

# On the Hardness of the One-Sided Code Sparsifier Problem

Elena Grigorescu ✉ 

David R. Cheriton School of Computer Science, University of Waterloo, Canada

Alice Moayyedi ✉ 

David R. Cheriton School of Computer Science, University of Waterloo, Canada

---

## Abstract

---

The notion of *code sparsification* was introduced by Khanna, Putterman and Sudan (SODA 2024) as an analogue to the more established notion of cut sparsification in graphs and hypergraphs. In particular, for  $\alpha \in (0, 1)$ , an (unweighted) one-sided  $\alpha$ -sparsifier for a linear code  $\mathcal{C} \subseteq \mathbf{F}_2^n$  is a subset  $S \subseteq [n]$  such that the weight of each codeword projected onto the coordinates in  $S$  is preserved up to an  $\alpha$  fraction. Recently, Gharan and Sahami (arXiv:2502.02799) show the existence of one-sided  $\frac{1}{2}$ -sparsifiers of size  $n/2 + O(\sqrt{kn})$  for any linear code, where  $k$  is the dimension of  $\mathcal{C}$ . In this paper, we consider the computational problem of finding a one-sided  $\frac{1}{2}$ -sparsifier of minimal size, and show that it is NP-hard, via a reduction from the classical nearest codeword problem. We also show hardness of approximation results.

**2012 ACM Subject Classification** Theory of computation  $\rightarrow$  Error-correcting codes

**Keywords and phrases** Code sparsifiers, NP-hardness, Approximation hardness

**Digital Object Identifier** 10.4230/LIPIcs.STACS.2026.47

## 1 Introduction

For  $\alpha \in (0, 1)$ , a one-sided  $\alpha$ -sparsifier for a code  $\mathcal{C} \subseteq \mathbf{F}_2^n$  is a set  $S \subseteq [n]$  such that the projection of any codeword  $c \in \mathcal{C}$  onto  $S$  results in a vector  $c_S$  whose weight is preserved up to an  $\alpha$ -factor, namely that  $wt(c_S) \geq \alpha \cdot wt(c)$ . Here  $wt(c) = |\{i \mid c_i \neq 0\}|$ , the number of nonzero elements in  $c$ .

The notion of (weighted) two-sided code sparsifiers was recently introduced by Khanna, Putterman, and Sudan [7], as an analogue of the notion of cut sparsifiers for graphs and hypergraphs [6], which have been studied even more broadly in the context of constraint satisfaction problems [9]. Recently, Gharan and Sahami [5] study the unweighted one-sided code sparsifier problem for linear codes (i.e. subspaces). They give an elegant short proof of the existence of one-sided  $\frac{1}{2}$ -sparsifiers of size  $n/2 + O(\sqrt{kn})$ , where  $k = \log |\mathcal{C}|$  is the dimension of  $\mathcal{C}$ .

Here we study the computational problem of *finding* one-sided unweighted  $\frac{1}{2}$ -sparsifiers, defined as follows.

Minimal One-Sided  $\frac{1}{2}$ -Sparsifier Problem (OPTHALFSPARSIFIER)

**Instance:** A linear code  $\mathcal{C}$ , given by its generators.

**Output:** A set  $S \subseteq [n]$  such that:

- (Feasibility) for all  $c \in \mathcal{C}$ ,  $wt(c_S) \geq \frac{1}{2}wt(c)$ ;
- (Optimality)  $S$  is of smallest size among all sets that satisfy the above.

To the best of our knowledge, the complexity of finding minimum-sized sparsifiers has not been studied before. We show the following hardness results.

► **Theorem 1** (Hardness of OPTHALFSPARSIFIER). *OPTHALFSPARSIFIER is NP-hard.*

In fact, our results hold more generally:

► **Theorem 2** (Approximation Hardness of OPTHALFSPARSIFIER). *Let  $S^* \subseteq [n]$  be a one-sided  $\frac{1}{2}$ -sparsifier of minimal size for a linear code  $\mathcal{C}$ . The problem of finding a one-sided  $\frac{1}{2}$ -sparsifier  $S \subseteq [n]$  for  $\mathcal{C}$  such that  $\gamma \cdot |\bar{S}| \geq |\bar{S}^*|$  (where  $\bar{S} = [n] \setminus S$ ) is:*

- NP-hard for any constant  $\gamma \geq 1$ ;
- with  $\gamma = 2^{\log^{1-\epsilon} n}$  for any constant  $\epsilon > 0$ , impossible to solve in polynomial time assuming  $\text{NP} \not\subseteq \text{DTIME}(2^{\log^{O(1)} n})$ ;
- with  $\gamma = n^{c/\log \log n}$  for some constant  $c > 0$ , impossible to solve in polynomial time assuming  $\text{NP} \not\subseteq \bigcap_{\delta > 0} \text{DTIME}(2^{n^\delta})$ .

Our proofs show Turing reductions from the fundamental problem of computing a nearest codeword to a received string, as defined below.

#### Nearest Codeword Problem (NCP)

**Instance:** A linear code  $\mathcal{C}$  given by its generators, a received string  $s \in \mathbf{F}_2^n$ , and an integer  $k$ .

**Output:** (YES) if there exists  $c \in \mathcal{C}$  such that  $\text{wt}(c + s) \leq k$ , and (NO) otherwise.

The NP-hardness of the nearest codeword problem was first shown by Vardy [11], followed by proofs by Dumer, Micciancio, and Sudan [4] of the hardness of the problem for promise additive and multiplicative approximation versions, under RUR reductions. A sequence of follow-ups [8, 1, 10, 3, 2] have now established that the multiplicative approximation variant is NP-hard under deterministic Karp reductions.

## 1.1 Preliminaries

Let  $\mathcal{C}$  be a linear code over  $\mathbf{F}_2^n$ ; that is,  $\mathcal{C}$  is a subset of  $\mathbf{F}_2^n$  such that  $c, c' \in \mathcal{C} \Rightarrow c + c' \in \mathcal{C}$ . We may define a linear code  $\mathcal{C}$  as the span of the columns of a matrix  $M$ ; in this case, we call  $M$  the *generator matrix* of  $\mathcal{C}$ , and we call the columns of  $M$  the *generators* of  $\mathcal{C}$ . We say that  $\mathcal{C}$  has dimension  $k$  if  $|\mathcal{C}| = 2^k$ . For  $h \in \mathbf{F}_2^n$ ,  $h + \mathcal{C}$  is called an *affine subspace*, or a *coset* of  $\mathcal{C}$  in  $\mathbf{F}_2^n$ .

We denote by  $\mathbf{0}$  and  $\mathbf{1}$  the all-zeroes and the all-ones vectors, respectively, in  $\mathbf{F}_2^n$ .

We define the *weight* of a string  $s \in \mathbf{F}_2^n$ ,  $\text{wt}(s)$ , as the number of nonzero coordinates of  $s$ . For a set of coordinates  $S \subseteq [n]$  and a string  $c \in \mathbf{F}_2^n$ , we define  $c_S$  as the projection of  $c$  onto the coordinates in  $S$ .

► **Definition 3.** *A set  $S \subseteq [n]$  is  $\alpha$ -thin with respect to  $\mathcal{C}$  if for every codeword  $c \in \mathcal{C}$ ,*

$$\text{wt}(c_S) \leq \alpha \cdot \text{wt}(c).$$

Likewise, we call a set  $S$   $\alpha$ -thick with respect to  $\mathcal{C}$  if  $\text{wt}(c_S) \geq \alpha \cdot \text{wt}(c)$  for all  $c \in \mathcal{C}$ . If we identify the set  $S$  with its indicator vector  $s \in \mathbf{F}_2^n$ , the weight  $\text{wt}(c \circ s)$  is equal to the weight  $\text{wt}(c_S)$ , where  $\circ$  is the Schur or element-wise product. We say that a string  $s$  is  $\alpha$ -thin or  $\alpha$ -thick with respect to a code  $\mathcal{C}$  exactly when its corresponding set  $S$  is. Note that the complement of an  $\alpha$ -thin set or string is a  $(1 - \alpha)$ -thick set or string, and vice versa. We denote by  $\bar{S}$  the complement of the set  $S$  with respect to  $[n]$ .

The terminology  $\alpha$ -thin is by analogy to  $\alpha$ -thinness in the context of graphs. An  $\alpha$ -thin subgraph (usually, a tree) of a graph  $G$  is a subgraph  $T \subseteq G$  such that, for any cut  $\delta(S)$  of  $G$ , the number of edges in  $T$  which are in  $\delta(S)$  is at most an  $\alpha$  fraction of the total number of edges in  $\delta(S)$ . This notion corresponds directly to the above definition of  $\alpha$ -thinness in linear codes via the following relation:

Given a graph  $G = (V, E)$ , we define the incidence matrix  $M$  as a  $|V| \times |E|$  matrix such that for  $v \in V, e \in E$ ,

$$M_{i,e} = \begin{cases} 1 & \text{if } v \in e \\ 0 & \text{if } v \notin e \end{cases}$$

That is,  $M_{v,e} = 1$  exactly when  $e$  is incident on  $v$ , and  $M_{v,e} = 0$  otherwise. If we use  $M$  as the generator matrix for a linear code  $C(M)$ , then the  $\alpha$ -thin subgraphs of  $G$  correspond to the  $\alpha$ -thin sets of  $E$  with respect to  $C(M)$ . If we define  $\alpha$ -thick subgraphs as the complements of  $\alpha$ -thin subgraphs, these also correspond to the  $\alpha$ -thick sets of  $E$ .

There is also a direct correspondence between this notion of  $\alpha$ -thickness and the notion of a one-sided  $\alpha$ -sparsifier. If a set is  $\alpha$ -thick with respect to a code  $\mathcal{C}$ , it is also a one-sided  $\alpha$ -sparsifier for that code, and vice versa.

In Section 3, we use the following hardness of approximation theorem for the nearest codeword problem [11, 4, 8, 1, 10, 3, 2].

► **Theorem 4.** *Given an affine subspace  $V \subseteq \mathbf{F}_2^n$  and an integer  $k > 0$ , there is no polynomial-time algorithm which distinguishes between the following cases:*

- (YES): *there exists  $x \in V$  with  $\text{wt}(x) \leq k$ .*
- (NO): *for all  $x \in V$ , it is the case that  $\text{wt}(x) \geq \gamma \cdot k$ .*

1. *when  $\gamma > 1$  is a constant, assuming  $P \neq \text{NP}$*
2. *when  $\gamma = 2^{\log^{1-\epsilon} n}$ , for any  $\epsilon$ , assuming  $\text{NP} \not\subseteq \text{DTIME}(2^{\log^{O(1)} n})$*
3. *when  $\gamma = n^{c/\log \log n}$ , for some  $c > 0$ , assuming  $\text{NP} \not\subseteq \bigcap_{\delta > 0} \text{DTIME}(2^{n^\delta})$ .*

## 1.2 Organization

In Section 2 we give a structural theorem of  $\frac{1}{2}$ -thick sets. We use this to show that the problem of finding  $\frac{1}{2}$ -thick sets is strongly related to the nearest codeword problem. In Section 3 we reduce NCP to the problem of finding optimal  $\frac{1}{2}$ -thick sets (equivalently, optimal one-sided  $\frac{1}{2}$ -sparsifiers) through this relationship, proving the main theorems.

## 2 Representatives of Largest Weight

In this section we characterise  $\frac{1}{2}$ -thick sets, and connect them to the nearest codewords of elements in the same coset.

For a linear code  $\mathcal{C} \subseteq \mathbf{F}_2^n$ , we consider the quotient space  $\mathbf{F}_2^n/\mathcal{C}$ . The equivalence classes or cosets  $H \in \mathbf{F}_2^n/\mathcal{C}$  are the sets of strings such that  $h, h' \in H \Rightarrow h + h' \in \mathcal{C}$  and  $h \in H, s \notin H \Rightarrow h + s \notin \mathcal{C}$ . For a given coset  $H$ , we define the set  $H^* \subseteq H$  to be the set of strings in  $H$  of largest weight;  $H^* = \{h^* \in H : \text{wt}(h^*) = \max_{h \in H} \text{wt}(h)\}$ . Our first theorem is a characterisation of the  $\frac{1}{2}$ -thick strings in a given coset:

► **Theorem 5.** *For a member  $h$  of a coset  $H \in \mathbf{F}_2^n/\mathcal{C}$ , the following are equivalent:*

1.  *$h$  is among the elements of  $H$  of greatest weight;*
2.  *$h$  is  $\frac{1}{2}$ -thick with respect to  $\mathcal{C}$ ;*
3. *The all-zeroes string,  $\mathbf{0}$ , is a nearest codeword to the complement of  $h$ ,  $\bar{h}$  (that is,  $h + \mathbf{1}$ ).*

We prove Theorem 5 in two parts: Lemma 6 (equivalence of 1. and 2.) and Lemma 7 (equivalence of 1. and 3.).

## 47:4 On the Hardness of the One-Sided Code Sparsifier Problem

We say that  $c \in \mathcal{C}$  is a nearest codeword to a string  $s \in \mathbf{F}_2^n$  when the Hamming distance between  $s$  and  $c$ ,  $\text{wt}(s + c)$ , is minimal among all elements of  $\mathcal{C}$ . In general, we denote the complement of a string  $s$  as  $\bar{s}$ , and the all-zeroes and all-ones strings as  $\mathbf{0}$  and  $\mathbf{1}$  respectively. Note that  $\bar{\bar{s}} = s + \mathbf{1}$ .

► **Lemma 6.** *1. and 2. above are equivalent.*

**Proof.** Oveis Gharan and Sahami [5] present a proof that 1. implies 2., which we will briefly reproduce here for completeness. Suppose that  $h^* \in H^*$  is a string of maximal weight among strings in  $H$ , and suppose that  $h^*$  is not  $\frac{1}{2}$ -thick with respect to  $\mathcal{C}$ ; that is,  $\exists c \in \mathcal{C}$  such that  $\text{wt}(c \circ h^*) < \frac{1}{2}\text{wt}(c)$ . Take the string  $h' = h^* + c$ , noting that  $h' \in H$ . We must have that  $\text{wt}(h') > \text{wt}(h^*)$ , since  $c \circ h^* < c \circ \bar{h}^*$  – adding  $c$  turns more coordinates of  $h^*$  to one than it turns to zero. This contradicts the assumption that  $h^*$  is of maximal weight among strings in  $H$ ; hence,  $h^*$  is  $\frac{1}{2}$ -thick with respect to  $\mathcal{C}$ .

Now, to show that 2. implies 1., suppose that  $h \in H$  is a  $\frac{1}{2}$ -thick string, and that there is some  $h^* \in H$  such that  $\text{wt}(h^*) > \text{wt}(h)$ . Then consider the codeword  $c = h^* + h$ , with  $c \in \mathcal{C}$ . Since  $c + h = h^*$ , and  $\text{wt}(h^*) > \text{wt}(h)$ , we must have that  $\text{wt}(c \circ h) < \text{wt}(c \circ \bar{h})$ ; hence,  $\text{wt}(c \circ h) < \frac{1}{2}\text{wt}(c)$ . Then  $h$  is not  $\frac{1}{2}$ -thick, a contradiction; any  $\frac{1}{2}$ -thick string must be of maximal weight among its equivalence class. ◀

► **Lemma 7.** *1. and 3. above are equivalent.*

**Proof.** To show that 1. implies 3., take some  $h^* \in H$  of maximal weight, and suppose that there exists some codeword  $c \in \mathcal{C}$  such that  $\text{wt}(c + \bar{h}^*) < \text{wt}(\bar{h}^* + \mathbf{0}) = \text{wt}(\bar{h}^*)$ . Then, noting that for any  $a \in \mathbf{F}_2^n$ ,  $\text{wt}(a) = n - \text{wt}(\bar{a}) = n - \text{wt}(a + \mathbf{1})$ , we have that:

$$\text{wt}(c + \bar{h}^*) < \text{wt}(\bar{h}^*) \quad \text{iff} \quad n - \text{wt}(\mathbf{1} + c + \bar{h}^*) < n - \text{wt}(h^*) \quad \text{iff} \quad \text{wt}(c + h^*) > \text{wt}(h^*).$$

Hence  $c + h^*$  is an element of  $H$  of greater weight than  $h^*$ , a contradiction. More intuitively, if  $\bar{h}^*$  is closer to  $c$  than to  $\mathbf{0}$ , it must be the case that it shares more coordinates with  $c$  than it has zero coordinates. Then  $h^*$  must differ on more coordinates with  $c$  than it has nonzero coordinates – so  $h^* + c$ , the string consisting of all coordinates in which  $h^*$  and  $c$  differ, must be of higher weight than  $h^*$  itself. So  $\mathbf{0}$  must be a nearest codeword in  $\mathcal{C}$  to  $\bar{h}^*$ .

To show that 3. implies 1., suppose that  $h \in H$  is of less than maximal weight; that is, there is some  $h^* \in H$  with  $\text{wt}(h^*) > \text{wt}(h)$ . Then  $c = h^* + h$  is a codeword in  $\mathcal{C}$  which is closer to  $\bar{h}$  than  $\mathbf{0}$ :

$$\text{wt}(\bar{h}^*) < \text{wt}(\bar{h}) \quad \text{iff} \quad \text{wt}((h^* + h) + \bar{h}) < \text{wt}(\bar{h}) \quad \text{iff} \quad \text{wt}(c + \bar{h}) < \text{wt}(\bar{h})$$

So  $\mathbf{0}$  is not a nearest codeword to  $\bar{h}$ ; if  $\mathbf{0}$  is a nearest codeword to some  $\bar{h}^*$  with  $h^* \in H$ ,  $h^*$  is of maximal weight among elements of  $H$ . ◀

This concludes the proof of Theorem 5.

The fundamental property at hand here is that the operation of shifting by some constant string is an isometry; once the nearest (or furthest) codewords of one element  $h$  of a coset  $H$  have been determined, the nearest codewords of every other element  $h'$  in  $H$  are fully determined by the difference between  $h$  and  $h'$ . Thus, since any element in the coset  $H$  must have some nearest codewords, there must be some elements of  $H$  which have any particular codeword as their nearest codeword – if  $h \in H$  has  $c$  as a nearest codeword,  $h + (c + c')$  has  $c'$  as a nearest codeword. The elements of  $\mathbf{1} + H$  which are closest to  $\mathbf{0}$  – corresponding to the elements of  $H$  which are furthest from  $\mathbf{0}$  – must therefore not merely be closer to  $\mathbf{0}$  than any other element of  $\mathbf{1} + H$ , but also closer to  $\mathbf{0}$  than they are to any other element of  $\mathcal{C}$ . This allows us to extract from  $H^*$  knowledge of the nearest codewords of every element of  $\mathbf{1} + H$ :

► **Theorem 8.** *Let  $H \in \mathbf{F}_2^n/\mathcal{C}$  be some equivalence class and  $H^*$  be the set of strings of maximal weight in  $H$ . Consider any string  $h \in H$  and any string  $h^* \in H^*$ . Their sum,  $h + h^*$ , is a codeword in  $\mathcal{C}$  of minimum Hamming distance from  $\bar{h}$ . Likewise, for any  $h$ , the codewords  $C$  of minimum Hamming distance from  $\bar{h}$  are of the form  $h + h^*$  for some  $h^* \in H^*$ .*

**Proof.** Take any  $h \in H$  and  $h^* \in H^*$ , and suppose that there exists some string  $c \in \mathcal{C}$  such that  $\text{wt}(\bar{h} + c) < \text{wt}(\bar{h} + (h + h^*))$ . Then:

$$n - \text{wt}(h + c) < n - \text{wt}(h^*) \quad \text{iff} \quad \text{wt}(h + c) > \text{wt}(h^*).$$

Since  $h + c$  is in  $H$ , and elements of  $H^*$  have maximum weight among elements of  $H$ , this is a contradiction; no such  $c$  can exist. Furthermore,  $\text{wt}(\bar{h} + (h + h^*)) = n - \text{wt}(h^*)$ , which is the same quantity for all  $h^* \in H^*$ ; hence, all codewords of the form  $h + h^*$  have equal and minimal Hamming distance from  $\bar{h}$ . This proves the first direction. For the second direction, suppose that there exists a codeword  $c \in \mathcal{C}$  of minimum Hamming distance from  $\bar{h}$ ; that is,  $\text{wt}(\bar{h} + c) \leq \text{wt}(\bar{h} + (h + h^*))$  for any  $h^* \in H^*$ . By the first direction, this is an equality;  $\text{wt}(\bar{h} + c) = \text{wt}(\bar{h} + (h + h^*))$ , since if  $\text{wt}(\bar{h} + c) < \text{wt}(\bar{h} + (h + h^*))$  then  $h + h^*$  would not be a nearest codeword to  $\bar{h}$ . Then  $\text{wt}(h + c) = \text{wt}(h^*)$ ; as  $h + c \in H$ , it must be the case that  $h + c \in H^*$ . So  $c = h + (h + c) = h + h^*$  for some  $h^* \in H^*$ . This proves the second direction.

Alternatively, we may observe that since  $\mathbf{0}$  is a nearest codeword to  $\bar{h}^*$ ,  $\mathbf{0} + c = c$  must be a nearest codeword to  $\bar{h}^* + c$ ; that is, if  $c = h^* + h$ , then  $c$  is a nearest codeword to  $\bar{h}^* + (h^* + h) = \bar{h}$ . The inverse direction follows straightforwardly by reversing the argument. ◀

This also demonstrates that, for any elements  $h, h'$  in some coset  $H$  of  $\mathbf{F}_2^n/\mathcal{C}$ , the nearest codewords in  $\mathcal{C}$  to  $h$  are the same distance from  $h$  as the nearest codewords to  $h'$ . We can therefore talk about the “distance” of a coset from the code; the distance of a coset  $H$  to  $\mathcal{C}$  is the distance from any element of  $H$  to its nearest codewords in  $\mathcal{C}$ . Since  $\mathbf{0}$  is always a codeword in any linear code, and the elements of minimum weight in a given coset must have  $\mathbf{0}$  as a nearest codeword, the distance of any coset is equal to the minimum weight among elements in that coset,  $\text{minwt}(H)$ .

Since the  $\frac{1}{2}$ -thick strings in a coset directly correspond to the nearest codewords of elements in that coset, we have the following:

► **Corollary 9.** *Given a linear code  $\mathcal{C} \subseteq \mathbf{F}_2^n$ , and integer  $k$ , the problem of determining whether a string in a given coset of  $\mathcal{C}$  in  $\mathbf{F}_2^n$  exists of weight at least  $k$ , or equivalently whether the weight of the  $\frac{1}{2}$ -thick elements in a given coset are at least  $k$ , is NP-complete, under deterministic Karp reductions.*

**Proof.** This problem is obviously in NP; we can certify any (YES) instance with a string of weight at least  $k$  in the given coset. We show NP-hardness by reduction from the nearest codeword problem, as defined in Section 1.

Let COSETHEAVY be the problem above:

Heaviest Element Problem (COSETHEAVY)

**Instance:** A generator matrix  $M$  for a linear code  $\mathcal{C}$ , a string  $h$  in an equivalence class  $H \in \mathbf{F}_2^n/\mathcal{C}$ , and an integer  $k$ .

**Output:** (YES) if there exists  $h^* \in H$  such that  $\text{wt}(h^*) \geq k$ , and (NO) otherwise.

We define the mapping from NCP to COSETHEAVY instances as follows:

$$(M, s, k) \rightarrow (M, \bar{s}, n - k)$$

This map can obviously be computed in polynomial time.

Suppose that the NCP instance  $(M, s, k)$  is a (YES) instance; there exists some  $c \in \mathcal{C}$ , with  $\mathcal{C}$  the linear code generated by  $M$ , such that  $\text{wt}(c + s) \leq k$ . Then, specifically, the distance between  $c + s$  and  $\mathbf{0} - \text{wt}(c + s + \mathbf{0}) -$  is also at most  $k$ . So, following Theorem 8, there exists a string in the coset containing  $s$  of  $\mathcal{C}$  in  $\mathbf{F}_2^n$  with weight at most  $k$ ; there exists a string in the coset containing  $\bar{s}$  with weight at least  $n - k$ . So the COSETHEAVY instance  $(M, \bar{s}, n - k)$  is a (YES) instance.

Now suppose that the COSETHEAVY instance is a (YES) instance. Then there is a string in the coset  $H$  containing  $s$  with weight at most  $k$ , and for any element of  $H$  there is a codeword in  $\mathcal{C}$  of distance at most  $k$ ; the NCP instance is a (YES) instance.

Thus,  $\text{NCP} \leq_p \text{COSETHEAVY}$ , and COSETHEAVY is NP-complete. ◀

Note that this reduction is in fact surjective, and  $\text{NCP} \equiv_p \text{COSETHEAVY}$ .

### 3 Optimal One-Sided Sparsifiers (Proof of Main Theorems)

Of course, in order to sparsify a code, we are not actually interested in finding the  $\frac{1}{2}$ -thick strings among a particular coset. Instead, we are interested in finding the  $\frac{1}{2}$ -thick strings among all strings in  $\mathbf{F}_2^n$ . Specifically, we are interested in the problem of finding the smallest strings which are  $\frac{1}{2}$ -thick. We call a string an *optimal*  $\alpha$ -thick (thin) string with respect to a code  $\mathcal{C}$  if it is  $\alpha$ -thick (thin) with respect to  $\mathcal{C}$  and it is of least (resp. greatest) weight among all  $\alpha$ -thick (thin) strings. Similarly, we call a set an optimal  $\alpha$ -thick set if it is of smallest size among such sets.

► **Corollary 10.** *The optimal  $\frac{1}{2}$ -thick strings with respect to a code  $\mathcal{C}$  are exactly the complements of the strings of greatest weight which have  $\mathbf{0}$  as a nearest codeword in  $\mathcal{C}$ .*

**Proof.** By Theorem 5, the  $\frac{1}{2}$ -thick strings with respect to a code  $\mathcal{C}$  are exactly the complements of the strings which have  $\mathbf{0}$  as a nearest codeword – the  $\frac{1}{2}$ -thin strings with respect to  $\mathcal{C}$ . Further, the  $\frac{1}{2}$ -thick strings of least weight are the complements of the  $\frac{1}{2}$ -thin strings of greatest weight. ◀

In fact, since the distance to the nearest codeword is constant among all elements of a given coset of  $\mathcal{C}$  in  $\mathbf{F}_2^n$ , the cosets containing the optimal  $\frac{1}{2}$ -thin strings are those where every element is of greatest distance to their nearest codewords. The cosets containing the optimal  $\frac{1}{2}$ -thick strings, then, are cosets obtained by adding  $\mathbf{1}$  to the cosets containing the optimal  $\frac{1}{2}$ -thin strings (note that in a code containing the codeword  $\mathbf{1}$ , the cosets containing the optimal  $\frac{1}{2}$ -thin strings are the same as the cosets containing the optimal  $\frac{1}{2}$ -thick strings).

► **Theorem 1 (Hardness of OPTHALFSPARSIFIER).** *OPTHALFSPARSIFIER is NP-hard.*

**Proof.** We demonstrate this by polynomial-time Turing reduction from NCP. We restate the problem for convenience:

Minimal One-Sided  $\frac{1}{2}$ -Sparsifier Problem (OPTHALFSPARSIFIER)

**Instance:** A linear code  $\mathcal{C}$  given by its generators.

**Output:** A set  $S \subseteq [n]$  such that:

- (Feasibility) for all  $c \in \mathcal{C}$ ,  $\text{wt}(c_S) \geq \frac{1}{2}|c|$ ;
- (Optimality)  $S$  is of smallest size among all sets which satisfy the above.

■ **Algorithm 1** Solving NCP using OPTHALFSPARSIFIER.

---

**Input** : A generator matrix  $M$  for a linear code  $\mathcal{C}$ ;  
 a received string  $s \in \mathbf{F}_2^n$ ;  
 an integer  $k$ ;  
 a subroutine  $\text{ohs}(M')$  which solves OPTHALFSPARSIFIER, with output in the form of the indicator vector of the produced set.

**Output** : Whether the NCP instance  $(M, s, k)$  is a (YES) instance or a (NO) instance.

$M_0 \leftarrow M$   
 $i \leftarrow 0$

1 **while** a decision has not been made **do**

Let  $\mathcal{C}_i$  denote the linear code generated by the matrix  $M_i$ .  
 $h_i^* \leftarrow \text{ohs}(M_i)$

2 **if**  $n - \text{wt}(h_i^*) \leq k$  **then**

Output (YES).

3 **if**  $\bar{h}_i^* + s \in \mathcal{C}_i$  **then**

Output (NO).  
 $a_i \leftarrow \bar{h}_i^* + s$   
 $M_{i+1} \leftarrow (a_i \text{ concatenated to } M_i)$   
 $i \leftarrow i + 1$

---

We use Algorithm 1. This algorithm terminates in at most  $n - \dim(\mathcal{C})$  calls to the OPTHALFSPARSIFIER subroutine, with  $O(n - k)$  extra runtime: since  $a_i \notin \mathcal{C}_i$  for all  $i$ , the dimension of  $\mathcal{C}_{i+1}$  is one greater than the dimension of  $\mathcal{C}_i$  for all  $i$ . When the dimension of  $\mathcal{C}_i$  is  $n$ , then it must be the case that  $s \in \mathcal{C}_i$ ; thus, the loop at line 1 is run at most  $n - \dim(\mathcal{C})$  times.

It remains to show correctness. We wish to maintain the invariant that, for each  $i$ , the distance from  $s$  to the nearest codeword in  $\mathcal{C}_i$  is no closer than the distance from  $s$  to the nearest codeword in  $\mathcal{C}$ . So, towards a proof by contradiction, suppose that there is some  $i$  such that  $\mathcal{C}_{i+1}$  has an element nearer to  $s$  than the nearest element in  $\mathcal{C}_i$ . Note that since  $\mathcal{C}_{i+1} = \mathcal{C}_i \cup (a_i + \mathcal{C}_i)$ , if  $\mathcal{C}_{i+1}$  has an element closer to  $s$  than all elements in  $\mathcal{C}_i$ , that element must be in  $a_i + \mathcal{C}_i$ . So we have:

$$\begin{aligned} \min_{\mathcal{C}_i} \text{wt}(s + \mathcal{C}_i) &> \min_{\mathcal{C}_i} \text{wt}(s + a_i + \mathcal{C}_i) \\ \text{iff } \min_{\mathcal{C}_i} \text{wt}(s + \mathcal{C}_i) &> \min_{\mathcal{C}_i} \text{wt}(s + \bar{h}_i^* + s + \mathcal{C}_i) \\ \text{iff } \min_{\mathcal{C}_i} \text{wt}(s + \mathcal{C}_i) &> \min_{\mathcal{C}_i} \text{wt}(\bar{h}_i^* + \mathcal{C}_i) \end{aligned}$$

Where, for  $S \subseteq \mathbf{F}_2^n$ ,  $\min_{\mathcal{C}_i} \text{wt}(S)$  is the smallest weight among elements of  $S$ . Thus, the smallest weight among elements of the coset  $s + \mathcal{C}_i$  is larger than the smallest weight among elements of the coset  $\bar{h}_i^* + \mathcal{C}_i$ . But then the largest element in the coset  $\bar{s} + \mathcal{C}_i$  is smaller than the largest element in the coset  $h_i^*$ . By Theorem 5, all largest elements of any coset are  $\frac{1}{2}$ -thick, and the largest element of  $\bar{s} + \mathcal{C}_i$  is  $\frac{1}{2}$ -thick; thus,  $h_i^*$  cannot be a  $\frac{1}{2}$ -thick string of minimal weight among all  $\frac{1}{2}$ -thick strings, a contradiction. So  $\mathcal{C}_{i+1}$  cannot have any elements nearer to  $s$  than the nearest elements in  $\mathcal{C}_i$ ; by induction,  $\mathcal{C}_i$  has no closer elements to  $s$  than does  $\mathcal{C}$  for any  $i$ .

We can see this property more intuitively by noting that adding a basis element to a linear code combines pairs of cosets which differ by that element; the maximal (minimal) elements in the resulting coset will be the maximal (minimal) elements among the two. Thus,

given that the coset containing  $\bar{h}^*$  contains a largest minimal element among any coset, the coset  $(\bar{h}^* + \mathcal{C}_i) \cup (\bar{s} + \mathcal{C}_i)$  of  $\mathcal{C}_{i+1}$  will contain the minimal element of  $\bar{s} + \mathcal{C}_i$  as a minimal element; repeatedly merging cosets in this manner never moves the coset containing  $s$  “closer” to the code.

Given this invariant, we proceed to show that the two conditionals on line 2 and line 3 can only be satisfied if the instance is a (YES) or (NO) instance respectively. We begin with line 2. Since  $\bar{h}_i^*$  is a string of greatest distance from any codeword in  $\mathcal{C}_i$ , and has  $\mathbf{0}$  as a nearest codeword, if  $\text{wt}(\bar{h}_i^*) = n - \text{wt}(h_i^*) \leq k$ , then the greatest distance from any string among elements in  $\mathcal{C}_i$  is at most  $k$ ; thus, there must be a codeword in  $\mathcal{C}_i$  at least that close to  $s$ . By the invariant above, there also must be a codeword in  $\mathcal{C}$  which is that close, and the NCP instance  $(M, s, k)$  is a (YES) instance. To show that the conditional on line 3 is satisfied only for (NO) instances, note that if  $\bar{h}_i^* + s \in \mathcal{C}_i$ , then  $s$  is in a coset with  $\bar{h}_i^*$ . This implies that the distance from  $s$  to its nearest codewords in  $\mathcal{C}_i$  is the same as that of  $\bar{h}_i^*$  to its nearest codewords; since  $\bar{h}_i^*$  has  $\mathbf{0}$  as a nearest codeword, the distance from  $s$  to its nearest codeword is exactly  $\text{wt}(\bar{h}_i^*) = n - \text{wt}(h_i^*)$ . Since the conditional on line 2 was not satisfied, then, there exists no codeword in  $\mathcal{C}_i$  which has distance at most  $k$  from  $s$ . Since  $\mathcal{C}_i \supseteq \mathcal{C}$ , there exists no codeword in  $\mathcal{C}$  of at most that distance, and the NCP instance  $(M, s, k)$  is a (NO) instance.

We therefore have a polynomial-time Turing reduction from NCP to OPTHALFSPARSIFIER, and OPTHALFSPARSIFIER is NP-hard. ◀

From known hardness of approximation results ([2]) of NCP, we can also derive similar results for the OPTHALFSPARSIFIER problem.

► **Theorem 2 (Approximation Hardness of OPTHALFSPARSIFIER).** *Let  $S^* \subseteq [n]$  be a one-sided  $\frac{1}{2}$ -sparsifier of minimal size for a linear code  $\mathcal{C}$ . The problem of finding a one-sided  $\frac{1}{2}$ -sparsifier  $S \subseteq [n]$  for  $\mathcal{C}$  such that  $\gamma \cdot |\bar{S}| \geq |\bar{S}^*|$  (where  $\bar{S} = [n] \setminus S$ ) is:*

- NP-hard for any constant  $\gamma \geq 1$ ;
- with  $\gamma = 2^{\log^{1-\epsilon} n}$  for any constant  $\epsilon > 0$ , impossible to solve in polynomial time assuming  $\text{NP} \not\subseteq \text{DTIME}(2^{\log^{O(1)} n})$ ;
- with  $\gamma = n^{c/\log \log n}$  for some constant  $c > 0$ , impossible to solve in polynomial time assuming  $\text{NP} \not\subseteq \bigcap_{\delta > 0} \text{DTIME}(2^{n^\delta})$ .

**Proof.** This follows directly from the algorithm above and the hardness of approximation of NCP given by Theorem 4. Suppose that instead of an algorithm which solves OPTHALFSPARSIFIER, we have an algorithm which solves the following approximation of OPTHALFSPARSIFIER:

$\gamma$ -Approximate Minimal One-Sided  $\frac{1}{2}$ -Sparsifier Problem (APPROXOHS $_\gamma$ )

**Instance:** A linear code  $\mathcal{C}$  given by its generators.

**Output:** A set  $S \subseteq [n]$  such that:

- (Feasibility) for all  $c \in \mathcal{C}$ ,  $\text{wt}(c_S) \geq \frac{1}{2}c$ ;
- (Approximate Optimality) if  $S^*$  is a set which satisfies the above feasibility condition, then  $\gamma \cdot |\bar{S}| \geq |\bar{S}^*|$

If  $\gamma = 1$ , then this problem is OPTHALFSPARSIFIER. Note that the multiplicative factor here is a constraint not on the size of the set itself, but the size of its conjugate; it is trivial to find a  $\frac{1}{2}$ -thick set with a size within a factor of 2 of that of the smallest  $\frac{1}{2}$ -thick set, by simply taking every coordinate which is represented among codewords in  $\mathcal{C}$ . It is easy to see from this that approximation up to a constant factor of the set size is trivial.

Given such a subroutine we can use Algorithm 1 to approximate NCP, solving the following problem:

Nearest Codeword Problem with Multiplicative Gap  $\gamma$  (MULTGAPNCP $_{\gamma}$ )

**Instance:** A generator matrix  $M$  for a linear code  $\mathcal{C}$ , a received string  $s \in \mathbb{F}_2^n$ , and an integer  $k$ .

**Output:** (YES) if there exists  $c \in \mathcal{C}$  such that  $\text{wt}(c + s) \leq k$ . (NO) if for every  $c \in \mathcal{C}$ ,  $\text{wt}(c + s) > \gamma k$ .

To show that this reduction goes through, we follow the previous proof with a small modification: instead of maintaining that the augmented codes  $\mathcal{C}_i$  each have no codewords closer to  $s$  than the closest codewords in  $\mathcal{C}$ , we maintain that each  $\mathcal{C}_i$  has no codewords outside of  $\mathcal{C}$  of distance smaller than  $k$  to  $s$ . We note first that if  $h_i^* + \mathcal{C}_i$  has a distance from  $\mathcal{C}_i$  at most  $k$ , then we will have answered (YES) during that iteration of the loop at line 1; we only proceed to add  $h_i^* + s$  to the generator matrix if  $h_i^* + \mathcal{C}_i$  is further from  $\mathcal{C}_i$  than  $k$ . Thus, the set  $(s + a_i + \mathcal{C}_i) = (h_i^* + \mathcal{C}_i)$  has a distance from  $\mathcal{C}$  no smaller than  $k$ . Formally,  $\min_{c \in \mathcal{C}} \text{wt}(s + a_i + c) = \min_{c \in \mathcal{C}_i} \text{wt}(h_i^* + c) > k$ .

The reduction from MULTGAPNCP $_{\gamma}$  to APPROXOHS $_{\gamma}$  then follows from the same arguments as in the proof of Theorem 1. Briefly, if the MULTGAPNCP $_{\gamma}$  instance is a (YES) instance, then it will certainly never be the case that the distance from  $s$  to  $\mathcal{C}_i$  is greater than  $k$  for any  $\mathcal{C}_i$ , so Algorithm 1 will output (YES); if the MULTGAPNCP $_{\gamma}$  instance is a (NO) instance, then, given the invariant above, Algorithm 1 may never output (YES), and must output (NO). ◀

---

## References

- 1 Per Austrin and Subhash Khot. A simple deterministic reduction for the gap minimum distance of code problem. *IEEE Trans. Inf. Theory*, 60(10):6636–6645, 2014. doi:10.1109/TIT.2014.2340869.
- 2 Vijay Bhattiprolu, Venkatesan Guruswami, Euiwoong Lee, and Xuandi Ren. Inapproximability of finding sparse vectors in codes, subspaces, and lattices, 2025. arXiv:2410.02636.
- 3 Vijay Bhattiprolu and Euiwoong Lee. Inapproximability of sparsest vector in a real subspace. *Electron. Colloquium Comput. Complex.*, TR24-151, 2024. URL: <https://eccc.weizmann.ac.il/report/2024/151>.
- 4 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. doi:10.1109/TIT.2002.806118.
- 5 Shayan Oveis Gharan and Arvin Sahami. Unweighted one-sided code sparsifiers and thin subgraphs, 2025. arXiv:2502.02799.
- 6 David R. Karger. Using randomized sparsification to approximate minimum cuts. In Daniel Dominic Sleator, editor, *Proceedings of the Fifth Annual ACM-SIAM Symposium on Discrete Algorithms. 23-25 January 1994, Arlington, Virginia, USA*, pages 424–432. ACM/SIAM, 1994. URL: <http://dl.acm.org/citation.cfm?id=314464.314582>.
- 7 Sanjeev Khanna, Aaron (Louie) Putterman, and Madhu Sudan. Code sparsification and its applications. In David P. Woodruff, editor, *Proceedings of the 2024 ACM-SIAM Symposium on Discrete Algorithms, SODA 2024, Alexandria, VA, USA, January 7-10, 2024*, pages 5145–5168. SIAM, 2024. doi:10.1137/1.9781611977912.185.
- 8 Subhash Khot. Hardness of approximating the shortest vector problem in lattices. *J. ACM*, 52(5):789–808, 2005. doi:10.1145/1089023.1089027.
- 9 Dmitry Kogan and Robert Krauthgamer. Sketching cuts in graphs and hypergraphs. In Tim Roughgarden, editor, *Proceedings of the 2015 Conference on Innovations in Theoretical*

## 47:10 On the Hardness of the One-Sided Code Sparsifier Problem

*Computer Science, ITCS 2015, Rehovot, Israel, January 11-13, 2015*, pages 367–376. ACM, 2015. doi:10.1145/2688073.2688093.

- 10 Daniele Micciancio. Locally dense codes. In *IEEE 29th Conference on Computational Complexity, CCC 2014, Vancouver, BC, Canada, June 11-13, 2014*, pages 90–97. IEEE, IEEE Computer Society, 2014. doi:10.1109/CCC.2014.17.
- 11 Alexander Vardy. The intractability of computing the minimum distance of a code. *IEEE Trans. Inf. Theory*, 43(6):1757–1766, 1997. doi:10.1109/18.641542.