorithm running in time $T(k)n^{o(k^p)}$ which decides γ-CVP$_p$ with probability at least $2/3$, where n is the rank of the input lattice.*

Definition 2.13 (Shortest Vector Problem)**.** The *γ-approximate Shortest Vector Problem with respect to the ℓ_p-norm (γ-SVP$_p$)* is the decisional promise problem defined as follows. On input a generator matrix $B \in \mathbb{Z}^{m \times n}$ and a distance parameter $k \in \mathbb{Z}^+$, the goal is to decide between the following two cases when one is guaranteed to hold:

- (YES) $\lambda_1^{(p)}(\mathcal{L}(B)) \leqslant k$,
- (NO) $\lambda_1^{(p)}(\mathcal{L}(B)) > \gamma k$.

The parameter of interest is k.

2.4.1 Tensoring lattices

Analogously to the coding setting, we can also consider the tensor product of lattices. Given two lattices $\mathcal{L}(B_1)$ and $\mathcal{L}(B_2)$ with basis matrices $B_1 \in \mathbb{Z}^{m_1 \times n_1}$ and $B_2 \in \mathbb{Z}^{m_2 \times n_2}$, we define the associated tensor product lattice as

$$\mathcal{L}(B_1) \otimes \mathcal{L}(B_2) := \mathcal{L}(B_1 \otimes B_2) \ ,$$

where $B_1 \otimes B_2 \in \mathbb{Z}^{m_1 m_2 \times n_1 n_2}$ is the Kronecker product of B_1 and B_2. The tensor product lattice is independent of the bases we choose for the two underlying lattices.

Unlike for codes, it is not true that repeated tensoring of lattices allows us to generically boost the approximation factor in known hardness results for CVP and SVP. Indeed, while it always holds that

$$\lambda_1^{(p)}(\mathcal{L}(B_1) \otimes \mathcal{L}(B_2)) \leqslant \lambda_1^{(p)}(\mathcal{L}(B_1)) \cdot \lambda_1^{(p)}(\mathcal{L}(B_2)) \ ,$$

it may happen that the left-hand side of this inequality is significantly smaller than the right-hand side. For an example, see [HR12, Lemma 2.3]. Therefore, additional effort is required to prove special structural properties of our CVP and SVP instances to ensure that tensoring them allows us to boost the approximation factor in our hardness results.

2.5 Locally dense codes and lattices

Our randomized FPT reductions from NCP to MDP and from CVP to SVP use families of *locally dense* codes and lattices with appropriate parameters. Precise definitions of such objects follow below.

Definition 2.14 (Locally dense code)**.** Fix a real number $\alpha \in (0,1)$, positive integers d, N, m, n, and a prime power q. A (q, α, d, N, m, n)-*locally dense code* is specified by a generator matrix $A \in \mathbb{F}_q^{m \times n}$ and a target vector $\boldsymbol{s} \in \mathbb{F}_q^m$ with the following properties:

- $\lambda(\mathcal{C}(A)) > d$.
- $|(\mathcal{C}(A) - \boldsymbol{s}) \cap \mathcal{B}_{q,m}(\alpha d)| \geqslant N$.

That is, the code $\mathcal{C}(A)$ has block length m, dimension n, (design) minimum distance d, is over $\mathbb{F}_q$, and a "bad list decoding configuration" with at least N codewords within Hamming distance $\alpha d < d$ of $\boldsymbol{s}$.

Definition 2.15 (Locally dense lattice)**.** Fix real numbers $\alpha \in (0,1)$ and $p \geqslant 1$ and positive integers d, N, m, n. A (p, α, d, N, m, n)-*locally dense lattice* is specified by a basis $A \in \mathbb{Z}^{m \times n}$ and a target vector $\boldsymbol{s} \in \mathbb{Z}^m$ with the following properties:

- $\lambda_1^{(p)}(\mathcal{L}(A)) > d$.
- $|(\mathcal{L}(A) - \boldsymbol{s}) \cap \mathcal{B}_m^{(p)}(\alpha d)| \geqslant N$.

3 The FPT NCP_q to MDP_q reduction

We next describe and analyze a randomized FPT reduction from approximate NCP_q to approximate MDP_q which works over any finite field. Our reduction is obtained by adapting Khot's reduction [Kho05, BBE+21] from approximate CVP to approximate SVP to the coding setting and combining it with locally dense codes constructed with the help of BCH codes over general finite fields. Combined with Theorem 2.7, our reduction yields Theorem 1.1, which we restate here.

Theorem 1.1. *For any fixed prime power q and constant $\gamma \geqslant 1$, γ-MDP_q is $\mathsf{W}[1]$-hard under randomized FPT reductions with two-sided error.*

3.1 A reduction with advice

For the sake of exposition, we begin by describing our FPT reduction from NCP to MDP in a modular fashion assuming that we are given an appropriate locally dense code as advice. Later on in Section 3.3 we give an FPT randomized algorithm to construct locally dense codes with the desired parameters and replace the advice with this construction to yield the desired FPT reduction from approximate NCP to approximate MDP with two-sided error. We establish the following result.

Theorem 3.1. *Fix a prime power $q \geqslant 2$ and real numbers $\gamma, \gamma' \geqslant 1$ and $\alpha \in (0,1)$ additionally satisfying*

$$\gamma' \leqslant \frac{\gamma}{1 + \alpha\gamma}.$$

Then, there is a randomized algorithm which, for m large enough, on input a γ-NCP$_q$ instance $(G, \boldsymbol{t}, k)$ with $G \in \mathbb{F}_q^{m \times n}$, $\boldsymbol{t} \in \mathbb{F}_q^m$, and $k \in \mathbb{Z}^+$ and a $(q, \alpha, d = \gamma k, N \geqslant 100q^{10} \cdot (qm)^d, m', n')$-locally dense code $(A, \boldsymbol{s})$ outputs in time $\mathrm{poly}(m, m')$ an instance (G_{final}, k') of γ'-MDP$_q$ with $k' < \gamma k$ satisfying the following properties with probability at least 0.99:

- *If $(G, \boldsymbol{t}, k)$ is a YES instance of γ-NCP$_q$, then (G_{final}, k') is a YES instance of γ'-MDP$_q$;*
- *If $(G, \boldsymbol{t}, k)$ is a NO instance of γ-NCP$_q$, then (G_{final}, k') is a NO instance of γ'-MDP$_q$.*

We prove Theorem 3.1 by analyzing the following algorithm. On input a γ-NCP$_q$ instance $(G, \boldsymbol{t}, k)$ with $G \in \mathbb{F}_q^{m \times n}$ and $\boldsymbol{t} \in \mathbb{F}_q^m$, we set $\mathcal{C}_{\mathrm{int}}$ to be the code with generator matrix

$$G_{\mathrm{int}} := \begin{pmatrix} G & 0_{m \times n'} & -\boldsymbol{t} \\ 0_{m' \times n} & A & -\boldsymbol{s} \end{pmatrix} \in \mathbb{F}_q^{(m+m') \times (n+n'+1)},$$

where $(A, \boldsymbol{s})$ is the locally dense code described in the statement of Theorem 3.1. Note that G_{int} has full column rank (over $\mathbb{F}_q$) whenever G and A have full column rank, since we always have $\boldsymbol{s} \notin \mathcal{C}(A)$ (observe that there exists at least one codeword of $\mathcal{C}(A)$ within distance $\alpha d < d$ of $\boldsymbol{s}$).

We will take the intersection of $\mathcal{C}_{\mathrm{int}}$ with an appropriate random code $\mathcal{C}_{\mathrm{rand}} \subseteq \mathbb{F}_q^{m+m'}$ of codimension at most $h = \lceil 7 + d(1 + \log_q m) \rceil$. More precisely, we sample $\mathcal{C}_{\mathrm{rand}}$ by first sampling the entries of a parity-check matrix $H \in \mathbb{F}_q^{h \times (m+m')}$ independently and uniformly at random from $\mathbb{F}_q$ and setting $\mathcal{C}_{\mathrm{rand}} = \ker(H)$. Then, we compute a generator matrix G_{final} of $\mathcal{C}_{\mathrm{final}} = \mathcal{C}_{\mathrm{int}} \cap \mathcal{C}_{\mathrm{rand}}$ and $k' := k + \alpha d$, and output (G_{final}, k') as the MDP instance. Note that $k' \leqslant d/\gamma'$ by the constraints imposed on γ, γ', and α.

3.2 Proof of Theorem 3.1

In order to prove Theorem 3.1, we begin by establishing some useful properties of the intermediate code $\mathcal{C}_{\mathrm{int}}$ constructed by the algorithm from Section 3.1.

Lemma 3.2. *Fix a prime power $q \geqslant 2$ and real numbers $\gamma, \gamma' \geqslant 1$ and $\alpha \in (0, 1)$ satisfying*

$$\gamma' \leqslant \frac{\gamma}{1 + \alpha\gamma}.$$

Given a γ-NCP$_q$ instance $(G, \boldsymbol{t}, k)$ with $G \in \mathbb{F}_q^{m \times n}$ and $\boldsymbol{t} \in \mathbb{F}_q^m$ and a $(q, \alpha, d = \gamma k, N \geqslant 100q^{10} \cdot (qm)^d, m', n')$-locally dense code $(A, \boldsymbol{s})$, the algorithm from Section 3.1 constructs $\mathcal{C}_{\mathrm{int}} = \mathcal{C}(G_{\mathrm{int}})$ in time $\mathrm{poly}(m, m')$ satisfying the following properties:

- *If $(G, \boldsymbol{t}, k)$ is a YES instance of γ-NCP$_q$, then there are at least N nonzero vectors in $\mathcal{C}_{\mathrm{int}}$ of Hamming weight at most k'. We call such vectors* good*;*
- *If $(G, \boldsymbol{t}, k)$ is a NO instance of γ-NCP$_q$, then there are at most $(qm)^d$ nonzero vectors in $\mathcal{C}_{\mathrm{int}}$ of Hamming weight at most $d = \gamma k \geqslant \gamma' k'$. We call such vectors* annoying*.*

Proof. The claim regarding the time required to construct $\mathcal{C}_{\mathrm{int}}$ is directly verifiable. We proceed to argue the two items of the lemma statement.

First, suppose that $(G, \boldsymbol{t}, k)$ is a YES instance of γ-NCP$_q$. This means that there is a vector $\boldsymbol{x} \in \mathbb{F}_q^n$ such that $\|G\boldsymbol{x} - \boldsymbol{t}\|_0 \leqslant k$. Moreover, we know that there are at least $N \geqslant 100q^{10} \cdot (qm)^d$ vectors $\boldsymbol{y} \in \mathbb{F}_q^{n'}$ such that

$$\|A\boldsymbol{y} - \boldsymbol{s}\|_0 \leqslant \alpha d.$$

For each such $\boldsymbol{y}$, consider the associated vector $\boldsymbol{z}_{\boldsymbol{y}} = (\boldsymbol{x}, \boldsymbol{y}, 1)$ and note that

$$\|G_{\text{int}}\boldsymbol{z}_{\boldsymbol{y}}\|_0 = \|G\boldsymbol{x} - \boldsymbol{t}\|_0 + \|A\boldsymbol{y} - \boldsymbol{s}\|_0 \leqslant k + \alpha d = k'.$$

Therefore, there are at least $N \geqslant 100q^{10} \cdot (qm)^d$ good vectors in $\mathcal{C}_{\text{int}}$, as desired.

On the other hand, suppose that $(G, \boldsymbol{t}, k)$ is a NO instance of γ-NCP$_q$. This means that for every $\boldsymbol{x} \in \mathbb{F}_q^n$ and $\beta \in \mathbb{F}_q \setminus \{0\}$ it holds that $\|G\boldsymbol{x} - \beta\boldsymbol{t}\|_0 > \gamma k = d$. Consider an arbitrary vector $\boldsymbol{z} = (\boldsymbol{x}, \boldsymbol{y}, -\beta) \in \mathbb{F}_q^{n+n'+1}$. We claim that if $G_{\text{int}}\boldsymbol{z}$ is annoying it must be the case that $\boldsymbol{y} = \boldsymbol{0}$ and $\beta = 0$. To see this, first note that if $\beta \neq 0$ then

$$\|G_{\text{int}}\boldsymbol{z}\|_0 \geqslant \|G\boldsymbol{x} - \beta\boldsymbol{t}\|_0 > d$$

since $(G, \boldsymbol{t}, k)$ is a NO instance of γ-NCP$_q$ and $d = \gamma k$. Therefore, we may assume that $\beta = 0$. Under this assumption, it holds that

$$\|G_{\text{int}}\boldsymbol{z}\|_0 \geqslant \|A\boldsymbol{y}\|_0 > d$$

if $\boldsymbol{y} \neq \boldsymbol{0}$, which yields the claim. This allows us to conclude that all vectors $\boldsymbol{z}$ such that $G_{\text{int}}\boldsymbol{z}$ is annoying are of the form $\boldsymbol{z} = (\boldsymbol{x}, \boldsymbol{0}, 0)$ for some $\boldsymbol{x} \in \mathbb{F}_q^n$. As a result, the number of annoying vectors is at most

$$|\mathcal{C}(G) \cap \mathcal{B}_{q,m}(d)| \leqslant |\mathcal{B}_{q,m}(d)| \leqslant (qm)^d,$$

as desired. □

We are now ready to prove Theorem 3.1 with the help of Lemma 3.2.

Proof of Theorem 3.1. The claims regarding the time required to construct $\mathcal{C}_{\text{final}}$ and the bound on k' are directly verifiable. We proceed to argue the two items of the theorem statement.

Recall that we construct $\mathcal{C}_{\text{final}}$ by intersecting $\mathcal{C}_{\text{int}}$ with an appropriate random code $\mathcal{C}_{\text{rand}}$ of codimension at most $h = \lceil 7 + d(1 + \log_q m) \rceil$. More precisely, $\mathcal{C}_{\text{rand}}$ is obtained by sampling the entries of a parity-check matrix $H \in \mathbb{F}_q^{h \times (m+m')}$ independently and uniformly at random from $\mathbb{F}_q$ and setting $\mathcal{C}_{\text{rand}} = \ker(H)$. Observe that for any given $\boldsymbol{v} \in \mathbb{F}_q^{m+m'} \setminus \{\boldsymbol{0}\}$ we have

$$\Pr_H[H\boldsymbol{v} = \boldsymbol{0}] = q^{-h}. \tag{3}$$

Moreover, the random variables $\mathbf{1}_{\{H\boldsymbol{v}=\boldsymbol{0}\}}$ and $\mathbf{1}_{\{H\boldsymbol{w}=\boldsymbol{0}\}}$ are pairwise independent whenever $\boldsymbol{v}$ and $\boldsymbol{w}$ are linearly independent. Let $Z_{\boldsymbol{v}} = \mathbf{1}_{\{H\boldsymbol{v}=\boldsymbol{0}\}}$ and write $Z_{\mathcal{S}} = \sum_{\boldsymbol{v} \in \mathcal{S}} Z_{\boldsymbol{v}}$ for any set $\mathcal{S}$.

Suppose that $(G, \boldsymbol{t}, k)$ is a YES instance of γ-NCP$_q$. By Lemma 3.2, this means that there are at least $100q^{10} \cdot (qm)^d$ good nonzero vectors in $\mathcal{C}_{\text{int}}$ of Hamming weight at most k'. Let $\mathcal{G}$ denote the set of such good vectors. We claim that

$$\Pr[Z_{\mathcal{G}} = 0] \leqslant 0.01,$$

i.e., at least one good vector survives with probability at least 0.99 over the sampling of $\mathcal{C}_{\text{rand}}$. Note that there exists a subset $\mathcal{G}' \subseteq \mathcal{G}$ of size

$$|\mathcal{G}'| \geqslant |\mathcal{G}|/q \geqslant 100q^9 \cdot (qm)^d \tag{4}$$

such that all vectors in $\mathcal{G}'$ are pairwise linearly independent. This set $\mathcal{G}'$ can be obtained by keeping only one element of $\mathcal{G}$ per line in $\mathbb{F}_q^{m+m'}$. Note that the variables $\{Z_{\boldsymbol{v}}\}_{\boldsymbol{v}\in\mathcal{G}'}$ are pairwise independent Bernoulli random variables with success probability q^{-h}, and so Lemma 2.1 guarantees that

$$\Pr[Z_{\mathcal{G}} = 0] \leqslant \Pr[Z_{\mathcal{G}'} = 0] \leqslant \frac{q^h}{|\mathcal{G}'|} \leqslant \frac{q^{h+1}}{|\mathcal{G}|} \leqslant 0.01,$$

by our choice of h and the lower bound on $|\mathcal{G}'|$ from Equation (4). Therefore, with probability at least 0.99 there is a codeword $\boldsymbol{v} \in \mathcal{C}_{\text{final}} \setminus \{\mathbf{0}\}$ such that $\|\boldsymbol{v}\|_0 \leqslant k'$, and so (G_{final}, k') is a YES instance of γ'-MDP$_q$.

Now, suppose that $(G, \boldsymbol{t}, k)$ is a NO instance of γ-NCP$_q$. In this case, Lemma 3.2 ensures that there are at most $(qm)^d$ nonzero vectors in $\mathcal{C}_{\text{int}}$ with Hamming weight at most d. Let $\mathcal{A}$ denote the set of such annoying vectors. Note that

$$\Pr[Z_{\mathcal{A}} \geqslant 1] \leqslant \frac{(qm)^d}{q^h} \leqslant 0.01,$$

where the first inequality follows from Equation (3) and a union bound over all $|\mathcal{A}| \leqslant (qm)^d$ annoying vectors, and the second inequality follows from the choice of h above. Therefore, all annoying vectors are removed from $\mathcal{C}_{\text{final}}$ with probability at least 0.99. This means that $\mathcal{C}_{\text{final}}$ has minimum distance larger than d, and so (G_{final}, k') is a NO instance of γ'-MDP$_q$. □

3.3 Finalizing the reduction

In this section we provide a randomized construction of locally dense codes based on BCH codes [Hoc59, BR60] which can be combined with Theorem 3.1 to yield the desired FPT reduction with two-sided error and without advice. More precisely, we prove the following theorem.

Theorem 3.3. *Fix a prime power $q \geqslant 2$ and set $\gamma = 4q$. There exists a randomized algorithm which when given as input positive integers m and $k \leqslant m$ runs in time $\text{poly}(m)$ and outputs with probability at least 0.99 a $(q, \alpha, d, N, m', n')$-locally dense code $(A, \boldsymbol{s})$, where*

$$\begin{aligned} &m', n' \leqslant \text{poly}(m), \\ &d = \gamma k = 4qk, \\ &\alpha = 1 - \frac{1}{2q}, \\ &N = \frac{(qm)^{2d}}{100} \geqslant 100q^{10} \cdot (qm)^d, \end{aligned}$$

provided that m is sufficiently large compared to q.

Combining Theorems 2.7, 3.1 and 3.3 shows that γ'-MDP$_q$ is W[1]-hard under randomized reductions with two-sided error and $\gamma' = \frac{4q}{4q-1} > 1$. Then, as discussed in Section 2.3.1, coupling this result with a tensoring argument immediately shows that γ-MDP$_q$ is W[1]-hard for an arbitrary constant $\gamma \geqslant 1$, leading to Theorem 1.1. Similarly, since $k' = O(k)$ in Theorem 3.1, combining Theorems 2.8, 3.1 and 3.3 yields Theorem 1.5.

3.3.1 BCH codes over $\mathbb{F}_q$

The following theorem states the main properties of (narrow-sense, primitive) BCH codes with design minimum distance over an arbitrary finite field (see [Gur10] for a discussion of BCH codes and related objects). Although versions of this theorem are well-known, we present a proof in Appendix A for completeness.

Theorem 3.4 (q-ary BCH codes)**.** *Fix a prime power q. Then, given integers $m' = q^r - 1$ and $1 \leqslant d \leqslant m'$ for some integer r, it is possible to construct in time $\mathrm{poly}(m')$ a generator matrix $G_{\mathrm{BCH}} \in \mathbb{F}_q^{m' \times n'}$ such that $\mathcal{C}_{\mathrm{BCH}} = \mathcal{C}(G_{\mathrm{BCH}}) \subseteq \mathbb{F}_q^{m'}$ has minimum distance at least d and codimension*

$$m' - n' \leqslant \lceil (d-1)(1-1/q) \rceil \log_q(m'+1).$$

3.3.2 Locally dense codes from BCH codes

We now show how to use q-ary BCH codes (Theorem 3.4) with appropriate parameters to construct locally dense codes satisfying Theorem 3.3. This construction is similar in spirit to the construction of locally dense lattices by Khot [Kho05].

Proof of Theorem 3.3. Suppose that we are given as input q, k, m. Let $d = 4qk$. Choose m' to be the smallest number of the form $q^r - 1$ larger than or equal to $(dqm)^{4q}$, and set $\alpha = 1 - \frac{1}{2q}$. Let $G_{\mathrm{BCH}} \in \mathbb{F}_q^{m' \times n'}$ be the generator matrix of the $\mathcal{C}_{\mathrm{BCH}}$ code with minimum distance at least $d+1$ and codimension

$$m' - n' \leqslant \lceil d(1-1/q) \rceil \log_q(m'+1) = d(1-1/q)\log_q(m'+1) \tag{5}$$

guaranteed by Theorem 3.4, where the last equality holds by our choice of d. We sample our locally dense code $(A, \boldsymbol{s})$ as follows:

1. Set $A = G_{\mathrm{BCH}}$ with G_{BCH} as defined above;
2. Sample $\boldsymbol{s}$ uniformly at random from the set of vectors in $\{0,1\}^{m'}$ of Hamming weight exactly αd, which we denote by $B_{m',\alpha d}$.

By Theorem 3.4, this procedure runs in time $\mathrm{poly}(m') = \mathrm{poly}(m)$. We now show that this procedure outputs with probability at least 0.99 an $(q, \alpha, d, N, m', n')$-locally dense code $(A, \boldsymbol{s})$, where

$$\begin{aligned} &m', n' \leqslant \mathrm{poly}(m), \\ &d = 4qk, \\ &\alpha = 1 - \frac{1}{2q}, \\ &N = \frac{(qm)^{2d}}{100}, \end{aligned}$$

which yields Theorem 3.3. It follows directly from the length, codimension, and minimum distance of $\mathcal{C}_{\mathrm{BCH}} = \mathcal{C}(G_{\mathrm{BCH}}) = \mathcal{C}(A)$ that $m', n' \leqslant \mathrm{poly}(m)$ and $d = \gamma k = 4qk$. It remains to show that with probability at least 0.99 over the sampling of $\boldsymbol{s}$ as above it holds that

$$|(\mathcal{C}_{\mathrm{BCH}} - \boldsymbol{s}) \cap \mathcal{B}_{q,m'}(\alpha d)| \geqslant \frac{\binom{m'}{\alpha d}}{100(m'+1)^{d(1-1/q)}} \geqslant \frac{(qm)^{2d}}{100} = N. \tag{6}$$

We follow the reasoning of [Kho05, Lemma 4.3] to prove the leftmost inequality. Call a coset[11] V

[11] A *coset* of a linear code $\mathcal{C} \subseteq \mathbb{F}_q^n$ is a set of the form $\boldsymbol{v} + \mathcal{C}$ for some vector $\boldsymbol{v} \in \mathbb{F}_q^n$.

of $\mathcal{C}_{\mathrm{BCH}} = \mathcal{C}(A)$ *good* if

$$|V \cap B_{m',\alpha d}| \geqslant \frac{|B_{m',\alpha d}|}{100 \cdot q^{m'-n'}}$$

and call V *bad* otherwise. Note that if V is a good coset it follows that

$$|V \cap \mathcal{B}_{q,m'}(\alpha d)| \geqslant |V \cap B_{m',\alpha d}| \geqslant \frac{|B_{m',\alpha d}|}{100 \cdot q^{m'-n'}} \geqslant \frac{\binom{m'}{\alpha d}}{100(m'+1)^{d(1-1/q)}},$$

where the third inequality holds because of Equation (5). As a result, the leftmost inequality of Equation (6) holds, and so it is enough to show that $\boldsymbol{s}$ lands in a good coset with probability at least 0.99.

Since $\boldsymbol{s}$ is sampled uniformly at random from $B_{m',\alpha d}$, we obtain a representative of coset V with probability

$$\frac{|V \cap B_{m',\alpha d}|}{|B_{m',\alpha d}|}.$$

Therefore, we have that

$$\begin{aligned}
\Pr[\boldsymbol{s} \text{ lands in a bad coset}] &= \sum_{V : V \text{ is a bad coset}} \frac{|V \cap B_{m',\alpha d}|}{|B_{m',\alpha d}|} \\
&< \sum_{V : V \text{ is a bad coset}} \frac{1}{100 q^{m'-n'}} \\
&\leqslant \frac{1}{100}.
\end{aligned}$$

The first inequality follows from the definition of a bad coset. The second inequality holds because there are at most $q^{m'-n'}$ bad cosets.

It remains to prove the rightmost inequality of Equation (6). Recalling that $m' \geqslant (dqm)^{4q}$, we have that

$$\begin{aligned}
\frac{\binom{m'}{\alpha d}}{100(m'+1)^{d(1-1/q)}} &\geqslant \frac{\binom{m'}{\alpha d}}{100(2m')^{d(1-1/q)}} \\
&\geqslant \frac{(m')^{\alpha d}}{100 d^d (2m')^{d(1-1/q)}} \\
&\geqslant \frac{(m')^{\frac{d}{2q}}}{100(2d)^d} \\
&\geqslant \frac{(dqm)^{2d}}{100(2d)^d} \\
&\geqslant \frac{(qm)^{2d}}{100}.
\end{aligned}$$

The first inequality uses the fact that $m'+1 \leqslant 2m'$. The second inequality holds because $\binom{m'}{\alpha d} \geqslant \left(\frac{m'}{\alpha d}\right)^{\alpha d} \geqslant \frac{(m')^{\alpha d}}{d^d}$. The third inequality follows from the choice of α and since $2^{d(1-1/q)} \leqslant 2^d$. The fourth inequality holds since $m' \geqslant (dqm)^{4q}$. The fifth inequality holds because $d^{2d} \geqslant (2d)^d$ for all $d \geqslant 2$. □

4 The FPT CVP_p to SVP_p reduction

In this section we describe and analyze an FPT reduction from approximate CVP_p to approximate SVP_p which works for all $p \geqslant 1$. Our reduction is obtained by combining Khot's reduction [Kho05, BBE+21] from approximate CVP to approximate SVP with locally dense lattices stemming from Reed-Solomon-based Construction-A lattices, as first studied by Bennett and Peikert [BP23]. In conjunction with Theorem 2.11, our reduction yields Theorem 1.2, which we restate here.

Theorem 1.2. *For any fixed $p \in [1, \infty)$ and constant $\gamma \in [1, 2^{1/p})$, γ-SVP_p is $\mathsf{W}[1]$-hard under randomized FPT reductions with two-sided error.*

4.1 A reduction with advice

As in Section 3.1, we begin by describing our FPT reduction from CVP to SVP in a modular fashion assuming that we are given an appropriate locally dense lattice as advice. In Section 4.3, we give an FPT randomized algorithm to construct locally dense lattices with the desired parameters and replace the advice with this construction to yield the desired FPT reduction from approximate NCP to approximate MDP with two-sided error. More precisely, we have the following result.

Theorem 4.1. *Fix real numbers $p, \gamma, \gamma' \geqslant 1$ and $\alpha \in (0, 1)$ additionally satisfying*

$$\alpha \leqslant \frac{((\gamma/\gamma')^p - 2)^{1/p}}{\gamma}.$$

Then, there is a randomized algorithm which, for m large enough, on input a γ-CVP_p instance $(B, \boldsymbol{t}, k)$ with $B \in \mathbb{Z}^{m \times n}$, $\boldsymbol{t} \in \mathbb{Z}^m$, and $k \in \mathbb{Z}^+$ and a $(p, \alpha, d = \gamma k, N \geqslant 10^5 \cdot (2m(1+\gamma k))^{(\gamma k)^p}, m', n')$-locally dense lattice $(A, \boldsymbol{s})$ outputs in time polynomial in m, m', k, and the description bitlength of B and $\boldsymbol{t}$ an instance (B_{final}, k') of γ'-SVP_p with $k' < \gamma k$ satisfying the following properties with probability at least 0.99:

- *If $(B, \boldsymbol{t}, k)$ is a YES instance of γ-CVP_p, then (B_{final}, k') is a YES instance of γ'-SVP_p;*
- *If $(B, \boldsymbol{t}, k)$ is a NO instance of γ-CVP_p, then (B_{final}, k') is a NO instance of γ'-SVP_p.*

We prove Theorem 4.1 via the following algorithm. On input a γ-CVP_p instance $(B, \boldsymbol{t}, k)$ with $B \in \mathbb{Z}^{m \times n}$ and $\boldsymbol{t} \in \mathbb{Z}^m$, we set the intermediate lattice $\mathcal{L}_{\mathrm{int}}$ to be the lattice generated by

$$B_{\mathrm{int}} := \begin{pmatrix} B & 0_{m \times n'} & -\boldsymbol{t} \\ 0_{m' \times n} & A & -\boldsymbol{s} \\ \mathbf{0}_n & \mathbf{0}_{n'} & 1 \end{pmatrix},$$

where $(A, \boldsymbol{s})$ is the locally dense lattice described in Theorem 4.1. We add the bottom $(0, \ldots, 0, 1)$ row to B_{int} to ensure that it has full column rank whenever A and B have full column rank as well.

Then, we add an appropriate random constraint to $\mathcal{L}_{\mathrm{int}}$ in order to obtain the final SVP_p instance. More precisely, let ρ be a prime in the interval[12] $(100(2m(1+\gamma k))^{(\gamma k)^p}, 200(2m(1+\gamma k))^{(\gamma k)^p}]$. Sample a vector $\boldsymbol{v} \in \mathbb{Z}^{m+m'+1}$ by sampling each entry independently and uniformly at

[12] Let $a = 100(2m(1+\gamma k))^{(\gamma k)^p}$. By the prime number theorem, the density of primes in the interval $(a, 2a]$ is $1/\operatorname{poly}(\log a) = 1/\operatorname{poly}(m, k)$. Therefore, we can sample a prime ρ with high probability in time $\operatorname{poly}(m, k)$ by repeatedly sampling an integer from this interval uniformly at random and then checking whether it is prime (which can be done in time $\operatorname{poly}(\log a) = \operatorname{poly}(m, k)$ [AKS04]).

random from $\{0, \dots, \rho - 1\}$. We define B_{final} to be the (integral) basis of the sublattice $\mathcal{L}_{\text{final}} = \mathcal{L}(B_{\text{final}}) \subseteq \mathcal{L}_{\text{int}}$ defined as

$$\mathcal{L}_{\text{final}} = \{\boldsymbol{w} \in \mathcal{L}_{\text{int}} : \langle \boldsymbol{v}, \boldsymbol{w} \rangle = 0 \pmod{\rho}\}.$$

For fixed $p, \gamma \geqslant 1$, we can compute B_{final} given B_{int}, $\boldsymbol{v}$, and ρ as inputs in time polynomial in m', k, $\log \rho$, and the description bitlength of B and $\boldsymbol{t}$. This can be done by first computing a basis B' of the lattice $\mathcal{L}_{\boldsymbol{v},\rho} = \{\boldsymbol{w} \in \mathbb{Z}^{m+m'+1} : \langle \boldsymbol{v}, \boldsymbol{w} \rangle = 0 \pmod{\rho}\}$. This step can be done in time $\text{poly}(m, m', \log \rho)$ since $\mathcal{L}_{\boldsymbol{v},\rho}$ is the Construction-A lattice $\mathcal{L}_{\boldsymbol{v},\rho} = \mathcal{C}_{\boldsymbol{v},\rho} + \rho\mathbb{Z}^{m+m'+1}$ with $\mathcal{C}_{\boldsymbol{v},\rho}$ the ρ-ary linear code with parity-check matrix $H = \boldsymbol{v}^T$. Then, we compute a basis of $\mathcal{L}_{\text{final}} = \mathcal{L}_{\text{int}} \cap \mathcal{L}_{\boldsymbol{v},\rho}$ from B_{int} and B' following the efficient procedure discussed in [Mic10, Section 4]. Overall, this requires time polynomial in m', $\log \rho$, and the description bitlength of B_{int}. In turn, this is polynomial in m', k, and the description bitlength of B and $\boldsymbol{t}$, as desired.

Finally, we compute[13] $k' = \frac{\gamma k}{\gamma'}$ and output (B_{final}, k') as the SVP_p instance.

4.2 Proof of Theorem 4.1

In order to prove Theorem 4.1, we begin by establishing some useful properties of the intermediate lattice $\mathcal{L}_{\text{int}}$ constructed by the algorithm from Section 4.1.

Lemma 4.2. *Fix real numbers $p, \gamma, \gamma' \geqslant 1$ and $\alpha \in (0, 1)$ additionally satisfying*

$$\alpha \leqslant \frac{((\gamma/\gamma')^p - 2)^{1/p}}{\gamma}.$$

Given a γ-CVP_p instance $(B, \boldsymbol{t}, k)$ with $B \in \mathbb{Z}^{m \times n}$ and $\boldsymbol{t} \in \mathbb{Z}^m \setminus \mathcal{L}(B)$ and a $(p, \alpha, d = \gamma k, N \geqslant 10^5 (2m(1+\gamma k))^{(\gamma k)^p}, m', n')$-locally dense lattice, the algorithm from Section 4.1 constructs $\mathcal{L}_{\text{int}} = \mathcal{L}(B_{\text{int}})$ in time polynomial in m', k, $\log \rho$, and the description bitlength of B and $\boldsymbol{t}$ satisfying the following properties with probability at least 0.99:

- *If $(B, \boldsymbol{t}, k)$ is a YES instance of γ-CVP_p, then there are at least N vectors $\boldsymbol{z}$ in $\mathcal{L}_{\text{int}}$ such that $\|\boldsymbol{z}\|_p \leqslant k' = \frac{\gamma k}{\gamma'}$ and whose last coordinate equals 1. We call such vectors* good*;*
- *If $(B, \boldsymbol{t}, k)$ is a NO instance of γ-CVP_p, then there are at most $(2m(1+\gamma k))^{(\gamma k)^p}$ nonzero vectors $\boldsymbol{z}$ in $\mathcal{L}_{\text{int}}$ such that $\|\boldsymbol{z}\|_p \leqslant \gamma k$. We call such vectors* annoying*.*

Proof. The claim about the running time is directly verifiable. We proceed to argue that the two items hold.

First, suppose that $(B, \boldsymbol{t}, k)$ is a YES instance of γ-CVP_p. This means that there is a vector $\boldsymbol{x} \in \mathbb{Z}^n$ such that $\|B\boldsymbol{x} - \boldsymbol{t}\|_p \leqslant k$. Moreover, we know that there are at least $N \geqslant 10^5 \cdot (2m(1+\gamma k))^{(\gamma k)^p}$ vectors $\boldsymbol{y} \in \mathbb{Z}^{n'}$ such that

$$\|A\boldsymbol{y} - \boldsymbol{s}\|_p \leqslant \alpha d.$$

For each such $\boldsymbol{y}$, consider the associated vector $\boldsymbol{z}_{\boldsymbol{y}} = (\boldsymbol{x}, \boldsymbol{y}, 1)$ and note that

$$\|B_{\text{int}}\boldsymbol{z}_{\boldsymbol{y}}\|_p^p = \|B\boldsymbol{x} - \boldsymbol{t}\|_p^p + \|A\boldsymbol{y} - \boldsymbol{s}\|_p^p + 1 \leqslant k^p + (\alpha d)^p + 1 \leqslant \left(\frac{\gamma k}{\gamma'}\right)^p = (k')^p,$$

[13] We can always set γ to be an integer multiple of γ' to ensure that k' is an integer. For the sake of readability, we avoid taking floors and ceilings.

where the last inequality follows by the constraints on γ, γ', α in the lemma statement and the fact that $d = \gamma k$. Since the last coordinate of $B_{\text{int}}\boldsymbol{z_y}$ is always 1, we conclude that there are at least N good vectors.

On the other hand, suppose that $(B, \boldsymbol{t}, k)$ is a NO instance of γ-CVP$_p$. This means that for every vector $\boldsymbol{x} \in \mathbb{Z}^n$ and $\alpha \in \mathbb{Z} \setminus \{0\}$ it holds that $\|B\boldsymbol{x} - \alpha\boldsymbol{t}\|_p > \gamma k$. Consider any vector $\boldsymbol{z} = (\boldsymbol{x}, \boldsymbol{u}, \alpha)$. Note that every annoying vector $\boldsymbol{z}$ must have $\alpha = 0$ by this property. Furthermore, because $\alpha = 0$, it must also be the case that $\boldsymbol{u} = 0$, since $\|A\boldsymbol{u}\|_p > \gamma k$ for all $\boldsymbol{u} \in \mathbb{Z}^{n'} \setminus \{\boldsymbol{0}\}$. Therefore, the number of annoying vectors is upper bounded by

$$|\mathcal{L}(B) \cap \mathcal{B}_m^{(p)}(\gamma k)| \leqslant \binom{m}{(\gamma k)^p}(1 + 2\gamma k)^{(\gamma k)^p} \leqslant (2m(1 + \gamma k))^{(\gamma k)^p}. \qquad \square$$

We are now ready to prove Theorem 4.1 with the help of Lemma 4.2.

Proof of Theorem 4.1. The claims regarding the time required to construct $\mathcal{L}_{\text{final}}$ and the bound on k' are directly verifiable. We proceed to argue the two items of the theorem statement.

Recall that $\mathcal{L}_{\text{final}}$ is defined as the random sublattice

$$\mathcal{L}_{\text{final}} = \{\boldsymbol{w} \in \mathcal{L}_{\text{int}} : \langle \boldsymbol{v}, \boldsymbol{w} \rangle = 0 \pmod{\rho}\},$$

where ρ is a prime in the interval $(100(2m(1 + \gamma k))^{(\gamma k)^p}, 200(2m(1 + \gamma k))^{(\gamma k)^p}]$ and the entries of $\boldsymbol{v} \in \mathbb{Z}^{m+m'+1}$ are sampled independently and uniformly at random from $\{0, \ldots, \rho - 1\}$.

Suppose that $(B, \boldsymbol{t}, k)$ is a YES instance of γ-CVP$_p$. Our goal is to show that, with probability at least 0.99 over the randomness of the algorithm, there is $\boldsymbol{w'} \in \mathcal{L}_{\text{final}}$ such that $\|\boldsymbol{w'}\|_p \leqslant k' = \frac{\gamma k}{\gamma'}$. In this case, Lemma 4.2 ensures that there are at least $N = 10^5 \cdot (2m(1 + \gamma k))^{(\gamma k)^p} \geqslant 100\rho$ good vectors $\boldsymbol{w} \in \mathcal{L}_{\text{int}}$ such that $\|\boldsymbol{w}\|_p \leqslant k'$ and whose last coordinate equals 1. Then, it follows that any two distinct good vectors $\boldsymbol{w}^{(1)}, \boldsymbol{w}^{(2)} \in \mathcal{L}_{\text{int}}$ are linearly independent modulo ρ, because their last coordinates equal 1 and all their entries are bounded in absolute value by $k' < \rho/2$. As a result, the random variables $\mathbf{1}_{\{\langle \boldsymbol{v}, \boldsymbol{w}^{(1)} \rangle = 0 \pmod{\rho}\}}$ and $\mathbf{1}_{\{\langle \boldsymbol{v}, \boldsymbol{w}^{(2)} \rangle = 0 \pmod{\rho}\}}$ are independent. Since all these (at least N) pairwise independent random variables follow a Bernoulli distribution with success probability exactly $1/\rho$, Lemma 2.1 implies that the probability over the sampling of $\boldsymbol{v}$ that there exists at least one good vector $\boldsymbol{w} \in \mathcal{L}_{\text{int}}$ such that $\langle \boldsymbol{v}, \boldsymbol{w} \rangle = 0 \pmod{\rho}$, and hence $\boldsymbol{w} \in \mathcal{L}_{\text{final}}$, is at least

$$1 - \rho/N \geqslant 0.99.$$

It follows that (B_{final}, k') is a YES instance of γ'-SVP$_p$ with probability at least 0.99.

Now, suppose that $(B, \boldsymbol{t}, k)$ is a NO instance of γ-CVP$_p$. Lemma 4.2 guarantees that there are at most $N' = (2m(1 + \gamma k))^{(\gamma k)^p}$ annoying nonzero vectors $\boldsymbol{w} \in \mathcal{L}_{\text{int}}$ such that $\|\boldsymbol{w}\|_p \leqslant \gamma k$. For any given nonzero integer vector $\boldsymbol{w}$, the probability (over the sampling of $\boldsymbol{v}$) that $\langle \boldsymbol{v}, \boldsymbol{w} \rangle = 0 \pmod{\rho}$ is exactly $1/\rho$. Consequently, a union bound over the at most N' annoying vectors in $\mathcal{L}_{\text{int}}$ shows that the probability that there exists a vector $\boldsymbol{w'} \in \mathcal{L}_{\text{final}}$ such that $\|\boldsymbol{w'}\|_p \leqslant \gamma k$ is at most

$$N'/\rho \leqslant 0.01.$$

It follows that (B_{final}, k') is a NO instance of γ'-SVP$_p$ with probability at least 0.99. $\square$

4.3 Finalizing the reduction

In this section we provide a randomized construction of locally dense lattices based on Construction A lattices stemming from Reed-Solomon codes, which can be combined with Theorem 4.1

to yield the desired randomized FPT reduction with two-sided error and without advice. More precisely, we prove the following theorem.

Theorem 4.3. *Fix real numbers $p \geqslant 1$ and $\gamma' \in [1, 2^{1/p})$. Let $\varepsilon = (\gamma')^{-p} - 1/2 > 0$ and set*[14]

$$\gamma = \left\lceil \max\left(12/\varepsilon, \frac{1}{(1+\varepsilon/2)^{1/p} - 1}\right)\right\rceil.$$

There exists a randomized algorithm which when given as input positive integers m and k runs in time $\mathrm{poly}(m,k)$ and outputs with probability at least 0.99 a $(p, \alpha, d, N, m', n')$-locally dense lattice $(A, \boldsymbol{s})$, where

$$\begin{aligned}
&\alpha = \left(\frac{1}{(\gamma')^p} - \frac{2}{\gamma^p}\right)^{1/p},\\
&d = \gamma k,\\
&N = (2m(1+\gamma k))^{3(\gamma k)^p} \geqslant 10^5 \cdot (2m(1+\gamma k))^{(\gamma k)^p},\\
&n', m' = \mathrm{poly}(m,k),
\end{aligned}$$

provided that m is sufficiently large compared to p.

Combining Theorems 2.11, 4.1 and 4.3 yields Theorem 1.2. Similarly, observing that $k' = O(k)$ in Theorem 4.1, combining Theorems 2.12, 4.1 and 4.3 yields Theorem 1.6.

4.3.1 Locally dense lattices from Reed-Solomon codes

The locally dense lattices described in Theorem 4.3 are obtained via Construction A lattices based on Reed-Solomon codes, which were analyzed in [BP23].

For a fixed prime q, we define the Reed-Solomon code with block length q and dimension ℓ, which we denote by $\mathrm{RS}_{q,\ell}$, as

$$\mathrm{RS}_{q,\ell} = \left\{(f(\zeta))_{\zeta \in \mathbb{F}_q} : f \in \mathbb{F}_q[x], \deg(f) < \ell\right\}.$$

Note that $\ell \leqslant q$. It is well known that it is possible to construct a generator matrix for $\mathrm{RS}_{q,\ell}$ in time $\mathrm{poly}(q)$.

Given a Reed-Solomon code $\mathrm{RS}_{q,\ell}$, we define the associated Reed-Solomon (Construction A) lattice $\mathcal{L}_{\mathrm{RS}_{q,\ell}}$ as

$$\mathcal{L}_{\mathrm{RS}_{q,\ell}} = \{\boldsymbol{x} \in \mathbb{Z}^q : \boldsymbol{x} \pmod q \in \mathrm{RS}_{q,\ell}\} = \mathrm{RS}_{q,\ell} + q\mathbb{Z}^q.$$

Since we can construct a generator matrix of $\mathrm{RS}_{q,\ell}$ in time $\mathrm{poly}(q)$, we can also construct a basis of $\mathcal{L}_{\mathrm{RS}_{q,\ell}}$ in time $\mathrm{poly}(q)$. Bennett and Peikert established two important properties of $\mathcal{L}_{\mathrm{RS}}$.

Lemma 4.4 ([BP23, Theorem 3.1, adapted])**.** *Suppose that $\ell \leqslant q/2$. Then, it holds that*

$$\lambda_1^{(p)}(\mathcal{L}_{\mathrm{RS}_{q,\ell}}) \geqslant (2\ell)^{1/p}$$

for every $p \geqslant 1$.

Lemma 4.5 ([BP23, Lemma 3.3, adapted])**.** *Fix $\delta > 0$ and denote the set of vectors in $\{0,1\}^q$ of Hamming weight w by $B_{q,w}$. Sample $\boldsymbol{s}$ uniformly at random from $B_{q,w}$. Then,*

$$\Pr\left[\left|(\mathcal{L}_{\mathrm{RS}_{q,\ell}} - \boldsymbol{s}) \cap B_{q,w}\right| \geqslant \delta \cdot \binom{q}{w}/q^\ell\right] \geqslant 1 - \delta.$$

[14] We made no effort to optimize constants.

We now move to prove Theorem 4.3.

Proof of Theorem 4.3. Fix real numbers $p \geqslant 1$ and $\gamma' \in [1, 2^{1/p})$, and set $\varepsilon = (\gamma')^{-p} - 1/2$ and $\gamma = \left\lceil \max\left(12/\varepsilon, \frac{1}{(1+\varepsilon/2)^{1/p}-1}\right)\right\rceil$. Suppose that we are given k and m as inputs. Choose integers

$$\ell = \left\lceil \frac{(1+\gamma k)^p}{2} \right\rceil,$$
$$w = \left\lfloor ((\gamma/\gamma')^p - 2)k^p \right\rfloor,$$

and q to be the smallest prime larger than

$$(300w(\gamma k)^p \cdot (2m(1+\gamma k)))^{9/\varepsilon} = \mathrm{poly}(m,k).$$

Note that $q = \mathrm{poly}(m,k)$ by Bertrand's postulate and that we can naively verify whether q is prime in time $\mathrm{poly}(\log m, \log k)$ [AKS04]. We consider the following candidate construction of a locally dense lattice $(A, \boldsymbol{s})$, which runs in overall time $\mathrm{poly}(q) = \mathrm{poly}(m,k)$:

1. Set A to be a basis of $\mathcal{L}_{\mathrm{RS}_{q,\ell}}$;
2. Sample $\boldsymbol{s}$ uniformly at random from $B_{q,w}$. (Recall that $B_{q,w}$ denotes the set of vectors in $\{0,1\}^q$ of Hamming weight w.)

We now argue that $(A, \boldsymbol{s})$ is a $(p, \alpha, d, N, m', n')$-locally dense lattice with the properties described in Theorem 4.3 with probability at least 0.99 over the sampling of $(A, \boldsymbol{s})$. First, note that $m' = q$ and $n' = \ell$, and so it can be directly verified that $n', m' = \mathrm{poly}(m,k)$. To bound d, note that $\ell \leqslant q/2$. Therefore, Lemma 4.4 and the choice of ℓ above guarantee that

$$\lambda_1^{(p)}(\mathcal{L}(A)) \geqslant (2\ell)^{1/p} > \gamma k,$$

and so we may take $d = \gamma k$. It remains to show that

$$|(\mathcal{L}(A) - \boldsymbol{s}) \cap \mathcal{B}_q^{(p)}(\alpha d)| \geqslant N$$

with probability at least 0.99 over the sampling of $\boldsymbol{s}$, where

$$\alpha = \frac{((\gamma/\gamma')^p - 2)^{1/p}}{\gamma},$$
$$N = (2m(1+\gamma k))^{3(\gamma k)^p}.$$

Since $w \leqslant (\alpha d)^p$ and $B_{q,w} \subseteq \mathcal{B}_q^{(p)}(\alpha d)$, it suffices to show that

$$|(\mathcal{L}(A) - \boldsymbol{s}) \cap B_{q,w}| \geqslant N$$

with probability at least 0.99 over the sampling of $\boldsymbol{s}$. Invoking Lemma 4.5 with $\delta = 0.01$ shows that

$$\Pr\left[|(\mathcal{L}(A) - \boldsymbol{s}) \cap B_{q,w}| \geqslant \frac{\binom{q}{w}}{100q^\ell} \right] \geqslant 0.99.$$

We claim that $\frac{\binom{q}{w}}{100q^\ell} \geqslant N$ with the choices of ℓ, w, and q above, which concludes the proof of Theorem 4.3. To see this, first note that

$$\begin{aligned}
w - \ell &= \lfloor ((\gamma/\gamma')^p - 2)k^p \rfloor - \left\lceil \frac{(1+\gamma k)^p}{2} \right\rceil \\
&\geqslant ((\gamma/\gamma')^p - 2)k^p - \frac{(1+\gamma k)^p}{2} - 2 \\
&\geqslant ((1/2+\varepsilon)\gamma^p - 2)k^p - (1/2+\varepsilon/2)(\gamma k)^p - 2 && (7) \\
&= \frac{\varepsilon}{2}(\gamma k)^p - 2k^p - 2 \\
&\geqslant \frac{\varepsilon}{2}(\gamma k)^p - 4k^p && (8) \\
&\geqslant \frac{\varepsilon}{3}(\gamma k)^p, && (9)
\end{aligned}$$

where Equation (7) uses the observation that $(1+\gamma k)^p \leqslant (1+\varepsilon/2)(\gamma k)^p$, which follows from the fact that $\gamma \geqslant \frac{1}{(1+\varepsilon/2)^{1/p}-1}$, Equation (8) holds because $k \geqslant 1$, and Equation (9) uses the fact that $\gamma \geqslant 12/\varepsilon$.

Then, we have that

$$\begin{aligned}
\frac{\binom{q}{w}}{100q^\ell} &\geqslant \frac{q^{w-\ell}}{100w^w} && (10) \\
&\geqslant \frac{q^{\frac{\varepsilon}{3}(\gamma k)^p}}{100w^w} && (11) \\
&\geqslant \frac{(300w(2m(1+\gamma k)))^{3(\gamma k)^p}}{100w^w} && (12) \\
&\geqslant (2m(1+\gamma k))^{3(\gamma k)^p} && (13) \\
&= N.
\end{aligned}$$

Equation (10) holds because $\binom{q}{w} \geqslant (q/w)^w$. Equation (11) follows from the lower bound on $w - \ell$ from Equation (9). Equation (12) holds since $q > (300w(\gamma k)^p \cdot (2m(1+\gamma k)))^{9/\varepsilon}$. Finally, Equation (13) holds because $w < 3(\gamma k)^p$. □

5 W[1]-hardness of SVP_p for any approximation factor

In this section we analyze an FPT reduction from approximate NCP_2 to approximate SVP_p which when combined with results of Haviv and Regev [HR12] leads to Theorem 1.3, which we restate here.

Theorem 1.3. *For any fixed $p \in (1, \infty)$ and constant $\gamma \geqslant 1$, γ-SVP_p is $\mathsf{W}[1]$-hard under randomized FPT reductions with two-sided error.*

5.1 The Haviv-Regev conditions for tensoring of SVP instances

We will use the following generalization of a result of Haviv and Regev [HR12] which establishes conditions under which an SVP instance behaves well under tensoring.

Lemma 5.1. *Fix an integer $c \geqslant 1$ and real numbers $p, \gamma \geqslant 1$. Suppose that (B, k) with $B \in \mathbb{Z}^{m \times n}$ and $k \in \mathbb{Z}^+$ is an instance of γ-SVP_p with the additional property that if (B, k) is a NO instance*

of γ-SVP$_p$, then every nonzero vector $\boldsymbol{w} \in \mathcal{L}(B)$ satisfies at least one of the following conditions, where $d = \gamma k$:

- $\|\boldsymbol{w}\|_0 > d^p$;
- $\boldsymbol{w} \in 2\mathbb{Z}^m$ *and* $\|\boldsymbol{w}\|_0 > d^p/2^p$;
- $\boldsymbol{w} \in 2\mathbb{Z}^m$ *and* $\|\boldsymbol{w}\|_p > d^{c+3p/2}$.

Then, $(B^{\otimes c}, k^c)$ is a YES (resp. NO) instance of γ^c-SVP$_p$ if (B, k) is a YES (resp. NO) instance of γ-SVP$_p$, where $B^{\otimes c}$ denotes the c-fold tensor product of B with itself.

Haviv and Regev [HR12, Lemma 3.4] proved a similar result for the case $p = 2$ only. It turns out to be not hard to generalize for any $p \geqslant 1$. Note, however, that the conditions of Lemma 5.1 depend a priori on the (constant) number of times c we will be tensoring the base SVP$_p$ instance. We provide a proof of Lemma 5.1 in Appendix B.

5.2 The FPT NCP$_2$ to SVP$_p$ reduction amenable to tensoring

We proceed to describe an FPT reduction from approximate NCP$_2$, which we know is W[1]-hard by Theorem 2.7, to approximate SVP$_p$ that yields the following result.

Theorem 5.2. *Fix an even integer $\gamma \geqslant 2$, an integer $c \geqslant 1$, and real numbers $p > 1$, $\gamma' \geqslant 1$, and $\alpha \in (1/2 + 2^{-p}, 1)$ additionally satisfying*

$$\gamma' < \left(\frac{\gamma}{2^p + 1 + \alpha\gamma}\right)^{1/p}.$$

Then, there is a randomized algorithm which, for m large enough, on input a γ-NCP$_2$ instance $(G, \boldsymbol{t}, k)$ with $G \in \mathbb{F}_2^{m \times n}$, $\boldsymbol{t} \in \mathbb{F}_2^m$, and $k \in \mathbb{Z}^+$ outputs in time poly(m) *an instance (B_{final}, k') of γ'-SVP$_p$ with $k' = (2^p k + \alpha\gamma k + 1)^{1/p} < (\gamma k)^{1/p}$ satisfying the following properties with probability at least* 0.9*:*

- *If $(G, \boldsymbol{t}, k)$ is a YES instance of γ-NCP$_2$, then (B_{final}, k') is a YES instance of γ'-SVP$_p$;*
- *If $(G, \boldsymbol{t}, k)$ is a NO instance of γ-NCP$_2$, then (B_{final}, k') is a NO instance of γ'-SVP$_p$ such that every nonzero vector $\boldsymbol{w} \in \mathcal{L}(B_{\text{final}})$ satisfies at least one of the following conditions:*
  - $\|\boldsymbol{w}\|_0 > (\gamma' k')^p$;
  - $\boldsymbol{w} \in 2\mathbb{Z}^m$ *and* $\|\boldsymbol{w}\|_0 > (\gamma' k')^p/2^p$;
  - $\boldsymbol{w} \in 2\mathbb{Z}^m$ *and* $\|\boldsymbol{w}\|_p > (\gamma' k')^{c+3p/2}$.

Combining Theorem 5.2 with Lemma 5.1 and Theorem 2.7 immediately yields Theorem 1.3. This is because Lemma 5.1 guarantees that we can directly tensor the γ'-SVP$_p$ instances from Theorem 3.1 with $\gamma' > 1$ an arbitrary (constant) number of times $c \geqslant 1$ to conclude that γ''-SVP$_p$ is W[1]-hard for any constant $\gamma'' \geqslant 1$.

We proceed to describe the algorithm we use to prove Theorem 5.2. First, we need to set up some auxiliary objects and lemmas. Suppose that we are given as input an instance $(G, \boldsymbol{t}, k)$ of γ-NCP$_2$, where $G \in \mathbb{F}_2^{m \times n}$, $\boldsymbol{t} \in \mathbb{F}_2^m$, and $k \in \mathbb{Z}^+$. Let $d = \gamma k$. We denote by $B_{\text{NCP}} \in \mathbb{Z}^{m \times m}$ the basis of the Construction A lattice

$$\mathcal{L}_{\text{NCP}} = \mathcal{C}(G) + 2\mathbb{Z}^m,$$

which we can compute in time $\mathrm{poly}(m)$.

With some hindsight, set m' to be the smallest integer of the form $2^r - 1$ larger than

$$\max\left(m+1, (10^8 d^{12c})^{\frac{1}{\alpha-(1/2+2^{-p})}}\right) = \mathrm{poly}(m),$$

and let $\mathcal{C}_{\mathrm{BCH}} \subseteq \mathbb{F}_2^{m'}$ be a binary BCH code with minimum distance at least $d+1$ and codimension

$$h \leqslant \left\lceil \frac{d}{2} \right\rceil \log(m'+1) = \frac{d}{2}\log(m'+1)$$

guaranteed by Theorem 3.4. We denote by $B_{\mathrm{BCH}} \in \mathbb{Z}^{m' \times m'}$ the basis of the Construction A lattice

$$\mathcal{L}_{\mathrm{BCH}} = \mathcal{C}_{\mathrm{BCH}} + 2\mathbb{Z}^{m'}.$$

Note that we can compute a basis of $\mathcal{L}_{\mathrm{BCH}}$ in time $\mathrm{poly}(m') = \mathrm{poly}(m)$. Furthermore, we sample a target vector $\boldsymbol{s} \in \mathbb{F}_2^{m'}$ uniformly at random from $B_{m',\alpha d}$, where we recall that $B_{m',\alpha d}$ is the set of vectors in $\{0,1\}^{m'}$ with Hamming weight αd. As in previous sections, the tuple $(B_{\mathrm{BCH}}, \boldsymbol{s})$ satisfies a local density property with high probability over the sampling of $\boldsymbol{s}$, as described in the following lemma.

Lemma 5.3. *It holds with probability at least* 0.99 *over the sampling of* $\boldsymbol{s}$ *from* $B_{m',\alpha d}$ *that*

$$|(\mathcal{L}_{\mathrm{BCH}} - \boldsymbol{s}) \cap \mathcal{B}_{m'}^{(p)}((\alpha d)^{1/p})| \geqslant \frac{\binom{m'}{\alpha d}}{100(m'+1)^{d/2}} =: N.$$

Proof. Consider the vectors $\boldsymbol{v}$ of $\mathcal{L}_{\mathrm{BCH}}$ which lie in $\mathcal{C}_{\mathrm{BCH}}$ (seen as a subset of $\mathbb{Z}^{m'}$). Since both $\boldsymbol{v}$ and $\boldsymbol{s}$ lie in $\{0,1\}^{m'}$, it follows that

$$\|\boldsymbol{v} - \boldsymbol{s}\|_0 = \|\boldsymbol{v} - \boldsymbol{s} \pmod 2\|_0 = \|\boldsymbol{v} - \boldsymbol{s}\|_p^p.$$

This implies that

$$|(\mathcal{L}_{\mathrm{BCH}} - \boldsymbol{s}) \cap \mathcal{B}_{m'}^{(p)}((\alpha d)^{1/p})| \geqslant |(\mathcal{C}_{\mathrm{BCH}} - \boldsymbol{s} \pmod 2) \cap \mathcal{B}_{2,m'}(\alpha d)|. \tag{14}$$

Following the derivation of Equation (6) from the proof of Theorem 3.3 on locally dense codes with $q = 2$ guarantees that with probability at least 0.99 over the sampling of $\boldsymbol{s}$ it holds that $|(\mathcal{C}_{\mathrm{BCH}} - \boldsymbol{s} \pmod 2) \cap \mathcal{B}_{2,m'}(\alpha d)| \geqslant \frac{\binom{m'}{\alpha d}}{100(m'+1)^{d/2}}$. Combining this with Equation (14) yields the desired result. □

Equipped with the above, we consider the intermediate lattice $\mathcal{L}_{\mathrm{int}}$ generated by the basis

$$B_{\mathrm{int}} := \begin{pmatrix} 2B_{\mathrm{NCP}} & 0_{m\times m'} & -2\boldsymbol{t} \\ 0_{m'\times m} & B_{\mathrm{BCH}} & -\boldsymbol{s} \\ \mathbf{0}_m & \mathbf{0}_{m'} & 1 \end{pmatrix} \in \mathbb{Z}^{(m+m'+1)\times(m+m'+1)}. \tag{15}$$

The bottom $(0,\ldots,0,1)$ row is added to ensure that B_{int} has full column rank over $\mathbb{R}$.

Then, we add a random constraint $\mathcal{L}_{\mathrm{int}}$ to obtain our final SVP instance. More precisely, we set ρ to be a prime in the interval $(N/100, N/50]$ (as discussed in Section 4, we can sample ρ with high probability in time $\mathrm{poly}(m)$), sample a vector $\boldsymbol{v} \in \mathbb{Z}^{m+m'+1}$ by sampling each entry of $\boldsymbol{v}$

independently and uniformly at random from $\{0, \ldots, \rho - 1\}$, and construct in time $\mathrm{poly}(m)$ a basis B_{final} of the random sublattice

$$\mathcal{L}_{\mathrm{final}} = \{\boldsymbol{w} \in \mathcal{L}_{\mathrm{int}} : \langle \boldsymbol{v}, \boldsymbol{w} \rangle = 0 \pmod{\rho}\}$$

by following the procedure discussed in Section 4.1. Then, we set $k' = (2^p k + \alpha d + 1)^{1/p}$ and output (B_{final}, k') as our γ'-SVP$_p$ instance.[15]

5.3 Proof of Theorem 5.2

In order to prove Theorem 5.2, we begin by establishing some useful properties of the intermediate lattice $\mathcal{L}_{\mathrm{int}}$ captured in the following lemma.

Lemma 5.4. *Fix an even integer $\gamma \geqslant 2$, an integer $c \geqslant 1$, and real numbers $p > 1$, $\gamma' \geqslant 1$, and $\alpha \in (1/2 + 2^{-p}, 1)$ additionally satisfying*

$$\gamma' < \left(\frac{\gamma}{2^p + 1 + \alpha\gamma}\right)^{1/p}.$$

Given a γ-NCP$_2$ instance $(G, \boldsymbol{t}, k)$ with $G \in \mathbb{F}_2^{m \times n}$, $\boldsymbol{t} \in \mathbb{Z}^m$, and $k \in \mathbb{Z}^+$, the algorithm from Section 5.2 constructs $\mathcal{L}_{\mathrm{int}} = \mathcal{L}(B_{\mathrm{int}}) \subseteq \mathbb{Z}^{m+m'+1}$ in time $\mathrm{poly}(m)$ satisfying the following properties with probability at least 0.99, where we recall that $d = \gamma k$ and $k' = (2^p k + \alpha d + 1)^{1/p}$:

- *If $(G, \boldsymbol{t}, k)$ is a YES instance of γ-NCP$_2$, then there are at least $N = \frac{\binom{m'}{\alpha d}}{100(m'+1)^{d/2}}$ vectors $\boldsymbol{w}$ in $\mathcal{L}_{\mathrm{int}}$ such that $\|\boldsymbol{w}\|_p \leqslant k'$ and whose last coordinate equals 1. We call such vectors* good*;*
- *If $(G, \boldsymbol{t}, k)$ is a NO instance of γ-NCP$_2$, then there are at most $A \leqslant 10^{-5} N$ nonzero vectors $\boldsymbol{w}$ in $\mathcal{L}_{\mathrm{int}}$ that satisfy* all *of the following properties:*
  - $\|\boldsymbol{w}\|_0 \leqslant (\gamma' k')^p$*;*
  - *Either $\boldsymbol{w} \notin 2\mathbb{Z}^{m+m'+1}$ or $\|\boldsymbol{w}\|_0 \leqslant (\gamma' k')^p / 2^p$;*
  - *Either $\boldsymbol{w} \notin 2\mathbb{Z}^{m+m'+1}$ or $\|\boldsymbol{w}\|_p \leqslant (\gamma' k')^{c+3p/2}$.*

  We call such vectors annoying.[16]

Proof. The claim about the running time of the algorithm is directly verifiable. We proceed to argue the two items in the lemma statement.

Suppose that $(G, \boldsymbol{t}, k)$ is a YES instance of γ-NCP$_2$. This means that there is a codeword $\boldsymbol{c} \in \mathcal{C}(G)$ such that

$$\|\boldsymbol{c} - \boldsymbol{t}\|_0 \leqslant k.$$

Noting that $\mathcal{C}(G) \subseteq \mathcal{L}_{\mathrm{NCP}}$ (when seen as a subset of $\{0,1\}^m \subseteq \mathbb{Z}^m$), we conclude that there is $\boldsymbol{x} \in \mathbb{Z}^m$ such that $B_{\mathrm{NCP}}\boldsymbol{x} = \boldsymbol{c}$, and so

$$\|B_{\mathrm{NCP}}\boldsymbol{x} - \boldsymbol{t}\|_p^p \leqslant k.$$

[15] Our choice of k' may not be an integer. For the sake of readability, we avoid working through the argument with floors and ceilings. It is also relevant to note that γ can be chosen so that our choice of k' is the pth root of an integer, which already matches the requirements of the definition of approximate SVP$_p$ in [BBE+21].

[16] Annoying vectors are the ones that do not satisfy the properties of NO instances laid out in Theorem 5.2.

Moreover, by Lemma 5.3 we also know that with probability at least 0.99 there are at least $N = \frac{\binom{m'}{\alpha d}}{100(m'+1)^{d/2}}$ vectors $\boldsymbol{y} \in \mathbb{Z}^{m'}$ such that

$$\|B_{\mathrm{BCH}}\boldsymbol{y} - \boldsymbol{s}\|_p^p \leqslant \alpha d.$$

For each such good $\boldsymbol{y}$, consider the vector $\boldsymbol{z_y} = (\boldsymbol{x}, \boldsymbol{y}, 1)$. Then, we have that

$$\|B_{\mathrm{int}}\boldsymbol{z_y}\|_p^p = \|2B_{\mathrm{NCP}}\boldsymbol{x} - 2\boldsymbol{t}\|_p^p + \|B_{\mathrm{BCH}}\boldsymbol{y} - \boldsymbol{s}\|_p^p + 1 \leqslant 2^p k + \alpha d + 1,$$

and so $\|B_{\mathrm{int}}\boldsymbol{z_y}\|_p \leqslant (2^p k + \alpha d + 1)^{1/p} = k'$. Furthermore, the last coordinate of $B_{\mathrm{int}}\boldsymbol{z_y}$ is always 1. As a result, there are at least N good vectors in $\mathcal{L}_{\mathrm{int}}$.

On the other hand, suppose that $(G, \boldsymbol{t}, k)$ is a NO instance of γ-NCP$_2$. This means that for every $\boldsymbol{c} \in \mathcal{C}(G)$ it holds that

$$\|\boldsymbol{c} - \boldsymbol{t}\|_0 > d = \gamma k \geqslant (\gamma' k')^p,$$

where the last inequality follows by our choice of k' and the constraints on p, γ, γ', and α. Recall that our goal is to bound the number of annoying vectors in $\mathcal{L}_{\mathrm{int}}$ appropriately. Consider an arbitrary vector $\boldsymbol{z} = (\boldsymbol{x}, \boldsymbol{y}, \beta) \in \mathbb{Z}^{m+m'+1}$. We proceed by case analysis:

1. $\beta \notin 2\mathbb{Z}$: In this case, we have

   $$\|B_{\mathrm{int}}\boldsymbol{z}\|_0 \geqslant \|B_{\mathrm{NCP}}\boldsymbol{x} - \beta\boldsymbol{t}\|_0 \geqslant \|G\boldsymbol{x} - \boldsymbol{t} \pmod 2\|_0 > d \geqslant (\gamma' k')^p,$$

   and so no vector of this form is annoying.

2. $\beta \in 2\mathbb{Z}$ and $B_{\mathrm{BCH}}\boldsymbol{y} \notin 2\mathbb{Z}^{m'}$: In this case, we have

   $$\|B_{\mathrm{int}}\boldsymbol{z}\|_0 \geqslant \|B_{\mathrm{BCH}}\boldsymbol{y} - \beta\boldsymbol{s}\|_0 \geqslant \|B_{\mathrm{BCH}}\boldsymbol{y} \pmod 2\|_0 > d \geqslant (\gamma' k')^p,$$

   and so no vector of this form is annoying. The third inequality uses the fact that $B_{\mathrm{BCH}}\boldsymbol{y} \pmod 2$ is a nonzero codeword of $\mathcal{C}_{\mathrm{BCH}}$, which has minimum distance larger than $d = \gamma k$.

3. $\beta \in 2\mathbb{Z}$ and $B_{\mathrm{BCH}}\boldsymbol{y} \in 2\mathbb{Z}^{m'}$: In this case, it holds that all coordinates of $B_{\mathrm{int}}\boldsymbol{z}$ are even. Therefore, in order for $B_{\mathrm{int}}\boldsymbol{z}$ to be annoying it must be that $\|B_{\mathrm{int}}\boldsymbol{z}\|_0 \leqslant (\gamma' k')^p / 2^p \leqslant d/2^p$ and $\|B_{\mathrm{int}}\boldsymbol{z}\|_p \leqslant (\gamma' k')^{c+3p/2} \leqslant d^{3c}$. There are at most

   $$(2d^{3c} + 1)^{d/2^p} \binom{m + m' + 1}{d/2^p}$$

   vectors in $\mathcal{L}_{\mathrm{int}}$ with these properties.

We conclude that there are at most $A = (2d^{3c} + 1)^{d/2^p} \binom{m+m'+1}{d/2^p}$ annoying vectors in $\mathcal{L}_{\mathrm{int}}$. Finally, we claim that, since we chose m' to be larger than $\max\left(m + 1, (10^8 d^{12c})^{\frac{1}{\alpha - (1/2 + 2^{-p})}}\right)$ in Section 5.2, it follows that $A \leqslant 10^{-5} N$. To see this, note that

$$A \leqslant (2d^{3c} + 1)^{d/2^p} (m + m' + 1)^{d/2^p} \leqslant (3d^{3c})^d (2m')^{d/2^p}$$

and

$$N \geqslant \frac{(m')^{(\alpha - 1/2)d}}{100 \cdot 2^d \cdot d^d},$$

where we have used the fact that $m' \geqslant m + 1$. Therefore, after simple algebraic manipulation, it follows that the desired inequality holds whenever

$$(m')^{(\alpha - (1/2 + 2^{-p}))d} \geqslant 10^7 \cdot (6d^{4c})^d,$$

which is in turn satisfied by our choice of m'. □

Proof of Theorem 5.2. Theorem 5.2 follows by combining Lemma 5.4 with a standard sparsification argument, as carried out in the proof of Theorem 4.1. To avoid repetition, we omit the full argument here and simply give a sketch.

Let $\mathcal{L}_{\text{int}} := \mathcal{L}(B_{\text{int}})$, where B_{int} is as defined in Equation (15). First, we note that the random sublattice $\mathcal{L}_{\text{final}}$ of $\mathcal{L}_{\text{int}}$ is defined in Section 5.2 with respect to a prime ρ that satisfies $100A \leqslant \rho \leqslant N/100$. Additionally, we note that there at least N distinct good vectors of the form $(\boldsymbol{x}^T, \boldsymbol{y}_i^T, 1)^T \in \mathcal{L}_{\text{int}}$ such that $\boldsymbol{y}_i \in (\mathcal{L}(B_{\text{BCH}}) - \boldsymbol{s})$ is a binary vector and $\boldsymbol{y}_i, \boldsymbol{y}_j$ have 1s in distinct coordinates for all $i, j \in [N], i \neq j$. Then, by Lemma 2.1 and the fact that any two such distinct good vectors $(\boldsymbol{x}^T, \boldsymbol{y}_i^T, 1)^T, (\boldsymbol{x}^T, \boldsymbol{y}_j^T, 1)^T$ are linearly independent modulo ρ (which follows from the fact that $\boldsymbol{y}_i, \boldsymbol{y}_j$ have 1s in distinct coordinates), with probability at least 0.99 in the YES case there is at least one good vector left in $\mathcal{L}_{\text{final}}$, in which case (B_{final}, k') is a YES instance of γ'-SVP$_p$. Moreover, using the fact that each annoying vector will be kept with probability at most $1/\rho$ and taking a union bound, with probability at least 0.99 in the NO case there are no annoying vectors left in $\mathcal{L}_{\text{final}}$, meaning that (B_{final}, k') is a NO instance of γ'-SVP$_p$ with the additional properties outlined in Theorem 5.2. □

References

[ABGS21] Divesh Aggarwal, Huck Bennett, Alexander Golovnev, and Noah Stephens-Davidowitz. Fine-grained hardness of CVP(P) — everything that we can prove (and nothing else). In *SODA*, 2021.

[ABSS97] Sanjeev Arora, László Babai, Jacques Stern, and Z. Sweedyk. The hardness of approximate optima in lattices, codes, and systems of linear equations. *J. Comput. Syst. Sci.*, 54(2):317–331, 1997. Preliminary version in FOCS 1993.

[Ajt98] Miklós Ajtai. The Shortest Vector Problem in L_2 is NP-hard for randomized reductions (extended abstract). In *STOC*, pages 10–19, 1998.

[AK14] Per Austrin and Subhash Khot. A simple deterministic reduction for the gap minimum distance of code problem. *IEEE Transactions on Information Theory*, 60(10):6636–6645, 2014. Preliminary version in ICALP 2011.

[AKS04] Manindra Agrawal, Neeraj Kayal, and Nitin Saxena. PRIMES is in P. *Annals of Mathematics*, pages 781–793, 2004.

[AS18] Divesh Aggarwal and Noah Stephens-Davidowitz. (Gap/S)ETH hardness of SVP. In *STOC*, 2018.

[BBE+21] Arnab Bhattacharyya, Édouard Bonnet, László Egri, Suprovat Ghoshal, Karthik C. S., Bingkai Lin, Pasin Manurangsi, and Dániel Marx. Parameterized intractability of even set and shortest vector problem. *J. ACM*, 68(3), mar 2021.

[BCGR23] Huck Bennett, Mahdi Cheraghchi, Venkatesan Guruswami, and João Ribeiro. Parameterized inapproximability of the minimum distance problem over all fields and the shortest vector problem in all ℓ_p norms. In *Proceedings of the 55th Annual ACM Symposium on Theory of Computing (STOC 2023)*, page 553–566, 2023.

[BGKM18] Arnab Bhattacharyya, Suprovat Ghoshal, Karthik C. S., and Pasin Manurangsi. Parameterized intractability of even set and shortest vector problem from Gap-ETH.

In Ioannis Chatzigiannakis, Christos Kaklamanis, Dániel Marx, and Donald Sannella, editors, *45th International Colloquium on Automata, Languages, and Programming, ICALP 2018*, volume 107 of *LIPIcs*, pages 17:1–17:15. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2018.

[BGS17] Huck Bennett, Alexander Golovnev, and Noah Stephens-Davidowitz. On the quantitative hardness of CVP. In *FOCS*, 2017.

[BMvT78] Elwyn Berlekamp, Robert McEliece, and Henk van Tilborg. On the inherent intractability of certain coding problems (corresp.). *IEEE Transactions on Information Theory*, 24(3):384–386, 1978.

[BP23] Huck Bennett and Chris Peikert. Hardness of the (approximate) shortest vector problem: A simple proof via Reed-Solomon codes. In Nicole Megow and Adam Smith, editors, *Approximation, Randomization, and Combinatorial Optimization. Algorithms and Techniques (APPROX/RANDOM 2023)*, pages 37:1–37:20, 2023.

[BPT22] Huck Bennett, Chris Peikert, and Yi Tang. Improved hardness of BDD and SVP under Gap-(S)ETH. In *ITCS*, 2022.

[BR60] R. C. Bose and Dwijendra K. Ray-Chaudhuri. On a class of error correcting binary group codes. *Inf. Control.*, 3(1):68–79, 1960.

[CFK+15] Marek Cygan, Fedor V. Fomin, Lukasz Kowalik, Daniel Lokshtanov, Dániel Marx, Marcin Pilipczuk, Michal Pilipczuk, and Saket Saurabh. *Parameterized Algorithms*. Springer, 2015.

[CN98] Jin-Yi Cai and Ajay Nerurkar. Approximating the SVP to within a factor $(1+1/\dim^{\epsilon})$ is NP-hard under randomized reductions. In *CCC*, 1998.

[CW12] Qi Cheng and Daqing Wan. A deterministic reduction for the gap minimum distance problem. *IEEE Transactions on Information Theory*, 58(11):6935–6941, 2012. Preliminary version in STOC 2009.

[DF99] Rodney G. Downey and Michael R. Fellows. *Parameterized Complexity*. Monographs in Computer Science. Springer, 1999.

[DF13] Rodney G. Downey and Michael Fellows. *Fundamentals of Parameterized Complexity*. Texts in computer science. Springer, London, England, 2013 edition, 2013.

[DFVW99] Rodney G. Downey, Michael R. Fellows, Alexander Vardy, and Geoff Whittle. The parametrized complexity of some fundamental problems in coding theory. *SIAM Journal on Computing*, 29(2):545–570, 1999.

[Din02] Irit Dinur. Approximating SVP_∞ to within almost-polynomial factors is NP-hard. *Theor. Comput. Sci.*, 285(1):55–71, 2002. Preliminary version in CIAC 2000.

[Din16] Irit Dinur. Mildly exponential reduction from gap 3SAT to polynomial-gap label-cover. 2016. https://eccc.weizmann.ac.il/report/2016/128/.

[DMS03] Ilya Dumer, Daniele Micciancio, and Madhu Sudan. Hardness of approximating the minimum distance of a linear code. *IEEE Trans. Inf. Theory*, 49(1):22–37, 2003. Preliminary version in FOCS 1999.

[FKLM20] Andreas Emil Feldmann, Karthik C. S., Euiwoong Lee, and Pasin Manurangsi. A survey on approximation in parameterized complexity: Hardness and algorithms. *Algorithms*, 13(6), 2020.

[Gur10] Venkatesan Guruswami. Notes 6: Reed-Solomon, BCH, Reed-Muller, and concatenated codes, 2010. Lecture notes for the "Introduction to Coding Theory" course at CMU, available at https://www.cs.cmu.edu/~venkatg/teaching/codingtheory/notes/notes6.pdf.

[Hoc59] Alexis Hocquenghem. Codes correcteurs d'erreurs. *Chiffres*, 2:147–156, 1959.

[HR12] Ishay Haviv and Oded Regev. Tensor-based hardness of the shortest vector problem to within almost polynomial factors. *Theory Comput.*, 8(1):513–531, 2012. Preliminary version in STOC 2007.

[Kho04] Subhash Khot. Lecture 6: Minimum distance of a linear code. Lecture notes for CS 8002: PCPs and Hardness of Approximation, taught at Georgia Tech. Available at https://cs.nyu.edu/~khot/pcp-lecnotes/lec6.ps., 2004.

[Kho05] Subhash Khot. Hardness of approximating the shortest vector problem in lattices. *J. ACM*, 52(5):789–808, sep 2005. Preliminary version in FOCS 2004.

[Lin18] Bingkai Lin. The parameterized complexity of the k-biclique problem. *J. ACM*, 65(5):34:1–34:23, 2018. Preliminary version in SODA 2015.

[LRSZ20] Daniel Lokshtanov, M. S. Ramanujan, Saket Saurabh, and Meirav Zehavi. Parameterized complexity and approximability of directed odd cycle transversal. In *Proceedings of the Thirty-First Annual ACM-SIAM Symposium on Discrete Algorithms (SODA 2020)*, pages 2181–2200, USA, 2020.

[Man20] Pasin Manurangsi. Tight running time lower bounds for strong inapproximability of maximum k-coverage, unique set cover and related problems (via t-wise agreement testing theorem). In *Proceedings of the Thirty-First Annual ACM-SIAM Symposium on Discrete Algorithms*, SODA '20, page 62–81, USA, 2020. Society for Industrial and Applied Mathematics.

[Man22] Pasin Manurangsi. Personal communication, 2022.

[Mic00] Daniele Micciancio. The shortest vector in a lattice is hard to approximate to within some constant. *SIAM J. Comput.*, 30(6):2008–2035, 2000. Preliminary version in FOCS 1998.

[Mic10] Daniele Micciancio. Notes 2: Basic Algorithms, 2010. Lecture notes for the "Lattice Algorithms and Applications" course at UCSD, available at https://cseweb.ucsd.edu/classes/wi10/cse206a/lec2.pdf.

[Mic12] Daniele Micciancio. Inapproximability of the shortest vector problem: Toward a deterministic reduction. *Theory Comput.*, 8(1):487–512, 2012.

[Mic14] Daniele Micciancio. Locally dense codes. In *2014 IEEE 29th Conference on Computational Complexity (CCC)*, pages 90–97, 2014.

[MR17] Pasin Manurangsi and Prasad Raghavendra. A birthday repetition theorem and complexity of approximating dense CSPs. In *44th International Colloquium on Automata, Languages, and Programming, ICALP 2017*, pages 78:1–78:15, 2017.

[RR06] Oded Regev and Ricky Rosen. Lattice problems and norm embeddings. In *STOC*, 2006.

[SV19] Noah Stephens-Davidowitz and Vinod Vaikuntanathan. SETH-hardness of coding problems. In *2019 IEEE 60th Annual Symposium on Foundations of Computer Science (FOCS)*, pages 287–301, 2019.

[Var97] Alexander Vardy. The intractability of computing the minimum distance of a code. *IEEE Transactions on Information Theory*, 43(6):1757–1766, 1997.

[vEB81] Peter van Emde Boas. Another NP-complete partition problem and the complexity of computing short vectors in a lattice. Technical Report, 1981. Available at https://staff.fnwi.uva.nl/p.vanemdeboas/vectors/mi8104c.html.

A Proof of Theorem 3.4

We restate Theorem 3.4 here for convenience.

Theorem A.1 (Restatement of Theorem 3.4, q-ary BCH codes)**.** *Fix a prime power q. Then, given integers $m' = q^r - 1$ and $1 \leqslant d \leqslant m'$ for some positive integer r, it is possible to construct in time $\mathrm{poly}(m')$ a generator matrix $G_{\mathrm{BCH}} \in \mathbb{F}_q^{m' \times n'}$ such that $\mathcal{C}_{\mathrm{BCH}} = \mathcal{C}(G_{\mathrm{BCH}}) \subseteq \mathbb{F}_q^{m'}$ has minimum distance at least d and codimension*

$$m' - n' \leqslant \lceil (d-1)(1-1/q) \rceil \log_q(m'+1).$$

Proof. Let α be a primitive element of $\mathbb{F}_{q^r}$. For a given block length $m' = q^r - 1$ and design distance $d \leqslant m'$, we define the (narrow-sense, primitive) q-ary BCH code $\mathcal{C}_{\mathrm{BCH}} \subseteq \mathbb{F}_q^{m'}$ as

$$\mathcal{C}_{\mathrm{BCH}} = \left\{ (c_0, \ldots, c_{m'-1}) \in \mathbb{F}_q : f(x) = \sum_{i=0}^{m'-1} c_i x^i \in \mathbb{F}_{q^r}[x] \text{ satisfies } f(\alpha^i) = 0 \text{ for } i = 1, \ldots, d-1 \right\}. \tag{16}$$

By [Gur10, Theorem 3], it holds that $\mathcal{C}_{\mathrm{BCH}}$ is a subset of a Reed-Solomon code over $\mathbb{F}_{q^r}$ with block length m' and minimum distance d. Therefore, $\mathcal{C}_{\mathrm{BCH}}$ has minimum distance at least d.

To see the claim about the codimension, note that each constraint of the form $f(\alpha^i) = 0$ over $\mathbb{F}_{q^r}$ corresponds to $r = \log_q(m'+1)$ linear constraints over $\mathbb{F}_q$. Moreover, if $f(\gamma) = 0$ for some $\gamma \in \mathbb{F}_{q^r}$, it follows that

$$0 = f(\gamma)^q = \sum_{i=0}^{m'-1} c_i^q \left(\gamma^i\right)^q = \sum_{i=0}^{m'-1} c_i \left(\gamma^q\right)^i = f(\gamma^q),$$

where we have used the fact that $(\beta_1 + \beta_2)^q = \beta_1^q + \beta_2^q$ for any $\beta_1, \beta_2 \in \mathbb{F}_{q^r}$ and that $\beta^q = \beta$ for all $\beta \in \mathbb{F}_q$ (with the natural embedding of $\mathbb{F}_q$ in $\mathbb{F}_{q^r}$). Therefore, there are at least $\left\lfloor \frac{d-1}{q} \right\rfloor$ redundant constraints over $\mathbb{F}_{q^r}$ in Equation (16). Combining both observations above shows that the codimension of $\mathcal{C}_{\mathrm{BCH}}$ is at most

$$\lceil (d-1)(1-1/q) \rceil \log_q(m'+1).$$

Finally, since each constraint $f(\alpha^i) = 0$ over $\mathbb{F}_{q^r}$ can be transformed into r linear constraints over $\mathbb{F}_q$ in time $\mathrm{poly}(q^r) = \mathrm{poly}(m')$, we can construct the parity-check matrix (and hence the generator matrix G_{BCH}) of $\mathcal{C}_{\mathrm{BCH}}$ in time $\mathrm{poly}(m')$ as well. □

B The Haviv-Regev conditions for general ℓ_p norms

In this section, we prove a generalization of Haviv-Regev's conditions which allow tensoring of approximate SVP_p instances for all $p \geqslant 1$. Our argument follows that of [HR12] for $p = 2$ closely. For convenience, we restate the key lemma here.

Lemma 5.1. *Fix an integer $c \geqslant 1$ and real numbers $p, \gamma \geqslant 1$. Suppose that (B, k) with $B \in \mathbb{Z}^{m \times n}$ and $k \in \mathbb{Z}^+$ is an instance of γ-SVP_p with the additional property that if (B, k) is a NO instance of γ-SVP_p, then every nonzero vector $\boldsymbol{w} \in \mathcal{L}(B)$ satisfies at least one of the following conditions, where $d = \gamma k$:*

- $\|\boldsymbol{w}\|_0 > d^p$;
- $\boldsymbol{w} \in 2\mathbb{Z}^m$ *and* $\|\boldsymbol{w}\|_0 > d^p/2^p$;
- $\boldsymbol{w} \in 2\mathbb{Z}^m$ *and* $\|\boldsymbol{w}\|_p > d^{c+3p/2}$.

Then, $(B^{\otimes c}, k^c)$ is a YES (resp. NO) instance of γ^c-SVP_p if (B, k) is a YES (resp. NO) instance of γ-SVP_p, where $B^{\otimes c}$ denotes the c-fold tensor product of B with itself.

We will require some auxiliary lemmas in order to prove Lemma 5.1, starting with a version of Minkowski's first theorem for (possibly non-full-rank) lattices in the ℓ_2 norm.

Lemma B.1 (Minkowski's first theorem)**.** *Let $\mathcal{L}$ be a rank-r lattice. Then, it holds that*

$$\det(\mathcal{L}) \geqslant \left(\frac{\lambda_1^{(2)}(\mathcal{L})}{\sqrt{r}}\right)^r.$$

We will also need to relate ℓ_p norms to the ℓ_2 norm. When $p \geqslant 2$, a standard application of Hölder's inequality yields

$$\|\boldsymbol{v}\|_2 \geqslant \|\boldsymbol{v}\|_p \geqslant |\mathsf{supp}(\boldsymbol{v})|^{1/p-1/2}\|\boldsymbol{v}\|_2,$$

where $\mathsf{supp}(\boldsymbol{v}) = \{i \in [m] : \boldsymbol{v}_i \neq 0\}$ is the support of $\boldsymbol{v}$, for any vector $\boldsymbol{v} \in \mathbb{R}^m$. When $p < 2$ we have that

$$\|\boldsymbol{v}\|_p \geqslant \|\boldsymbol{v}\|_2.$$

We can combine the inequalities above to conclude in particular that

$$\|\boldsymbol{v}\|_p \geqslant |\mathsf{supp}(\boldsymbol{v})|^{1/\max(2,p)-1/2}\|\boldsymbol{v}\|_2 \tag{17}$$

for all $p \geqslant 1$. Moreover, using the fact that $\|\boldsymbol{v}\|_1 \leqslant |\mathsf{supp}(\boldsymbol{v})|^{1/2}\|\boldsymbol{v}\|_2$ by Cauchy-Schwarz and that $\|\boldsymbol{v}\|_1 \geqslant \|\boldsymbol{v}\|_p$ for all $p \geqslant 1$, it holds that

$$\|\boldsymbol{v}\|_2 \geqslant |\mathsf{supp}(\boldsymbol{v})|^{-1/2}\|\boldsymbol{v}\|_p \tag{18}$$

for all $p \geqslant 1$.

We prove a generalization of [HR12, Claim 3.5] for ℓ_p norms with $p \geqslant 1$.

Lemma B.2 (Generalization of [HR12, Claim 3.5]). *Let (B, k) be a NO instance of γ-SVP$_p$ with the properties outlined in Lemma 5.1, and let $\mathcal{L}$ be a sublattice of $\mathcal{L}(B)$ of rank r. Then, at least one of the following properties holds, where $d = \gamma k$:*

- *Every basis matrix of $\mathcal{L}$ has more than d^p nonzero rows;*
- *Every basis matrix of $\mathcal{L}$ has only even entries and has more than $(d/2)^p$ nonzero rows;*
- $\det(\mathcal{L}) > d^{r(c+p/2)}$ *and there is a basis matrix of $\mathcal{L}$ which has at most $(d/2)^p$ nonzero rows. In particular, every vector in $\mathcal{L}$ has Hamming weight at most $(d/2)^p$.*

Proof. We assume that the first two properties do not hold and show that the third property holds in that case. Since there is a basis matrix of $\mathcal{L}$ with at most d^p nonzero rows, we conclude that $r \leqslant d^p$ and that every vector in $\mathcal{L}$ has Hamming weight at most d^p. By the properties of (B, k), this implies that $\mathcal{L} \subseteq 2\mathbb{Z}^m$. Therefore, it must be the case that there is a basis matrix of $\mathcal{L}$ that has at most $(d/2)^p$ nonzero rows, and so every vector in $\mathcal{L}$ has Hamming weight at most $(d/2)^p$. As a result, we also gather that all nonzero vectors $\boldsymbol{w} \in \mathcal{L}$ satisfy $\|\boldsymbol{w}\|_p > d^{c+3p/2}$, and so

$$\det(\mathcal{L}) \geqslant \left(\frac{\lambda_1^{(2)}(\mathcal{L})}{\sqrt{r}}\right)^r \geqslant \left(\frac{d^{-p/2}\lambda_1^{(p)}(\mathcal{L})}{\sqrt{r}}\right)^r > d^{r(c+p/2)},$$

where the first inequality follows from Lemma B.1, the second inequality holds by Equation (18) and the fact that every vector in $\mathcal{L}$ has Hamming weight at most d^p, and the third inequality uses the fact that $r \leqslant d^p$. □

We will also need the following technical lemma from [HR12], which is specific for the ℓ_2 norm.

Lemma B.3 ([HR12, Claim 3.6]). *Let $\mathcal{L}_1$ and $\mathcal{L}_2$ be integer lattices of rank $r \geqslant 1$ generated by the bases $U = (\boldsymbol{u}_1, \ldots, \boldsymbol{u}_r)$ and $W = (\boldsymbol{w}_1, \ldots, \boldsymbol{w}_r)$, respectively. Consider the vector $\boldsymbol{v} = \sum_{i=1}^r \boldsymbol{u}_i \otimes \boldsymbol{w}_i \in \mathcal{L}_1 \otimes \mathcal{L}_2$. Then, it holds that*

$$\|\boldsymbol{v}\|_2 \geqslant \sqrt{r}(\det(\mathcal{L}_1) \cdot \det(\mathcal{L}_2))^{1/r}.$$

We are now ready to prove Lemma 5.1.

Proof of Lemma 5.1. It suffices to show that if (B, k) is a NO instance of γ-SVP$_p$ satisfying the conditions from the lemma statement, $\mathcal{L}_1 = \mathcal{L}(B)$, and $\mathcal{L}_2 = \bigotimes_{i=1}^{c'} \mathcal{L}$ for some integer $1 \leqslant c' < c$ and such that $\lambda_1^{(p)}(\mathcal{L}_2) > d^{c'}$, then

$$\lambda_1^{(p)}(\mathcal{L}_1 \otimes \mathcal{L}_2) > d^{c'+1}.$$

Consider an arbitrary nonzero vector $\boldsymbol{v} \in \mathcal{L}_1 \otimes \mathcal{L}_2$. As shown in [HR12, Proof of Lemma 3.4], we can write $\boldsymbol{v} = B_1'(B_2')^T$ for full-column-rank matrices B_1' and B_2' (note that B_1' and B_2' have the same number of columns) such that $\mathcal{L}_i' = \mathcal{L}(B_i') \subseteq \mathcal{L}_i$ for $i = 1, 2$. We now proceed by case analysis based on Lemma B.2 applied to $\mathcal{L}_1'$:

- B_1' has more than d^p nonzero rows: In this case, more than d^p rows of $B_1'(B_2')^T$ are nonzero vectors from $\mathcal{L}_2'$, and so

$$\|\boldsymbol{v}\|_p > d \cdot \lambda_1^{(p)}(\mathcal{L}_2') \geqslant d \cdot \lambda_1^{(p)}(\mathcal{L}_2) > d^{c'+1}.$$

- B_1' has only even entries and has more than $(d/2)^p$ nonzero rows: In this case, more than $(d/2)^p$ rows of $B_1'(B_2')^T$ are even multiples of nonzero vectors from $\mathcal{L}_2'$, and so

$$\|\boldsymbol{v}\|_p > 2((d/2)^p)^{1/p}\lambda_1^{(p)}(\mathcal{L}_2') \geqslant d \cdot \lambda_1^{(p)}(\mathcal{L}_2) > d^{c'+1}.$$

- $\det(\mathcal{L}_1') > d^{r(c+p/2)}$ and B_1' has at most $(d/2)^p$ nonzero rows. In particular, $r \leqslant (d/2)^p$ and every vector in $\mathcal{L}_1'$ has Hamming weight at most $(d/2)^p$: In this case, we have

$$\begin{aligned}
\|\boldsymbol{v}\|_p &\geqslant d^{\frac{p}{\max(2,p)}-\frac{p}{2}}\|\boldsymbol{v}\|_2 \\
&\geqslant d^{\frac{p}{\max(2,p)}-\frac{p}{2}}\sqrt{r}(\det(\mathcal{L}_1')\cdot\det(\mathcal{L}_2'))^{1/r} \\
&\geqslant d^{\frac{p}{\max(2,p)}-\frac{p}{2}}\sqrt{r}\det(\mathcal{L}_1')^{1/r} \\
&> d^{\frac{p}{\max(2,p)}-\frac{p}{2}}\cdot d^{c+p/2} \\
&\geqslant d^{c'+1}\ .
\end{aligned}$$

  The first inequality follows from Equation (17) and the fact that $\boldsymbol{v}$ has support size at most d^p. The second inequality holds via Lemma B.3 applied to $\mathcal{L}_1'$ and $\mathcal{L}_2'$, which are both rank-r lattices. The third inequality uses the fact that $\det(\mathcal{L}_2') \geqslant 1$, which holds because $\mathcal{L}_2'$ is a non-trivial integer lattice (this in turn holds by the definition of $\mathcal{L}_2'$ and the fact that $\boldsymbol{v} \neq \mathbf{0}$.). The fourth inequality holds by the lower bound on $\det(\mathcal{L}_1')$ and the fact that $c' \leqslant c-1$. □